CC23xx SimpleLink™ Wireless Microcontroller Unit

Technical Reference Manual

Literature Number: SWCU193A
APRIL 2023 – REVISED AUGUST 2024

Table of Contents

About This Manual

This document is organized into sections that correspond to each major feature; it explains the features and functionality of each module, and it also explains how to use them. For each feature, references are given to the documentation for the driver of the corresponding operating systems. This document does not contain performance characteristics of the device or modules, which are gathered in the corresponding device data sheets. This manual is intended for system software developers, hardware designers, and application developers.

Note

TI is transitioning to use more inclusive terminology. Some language may be different than what you would expect to see for certain technology areas.

Devices

The CC23xx device platform features different memory sizes, peripherals, and package options. All devices are centered around an Arm® Cortex®-M0+ series processor that handles the application layer and protocol stack.

Register, Field, and Bit Calls

The naming convention applied for a call consists of:

- For a register call: *<Module name>.<Register name>*; for example: UART.UASR
- For a bit field call:
  - *<Module name>.<Register name>[End:Start] <Field name> field*; for example, UART.UASR[4:0] SPEED bit field
  - *<Field name> field <Module name>.<Register name>[End:Start]*; for example, SPEED bit field UART.UASR[4:0]
- For a bit call:
  - *<Module name>.<Register name>[pos] <Bit name> bit*; for example, UART.UASR[5] BIT_BY_CHAR bit
  - *<Bit name> bit <Module name>.<Register name>[pos]*; for example, BIT_BY_CHAR bit UART.UASR[5]

Related Documentation

The following related documents are available on the CC23xx device product pages at www.ti.com:

1. CC2340R5:
   - CC2340R5 data sheet and errata (Technical Documents)

Note

This list of documents was current as of publication date. Check the website for additional documentation, application notes, and white papers.

Additional, related documentation follows:

1. The Institute of Electrical and Electronic Engineers, Inc., *IEEE Standard Test Access Port and Boundary Scan Architecture, IEEE Std 1149.1a 1993 and Supplement Std. 1149.1b 1994* (see IEEExplore.ieee.org)
2. The Institute of Electrical and Electronic Engineers, Inc., *IEEE 1149.7 Standard for Reduced-Pin and Enhanced-Functionality Test Access Port and Boundary-Scan Architecture* (see IEEExplore.ieee.org)
3. National Institute of Standards and Technology, *NIST Special Publication 800-38A, Recommendation for Block Cipher Modes of Operation Methods and Techniques* (see NIST.gov)
4. National Institute of Standards and Technology, *NIST Special Publication 800-38D, Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC* (see NIST.gov)
5. National Institute of Standards and Technology, *FIPS 197, Advanced Encryption Standard (AES)* (see NIST.gov)
6. Bluetooth SIG, Inc., *Bluetooth* Specification versions 4.0, 4.1, 4.2, and 5.3 (see Bluetooth.com)
7. *Cortex-M0+ Devices Generic User Guide* (see documentation at Arm.com)
8. *Cortex-M0+ Technical Reference Manual* (see documentation at Arm.com)
9. *Arm®v6-M Architecture Reference Manual* (see documentation at Arm.com)
10. *Arm® Debug Interface V5 Architecture Specification* (see documentation at Arm.com)

Trademarks

SimpleLink™ and SmartRF™ are trademarks of Texas Instruments.
CoreSight™ is a trademark of Arm Limited (or its subsidiaries) in the US and/or elsewhere.
Motorola™ is a trademark of Motorola, Inc..
Arm®, and Cortex®, and Thumb® are registered trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere.
Bluetooth® is a registered trademark of Bluetooth SIG, Inc..
Zigbee® is a registered trademark of Zigbee Alliance.
All trademarks are the property of their respective owners.

Chapter 1

Architectural Overview

The CC23xx SimpleLink™ ultra-low-power wireless MCUs provide solutions for a wide range of applications. To help the user develop these applications, this user's guide focuses on the use of the different building blocks of the devices. For detailed device descriptions, complete feature lists, and performance numbers, see the data sheet for the specific device. The following subsections provide easy access to relevant information and guide the reader to the different chapters in this document.

The CC23xx SimpleLink ultra-low-power wireless MCUs are optimized for ultra-low power while providing fast and capable MCU systems to enable short processing times and high integration. The combination of an Arm® Cortex®-M0+ processing core at 48MHz, flash memory, and a wide selection of peripherals makes the CC23xx specifically designed for single-chip implementation or network processor implementations of lower-power RF nodes.

1.1 Target Applications

The device is positioned for low-power wireless applications, such as:

- Consumer electronics
- Mobile phone accessories
- Sports and fitness equipment
- HID applications
- Home and building automation
- Grid infrastructure
  - Solar inverter
  - E-meter
- Lighting control
- Alarm and security
- Electronic shelf labeling
- Proximity tags
- Medical electronics
  - Glucose monitor
  - Heart rate sensor
- Remote controls
- Smart metering
- Asset tracking
- Wireless sensor networks
- Car access and security systems
  - Passive entry passive start (PEPS)
  - Remote keyless entry (RKE)
- Advanced driver assistance systems (ADAS)
- Tire pressure monitoring systems (TPMS)

1.2 Introduction

Figure 1-1 shows the building blocks of the CC23xx platform. The following sections provide an overview of the features of the CC23xx.

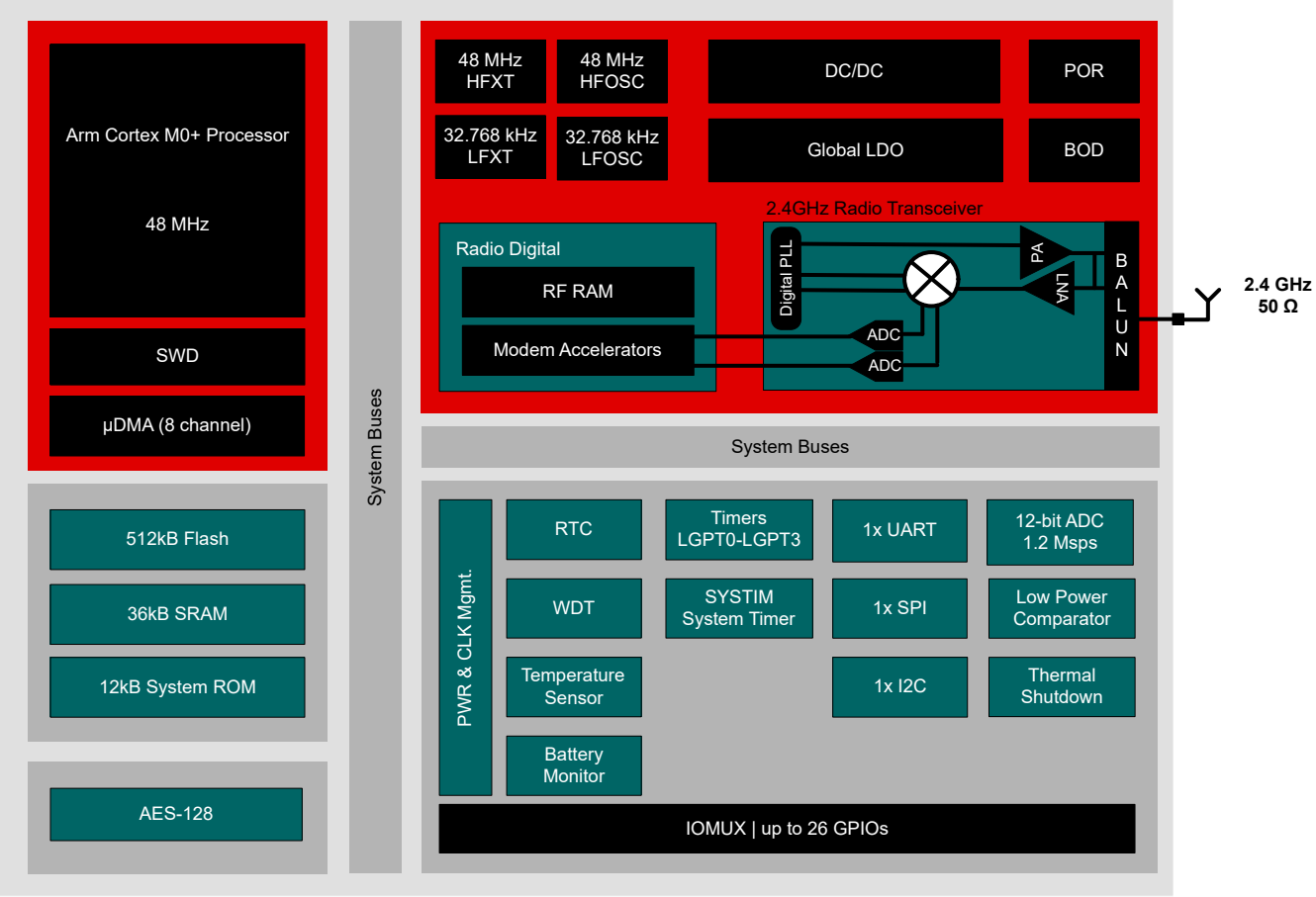


Figure 1-1. CC23xx Block Diagram

CC23xx devices have the following features:

- Arm® Cortex® M0+ processor core
  - Arm Cortex SysTick timer
  - Nested Vectored Interrupt Controller (NVIC)
- Clocks
  - 48MHz RC oscillator and 48MHz crystal oscillator
  - 32kHz crystal oscillator and 32kHz RC oscillator
- On-chip memory
  - Up to 512KB of in-system programmable flash
  - Up to 36KB of ultra-low leakage SRAM. Retained in standby mode
- Power Management
  - Wide supply voltage range
  - Efficient on-chip DC/DC converter for reduced power consumption
  - Flexible low-power modes allowing low energy consumption in duty-cycled applications
- Advanced Serial Integration
  - Universal asynchronous receiver-transmitter (UART)
  - Inter-Integrated Circuit (I^2C)

  - Serial peripheral interface (SPI)
- System Integration
  - Direct memory access controller (µDMA)
  - Up to four general-purpose timers capable of pulse width modulation (PWM), synchronization, capture, counting, and quadrature decoding
  - 32kHz real-time clock (RTC)
  - Watchdog timer
  - System Timer (SYSTIM) with the same time base as RTC but up to 250ns resolution
  - Battery Monitor: On-chip temperature and supply voltage sensing
  - GPIO with normal or high-drive capabilities
  - GPIO with analog capability for ADC and comparator
  - Configurable pin multiplexing
  - Low power comparator
  - 12-bit ADC, 1.2Mbps with external reference, 260kbps with internal reference, up to eight external ADC inputs
  - Thermal shutdown module
  - Security enablers
    - AES 128-bit cryptographic accelerator
    - Random number generator from on-chip analog noise
- Arm SWD debug interface
- 2.4GHz RF transceiver compatible with Bluetooth® 5.3 Low Energy and IEEE 802.15.4 PHY and MAC
- Integrated Balun
- Output power up to +8dBm with temperature compensation
- Wireless protocol support:
  - Bluetooth® 5.3 Low Energy
  - Zigbee®
  - SimpleLink™ TI 15.4 stack
  - Proprietary Systems
- For packaging options see the device-specific data sheet

For applications requiring extreme conservation of power, the CC23xx device features a power-management system to efficiently power down the device to a low-power state during extended periods of inactivity. A power-up and power-down sequencer, a 32-bit sleep timer (an RTC) with interrupt capabilities, and ultra-low-leakage (ULL) RAM with retention in all power modes position the MCU for battery applications. The CC23xx device platform offers the advantages of the widely available development tools of Arm, SoC infrastructure IP applications, and a large user community. Additionally, the microcontroller uses Arm Thumb®-compatible Thumb-2 instruction set to reduce memory requirements.

TI offers a complete support package to assist in getting to market quickly, with evaluation and development boards, white papers and application notes, an easy-to-use peripheral driver library, a software development kit (SDK) with qualified wireless protocols, and a strong support, sales, and distributor network.

1.3 Arm Cortex M0+

The following subsections provide an overview of the Arm Cortex M0+, the integrated system timer (SysTick), and the NVIC.

1.3.1 Processor Core

The CC23xx device is designed around an Arm Cortex M0+ processor core. The Arm Cortex M0+ processor is the core for a high-performance, low-cost platform that meets the needs of minimal memory implementation, reduced pin count, and low power consumption.

Features of the processor core are as follows:

- The Thumb instruction set combines high code density with 32-bit performance
- Integrated sleep modes for low power consumption
- Hardware multiplier

- Deterministic interrupt handling for time-critical applications
- Serial Wire Debug reduces the number of pins required for debugging.

1.3.2 SysTick Timer

The Arm Cortex M0+ processor includes an integrated Sytick Timer. SysTick provides a simple, 24-bit, clear-on-write, decrementing, wrap-on-zero counter with a flexible control mechanism. The counter can be used in several different ways, for example:

- A variable rate alarm or signal timer—the duration is range-dependent on the reference clock used and the dynamic range of the counter
- A simple counter used to measure time to completion and time used
- An internal clock-source control based on missing or meeting durations

1.3.3 Nested Vectored Interrupt Controller

The CC23xx device controller includes the Arm NVIC. The NVIC and Arm Cortex M0+ prioritize and handle all exceptions in handler mode. The processor state is automatically stored to the stack on an exception and automatically restored from the stack at the end of the interrupt service routine (ISR). The processor supports tail-chaining, that is, back-to-back interrupts can be performed without the overhead of state saving and restoration. The software can set priority/preemption grouping in eight levels on internal CPU exceptions and interrupts.

Features of the NVIC include:

- Deterministic, fast interrupt processing
- External non-maskable interrupt (NMI) signal available for immediate execution of NMI handler
- Dynamically reprioritizable interrupts
- Exceptional interrupt handling through hardware implementation of required register manipulations

1.3.4 System Control Block (SCB)

The system control block (SCB) provides system implementation information and system control (configuration, control, and reporting of system exceptions).

1.4 On-Chip Memory

The following subsections describe the on-chip memory modules.

1.4.1 SRAM

The CC23xx devices have up to 36KB of low-leakage, on-chip SRAM with retention in all power modes except shutdown. Data can be transferred to and from the SRAM using the micro DMA (µDMA) controller. The ultra-low leakage system static RAM (SRAM) can be used to store data and execute code.

1.4.2 Flash

The flash block provides an in-circuit programmable, nonvolatile program memory for the device. Up to 512KB of flash memory is organized as a set of 2KB sectors that can be individually erased. Erasing a sector causes the entire contents of the sector to be reset to all 1s. These sectors can be write/erase protected. Write/erase protected sectors cannot be erased or programmed, protecting the contents of those sectors from being modified. In addition to holding program code and constants, the nonvolatile memory allows the application to save data that must be preserved so that the data is available after restarting the device. Using this feature lets the user use saved network-specific data and avoids the need for a full start-up and network find-and-join process.

1.4.3 ROM

The ROM is preprogrammed with a boot sequence, hardware APIs (HAPI), and a serial bootloader (SPI or UART).

1.5 Power Supply System

There are multiple voltage levels in use on the CC23xx to effectively optimize the power consumption of various modules operating in different power modes. Figure 1-2 shows an overview of the supply system.

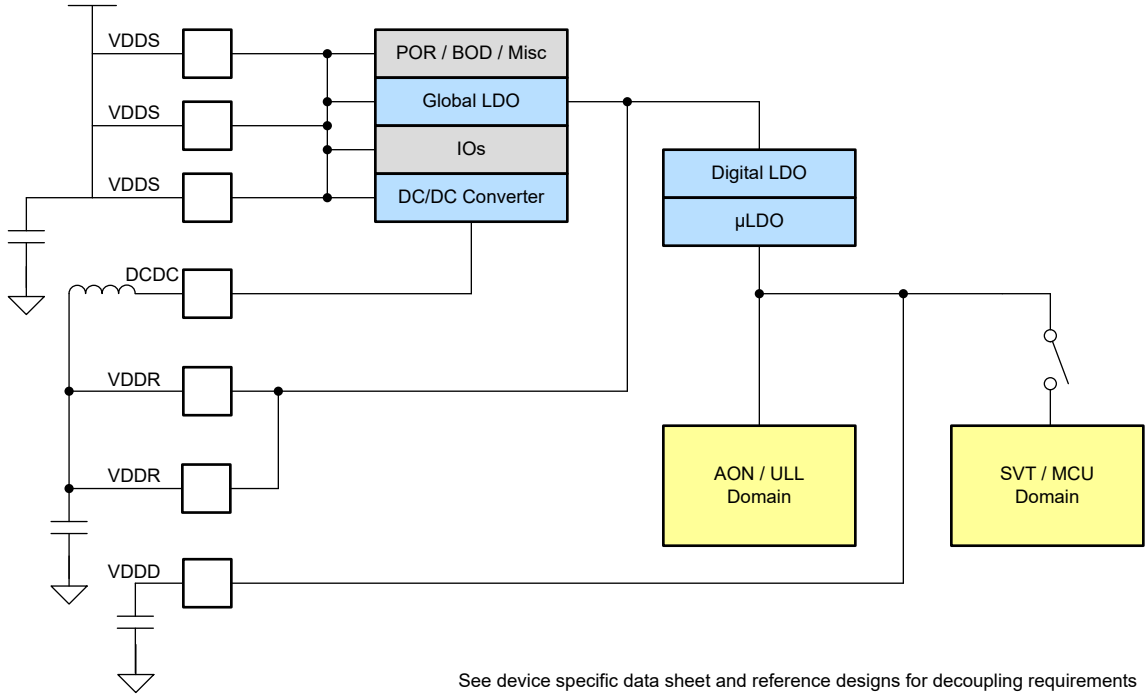


Figure 1-2. Supply System

1.5.1 VDDS

The battery voltage on the CC23xx device is called VDDS (supply). This supply has the highest voltage level in the system and is the only one provided by the user.

Note

All VDDS pins must be at the same voltage level.

1.5.2 VDDR

The two VDDR (regulated) pins are normally powered by one of the internal regulators. VDDR operates at 1.5V. For the lowest power, TI recommends using the internal DC/DC regulator. Using the Global LDO is also an option. See Section 6.3.1 for further details on this configuration.

Note

The VDDR pins cannot be used to supply external circuitry.

1.5.3 VDDD Digital Core Supply

The digital core of the CC23xx device is supplied by a 1.28V regulator connected to VDDR. The output of this regulator requires an external decoupling capacitor for proper operation. This capacitor must be connected to the VDDD pin.

Note

The VDDD pin cannot be used to supply external circuitry.

When the system is in standby, a small low-power regulator (µLDO) with limited current capacity supplies the digital domain to ensure enabled modules still have power.

1.5.4 DC/DC Converter

The on-chip buck-mode DC/DC converter provides a simple way to reduce the power consumption of the device. The DC/DC converter is integrated into the supply system and handles bias and clocks automatically through the system controller. The DC/DC converter is controlled through the PMCTL.VDDRCTL register. To enable the DC/DC converter when the system is active, the PMCTL.VDDRCTL[0] SELECT bit must be set. The DC/DC converter is also used periodically when the device is in standby mode to maintain voltage on the VDDR domain. The output voltage of the DC/DC regulator is trimmed to 1.5V. The voltage level controlled automatically by the device and cannot be changed by the user.

Note

The DC/DC regulator output cannot be used to supply external circuitry.

1.6 Radio

The CC23xx device provides a highly integrated low-power 2.4GHz radio transceiver with support for multiple modulations and packet formats. The radio subsystem provides an interface between the MCU and the radio transceiver, which makes it possible to issue commands, read status, and automate and sequence radio events. The RF path incorporates a balun to reduce system component count and simplify the design process.

1.7 AES 128-bit Cryptographic Accelerator

The CC23xx device integrates an AES-128 cryptography hardware accelerator which reduces code footprint and execution time for cryptographic operations. The AES accelerator also has the benefit of consuming less power and improves availability and responsiveness of the system because the cryptography operations run in a background hardware thread. The AES hardware accelerators supports the following block cipher modes and message authentication codes:

- AES ECB encrypt
- AES CBC encrypt
- AES CTR encrypt/decrypt
- AES CFB encrypt/decrypt
- AES OFB encrypt/decrypt
- AES CBC-MAC
- AES CCM (uses a combination of CTR + CBC-MAC hardware with software drivers)

Together with a large selection of open-source cryptography libraries provided with the Software Development Kit (SDK), this allows secure and future proof IoT applications to be easily built on top of the platform.

The CC23xx device supports Random Number Generation (RNG) using on-chip analog noise as the non-deterministic noise source for the purpose of generating a seed for a cryptographically secure counter deterministic random bit generator (CTR-DRBG) that in turn is used to generate random numbers for keys, initialization vectors (IVs), and other random number requirements. Hardware supports acceleration of AES CTR-DRBG.

1.8 System Timer (SYSTIM)

The SYSTIM is a 34-bit, 5-channel wrap-around timer with a per-channel selectable 32b slice with either a 1 us resolution and 1h11m35s range or 250ns resolution and 17m54s range. All channels support both capture and single-shot compare (posting an event) operation. One channel is reserved for system software, three channels are reserved for radio software, and one channel is freely available to user applications.

For software convenience, there is a hardware synchronization mechanism that automatically ensures that the RTC and SYSTIM share a common time base (albeit with different resolutions/spans). Another software convenience feature is that SYSTIM qualifies any submitted compare values so that the timer channel

immediately triggers if the submitted event is in the immediate past (4.294s with 1µs resolution and 1.049s with 250ns resolution).

1.9 General Purpose Timers (LGPT)

General-purpose timers can be used to count or time external events that drive the timer-input pins.

The general-purpose timer module (LGPT) contains up to four LGPT blocks with the following functional options. To determine which timers support which functions see the device-specific data sheet:

- 16-bit or 24-bit bit counter width
- Three capture/compare channels per timer
- One-shot or periodic counting
- Time counting between edges and edge counting
- Input filter on each of the channels for all timers
- 15 different channel Capture/Compare actions
- PWM Generation with programmable deadband insertion and park on fault mode
- IR Signal generation
- Quadrature decoding (QDEC)
- Timer synchronization and chaining
- Efficient transfers using the µDMA controller

1.10 Always-ON (AON) or Ultra-Low Leakage (ULL) Domain

The AON/ULL domain contains circuitry that is always enabled, except for in the shutdown power state (where the digital supply is off). For more information on power states see Chapter 6.

This domain includes the following components.

1.10.1 Watchdog Timer

The watchdog timer is used to regain control when the system fails because of a software error or when an external device fails to respond properly. The watchdog timer can generate a reset when a predefined time-out value is reached.

The watchdog timer runs on a 32kHz clock rate and operates in device active, idle, and standby modes and cannot be stopped once enabled.

1.10.2 Battery and Temperature Monitor

A combined temperature and battery voltage monitor is available in the CC23xx device. The battery and temperature monitor allows an application to continuously monitor on-chip temperature and supply voltage and respond to changes in environmental conditions as needed. The module contains window comparators to interrupt the system CPU when temperature or supply voltage goes outside defined windows. These events can also be used to wake up the device from standby mode through the Always-On (ULL/AON) event fabric.

1.10.3 Real-time Clock (RTC)

The RTC can be used to wake the CC23xx device from any state where the RTC is active. The RTC contains one capture and one compare channel. With software support, the RTC can be used for clock and calendar operation. The RTC is clocked from the 32kHz RC oscillator or the 32kHz crystal oscillator. The CLKLF period is measured in hardware and the resulting LFINC is used to increment RTC time on every CLKLF edge.

1.10.4 Low Power Comparator

The low-power comparator is an ultra-low-power clocked (on CLKLF) comparator that can be used for medium accuracy, and low-speed operations where power consumption is the main concern. Typical applications are wake-up on analog events like power supply monitoring or external transducers that generate analog output signals. The comparator includes input multiplexers on both the signal (positive) side and reference (negative) side, a capacitive divider for low-power division of either the reference signal or the input signal, and programmable hysteresis that can be configured to trigger on both low and high comparator output. The low-power comparator is also active in standby mode.

1.11 Direct Memory Access

The CC23xx device includes a DMA controller, known as µDMA. The µDMA controller provides a way to offload data transfer tasks from the Arm Cortex M0+ processor, allowing more efficient use of the processor and the available bus bandwidth. The µDMA controller can perform transfers between memory and peripherals. Channels in the µDMA are multiplexed between each supported on-chip module and can be programmed to automatically perform transfers between peripherals and memory when the peripheral is ready to transfer more data.

1.12 System Control and Clock

System control determines the overall operation of the CC23xx device. System control provides information about the devices, controls power-saving features, controls the clocking of the devices and individual peripherals, and handles reset detection and reporting.

- Power Control:
  - On-chip fixed DC/DC converter and global low drop-out (GLDO) voltage regulators
  - Handles the power-up sequencing, power-down sequencing, and control for the core digital-logic and analog circuits
  - Low-power options for on-chip modules:
    - Software controls the shutdown of individual peripherals and memory.
    - SRAM is retained in all power modes except shutdown.
  - Configurable wake-up from standby by RTC or any IOC interrupt
  - Voltage supervision circuitry
- Multiple clock sources for microcontroller system clock:
  - High Frequency Clock:
    - RC oscillator (HFOSC): on-chip 48MHz RC oscillator
    - External oscillator (HFXT): an external 48MHz crystal oscillator connected across the X48P input and X48N output pins
    - Radio operation requires an external oscillator.
  - Low Frequency Clock:
    - RC oscillator (LFOSC): on-chip 32kHz RC oscillator
    - External oscillator (LFXT): external 32.768kHz crystal oscillator connected across the X32P input and X32N output pins
    - Designed for accurate RTC operation or synchronous network timing
    - Used during power-saving modes and for RTC

1.13 Communication Peripherals

The CC23xx device platform supports both asynchronous and synchronous serial communication including:

- UART
- I^2C
- SPI

The following subsections provide more detail on each of the communication modules.

1.13.1 UART

A UART is an integrated circuit used for TTL serial communications. A UART contains a transmitter (parallel-to-serial converter) and a receiver (serial-to-parallel converter).

The CC23xx device includes a fully programmable UART. The UART can generate individually masked interrupts from the receive (RX), transmit (TX), modem flow control, and error conditions. The module generates one combined interrupt when any interrupts are asserted and unmasked.

The UART has the following features:

- Programmable baud rate generator allowing speeds up to 3Mbps
- Separate 8 × 8 transmit (TX) and 8 × 12 receive (RX) first-in-first-out (FIFO) buffers to reduce CPU interrupt service loading

- RX/TX FIFOs can be reconfigured to a 16x8b TX FIFO for unidirectional output
- Programmable FIFO length, including 1-byte deep operation providing a conventional double-buffered interface
- FIFO trigger levels of ¼, ½, ¾
- Line-break generation and detection
- Fully programmable serial interface characteristics:
  - 5, 6, 7, or 8 data bits
  - Even, odd, stick, or no parity bit generation and detection
  - 1 or 2 stop-bit generation
- FIFO, RX FIFO RX time-out, modem status, and error conditions
- Standard FIFO-level and end-of-transmission interrupts.
- Efficient transfers using micro direct memory access controller (µDMA)
- Separate µDMA channels for transmit and receive.
  - Receive single request asserted when data is in the FIFO; burst request asserted at programmed FIFO level.
  - Transmit single request is asserted when there is space in the FIFO; burst request is asserted at the programmed FIFO level.
- Programmable hardware flow control
- Support for standard Infrared Data Association (IrDA) and low-power IrDA protocols
- Provision to combine both TX and RX FIFOs in transmit mode

1.13.2 I^2C

The I^2C bus provides bidirectional data transfer through a 2-wire design (a serial data line SDA and a serial clock line SCL). The I^2C bus interfaces to external I^2C devices such as serial memory (RAM and ROM), networking devices, LCDs, tone generators, and so on. The I^2C bus can also be used for system testing and diagnostic purposes in product development and manufacturing.

The CC23xx device includes an I^2C module with the following features:

- Devices on the I^2C bus can be designated as either a controller or target:
  - Supports both transmitting and receiving data as either a controller or a target
  - Supports simultaneous controller and target operation
- Four I^2C modes:
  - Controller transmit
  - Controller receive
  - Target transmit
  - Target receive
- Two transmission speeds:
  - Standard (100kbps)
  - Fast (400kbps)
- Controller and target interrupt generation:
  - Controller generates interrupts when a TX or RX operation completes (or aborts due to an error)
  - Target generates interrupts when data is transferred or requested by a controller or when a START or STOP condition is detected.
  - Controller with arbitration and clock synchronization, multicontroller support, and 7-bit addressing mode

1.13.3 SPI

The SPI module is a 3-wire or 4-wire bidirectional communication interface that converts data between parallel and serial formats. The SPI performs serial-to-parallel conversion on data received from a target device and performs parallel-to-serial conversion on data transmitted to a target device. The SPI can be configured as either a controller or a peripheral device. As a peripheral device, the SPI can be configured to disable the SPI output, which allows the coupling of a controller device with multiple peripheral devices. The TX and RX paths are buffered with separate internal FIFOs.

The SPI also includes a programmable bit rate clock divider and prescaler to generate the output serial clock derived from the input clock of the SPI. Bit rates are generated based on the input clock, see the device-specific data sheet for maximum bit rates.

The SPI module supports the following features:

- Programmable interface operation for Motorola SPI (3-wire and 4-wire), MICROWIRE, or TI Synchronous Serial format
- Configurable as a controller or a peripheral on the interface
- Programmable clock bit rate and prescaler
- CRC8-CCITT or CRC16-CCITT CRC capability and auto insertion in TX upon underflow
- Separate transmit (TX) and receive (RX) first-in-first-out buffers (FIFOs)
  - If Data Size Select (DSS) is 4 to 8 bits FIFOs are 16 locations deep and 8 bits wide
  - If Data Size Select (DSS) is 9 to 16 bits FIFOS are 8 locations deep and 16 bits wide
- Programmable data frame size from 4 bits to 16 bits (controller mode) or 7 to 16 bits (peripheral mode)
- Internal loopback test mode for diagnostic and debug testing
- Interrupts for transmit and receive FIFOs, overrun and time-out interrupts, and DMA-done interrupts
- Efficient transfers using micro direct memory access controller (µDMA):
  - Separate channels for transmit and receive
  - Receive single request asserted when data is in the FIFO; burst request asserted when FIFO contains a configurable number of entries
  - Transmit single request asserted when there is space in the FIFO; burst request asserted when FIFO contains a configurable number of entries

1.14 Programmable I/Os

I/O pins offer flexibility for a variety of connections. The CC23xx device supports configurable I/O pins that can be multiplexed to digital and analog peripherals through the I/O Controller. For information on what pins can be multiplexed to what peripherals see the device-specific data sheet.

- Up to 26 GPIO, depending on the package
- Up to 6 high drive strength pins, drive strength is configurable for the high drive I/Os. Drive strength be set to AUTO where I/Os automatically use the drive strength required to support a 12MHz toggle rate based on measured VDDS voltage
- Up to 12 analog-capable pins
- Programmable control for GPIO interrupts:
  - Interrupt generation masking per pin
  - Edge-triggered on rising or falling edges
- Can initiate a µDMA transfer
- The pin state can be retained during all sleep modes
- Wake-up from IOC supported on all pins in all power modes
- Pins configured as digital inputs are Schmitt-triggered
- Programmable control for DIO configuration:
  - Weak pullup or pulldown resistors
  - Digital input buffer enable controls

1.15 Serial Wire Debug (SWD)

SWD is an industry standard 2-pin ARM SWD interface used for device programming, configuration and in-circuit debugging. The 2-wire (SWDIO, SWCLK) debug interface is compatible with both TI and 3rd party debug probes and features:

- On-chip pullup/pulldown resistors for SWDIO and SWCLK, respectively, enabled by default
- Support for disabling SWD functions to use SWD pins as general purpose input/output pins
- Capability of waking the device from shutdown mode upon valid SWD activity

Chapter 2

Arm® Cortex®-M0+ Processor

The CC23xx builds on the Arm Cortex-M0+ core to bring high-performance 32-bit computing to cost-sensitive embedded microcontroller applications, such as electronic shelf labels, tire pressure monitoring systems, and fitness applications. This chapter provides information on the CC23xx implementation of the Arm Cortex-M0+ processor. For technical details on the instruction set, see the *Cortex-M0+ Technical Reference Manual*.

2.1 Introduction

The Cortex-M0+ processor is an entry-level 32-bit ARM Cortex processor designed for a broad range of embedded applications. The Cortex-M0+ processor offers significant benefits to developers, including:

- A simple architecture that is easy to learn and program
- Excellent code density
- 48MHz operation
- Implements the ARMv6-M instruction set architecture (ISA)
- Enhanced system debugging with up to four breakpoints
- Single cycle 32 × 32 multiply instruction
- User and privileged execution modes
- Integrated 24-bit system tick timer (SysTick)
- Configurable vector table offset
- Nested Vectored Interrupt Controller (NVIC)

Features include:

- Ultra-low power, energy-efficient operation
- Deterministic, high-performance interrupt handling
- Upward compatibility with Cortex-M processor family

Additionally, the CC23xx devices are compatible with all ARM tools and software.

2.2 Block Diagram

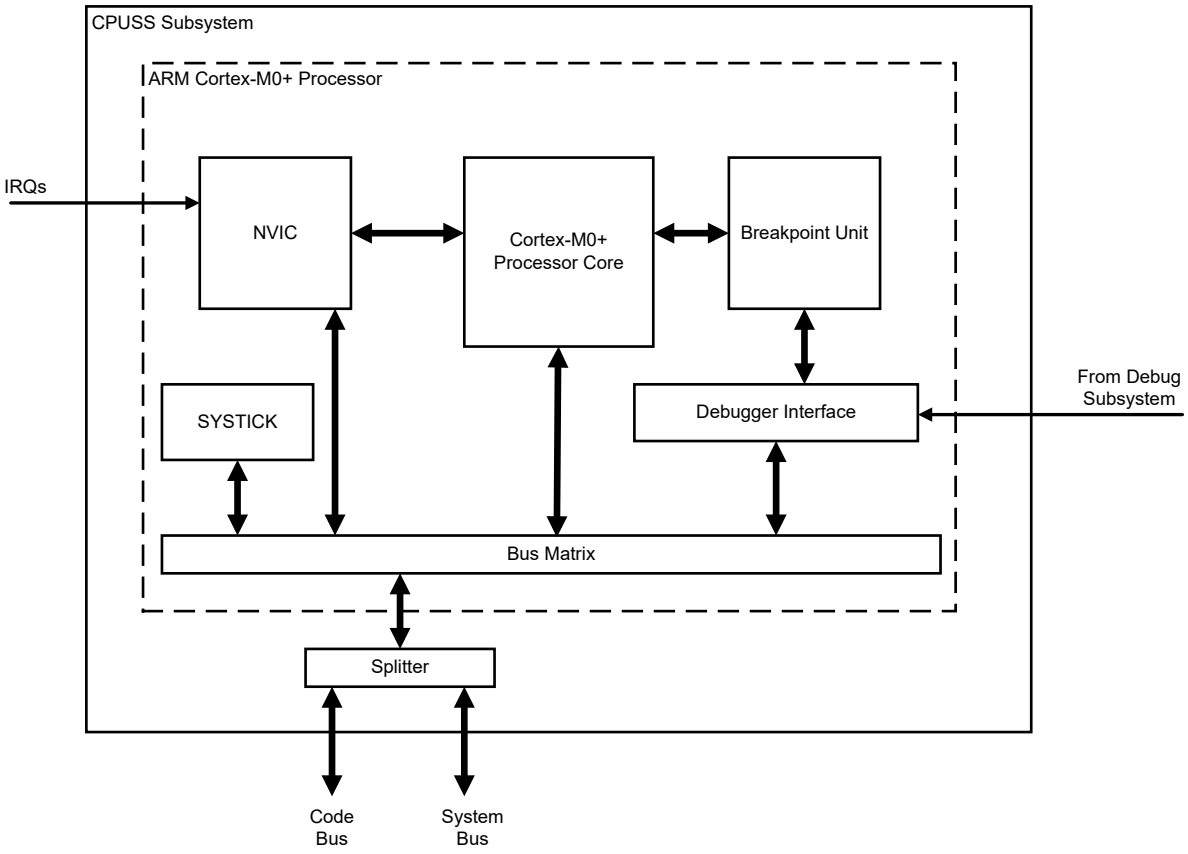


Figure 2-1. CPUSS Block Diagram

2.3 Overview

2.3.1 Peripherals

This sections describes the available ARM peripherals.

Nested Vectored Interrupt Controller (NVIC)

The NVIC is an embedded interrupt controller that supports low latency interrupt processing.

System Control Block

The System Control Block (SCB) is the programmer's model interface to the processor. The SCB provides system implementation information and system control, including configuration, control, and reporting of system exceptions.

SysTick

SysTick is a 24-bit countdown timer. This can be used as a Real Time Operating System (RTOS) tick timer or as a simple counter.

Memory Map

The Cortex-M0+ processor has a fixed default memory map that provides up to 4GB of addressable memory. The Cortex-M0+ processor memory map is shown below in Figure 2-2. For a more detailed view on how memory is mapped for CC23xx see Chapter 3.

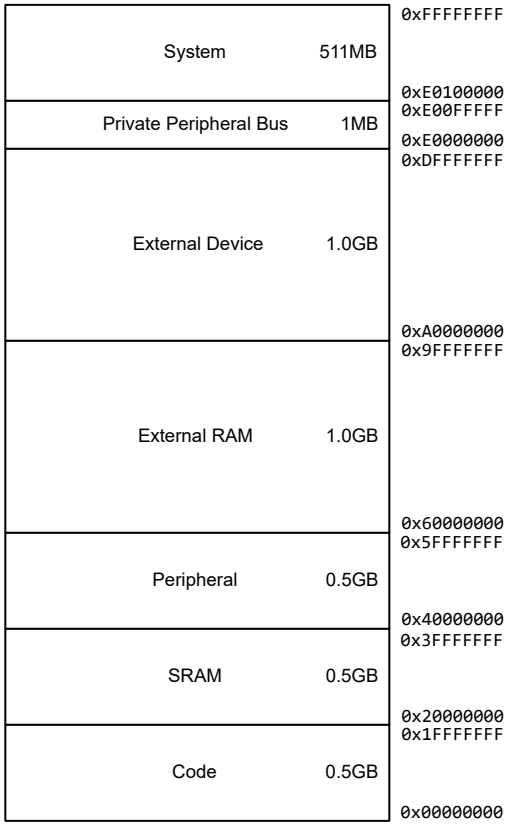


Figure 2-2. Memory Model

The processor reserves regions of the private peripheral bus (PPB) address range for core peripheral registers.

Table 2-1. Core Peripheral Register Regions

Address	Core Peripheral	Link
0xE000E008-0xE000E00F	System Control Block	Cortex-M0+ Devices Generic User Guide
0xE000E010-0xE000E01F	SysTick	Cortex-M0+ Devices Generic User Guide
0xE000E100-0xE000E4EF	Nested Vectored Interrupt Controller	Cortex-M0+ Devices Generic User Guide
0xE000ED00-0xE000ED3F	System Control Block	Cortex-M0+ Devices Generic User Guide
0xE000ED90-0xE000EDB8	Reserved	Cortex-M0+ Devices Generic User Guide
0xE000EF00-0xE000EF03	Nested Vectored Interrupt Controller	Cortex-M0+ Devices Generic User Guide

2.3.2 Programmer's Model

For information on the Cortex-M0+ programmers model, see the Cortex-M0+ Devices Generic User Guide.

2.3.3 Instruction Set Summary

The processor implements the Arm® Thumb® instruction set. See the Cortex-M0+ Devices Generic User Guide for the supported instructions.

2.3.4 Memory Model

For information on the Cortex-M0+ memory model, see the Cortex-M0+ Devices Generic User Guide.

2.4 Registers

See Chapter 3 for the register base addresses.

2.4.1 BPU Registers

Table 2-2 lists the memory-mapped registers for the BPU registers. All register offset addresses not listed in Table 2-2 should be considered as reserved locations and the register contents should not be modified.

Table 2-2. BPU Registers

Offset	Acronym	Register Name	Section
0h	BP_CTRL	Breakpoint Control Register	Go
8h	BP_COMP0	Breakpoint Comparator Register 0	Go
Ch	BP_COMP1	Breakpoint Comparator Register 1	Go
10h	BP_COMP2	Breakpoint Comparator Register 2	Go
14h	BP_COMP3	Breakpoint Comparator Register 3	Go
FD0h	PIDR4	Peripheral ID Register 4	Go
FD4h	PIDR5	Peripheral ID Register 5	Go
FD8h	PIDR6	Peripheral ID Register 6	Go
FDCh	PIDR7	Peripheral ID Register 7	Go
FE0h	PIDR0	Peripheral ID Register 0	Go
FE4h	PIDR1	Peripheral ID Register 1	Go
FE8h	PIDR2	Peripheral ID Register 2	Go
FECh	PIDR3	Peripheral ID Register 3	Go
FF0h	CIDR0	Component ID Register 0	Go
FF4h	CIDR1	Component ID Register 1	Go
FF8h	CIDR2	Component ID Register 2	Go
FFCh	CIDR3	Component ID Register 3	Go

Complex bit access types are encoded to fit into small table cells. Table 2-3 shows the codes that are used for access types in this section.

Table 2-3. BPU Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

2.4.1.1 BP_CTRL Register (Offset = 0h) [Reset = 00000040h]

BP_CTRL is shown in Table 2-4.

Return to the Summary Table.

Breakpoint Control Register
Use the Breakpoint Control Register to enable the Breakpoint block

Table 2-4. BP_CTRL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-4	NUM_CODE	R	4h	Number of comparators.
3-2	RESERVED	R	0h	Reserved
1	KEY	W	0h	Key field. To write to the Breakpoint Control Register, you must write a 1 to this write-only bit. This bit reads as zero.
0	ENABLE	R/W	0h	Breakpoint unit enable bit. DBGRESETn clears the ENABLE bit. 0h = Breakpoint unit disabled 1h = Breakpoint unit enabled

2.4.1.2 BP_COMP0 Register (Offset = 8h) [Reset = 00000000h]

BP_COMP0 is shown in Table 2-5.

Return to the Summary Table.

Breakpoint Comparator Register 0
Use the Breakpoint Comparator Registers to store the values to compare with the instruction address.

Table 2-5. BP_COMP0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-30	BP_MATCH	R/W	0h	This selects what happens when the COMP address is matched 0h = no breakpoint generated 1h = set breakpoint on lower halfword, upper is unaffected 2h = set breakpoint on upper halfword, lower is unaffected 3h = set breakpoint on both lower and upper halfwords
29	RESERVED	R	0h	Reserved
28-2	COMP	R/W	0h	Comparison address, UNKNOWN on reset.
1	RESERVED	R	0h	Reserved
0	ENABLE	R/W	0h	Compare enable for Breakpoint Comparator Register 0. The ENABLE bit of BP_CTRL must also be set to enable comparisons. DBGRESETn clears the ENABLE bit. 0h = Breakpoint Comparator Register 0 compare disabled 1h = Breakpoint Comparator Register 0 compare enabled

2.4.1.3 BP_COMP1 Register (Offset = Ch) [Reset = 00000000h]

BP_COMP1 is shown in Table 2-6.

Return to the Summary Table.

Breakpoint Comparator Register 1
Use the Breakpoint Comparator Registers to store the values to compare with the instruction address.

Table 2-6. BP_COMP1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-30	BP_MATCH	R/W	0h	This selects what happens when the COMP address is matched 0h = No breakpoint generated 1h = Set breakpoint on lower halfword, upper is unaffected 2h = Set breakpoint on upper halfword, lower is unaffected 3h = Set breakpoint on both lower and upper halfwords
29	RESERVED	R	0h	Reserved
28-2	COMP	R/W	0h	Comparison address. Although it is architecturally Unpredictable whether breakpoint matches on the address of the second halfword of a 32-bit instruction to generates a debug event, in this processor it is predictable and a debug event is generated.
1	RESERVED	R	0h	Reserved
0	ENABLE	R/W	0h	Comparison address, UNKNOWN on reset. 0h = Breakpoint Comparator Register 1 compare disabled 1h = Breakpoint Comparator Register 1 compare enabled

2.4.1.4 BP_COMP2 Register (Offset = 10h) [Reset = 00000000h]

BP_COMP2 is shown in Table 2-7.

Return to the Summary Table.

Breakpoint Comparator Register 2
Use the Breakpoint Comparator Registers to store the values to compare with the PC address.

Table 2-7. BP_COMP2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-30	BP_MATCH	R/W	0h	This selects what happens when the COMP address is matched 0h = No breakpoint matching 1h = Set breakpoint on lower halfword, upper is unaffected 2h = Set breakpoint on upper halfword, lower is unaffected 3h = Set breakpoint on both lower and upper halfwords
29	RESERVED	R	0h	Reserved
28-2	COMP	R/W	0h	Comparison address, UNKNOWN on reset.
1	RESERVED	R	0h	Reserved
0	ENABLE	R/W	0h	Compare enable for Breakpoint Comparator Register 2. The ENABLE bit of BP_CTRL must also be set to enable comparisons. DBGRESETn clears the ENABLE bit. 0h = Breakpoint Comparator Register 2 compare disabled 1h = Breakpoint Comparator Register 2 compare enabled

2.4.1.5 BP_COMP3 Register (Offset = 14h) [Reset = 00000000h]

BP_COMP3 is shown in Table 2-8.

Return to the Summary Table.

Breakpoint Comparator Register 3
Use the Breakpoint Comparator Registers to store the values to compare with the instruction address.

Table 2-8. BP_COMP3 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-30	BP_MATCH	R/W	0h	This selects what happens when the COMP address is matched 0h = No breakpoint generated 1h = Set breakpoint on lower halfword, upper is unaffected 2h = Set breakpoint on upper halfword, lower is unaffected 3h = Set breakpoint on both lower and upper halfwords
29	RESERVED	R	0h	Reserved
28-2	COMP	R/W	0h	Comparison address, UNKNOWN on reset.
1	RESERVED	R	0h	Reserved
0	ENABLE	R/W	0h	Compare enable for Breakpoint Comparator Register 3. The ENABLE bit of BP_CTRL must also be set to enable comparisons. DBGRESETn clears the ENABLE bit. 0h = Breakpoint Comparator Register 3 compare disabled 1h = Breakpoint Comparator Register 3 compare enabled

2.4.1.6 PIDR4 Register (Offset = FD0h) [Reset = 00000004h]

PIDR4 is shown in Table 2-9.

Return to the Summary Table.

Peripheral ID Register 4
Part of the set of Peripheral Identification registers. Contains part of the designer identity and the memory footprint indicator.

Table 2-9. PIDR4 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-4	SIZE	R	0h	This is a 4-bit value that indicates the total contiguous size of the memory window used by this component in powers of 2 from the standard 4KB. If a component only requires the standard 4KB then this should read as 0x0, 4KB only, for 8KB set to 0x1, 16KB == 0x2, 32KB == 0x3, and so on.
3-0	DES_2	R	4h	Number of JEDEC continuation codes. Indicates the designer of the component (along with the identity code)

2.4.1.7 PIDR5 Register (Offset = FD4h) [Reset = 00000000h]

PIDR5 is shown in Table 2-10.

Return to the Summary Table.

Peripheral ID Register 5
Reserved

Table 2-10. PIDR5 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	RESERVED	R	0h	Reserved

2.4.1.8 PIDR6 Register (Offset = FD8h) [Reset = 00000000h]

PIDR6 is shown in Table 2-11.

Return to the Summary Table.

Peripheral ID Register 6
Reserved

Table 2-11. PIDR6 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	RESERVED	R	0h	Reserved

2.4.1.9 PIDR7 Register (Offset = FDCh) [Reset = 00000000h]

PIDR7 is shown in Table 2-12.

Return to the Summary Table.

Peripheral ID Register 7
Reserved

Table 2-12. PIDR7 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	RESERVED	R	0h	Reserved

2.4.1.10 PIDR0 Register (Offset = FE0h) [Reset = 0000000Bh]

PIDR0 is shown in Table 2-13.

Return to the Summary Table.

Peripheral ID Register 0
Part of the set of Peripheral Identification registers. Contains part of the designer specific part number.

Table 2-13. PIDR0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	PART_0	R	Bh	Bits [7:0] of the component's part number. This is selected by the designer of the component.

2.4.1.11 PIDR1 Register (Offset = FE4h) [Reset = 000000B0h]

PIDR1 is shown in Table 2-14.

Return to the Summary Table.

Peripheral ID Register 1
Part of the set of Peripheral Identification registers. Contains part of the designer specific part number and part of the designer identity.

Table 2-14. PIDR1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-4	DES_0	R	Bh	Bits [3:0] of the JEDEC identity code indicating the designer of the component (along with the continuation code)
3-0	PART_1	R	0h	Bits [11:8] of the component's part number. This is selected by the designer of the component.

2.4.1.12 PIDR2 Register (Offset = FE8h) [Reset = 0000000Bh]

PIDR2 is shown in Table 2-15.

Return to the Summary Table.

Peripheral ID Register 2
Part of the set of Peripheral Identification registers. Contains part of the designer identity and the product revision.

Table 2-15. PIDR2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-4	REVISION	R	0h	The Revision field is an incremental value starting at 0x0 for the first design of this component. This only increases by 1 for both major and minor revisions and is simply used as a look-up to establish the exact major/minor revision.
3	JEDEC	R	1h	Always set. Indicates that a JEDEC assigned value is used
2-0	DES_1	R	3h	Bits [6:4] of the JEDEC identity code indicating the designer of the component (along with the continuation code)

2.4.1.13 PIDR3 Register (Offset = FECh) [Reset = 00000000h]

PIDR3 is shown in Table 2-16.

Return to the Summary Table.

Peripheral ID Register 3
Part of the set of Peripheral Identification registers. Contains the RevAnd and Customer Modified fields.

Table 2-16. PIDR3 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-4	REVAND	R	0h	This field indicates minor errata fixes specific to this design, for example metal fixes after implementation. In most cases this field is zero. It is recommended that component designers ensure this field can be changed by a metal fix if required, for example by driving it from registers that reset to zero.
3-0	CMOD	R	0h	Where the component is reusable IP, this value indicates if the customer has modified the behavior of the component. In most cases this field is zero.

2.4.1.14 CIDR0 Register (Offset = FF0h) [Reset = 0000000Dh]

CIDR0 is shown in Table 2-17.

Return to the Summary Table.

Component ID Register 0
A component identification register, that indicates that the identification registers are present.

Table 2-17. CIDR0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	PRMBL_0	R	Dh	Contains bits [7:0] of the component identification

2.4.1.15 CIDR1 Register (Offset = FF4h) [Reset = 000000E0h]

CIDR1 is shown in Table 2-18.

Return to the Summary Table.

Component ID Register 1
A component identification register, that indicates that the identification registers are present. This register also indicates the component class.

Table 2-18. CIDR1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-4	CLASS	R	Eh	Class of the component. for example, . ROM table, CoreSight component and so on. Constitutes bits [15:12] of the component identification.
3-0	PRMBL_1	R	0h	Contains bits [11:8] of the component identification

2.4.1.16 CIDR2 Register (Offset = FF8h) [Reset = 00000005h]

CIDR2 is shown in Table 2-19.

Return to the Summary Table.

Component ID Register 2
A component identification register, that indicates that the identification registers are present.

Table 2-19. CIDR2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	PRMBL_2	R	5h	Contains bits [23:16] of the component identification

2.4.1.17 CIDR3 Register (Offset = FFCh) [Reset = 000000B1h]

CIDR3 is shown in Table 2-20.

Return to the Summary Table.

Component ID Register 3
A component identification register, that indicates that the identification registers are present.

Table 2-20. CIDR3 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	PRMBL_3	R	B1h	Contains bits [31:24] of the component identification

2.4.2 CPU_ROM_TABLE Registers

Table 2-21 lists the memory-mapped registers for the CPU_ROM_TABLE registers. All register offset addresses not listed in Table 2-21 should be considered as reserved locations and the register contents should not be modified.

Table 2-21. CPU_ROM_TABLE Registers

Offset	Acronym	Register Name	Section
0h	EntrySCS	ROM Table Entry	Go
4h	EntryDWT	ROM Table Entry	Go
8h	EntryBPU	ROM Table Entry	Go
Ch	EntryEnd	ROM Table Entry	Go
FCCh	MEMTYPE	MEMTYPE Register	Go
FD0h	PIDR4	Peripheral ID Register 4	Go
FD4h	PIDR5	Peripheral ID Register 5	Go
FD8h	PIDR6	Peripheral ID Register 6	Go
FDCh	PIDR7	Peripheral ID Register 7	Go
FE0h	PIDR0	Peripheral ID Register 0	Go
FE4h	PIDR1	Peripheral ID Register 1	Go
FE8h	PIDR2	Peripheral ID Register 2	Go
FECh	PIDR3	Peripheral ID Register 3	Go
FF0h	CIDR0	Component ID Register 0	Go
FF4h	CIDR1	Component ID Register 1	Go
FF8h	CIDR2	Component ID Register 2	Go
FFCh	CIDR3	Component ID Register 3	Go

Complex bit access types are encoded to fit into small table cells. Table 2-22 shows the codes that are used for access types in this section.

Table 2-22. CPU_ROM_TABLE Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Reset or Default Value		
-n		Value after reset or the default value

2.4.2.1 EntrySCS Register (Offset = 0h) [Reset = FFF0F003h]

EntrySCS is shown in Table 2-23.

Return to the Summary Table.

ROM Table Entry
Points to the System Control Space (SCS) at 0xE000E000. This includes core debug control registers.

Table 2-23. EntrySCS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-12	AddressOffset	R	000FFF0Fh	Base address of the highest 4KB block for the component, relative to the ROM address.
11-2	RESERVED	R	0h	Reserved
1	Format	R	1h	Base address of the highest 4KB block for the component, relative to the ROM address.
0	EntryPresent	R	1h	This bit indicates whether an entry is present at this location in the ROM table. 0x0:Rom table entry is not present and must be skipped. 0x1:Rom table entry is present.

2.4.2.2 EntryDWT Register (Offset = 4h) [Reset = FFF02003h]

EntryDWT is shown in Table 2-24.

Return to the Summary Table.

ROM Table Entry
Points to the DW unit at 0xE0001000.

Table 2-24. EntryDWT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-12	AddressOffset	R	000FFF02h	Base address of the highest 4KB block for the component, relative to the ROM address.
11-2	RESERVED	R	0h	Reserved
1	Format	R	1h	Base address of the highest 4KB block for the component, relative to the ROM address.
0	EntryPresent	R	1h	This bit indicates whether an entry is present at this location in the ROM table. 0x0:Rom table entry is not present and must be skipped. 0x1:Rom table entry is present.

2.4.2.3 EntryBPU Register (Offset = 8h) [Reset = FFF03002h]

EntryBPU is shown in Table 2-25.

Return to the Summary Table.

ROM Table Entry
Points to the BPU at 0xE0002000.

Table 2-25. EntryBPU Register Field Descriptions

Bit	Field	Type	Reset	Description
31-12	AddressOffset	R	000FFF03h	Base address of the highest 4KB block for the component, relative to the ROM address.
11-2	RESERVED	R	0h	Reserved
1	Format	R	1h	Base address of the highest 4KB block for the component, relative to the ROM address.
0	EntryPresent	R	0h	This bit indicates whether an entry is present at this location in the ROM table. 0x0:Rom table entry is not present and must be skipped. 0x1:Rom table entry is present.

2.4.2.4 EntryEnd Register (Offset = Ch) [Reset = 00000000h]

EntryEnd is shown in Table 2-26.

Return to the Summary Table.

ROM Table Entry
Marks of end of table.

Table 2-26. EntryEnd Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	END	R	0h	Blank ROM Table entry indicating the end of the ROM Table content.

2.4.2.5 MEMTYPE Register (Offset = FCCh) [Reset = 00000001h]

MEMTYPE is shown in Table 2-27.

Return to the Summary Table.

MEMTYPE Register
Identifies the type of memory present on the bus that connects the DAP to the ROM Table.

Table 2-27. MEMTYPE Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Reserved
0	SYSMEM	R	1h	System memory present. Indicates whether system memory is present on the bus that connects to the ROM table. 0x0:System memory not present on bus. this is a dedicated debug bus. 0x1:System memory is present on bus.

2.4.2.6 PIDR4 Register (Offset = FD0h) [Reset = 00000004h]

PIDR4 is shown in Table 2-28.

Return to the Summary Table.

Peripheral ID Register 4
Part of the set of Peripheral Identification registers. Contains part of the designer identity and the memory footprint indicator.

Table 2-28. PIDR4 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-4	SIZE	R	0h	This is a 4-bit value that indicates the total contiguous size of the memory window used by this component in powers of 2 from the standard 4KB. If a component only requires the standard 4KB then this should read as 0x0, 4KB only, for 8KB set to 0x1, 16KB == 0x2, 32KB == 0x3, and so on.
3-0	DES_2	R	4h	Number of JEDEC continuation codes. Indicates the designer of the component (along with the identity code)

2.4.2.7 PIDR5 Register (Offset = FD4h) [Reset = 00000000h]

PIDR5 is shown in Table 2-29.

Return to the Summary Table.

Peripheral ID Register 5
Reserved

Table 2-29. PIDR5 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	RESERVED	R	0h	Reserved

2.4.2.8 PIDR6 Register (Offset = FD8h) [Reset = 00000000h]

PIDR6 is shown in Table 2-30.

Return to the Summary Table.

Peripheral ID Register 6
Reserved

Table 2-30. PIDR6 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	RESERVED	R	0h	Reserved

2.4.2.9 PIDR7 Register (Offset = FDCh) [Reset = 00000000h]

PIDR7 is shown in Table 2-31.

Return to the Summary Table.

Peripheral ID Register 7
Reserved

Table 2-31. PIDR7 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	RESERVED	R	0h	Reserved

2.4.2.10 PIDR0 Register (Offset = FE0h) [Reset = 000000C0h]

PIDR0 is shown in Table 2-32.

Return to the Summary Table.

Peripheral ID Register 0
Part of the set of Peripheral Identification registers. Contains part of the designer specific part number.

Table 2-32. PIDR0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	PART_0	R	C0h	Bits [7:0] of the component's part number. This is selected by the designer of the component.

2.4.2.11 PIDR1 Register (Offset = FE4h) [Reset = 000000B4h]

PIDR1 is shown in Table 2-33.

Return to the Summary Table.

Peripheral ID Register 1
Part of the set of Peripheral Identification registers. Contains part of the designer specific part number and part of the designer identity.

Table 2-33. PIDR1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-4	DES_0	R	Bh	Bits [3:0] of the JEDEC identity code indicating the designer of the component (along with the continuation code)
3-0	PART_1	R	4h	Bits [11:8] of the component's part number. This is selected by the designer of the component.

2.4.2.12 PIDR2 Register (Offset = FE8h) [Reset = 0000000Bh]

PIDR2 is shown in Table 2-34.

Return to the Summary Table.

Peripheral ID Register 2
Part of the set of Peripheral Identification registers. Contains part of the designer identity and the product revision.

Table 2-34. PIDR2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-4	REVISION	R	0h	The Revision field is an incremental value starting at 0x0 for the first design of this component. This only increases by 1 for both major and minor revisions and is simply used as a look-up to establish the exact major/minor revision.
3	JEDEC	R	1h	Always set. Indicates that a JEDEC assigned value is used
2-0	DES_1	R	3h	Bits [6:4] of the JEDEC identity code indicating the designer of the component (along with the continuation code)

2.4.2.13 PIDR3 Register (Offset = FECh) [Reset = 00000000h]

PIDR3 is shown in Table 2-35.

Return to the Summary Table.

Peripheral ID Register 3
Part of the set of Peripheral Identification registers. Contains the RevAnd and Customer Modified fields.

Table 2-35. PIDR3 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-4	REVAND	R	0h	This field indicates minor errata fixes specific to this design, for example metal fixes after implementation. In most cases this field is zero. It is recommended that component designers ensure this field can be changed by a metal fix if required, for example by driving it from registers that reset to zero.
3-0	CMOD	R	0h	Where the component is reusable IP, this value indicates if the customer has modified the behavior of the component. In most cases this field is zero.

2.4.2.14 CIDR0 Register (Offset = FF0h) [Reset = 0000000Dh]

CIDR0 is shown in Table 2-36.

Return to the Summary Table.

Component ID Register 0
A component identification register, that indicates that the identification registers are present.

Table 2-36. CIDR0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	PRMBL_0	R	Dh	Contains bits [7:0] of the component identification

2.4.2.15 CIDR1 Register (Offset = FF4h) [Reset = 00000010h]

CIDR1 is shown in Table 2-37.

Return to the Summary Table.

Component ID Register 1
A component identification register, that indicates that the identification registers are present. This register also indicates the component class.

Table 2-37. CIDR1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-4	CLASS	R	1h	Class of the component. For example, ROM table, CoreSight component and so on. Constitutes bits [15:12] of the component identification.
3-0	PRMBL_1	R	0h	Contains bits [11:8] of the component identification

2.4.2.16 CIDR2 Register (Offset = FF8h) [Reset = 00000005h]

CIDR2 is shown in Table 2-38.

Return to the Summary Table.

Component ID Register 2
A component identification register, that indicates that the identification registers are present.

Table 2-38. CIDR2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	PRMBL_2	R	5h	Contains bits [23:16] of the component identification

2.4.2.17 CIDR3 Register (Offset = FFCh) [Reset = 000000B1h]

CIDR3 is shown in Table 2-39.

Return to the Summary Table.

Component ID Register 3
A component identification register, that indicates that the identification registers are present.

Table 2-39. CIDR3 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	PRMBL_3	R	B1h	Contains bits [31:24] of the component identification

2.4.3 DCB Registers

Table 2-40 lists the memory-mapped registers for the DCB registers. All register offset addresses not listed in Table 2-40 should be considered as reserved locations and the register contents should not be modified.

Table 2-40. DCB Registers

Offset	Acronym	Register Name	Section
0h	DFSR	Debug Fault Status Register	Go
C0h	DHCSR	Debug Halting Control and Status Register	Go
C4h	DCRSR	Debug Core Register Selector Register	Go
C8h	DCRDR	Debug Core Register Data Register	Go
CCh	DEMCR	Debug Exception and Monitor Control Register	Go

Complex bit access types are encoded to fit into small table cells. Table 2-41 shows the codes that are used for access types in this section.

Table 2-41. DCB Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

2.4.3.1 DFSR Register (Offset = 0h) [Reset = 00000000h]

DFSR is shown in Table 2-42.

Return to the Summary Table.

Debug Fault Status Register
Use the Debug Fault Status Register to monitor external debug requests, vector catches, data watchpoint match, BKPT instruction execution and BPU comparator matches, halt requests. Write one to clear. C_DEBUGEN must be set before any bits in DFSR are updated.

Table 2-42. DFSR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-5	RESERVED	R	0h	Reserved
4	EXTERNAL	R/W	0h	External debug request flag 0x0:No edbgrq external debug request occurred 0x1:Edbgrq has halted the core
3	VCATCH	R/W	0h	Vector catch flag. When the VCATCH flag is set, a flag in the Debug Exception and Monitor Control Register is also set to indicate the type of vector catch. 0x0:No vector catch occurred 0x1:Vector catch occurred
2	DWTRAP	R/W	0h	Data Watchpoint (DW) flag. 0x0:No dw match 0x1:Dw match
1	BKPT	R/W	0h	The BKPT flag is set by the execution of the BKPT instruction or on an instruction whose address triggered the breakpoint comparator match. When the processor has halted, the return PC points to the address of the breakpointed instruction. 0x0:No bkpt instruction or hardware breakpoint match 0x1:Bkpt instruction or hardware breakpoint match
0	HALTED	R/W	0h	Halt request flag 0x0:No halt request 0x1:Halt requested by dap access to c_halt or halted with c_step asserted

2.4.3.2 DHCSR Register (Offset = C0h) [Reset = 00000000h]

DHCSR is shown in Table 2-43.

Return to the Summary Table.

Debug Halting Control and Status Register
The purpose of the Debug Halting Control and Status Register (DHCSR) is to provide status information about the state of the processor, enable core debug, halt and step the processor. For writes, 0xA05F must be written to bits [31:16], otherwise the write operation is ignored and no bits are written into the register.

Table 2-43. DHCSR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-26	RESERVED	R	0h	Reserved
25	S_RESET_ST	R	0h	Indicates that the core has been reset, or is now being reset, since the last time this bit was read. This a sticky bit that clears on read. So, reading twice and getting 1 then 0 means it was reset in the past. Reading twice and getting 1 both times means that it is currently reset and held in reset.
24	S_RETIRE_ST	R	0h	Core has retired at least part of an instruction since last read. This is a sticky bit that clears on read.
23-18	RESERVED	R	0h	Reserved
17	S_HALT	R	0h	The core is halted in debug state only if S_HALT is set.
16	S_REGRDY	R	0h	Register Read/Write to the Debug Core Register Selector Register is available. Set in response to a successful register access.
15-4	RESERVED	R	0h	Reserved
3	C_MASKINTS	R/W	0h	When this bit is set and debug is enabled, external interrupts, SysTick, and PendSV are masked. Does not affect NMI, Hard Fault or SVCall. When C_DEBUGEN = 0, this bit has no effect.
2	C_STEP	R/W	0h	Causes a debug event on any instruction or exception being executed, resulting in the core single stepping.
1	C_HALT	R/W	0h	Halts the core. This bit is set automatically when the core triggers a debug event, for example, on a breakpoint. This bit clears on core reset. When C_DEBUGEN = 0, this bit has no effect.
0	C_DEBUGEN	R/W	0h	Enables or disable debug 0h = Debug disabled 1h = Debug enabled

2.4.3.3 DCRSR Register (Offset = C4h) [Reset = 00000000h]

DCRSR is shown in Table 2-44.

Return to the Summary Table.

Debug Core Register Selector Register
The purpose of the Debug Core Register Selector Register (DCRSR) is to select the processor register to transfer data to or from.

Table 2-44. DCRSR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-17	RESERVED	R	0h	Reserved
16	REGWnR	W	0h	Register Write-not-Read 0x0:Read 0x1:Write
15-5	RESERVED	R	0h	Reserved
4-0	REGSEL	W	0h	Select processor register 0x00:R0 0x01:R1 0x02:R2 0x03:R3 0x04:R4 0x05:R5 0x06:R6 0x07:R7 0x08:R8 0x09:R9 0x0a:R10 0x0b:R11 0x0c:R12 0x0d:Current sp 0x0e:Lr 0x0f:Debug Return Address 0x10:Xpsr flags, execution number, and state information 0x11:Msp (main sp) 0x12:Psp (process sp) 0x14:Control (dcrdr[25:24]), primask (dcrdr[0])

2.4.3.4 DCRDR Register (Offset = C8h) [Reset = 00000000h]

DCRDR is shown in Table 2-45.

Return to the Summary Table.

Debug Core Register Data Register
The purpose of the Debug Core Register Data Register (DCRDR) is to hold data read from or written to core registers.

Table 2-45. DCRDR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	DBGTMP	R/W	0h	Data temporary cache, for reading and writing registers.

2.4.3.5 DEMCR Register (Offset = CCh) [Reset = 00000000h]

DEMCR is shown in Table 2-46.

Return to the Summary Table.

Debug Exception and Monitor Control Register
The purpose of the Debug Exception and Monitor Control Register (DEMCR) is: Global enable for the DW unit, Vector catching (that is, causes debug entry on execution of a specified vector.)

Table 2-46. DEMCR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-25	RESERVED	R	0h	Reserved
24	DWTENA	R/W	0h	Global enable or disable for the DW unit 0x0:Dw unit disabled. watchpoint cannot halt the core. the dw pcsr reads as oxffffffff. 0x1:Dw unit enabled
23-11	RESERVED	R	0h	Reserved
10	VC_HARDERR	R/W	0h	Debug trap on a Hard Fault
9-1	RESERVED	R	0h	Reserved
0	VC_CORERESET	R/W	0h	Reset Vector Catch. Halt running system if HRESETn is asserted

2.4.4 SCB Registers

Table 2-47 lists the memory-mapped registers for the SCB registers. All register offset addresses not listed in Table 2-47 should be considered as reserved locations and the register contents should not be modified.

Table 2-47. SCB Registers

Offset	Acronym	Register Name	Section
0h	CPUID	CPUID Base Register	Go
4h	ICSR	Interrupt Control State Register	Go
8h	VTOR	Vector Table Offset Register	Go
Ch	AIRCR	Application Interrupt and Reset Control Register	Go
10h	SCR	System Control Register	Go
14h	CCR	Configuration and Control Register	Go
1Ch	SHPR2	System Handler Priority Register 2	Go
20h	SHPR3	System Handler Priority Register 3	Go
24h	SHCSR	System Handler Control and State Register	Go

Complex bit access types are encoded to fit into small table cells. Table 2-48 shows the codes that are used for access types in this section.

Table 2-48. SCB Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

2.4.4.1 CPUID Register (Offset = 0h) [Reset = 410CC601h]

CPUID is shown in Table 2-49.

Return to the Summary Table.

CPUID Base Register
Read the CPU ID Base Register to determine: the ID number of the processor core, the version number of the processor core, the implementation details of the processor core.

Table 2-49. CPUID Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	IMPLEMENTER	R	41h	Implementor code: 0x41 = ARM
23-20	VARIANT	R	0h	Implementation defined variant number: 0x0 (for r0)
19-16	CONSTANT	R	Ch	Reads as 0xC
15-4	PARTNO	R	C60h	Number of processor within family: 0xC20
3-0	REVISION	R	1h	Implementation defined revision number: 0x1 = processor p1 revision.

2.4.4.2 ICSR Register (Offset = 4h) [Reset = 00000000h]

ICSR is shown in Table 2-50.

Return to the Summary Table.

Interrupt Control State Register
Use the Interrupt Control State Register to set a pending Non-Maskable Interrupt (NMI), set or clear a pending PendSV, set or clear a pending SysTick, check for pending exceptions, check the vector number of the highest priority pended exception, check the vector number of the active exception.

Table 2-50. ICSR Register Field Descriptions

Bit	Field	Type	Reset	Description
31	NMIPENDSET	R/W	0h	Setting this bit will activate an NMI. Since NMI is the highest priority exception, it will activate as soon as it is registered. Reads back with current state (1 if Pending, 0 if not). 0x0:No effect 0x1:Set pending nmi
30-29	RESERVED	R	0h	Reserved
28	PENDSVSET	R/W	0h	Set pending PendSV bit. On reads this bit returns the pending state of PendSV 0x0:No effect 0x1:Set pending pendsv
27	PENDSVCLR	W	0h	Clear pending PendSV bit 0x0:No effect 0x1:Clear pending pendsv
26	PENDSTSET	R/W	0h	Set a pending SysTick bit. On reads this bit returns the pending state of SysTick. 0x0:No effect 0x1:Set pending systick
25	PENDSTCLR	W	0h	Clear pending SysTick bit 0x0:No effect 0x1:Clear pending systick
24	RESERVED	R	0h	Reserved
23	ISRPREEMPT	R	0h	The system can only access this bit when the core is halted. It indicates that a pending interrupt is to be taken in the next running cycle. If C_MASKINTS is clear in the Debug Halting Control and Status Register, the interrupt is serviced. 0x0:A pending exception is not serviced. 0x1:A pending exception is serviced on exit from the debug halt state
22	ISRPENDING	R	0h	External interrupt pending flag 0x0:Interrupt not pending 0x1:Interrupt pending
21	RESERVED	R	0h	Reserved
20-12	VECTPENDING	R	0h	Indicates the exception number for the highest priority pending exception: 0 = no pending exceptions. Non zero = The pending state includes the effect of memory-mapped enable and mask registers. It does not include the PRIMASK special-purpose register qualifier.
11-9	RESERVED	R	0h	Reserved
8-0	VECTACTIVE	R	0h	Active exception number field. Reset clears the VECTACTIVE field.

2.4.4.3 VTOR Register (Offset = 8h) [Reset = 00000000h]

VTOR is shown in Table 2-51.

Return to the Summary Table.

Vector Table Offset Register
The VTOR holds the vector table offset address.

Table 2-51. VTOR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	TBLOFF	R/W	0h	Bits [31:8] of the indicate the vector table offset address.
7-0	RESERVED	R	0h	Reserved

2.4.4.4 AIRCR Register (Offset = Ch) [Reset = FA058000h]

AIRCR is shown in Table 2-52.

Return to the Summary Table.

Application Interrupt and Reset Control Register
Use the Application Interrupt and Reset Control Register to: determine data endianness, clear all active state information from debug halt mode, request a system reset.

Table 2-52. AIRCR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	VECTKEY	W	FA05h	Register key. To write to other parts of this register, you must ensure 0x5FA is written into the VECTKEY field.
15	ENDIANESS	R	1h	Data endianness bit. The read value depends on the endian configuration implemented 0x0:Little endian 0x1:Be-8 big-endian
14-3	RESERVED	R	0h	Reserved
2	SYSRESETREQ	W	0h	Writing 1 to this bit causes the SYSRESETREQ signal to the outer system to be asserted to request a reset. The intention is to force a large system reset of all major components except for debug. The C_HALT bit in the DHCSR is cleared as a result of the system reset requested. The debugger does not lose contact with the device.
1	VECTCLRACTIVE	W	0h	Clears all active state information for fixed and configurable exceptions. This bit: is self-clearing, can only be set by the DAP when the core is halted. When set: clears all active exception status of the processor, forces a return to Thread mode, forces an IPSR of 0. A debugger must re-initialize the stack.
0	RESERVED	R	0h	Reserved

2.4.4.5 SCR Register (Offset = 10h) [Reset = 00000000h]

SCR is shown in Table 2-53.

Return to the Summary Table.

System Control Register
System Control Register. Use the System Control Register for power-management functions: signal to the system when the processor can enter a low power state, control how the processor enters and exits low power states.

Table 2-53. SCR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-5	RESERVED	R	0h	Reserved
4	SEVONPEND	R/W	0h	When enabled, this causes WFE to wake up when an interrupt moves from inactive to pended. Otherwise, WFE only wakes up from an event signal, external and SEV instruction generated. The event input, RXEV, is registered even when not waiting for an event, and so effects the next WFE.
3	RESERVED	R	0h	Reserved
2	SLEEPDEEP	R/W	0h	Sleep deep bit. 0h = not OK to turn off system clock 1h = indicates to the system that Cortex-M0 clock can be stopped. Setting this bit causes the SLEEPDEEP port to be asserted when the processor can be stopped.
1	SLEEPONEXIT	R/W	0h	Sleep on exit when returning from Handler mode to Thread mode. Enables interrupt driven applications to avoid returning to empty main application. 0h = Do not sleep when returning to thread mode 1h = Sleep on return to thread mode
0	RESERVED	R	0h	Reserved

2.4.4.6 CCR Register (Offset = 14h) [Reset = 00000208h]

CCR is shown in Table 2-54.

Return to the Summary Table.

Configuration and Control Register
The Configuration and Control Register permanently enables stack alignment and causes unaligned accesses to result in a Hard Fault.

Table 2-54. CCR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-10	RESERVED	R	0h	Reserved
9	STKALIGN	R	1h	Always set to 1. On exception entry, all exceptions are entered with 8-byte stack alignment and the context to restore it is saved. The SP is restored on the associated exception return.
8-4	RESERVED	R	0h	Reserved
3	UNALIGN_TRP	R	1h	Indicates that all unaligned accesses results in a Hard Fault. Trap for unaligned access is fixed at 1.
2-0	RESERVED	R	0h	Reserved

2.4.4.7 SHPR2 Register (Offset = 1Ch) [Reset = 00000000h]

SHPR2 is shown in Table 2-55.

Return to the Summary Table.

System Handler Priority Register 2
System handlers are a special class of exception handler that can have their priority set to any of the priority levels. Use the System Handler Priority Register 2 to set the priority of SVCall.

Table 2-55. SHPR2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-30	PRI_11	R/W	0h	Priority of system handler 11, SVCall
29-0	RESERVED	R	0h	Reserved

2.4.4.8 SHPR3 Register (Offset = 20h) [Reset = 00000000h]

SHPR3 is shown in Table 2-56.

Return to the Summary Table.

System Handler Priority Register 3
System handlers are a special class of exception handler that can have their priority set to any of the priority levels. Use the System Handler Priority Register 3 to set the priority of PendSV and SysTick.

Table 2-56. SHPR3 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-30	PRI_15	R/W	0h	Priority of system handler 15, SysTick
29-24	RESERVED	R	0h	Reserved
23-22	PRI_14	R/W	0h	Priority of system handler 14, PendSV
21-0	RESERVED	R	0h	Reserved

2.4.4.9 SHCSR Register (Offset = 24h) [Reset = 00000000h]

SHCSR is shown in Table 2-57.

Return to the Summary Table.

System Handler Control and State Register
Use the System Handler Control and State Register to determine or clear the pending status of SVCall.

Table 2-57. SHCSR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15	SVCALLPENDED	R/W	0h	Reads as 1 if SVCall is Pending. Write 1 to set pending SVCall, write 0 to clear pending SVCall.
14-0	RESERVED	R	0h	Reserved

2.4.5 SCSCS Registers

Table 2-58 lists the memory-mapped registers for the SCSCS registers. All register offset addresses not listed in Table 2-58 should be considered as reserved locations and the register contents should not be modified.

Table 2-58. SCSCS Registers

Offset	Acronym	Register Name	Section
10h	PIDR4	Peripheral ID Register 4	Go
14h	PIDR5	Peripheral ID Register 5	Go
18h	PIDR6	Peripheral ID Register 6	Go
1Ch	PIDR7	Peripheral ID Register 7	Go
20h	PIDR0	Peripheral ID Register 0	Go
24h	PIDR1	Peripheral ID Register 1	Go
28h	PIDR2	Peripheral ID Register 2	Go
2Ch	PIDR3	Peripheral ID Register 3	Go
30h	CIDR0	Component ID Register 0	Go
34h	CIDR1	Component ID Register 1	Go
38h	CIDR2	Component ID Register 2	Go
3Ch	CIDR3	Component ID Register 3	Go

Complex bit access types are encoded to fit into small table cells. Table 2-59 shows the codes that are used for access types in this section.

Table 2-59. SCSCS Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Reset or Default Value		
-n		Value after reset or the default value

2.4.5.1 PIDR4 Register (Offset = 10h) [Reset = 00000004h]

PIDR4 is shown in Table 2-60.

Return to the Summary Table.

Peripheral ID Register 4
Part of the set of Peripheral Identification registers. Contains part of the designer identity and the memory footprint indicator.

Table 2-60. PIDR4 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-4	SIZE	R	0h	This is a 4-bit value that indicates the total contiguous size of the memory window used by this component in powers of 2 from the standard 4KB. If a component only requires the standard 4KB then this should read as 0x0, 4KB only, for 8KB set to 0x1, 16KB == 0x2, 32KB == 0x3, and so on.
3-0	DES_2	R	4h	Number of JEDEC continuation codes. Indicates the designer of the component (along with the identity code)

2.4.5.2 PIDR5 Register (Offset = 14h) [Reset = 00000000h]

PIDR5 is shown in Table 2-61.

Return to the Summary Table.

Peripheral ID Register 5
Reserved

Table 2-61. PIDR5 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	RESERVED	R	0h	Reserved

2.4.5.3 PIDR6 Register (Offset = 18h) [Reset = 00000000h]

PIDR6 is shown in Table 2-62.

Return to the Summary Table.

Peripheral ID Register 6
Reserved

Table 2-62. PIDR6 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	RESERVED	R	0h	Reserved

2.4.5.4 PIDR7 Register (Offset = 1Ch) [Reset = 00000000h]

PIDR7 is shown in Table 2-63.

Return to the Summary Table.

Peripheral ID Register 7
Reserved

Table 2-63. PIDR7 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	RESERVED	R	0h	Reserved

2.4.5.5 PIDR0 Register (Offset = 20h) [Reset = 00000008h]

PIDR0 is shown in Table 2-64.

Return to the Summary Table.

Peripheral ID Register 0
Part of the set of Peripheral Identification registers. Contains part of the designer specific part number.

Table 2-64. PIDR0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	PART_0	R	8h	Bits [7:0] of the component's part number. This is selected by the designer of the component.

2.4.5.6 PIDR1 Register (Offset = 24h) [Reset = 000000B0h]

PIDR1 is shown in Table 2-65.

Return to the Summary Table.

Peripheral ID Register 1
Part of the set of Peripheral Identification registers. Contains part of the designer specific part number and part of the designer identity.

Table 2-65. PIDR1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-4	DES_0	R	Bh	Bits [3:0] of the JEDEC identity code indicating the designer of the component (along with the continuation code)
3-0	PART_1	R	0h	Bits [11:8] of the component's part number. This is selected by the designer of the component.

2.4.5.7 PIDR2 Register (Offset = 28h) [Reset = 0000000Bh]

PIDR2 is shown in Table 2-66.

Return to the Summary Table.

Peripheral ID Register 2
Part of the set of Peripheral Identification registers. Contains part of the designer identity and the product revision.

Table 2-66. PIDR2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-4	REVISION	R	0h	The Revision field is an incremental value starting at 0x0 for the first design of this component. This only increases by 1 for both major and minor revisions and is simply used as a look-up to establish the exact major/minor revision.
3	JEDEC	R	1h	Always set. Indicates that a JEDEC assigned value is used
2-0	DES_1	R	3h	Bits [6:4] of the JEDEC identity code indicating the designer of the component (along with the continuation code)

2.4.5.8 PIDR3 Register (Offset = 2Ch) [Reset = 00000000h]

PIDR3 is shown in Table 2-67.

Return to the Summary Table.

Peripheral ID Register 3
Part of the set of Peripheral Identification registers. Contains the RevAnd and Customer Modified fields.

Table 2-67. PIDR3 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-4	REVAND	R	0h	This field indicates minor errata fixes specific to this design, for example metal fixes after implementation. In most cases this field is zero. It is recommended that component designers ensure this field can be changed by a metal fix if required, for example by driving it from registers that reset to zero.
3-0	CMOD	R	0h	Where the component is reusable IP, this value indicates if the customer has modified the behavior of the component. In most cases this field is zero.

2.4.5.9 CIDR0 Register (Offset = 30h) [Reset = 0000000Dh]

CIDR0 is shown in Table 2-68.

Return to the Summary Table.

Component ID Register 0
A component identification register, that indicates that the identification registers are present.

Table 2-68. CIDR0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	PRMBL_0	R	Dh	Contains bits [7:0] of the component identification

2.4.5.10 CIDR1 Register (Offset = 34h) [Reset = 000000E0h]

CIDR1 is shown in Table 2-69.

Return to the Summary Table.

Component ID Register 1
A component identification register, that indicates that the identification registers are present. This register also indicates the component class.

Table 2-69. CIDR1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-4	CLASS	R	Eh	Class of the component. For example, ROM table, CoreSight component and so on. Constitutes bits [15:12] of the component identification.
3-0	PRMBL_1	R	0h	Contains bits [11:8] of the component identification

2.4.5.11 CIDR2 Register (Offset = 38h) [Reset = 00000005h]

CIDR2 is shown in Table 2-70.

Return to the Summary Table.

Component ID Register 2
A component identification register, that indicates that the identification registers are present.

Table 2-70. CIDR2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	PRMBL_2	R	5h	Contains bits [23:16] of the component identification

2.4.5.12 CIDR3 Register (Offset = 3Ch) [Reset = 000000B1h]

CIDR3 is shown in Table 2-71.

Return to the Summary Table.

Component ID Register 3
A component identification register, that indicates that the identification registers are present.

Table 2-71. CIDR3 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	PRMBL_3	R	B1h	Contains bits [31:24] of the component identification

2.4.6 NVIC Registers

Table 2-72 lists the memory-mapped registers for the NVIC registers. All register offset addresses not listed in Table 2-72 should be considered as reserved locations and the register contents should not be modified.

Table 2-72. NVIC Registers

Offset	Acronym	Register Name	Section
100h	ISER	Interrupt Set-Enable Register	Go
180h	ICER	Interrupt Clear-Enable Register	Go
200h	ISPR	Interrupt Set-Pending Register	Go
280h	ICPR	Interrupt Clear-Pending Register	Go
400h	IPR0	Interrupt Priority Register 0	Go
404h	IPR1	Interrupt Priority Register 1	Go
408h	IPR2	Interrupt Priority Register 2	Go
40Ch	IPR3	Interrupt Priority Register 3	Go
410h	IPR4	Interrupt Priority Register 4	Go
414h	IPR5	Interrupt Priority Register 5	Go
418h	IPR6	Interrupt Priority Register 6	Go
41Ch	IPR7	Interrupt Priority Register 7	Go

Complex bit access types are encoded to fit into small table cells. Table 2-73 shows the codes that are used for access types in this section.

Table 2-73. NVIC Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

2.4.6.1 ISER Register (Offset = 100h) [Reset = 00000000h]

ISER is shown in Table 2-74.

Return to the Summary Table.

Interrupt Set-Enable Register
Use the Interrupt Set-Enable Register to enable interrupts and determine which interrupts are currently enabled.

Table 2-74. ISER Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	SETENA	R/W	0h	Writing 0 to a SETENA bit has no effect, writing 1 to a bit enables the corresponding interrupt. Reading the bit returns its current enable state. Reset clears the SETENA fields.

2.4.6.2 ICER Register (Offset = 180h) [Reset = 00000000h]

ICER is shown in Table 2-75.

Return to the Summary Table.

Interrupt Clear-Enable Register
Use the Interrupt Clear-Enable Registers to disable interrupts and determine which interrupts are currently enabled.

Table 2-75. ICER Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	CLRENA	R/W	0h	Writing 0 to a CLRENA bit has no effect, writing 1 to a bit disables the corresponding interrupt. Reading the bit returns its current enable state. Reset clears the CLRENA field.

2.4.6.3 ISPR Register (Offset = 200h) [Reset = 00000000h]

ISPR is shown in Table 2-76.

Return to the Summary Table.

Interrupt Set-Pending Register
Use the Interrupt Set-Pending Register to force interrupts into the pending state and determine which interrupts are currently pending

Table 2-76. ISPR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	SETPEND	R/W	0h	Interrupt set-pending bits for a: Write: 1 = pend interrupt 0 = no effect Read: 1 = interrupt is pending 0 = interrupt is not pending.

2.4.6.4 ICPR Register (Offset = 280h) [Reset = 00000000h]

ICPR is shown in Table 2-77.

Return to the Summary Table.

Interrupt Clear-Pending Register
Use the Interrupt Clear-Pending Register to clear pending interrupts and determine which interrupts are currently pending.

Table 2-77. ICPR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	CLRPEND	R/W	0h	Interrupt clear-pending bits: Write: 1 = clear interrupt pending bit, 0 = no effect Read: 1 = interrupt is pending 0 = interrupt is not pending.

2.4.6.5 IPR0 Register (Offset = 400h) [Reset = 00000000h]

IPR0 is shown in Table 2-78.

Return to the Summary Table.

Interrupt Priority Register 0
Use the Interrupt Priority Registers to assign a priority from 0 to 3 to each of the available interrupts. 0 is the highest priority, and 3 is the lowest.

Table 2-78. IPR0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-30	IP_3	R/W	0h	Priority of interrupt 3
29-24	RESERVED	R	0h	Reserved
23-22	IP_2	R/W	0h	Priority of interrupt 2
21-16	RESERVED	R	0h	Reserved
15-14	IP_1	R/W	0h	Priority of interrupt 1
13-8	RESERVED	R	0h	Reserved
7-6	IP_0	R/W	0h	Priority of interrupt 0
5-0	RESERVED	R	0h	Reserved

2.4.6.6 IPR1 Register (Offset = 404h) [Reset = 00000000h]

IPR1 is shown in Table 2-79.

Return to the Summary Table.

Interrupt Priority Register 1
Use the Interrupt Priority Registers to assign a priority from 0 to 3 to each of the available interrupts. 0 is the highest priority, and 3 is the lowest.

Table 2-79. IPR1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-30	IP_7	R/W	0h	Priority of interrupt 7
29-24	RESERVED	R	0h	Reserved
23-22	IP_6	R/W	0h	Priority of interrupt 6
21-16	RESERVED	R	0h	Reserved
15-14	IP_5	R/W	0h	Priority of interrupt 5
13-8	RESERVED	R	0h	Reserved
7-6	IP_4	R/W	0h	Priority of interrupt 4
5-0	RESERVED	R	0h	Reserved

2.4.6.7 IPR2 Register (Offset = 408h) [Reset = 00000000h]

IPR2 is shown in Table 2-80.

Return to the Summary Table.

Interrupt Priority Register 2
Use the Interrupt Priority Registers to assign a priority from 0 to 3 to each of the available interrupts. 0 is the highest priority, and 3 is the lowest.

Table 2-80. IPR2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-30	IP_11	R/W	0h	Priority of interrupt 11
29-24	RESERVED	R	0h	Reserved
23-22	IP_10	R/W	0h	Priority of interrupt 10
21-16	RESERVED	R	0h	Reserved
15-14	IP_9	R/W	0h	Priority of interrupt 9
13-8	RESERVED	R	0h	Reserved
7-6	IP_8	R/W	0h	Priority of interrupt 8
5-0	RESERVED	R	0h	Reserved

2.4.6.8 IPR3 Register (Offset = 40Ch) [Reset = 00000000h]

IPR3 is shown in Table 2-81.

Return to the Summary Table.

Interrupt Priority Register 3
Use the Interrupt Priority Registers to assign a priority from 0 to 3 to each of the available interrupts. 0 is the highest priority, and 3 is the lowest.

Table 2-81. IPR3 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-30	IP_15	R/W	0h	Priority of interrupt 15
29-24	RESERVED	R	0h	Reserved
23-22	IP_14	R/W	0h	Priority of interrupt 14
21-16	RESERVED	R	0h	Reserved
15-14	IP_13	R/W	0h	Priority of interrupt 13
13-8	RESERVED	R	0h	Reserved
7-6	IP_12	R/W	0h	Priority of interrupt 12
5-0	RESERVED	R	0h	Reserved

2.4.6.9 IPR4 Register (Offset = 410h) [Reset = 00000000h]

IPR4 is shown in Table 2-82.

Return to the Summary Table.

Interrupt Priority Register 4
Use the Interrupt Priority Registers to assign a priority from 0 to 3 to each of the available interrupts. 0 is the highest priority, and 3 is the lowest.

Table 2-82. IPR4 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-30	IP_19	R/W	0h	Priority of interrupt 19
29-24	RESERVED	R	0h	Reserved
23-22	IP_18	R/W	0h	Priority of interrupt 18
21-16	RESERVED	R	0h	Reserved
15-14	IP_17	R/W	0h	Priority of interrupt 17
13-8	RESERVED	R	0h	Reserved
7-6	IP_16	R/W	0h	Priority of interrupt 16
5-0	RESERVED	R	0h	Reserved

2.4.6.10 IPR5 Register (Offset = 414h) [Reset = 00000000h]

IPR5 is shown in Table 2-83.

Return to the Summary Table.

Interrupt Priority Register 5
Use the Interrupt Priority Registers to assign a priority from 0 to 3 to each of the available interrupts. 0 is the highest priority, and 3 is the lowest.

Table 2-83. IPR5 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-30	IP_23	R/W	0h	Priority of interrupt 23
29-24	RESERVED	R	0h	Reserved
23-22	IP_22	R/W	0h	Priority of interrupt 22
21-16	RESERVED	R	0h	Reserved
15-14	IP_21	R/W	0h	Priority of interrupt 21
13-8	RESERVED	R	0h	Reserved
7-6	IP_20	R/W	0h	Priority of interrupt 20
5-0	RESERVED	R	0h	Reserved

2.4.6.11 IPR6 Register (Offset = 418h) [Reset = 00000000h]

IPR6 is shown in Table 2-84.

Return to the Summary Table.

Interrupt Priority Register 6
Use the Interrupt Priority Registers to assign a priority from 0 to 3 to each of the available interrupts. 0 is the highest priority, and 3 is the lowest.

Table 2-84. IPR6 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-30	IP_27	R/W	0h	Priority of interrupt 27
29-24	RESERVED	R	0h	Reserved
23-22	IP_26	R/W	0h	Priority of interrupt 26
21-16	RESERVED	R	0h	Reserved
15-14	IP_25	R/W	0h	Priority of interrupt 25
13-8	RESERVED	R	0h	Reserved
7-6	IP_24	R/W	0h	Priority of interrupt 24
5-0	RESERVED	R	0h	Reserved

2.4.6.12 IPR7 Register (Offset = 41Ch) [Reset = 00000000h]

IPR7 is shown in Table 2-85.

Return to the Summary Table.

Interrupt Priority Register 7
Use the Interrupt Priority Registers to assign a priority from 0 to 3 to each of the available interrupts. 0 is the highest priority, and 3 is the lowest.

Table 2-85. IPR7 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-30	IP_31	R/W	0h	Priority of interrupt 31
29-24	RESERVED	R	0h	Reserved
23-22	IP_30	R/W	0h	Priority of interrupt 30
21-16	RESERVED	R	0h	Reserved
15-14	IP_29	R/W	0h	Priority of interrupt 29
13-8	RESERVED	R	0h	Reserved
7-6	IP_28	R/W	0h	Priority of interrupt 28
5-0	RESERVED	R	0h	Reserved

2.4.7 SYSTICK Registers

Table 2-86 lists the memory-mapped registers for the SYSTICK registers. All register offset addresses not listed in Table 2-86 should be considered as reserved locations and the register contents should not be modified.

Table 2-86. SYSTICK Registers

Offset	Acronym	Register Name	Section
0h	CSR	SysTick Control and Status Register	Go
4h	RVR	SysTick Reload Value Register	Go
8h	CVR	SysTick Current Value Register	Go
Ch	CALIB	SysTick Calibration Value Register	Go

Complex bit access types are encoded to fit into small table cells. Table 2-87 shows the codes that are used for access types in this section.

Table 2-87. SYSTICK Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

2.4.7.1 CSR Register (Offset = 0h) [Reset = 00000000h]

CSR is shown in Table 2-88.

Return to the Summary Table.

SysTick Control and Status Register
Use the SysTick Control and Status Register to enable the SysTick features.

Table 2-88. CSR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-17	RESERVED	R	0h	Reserved
16	COUNTFLAG	R	0h	Returns 1 if timer counted to 0 since last time this was read. Clears on read by application or debugger.
15-3	RESERVED	R	0h	Reserved
2	CLKSOURCE	R	0h	SysTick clock source. Always reads as one if STCALIB reports NOREF. 0x0:Systick driven by external reference clock. 0x1:Systick driven by processor clock
1	TICKINT	R/W	0h	0x0:Counting down to zero does not pend the systick handler. software can use countflag to determine if the systick handler has ever counted to zero. 0x1:Counting down to zero pends the systick handler.
0	ENABLE	R/W	0h	Enable SysTick counter 0x0:Counter disabled 0x1:Counter operates in a multi-shot way. that is, counter loads with the reload value and then begins counting down. on reaching 0, it sets the countflag to 1 and optionally pends the systick handler, based on tickint. it then loads the reload value again, and begins counting.

2.4.7.2 RVR Register (Offset = 4h) [Reset = 00000000h]

RVR is shown in Table 2-89.

Return to the Summary Table.

SysTick Reload Value Register
Use the SysTick Reload Value Register to specify the start value to load into the current value register when the counter reaches 0. It can be any value between 0 and 0x00FFFFFF. A start value of 0 is possible, but has no effect because the SysTick interrupt and COUNTFLAG are activated when counting from 1 to 0. The reset value of this register is UNKNOWN.

Table 2-89. RVR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	RESERVED	R	0h	Reserved
23-0	RELOAD	R/W	0h	Value to load into the SysTick Current Value Register when the counter reaches 0.

2.4.7.3 CVR Register (Offset = 8h) [Reset = 00000000h]

CVR is shown in Table 2-90.

Return to the Summary Table.

SysTick Current Value Register
Use the SysTick Current Value Register to find the current value in the register. The reset value of this register is UNKNOWN.

Table 2-90. CVR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	RESERVED	R	0h	Reserved
23-0	CURRENT	R/W	0h	Reads return the current value of the SysTick counter. This register is write-clear. Writing to it with any value clears the register to 0. Clearing this register also clears the COUNTFLAG bit of the SysTick Control and Status Register.

2.4.7.4 CALIB Register (Offset = Ch) [Reset = 00000000h]

CALIB is shown in Table 2-91.

Return to the Summary Table.

SysTick Calibration Value Register
Use the SysTick Calibration Value Register to enable software to scale to any required speed using divide and multiply.

Table 2-91. CALIB Register Field Descriptions

Bit	Field	Type	Reset	Description
31	NOREF	R	0h	If reads as 1, the Reference clock is not provided - the CLKSOURCE bit of the SysTick Control and Status register will be forced to 1 and cannot be cleared to 0.
30	SKEW	R	0h	If reads as 1, the calibration value for 10ms is inexact (due to clock frequency).
29-24	RESERVED	R	0h	Reserved
23-0	TENMS	R	0h	An optional Reload value to be used for 10ms (100Hz) timing, subject to system clock skew errors. If the value reads as 0, the calibration value is not known.

3.1 Memory Map

All CC23xx devices share a common platform memory map. Peripherals are assigned a fixed address space and have the same address space on all devices within the family. The memory map is compliant with the standard Arm Cortex-M memory regions.

Table 3-1. Memory Map

Module	Module Name	Base Address
FLASHMEM	Program Flash Memory	0x0000 0000
ROM	System ROM	0x0F00 0000
SRAM	Static RAM	0x2000 0000
PMCTL	Power Mode Controller	0x4000 0000
CKMD	Clock Manager	0x4000 1000
RTC	Real-Time Clock	0x4000 2000
IOC	I/O Controller	0x4000 3000
SYS0	LPCOMP, System Control and Trim	0x4000 4000
EVTULL	AON/ULL Event Fabric	0x4000 5000
PMUD	Power Management (BATMON and DCDC)	0x4000 6000
DBGSS	Debug Subsystem	0x4000 F000
CLKCTL	IP Clock Control	0x4002 0000
FLASH	Flash Subsystem Controller	0x4002 1000
SYSTIM	System Timer	0x4002 2000
GPIO	General Purpose I/O	0x4002 3000
VIMS	Versatile Instruction Memory System	0x4002 4000
EVTSVT	SVT/MCU Event Fabric	0x4002 5000
DMA	µDMA Controller	0x4002 6000
SPI0	Serial Peripheral Interface (SPI) 0	0x4003 0000
UART0	Universal Asynchronous Receiver Transmitter (UART) 0	0x4003 4000
I2C0	Inter-Integrated Circuit (I^2C) 0	0x4003 8000
ADC	Analog-to-Digital Converter	0x4005 0000
LGPT0	General Purpose Timer 0	0x4006 0000
LGPT1	General Purpose Timer 1	0x4006 1000
LGPT2	General Purpose Timer 2	0x4006 2000
LGPT3	General Purpose Timer 3	0x4006 3000
LRFD	Low-power Radio	0x4008 0000
PBERAM	Radio PBE RAM	0x4009 0000
BUFRAM	Radio BUF RAM	0x4009 2000
MCERAM	Radio MCE RAM	0x4009 4000
RFERAM	Radio RFE RAM	0x4009 6000

Table 3-1. Memory Map (continued)

Module	Module Name	Base Address
S2RRAM	Radio S2R RAM	0x4009 8000
AES	AES Accelerator	0x400C 0000
FCFG	Factory Configuration	0x4E00 0000
CCFG	Customer Configuration	0x4E02 0000
CPUSS	CPU Subsystem	0xE000 0000
CPU_BPU	Breakpoint Unit	0xE000 2000
CPU_NVIC	Nested Vectored Interrupt Controller	0xE000 E000
CPU_SYSTICK	CPU SysTick	0xE000 E010
CPU_SCB	System Control Block	0xE000 ED00
CPU_DCB	Debug Control Block	0xE000 ED30
CPU_SCSCS	System Control Space CoreSight™	0xE000 EFC0
CPU_ROM_TABLE	CPU ROM Table	0xE00F F000

Chapter 4

Interrupts and Events

This section describes the device exceptions, faults, and the functions of the event fabric system.

4.1 Exception Model

The processor implements advanced exception and interrupt handling, as described in the *ARMv6-M Architecture Reference Manual*.

To minimize interrupt latency, the processor abandons any load-multiple or store-multiple instruction to take any pending interrupt. On return from the interrupt handler, the processor restarts the load-multiple or store-multiple instruction from the beginning.

To reduce interrupt latency and jitter, the Cortex-M0+ processor implements both interrupt late-arrival and interrupt tail-chaining mechanisms, as defined by the ARMv6-M architecture. The worst case interrupt latency, for the highest priority active interrupt in a zero wait-state system not using jitter suppression, is 15 cycles.

4.1.1 Exception States

Each exception is in one of the following states:

- **Inactive**: The exception is not active and not pending.
- **Pending**: The exception is waiting to be serviced by the processor. An interrupt request from a peripheral or from software can change the state of the corresponding interrupt to pending.
- **Active**: An exception is being serviced by the processor but has not completed. An exception handler can interrupt the execution of another exception handler. In this case, both exceptions are in the active state.
- **Active and Pending**: The exception is being serviced by the processor, and there is a pending exception from the same source

4.1.2 Exception Types

The exception types are:

- **Reset**: Reset is invoked on power up or a warm reset. The exception model treats reset as a special form of exception. When reset is asserted, the operation of the processor stops, potentially at any point in an instruction. When reset is deasserted, execution restarts from the address provided by the reset entry in the vector table. Execution restarts in Thread mode.
- **NMI**: A Non-maskable Interrupt (NMI) can be signaled by a peripheral or triggered by software. This is the highest priority exception other than reset. It is permanently enabled and has a fixed priority of –2. NMIs cannot be:
  - masked or prevented from activation by any other exception.
  - preempted by any exception other than Reset.
- **HardFault**: A HardFault is an exception that occurs because of an error during normal or exception processing. HardFaults have a fixed priority of –1, meaning they have higher priority than any exception with configurable priority.
- **SVCall**: A supervisor call (SVC) is an exception that is triggered by the SVC instruction. In an OS environment, applications can use SVC instructions to access OS kernel functions and device drivers.
- **PendSV**: Pendable Service (PendSV) is an interrupt-driven request for system-level service. In an OS environment, use PendSV for context switching when no other exception is active.
- **SysTick**: a SysTick exception is an exception the system timer generates when it reaches zero. Software can also generate a SysTick exception. In an OS environment, the device can use this exception as system tick.
- **Interrupt (IRQ)**: An interrupt, or IRQ, is an exception signalled by a peripheral, or generated by a software request. All interrupts are asynchronous to instruction execution. In the system, peripherals use interrupts to communicate with the processor.

Table 4-1. Properties of the different exception types

Exception Number	IRQ Number	Exception Type	Priority	Vector Address	Activation
1	-	Reset	–3, the highest	0x00000004	Asynchronous
2	–14	NMI	–2	0x00000008	Asynchronous
3	–13	HardFault	–1	0x0000000C	Synchronous
4-10	–	Reserved	–	–	-

Table 4-1. Properties of the different exception types (continued)

Exception Number	IRQ Number	Exception Type	Priority	Vector Address	Activation
11	-5	SVCall	Configurable	0x0000002C	Synchronous
12-13	–	Reserved	–	–	-
14	–2	PendSV	Configurable	0x00000038	Asynchronous
15	–1	SysTick	Configurable	0x0000003C	Asynchronous
16 and above	0 and above	IRQ	Configurable	0x00000040 and above (increasing in steps of 4)	Asynchronous

Note

To simplify the software layer, the CMSIS only uses IRQ numbers and therefore uses negative values for exceptions other than interrupts.

For an asynchronous exception, other than reset, the processor can continue executing instructions between when the exception is triggered and when the processor enters the exception handler.

Privileged software can disable the exceptions that have configurable priority.

4.1.3 Exception Handlers

The processor handles exceptions with:

- **Interrupt Service Routine (ISR)**: The IRQ interrupts are the exceptions handled by ISRs.
- **Fault Handler**: HardFault is the only exception handled by the fault handler.
- **System Handlers**:NMI, PendSV, SVCall SysTick, and HardFault are all system exceptions handled by system handlers.

4.1.4 Vector Table

The vector table contains the reset value of the stack pointer, and the start addresses, also called exception vectors, for all exception handlers. Table 4-2 shows the order of the exception vectors in the vector table. The least-significant bit of each vector must be 1, indicating that the exception handler is written in Thumb code.

Table 4-2. Vector Table

Exception Number	IRQ Number	Vector	Offset
16+n	n	IRQ n	0x40 + 4n
.		.	.
.		.	.
.		.	.
18	2	IRQ2	0x48
17	1	IRQ1	0x44
16	0	IRQ0	0x40
15	-1	SysTick	0x3C
14	-2	PendSV	0x38
13		Reserved	
12			
11	-5	SVCall	0x2C

Table 4-2. Vector Table (continued)

Exception Number	IRQ Number	Vector	Offset
10		Reserved	
9			
8			
7			
6			
5			
4			
3	-13	HardFault	0x0C
2	-14	NMI	0x08
1		Reset	0x04
		Initial SP Value	0x00

The vector table is fixed at address 0x00000000.

4.1.5 Exception Priorities

All exceptions have an associated priority, with a lower priority value indicating a higher priority. All exceptions have configurable priorities except Reset, Hardfault, and NMI.

Configurable priority values are in the range 0-192, in steps of 64. The Reset, HardFault, and NMI exceptions, with fixed negative priority values, always have higher priority than any other exception.

Assigning a higher priority value to IRQ[0] and a lower priority value to IRQ[1] means that IRQ[1] has higher priority than IRQ[0]. If both IRQ[1] and IRQ[0] are asserted, IRQ[1] is processed before IRQ[0].

If multiple pending exceptions have the same priority, the pending exception with the lowest exception number takes precedence. For example, if both IRQ[0] and IRQ[1] are pending and have the same priority, then IRQ[0] is processed before IRQ[1].

When the processor is executing an exception handler, the exception handler is preempted if a higher priority exception occurs. If an exception occurs with the same priority as the exception being handled, the handler is not preempted, irrespective of the exception number. However, the status of the new interrupt changes to pending.

4.1.6 Exception Entry and Return

Descriptions of exception handling use the following terms.

- **Preemption**: When the processor is executing an exception handler, an exception can preempt the exception handler if the exemption priority is higher than the priority of the exception being handled. When one exception preempts another, the exceptions are called nested exceptions.
- **Return**: This occurs when the exception handler is completed, and:
  - there is no pending exception with sufficient priority to be serviced
  - the completed exception handler was not handling a late-arriving exception.

  The processor pops the stack and restores the processor state to the state the stack had before the interrupt occurred.
- **Tail-chaining**: This mechanism speeds up exception servicing. On completion of an exception handler, if there is a pending exception that meets the requirements for exception entry, the stack pop is skipped and control transfers to the new exception handler.
- **Late-arriving**: This mechanism speeds up preemption. If a higher priority exception occurs during state saving for a previous exception, the processor switches to handle the higher priority exception and initiates the vector fetch for that exception. State saving is not affected by late arrival because the state saved would be the same for both exceptions. On return from the exception handler of the late-arriving exception, the normal tail-chaining rules apply.

4.1.6.1 Exception Entry

Exception entry occurs when there is a pending exception with sufficient priority and either:

- the processor is in Thread mode.
- the new exception is of higher priority than the exception being handled, in which case the new exception preempts the exception being handled.

When one exception preempts another, the exceptions are nested.

Sufficient priority means the exception has greater priority than any limit set by the mask register. An exception with less priority than this is pending but is not handled by the processor.

When the processor takes an exception, unless the exception is a tail-chained or a late-arriving exception, the processor pushes information onto the current stack. This operation is referred to as stacking and the structure of eight data words is referred to as a stack frame. The stack frame contains the following information:

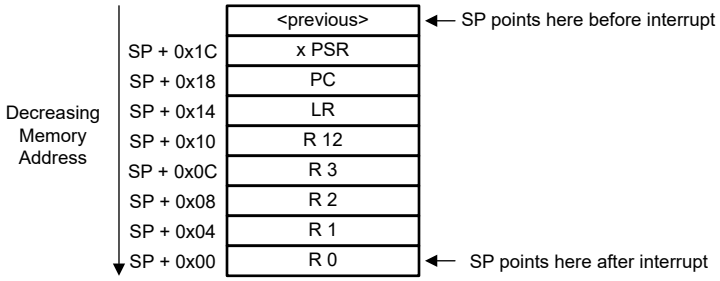


Figure 4-1. Stack Frame

Immediately after stacking, the stack pointer indicates the lowest address in the stack frame. The stack frame is aligned to a double-word address.

The stack frame includes the return address. This is the address of the next instruction in the interrupted program. This value is restored to the PC at exception return so that the interrupted program resumes.

The processor performs a vector fetch that reads the exception handler start address from the vector table. When stacking is complete, the processor starts executing the exception handler. At the same time, the processor writes an EXC_RETURN value to the LR. This indicates the stack pointer corresponding to the stack frame and the operation mode the processor was in before the entry occurred.

If no higher priority exception occurs during exception entry, the processor starts executing the exception handler and automatically changes the status of the corresponding pending interrupt to active.

If another higher priority exception occurs during exception entry, the processor starts executing the exception handler for this exception and does not change the pending status of the earlier exception. This is the late arrival case.

4.1.6.2 Exception Return

Exception return occurs when the processor is in Handler mode and execution of one of the following instructions attempts to set the PC to an EXC_RETURN value:

- a POP instruction that loads the PC.
- a BX instruction using any register.

The processor saves an EXC_RETURN value to the LR on exception entry. The exception mechanism relies on this value to detect when the processor has completed an exception handler. Bits[31:4] of an EXC_RETURN value are 0xFFFFFFF.

When the processor loads a value matching this pattern to the PC it detects that the operation is a not a normal branch operation and, instead, that the exception is complete. Therefore, it starts the exception return sequence. Bits[3:0] of the EXC_RETURN value indicate the required return stack and processor mode, as shown in Table 4-3.

Table 4-3. Exception Return Behavior

EXC_RETURN	Description
0xFFFFFFF1	Return to Handler mode. Exception return gets state from the main stack. Execution uses MSP after return.
0xFFFFFFF9	Return to Thread mode. Exception return gets state from MSP. Execution uses MSP after return.
0xFFFFFFFD	Return to Thread mode. Exception return gets state from PSP. Execution uses PSP after return.
All other values	Reserved

4.2 Fault Handling

Faults are a subset of exceptions, see Exception Model. All faults result in the HardFault exception being taken or cause lockup if they occur in the NMI or HardFault handler. The faults are:

- execution of an SVC instruction at a priority equal or higher than SVCall
- execution of a BKPT instruction without a debugger attached
- a system-generated bus error on a load or store
- execution of an instruction from an XN memory address
- execution of an instruction from a location for which the system generates a bus fault
- a system-generated bus error on a vector fetch
- execution of an Undefined instruction
- execution of an instruction when not in Thumb-State as a result of the T-bit being previously cleared to 0
- an attempted load or store to an unaligned address.

Only Reset and NMI can preempt the fixed priority HardFault handler. A HardFault can preempt any exception other than Reset, NMI, or another HardFault.

4.2.1 Lockup

The processor enters a lockup state if a fault occurs when executing the NMI or HardFault handlers, or if the system generates a bus error when unstacking the PSR on an exception return using the MSP. When the processor is in lockup state it does not execute any instructions. The processor remains in lockup state until one of the following occurs

- The processor is reset.
- A debugger halts the processor.
- An NMI occurs and the current lockup is in the HardFault handler

If the lockup state occurs in the NMI handler, a subsequent NMI does not cause the processor to leave lockup state.

4.3 Event Fabric

4.3.1 Introduction

The event fabric is a combinational router between event sources/publishers and event subscribers. The event inputs are routed to a central event-bus where a subscriber can select the appropriate events and output those as inputs to peripherals.

Several of the events (signals) are statically routed, and a small number of configurable output lines go to the event subscribers. A configurable output line from a subscriber can choose from a list of several input events available to the specific subscriber in question.

Subscribers output event signaling is identical to input signaling. That is, events are simply passed through the event fabric as presented to the input ports. Possible event types include system hardware interrupts and DMA triggers.

All ULL/AON event inputs are considered level-triggered events active high. SVT/MCU event inputs can be either active high level or pulse triggered events. Events like DMA triggers can be level-type signals.

The event fabric is not a peripheral in itself, but rather a block of routing between the peripherals and more. The lines that have configurable inputs are controlled by selection registers that are connected to a MUX, which forward the selected input in the subscriber to the peripherals.

Figure 4-2 shows a simple illustration of the event fabric concept

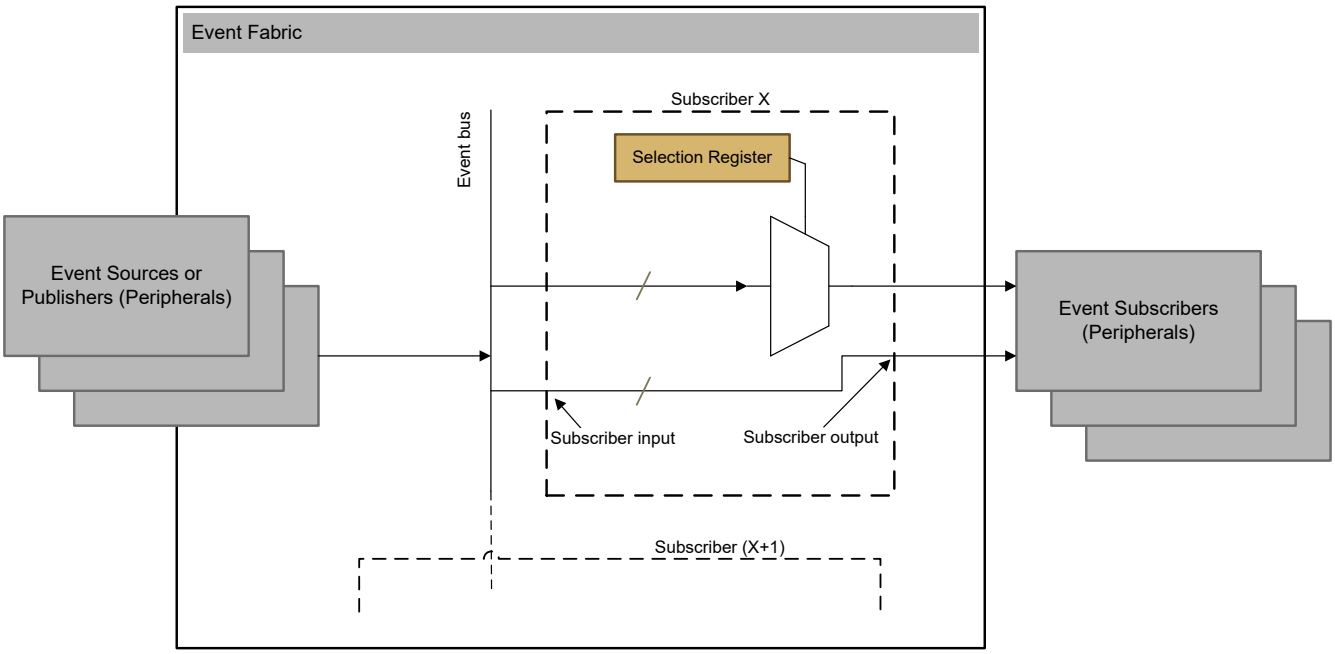


Figure 4-2. Event Fabric Concept

4.3.2 Overview

There are two main event fabric blocks. The MCU event fabric and the AON event fabric. The MCU event fabric is in the SVT/MCU power domain and is configured with the EVTSVT registers. The AON event fabric is in the ULL/AON domain and is configured with the EVTULL registers.

Figure 4-3 shows a simplified overview of the two modules together. The MCU event fabric is one of the subscribers to the AON event fabric.

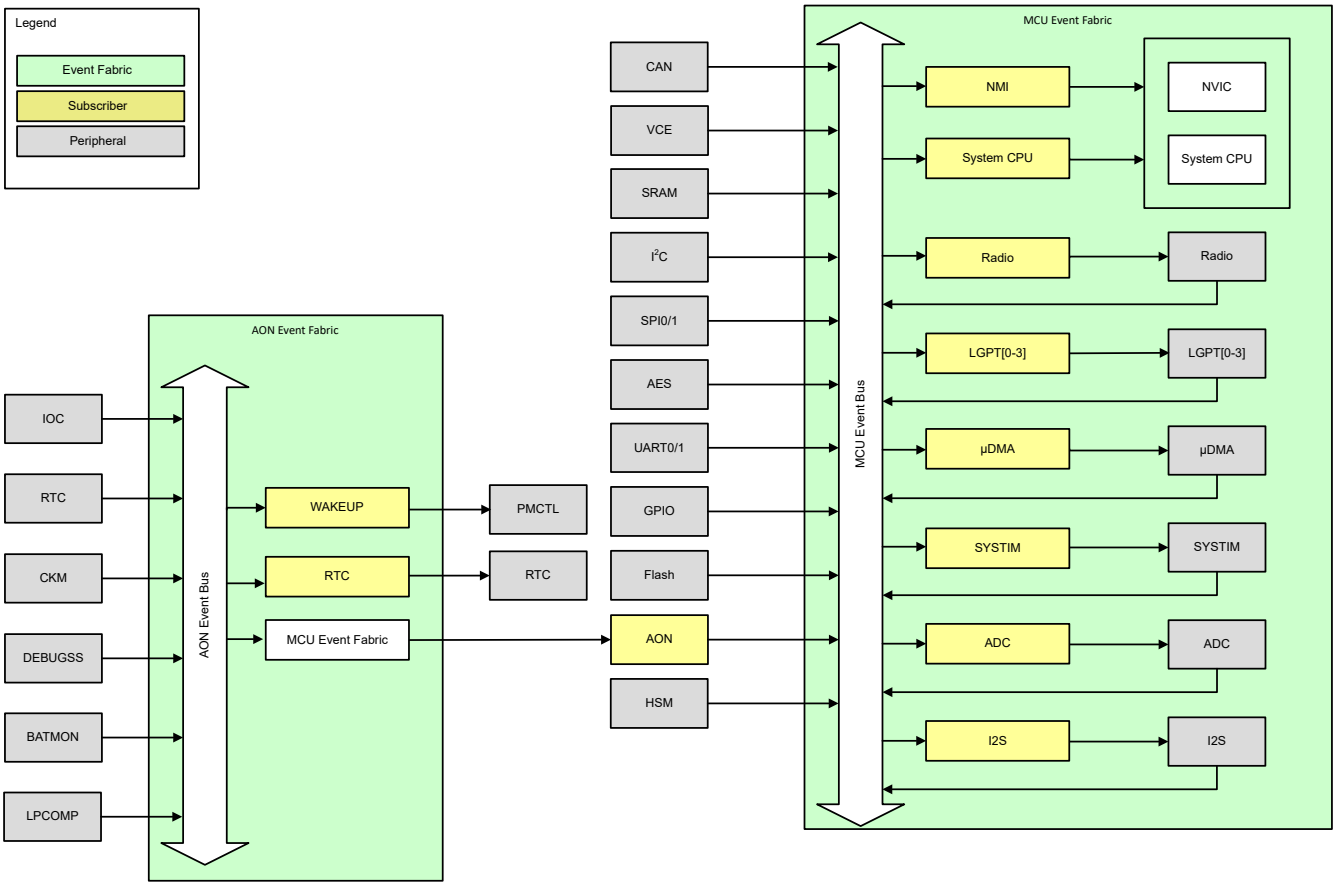


Figure 4-3. Event Fabric Overview (Simplified)

4.3.3 Registers

The event fabric has two types of registers. The first type, a configuration register, is used to control and report the selection settings for a subscriber output. For each subscriber output, an address is mapped for a read register that contains a value representing the selection of the input event currently set for that subscriber output. For non-configurable outputs, only a read-only register is implemented. A read to that address returns the static, predefined value. The second type of register in the event fabric are description registers and one register that can be used for observation of event signals on the pads.

4.3.4 AON Event Fabric

The AON event fabric resides in the AON/ULL power domain.

4.3.4.1 AON Common Input Events List

Table 4-4 lists the input events for the AON event fabric (event numbers 0x2 to 0x7). The sources for these events are considered level-triggered active high.

4.3.4.2 AON Event Subscribers

There are three subscribers in the AON event fabric as can be seen in Figure 4-3. The first subscriber is the MCU event fabric, which resides in the MCU power domain. The other two subscribers, the PMCTL and RTC, are presented in the following subsections.

4.3.4.3 Power Management Controller (PMCTL)

The PMCTL subscriber has 6 programmable events in AON event fabric, which are ORed together to form a single wake-up event to PMCTL configurable by the EVTULL.WKUPMASK register. This wake-up event triggers the wake-up of the MCU power domain from standby mode. Any of the events listed in Table 4-4 can be chosen

as input by selecting the appropriate event publisher or publishers. By default, this register is set to 0, meaning no publishers are selected to drive the wake-up event towards PMCTL.

4.3.4.4 Real Time Clock (RTC)

The RTC has a programmable event, which can be configured in the EVTULL.RTCCPTSEL register. This register can be used to choose between 6 programmable sources within AON.

4.3.4.5 AON to MCU Event Fabric

AON Event Fabric ORs 6 programmable events together to generate a single NMI (Non-maskable interrupt) from AON peripherals, configurable with the EVTULL.NMISEL register.

The 6 AON events are also exported separately as a bus to MCU Event Fabric and can be used as publishers within MCU peripherals.

Note

Since the events from AON are double synchronized within MCU event fabric before being used as publishers, care must be taken to make sure that when the AON events are cleared, a subsequent event does not get set immediately (within two CLKSVT clock cycles). This is so that a new rising edge is generated for the second event. Otherwise the second event can get missed if the cleared AON event pulse does not get synchronized within MCU event fabric before getting set again.

4.3.5 MCU Event Fabric

The MCU event fabric resides in the MCU power domain and routes signals between most of the peripherals and different internal blocks. Only a few of the subscribers in the MCU event fabric are described in this section. For more information on the remaining subscribers, refer to the specific peripheral chapters for the appropriate consumer (peripheral) for that specific subscriber.

4.3.5.1 Common Input Event List

Table 4-4 lists the input events for the MCU event fabric.

Table 4-4. Common Input Event List

Event Number	Name	Description
0x0	NONE	Always inactive
0x2	AON_PMU_COMB	PMU combined interrupt request for BATMON, interrupt flags can be found here PMUD.EVENT
0x3	AON_CKM_COMB	CKMD combined interrupt request, interrupt flags can be found here CKMD.MIS
0x4	AON_RTC_COMB	AON_RTC event, controlled by the RTC.IMASK setting
0x5	AON_DBG_COMB	DebugSS combined interrupt, interrupt flags can be found here DBGSS.MIS
0x6	AON_LPMCMP_IRQ	AON LPCMP interrupt, controlled by SYS0.LPCMPCFG
0x7	AON_IOC_COMB	IOC synchronous combined event, controlled by IOC.EVTCFG
0x8	SYSTIM_COMB	SYSTIM combined interrupt, interrupt flags are found here SYSTIM.MIS
0x9	GPIO_COMB	GPIO combined wake up interrupt, interrupt flags can be found here GPIO.MIS
0xA	GPIO_EVT	GPIO generic published event, controlled by GPIO.EVTCFG
0xB	FLASH_IRQ	Flash interrupt indicating that the flash operation has completed, interrupt flags can be found here FLASH.MIS

Table 4-4. Common Input Event List (continued)

Event Number	Name	Description
0xC	LRFD_IRQ0	LRFD combined event, interrupt flags can be found here LRFDDBELL.MIS0
0xD	LRFD_IRQ1	LRFD combined event, interrupt flags can be found here LRFDDBELL.MIS1
0xE	LRFD_IRQ2	LRFD combined event, interrupt flags can be found here LRFDDBELL.MIS2
0xF	SPI0_COMB	SPI0 combined interrupt request, interrupt flags can be found here SPI0.MIS
0x10	ADC_COMB	ADC combined interrupt request, interrupt flags can be found here ADC.MIS0
0x11	ADC_EVT	ADC general published event, interrupt flags be found here ADC.MIS1
0x12	LGPT0_COMB	LGPT0 combined interrupt, interrupt flags are found here LGPT0.MIS
0x13	LGPT1_COMB	LGPT1 combined interrupt, interrupt flags are found here LGPT1.MIS
0x14	DMA_DONE_COMB	DMA combined done interrupt, corresponding flags can be found here DMA.REQDONE
0x15	DMA_ERR	DMA bus error, corresponds to DMA.ERROR[0] STATUS bit
0x16	AES_COMB	AES accelerator combined interrupt request, interrupt flags can be found here AES.MIS
0x17	UART0_COMB	UART0 combined interrupt, interrupt flags are found here UART0.MIS
0x18	I2C0_IRQ	Interrupt event from I2C0, interrupt flags can be found here I2C0.MIS
0x19	SYSTIM_HB	SYSTIM heartbeat, can be set by SYSTIM.TIMEBIT
0x1A	SYSTIM_LT	SYSTIM interrupt driven by synchronizing LFTICK signal to SVT clock
0x1B	SYSTIM0	SYSTIM Channel 0 event, event flag is SYSTIM.MIS[0] EVT0 bit
0x1C	SYSTIM1	SYSTIM Channel 1 event, event flag is SYSTIM.MIS[1] EVT1 bit
0x1D	SYSTIM2	SYSTIM Channel 2 event, event flag is SYSTIM.MIS[2] EVT2 bit
0x1E	SYSTIM3	SYSTIM Channel 3 event, event flag is SYSTIM.MIS[3] EVT3 bit
0x1F	SYSTIM4	SYSTIM Channel 4 event, event flag is SYSTIM.MIS[4] EVT4 bit
0x20	LGPT0C0	LGPT0 compare/capture output event 0, controlled by LGPT0.C0CFG setting
0x21	LGPT0C1	LGPT0 compare/capture output event 1, controlled by LGPT0.C1CFG setting
0x22	LGPT0C2	LGPT0 compare/capture output event 2, controlled by LGPT0.C2CFG setting
0x23	LGPT0_DMA	LGPT0 DMA request event, controlled by LGPT0.DMA setting
0x24	LGPT0_ADC	LGPT0 ADC trigger event, controlled by LGPT0.ADCTRG setting
0x25	LGPT1C0	LGPT1 compare/capture output event 0, controlled by LGPT1.C0CFG setting

Table 4-4. Common Input Event List (continued)

Event Number	Name	Description
0x26	LGPT1C1	LGPT1 compare/capture output event 1, controlled by LGPT1.C1CFG setting
0x27	LGPT1C2	LGPT1 compare/capture output event 2, controlled by LGPT1.C2CFG setting
0x28	LGPT1_DMA	LGPT1 DMA request event, controlled by LGPT1.DMA setting
0x29	LGPT1_ADC	LGPT1 ADC trigger event, controlled by LGPT1.ADCTRG setting
0x2A	LRFD_EVT0	LRFD interrupt to SYSTIM, controlled by LRFDDBELL.SYSTIMOEV[3:0] SRC0 bit field
0x2B	LRFD_EVT1	LRFD interrupt to SYSTIM, controlled by LRFDDBELL.SYSTIMOEV[7:4] SRC1 bit field
0x2C	LRFD_EVT2	LRFD interrupt to SYSTIM, controlled by LRFDDBELL.SYSTIMOEV[11:8] SRC2 bit field
0x2E	LGPT2C0	LGPT2 compare/capture output event 0, controlled by LGPT2.C0CFG setting
0x2F	LGPT2C1	LGPT2 compare/capture output event 1, controlled by LGPT2.C1CFG setting
0x30	LGPT2C2	LGPT0 compare/capture output event 2, controlled by LGPT2.C2CFG setting
0x31	LGPT2_COMB	LGPT2 combined interrupt, interrupt flags are found here LGPT2.MIS
0x32	LGPT2_DMA	LGPT2 DMA request event, controlled by LGPT2.DMA setting
0x33	LGPT2_ADC	LGPT2 ADC trigger event, controlled by LGPT2.ADCTRG setting
0x34	LGPT3C0	LGPT3 compare/capture output event 0, controlled by LGPT3.C0CFG setting
0x35	LGPT3C1	LGPT3 compare/capture output event 1, controlled by LGPT3.C1CFG setting
0x36	LGPT3C2	LGPT3 compare/capture output event 2, controlled by LGPT3.C2CFG setting
0x37	LGPT3_COMB	LGPT3 combined interrupt, interrupt flags are found here LGPT3.MIS
0x38	LGPT3_DMA	LGPT3 DMA request event, controlled by LGPT3.DMA setting
0x39	LGPT3_ADC	LGPT3 ADC trigger event, controlled by LGPT3.ADCTRG setting

4.3.5.2 MCU Event Subscribers

There are five peripheral subscribers for the MCU event fabric. These subscribers are different peripherals that must be configured differently according to the purpose of those specific peripherals. The following subscribers are not described in this chapter, but rather in each of the corresponding peripheral chapters:

- Micro Direct Memory Access (µDMA) - Chapter 15
- Four Low Power General-Purpose Timers - Chapter 10
- System Timer - Chapter 11
- Radio - Chapter 22
- ADC - Chapter 17

The following two subscribers are described below:

- System CPU

- Non-maskable Interrupt (NMI) to System CPU

4.3.5.2.1 System CPU

Interrupts with vectors numbered from 16 to 34 (19 total) are sourced by the events routed in the MCU event fabric to the system CPU.

Six of the interrupts are configurable out from the sources present in Table 4-4, and 13 interrupts are fixed routes from MCU peripherals to the CPU. For information on the available routing see the EVTSVT.CPUIRQ ***n*** SEL descriptions.

4.3.5.2.2 Non-Maskable Interrupt (NMI)

The NMI subscriber has to be configured by two registers if an AON publisher needs to be used. EVTULL.NMISEL selects the AON publisher. Setting the EVTSVT.NMISEL[5:0] PUBID bit field to 0x1 routes the AON publisher to the CPU's NMI port.

EVTSVT.NMISEL can be used to select any MCU publisher event from Table 4-4 to drive this to the CPU NMI port.

4.4 Digital Test Bus (DTB)

Both MCU and AON event fabric structures include a feature to enable porting a set of selected internal signals to the pads for observation purposes.

The signals are grouped into several 16-bit buses, which can be selected through the DTB.SEL register. A value of zero within DTB.SEL disables this feature and only zeros are exported.

The signals that can be observed from within EVTULL are:

EVTULL.DTB[0] SEL bit	Bits	Observed signal
0	[15:0]	All zeros
1	[15:8]	All zeros
	[7]	AON_RTC_COMB event
	[6]	AON_PMU_COMB event
	[5]	AON_CKM_COMB event
	[4]	AON_DBG_COMB event
	[3]	AON_IOC_COMB event
	[2]	AON_LPMCMP_IRQ event
	[1]	Event selected for wakeup by EVTULL.WKUPMASK
	[0]	Event selected for ULL NMI by EVTULL.NMISEL

The signals that can be observed from within EVTSVT are:

EVTSVT.DTB[1:0] SEL bit field	Bits	Observed signal
0	[15:0]	All zeros

1	[15]	SYSTIM_COMB event
	[14]	LGPT0_COMB event
	[13]	LGPT1_COMB event
	[12]	LGPT2_COMB event
	[11]	LGPT3_COMB event
	[10]	UART0_COMB event
	[9]	I2C0_IRQ event
	[8]	SPI0_COMB event
	[7]	ADC_COMB event
	[6]	GPIO_COMB event
	[5]	FLASH_IRQ event
	[4]	AES_COMB event
	[3]	DMA_DONE_COMB event
	[2]	LRFD_EVT0 event
	[1]	LRFD_EVT1 event
	[0]	LRFD_EVT2 event
2	[15]	Event selected by EVTSVT.CPUIRQ0SEL
	[14]	Event selected by EVTSVT.CPUIRQ2SEL
	[13]	Event selected by EVTSVT.SYSTIMC1SEL
	[12]	Event selected by EVTSVT.NMISEL
	[11]	DMA burst request selected by EVTSVT.DMACH0SEL
	[10]	DMA single request selected by EVTSVT.DMACH0SEL
	[9]	DMA burst request selected by EVTSVT.DMACH1SEL
	[8]	DMA single request selected by EVTSVT.DMACH1SEL
	[7]	DMA burst request selected by EVTSVT.DMACH2SEL
	[6]	DMA single request selected by EVTSVT.DMACH2SEL
	[5]	DMA burst request selected by EVTSVT.DMACH3SEL
	[4]	DMA single request selected by EVTSVT.DMACH3SEL
	[3]	DMA burst request selected by EVTSVT.DMACH4SEL
	[2]	DMA single request selected by EVTSVT.DMACH4SEL
	[1]	DMA burst request selected by EVTSVT.DMACH5SEL
	[0]	DMA single request selected by EVTSVT.DMACH5SEL
3	[15:4]	All zeros
	[3]	DMA burst request selected by EVTSVT.DMACH6SEL
	[2]	DMA single request selected by EVTSVT.DMACH6SEL
	[1]	DMA burst request selected by EVTSVT.DMACH7SEL
	[0]	DMA single request selected by EVTSVT.DMACH7SEL

Further details about routing these set of signals to the pads can be found in Chapter 18.

4.5 EVTULL Registers

Table 4-5 lists the memory-mapped registers for the EVTULL registers. All register offset addresses not listed in Table 4-5 should be considered as reserved locations and the register contents should not be modified.

Table 4-5. EVTULL Registers

Offset	Acronym	Register Name	Section
0h	DESC	Description	Go
4h	DESCEX	Extended Description	Go
64h	DTB	Digital test bus control register	Go
400h	NMISEL	Output Selection for CPU NMI Exception	Go
404h	RTCCPTSEL	Output Selection for RTCCPT	Go
800h	WKUPMASK	WAKEUP Mask	Go

Complex bit access types are encoded to fit into small table cells. Table 4-6 shows the codes that are used for access types in this section.

Table 4-6. EVTULL Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

4.5.1 DESC Register (Offset = 0h) [Reset = 30451010h]

DESC is shown in Table 4-7.

Return to the Summary Table.

Description
This register provides IP module ID, revision information, instance index and standard MMR registers offset.

Table 4-7. DESC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	MODID	R	3045h	Module identifier used to uniquely identify this IP.
15-12	STDIPOFF	R	1h	Standard IP MMR block offset. Standard IP MMRs are the set of from aggregated IRQ registers till DTB. 0: Standard IP MMRs do not exist 0x1-0xF: Standard IP MMRs begin at offset of (64*STDIPOFF from the base IP address)
11-8	INSTIDX	R	0h	IP Instance ID number. If multiple instances of IP exist in the device, this field can identify the instance number (0-15).
7-4	MAJREV	R	1h	Major revision of IP (0-15).
3-0	MINREV	R	0h	Minor revision of IP (0-15).

4.5.2 DESCEX Register (Offset = 4h) [Reset = 00010106h]

DESCEX is shown in Table 4-8.

Return to the Summary Table.

Extended Description
This register provides configuration details of the IP to software drivers and end users.

Table 4-8. DESCEX Register Field Descriptions

Bit	Field	Type	Reset	Description
31-22	IDMA	R	0h	Number of DMA input channels
21-17	NDMA	R	0h	Number of DMA output channels
16	PD	R	1h	Power Domain. 0 : SVT 1 : ULL
15-8	NSUB	R	1h	Number of Subscribers
7-0	NPUB	R	6h	Number of Publishers

4.5.3 DTB Register (Offset = 64h) [Reset = 00000000h]

DTB is shown in Table 4-9.

Return to the Summary Table.

Digital test bus control register
This register can be used to bring out IP internal signals to the pads for observation. 16 signals can be observed per select value.

Table 4-9. DTB Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
0	SEL	R/W	0h	Digital test bus selection mux control Non-zero select values output a 16 bit selected group of signals per value.

4.5.4 NMISEL Register (Offset = 400h) [Reset = 00000000h]

NMISEL is shown in Table 4-10.

Return to the Summary Table.

Output Selection for CPU NMI Exception

Table 4-10. NMISEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive 2h = PMU combined interrupt request for BATMON, interrupt flags can be found here PMUD:EVENT 3h = CKMD combined interrupt request, interrupt flags can be found here CKMD:MIS 4h = AON_RTC event, controlled by the RTC:IMASK setting 5h = DebugSS combined interrupt, interrupt flags can be found here DBGSS:MIS 6h = AON LPCMP interrupt, controlled by SYS0:LPCMPCFG 7h = IOC synchronous combined event, controlled by IOC:EVTCFG

4.5.5 RTCCPTSEL Register (Offset = 404h) [Reset = 00000000h]

RTCCPTSEL is shown in Table 4-11.

Return to the Summary Table.

Output Selection for RTCCPT

Table 4-11. RTCCPTSEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive 2h = PMU combined interrupt request for BATMON, interrupt flags can be found here PMUD:EVENT 3h = CKMD combined interrupt request, interrupt flags can be found here CKMD:MIS 4h = AON_RTC event, controlled by the RTC:IMASK setting 5h = DebugSS combined interrupt, interrupt flags can be found here DBGSS:MIS 6h = AON LPCMP interrupt, controlled by SYS0:LPCMPCFG 7h = IOC synchronous combined event, controlled by IOC:EVTCFG

4.5.6 WKUPMASK Register (Offset = 800h) [Reset = 00000000h]

WKUPMASK is shown in Table 4-12.

Return to the Summary Table.

WAKEUP Mask

Table 4-12. WKUPMASK Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
7	AON_IOC_COMB	R/W	0h	Wake-up mask for AON_IOC_COMB. 0 - Wakeup Disabled 1 - Wakeup Enabled
6	AON_LPMCMP_IRQ	R/W	0h	Wake-up mask for AON_LPMCMP_IRQ. 0 - Wakeup Disabled 1 - Wakeup Enabled
5	AON_DBG_COMB	R/W	0h	Wake-up mask for AON_DBG_COMB. 0 - Wakeup Disabled 1 - Wakeup Enabled
4	AON_RTC_COMB	R/W	0h	Wake-up mask for AON_RTC_COMB. 0 - Wakeup Disabled 1 - Wakeup Enabled
3	AON_CKM_COMB	R/W	0h	Wake-up mask for AON_CKM_COMB. 0 - Wakeup Disabled 1 - Wakeup Enabled
2	AON_PMU_COMB	R/W	0h	Wake-up mask for AON_PMU_COMB. 0 - Wakeup Disabled 1 - Wakeup Enabled
1-0	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior

4.6 EVTSVT Registers

Table 4-13 lists the memory-mapped registers for the EVTSVT registers. All register offset addresses not listed in Table 4-13 should be considered as reserved locations and the register contents should not be modified.

Table 4-13. EVTSVT Registers

Offset	Acronym	Register Name	Section
0h	DESC	Description	Go
4h	DESCEX	Extended Description	Go
64h	DTB	Digital test bus control	Go
400h	NMISEL	Output Selection for CPU NMI Exception	Go
404h	CPUIRQ0SEL	Output Selection for CPU Interrupt CPUIRQ0	Go
408h	CPUIRQ1SEL	Output Selection for CPU Interrupt CPUIRQ1	Go
40Ch	CPUIRQ2SEL	Output Selection for CPU Interrupt CPUIRQ2	Go
410h	CPUIRQ3SEL	Output Selection for CPU Interrupt CPUIRQ3	Go
414h	CPUIRQ4SEL	Output Selection for CPU Interrupt CPUIRQ4	Go
418h	CPUIRQ5SEL	Output Selection for CPU Interrupt CPUIRQ5	Go
41Ch	CPUIRQ6SEL	Output Selection for CPU Interrupt CPUIRQ6	Go
420h	CPUIRQ7SEL	Output Selection for CPU Interrupt CPUIRQ7	Go
424h	CPUIRQ8SEL	Output Selection for CPU Interrupt CPUIRQ8	Go
428h	CPUIRQ9SEL	Output Selection for CPU Interrupt CPUIRQ9	Go
42Ch	CPUIRQ10SEL	Output Selection for CPU Interrupt CPUIRQ10	Go
430h	CPUIRQ11SEL	Output Selection for CPU Interrupt CPUIRQ11	Go
434h	CPUIRQ12SEL	Output Selection for CPU Interrupt CPUIRQ12	Go
438h	CPUIRQ13SEL	Output Selection for CPU Interrupt CPUIRQ13	Go
43Ch	CPUIRQ14SEL	Output Selection for CPU Interrupt CPUIRQ14	Go
440h	CPUIRQ15SEL	Output Selection for CPU Interrupt CPUIRQ15	Go
444h	CPUIRQ16SEL	Output Selection for CPU Interrupt CPUIRQ16	Go
448h	CPUIRQ17SEL	Output Selection for CPU Interrupt CPUIRQ17	Go
44Ch	CPUIRQ18SEL	Output Selection for CPU Interrupt CPUIRQ18	Go
450h	SYSTIMC0SEL	Output Selection for SYSTIMC0	Go
454h	SYSTIMC1SEL	Output Selection for SYSTIMC1	Go
458h	SYSTIMC2SEL	Output Selection for SYSTIMC2	Go
45Ch	SYSTIMC3SEL	Output Selection for SYSTIMC3	Go
460h	SYSTIMC4SEL	Output Selection for SYSTIMC4	Go
464h	ADCTRGSEL	Output Selection for ADCTRG	Go
468h	LGPTSYNCSEL	Output Selection for LGPTSYNC	Go
46Ch	LGPT0IN0SEL	Output Selection for LGPT0IN0	Go
470h	LGPT0IN1SEL	Output Selection for LGPT0IN1	Go
474h	LGPT0IN2SEL	Output Selection for LGPT0IN2	Go
478h	LGPT0TENSEL	Output Selection for LGPT0TEN	Go
47Ch	LGPT1IN0SEL	Output Selection for LGPT1IN0	Go
480h	LGPT1IN1SEL	Output Selection for LGPT1IN1	Go
484h	LGPT1IN2SEL	Output Selection for LGPT1IN2	Go
488h	LGPT1TENSEL	Output Selection for LGPT1TEN	Go
48Ch	LGPT2IN0SEL	Output Selection for LGPT2IN0	Go
490h	LGPT2IN1SEL	Output Selection for LGPT2IN1	Go
494h	LGPT2IN2SEL	Output Selection for LGPT2IN2	Go

Table 4-13. EVTSVT Registers (continued)

Offset	Acronym	Register Name	Section
498h	LGPT2TENSEL	Output Selection for LGPT2TEN	Go
49Ch	LGPT3IN0SEL	Output Selection for LGPT3IN0	Go
4A0h	LGPT3IN1SEL	Output Selection for LGPT3IN1	Go
4A4h	LGPT3IN2SEL	Output Selection for LGPT3IN2	Go
4A8h	LGPT3TENSEL	Output Selection for LGPT3TEN	Go
4ACh	LRFDIN0SEL	Output Selection for LRFDIN0	Go
4B0h	LRFDIN1SEL	Output Selection for LRFDIN1	Go
4B4h	LRFDIN2SEL	Output Selection for LRFDIN2	Go
C00h	DMACH0SEL	Output Selection for DMA CH0	Go
C04h	DMACH1SEL	Output Selection for DMA CH1	Go
C08h	DMACH2SEL	Output Selection for DMA CH2	Go
C0Ch	DMACH3SEL	Output Selection for DMA CH3	Go
C10h	DMACH4SEL	Output Selection for DMA CH4	Go
C14h	DMACH5SEL	Output Selection for DMA CH5	Go
C18h	DMACH6SEL	Output Selection for DMA CH6	Go
C1Ch	DMACH7SEL	Output Selection for DMA CH7	Go

Complex bit access types are encoded to fit into small table cells. Table 4-14 shows the codes that are used for access types in this section.

Table 4-14. EVTSVT Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

4.6.1 DESC Register (Offset = 0h) [Reset = 30451010h]

DESC is shown in Table 4-15.

Return to the Summary Table.

Description
This register provides IP module ID, revision information, instance index and standard MMR registers offset.

Table 4-15. DESC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	MODID	R	3045h	Module identifier used to uniquely identify this IP.
15-12	STDIPOFF	R	1h	Standard IP MMR block offset. Standard IP MMRs are the set of from aggregated IRQ registers till DTB. 0: Standard IP MMRs do not exist 0x1-0xF: Standard IP MMRs begin at offset of (64*STDIPOFF from the base IP address)
11-8	INSTIDX	R	0h	IP Instance ID number. If multiple instances of IP exist in the device, this field can identify the instance number (0-15).
7-4	MAJREV	R	1h	Major revision of IP (0-15).
3-0	MINREV	R	0h	Minor revision of IP (0-15).

4.6.2 DESCEX Register (Offset = 4h) [Reset = 02182D31h]

DESCEX is shown in Table 4-16.

Return to the Summary Table.

Extended Description
This register provides configuration details of the IP to software drivers and end users.

Table 4-16. DESCEX Register Field Descriptions

Bit	Field	Type	Reset	Description
31-22	IDMA	R	8h	Number of DMA input channels
21-17	NDMA	R	Ch	Number of DMA output channels
16	PD	R	0h	Power Domain. 0 : SVT 1 : ULL
15-8	NSUB	R	2Dh	Number of Subscribers
7-0	NPUB	R	31h	Number of Publishers

4.6.3 DTB Register (Offset = 64h) [Reset = 00000000h]

DTB is shown in Table 4-17.

Return to the Summary Table.

Digital test bus control
This register can be used to bring out IP internal signals to the pads for observation. 16 signals can be observed per select value.

Table 4-17. DTB Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
1-0	SEL	R/W	0h	Digital test bus selection mux control. Non-zero select values output a 16 bit selected group of signals per value.

4.6.4 NMISEL Register (Offset = 400h) [Reset = 00000000h]

NMISEL is shown in Table 4-18.

Return to the Summary Table.

Output Selection for CPU NMI Exception

Table 4-18. NMISEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior

Table 4-18. NMISEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive 1h = Selects an AON_NMI source, controlled by EVTULL:NMISEL 8h = SYSTIM combined interrupt, interrupt flags are found here SYSTIM:MIS 9h = GPIO combined wake up interrupt, interrupt flags can be found here GPIO:MIS Ah = GPIO generic published event, controlled by GPIO:EVTCFG Bh = NoWrapper Flash interrupt indicating that the flash operation has completed, interrupt flags can be found here FLASH:MIS Ch = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS0 Dh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS1 Eh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS2 Fh = SPI0 combined interrupt request, interrupt flags can be found here SPI0:MIS 10h = ADC combined interrupt request, interrupt flags can be found here ADC:MIS0 11h = ADC general published event, interrupt flags can be found here ADC:MIS1 12h = LGPT0 combined interrupt, interrupt flags are found here LGPT0:MIS 13h = LGPT1 combined interrupt, interrupt flags are found here LGPT1:MIS 14h = DMA combined done interrupt, corresponding flags can be found here DMA:REQDONE 15h = DMA bus error, corresponds to DMA:ERROR.STATUS 16h = AES accelerator combined interrupt request, interrupt flags can be found here AES:MIS 17h = UART0 combined interrupt, interrupt flags are found here UART0:MIS 18h = Interrupt event from I2C0, interrupt flags can be found here I2C0:MIS 19h = SYSTIM heartbeat, can be set by SYSTIM:TIMEBIT 1Ah = SYSTIM interrupt driven by synchronizing LFTICK signal to SVT clock 1Bh = SYSTIM Channel 0 event, event flag is SYSTIM:MIS.EVT0 1Ch = SYSTIM Channel 1 event, event flag is SYSTIM:MIS.EVT1 1Dh = SYSTIM Channel 2 event, event flag is SYSTIM:MIS.EVT2 1Eh = SYSTIM Channel 3 event, event flag is SYSTIM:MIS.EVT3 1Fh = SYSTIM Channel 4 event, event flag is SYSTIM:MIS.EVT4 20h = LGPT0 compare/capture output event 0, controlled by LGPT0:C0CFG setting 21h = LGPT0 compare/capture output event 1, controlled by LGPT0:C1CFG setting 22h = LGPT0 compare/capture output event 2, controlled by LGPT0:C2CFG setting 23h = LGPT0 DMA request event, controlled by LGPT0:DMA setting 24h = LGPT0 ADC trigger event, controlled by LGPT0:ADCTRG setting 25h = LGPT1 compare/capture output event 0, controlled by LGPT1:C0CFG setting 26h = LGPT1 compare/capture output event 1, controlled by LGPT1:C1CFG setting 27h = LGPT1 compare/capture output event 2, controlled by LGPT1:C2CFG setting 28h = LGPT1 DMA request event, controlled by LGPT1:DMA setting 29h = LGPT1 ADC trigger event, controlled by LGPT1:ADCTRG setting 2Ah = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC0

Table 4-18. NMISEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
				2Bh = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC1 2Ch = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC2 2Eh = LGPT2 compare/capture output event 0, controlled by LGPT2:C0CFG setting 2Fh = LGPT2 compare/capture output event 1, controlled by LGPT2:C1CFG setting 30h = LGPT0 compare/capture output event 2, controlled by LGPT2:C2CFG setting 31h = LGPT2 combined interrupt, interrupt flags are found here LGPT2:MIS 32h = LGPT2 DMA request event, controlled by LGPT2:DMA setting 33h = LGPT2 ADC trigger event, controlled by LGPT2:ADCTRG setting 34h = LGPT3 compare/capture output event 0, controlled by LGPT3:C0CFG setting 35h = LGPT3 compare/capture output event 1, controlled by LGPT3:C1CFG setting 36h = LGPT3 compare/capture output event 2, controlled by LGPT3:C2CFG setting 37h = LGPT3 combined interrupt, interrupt flags are found here LGPT3:MIS 38h = LGPT3 DMA request event, controlled by LGPT3:DMA setting 39h = LGPT3 ADC trigger event, controlled by LGPT3:ADCTRG setting

4.6.5 CPUIRQ0SEL Register (Offset = 404h) [Reset = 00000000h]

CPUIRQ0SEL is shown in Table 4-19.

Return to the Summary Table.

Output Selection for CPU Interrupt CPUIRQ0

Table 4-19. CPUIRQ0SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior

Table 4-19. CPUIRQ0SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive 2h = PMU combined interrupt request for BATMON, interrupt flags can be found here PMUD:EVENT 3h = CKMD combined interrupt request, interrupt flags can be found here CKMD:MIS 4h = AON_RTC event, controlled by the RTC:IMASK setting 5h = DebugSS combined interrupt, interrupt flags can be found here DBGSS:MIS 6h = AON LPCMP interrupt, controlled by SYS0:LPCMPCFG 7h = IOC synchronous combined event, controlled by IOC:EVTCFG 8h = SYSTIM combined interrupt, interrupt flags are found here SYSTIM:MIS 9h = GPIO combined wake up interrupt, interrupt flags can be found here GPIO:MIS Ah = GPIO generic published event, controlled by GPIO:EVTCFG Bh = NoWrapper Flash interrupt indicating that the flash operation has completed, interrupt flags can be found here FLASH:MIS Ch = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS0 Dh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS1 Eh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS2 Fh = SPI0 combined interrupt request, interrupt flags can be found here SPI0:MIS 10h = ADC combined interrupt request, interrupt flags can be found here ADC:MIS0 11h = ADC general published event, interrupt flags can be found here ADC:MIS1 12h = LGPT0 combined interrupt, interrupt flags are found here LGPT0:MIS 13h = LGPT1 combined interrupt, interrupt flags are found here LGPT1:MIS 14h = DMA combined done interrupt, corresponding flags can be found here DMA:REQDONE 15h = DMA bus error, corresponds to DMA:ERROR.STATUS 16h = AES accelerator combined interrupt request, interrupt flags can be found here AES:MIS 17h = UART0 combined interrupt, interrupt flags are found here UART0:MIS 18h = Interrupt event from I2C0, interrupt flags can be found here I2C0:MIS 19h = SYSTIM heartbeat, can be set by SYSTIM:TIMEBIT 1Ah = SYSTIM interrupt driven by synchronizing LFTICK signal to SVT clock 1Bh = SYSTIM Channel 0 event, event flag is SYSTIM:MIS.EVT0 1Ch = SYSTIM Channel 1 event, event flag is SYSTIM:MIS.EVT1 1Dh = SYSTIM Channel 2 event, event flag is SYSTIM:MIS.EVT2 1Eh = SYSTIM Channel 3 event, event flag is SYSTIM:MIS.EVT3 1Fh = SYSTIM Channel 4 event, event flag is SYSTIM:MIS.EVT4 20h = LGPT0 compare/capture output event 0, controlled by LGPT0:C0CFG setting 21h = LGPT0 compare/capture output event 1, controlled by LGPT0:C1CFG setting 22h = LGPT0 compare/capture output event 2, controlled by LGPT0:C2CFG setting 23h = LGPT0 DMA request event, controlled by LGPT0:DMA setting 24h = LGPT0 ADC trigger event, controlled by LGPT0:ADCTRG setting 25h = LGPT1 compare/capture output event 0, controlled by LGPT1:C0CFG setting

Table 4-19. CPUIRQ0SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
				26h = LGPT1 compare/capture output event 1, controlled by LGPT1:C1CFG setting 27h = LGPT1 compare/capture output event 2, controlled by LGPT1:C2CFG setting 28h = LGPT1 DMA request event, controlled by LGPT1:DMA setting 29h = LGPT1 ADC trigger event, controlled by LGPT1:ADCTRG setting 2Ah = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC0 2Bh = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC1 2Ch = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC2 2Eh = LGPT2 compare/capture output event 0, controlled by LGPT2:C0CFG setting 2Fh = LGPT2 compare/capture output event 1, controlled by LGPT2:C1CFG setting 30h = LGPT0 compare/capture output event 2, controlled by LGPT2:C2CFG setting 31h = LGPT2 combined interrupt, interrupt flags are found here LGPT2:MIS 32h = LGPT2 DMA request event, controlled by LGPT2:DMA setting 33h = LGPT2 ADC trigger event, controlled by LGPT2:ADCTRG setting 34h = LGPT3 compare/capture output event 0, controlled by LGPT3:C0CFG setting 35h = LGPT3 compare/capture output event 1, controlled by LGPT3:C1CFG setting 36h = LGPT3 compare/capture output event 2, controlled by LGPT3:C2CFG setting 37h = LGPT3 combined interrupt, interrupt flags are found here LGPT3:MIS 38h = LGPT3 DMA request event, controlled by LGPT3:DMA setting 39h = LGPT3 ADC trigger event, controlled by LGPT3:ADCTRG setting

4.6.6 CPUIRQ1SEL Register (Offset = 408h) [Reset = 00000000h]

CPUIRQ1SEL is shown in Table 4-20.

Return to the Summary Table.

Output Selection for CPU Interrupt CPUIRQ1

Table 4-20. CPUIRQ1SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior

Table 4-20. CPUIRQ1SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive 2h = PMU combined interrupt request for BATMON, interrupt flags can be found here PMUD:EVENT 3h = CKMD combined interrupt request, interrupt flags can be found here CKMD:MIS 4h = AON_RTC event, controlled by the RTC:IMASK setting 5h = DebugSS combined interrupt, interrupt flags can be found here DBGSS:MIS 6h = AON LPCMP interrupt, controlled by SYS0:LPCMPCFG 7h = IOC synchronous combined event, controlled by IOC:EVTCFG 8h = SYSTIM combined interrupt, interrupt flags are found here SYSTIM:MIS 9h = GPIO combined wake up interrupt, interrupt flags can be found here GPIO:MIS Ah = GPIO generic published event, controlled by GPIO:EVTCFG Bh = NoWrapper Flash interrupt indicating that the flash operation has completed, interrupt flags can be found here FLASH:MIS Ch = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS0 Dh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS1 Eh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS2 Fh = SPI0 combined interrupt request, interrupt flags can be found here SPI0:MIS 10h = ADC combined interrupt request, interrupt flags can be found here ADC:MIS0 11h = ADC general published event, interrupt flags can be found here ADC:MIS1 12h = LGPT0 combined interrupt, interrupt flags are found here LGPT0:MIS 13h = LGPT1 combined interrupt, interrupt flags are found here LGPT1:MIS 14h = DMA combined done interrupt, corresponding flags can be found here DMA:REQDONE 15h = DMA bus error, corresponds to DMA:ERROR.STATUS 16h = AES accelerator combined interrupt request, interrupt flags can be found here AES:MIS 17h = UART0 combined interrupt, interrupt flags are found here UART0:MIS 18h = Interrupt event from I2C0, interrupt flags can be found here I2C0:MIS 19h = SYSTIM heartbeat, can be set by SYSTIM:TIMEBIT 1Ah = SYSTIM interrupt driven by synchronizing LFTICK signal to SVT clock 1Bh = SYSTIM Channel 0 event, event flag is SYSTIM:MIS.EVT0 1Ch = SYSTIM Channel 1 event, event flag is SYSTIM:MIS.EVT1 1Dh = SYSTIM Channel 2 event, event flag is SYSTIM:MIS.EVT2 1Eh = SYSTIM Channel 3 event, event flag is SYSTIM:MIS.EVT3 1Fh = SYSTIM Channel 4 event, event flag is SYSTIM:MIS.EVT4 20h = LGPT0 compare/capture output event 0, controlled by LGPT0:C0CFG setting 21h = LGPT0 compare/capture output event 1, controlled by LGPT0:C1CFG setting 22h = LGPT0 compare/capture output event 2, controlled by LGPT0:C2CFG setting 23h = LGPT0 DMA request event, controlled by LGPT0:DMA setting 24h = LGPT0 ADC trigger event, controlled by LGPT0:ADCTRG setting 25h = LGPT1 compare/capture output event 0, controlled by LGPT1:C0CFG setting

Table 4-20. CPUIRQ1SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
				26h = LGPT1 compare/capture output event 1, controlled by LGPT1:C1CFG setting 27h = LGPT1 compare/capture output event 2, controlled by LGPT1:C2CFG setting 28h = LGPT1 DMA request event, controlled by LGPT1:DMA setting 29h = LGPT1 ADC trigger event, controlled by LGPT1:ADCTRG setting 2Ah = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC0 2Bh = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC1 2Ch = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC2 2Eh = LGPT2 compare/capture output event 0, controlled by LGPT2:C0CFG setting 2Fh = LGPT2 compare/capture output event 1, controlled by LGPT2:C1CFG setting 30h = LGPT0 compare/capture output event 2, controlled by LGPT2:C2CFG setting 31h = LGPT2 combined interrupt, interrupt flags are found here LGPT2:MIS 32h = LGPT2 DMA request event, controlled by LGPT2:DMA setting 33h = LGPT2 ADC trigger event, controlled by LGPT2:ADCTRG setting 34h = LGPT3 compare/capture output event 0, controlled by LGPT3:C0CFG setting 35h = LGPT3 compare/capture output event 1, controlled by LGPT3:C1CFG setting 36h = LGPT3 compare/capture output event 2, controlled by LGPT3:C2CFG setting 37h = LGPT3 combined interrupt, interrupt flags are found here LGPT3:MIS 38h = LGPT3 DMA request event, controlled by LGPT3:DMA setting 39h = LGPT3 ADC trigger event, controlled by LGPT3:ADCTRG setting

4.6.7 CPUIRQ2SEL Register (Offset = 40Ch) [Reset = 00000000h]

CPUIRQ2SEL is shown in Table 4-21.

Return to the Summary Table.

Output Selection for CPU Interrupt CPUIRQ2

Table 4-21. CPUIRQ2SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive 2h = PMU combined interrupt request for BATMON, interrupt flags can be found here PMUD:EVENT 3h = CKMD combined interrupt request, interrupt flags can be found here CKMD:MIS 4h = AON_RTC event, controlled by the RTC:IMASK setting 5h = DebugSS combined interrupt, interrupt flags can be found here DBGSS:MIS 6h = AON LPCMP interrupt, controlled by SYS0:LPCMPCFG 7h = IOC synchronous combined event, controlled by IOC:EVTCFG 8h = SYSTIM combined interrupt, interrupt flags are found here SYSTIM:MIS 9h = GPIO combined wake up interrupt, interrupt flags can be found here GPIO:MIS Bh = NoWrapper Flash interrupt indicating that the flash operation has completed, interrupt flags can be found here FLASH:MIS Ch = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS0 Dh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS1 Eh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS2 Fh = SPI0 combined interrupt request, interrupt flags can be found here SPI0:MIS 10h = ADC combined interrupt request, interrupt flags can be found here ADC:MIS0 12h = LGPT0 combined interrupt, interrupt flags are found here LGPT0:MIS 13h = LGPT1 combined interrupt, interrupt flags are found here LGPT1:MIS 14h = DMA combined done interrupt, corresponding flags can be found here DMA:REQDONE 15h = DMA bus error, corresponds to DMA:ERROR.STATUS 16h = AES accelerator combined interrupt request, interrupt flags can be found here AES:MIS 17h = UART0 combined interrupt, interrupt flags are found here UART0:MIS 18h = Interrupt event from I2C0, interrupt flags can be found here I2C0:MIS 31h = LGPT2 combined interrupt, interrupt flags are found here LGPT2:MIS 37h = LGPT3 combined interrupt, interrupt flags are found here LGPT3:MIS

4.6.8 CPUIRQ3SEL Register (Offset = 410h) [Reset = 00000000h]

CPUIRQ3SEL is shown in Table 4-22.

Return to the Summary Table.

Output Selection for CPU Interrupt CPUIRQ3

Table 4-22. CPUIRQ3SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive 2h = PMU combined interrupt request for BATMON, interrupt flags can be found here PMUD:EVENT 3h = CKMD combined interrupt request, interrupt flags can be found here CKMD:MIS 4h = AON_RTC event, controlled by the RTC:IMASK setting 5h = DebugSS combined interrupt, interrupt flags can be found here DBGSS:MIS 6h = AON LPCMP interrupt, controlled by SYS0:LPCMPCFG 7h = IOC synchronous combined event, controlled by IOC:EVTCFG 8h = SYSTIM combined interrupt, interrupt flags are found here SYSTIM:MIS 9h = GPIO combined wake up interrupt, interrupt flags can be found here GPIO:MIS Bh = NoWrapper Flash interrupt indicating that the flash operation has completed, interrupt flags can be found here FLASH:MIS Ch = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS0 Dh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS1 Eh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS2 Fh = SPI0 combined interrupt request, interrupt flags can be found here SPI0:MIS 10h = ADC combined interrupt request, interrupt flags can be found here ADC:MIS0 12h = LGPT0 combined interrupt, interrupt flags are found here LGPT0:MIS 13h = LGPT1 combined interrupt, interrupt flags are found here LGPT1:MIS 14h = DMA combined done interrupt, corresponding flags can be found here DMA:REQDONE 15h = DMA bus error, corresponds to DMA:ERROR.STATUS 16h = AES accelerator combined interrupt request, interrupt flags can be found here AES:MIS 17h = UART0 combined interrupt, interrupt flags are found here UART0:MIS 18h = Interrupt event from I2C0, interrupt flags can be found here I2C0:MIS 31h = LGPT2 combined interrupt, interrupt flags are found here LGPT2:MIS 37h = LGPT3 combined interrupt, interrupt flags are found here LGPT3:MIS

4.6.9 CPUIRQ4SEL Register (Offset = 414h) [Reset = 00000000h]

CPUIRQ4SEL is shown in Table 4-23.

Return to the Summary Table.

Output Selection for CPU Interrupt CPUIRQ4

Table 4-23. CPUIRQ4SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive 2h = PMU combined interrupt request for BATMON, interrupt flags can be found here PMUD:EVENT 3h = CKMD combined interrupt request, interrupt flags can be found here CKMD:MIS 4h = AON_RTC event, controlled by the RTC:IMASK setting 5h = DebugSS combined interrupt, interrupt flags can be found here DBGSS:MIS 6h = AON LPCMP interrupt, controlled by SYS0:LPCMPCFG 7h = IOC synchronous combined event, controlled by IOC:EVTCFG 8h = SYSTIM combined interrupt, interrupt flags are found here SYSTIM:MIS 9h = GPIO combined wake up interrupt, interrupt flags can be found here GPIO:MIS Bh = NoWrapper Flash interrupt indicating that the flash operation has completed, interrupt flags can be found here FLASH:MIS Ch = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS0 Dh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS1 Eh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS2 Fh = SPI0 combined interrupt request, interrupt flags can be found here SPI0:MIS 10h = ADC combined interrupt request, interrupt flags can be found here ADC:MIS0 12h = LGPT0 combined interrupt, interrupt flags are found here LGPT0:MIS 13h = LGPT1 combined interrupt, interrupt flags are found here LGPT1:MIS 14h = DMA combined done interrupt, corresponding flags can be found here DMA:REQDONE 15h = DMA bus error, corresponds to DMA:ERROR.STATUS 16h = AES accelerator combined interrupt request, interrupt flags can be found here AES:MIS 17h = UART0 combined interrupt, interrupt flags are found here UART0:MIS 18h = Interrupt event from I2C0, interrupt flags can be found here I2C0:MIS 31h = LGPT2 combined interrupt, interrupt flags are found here LGPT2:MIS 37h = LGPT3 combined interrupt, interrupt flags are found here LGPT3:MIS

4.6.10 CPUIRQ5SEL Register (Offset = 418h) [Reset = 00000009h]

CPUIRQ5SEL is shown in Table 4-24.

Return to the Summary Table.

Output Selection for CPU Interrupt CPUIRQ5

Table 4-24. CPUIRQ5SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
5-0	PUBID	R	9h	Read only selection value 9h = GPIO combined wake up interrupt, interrupt flags can be found here GPIO:MIS

4.6.11 CPUIRQ6SEL Register (Offset = 41Ch) [Reset = 0000000Ch]

CPUIRQ6SEL is shown in Table 4-25.

Return to the Summary Table.

Output Selection for CPU Interrupt CPUIRQ6

Table 4-25. CPUIRQ6SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
5-0	PUBID	R	Ch	Read only selection value Ch = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS0

4.6.12 CPUIRQ7SEL Register (Offset = 420h) [Reset = 0000000Dh]

CPUIRQ7SEL is shown in Table 4-26.

Return to the Summary Table.

Output Selection for CPU Interrupt CPUIRQ7

Table 4-26. CPUIRQ7SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
5-0	PUBID	R	Dh	Read only selection value Dh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS1

4.6.13 CPUIRQ8SEL Register (Offset = 424h) [Reset = 00000014h]

CPUIRQ8SEL is shown in Table 4-27.

Return to the Summary Table.

Output Selection for CPU Interrupt CPUIRQ8

Table 4-27. CPUIRQ8SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
5-0	PUBID	R	14h	Read only selection value 14h = DMA combined done interrupt, corresponding flags can be found here DMA:REQDONE

4.6.14 CPUIRQ9SEL Register (Offset = 428h) [Reset = 00000016h]

CPUIRQ9SEL is shown in Table 4-28.

Return to the Summary Table.

Output Selection for CPU Interrupt CPUIRQ9

Table 4-28. CPUIRQ9SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
5-0	PUBID	R	16h	Read only selection value 16h = AES accelerator combined interrupt request, interrupt flags can be found here AES:MIS

4.6.15 CPUIRQ10SEL Register (Offset = 42Ch) [Reset = 0000000Fh]

CPUIRQ10SEL is shown in Table 4-29.

Return to the Summary Table.

Output Selection for CPU Interrupt CPUIRQ10

Table 4-29. CPUIRQ10SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
5-0	PUBID	R	Fh	Read only selection value Fh = SPI0 combined interrupt request, interrupt flags can be found here SPI0:MIS

4.6.16 CPUIRQ11SEL Register (Offset = 430h) [Reset = 00000017h]

CPUIRQ11SEL is shown in Table 4-30.

Return to the Summary Table.

Output Selection for CPU Interrupt CPUIRQ11

Table 4-30. CPUIRQ11SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
5-0	PUBID	R	17h	Read only selection value 17h = UART0 combined interrupt, interrupt flags are found here UART0:MIS

4.6.17 CPUIRQ12SEL Register (Offset = 434h) [Reset = 00000018h]

CPUIRQ12SEL is shown in Table 4-31.

Return to the Summary Table.

Output Selection for CPU Interrupt CPUIRQ12

Table 4-31. CPUIRQ12SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
5-0	PUBID	R	18h	Read only selection value 18h = Interrupt event from I2C0, interrupt flags can be found here I2C0:MIS

4.6.18 CPUIRQ13SEL Register (Offset = 438h) [Reset = 00000012h]

CPUIRQ13SEL is shown in Table 4-32.

Return to the Summary Table.

Output Selection for CPU Interrupt CPUIRQ13

Table 4-32. CPUIRQ13SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
5-0	PUBID	R	12h	Read only selection value 12h = LGPT0 combined interrupt, interrupt flags are found here LGPT0:MIS

4.6.19 CPUIRQ14SEL Register (Offset = 43Ch) [Reset = 00000013h]

CPUIRQ14SEL is shown in Table 4-33.

Return to the Summary Table.

Output Selection for CPU Interrupt CPUIRQ14

Table 4-33. CPUIRQ14SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
5-0	PUBID	R	13h	Read only selection value 13h = LGPT1 combined interrupt, interrupt flags are found here LGPT1:MIS

4.6.20 CPUIRQ15SEL Register (Offset = 440h) [Reset = 00000010h]

CPUIRQ15SEL is shown in Table 4-34.

Return to the Summary Table.

Output Selection for CPU Interrupt CPUIRQ15

Table 4-34. CPUIRQ15SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
5-0	PUBID	R	10h	Read only selection value 10h = ADC combined interrupt request, interrupt flags can be found here ADC:MIS0

4.6.21 CPUIRQ16SEL Register (Offset = 444h) [Reset = 00000000h]

CPUIRQ16SEL is shown in Table 4-35.

Return to the Summary Table.

Output Selection for CPU Interrupt CPUIRQ16

Table 4-35. CPUIRQ16SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior

Table 4-35. CPUIRQ16SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive 2h = PMU combined interrupt request for BATMON, interrupt flags can be found here PMUD:EVENT 3h = CKMD combined interrupt request, interrupt flags can be found here CKMD:MIS 4h = AON_RTC event, controlled by the RTC:IMASK setting 5h = DebugSS combined interrupt, interrupt flags can be found here DBGSS:MIS 6h = AON LPCMP interrupt, controlled by SYS0:LPCMPCFG 7h = IOC synchronous combined event, controlled by IOC:EVTCFG 8h = SYSTIM combined interrupt, interrupt flags are found here SYSTIM:MIS 9h = GPIO combined wake up interrupt, interrupt flags can be found here GPIO:MIS Ah = GPIO generic published event, controlled by GPIO:EVTCFG Bh = NoWrapper Flash interrupt indicating that the flash operation has completed, interrupt flags can be found here FLASH:MIS Ch = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS0 Dh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS1 Eh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS2 Fh = SPI0 combined interrupt request, interrupt flags can be found here SPI0:MIS 10h = ADC combined interrupt request, interrupt flags can be found here ADC:MIS0 11h = ADC general published event, interrupt flags can be found here ADC:MIS1 12h = LGPT0 combined interrupt, interrupt flags are found here LGPT0:MIS 13h = LGPT1 combined interrupt, interrupt flags are found here LGPT1:MIS 14h = DMA combined done interrupt, corresponding flags can be found here DMA:REQDONE 15h = DMA bus error, corresponds to DMA:ERROR.STATUS 16h = AES accelerator combined interrupt request, interrupt flags can be found here AES:MIS 17h = UART0 combined interrupt, interrupt flags are found here UART0:MIS 18h = Interrupt event from I2C0, interrupt flags can be found here I2C0:MIS 19h = SYSTIM heartbeat, can be set by SYSTIM:TIMEBIT 1Ah = SYSTIM interrupt driven by synchronizing LFTICK signal to SVT clock 1Bh = SYSTIM Channel 0 event, event flag is SYSTIM:MIS.EVT0 1Ch = SYSTIM Channel 1 event, event flag is SYSTIM:MIS.EVT1 1Dh = SYSTIM Channel 2 event, event flag is SYSTIM:MIS.EVT2 1Eh = SYSTIM Channel 3 event, event flag is SYSTIM:MIS.EVT3 1Fh = SYSTIM Channel 4 event, event flag is SYSTIM:MIS.EVT4 20h = LGPT0 compare/capture output event 0, controlled by LGPT0:C0CFG setting 21h = LGPT0 compare/capture output event 1, controlled by LGPT0:C1CFG setting 22h = LGPT0 compare/capture output event 2, controlled by LGPT0:C2CFG setting 23h = LGPT0 DMA request event, controlled by LGPT0:DMA setting 24h = LGPT0 ADC trigger event, controlled by LGPT0:ADCTRG setting 25h = LGPT1 compare/capture output event 0, controlled by LGPT1:C0CFG setting

Table 4-35. CPUIRQ16SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
				26h = LGPT1 compare/capture output event 1, controlled by LGPT1:C1CFG setting 27h = LGPT1 compare/capture output event 2, controlled by LGPT1:C2CFG setting 28h = LGPT1 DMA request event, controlled by LGPT1:DMA setting 29h = LGPT1 ADC trigger event, controlled by LGPT1:ADCTRG setting 2Ah = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC0 2Bh = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC1 2Ch = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC2 2Eh = LGPT2 compare/capture output event 0, controlled by LGPT2:C0CFG setting 2Fh = LGPT2 compare/capture output event 1, controlled by LGPT2:C1CFG setting 30h = LGPT0 compare/capture output event 2, controlled by LGPT2:C2CFG setting 31h = LGPT2 combined interrupt, interrupt flags are found here LGPT2:MIS 32h = LGPT2 DMA request event, controlled by LGPT2:DMA setting 33h = LGPT2 ADC trigger event, controlled by LGPT2:ADCTRG setting 34h = LGPT3 compare/capture output event 0, controlled by LGPT3:C0CFG setting 35h = LGPT3 compare/capture output event 1, controlled by LGPT3:C1CFG setting 36h = LGPT3 compare/capture output event 2, controlled by LGPT3:C2CFG setting 37h = LGPT3 combined interrupt, interrupt flags are found here LGPT3:MIS 38h = LGPT3 DMA request event, controlled by LGPT3:DMA setting 39h = LGPT3 ADC trigger event, controlled by LGPT3:ADCTRG setting

4.6.22 CPUIRQ17SEL Register (Offset = 448h) [Reset = 00000031h]

CPUIRQ17SEL is shown in Table 4-36.

Return to the Summary Table.

Output Selection for CPU Interrupt CPUIRQ17

Table 4-36. CPUIRQ17SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
5-0	PUBID	R	31h	Read only selection value 31h = LGPT2 combined interrupt, interrupt flags are found here LGPT2:MIS

4.6.23 CPUIRQ18SEL Register (Offset = 44Ch) [Reset = 00000037h]

CPUIRQ18SEL is shown in Table 4-37.

Return to the Summary Table.

Output Selection for CPU Interrupt CPUIRQ18

Table 4-37. CPUIRQ18SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
5-0	PUBID	R	37h	Read only selection value 37h = LGPT3 combined interrupt, interrupt flags are found here LGPT3:MIS

4.6.24 SYSTIMC0SEL Register (Offset = 450h) [Reset = 00000004h]

SYSTIMC0SEL is shown in Table 4-38.

Return to the Summary Table.

Output Selection for SYSTIMC0

Table 4-38. SYSTIMC0SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
5-0	PUBID	R	4h	Read only selection value 4h = AON_RTC event, controlled by the RTC:IMASK setting

4.6.25 SYSTIMC1SEL Register (Offset = 454h) [Reset = 00000000h]

SYSTIMC1SEL is shown in Table 4-39.

Return to the Summary Table.

Output Selection for SYSTIMC1

Table 4-39. SYSTIMC1SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior

Table 4-39. SYSTIMC1SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive 2h = PMU combined interrupt request for BATMON, interrupt flags can be found here PMUD:EVENT 3h = CKMD combined interrupt request, interrupt flags can be found here CKMD:MIS 4h = AON_RTC event, controlled by the RTC:IMASK setting 5h = DebugSS combined interrupt, interrupt flags can be found here DBGSS:MIS 6h = AON LPCMP interrupt, controlled by SYS0:LPCMPCFG 7h = IOC synchronous combined event, controlled by IOC:EVTCFG 8h = SYSTIM combined interrupt, interrupt flags are found here SYSTIM:MIS 9h = GPIO combined wake up interrupt, interrupt flags can be found here GPIO:MIS Ah = GPIO generic published event, controlled by GPIO:EVTCFG Bh = NoWrapper Flash interrupt indicating that the flash operation has completed, interrupt flags can be found here FLASH:MIS Ch = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS0 Dh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS1 Eh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS2 Fh = SPI0 combined interrupt request, interrupt flags can be found here SPI0:MIS 10h = ADC combined interrupt request, interrupt flags can be found here ADC:MIS0 11h = ADC general published event, interrupt flags can be found here ADC:MIS1 12h = LGPT0 combined interrupt, interrupt flags are found here LGPT0:MIS 13h = LGPT1 combined interrupt, interrupt flags are found here LGPT1:MIS 14h = DMA combined done interrupt, corresponding flags can be found here DMA:REQDONE 15h = DMA bus error, corresponds to DMA:ERROR.STATUS 16h = AES accelerator combined interrupt request, interrupt flags can be found here AES:MIS 17h = UART0 combined interrupt, interrupt flags are found here UART0:MIS 18h = Interrupt event from I2C0, interrupt flags can be found here I2C0:MIS 19h = SYSTIM heartbeat, can be set by SYSTIM:TIMEBIT 1Ah = SYSTIM interrupt driven by synchronizing LFTICK signal to SVT clock 1Bh = SYSTIM Channel 0 event, event flag is SYSTIM:MIS.EVT0 1Ch = SYSTIM Channel 1 event, event flag is SYSTIM:MIS.EVT1 1Dh = SYSTIM Channel 2 event, event flag is SYSTIM:MIS.EVT2 1Eh = SYSTIM Channel 3 event, event flag is SYSTIM:MIS.EVT3 1Fh = SYSTIM Channel 4 event, event flag is SYSTIM:MIS.EVT4 20h = LGPT0 compare/capture output event 0, controlled by LGPT0:C0CFG setting 21h = LGPT0 compare/capture output event 1, controlled by LGPT0:C1CFG setting 22h = LGPT0 compare/capture output event 2, controlled by LGPT0:C2CFG setting 23h = LGPT0 DMA request event, controlled by LGPT0:DMA setting 24h = LGPT0 ADC trigger event, controlled by LGPT0:ADCTRG setting 25h = LGPT1 compare/capture output event 0, controlled by LGPT1:C0CFG setting

Table 4-39. SYSTIMC1SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
				26h = LGPT1 compare/capture output event 1, controlled by LGPT1:C1CFG setting 27h = LGPT1 compare/capture output event 2, controlled by LGPT1:C2CFG setting 28h = LGPT1 DMA request event, controlled by LGPT1:DMA setting 29h = LGPT1 ADC trigger event, controlled by LGPT1:ADCTRG setting 2Ah = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC0 2Bh = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC1 2Ch = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC2 2Eh = LGPT2 compare/capture output event 0, controlled by LGPT2:C0CFG setting 2Fh = LGPT2 compare/capture output event 1, controlled by LGPT2:C1CFG setting 30h = LGPT0 compare/capture output event 2, controlled by LGPT2:C2CFG setting 31h = LGPT2 combined interrupt, interrupt flags are found here LGPT2:MIS 32h = LGPT2 DMA request event, controlled by LGPT2:DMA setting 33h = LGPT2 ADC trigger event, controlled by LGPT2:ADCTRG setting 34h = LGPT3 compare/capture output event 0, controlled by LGPT3:C0CFG setting 35h = LGPT3 compare/capture output event 1, controlled by LGPT3:C1CFG setting 36h = LGPT3 compare/capture output event 2, controlled by LGPT3:C2CFG setting 37h = LGPT3 combined interrupt, interrupt flags are found here LGPT3:MIS 38h = LGPT3 DMA request event, controlled by LGPT3:DMA setting 39h = LGPT3 ADC trigger event, controlled by LGPT3:ADCTRG setting

4.6.26 SYSTIMC2SEL Register (Offset = 458h) [Reset = 0000002Ah]

SYSTIMC2SEL is shown in Table 4-40.

Return to the Summary Table.

Output Selection for SYSTIMC2

Table 4-40. SYSTIMC2SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
5-0	PUBID	R	2Ah	Read only selection value 2Ah = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC0

4.6.27 SYSTIMC3SEL Register (Offset = 45Ch) [Reset = 0000002Bh]

SYSTIMC3SEL is shown in Table 4-41.

Return to the Summary Table.

Output Selection for SYSTIMC3

Table 4-41. SYSTIMC3SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
5-0	PUBID	R	2Bh	Read only selection value 2Bh = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC1

4.6.28 SYSTIMC4SEL Register (Offset = 460h) [Reset = 0000002Ch]

SYSTIMC4SEL is shown in Table 4-42.

Return to the Summary Table.

Output Selection for SYSTIMC4

Table 4-42. SYSTIMC4SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
5-0	PUBID	R	2Ch	Read only selection value 2Ch = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC2

4.6.29 ADCTRGSEL Register (Offset = 464h) [Reset = 00000000h]

ADCTRGSEL is shown in Table 4-43.

Return to the Summary Table.

Output Selection for ADCTRG

Table 4-43. ADCTRGSEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior

Table 4-43. ADCTRGSEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive 2h = PMU combined interrupt request for BATMON, interrupt flags can be found here PMUD:EVENT 3h = CKMD combined interrupt request, interrupt flags can be found here CKMD:MIS 4h = AON_RTC event, controlled by the RTC:IMASK setting 5h = DebugSS combined interrupt, interrupt flags can be found here DBGSS:MIS 6h = AON LPCMP interrupt, controlled by SYS0:LPCMPCFG 7h = IOC synchronous combined event, controlled by IOC:EVTCFG 8h = SYSTIM combined interrupt, interrupt flags are found here SYSTIM:MIS 9h = GPIO combined wake up interrupt, interrupt flags can be found here GPIO:MIS Ah = GPIO generic published event, controlled by GPIO:EVTCFG Bh = NoWrapper Flash interrupt indicating that the flash operation has completed, interrupt flags can be found here FLASH:MIS Ch = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS0 Dh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS1 Eh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS2 Fh = SPI0 combined interrupt request, interrupt flags can be found here SPI0:MIS 10h = ADC combined interrupt request, interrupt flags can be found here ADC:MIS0 11h = ADC general published event, interrupt flags can be found here ADC:MIS1 12h = LGPT0 combined interrupt, interrupt flags are found here LGPT0:MIS 13h = LGPT1 combined interrupt, interrupt flags are found here LGPT1:MIS 14h = DMA combined done interrupt, corresponding flags can be found here DMA:REQDONE 15h = DMA bus error, corresponds to DMA:ERROR.STATUS 16h = AES accelerator combined interrupt request, interrupt flags can be found here AES:MIS 17h = UART0 combined interrupt, interrupt flags are found here UART0:MIS 18h = Interrupt event from I2C0, interrupt flags can be found here I2C0:MIS 19h = SYSTIM heartbeat, can be set by SYSTIM:TIMEBIT 1Ah = SYSTIM interrupt driven by synchronizing LFTICK signal to SVT clock 1Bh = SYSTIM Channel 0 event, event flag is SYSTIM:MIS.EVT0 1Ch = SYSTIM Channel 1 event, event flag is SYSTIM:MIS.EVT1 1Dh = SYSTIM Channel 2 event, event flag is SYSTIM:MIS.EVT2 1Eh = SYSTIM Channel 3 event, event flag is SYSTIM:MIS.EVT3 1Fh = SYSTIM Channel 4 event, event flag is SYSTIM:MIS.EVT4 20h = LGPT0 compare/capture output event 0, controlled by LGPT0:C0CFG setting 21h = LGPT0 compare/capture output event 1, controlled by LGPT0:C1CFG setting 22h = LGPT0 compare/capture output event 2, controlled by LGPT0:C2CFG setting 23h = LGPT0 DMA request event, controlled by LGPT0:DMA setting 24h = LGPT0 ADC trigger event, controlled by LGPT0:ADCTRG setting 25h = LGPT1 compare/capture output event 0, controlled by LGPT1:C0CFG setting

Table 4-43. ADCTRGSEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
				26h = LGPT1 compare/capture output event 1, controlled by LGPT1:C1CFG setting 27h = LGPT1 compare/capture output event 2, controlled by LGPT1:C2CFG setting 28h = LGPT1 DMA request event, controlled by LGPT1:DMA setting 29h = LGPT1 ADC trigger event, controlled by LGPT1:ADCTRG setting 2Ah = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC0 2Bh = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC1 2Ch = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC2 2Eh = LGPT2 compare/capture output event 0, controlled by LGPT2:C0CFG setting 2Fh = LGPT2 compare/capture output event 1, controlled by LGPT2:C1CFG setting 30h = LGPT0 compare/capture output event 2, controlled by LGPT2:C2CFG setting 31h = LGPT2 combined interrupt, interrupt flags are found here LGPT2:MIS 32h = LGPT2 DMA request event, controlled by LGPT2:DMA setting 33h = LGPT2 ADC trigger event, controlled by LGPT2:ADCTRG setting 34h = LGPT3 compare/capture output event 0, controlled by LGPT3:C0CFG setting 35h = LGPT3 compare/capture output event 1, controlled by LGPT3:C1CFG setting 36h = LGPT3 compare/capture output event 2, controlled by LGPT3:C2CFG setting 37h = LGPT3 combined interrupt, interrupt flags are found here LGPT3:MIS 38h = LGPT3 DMA request event, controlled by LGPT3:DMA setting 39h = LGPT3 ADC trigger event, controlled by LGPT3:ADCTRG setting

4.6.30 LGPTSYNCSEL Register (Offset = 468h) [Reset = 00000000h]

LGPTSYNCSEL is shown in Table 4-44.

Return to the Summary Table.

Output Selection for LGPTSYNC

Table 4-44. LGPTSYNCSEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior

Table 4-44. LGPTSYNCSEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive 2h = PMU combined interrupt request for BATMON, interrupt flags can be found here PMUD:EVENT 3h = CKMD combined interrupt request, interrupt flags can be found here CKMD:MIS 4h = AON_RTC event, controlled by the RTC:IMASK setting 5h = DebugSS combined interrupt, interrupt flags can be found here DBGSS:MIS 6h = AON LPCMP interrupt, controlled by SYS0:LPCMPCFG 7h = IOC synchronous combined event, controlled by IOC:EVTCFG 8h = SYSTIM combined interrupt, interrupt flags are found here SYSTIM:MIS 9h = GPIO combined wake up interrupt, interrupt flags can be found here GPIO:MIS Ah = GPIO generic published event, controlled by GPIO:EVTCFG Bh = NoWrapper Flash interrupt indicating that the flash operation has completed, interrupt flags can be found here FLASH:MIS Ch = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS0 Dh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS1 Eh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS2 Fh = SPI0 combined interrupt request, interrupt flags can be found here SPI0:MIS 10h = ADC combined interrupt request, interrupt flags can be found here ADC:MIS0 11h = ADC general published event, interrupt flags can be found here ADC:MIS1 12h = LGPT0 combined interrupt, interrupt flags are found here LGPT0:MIS 13h = LGPT1 combined interrupt, interrupt flags are found here LGPT1:MIS 14h = DMA combined done interrupt, corresponding flags can be found here DMA:REQDONE 15h = DMA bus error, corresponds to DMA:ERROR.STATUS 16h = AES accelerator combined interrupt request, interrupt flags can be found here AES:MIS 17h = UART0 combined interrupt, interrupt flags are found here UART0:MIS 18h = Interrupt event from I2C0, interrupt flags can be found here I2C0:MIS 19h = SYSTIM heartbeat, can be set by SYSTIM:TIMEBIT 1Ah = SYSTIM interrupt driven by synchronizing LFTICK signal to SVT clock 1Bh = SYSTIM Channel 0 event, event flag is SYSTIM:MIS.EVT0 1Ch = SYSTIM Channel 1 event, event flag is SYSTIM:MIS.EVT1 1Dh = SYSTIM Channel 2 event, event flag is SYSTIM:MIS.EVT2 1Eh = SYSTIM Channel 3 event, event flag is SYSTIM:MIS.EVT3 1Fh = SYSTIM Channel 4 event, event flag is SYSTIM:MIS.EVT4 20h = LGPT0 compare/capture output event 0, controlled by LGPT0:C0CFG setting 21h = LGPT0 compare/capture output event 1, controlled by LGPT0:C1CFG setting 22h = LGPT0 compare/capture output event 2, controlled by LGPT0:C2CFG setting 23h = LGPT0 DMA request event, controlled by LGPT0:DMA setting 24h = LGPT0 ADC trigger event, controlled by LGPT0:ADCTRG setting 25h = LGPT1 compare/capture output event 0, controlled by LGPT1:C0CFG setting

Table 4-44. LGPTSYNCSEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
				26h = LGPT1 compare/capture output event 1, controlled by LGPT1:C1CFG setting 27h = LGPT1 compare/capture output event 2, controlled by LGPT1:C2CFG setting 28h = LGPT1 DMA request event, controlled by LGPT1:DMA setting 29h = LGPT1 ADC trigger event, controlled by LGPT1:ADCTRG setting 2Ah = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC0 2Bh = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC1 2Ch = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC2 2Eh = LGPT2 compare/capture output event 0, controlled by LGPT2:C0CFG setting 2Fh = LGPT2 compare/capture output event 1, controlled by LGPT2:C1CFG setting 30h = LGPT0 compare/capture output event 2, controlled by LGPT2:C2CFG setting 31h = LGPT2 combined interrupt, interrupt flags are found here LGPT2:MIS 32h = LGPT2 DMA request event, controlled by LGPT2:DMA setting 33h = LGPT2 ADC trigger event, controlled by LGPT2:ADCTRG setting 34h = LGPT3 compare/capture output event 0, controlled by LGPT3:C0CFG setting 35h = LGPT3 compare/capture output event 1, controlled by LGPT3:C1CFG setting 36h = LGPT3 compare/capture output event 2, controlled by LGPT3:C2CFG setting 37h = LGPT3 combined interrupt, interrupt flags are found here LGPT3:MIS 38h = LGPT3 DMA request event, controlled by LGPT3:DMA setting 39h = LGPT3 ADC trigger event, controlled by LGPT3:ADCTRG setting

4.6.31 LGPT0IN0SEL Register (Offset = 46Ch) [Reset = 00000000h]

LGPT0IN0SEL is shown in Table 4-45.

Return to the Summary Table.

Output Selection for LGPT0IN0

Table 4-45. LGPT0IN0SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior

Table 4-45. LGPT0IN0SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive 2h = PMU combined interrupt request for BATMON, interrupt flags can be found here PMUD:EVENT 3h = CKMD combined interrupt request, interrupt flags can be found here CKMD:MIS 4h = AON_RTC event, controlled by the RTC:IMASK setting 5h = DebugSS combined interrupt, interrupt flags can be found here DBGSS:MIS 6h = AON LPCMP interrupt, controlled by SYS0:LPCMPCFG 7h = IOC synchronous combined event, controlled by IOC:EVTCFG 8h = SYSTIM combined interrupt, interrupt flags are found here SYSTIM:MIS 9h = GPIO combined wake up interrupt, interrupt flags can be found here GPIO:MIS Ah = GPIO generic published event, controlled by GPIO:EVTCFG Bh = NoWrapper Flash interrupt indicating that the flash operation has completed, interrupt flags can be found here FLASH:MIS Ch = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS0 Dh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS1 Eh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS2 Fh = SPI0 combined interrupt request, interrupt flags can be found here SPI0:MIS 10h = ADC combined interrupt request, interrupt flags can be found here ADC:MIS0 11h = ADC general published event, interrupt flags can be found here ADC:MIS1 12h = LGPT0 combined interrupt, interrupt flags are found here LGPT0:MIS 13h = LGPT1 combined interrupt, interrupt flags are found here LGPT1:MIS 14h = DMA combined done interrupt, corresponding flags can be found here DMA:REQDONE 15h = DMA bus error, corresponds to DMA:ERROR.STATUS 16h = AES accelerator combined interrupt request, interrupt flags can be found here AES:MIS 17h = UART0 combined interrupt, interrupt flags are found here UART0:MIS 18h = Interrupt event from I2C0, interrupt flags can be found here I2C0:MIS 19h = SYSTIM heartbeat, can be set by SYSTIM:TIMEBIT 1Ah = SYSTIM interrupt driven by synchronizing LFTICK signal to SVT clock 1Bh = SYSTIM Channel 0 event, event flag is SYSTIM:MIS.EVT0 1Ch = SYSTIM Channel 1 event, event flag is SYSTIM:MIS.EVT1 1Dh = SYSTIM Channel 2 event, event flag is SYSTIM:MIS.EVT2 1Eh = SYSTIM Channel 3 event, event flag is SYSTIM:MIS.EVT3 1Fh = SYSTIM Channel 4 event, event flag is SYSTIM:MIS.EVT4 20h = LGPT0 compare/capture output event 0, controlled by LGPT0:C0CFG setting 21h = LGPT0 compare/capture output event 1, controlled by LGPT0:C1CFG setting 22h = LGPT0 compare/capture output event 2, controlled by LGPT0:C2CFG setting 23h = LGPT0 DMA request event, controlled by LGPT0:DMA setting 24h = LGPT0 ADC trigger event, controlled by LGPT0:ADCTRG setting 25h = LGPT1 compare/capture output event 0, controlled by LGPT1:C0CFG setting

Table 4-45. LGPT0IN0SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
				26h = LGPT1 compare/capture output event 1, controlled by LGPT1:C1CFG setting 27h = LGPT1 compare/capture output event 2, controlled by LGPT1:C2CFG setting 28h = LGPT1 DMA request event, controlled by LGPT1:DMA setting 29h = LGPT1 ADC trigger event, controlled by LGPT1:ADCTRG setting 2Ah = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC0 2Bh = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC1 2Ch = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC2 2Eh = LGPT2 compare/capture output event 0, controlled by LGPT2:C0CFG setting 2Fh = LGPT2 compare/capture output event 1, controlled by LGPT2:C1CFG setting 30h = LGPT0 compare/capture output event 2, controlled by LGPT2:C2CFG setting 31h = LGPT2 combined interrupt, interrupt flags are found here LGPT2:MIS 32h = LGPT2 DMA request event, controlled by LGPT2:DMA setting 33h = LGPT2 ADC trigger event, controlled by LGPT2:ADCTRG setting 34h = LGPT3 compare/capture output event 0, controlled by LGPT3:C0CFG setting 35h = LGPT3 compare/capture output event 1, controlled by LGPT3:C1CFG setting 36h = LGPT3 compare/capture output event 2, controlled by LGPT3:C2CFG setting 37h = LGPT3 combined interrupt, interrupt flags are found here LGPT3:MIS 38h = LGPT3 DMA request event, controlled by LGPT3:DMA setting 39h = LGPT3 ADC trigger event, controlled by LGPT3:ADCTRG setting

4.6.32 LGPT0IN1SEL Register (Offset = 470h) [Reset = 00000000h]

LGPT0IN1SEL is shown in Table 4-46.

Return to the Summary Table.

Output Selection for LGPT0IN1

Table 4-46. LGPT0IN1SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior

Table 4-46. LGPT0IN1SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive Ah = GPIO generic published event, controlled by GPIO:EVTCFG 11h = ADC general published event, interrupt flags can be found here ADC:MIS1 19h = SYSTIM heartbeat, can be set by SYSTIM:TIMEBIT 1Ah = SYSTIM interrupt driven by synchronizing LFTICK signal to SVT clock 1Bh = SYSTIM Channel 0 event, event flag is SYSTIM:MIS.EVT0 1Ch = SYSTIM Channel 1 event, event flag is SYSTIM:MIS.EVT1 1Dh = SYSTIM Channel 2 event, event flag is SYSTIM:MIS.EVT2 1Eh = SYSTIM Channel 3 event, event flag is SYSTIM:MIS.EVT3 1Fh = SYSTIM Channel 4 event, event flag is SYSTIM:MIS.EVT4 20h = LGPT0 compare/capture output event 0, controlled by LGPT0:C0CFG setting 21h = LGPT0 compare/capture output event 1, controlled by LGPT0:C1CFG setting 22h = LGPT0 compare/capture output event 2, controlled by LGPT0:C2CFG setting 23h = LGPT0 DMA request event, controlled by LGPT0:DMA setting 24h = LGPT0 ADC trigger event, controlled by LGPT0:ADCTRG setting 25h = LGPT1 compare/capture output event 0, controlled by LGPT1:C0CFG setting 26h = LGPT1 compare/capture output event 1, controlled by LGPT1:C1CFG setting 27h = LGPT1 compare/capture output event 2, controlled by LGPT1:C2CFG setting 28h = LGPT1 DMA request event, controlled by LGPT1:DMA setting 29h = LGPT1 ADC trigger event, controlled by LGPT1:ADCTRG setting 2Ah = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC0 2Bh = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC1 2Ch = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC2 2Eh = LGPT2 compare/capture output event 0, controlled by LGPT2:C0CFG setting 2Fh = LGPT2 compare/capture output event 1, controlled by LGPT2:C1CFG setting 30h = LGPT0 compare/capture output event 2, controlled by LGPT2:C2CFG setting 32h = LGPT2 DMA request event, controlled by LGPT2:DMA setting 33h = LGPT2 ADC trigger event, controlled by LGPT2:ADCTRG setting 34h = LGPT3 compare/capture output event 0, controlled by LGPT3:C0CFG setting 35h = LGPT3 compare/capture output event 1, controlled by LGPT3:C1CFG setting 36h = LGPT3 compare/capture output event 2, controlled by LGPT3:C2CFG setting 38h = LGPT3 DMA request event, controlled by LGPT3:DMA setting 39h = LGPT3 ADC trigger event, controlled by LGPT3:ADCTRG setting

4.6.33 LGPT0IN2SEL Register (Offset = 474h) [Reset = 00000000h]

LGPT0IN2SEL is shown in Table 4-47.

Return to the Summary Table.

Output Selection for LGPT0IN2

Table 4-47. LGPT0IN2SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior

Table 4-47. LGPT0IN2SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive Ah = GPIO generic published event, controlled by GPIO:EVTCFG 11h = ADC general published event, interrupt flags can be found here ADC:MIS1 19h = SYSTIM heartbeat, can be set by SYSTIM:TIMEBIT 1Ah = SYSTIM interrupt driven by synchronizing LFTICK signal to SVT clock 1Bh = SYSTIM Channel 0 event, event flag is SYSTIM:MIS.EVT0 1Ch = SYSTIM Channel 1 event, event flag is SYSTIM:MIS.EVT1 1Dh = SYSTIM Channel 2 event, event flag is SYSTIM:MIS.EVT2 1Eh = SYSTIM Channel 3 event, event flag is SYSTIM:MIS.EVT3 1Fh = SYSTIM Channel 4 event, event flag is SYSTIM:MIS.EVT4 20h = LGPT0 compare/capture output event 0, controlled by LGPT0:C0CFG setting 21h = LGPT0 compare/capture output event 1, controlled by LGPT0:C1CFG setting 22h = LGPT0 compare/capture output event 2, controlled by LGPT0:C2CFG setting 23h = LGPT0 DMA request event, controlled by LGPT0:DMA setting 24h = LGPT0 ADC trigger event, controlled by LGPT0:ADCTRG setting 25h = LGPT1 compare/capture output event 0, controlled by LGPT1:C0CFG setting 26h = LGPT1 compare/capture output event 1, controlled by LGPT1:C1CFG setting 27h = LGPT1 compare/capture output event 2, controlled by LGPT1:C2CFG setting 28h = LGPT1 DMA request event, controlled by LGPT1:DMA setting 29h = LGPT1 ADC trigger event, controlled by LGPT1:ADCTRG setting 2Ah = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC0 2Bh = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC1 2Ch = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC2 2Eh = LGPT2 compare/capture output event 0, controlled by LGPT2:C0CFG setting 2Fh = LGPT2 compare/capture output event 1, controlled by LGPT2:C1CFG setting 30h = LGPT0 compare/capture output event 2, controlled by LGPT2:C2CFG setting 32h = LGPT2 DMA request event, controlled by LGPT2:DMA setting 33h = LGPT2 ADC trigger event, controlled by LGPT2:ADCTRG setting 34h = LGPT3 compare/capture output event 0, controlled by LGPT3:C0CFG setting 35h = LGPT3 compare/capture output event 1, controlled by LGPT3:C1CFG setting 36h = LGPT3 compare/capture output event 2, controlled by LGPT3:C2CFG setting 38h = LGPT3 DMA request event, controlled by LGPT3:DMA setting 39h = LGPT3 ADC trigger event, controlled by LGPT3:ADCTRG setting

4.6.34 LGPT0TENSEL Register (Offset = 478h) [Reset = 00000000h]

LGPT0TENSEL is shown in Table 4-48.

Return to the Summary Table.

Output Selection for LGPT0TEN

Table 4-48. LGPT0TENSEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior

Table 4-48. LGPT0TENSEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive Ah = GPIO generic published event, controlled by GPIO:EVTCFG 11h = ADC general published event, interrupt flags can be found here ADC:MIS1 19h = SYSTIM heartbeat, can be set by SYSTIM:TIMEBIT 1Ah = SYSTIM interrupt driven by synchronizing LFTICK signal to SVT clock 1Bh = SYSTIM Channel 0 event, event flag is SYSTIM:MIS.EVT0 1Ch = SYSTIM Channel 1 event, event flag is SYSTIM:MIS.EVT1 1Dh = SYSTIM Channel 2 event, event flag is SYSTIM:MIS.EVT2 1Eh = SYSTIM Channel 3 event, event flag is SYSTIM:MIS.EVT3 1Fh = SYSTIM Channel 4 event, event flag is SYSTIM:MIS.EVT4 20h = LGPT0 compare/capture output event 0, controlled by LGPT0:C0CFG setting 21h = LGPT0 compare/capture output event 1, controlled by LGPT0:C1CFG setting 22h = LGPT0 compare/capture output event 2, controlled by LGPT0:C2CFG setting 23h = LGPT0 DMA request event, controlled by LGPT0:DMA setting 24h = LGPT0 ADC trigger event, controlled by LGPT0:ADCTRG setting 25h = LGPT1 compare/capture output event 0, controlled by LGPT1:C0CFG setting 26h = LGPT1 compare/capture output event 1, controlled by LGPT1:C1CFG setting 27h = LGPT1 compare/capture output event 2, controlled by LGPT1:C2CFG setting 28h = LGPT1 DMA request event, controlled by LGPT1:DMA setting 29h = LGPT1 ADC trigger event, controlled by LGPT1:ADCTRG setting 2Ah = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC0 2Bh = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC1 2Ch = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC2 2Eh = LGPT2 compare/capture output event 0, controlled by LGPT2:C0CFG setting 2Fh = LGPT2 compare/capture output event 1, controlled by LGPT2:C1CFG setting 30h = LGPT0 compare/capture output event 2, controlled by LGPT2:C2CFG setting 32h = LGPT2 DMA request event, controlled by LGPT2:DMA setting 33h = LGPT2 ADC trigger event, controlled by LGPT2:ADCTRG setting 34h = LGPT3 compare/capture output event 0, controlled by LGPT3:C0CFG setting 35h = LGPT3 compare/capture output event 1, controlled by LGPT3:C1CFG setting 36h = LGPT3 compare/capture output event 2, controlled by LGPT3:C2CFG setting 38h = LGPT3 DMA request event, controlled by LGPT3:DMA setting 39h = LGPT3 ADC trigger event, controlled by LGPT3:ADCTRG setting

4.6.35 LGPT1IN0SEL Register (Offset = 47Ch) [Reset = 00000000h]

LGPT1IN0SEL is shown in Table 4-49.

Return to the Summary Table.

Output Selection for LGPT1IN0

Table 4-49. LGPT1IN0SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior

Table 4-49. LGPT1IN0SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive 2h = PMU combined interrupt request for BATMON, interrupt flags can be found here PMUD:EVENT 3h = CKMD combined interrupt request, interrupt flags can be found here CKMD:MIS 4h = AON_RTC event, controlled by the RTC:IMASK setting 5h = DebugSS combined interrupt, interrupt flags can be found here DBGSS:MIS 6h = AON LPCMP interrupt, controlled by SYS0:LPCMPCFG 7h = IOC synchronous combined event, controlled by IOC:EVTCFG 8h = SYSTIM combined interrupt, interrupt flags are found here SYSTIM:MIS 9h = GPIO combined wake up interrupt, interrupt flags can be found here GPIO:MIS Ah = GPIO generic published event, controlled by GPIO:EVTCFG Bh = NoWrapper Flash interrupt indicating that the flash operation has completed, interrupt flags can be found here FLASH:MIS Ch = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS0 Dh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS1 Eh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS2 Fh = SPI0 combined interrupt request, interrupt flags can be found here SPI0:MIS 10h = ADC combined interrupt request, interrupt flags can be found here ADC:MIS0 11h = ADC general published event, interrupt flags can be found here ADC:MIS1 12h = LGPT0 combined interrupt, interrupt flags are found here LGPT0:MIS 13h = LGPT1 combined interrupt, interrupt flags are found here LGPT1:MIS 14h = DMA combined done interrupt, corresponding flags can be found here DMA:REQDONE 15h = DMA bus error, corresponds to DMA:ERROR.STATUS 16h = AES accelerator combined interrupt request, interrupt flags can be found here AES:MIS 17h = UART0 combined interrupt, interrupt flags are found here UART0:MIS 18h = Interrupt event from I2C0, interrupt flags can be found here I2C0:MIS 19h = SYSTIM heartbeat, can be set by SYSTIM:TIMEBIT 1Ah = SYSTIM interrupt driven by synchronizing LFTICK signal to SVT clock 1Bh = SYSTIM Channel 0 event, event flag is SYSTIM:MIS.EVT0 1Ch = SYSTIM Channel 1 event, event flag is SYSTIM:MIS.EVT1 1Dh = SYSTIM Channel 2 event, event flag is SYSTIM:MIS.EVT2 1Eh = SYSTIM Channel 3 event, event flag is SYSTIM:MIS.EVT3 1Fh = SYSTIM Channel 4 event, event flag is SYSTIM:MIS.EVT4 20h = LGPT0 compare/capture output event 0, controlled by LGPT0:C0CFG setting 21h = LGPT0 compare/capture output event 1, controlled by LGPT0:C1CFG setting 22h = LGPT0 compare/capture output event 2, controlled by LGPT0:C2CFG setting 23h = LGPT0 DMA request event, controlled by LGPT0:DMA setting 24h = LGPT0 ADC trigger event, controlled by LGPT0:ADCTRG setting 25h = LGPT1 compare/capture output event 0, controlled by LGPT1:C0CFG setting

Table 4-49. LGPT1IN0SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
				26h = LGPT1 compare/capture output event 1, controlled by LGPT1:C1CFG setting 27h = LGPT1 compare/capture output event 2, controlled by LGPT1:C2CFG setting 28h = LGPT1 DMA request event, controlled by LGPT1:DMA setting 29h = LGPT1 ADC trigger event, controlled by LGPT1:ADCTRG setting 2Ah = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC0 2Bh = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC1 2Ch = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC2 2Eh = LGPT2 compare/capture output event 0, controlled by LGPT2:C0CFG setting 2Fh = LGPT2 compare/capture output event 1, controlled by LGPT2:C1CFG setting 30h = LGPT0 compare/capture output event 2, controlled by LGPT2:C2CFG setting 31h = LGPT2 combined interrupt, interrupt flags are found here LGPT2:MIS 32h = LGPT2 DMA request event, controlled by LGPT2:DMA setting 33h = LGPT2 ADC trigger event, controlled by LGPT2:ADCTRG setting 34h = LGPT3 compare/capture output event 0, controlled by LGPT3:C0CFG setting 35h = LGPT3 compare/capture output event 1, controlled by LGPT3:C1CFG setting 36h = LGPT3 compare/capture output event 2, controlled by LGPT3:C2CFG setting 37h = LGPT3 combined interrupt, interrupt flags are found here LGPT3:MIS 38h = LGPT3 DMA request event, controlled by LGPT3:DMA setting 39h = LGPT3 ADC trigger event, controlled by LGPT3:ADCTRG setting

4.6.36 LGPT1IN1SEL Register (Offset = 480h) [Reset = 00000000h]

LGPT1IN1SEL is shown in Table 4-50.

Return to the Summary Table.

Output Selection for LGPT1IN1

Table 4-50. LGPT1IN1SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior

Table 4-50. LGPT1IN1SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive Ah = GPIO generic published event, controlled by GPIO:EVTCFG 11h = ADC general published event, interrupt flags can be found here ADC:MIS1 19h = SYSTIM heartbeat, can be set by SYSTIM:TIMEBIT 1Ah = SYSTIM interrupt driven by synchronizing LFTICK signal to SVT clock 1Bh = SYSTIM Channel 0 event, event flag is SYSTIM:MIS.EVT0 1Ch = SYSTIM Channel 1 event, event flag is SYSTIM:MIS.EVT1 1Dh = SYSTIM Channel 2 event, event flag is SYSTIM:MIS.EVT2 1Eh = SYSTIM Channel 3 event, event flag is SYSTIM:MIS.EVT3 1Fh = SYSTIM Channel 4 event, event flag is SYSTIM:MIS.EVT4 20h = LGPT0 compare/capture output event 0, controlled by LGPT0:C0CFG setting 21h = LGPT0 compare/capture output event 1, controlled by LGPT0:C1CFG setting 22h = LGPT0 compare/capture output event 2, controlled by LGPT0:C2CFG setting 23h = LGPT0 DMA request event, controlled by LGPT0:DMA setting 24h = LGPT0 ADC trigger event, controlled by LGPT0:ADCTRG setting 25h = LGPT1 compare/capture output event 0, controlled by LGPT1:C0CFG setting 26h = LGPT1 compare/capture output event 1, controlled by LGPT1:C1CFG setting 27h = LGPT1 compare/capture output event 2, controlled by LGPT1:C2CFG setting 28h = LGPT1 DMA request event, controlled by LGPT1:DMA setting 29h = LGPT1 ADC trigger event, controlled by LGPT1:ADCTRG setting 2Ah = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC0 2Bh = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC1 2Ch = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC2 2Eh = LGPT2 compare/capture output event 0, controlled by LGPT2:C0CFG setting 2Fh = LGPT2 compare/capture output event 1, controlled by LGPT2:C1CFG setting 30h = LGPT0 compare/capture output event 2, controlled by LGPT2:C2CFG setting 32h = LGPT2 DMA request event, controlled by LGPT2:DMA setting 33h = LGPT2 ADC trigger event, controlled by LGPT2:ADCTRG setting 34h = LGPT3 compare/capture output event 0, controlled by LGPT3:C0CFG setting 35h = LGPT3 compare/capture output event 1, controlled by LGPT3:C1CFG setting 36h = LGPT3 compare/capture output event 2, controlled by LGPT3:C2CFG setting 38h = LGPT3 DMA request event, controlled by LGPT3:DMA setting 39h = LGPT3 ADC trigger event, controlled by LGPT3:ADCTRG setting

4.6.37 LGPT1IN2SEL Register (Offset = 484h) [Reset = 00000000h]

LGPT1IN2SEL is shown in Table 4-51.

Return to the Summary Table.

Output Selection for LGPT1IN2

Table 4-51. LGPT1IN2SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior

Table 4-51. LGPT1IN2SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive Ah = GPIO generic published event, controlled by GPIO:EVTCFG 11h = ADC general published event, interrupt flags can be found here ADC:MIS1 19h = SYSTIM heartbeat, can be set by SYSTIM:TIMEBIT 1Ah = SYSTIM interrupt driven by synchronizing LFTICK signal to SVT clock 1Bh = SYSTIM Channel 0 event, event flag is SYSTIM:MIS.EVT0 1Ch = SYSTIM Channel 1 event, event flag is SYSTIM:MIS.EVT1 1Dh = SYSTIM Channel 2 event, event flag is SYSTIM:MIS.EVT2 1Eh = SYSTIM Channel 3 event, event flag is SYSTIM:MIS.EVT3 1Fh = SYSTIM Channel 4 event, event flag is SYSTIM:MIS.EVT4 20h = LGPT0 compare/capture output event 0, controlled by LGPT0:C0CFG setting 21h = LGPT0 compare/capture output event 1, controlled by LGPT0:C1CFG setting 22h = LGPT0 compare/capture output event 2, controlled by LGPT0:C2CFG setting 23h = LGPT0 DMA request event, controlled by LGPT0:DMA setting 24h = LGPT0 ADC trigger event, controlled by LGPT0:ADCTRG setting 25h = LGPT1 compare/capture output event 0, controlled by LGPT1:C0CFG setting 26h = LGPT1 compare/capture output event 1, controlled by LGPT1:C1CFG setting 27h = LGPT1 compare/capture output event 2, controlled by LGPT1:C2CFG setting 28h = LGPT1 DMA request event, controlled by LGPT1:DMA setting 29h = LGPT1 ADC trigger event, controlled by LGPT1:ADCTRG setting 2Ah = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC0 2Bh = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC1 2Ch = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC2 2Eh = LGPT2 compare/capture output event 0, controlled by LGPT2:C0CFG setting 2Fh = LGPT2 compare/capture output event 1, controlled by LGPT2:C1CFG setting 30h = LGPT0 compare/capture output event 2, controlled by LGPT2:C2CFG setting 32h = LGPT2 DMA request event, controlled by LGPT2:DMA setting 33h = LGPT2 ADC trigger event, controlled by LGPT2:ADCTRG setting 34h = LGPT3 compare/capture output event 0, controlled by LGPT3:C0CFG setting 35h = LGPT3 compare/capture output event 1, controlled by LGPT3:C1CFG setting 36h = LGPT3 compare/capture output event 2, controlled by LGPT3:C2CFG setting 38h = LGPT3 DMA request event, controlled by LGPT3:DMA setting 39h = LGPT3 ADC trigger event, controlled by LGPT3:ADCTRG setting

4.6.38 LGPT1TENSEL Register (Offset = 488h) [Reset = 00000000h]

LGPT1TENSEL is shown in Table 4-52.

Return to the Summary Table.

Output Selection for LGPT1TEN

Table 4-52. LGPT1TENSEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior

Table 4-52. LGPT1TENSEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive Ah = GPIO generic published event, controlled by GPIO:EVTCFG 11h = ADC general published event, interrupt flags can be found here ADC:MIS1 19h = SYSTIM heartbeat, can be set by SYSTIM:TIMEBIT 1Ah = SYSTIM interrupt driven by synchronizing LFTICK signal to SVT clock 1Bh = SYSTIM Channel 0 event, event flag is SYSTIM:MIS.EVT0 1Ch = SYSTIM Channel 1 event, event flag is SYSTIM:MIS.EVT1 1Dh = SYSTIM Channel 2 event, event flag is SYSTIM:MIS.EVT2 1Eh = SYSTIM Channel 3 event, event flag is SYSTIM:MIS.EVT3 1Fh = SYSTIM Channel 4 event, event flag is SYSTIM:MIS.EVT4 20h = LGPT0 compare/capture output event 0, controlled by LGPT0:C0CFG setting 21h = LGPT0 compare/capture output event 1, controlled by LGPT0:C1CFG setting 22h = LGPT0 compare/capture output event 2, controlled by LGPT0:C2CFG setting 23h = LGPT0 DMA request event, controlled by LGPT0:DMA setting 24h = LGPT0 ADC trigger event, controlled by LGPT0:ADCTRG setting 25h = LGPT1 compare/capture output event 0, controlled by LGPT1:C0CFG setting 26h = LGPT1 compare/capture output event 1, controlled by LGPT1:C1CFG setting 27h = LGPT1 compare/capture output event 2, controlled by LGPT1:C2CFG setting 28h = LGPT1 DMA request event, controlled by LGPT1:DMA setting 29h = LGPT1 ADC trigger event, controlled by LGPT1:ADCTRG setting 2Ah = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC0 2Bh = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC1 2Ch = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC2 2Eh = LGPT2 compare/capture output event 0, controlled by LGPT2:C0CFG setting 2Fh = LGPT2 compare/capture output event 1, controlled by LGPT2:C1CFG setting 30h = LGPT0 compare/capture output event 2, controlled by LGPT2:C2CFG setting 32h = LGPT2 DMA request event, controlled by LGPT2:DMA setting 33h = LGPT2 ADC trigger event, controlled by LGPT2:ADCTRG setting 34h = LGPT3 compare/capture output event 0, controlled by LGPT3:C0CFG setting 35h = LGPT3 compare/capture output event 1, controlled by LGPT3:C1CFG setting 36h = LGPT3 compare/capture output event 2, controlled by LGPT3:C2CFG setting 38h = LGPT3 DMA request event, controlled by LGPT3:DMA setting 39h = LGPT3 ADC trigger event, controlled by LGPT3:ADCTRG setting

4.6.39 LGPT2IN0SEL Register (Offset = 48Ch) [Reset = 00000000h]

LGPT2IN0SEL is shown in Table 4-53.

Return to the Summary Table.

Output Selection for LGPT2IN0

Table 4-53. LGPT2IN0SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior

Table 4-53. LGPT2IN0SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive 2h = PMU combined interrupt request for BATMON, interrupt flags can be found here PMUD:EVENT 3h = CKMD combined interrupt request, interrupt flags can be found here CKMD:MIS 4h = AON_RTC event, controlled by the RTC:IMASK setting 5h = DebugSS combined interrupt, interrupt flags can be found here DBGSS:MIS 6h = AON LPCMP interrupt, controlled by SYS0:LPCMPCFG 7h = IOC synchronous combined event, controlled by IOC:EVTCFG 8h = SYSTIM combined interrupt, interrupt flags are found here SYSTIM:MIS 9h = GPIO combined wake up interrupt, interrupt flags can be found here GPIO:MIS Ah = GPIO generic published event, controlled by GPIO:EVTCFG Bh = NoWrapper Flash interrupt indicating that the flash operation has completed, interrupt flags can be found here FLASH:MIS Ch = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS0 Dh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS1 Eh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS2 Fh = SPI0 combined interrupt request, interrupt flags can be found here SPI0:MIS 10h = ADC combined interrupt request, interrupt flags can be found here ADC:MIS0 11h = ADC general published event, interrupt flags can be found here ADC:MIS1 12h = LGPT0 combined interrupt, interrupt flags are found here LGPT0:MIS 13h = LGPT1 combined interrupt, interrupt flags are found here LGPT1:MIS 14h = DMA combined done interrupt, corresponding flags can be found here DMA:REQDONE 15h = DMA bus error, corresponds to DMA:ERROR.STATUS 16h = AES accelerator combined interrupt request, interrupt flags can be found here AES:MIS 17h = UART0 combined interrupt, interrupt flags are found here UART0:MIS 18h = Interrupt event from I2C0, interrupt flags can be found here I2C0:MIS 19h = SYSTIM heartbeat, can be set by SYSTIM:TIMEBIT 1Ah = SYSTIM interrupt driven by synchronizing LFTICK signal to SVT clock 1Bh = SYSTIM Channel 0 event, event flag is SYSTIM:MIS.EVT0 1Ch = SYSTIM Channel 1 event, event flag is SYSTIM:MIS.EVT1 1Dh = SYSTIM Channel 2 event, event flag is SYSTIM:MIS.EVT2 1Eh = SYSTIM Channel 3 event, event flag is SYSTIM:MIS.EVT3 1Fh = SYSTIM Channel 4 event, event flag is SYSTIM:MIS.EVT4 20h = LGPT0 compare/capture output event 0, controlled by LGPT0:C0CFG setting 21h = LGPT0 compare/capture output event 1, controlled by LGPT0:C1CFG setting 22h = LGPT0 compare/capture output event 2, controlled by LGPT0:C2CFG setting 23h = LGPT0 DMA request event, controlled by LGPT0:DMA setting 24h = LGPT0 ADC trigger event, controlled by LGPT0:ADCTRG setting 25h = LGPT1 compare/capture output event 0, controlled by LGPT1:C0CFG setting

Table 4-53. LGPT2IN0SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
				26h = LGPT1 compare/capture output event 1, controlled by LGPT1:C1CFG setting 27h = LGPT1 compare/capture output event 2, controlled by LGPT1:C2CFG setting 28h = LGPT1 DMA request event, controlled by LGPT1:DMA setting 29h = LGPT1 ADC trigger event, controlled by LGPT1:ADCTRG setting 2Ah = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC0 2Bh = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC1 2Ch = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC2 2Eh = LGPT2 compare/capture output event 0, controlled by LGPT2:C0CFG setting 2Fh = LGPT2 compare/capture output event 1, controlled by LGPT2:C1CFG setting 30h = LGPT0 compare/capture output event 2, controlled by LGPT2:C2CFG setting 31h = LGPT2 combined interrupt, interrupt flags are found here LGPT2:MIS 32h = LGPT2 DMA request event, controlled by LGPT2:DMA setting 33h = LGPT2 ADC trigger event, controlled by LGPT2:ADCTRG setting 34h = LGPT3 compare/capture output event 0, controlled by LGPT3:C0CFG setting 35h = LGPT3 compare/capture output event 1, controlled by LGPT3:C1CFG setting 36h = LGPT3 compare/capture output event 2, controlled by LGPT3:C2CFG setting 37h = LGPT3 combined interrupt, interrupt flags are found here LGPT3:MIS 38h = LGPT3 DMA request event, controlled by LGPT3:DMA setting 39h = LGPT3 ADC trigger event, controlled by LGPT3:ADCTRG setting

4.6.40 LGPT2IN1SEL Register (Offset = 490h) [Reset = 00000000h]

LGPT2IN1SEL is shown in Table 4-54.

Return to the Summary Table.

Output Selection for LGPT2IN1

Table 4-54. LGPT2IN1SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior

Table 4-54. LGPT2IN1SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive Ah = GPIO generic published event, controlled by GPIO:EVTCFG 11h = ADC general published event, interrupt flags can be found here ADC:MIS1 19h = SYSTIM heartbeat, can be set by SYSTIM:TIMEBIT 1Ah = SYSTIM interrupt driven by synchronizing LFTICK signal to SVT clock 1Bh = SYSTIM Channel 0 event, event flag is SYSTIM:MIS.EVT0 1Ch = SYSTIM Channel 1 event, event flag is SYSTIM:MIS.EVT1 1Dh = SYSTIM Channel 2 event, event flag is SYSTIM:MIS.EVT2 1Eh = SYSTIM Channel 3 event, event flag is SYSTIM:MIS.EVT3 1Fh = SYSTIM Channel 4 event, event flag is SYSTIM:MIS.EVT4 20h = LGPT0 compare/capture output event 0, controlled by LGPT0:C0CFG setting 21h = LGPT0 compare/capture output event 1, controlled by LGPT0:C1CFG setting 22h = LGPT0 compare/capture output event 2, controlled by LGPT0:C2CFG setting 23h = LGPT0 DMA request event, controlled by LGPT0:DMA setting 24h = LGPT0 ADC trigger event, controlled by LGPT0:ADCTRG setting 25h = LGPT1 compare/capture output event 0, controlled by LGPT1:C0CFG setting 26h = LGPT1 compare/capture output event 1, controlled by LGPT1:C1CFG setting 27h = LGPT1 compare/capture output event 2, controlled by LGPT1:C2CFG setting 28h = LGPT1 DMA request event, controlled by LGPT1:DMA setting 29h = LGPT1 ADC trigger event, controlled by LGPT1:ADCTRG setting 2Ah = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC0 2Bh = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC1 2Ch = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC2 2Eh = LGPT2 compare/capture output event 0, controlled by LGPT2:C0CFG setting 2Fh = LGPT2 compare/capture output event 1, controlled by LGPT2:C1CFG setting 30h = LGPT0 compare/capture output event 2, controlled by LGPT2:C2CFG setting 32h = LGPT2 DMA request event, controlled by LGPT2:DMA setting 33h = LGPT2 ADC trigger event, controlled by LGPT2:ADCTRG setting 34h = LGPT3 compare/capture output event 0, controlled by LGPT3:C0CFG setting 35h = LGPT3 compare/capture output event 1, controlled by LGPT3:C1CFG setting 36h = LGPT3 compare/capture output event 2, controlled by LGPT3:C2CFG setting 38h = LGPT3 DMA request event, controlled by LGPT3:DMA setting 39h = LGPT3 ADC trigger event, controlled by LGPT3:ADCTRG setting

4.6.41 LGPT2IN2SEL Register (Offset = 494h) [Reset = 00000000h]

LGPT2IN2SEL is shown in Table 4-55.

Return to the Summary Table.

Output Selection for LGPT2IN2

Table 4-55. LGPT2IN2SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior

Table 4-55. LGPT2IN2SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive Ah = GPIO generic published event, controlled by GPIO:EVTCFG 11h = ADC general published event, interrupt flags can be found here ADC:MIS1 19h = SYSTIM heartbeat, can be set by SYSTIM:TIMEBIT 1Ah = SYSTIM interrupt driven by synchronizing LFTICK signal to SVT clock 1Bh = SYSTIM Channel 0 event, event flag is SYSTIM:MIS.EVT0 1Ch = SYSTIM Channel 1 event, event flag is SYSTIM:MIS.EVT1 1Dh = SYSTIM Channel 2 event, event flag is SYSTIM:MIS.EVT2 1Eh = SYSTIM Channel 3 event, event flag is SYSTIM:MIS.EVT3 1Fh = SYSTIM Channel 4 event, event flag is SYSTIM:MIS.EVT4 20h = LGPT0 compare/capture output event 0, controlled by LGPT0:C0CFG setting 21h = LGPT0 compare/capture output event 1, controlled by LGPT0:C1CFG setting 22h = LGPT0 compare/capture output event 2, controlled by LGPT0:C2CFG setting 23h = LGPT0 DMA request event, controlled by LGPT0:DMA setting 24h = LGPT0 ADC trigger event, controlled by LGPT0:ADCTRG setting 25h = LGPT1 compare/capture output event 0, controlled by LGPT1:C0CFG setting 26h = LGPT1 compare/capture output event 1, controlled by LGPT1:C1CFG setting 27h = LGPT1 compare/capture output event 2, controlled by LGPT1:C2CFG setting 28h = LGPT1 DMA request event, controlled by LGPT1:DMA setting 29h = LGPT1 ADC trigger event, controlled by LGPT1:ADCTRG setting 2Ah = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC0 2Bh = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC1 2Ch = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC2 2Eh = LGPT2 compare/capture output event 0, controlled by LGPT2:C0CFG setting 2Fh = LGPT2 compare/capture output event 1, controlled by LGPT2:C1CFG setting 30h = LGPT0 compare/capture output event 2, controlled by LGPT2:C2CFG setting 32h = LGPT2 DMA request event, controlled by LGPT2:DMA setting 33h = LGPT2 ADC trigger event, controlled by LGPT2:ADCTRG setting 34h = LGPT3 compare/capture output event 0, controlled by LGPT3:C0CFG setting 35h = LGPT3 compare/capture output event 1, controlled by LGPT3:C1CFG setting 36h = LGPT3 compare/capture output event 2, controlled by LGPT3:C2CFG setting 38h = LGPT3 DMA request event, controlled by LGPT3:DMA setting 39h = LGPT3 ADC trigger event, controlled by LGPT3:ADCTRG setting

4.6.42 LGPT2TENSEL Register (Offset = 498h) [Reset = 00000000h]

LGPT2TENSEL is shown in Table 4-56.

Return to the Summary Table.

Output Selection for LGPT2TEN

Table 4-56. LGPT2TENSEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior

Table 4-56. LGPT2TENSEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive Ah = GPIO generic published event, controlled by GPIO:EVTCFG 11h = ADC general published event, interrupt flags can be found here ADC:MIS1 19h = SYSTIM heartbeat, can be set by SYSTIM:TIMEBIT 1Ah = SYSTIM interrupt driven by synchronizing LFTICK signal to SVT clock 1Bh = SYSTIM Channel 0 event, event flag is SYSTIM:MIS.EVT0 1Ch = SYSTIM Channel 1 event, event flag is SYSTIM:MIS.EVT1 1Dh = SYSTIM Channel 2 event, event flag is SYSTIM:MIS.EVT2 1Eh = SYSTIM Channel 3 event, event flag is SYSTIM:MIS.EVT3 1Fh = SYSTIM Channel 4 event, event flag is SYSTIM:MIS.EVT4 20h = LGPT0 compare/capture output event 0, controlled by LGPT0:C0CFG setting 21h = LGPT0 compare/capture output event 1, controlled by LGPT0:C1CFG setting 22h = LGPT0 compare/capture output event 2, controlled by LGPT0:C2CFG setting 23h = LGPT0 DMA request event, controlled by LGPT0:DMA setting 24h = LGPT0 ADC trigger event, controlled by LGPT0:ADCTRG setting 25h = LGPT1 compare/capture output event 0, controlled by LGPT1:C0CFG setting 26h = LGPT1 compare/capture output event 1, controlled by LGPT1:C1CFG setting 27h = LGPT1 compare/capture output event 2, controlled by LGPT1:C2CFG setting 28h = LGPT1 DMA request event, controlled by LGPT1:DMA setting 29h = LGPT1 ADC trigger event, controlled by LGPT1:ADCTRG setting 2Ah = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC0 2Bh = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC1 2Ch = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC2 2Eh = LGPT2 compare/capture output event 0, controlled by LGPT2:C0CFG setting 2Fh = LGPT2 compare/capture output event 1, controlled by LGPT2:C1CFG setting 30h = LGPT0 compare/capture output event 2, controlled by LGPT2:C2CFG setting 32h = LGPT2 DMA request event, controlled by LGPT2:DMA setting 33h = LGPT2 ADC trigger event, controlled by LGPT2:ADCTRG setting 34h = LGPT3 compare/capture output event 0, controlled by LGPT3:C0CFG setting 35h = LGPT3 compare/capture output event 1, controlled by LGPT3:C1CFG setting 36h = LGPT3 compare/capture output event 2, controlled by LGPT3:C2CFG setting 38h = LGPT3 DMA request event, controlled by LGPT3:DMA setting 39h = LGPT3 ADC trigger event, controlled by LGPT3:ADCTRG setting

4.6.43 LGPT3IN0SEL Register (Offset = 49Ch) [Reset = 00000000h]

LGPT3IN0SEL is shown in Table 4-57.

Return to the Summary Table.

Output Selection for LGPT3IN0

Table 4-57. LGPT3IN0SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior

Table 4-57. LGPT3IN0SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive 2h = PMU combined interrupt request for BATMON, interrupt flags can be found here PMUD:EVENT 3h = CKMD combined interrupt request, interrupt flags can be found here CKMD:MIS 4h = AON_RTC event, controlled by the RTC:IMASK setting 5h = DebugSS combined interrupt, interrupt flags can be found here DBGSS:MIS 6h = AON LPCMP interrupt, controlled by SYS0:LPCMPCFG 7h = IOC synchronous combined event, controlled by IOC:EVTCFG 8h = SYSTIM combined interrupt, interrupt flags are found here SYSTIM:MIS 9h = GPIO combined wake up interrupt, interrupt flags can be found here GPIO:MIS Ah = GPIO generic published event, controlled by GPIO:EVTCFG Bh = NoWrapper Flash interrupt indicating that the flash operation has completed, interrupt flags can be found here FLASH:MIS Ch = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS0 Dh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS1 Eh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS2 Fh = SPI0 combined interrupt request, interrupt flags can be found here SPI0:MIS 10h = ADC combined interrupt request, interrupt flags can be found here ADC:MIS0 11h = ADC general published event, interrupt flags can be found here ADC:MIS1 12h = LGPT0 combined interrupt, interrupt flags are found here LGPT0:MIS 13h = LGPT1 combined interrupt, interrupt flags are found here LGPT1:MIS 14h = DMA combined done interrupt, corresponding flags can be found here DMA:REQDONE 15h = DMA bus error, corresponds to DMA:ERROR.STATUS 16h = AES accelerator combined interrupt request, interrupt flags can be found here AES:MIS 17h = UART0 combined interrupt, interrupt flags are found here UART0:MIS 18h = Interrupt event from I2C0, interrupt flags can be found here I2C0:MIS 19h = SYSTIM heartbeat, can be set by SYSTIM:TIMEBIT 1Ah = SYSTIM interrupt driven by synchronizing LFTICK signal to SVT clock 1Bh = SYSTIM Channel 0 event, event flag is SYSTIM:MIS.EVT0 1Ch = SYSTIM Channel 1 event, event flag is SYSTIM:MIS.EVT1 1Dh = SYSTIM Channel 2 event, event flag is SYSTIM:MIS.EVT2 1Eh = SYSTIM Channel 3 event, event flag is SYSTIM:MIS.EVT3 1Fh = SYSTIM Channel 4 event, event flag is SYSTIM:MIS.EVT4 20h = LGPT0 compare/capture output event 0, controlled by LGPT0:C0CFG setting 21h = LGPT0 compare/capture output event 1, controlled by LGPT0:C1CFG setting 22h = LGPT0 compare/capture output event 2, controlled by LGPT0:C2CFG setting 23h = LGPT0 DMA request event, controlled by LGPT0:DMA setting 24h = LGPT0 ADC trigger event, controlled by LGPT0:ADCTRG setting 25h = LGPT1 compare/capture output event 0, controlled by LGPT1:C0CFG setting

Table 4-57. LGPT3IN0SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
				26h = LGPT1 compare/capture output event 1, controlled by LGPT1:C1CFG setting 27h = LGPT1 compare/capture output event 2, controlled by LGPT1:C2CFG setting 28h = LGPT1 DMA request event, controlled by LGPT1:DMA setting 29h = LGPT1 ADC trigger event, controlled by LGPT1:ADCTRG setting 2Ah = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC0 2Bh = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC1 2Ch = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC2 2Eh = LGPT2 compare/capture output event 0, controlled by LGPT2:C0CFG setting 2Fh = LGPT2 compare/capture output event 1, controlled by LGPT2:C1CFG setting 30h = LGPT0 compare/capture output event 2, controlled by LGPT2:C2CFG setting 31h = LGPT2 combined interrupt, interrupt flags are found here LGPT2:MIS 32h = LGPT2 DMA request event, controlled by LGPT2:DMA setting 33h = LGPT2 ADC trigger event, controlled by LGPT2:ADCTRG setting 34h = LGPT3 compare/capture output event 0, controlled by LGPT3:C0CFG setting 35h = LGPT3 compare/capture output event 1, controlled by LGPT3:C1CFG setting 36h = LGPT3 compare/capture output event 2, controlled by LGPT3:C2CFG setting 37h = LGPT3 combined interrupt, interrupt flags are found here LGPT3:MIS 38h = LGPT3 DMA request event, controlled by LGPT3:DMA setting 39h = LGPT3 ADC trigger event, controlled by LGPT3:ADCTRG setting

4.6.44 LGPT3IN1SEL Register (Offset = 4A0h) [Reset = 00000000h]

LGPT3IN1SEL is shown in Table 4-58.

Return to the Summary Table.

Output Selection for LGPT3IN1

Table 4-58. LGPT3IN1SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior

Table 4-58. LGPT3IN1SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive Ah = GPIO generic published event, controlled by GPIO:EVTCFG 11h = ADC general published event, interrupt flags can be found here ADC:MIS1 19h = SYSTIM heartbeat, can be set by SYSTIM:TIMEBIT 1Ah = SYSTIM interrupt driven by synchronizing LFTICK signal to SVT clock 1Bh = SYSTIM Channel 0 event, event flag is SYSTIM:MIS.EVT0 1Ch = SYSTIM Channel 1 event, event flag is SYSTIM:MIS.EVT1 1Dh = SYSTIM Channel 2 event, event flag is SYSTIM:MIS.EVT2 1Eh = SYSTIM Channel 3 event, event flag is SYSTIM:MIS.EVT3 1Fh = SYSTIM Channel 4 event, event flag is SYSTIM:MIS.EVT4 20h = LGPT0 compare/capture output event 0, controlled by LGPT0:C0CFG setting 21h = LGPT0 compare/capture output event 1, controlled by LGPT0:C1CFG setting 22h = LGPT0 compare/capture output event 2, controlled by LGPT0:C2CFG setting 23h = LGPT0 DMA request event, controlled by LGPT0:DMA setting 24h = LGPT0 ADC trigger event, controlled by LGPT0:ADCTRG setting 25h = LGPT1 compare/capture output event 0, controlled by LGPT1:C0CFG setting 26h = LGPT1 compare/capture output event 1, controlled by LGPT1:C1CFG setting 27h = LGPT1 compare/capture output event 2, controlled by LGPT1:C2CFG setting 28h = LGPT1 DMA request event, controlled by LGPT1:DMA setting 29h = LGPT1 ADC trigger event, controlled by LGPT1:ADCTRG setting 2Ah = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC0 2Bh = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC1 2Ch = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC2 2Eh = LGPT2 compare/capture output event 0, controlled by LGPT2:C0CFG setting 2Fh = LGPT2 compare/capture output event 1, controlled by LGPT2:C1CFG setting 30h = LGPT0 compare/capture output event 2, controlled by LGPT2:C2CFG setting 32h = LGPT2 DMA request event, controlled by LGPT2:DMA setting 33h = LGPT2 ADC trigger event, controlled by LGPT2:ADCTRG setting 34h = LGPT3 compare/capture output event 0, controlled by LGPT3:C0CFG setting 35h = LGPT3 compare/capture output event 1, controlled by LGPT3:C1CFG setting 36h = LGPT3 compare/capture output event 2, controlled by LGPT3:C2CFG setting 38h = LGPT3 DMA request event, controlled by LGPT3:DMA setting 39h = LGPT3 ADC trigger event, controlled by LGPT3:ADCTRG setting

4.6.45 LGPT3IN2SEL Register (Offset = 4A4h) [Reset = 00000000h]

LGPT3IN2SEL is shown in Table 4-59.

Return to the Summary Table.

Output Selection for LGPT3IN2

Table 4-59. LGPT3IN2SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior

Table 4-59. LGPT3IN2SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive Ah = GPIO generic published event, controlled by GPIO:EVTCFG 11h = ADC general published event, interrupt flags can be found here ADC:MIS1 19h = SYSTIM heartbeat, can be set by SYSTIM:TIMEBIT 1Ah = SYSTIM interrupt driven by synchronizing LFTICK signal to SVT clock 1Bh = SYSTIM Channel 0 event, event flag is SYSTIM:MIS.EVT0 1Ch = SYSTIM Channel 1 event, event flag is SYSTIM:MIS.EVT1 1Dh = SYSTIM Channel 2 event, event flag is SYSTIM:MIS.EVT2 1Eh = SYSTIM Channel 3 event, event flag is SYSTIM:MIS.EVT3 1Fh = SYSTIM Channel 4 event, event flag is SYSTIM:MIS.EVT4 20h = LGPT0 compare/capture output event 0, controlled by LGPT0:C0CFG setting 21h = LGPT0 compare/capture output event 1, controlled by LGPT0:C1CFG setting 22h = LGPT0 compare/capture output event 2, controlled by LGPT0:C2CFG setting 23h = LGPT0 DMA request event, controlled by LGPT0:DMA setting 24h = LGPT0 ADC trigger event, controlled by LGPT0:ADCTRG setting 25h = LGPT1 compare/capture output event 0, controlled by LGPT1:C0CFG setting 26h = LGPT1 compare/capture output event 1, controlled by LGPT1:C1CFG setting 27h = LGPT1 compare/capture output event 2, controlled by LGPT1:C2CFG setting 28h = LGPT1 DMA request event, controlled by LGPT1:DMA setting 29h = LGPT1 ADC trigger event, controlled by LGPT1:ADCTRG setting 2Ah = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC0 2Bh = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC1 2Ch = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC2 2Eh = LGPT2 compare/capture output event 0, controlled by LGPT2:C0CFG setting 2Fh = LGPT2 compare/capture output event 1, controlled by LGPT2:C1CFG setting 30h = LGPT0 compare/capture output event 2, controlled by LGPT2:C2CFG setting 32h = LGPT2 DMA request event, controlled by LGPT2:DMA setting 33h = LGPT2 ADC trigger event, controlled by LGPT2:ADCTRG setting 34h = LGPT3 compare/capture output event 0, controlled by LGPT3:C0CFG setting 35h = LGPT3 compare/capture output event 1, controlled by LGPT3:C1CFG setting 36h = LGPT3 compare/capture output event 2, controlled by LGPT3:C2CFG setting 38h = LGPT3 DMA request event, controlled by LGPT3:DMA setting 39h = LGPT3 ADC trigger event, controlled by LGPT3:ADCTRG setting

4.6.46 LGPT3TENSEL Register (Offset = 4A8h) [Reset = 00000000h]

LGPT3TENSEL is shown in Table 4-60.

Return to the Summary Table.

Output Selection for LGPT3TEN

Table 4-60. LGPT3TENSEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior

Table 4-60. LGPT3TENSEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive Ah = GPIO generic published event, controlled by GPIO:EVTCFG 11h = ADC general published event, interrupt flags can be found here ADC:MIS1 19h = SYSTIM heartbeat, can be set by SYSTIM:TIMEBIT 1Ah = SYSTIM interrupt driven by synchronizing LFTICK signal to SVT clock 1Bh = SYSTIM Channel 0 event, event flag is SYSTIM:MIS.EVT0 1Ch = SYSTIM Channel 1 event, event flag is SYSTIM:MIS.EVT1 1Dh = SYSTIM Channel 2 event, event flag is SYSTIM:MIS.EVT2 1Eh = SYSTIM Channel 3 event, event flag is SYSTIM:MIS.EVT3 1Fh = SYSTIM Channel 4 event, event flag is SYSTIM:MIS.EVT4 20h = LGPT0 compare/capture output event 0, controlled by LGPT0:C0CFG setting 21h = LGPT0 compare/capture output event 1, controlled by LGPT0:C1CFG setting 22h = LGPT0 compare/capture output event 2, controlled by LGPT0:C2CFG setting 23h = LGPT0 DMA request event, controlled by LGPT0:DMA setting 24h = LGPT0 ADC trigger event, controlled by LGPT0:ADCTRG setting 25h = LGPT1 compare/capture output event 0, controlled by LGPT1:C0CFG setting 26h = LGPT1 compare/capture output event 1, controlled by LGPT1:C1CFG setting 27h = LGPT1 compare/capture output event 2, controlled by LGPT1:C2CFG setting 28h = LGPT1 DMA request event, controlled by LGPT1:DMA setting 29h = LGPT1 ADC trigger event, controlled by LGPT1:ADCTRG setting 2Ah = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC0 2Bh = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC1 2Ch = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC2 2Eh = LGPT2 compare/capture output event 0, controlled by LGPT2:C0CFG setting 2Fh = LGPT2 compare/capture output event 1, controlled by LGPT2:C1CFG setting 30h = LGPT0 compare/capture output event 2, controlled by LGPT2:C2CFG setting 32h = LGPT2 DMA request event, controlled by LGPT2:DMA setting 33h = LGPT2 ADC trigger event, controlled by LGPT2:ADCTRG setting 34h = LGPT3 compare/capture output event 0, controlled by LGPT3:C0CFG setting 35h = LGPT3 compare/capture output event 1, controlled by LGPT3:C1CFG setting 36h = LGPT3 compare/capture output event 2, controlled by LGPT3:C2CFG setting 38h = LGPT3 DMA request event, controlled by LGPT3:DMA setting 39h = LGPT3 ADC trigger event, controlled by LGPT3:ADCTRG setting

4.6.47 LRFDIN0SEL Register (Offset = 4ACh) [Reset = 0000001Dh]

LRFDIN0SEL is shown in Table 4-61.

Return to the Summary Table.

Output Selection for LRFDIN0

Table 4-61. LRFDIN0SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
5-0	PUBID	R	1Dh	Read only selection value 1Dh = SYSTIM Channel 2 event, event flag is SYSTIM:MIS.EVT2

4.6.48 LRFDIN1SEL Register (Offset = 4B0h) [Reset = 0000001Eh]

LRFDIN1SEL is shown in Table 4-62.

Return to the Summary Table.

Output Selection for LRFDIN1

Table 4-62. LRFDIN1SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
5-0	PUBID	R	1Eh	Read only selection value 1Eh = SYSTIM Channel 3 event, event flag is SYSTIM:MIS.EVT3

4.6.49 LRFDIN2SEL Register (Offset = 4B4h) [Reset = 0000001Fh]

LRFDIN2SEL is shown in Table 4-63.

Return to the Summary Table.

Output Selection for LRFDIN2

Table 4-63. LRFDIN2SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
5-0	PUBID	R	1Fh	Read only selection value 1Fh = SYSTIM Channel 4 event, event flag is SYSTIM:MIS.EVT4

4.6.50 DMACH0SEL Register (Offset = C00h) [Reset = 00000000h]

DMACH0SEL is shown in Table 4-64.

Return to the Summary Table.

Output Selection for DMA CH0

Table 4-64. DMACH0SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-29	RESERVED	R	0h	Reserved
28-3	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
2-0	IPID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Selects spi0txtrg as channel source 7h = Selects uart0rxtrg as channel source

4.6.51 DMACH1SEL Register (Offset = C04h) [Reset = 00000000h]

DMACH1SEL is shown in Table 4-65.

Return to the Summary Table.

Output Selection for DMA CH1

Table 4-65. DMACH1SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-29	RESERVED	R	0h	Reserved
28-3	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
2-0	IPID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 1h = Selects spi0rxtrg as channel source 6h = Selects uart0txtrg as channel source

4.6.52 DMACH2SEL Register (Offset = C08h) [Reset = 00000000h]

DMACH2SEL is shown in Table 4-66.

Return to the Summary Table.

Output Selection for DMA CH2

Table 4-66. DMACH2SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-29	RESERVED	R	0h	Reserved
28-3	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
2-0	IPID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 2h = Reserved value. Should not be programmed. 6h = Selects uart0txtrg as channel source

4.6.53 DMACH3SEL Register (Offset = C0Ch) [Reset = 00000000h]

DMACH3SEL is shown in Table 4-67.

Return to the Summary Table.

Output Selection for DMA CH3

Table 4-67. DMACH3SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-29	RESERVED	R	0h	Reserved
28-3	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
2-0	IPID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 5h = Selects adc0trg as channel source 7h = Selects uart0rxtrg as channel source

4.6.54 DMACH4SEL Register (Offset = C10h) [Reset = 00000000h]

DMACH4SEL is shown in Table 4-68.

Return to the Summary Table.

Output Selection for DMA CH4

Table 4-68. DMACH4SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-29	RESERVED	R	0h	Reserved
28-3	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
2-0	IPID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 2h = Reserved value. Should not be programmed. 3h = Selects laestrga as channel source

4.6.55 DMACH5SEL Register (Offset = C14h) [Reset = 00000000h]

DMACH5SEL is shown in Table 4-69.

Return to the Summary Table.

Output Selection for DMA CH5

Table 4-69. DMACH5SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-29	RESERVED	R	0h	Reserved
28-3	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
2-0	IPID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 4h = Selects laestrgb as channel source 5h = Selects adc0trg as channel source

4.6.56 DMACH6SEL Register (Offset = C18h) [Reset = 00000000h]

DMACH6SEL is shown in Table 4-70.

Return to the Summary Table.

Output Selection for DMA CH6

Table 4-70. DMACH6SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-17	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
16	EDGDETDIS	R/W	0h	Edge detect disable. 0: Enabled. 1: Disabled
15-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior

Table 4-70. DMACH6SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive 2h = PMU combined interrupt request for BATMON, interrupt flags can be found here PMUD:EVENT 3h = CKMD combined interrupt request, interrupt flags can be found here CKMD:MIS 4h = AON_RTC event, controlled by the RTC:IMASK setting 5h = DebugSS combined interrupt, interrupt flags can be found here DBGSS:MIS 6h = AON LPCMP interrupt, controlled by SYS0:LPCMPCFG 7h = IOC synchronous combined event, controlled by IOC:EVTCFG 8h = SYSTIM combined interrupt, interrupt flags are found here SYSTIM:MIS 9h = GPIO combined wake up interrupt, interrupt flags can be found here GPIO:MIS Ah = GPIO generic published event, controlled by GPIO:EVTCFG Bh = NoWrapper Flash interrupt indicating that the flash operation has completed, interrupt flags can be found here FLASH:MIS Ch = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS0 Dh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS1 Eh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS2 Fh = SPI0 combined interrupt request, interrupt flags can be found here SPI0:MIS 10h = ADC combined interrupt request, interrupt flags can be found here ADC:MIS0 11h = ADC general published event, interrupt flags can be found here ADC:MIS1 12h = LGPT0 combined interrupt, interrupt flags are found here LGPT0:MIS 13h = LGPT1 combined interrupt, interrupt flags are found here LGPT1:MIS 14h = DMA combined done interrupt, corresponding flags can be found here DMA:REQDONE 15h = DMA bus error, corresponds to DMA:ERROR.STATUS 16h = AES accelerator combined interrupt request, interrupt flags can be found here AES:MIS 17h = UART0 combined interrupt, interrupt flags are found here UART0:MIS 18h = Interrupt event from I2C0, interrupt flags can be found here I2C0:MIS 19h = SYSTIM heartbeat, can be set by SYSTIM:TIMEBIT 1Ah = SYSTIM interrupt driven by synchronizing LFTICK signal to SVT clock 1Bh = SYSTIM Channel 0 event, event flag is SYSTIM:MIS.EVT0 1Ch = SYSTIM Channel 1 event, event flag is SYSTIM:MIS.EVT1 1Dh = SYSTIM Channel 2 event, event flag is SYSTIM:MIS.EVT2 1Eh = SYSTIM Channel 3 event, event flag is SYSTIM:MIS.EVT3 1Fh = SYSTIM Channel 4 event, event flag is SYSTIM:MIS.EVT4 20h = LGPT0 compare/capture output event 0, controlled by LGPT0:C0CFG setting 21h = LGPT0 compare/capture output event 1, controlled by LGPT0:C1CFG setting 22h = LGPT0 compare/capture output event 2, controlled by LGPT0:C2CFG setting 23h = LGPT0 DMA request event, controlled by LGPT0:DMA setting 24h = LGPT0 ADC trigger event, controlled by LGPT0:ADCTRG setting 25h = LGPT1 compare/capture output event 0, controlled by LGPT1:C0CFG setting

Table 4-70. DMACH6SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
				26h = LGPT1 compare/capture output event 1, controlled by LGPT1:C1CFG setting 27h = LGPT1 compare/capture output event 2, controlled by LGPT1:C2CFG setting 28h = LGPT1 DMA request event, controlled by LGPT1:DMA setting 29h = LGPT1 ADC trigger event, controlled by LGPT1:ADCTRG setting 2Ah = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC0 2Bh = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC1 2Ch = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC2 2Eh = LGPT2 compare/capture output event 0, controlled by LGPT2:C0CFG setting 2Fh = LGPT2 compare/capture output event 1, controlled by LGPT2:C1CFG setting 30h = LGPT0 compare/capture output event 2, controlled by LGPT2:C2CFG setting 31h = LGPT2 combined interrupt, interrupt flags are found here LGPT2:MIS 32h = LGPT2 DMA request event, controlled by LGPT2:DMA setting 33h = LGPT2 ADC trigger event, controlled by LGPT2:ADCTRG setting 34h = LGPT3 compare/capture output event 0, controlled by LGPT3:C0CFG setting 35h = LGPT3 compare/capture output event 1, controlled by LGPT3:C1CFG setting 36h = LGPT3 compare/capture output event 2, controlled by LGPT3:C2CFG setting 37h = LGPT3 combined interrupt, interrupt flags are found here LGPT3:MIS 38h = LGPT3 DMA request event, controlled by LGPT3:DMA setting 39h = LGPT3 ADC trigger event, controlled by LGPT3:ADCTRG setting

4.6.57 DMACH7SEL Register (Offset = C1Ch) [Reset = 00000000h]

DMACH7SEL is shown in Table 4-71.

Return to the Summary Table.

Output Selection for DMA CH7

Table 4-71. DMACH7SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-17	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior
16	EDGDETDIS	R/W	0h	Edge detect disable. 0: Enabled. 1: Disabled
15-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior

Table 4-71. DMACH7SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
5-0	PUBID	R/W	0h	Read/write selection value. Writing any other value than values defined by a ENUM may result in undefined behavior. 0h = Always inactive 2h = PMU combined interrupt request for BATMON, interrupt flags can be found here PMUD:EVENT 3h = CKMD combined interrupt request, interrupt flags can be found here CKMD:MIS 4h = AON_RTC event, controlled by the RTC:IMASK setting 5h = DebugSS combined interrupt, interrupt flags can be found here DBGSS:MIS 6h = AON LPCMP interrupt, controlled by SYS0:LPCMPCFG 7h = IOC synchronous combined event, controlled by IOC:EVTCFG 8h = SYSTIM combined interrupt, interrupt flags are found here SYSTIM:MIS 9h = GPIO combined wake up interrupt, interrupt flags can be found here GPIO:MIS Ah = GPIO generic published event, controlled by GPIO:EVTCFG Bh = NoWrapper Flash interrupt indicating that the flash operation has completed, interrupt flags can be found here FLASH:MIS Ch = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS0 Dh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS1 Eh = LRFD combined event, interrupt flags can be found here LRFDDBELL:MIS2 Fh = SPI0 combined interrupt request, interrupt flags can be found here SPI0:MIS 10h = ADC combined interrupt request, interrupt flags can be found here ADC:MIS0 11h = ADC general published event, interrupt flags can be found here ADC:MIS1 12h = LGPT0 combined interrupt, interrupt flags are found here LGPT0:MIS 13h = LGPT1 combined interrupt, interrupt flags are found here LGPT1:MIS 14h = DMA combined done interrupt, corresponding flags can be found here DMA:REQDONE 15h = DMA bus error, corresponds to DMA:ERROR.STATUS 16h = AES accelerator combined interrupt request, interrupt flags can be found here AES:MIS 17h = UART0 combined interrupt, interrupt flags are found here UART0:MIS 18h = Interrupt event from I2C0, interrupt flags can be found here I2C0:MIS 19h = SYSTIM heartbeat, can be set by SYSTIM:TIMEBIT 1Ah = SYSTIM interrupt driven by synchronizing LFTICK signal to SVT clock 1Bh = SYSTIM Channel 0 event, event flag is SYSTIM:MIS.EVT0 1Ch = SYSTIM Channel 1 event, event flag is SYSTIM:MIS.EVT1 1Dh = SYSTIM Channel 2 event, event flag is SYSTIM:MIS.EVT2 1Eh = SYSTIM Channel 3 event, event flag is SYSTIM:MIS.EVT3 1Fh = SYSTIM Channel 4 event, event flag is SYSTIM:MIS.EVT4 20h = LGPT0 compare/capture output event 0, controlled by LGPT0:C0CFG setting 21h = LGPT0 compare/capture output event 1, controlled by LGPT0:C1CFG setting 22h = LGPT0 compare/capture output event 2, controlled by LGPT0:C2CFG setting 23h = LGPT0 DMA request event, controlled by LGPT0:DMA setting 24h = LGPT0 ADC trigger event, controlled by LGPT0:ADCTRG setting 25h = LGPT1 compare/capture output event 0, controlled by LGPT1:C0CFG setting

Table 4-71. DMACH7SEL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
				26h = LGPT1 compare/capture output event 1, controlled by LGPT1:C1CFG setting 27h = LGPT1 compare/capture output event 2, controlled by LGPT1:C2CFG setting 28h = LGPT1 DMA request event, controlled by LGPT1:DMA setting 29h = LGPT1 ADC trigger event, controlled by LGPT1:ADCTRG setting 2Ah = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC0 2Bh = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC1 2Ch = LRFD interrupt to SYSTIM, controlled by LRFDDBELL:SYSTIMOEV.SRC2 2Eh = LGPT2 compare/capture output event 0, controlled by LGPT2:C0CFG setting 2Fh = LGPT2 compare/capture output event 1, controlled by LGPT2:C1CFG setting 30h = LGPT0 compare/capture output event 2, controlled by LGPT2:C2CFG setting 31h = LGPT2 combined interrupt, interrupt flags are found here LGPT2:MIS 32h = LGPT2 DMA request event, controlled by LGPT2:DMA setting 33h = LGPT2 ADC trigger event, controlled by LGPT2:ADCTRG setting 34h = LGPT3 compare/capture output event 0, controlled by LGPT3:C0CFG setting 35h = LGPT3 compare/capture output event 1, controlled by LGPT3:C1CFG setting 36h = LGPT3 compare/capture output event 2, controlled by LGPT3:C2CFG setting 37h = LGPT3 combined interrupt, interrupt flags are found here LGPT3:MIS 38h = LGPT3 DMA request event, controlled by LGPT3:DMA setting 39h = LGPT3 ADC trigger event, controlled by LGPT3:ADCTRG setting

Chapter 5

Debug Subsystem

This chapter discusses the features of the debug subsystem (DEBUGSS).

5.1 Introduction

The debug subsystem (DEBUGSS) interfaces the serial wire debug (SWD) two-wire physical interface to multiple debug functions within the device. Debugging of processor execution and the device state are supported. The DEBUGSS also provides a mailbox system for communicating with software through SWD.

Key features provided by the debug subsystem include:

- Two-wire (SWDIO, SWCLK) debug interface, compatible with both TI and 3rd party debug probes
  - On-chip pullup/pulldown resistors for SWDIO and SWCLK, respectively, enabled by default
  - Support for disabling SWD functions to use SWD pins as general-purpose input/output pins
  - Capability of waking the device from shutdown mode upon valid SWD activity
- Debug the processor
  - Run, halt, and step debug support
  - Four hardware breakpoints (BPU)
- Software-configurable peripheral behavior during processor debug
  - Ability to free run select peripherals through debug halt
  - Ability to halt select peripherals on a debug halt
  - Ability to request reset and mode changes to the Power Management Control (PMCTL)
- Debug subsystem mailbox (DSSM) for passing data and control signals between the SWD interface and boot ROM (as well as application software)
- Support for various security features, including software password authenticated debugging

5.2 Block Diagram

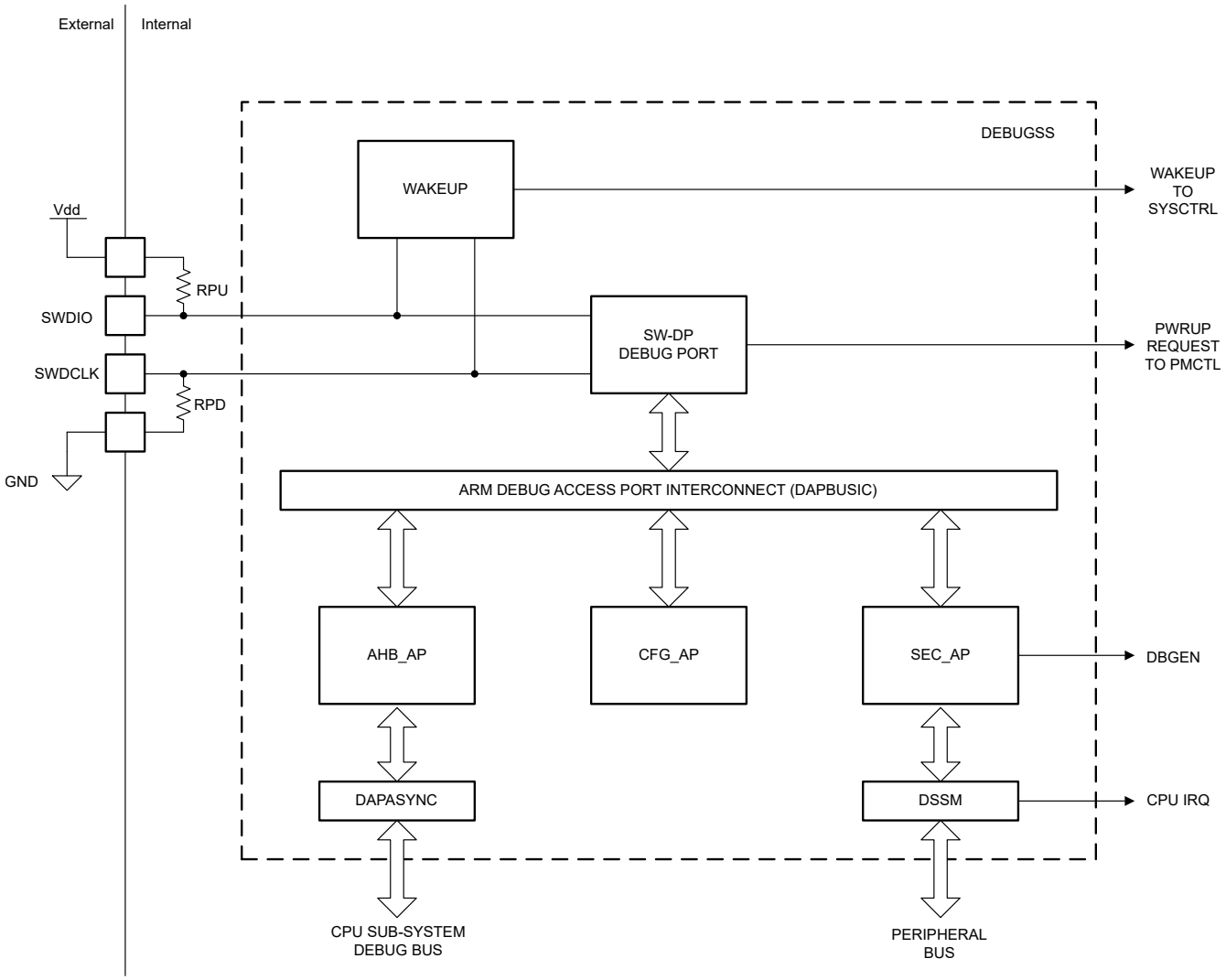


Figure 5-1. DEBUGSS Block Diagram

5.3 Overview

The SWD physical interface interacts with the Arm serial wire debug port (SW-DP) to gain access to the debug access port bus interconnect (DAPBUSIC) when the SW-DP is enabled. From TI, devices ship with the SW-DP enabled to allow SWD access to the device for development and production programming. The SW-DP can be configured to be permanently disabled through the boot configuration policy, see "debugCfg" in Section 9.2. The DAPBUSIC enables a debug probe to access one or more debug access ports. For a debug probe to be able to communicate with an access port, the SW-DP debug port must not be disabled by the boot configuration policy, and the target access port must also not be disabled by the boot configuration policy. The available access ports are given in Section 5.3.2.

5.3.1 Physical Interface

Debug connections to the device are supported through an Arm serial wire debug (SWD) compliant interface. The SWD interface requires two connections:

- A bidirectional data line (SWDIO) used to send data to and receive data from the device
- A unidirectional clock line (SWCLK) driven by the debug probe connecting to the device

The SWD interface uses the standard logic levels of the device for SWD communication. See the device-specific data sheet for input and output logic levels for a given supply voltage (VDDS). A SWCLK frequency of up to 10MHz is supported by the DEBUGSS.

During SWD operation, the SWDIO line can be driven high or driven low by either the target device or the debug probe. As either device can drive the line, when ownership of the shared SWDIO line is switched between the device and the debug probe, undriven time slots are inserted as a part of the SWD protocol. The primary purpose of the pullup resistor on the SWDIO line, and the pulldown resistor on the SWCLK line, is to place the SWD pins into a known state when no debug probe is attached. A minimum resistance of 100kΩ is recommended by Arm. The internal pullup/pulldown resistors fulfill this requirement and external resistors are not required for correct operation of the SWD interface.

After a power-on reset (POR), the target device configures the SWD pins in SWD mode with an internal pullup resistor enabled on the SWDIO line and an internal pulldown resistor enabled on the SWCLK line. If the device configuration has not permanently disabled all SWD access, then the SWD interface is enabled during the boot process and a debug probe can be connected to the DEBUGSS.

In the event that a device was configured by software to enter shutdown mode, and a debug probe is then connected to the SWD pins with SWCLK active, wake-up logic triggers an exit from shutdown mode. A debug connection can then be established to the DEBUGSS after the reset sequence completes.

Upon physical connection of a debug probe, a configuration sequence must be sent from the debug probe to the target device to initiate a valid SWD connection with the SW-DP. An invalid sequence doesn't wake the device from shutdown mode. Once the sequence is transmitted and the SWD connection is established, communication with enabled debug access points is possible and the boot code is alerted by asserting the DBGSS.DBGCTL[1] SWDSEL bit, which is continuously monitored in the boot code. The debug probe must be disconnected by sending a disconnection sequence from the debug probe to the target device.

Bootcode can disable the SWD interface in DEBUGSS, freeing the IOs to be used for general-purpose IO functionality. Once the boot code disables SWD functionality, SWD functionality cannot be re-enabled other than by triggering a POR. A POR automatically re-enables the SWD functionality and puts the SWD pins into SWD mode with pullup/pulldown resistors enabled. To regain debug access to a device, hold the device in a reset state with the RSTN pin during a POR. This prevents the boot code from starting and lets the debug probe gain access to the device.

5.3.2 Debug Access Ports

The debug access ports in the DEBUGSS are given in Table 5-1.

Table 5-1. DEBUGSS Access Port Listing

APSEL	AP	Port Description	Purpose
0x0	AHB-AP	CPUSS debug access port	Debug of the processor and peripherals
0x1	CFG-AP	Configuration access port	Access device type information
0x2	SEC-AP	Security access port	Access the debug mailbox

The AHB-AP provides the complete device debug functionality (processor debug, peripheral and memory bus access and processor state). See Section 5.4 for more information.

The CFG-AP provides device information to the debug probe so that the debug probe can identify device characteristics, including the device part number and the device revision.

The SEC-AP provides access to the mailbox for communicating with software running on the device through SWD. See Section 5.7 for more information.

5.4 Debug Features

The DEBUGSS supports processor debug and peripheral debug.

5.4.1 Processor Debug

The Arm Cortex-M0+ processor supports a wide range of features to simplify debugging of application software during development. Key features supported by CC23xx MCUs include:

- Ability to halt the processor through an assertion of a halt signal, a configured debug event (such as a hard fault entry or reset), or a BKPT instruction (for software breakpoints)
- Ability to step through instructions (with or without peripheral interrupts enabled)
- Ability to run through instructions (with or without peripheral interrupts enabled)
- Ability to read and write CPU registers when halted
- Ability to read exception information through the system control space (SCS)
- Support for four hardware breakpoints
- Support for accessing the device memory map

5.4.2 Breakpoint Unit (BPU)

The breakpoint unit (BPU) provides four comparators that can be used to generate a debug event when the address of an instruction fetch matches the address programmed into the one of the BPU comparators.

The BPU comparators match instruction fetches from the code memory region, meaning the comparators only operate on instruction read accesses. The comparators do not match data read or data write accesses.

Address matching is possible for half-word (16-bit) instructions and word (32-bit) instructions fetched from the CODE region (0x00000000 to 0x1FFFFFFF).

Hardware breakpoints are not available when debugging code in SRAM. When debugging code in SRAM, software breakpoints must be inserted by the debug probe.

5.4.3 Peripheral Debug

In addition to processor debug, the DEBUGSS can be used to access the device memory map from the perspective of the processor. Thus, a connected debug probe can be used to read and write memory-mapped peripheral registers, the system SRAM, and the flash memory.

Certain peripherals support advanced debug configuration options. These options are configured by application software (or optionally, the debug probe) by setting/clearing various debug control bits in the memory map of a given peripheral. In general, the debug behavior of a particular peripheral is specified in the EMU register of each peripheral. Many peripherals offer the option of halting the functional clock to the peripheral when the processor is halted for debug, thus pausing the peripheral together with the processor (default configuration), or letting the peripheral run even when the processor is halted for debug.

For example, the SYSTIM peripheral supports the RUN/STOP bits in the EMU register. Setting the RUN bit in EMU for a SYSTIM causes the SYSTIM to run even if the processor is halted for debug.

5.5 Behavior in Low Power Modes

The DEBUGSS supports maintaining a debug connection through SWD in all operating modes except shutdown.

Access to device memory and peripherals is possible in active mode and idle mode, in which a debug probe can be actively connected to the AHB-AP access port to interface with the processor. In standby mode, a debug connection can be established or maintained with the DEBUGSS, but not with the CPU debug access port.

While a debug connection to the DEBUGSS is not possible while the device is in shutdown mode, a debug probe can cause the device to exit shutdown mode by attempting to communicate with the SWD pins. The device detects attempted SWD communication even when the device is in shutdown. If any activity is detected, a shutdown exit is initiated and after which a debug connection can be made to the DEBUGSS through SWD. An active debug connection prohibits shutdown entry and the device can enter shutdown mode only after debug disconnection.

The DEBUGSS functionality by operating mode is given in Table 5-2.

Table 5-2. DEBUGSS Functionality by Operating Mode

Capability	Active	Idle	Standby	Shutdown
Processor debug	Y	Y	N	N
Memory map access	Y	Y	N	N
Debug status through SW-DP	Y	Y	Y	N
Debug state maintained	Y	Y	Y	N
Wake from SWD	-	-	-	Y

5.6 Restricting Debug Access

The debug subsystem supports several methods for restricting access to the device through the SWD interface. The debug access policy is determined by the user configuration specified in the CCFG flash region. See Section 9.2 for a detailed description of debug access control through the CCFG flash region.

There are three levels of access control, given in Table 5-3. By default, products shipped from TI arrive in a "debug enabled" state where the device is fully open. This state is not recommended for production. For production, TI recommends changing the debug configuration to password-protected or disabled.

Table 5-3. Debug Access Control

DEBUGSS Function	Debug Configuration		
	Debug Enabled (default)	Debug Enabled with Password	Debug Disabled
SW-DP(debug port)	EN	EN	DIS
CFG-AP	EN	EN	DIS
SEC-AP	EN	EN	DIS
AHB-AP (CPU Debug)	EN	EN w/ PW	DIS

When debug is enabled with a password, the debug access command together with the user-specified debug access password must be provided to the DEBUGSS mailbox by the debug probe.

When debug is disabled, the SW-DP is disabled during the boot process. Any commands previously sent to the mailbox are ignored during boot. Following boot, any attempt to connect to the SW-DP is ignored.

Debug access can be permanently locked by configuring the CCFG flash region to disable debug access while also configuring the CCFG flash region as statically write-protected (locked). Locking the CCFG configuration has the added security of preventing the bootstrap loader (BSL) and application code from changing the debug security policy.

5.7 Mailbox (DSSM)

The debug subsystem mailbox (DSSM) enables a debug probe to pass messages to the target device through the SWD interface, and for the target device to return data to the debug probe.

The DSSM supports the following functions:

- Transmission of commands to the device during boot, including authenticating the debug probe for password-protected debug, mass erase, and factory reset operations
- Communicating with application software running on the target device when no other communication interface is present

32-bit word data buffers are provided for transmit data (debug probe to target device) and receive data (target device to debug probe). These data buffers are implemented as the 32-bit memory-mapped registers DBGSS.TXD and DBGSS.RXD. The DBGSS.TXCTL and DBGSS.RXCTL registers are provided for enabling flow control and indicating status of the mailbox.

Table 5-4. DBGSS DSSM Register Functions

DBGSS Register	Description	Debug Probe	Target Device	Actions
TXD	Data buffer	RW	R	DBGSS.TXCTL[0] TXDSTA bit is set on write by the debug probe, and cleared on a read by the target device. DBGSS.RIS[0] TXIFG bit is also set on a write by the debug probe.
TXCTL	Flow control and status	RW	R	None
RXD	Data buffer	R	RW	DBGSS.RXCTL[0] RXDSTA bit is set on write by the target device, and cleared on a read by the debug probe. DBGSS.RIS[1] RXIFG bit is also set on a write by the target device
RXCTL	Flow control and status	R	RW	None

The DBGSS.TXCTL[0] TRANSMIT bit is set in the TXCTL register when a debug probe writes data to the DBGSS.TXD buffer register. The TRANSMIT flag remains set until the target device reads DBGSS.TXD or a POR occurs. The DBGSS.RXCTL[0] RECEIVE bit is set when the target device writes data to the DBGSS.RXD buffer register. The RECEIVE flag remains set until the debug probe or target device reads the data from DBGSS.RXD.

Software running on the target device cannot write to TXD, and software cannot clear the TRANSMIT flag other than by reading DGBSS.TXD. The DBGSS.TXCTL[7:1] bit field contains flag bits which can be set or cleared by the debug probe to implement a protocol if desired. Only the debug probe can write to the TRANSMIT_FLAGS field in TXCTL.

In a similar way, only the target device software can write to DBGSS.RXD and DBGSS.RXCTL. The debug probe cannot write to RX_DATA and can only clear the DEBUGSS.RXCTL[0] RECEIVE bit by reading DBGSS.RXD. DBGSS.RXCTL[7:1] FLAGS bit field contains flag bits. Software on the target device can set or clear bits in the DBGSS.RXCTL[7:1] FLAGS field to implement a protocol if desired. These flags can be read by the debug probe but can not be modified by the debug probe.

DBGSS.TXDPEEK and DBGSS.RXDPEEK registers can be read by the device to read DBGSS.TXD and DBGSS.RXD registers without affecting the FULL/EMPTY flag.

For a complete listing of commands that are supported by the boot configuration routine during device startup configuration, see Chapter 8.

5.8 Mailbox Events

The DSSM contains one event publisher and no event subscribers. One event publisher (INT_EVENT0) manages DSSM interrupt requests (IRQs) to the CPU subsystem through the AON event fabric.

The DSSM events are summarized in Table 5-5.

Table 5-5. DSSM Events

Event	Type	Source	Destination	Route	Configuration	Functionality
CPU Interrupt Event	Publisher	DEBUGSS	CPU Subsystem	Dynamic route	AON_DBG_COMB	DBGIRQ from DEBUGSS to CPU is a configurable IRQ

5.8.1 CPU Interrupt Event (AON_DBG_COMB)

The DSSM provides 4 interrupt sources which can be configured to source a CPU interrupt event. The CPU interrupt events from the DSSM are given in Table 5-6.

Table 5-6. DSSM CPU Interrupt Event Conditions (INT_EVENT0)

Name	Description
TXIFG	Indicates that the TX_DATA buffer in the DSSM has received data.
RXIFG	Indicates that the data in RX_DATA buffer in the DSSM was read.
PWRUPIFG	Indicates that the DEBUGSS was started due to a debug probe attaching to the device.

Table 5-6. DSSM CPU Interrupt Event Conditions (INT_EVENT0) (continued)

Name	Description
PWRDWNIFG	Indicates that the DEBUGSS was stopped due to a debug probe disconnecting from the device.

See Chapter 4 for guidance on configuring the DEBUGGSS event as a CPU interrupt.

5.9 Software Considerations

Enable the debug connection before connecting a debug probe by setting the DBGSS.DBGBCTRL[5] SWDCEN register bit and disable it after the debug session is completed by clearing the bit.

The debug software override feature is a failsafe approach for enabling a debug connection and is enabled by writing the DBGSS.DBGBCTRL[0] SWDOVR register bit. When this bit is set, the device does not wake up from shutdown due to a debug connection. This feature is used only for internal debugging purposes if there are issues seen in establishing a debug connection in standby/active modes.

5.10 DBGSS Registers

Table 5-7 lists the memory-mapped registers for the DBGSS registers. All register offset addresses not listed in Table 5-7 should be considered as reserved locations and the register contents should not be modified.

Table 5-7. DBGSS Registers

Offset	Acronym	Register Name	Section
0h	DESC	Module Description	Go
44h	IMASK	Interrupt mask	Go
4Ch	RIS	Raw interrupt status	Go
54h	MIS	Masked interrupt status	Go
5Ch	ISET	Interrupt set	Go
64h	ICLR	Interrupt clear	Go
6Ch	IMSET	Set Interupt Mask in IMASK	Go
74h	IMCLR	Clear Interupt Mask in IMASK	Go
100h	TXD	Transmit data register	Go
104h	TXCTL	Transmit control register	Go
108h	RXD	Receive data register	Go
10Ch	RXCTL	Receive control register	Go
110h	TXDPEEK	Transmit Data Peek Register	Go
114h	RXDPEEK	Receive Data Peek Register	Go
200h	SPECIAL_AUTH	Special enable authorization register	Go
204h	SPECIAL_AUTH_SET	Special enable authorization set register	Go
208h	SPECIAL_AUTH_CLR	Special enable authorization clear register	Go
210h	APP_AUTH	Application authorization register	Go
214h	APP_AUTH_SET	Application authorization set register	Go
218h	APP_AUTH_CLR	Application authorization clear register	Go
21Ch	DBGCTL	Debug control register	Go

Complex bit access types are encoded to fit into small table cells. Table 5-8 shows the codes that are used for access types in this section.

Table 5-8. DBGSS Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

5.10.1 DESC Register (Offset = 0h) [Reset = B24D1010h]

DESC is shown in Table 5-9.

Return to the Summary Table.

Description Register. This register provides IP module ID, revision information, instance index and standard MMR registers offset.

Table 5-9. DESC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	MODULEID	R/W	B24Dh	Module identifier used to uniquely identify this IP. 0h = Minimum value FFFFh = Maximum possible value
15-12	STDIPOFF	R/W	1h	Standard IP MMR block offset. Standard IP MMRs are the set of from aggregated IRQ registers till DTB. 0: Standard IP MMRs do not exist 0x1-0xF: Standard IP MMRs begin at offset of (64*STDIPOFF from the base IP address) 0: STDIP MMRs do not exist 0x1-0xF: These MMRs begin at offset 64*STDIPOFF from IP base address 0h = Minimum Value Fh = Maximum possible value
11-8	INSTIDX	R/W	0h	IP Instance ID number. If multiple instances of IP exist in the device, this field can identify the instance number (0-15). 0h = Minimum Value Fh = Maximum possible value
7-4	MAJREV	R/W	1h	Major revision of IP (0-15). 0h = Minimum Value Fh = Maximum possible value
3-0	MINREV	R/W	0h	Minor revision of IP (0-15). 0h = Minimum Value Fh = Maximum possible value

5.10.2 IMASK Register (Offset = 44h) [Reset = 00000000h]

IMASK is shown in Table 5-10.

Return to the Summary Table.

Interrupt mask. This register selects interrupt sources which are allowed to pass from RIS to MIS when the corresponding bit-fields are set to 1.

Table 5-10. IMASK Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3	PWRDWNIFG	R/W	0h	PWRDWNIFG interrupt mask 0h = Interrupt is masked out 1h = Interrupt will request an interrupt service routine and corresponding bit in MIS will be set
2	PWRUPIFG	R/W	0h	PWRUPIFG interrupt mask 0h = Interrupt is masked out 1h = Interrupt will request an interrupt service routine and corresponding bit in MIS will be set
1	RXIFG	R/W	0h	RXIFG interrupt mask 0h = Interrupt is masked out 1h = Interrupt will request an interrupt service routine and corresponding bit in MIS will be set
0	TXIFG	R/W	0h	TXIFG interrupt mask 0h = Disable Interrupt Mask 1h = Enable Interrupt Mask

5.10.3 RIS Register (Offset = 4Ch) [Reset = 00000000h]

RIS is shown in Table 5-11.

Return to the Summary Table.

Raw interrupt status. This register reflects the state of all pending interrupts, regardless of masking. This register allows the user to implement a poll scheme. A flag set in this register can be cleared by writing 1 to the corresponding ICLR register bit.

Table 5-11. RIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3	PWRDWNIFG	R/W	0h	Raw interrupt status for PWRDWNIFG 0h = PWRDWNIFG did not occur 1h = PWRDWNIFG occurred
2	PWRUPIFG	R/W	0h	Raw interrupt status for PWRUPIFG 0h = PWRUPIFG did not occur 1h = PWRUPIFG occurred
1	RXIFG	R/W	0h	Raw interrupt status for RXIFG 0h = RXIFG did not occur 1h = RXIFG occurred
0	TXIFG	R/W	0h	Raw interrupt status for TXIFG 0h = TXIFG did not occur 1h = TXIFG occurred

5.10.4 MIS Register (Offset = 54h) [Reset = 00000000h]

MIS is shown in Table 5-12.

Return to the Summary Table.

Masked interrupt status. This register is simply a bitwise AND of the contents of IMASK and RIS registers. A flag set in this register can be cleared by writing 1 to the corresponding ICLR register bit.

Table 5-12. MIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3	PWRDWNIFG	R/W	0h	Masked interrupt status for PWRDWNIFG 0h = PWRDWNIFG did not request an interrupt service routine 1h = PWRDWNIFG requests an interrupt service routine
2	PWRUPIFG	R/W	0h	Masked interrupt status for PWRUPIFG 0h = PWRUPIFG did not request an interrupt service routine 1h = PWRUPIFG requests an interrupt service routine
1	RXIFG	R/W	0h	Masked interrupt status for RXIFG 0h = RXIFG did not request an interrupt service routine 1h = RXIFG requests an interrupt service routine
0	TXIFG	R/W	0h	Masked interrupt status for TXIFG 0h = TXIFG did not request an interrupt service routine 1h = TXIFG requests an interrupt service routine

5.10.5 ISET Register (Offset = 5Ch) [Reset = 00000000h]

ISET is shown in Table 5-13.

Return to the Summary Table.

Interrupt set register. This register can be used by software for diagnostics and safety checking purposes. Writing a 1 to a bit in this register will set the event and the corresponding RIS bit also gets set. If the corresponding IMASK bit is set, then the corresponding MIS register bit also gets set.

Table 5-13. ISET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3	PWRDWNIFG	R/W	0h	Sets PWRDWNIFG in RIS register 0h = Writing a 0 has no effect 1h = Set interrupt
2	PWRUPIFG	R/W	0h	Sets PWRUPIFG in RIS register 0h = Writing a 0 has no effect 1h = Set interrupt
1	RXIFG	R/W	0h	Sets RXIFG in RIS register 0h = Writing a 0 has no effect 1h = Set interrupt
0	TXIFG	R/W	0h	Sets TXIFG in RIS register 0h = Writing a 0 has no effect 1h = Set interrupt

5.10.6 ICLR Register (Offset = 64h) [Reset = 00000000h]

ICLR is shown in Table 5-14.

Return to the Summary Table.

Interrupt clear register. This register allows software to clear interrupts. Writing a 1 to a bit in this register will clear the event and the corresponding RIS bit also gets cleared. If the corresponding IMASK bit is set, then the corresponding MIS register bit also gets cleared.

Table 5-14. ICLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3	PWRDWNIFG	R/W	0h	Clears PWRDWNIFG interrupt 0h = Writing a 0 has no effect 1h = Clear interrupt
2	PWRUPIFG	R/W	0h	Clears PWRUPIFG interrupt 0h = Writing a 0 has no effect 1h = Clear interrupt
1	RXIFG	R/W	0h	Clears RXIFG interrupt 0h = Writing a 0 has no effect 1h = Clear interrupt
0	TXIFG	R/W	0h	Clears TXIFG interrupt 0h = Writing a 0 has no effect 1h = Clear interrupt

5.10.7 IMSET Register (Offset = 6Ch) [Reset = 00000000h]

IMSET is shown in Table 5-15.

Return to the Summary Table.

Interrupt mask set register. Writing a 1 to a bit in this register will set the corresponding IMASK bit.

Table 5-15. IMSET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3	PWRDWNIFG	R/W	0h	Set PWRDWNIFG interrupt mask 0h = Writing a 0 has no effect 1h = Set interrupt mask
2	PWRUPIFG	R/W	0h	Set PWRUPIFG interrupt mask 0h = Writing a 0 has no effect 1h = Set interrupt mask
1	RXIFG	R/W	0h	Set RXIFG interrupt mask 0h = Writing a 0 has no effect 1h = Set interrupt mask
0	TXIFG	R/W	0h	Set TXIFG interrupt mask 0h = Writing a 0 has no effect 1h = Set interrupt mask

5.10.8 IMCLR Register (Offset = 74h) [Reset = 00000000h]

IMCLR is shown in Table 5-16.

Return to the Summary Table.

Interrupt mask clear register. Writing a 1 to a bit in this register will clear the corresponding IMASK bit.

Table 5-16. IMCLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3	PWRDWNIFG	R/W	0h	Clears PWRDWNIFG interrupt mask 0h = Writing a 0 has no effect 1h = IMASK bit corresponding to PWRDWNIFG is cleared
2	PWRUPIFG	R/W	0h	Clears PWRUPIFG interrupt mask 0h = Writing a 0 has no effect 1h = IMASK bit corresponding to PWRUPIFG is cleared
1	RXIFG	R/W	0h	Clears RXIFG interrupt mask 0h = Writing a 0 has no effect 1h = IMASK bit corresponding to RXIFG is cleared
0	TXIFG	R/W	0h	Clears TXIFG interrupt mask 0h = Writing a 0 has no effect 1h = IMASK bit corresponding to TXIFG is cleared

5.10.9 TXD Register (Offset = 100h) [Reset = 00000000h]

TXD is shown in Table 5-17.

Return to the Summary Table.

Transmit data register. This register is used for sending SACI (SECAP command interface) data from the host to the device.
The host (SWD interface) can write this register. This updates the value of TXD, and sets TXCTL.TXDSTA = FULL
The host should only write TXD while TXCTL.TXDSTA = EMPTY.
If the host incorrectly writes TXD while TXCTL.TXDSTA = FULL, this will just update the value of TXD.
The host (SWD interface) can read the TXD register. This does not affect TXCTL.TXDSTA.
The device (boot code) can only read the TXD register. This sets TXCTL.TXDSTA = EMPTY.
The device should only read TXD while TXCTL.TXDSTA = FULL.
If the device incorrectly reads TXD while TXCTL.TXDSTA = EMPTY, this will just return the value of TXD.
If the host writes TXD on the same clock cycle as the device reads TXD:
The device reads the old TXD value.
TXD is updated with the new value, and TXCTL.TXDSTA is set to FULL.

Table 5-17. TXD Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R	0h	SACI command/parameter word. Valid value when TXCTL.TXDSTA=1. TXCTL.TXDSTA gets automatically cleared upon read.

5.10.10 TXCTL Register (Offset = 104h) [Reset = 00000000h]

TXCTL is shown in Table 5-18.

Return to the Summary Table.

Transmit control register. This register contains status of the TXD register (full/empty), and also software defined flags that are used by the SACI protocol.
The host (SWD interface) can write the FLAGS field of the TXCTL register.
The host (SWD interface) can read the TXCTL register.
The device (boot code) can only read the TXCTL register.

Table 5-18. TXCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-1	FLAGS	R	0h	Software defined flags that are used by the SACI protocol (host to device).
0	TXDSTA	R	0h	Indicates whether the host has written a word to the TXD register, which can be read by the device: TXDSTA is automatically set upon write to TXD register in SECAP and automatically gets cleared upon read from TXD 0h = The TXD register does not contain a new SACI parameter word from the host, and should not be read by the device. 1h = The TXD register contains a new SACI parameter word from the host, which can be read by the device.

5.10.11 RXD Register (Offset = 108h) [Reset = 00000000h]

RXD is shown in Table 5-19.

Return to the Summary Table.

Receive data register. This register is used to send SACI command response data from the device to the host.
The device (boot code) can write the RXD register. This updates the value of RXD, and sets RXCTL.RXDSTA = FULL.
The device should only write RXD while RXCTL.RXDSTA = EMPTY.
If the device incorrectly writes RXD while RXCTL.RXDSTA = FULL, this will just update the value of RXD.
The device (boot code) can read the RXD register in order to flush it. This sets RXCTL.RXDSTA = EMPTY.
The host (SWD interface) can only read the RXD register. This sets RXCTL.RXDSTA = EMPTY.
The host should only read RXD while RXCTL.RXDSTA = FULL.
If the host incorrectly reads RXD while RXCTL.RXDSTA = EMPTY, this will just return the value of RXD.
If the device writes RXD on the same clock cycle as the host reads RXD:
The host reads the old RXD value.
RXD is updated with the new value, and RXCTL.RXDSTA is set to FULL.

Table 5-19. RXD Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R/W	0h	SACI command response word. RXCTL.RXDSTA automatically set upon write. RXCTL.RXDSTA automatically cleared upon read (flush operation).

5.10.12 RXCTL Register (Offset = 10Ch) [Reset = 00000000h]

RXCTL is shown in Table 5-20.

Return to the Summary Table.

Receive control register. This register contains status of the RXD register (full/empty), and also software defined flags that are used by the SACI protocol.
The device (boot code) can write the FLAGS field of the RXCTL register.
The device (boot code) can read the RXCTL register.
The host (SWD interface) can only read the RXCTL register

Table 5-20. RXCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-1	FLAGS	R/W	0h	Software defined flags that are used by the SACI protocol (device to host).
0	RXDSTA	R	0h	Indicates whether the device has written a word to the RXD register, which can be read by the host: RXDSTA is automatically set upon write to RXD and automatically cleared upon read from RXD register of SECAP or RXD. 0h = The RXD register does not contain a new SACI response word from the device, and should not be read by the host. 1h = The RXD register contains a new SACI response word from the device, which can be read by the host.

5.10.13 TXDPEEK Register (Offset = 110h) [Reset = 00000000h]

TXDPEEK is shown in Table 5-21.

Return to the Summary Table.

Transmit data peek register . This register is a read-only version of the TXD register that can be read by host and device without any side-effects.
This register is used to peek at the values in TXD without affecting the FULL/EMPTY flag.

Table 5-21. TXDPEEK Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R	0h	Transmit Data Peek Register. SACI command parameter word. TXCTL.TXDSTA not affected by read of TXDPEEK

5.10.14 RXDPEEK Register (Offset = 114h) [Reset = 00000000h]

RXDPEEK is shown in Table 5-22.

Return to the Summary Table.

Receive data peek register. The RXDPEEK register is a read-only version of the RXD register that can be read by host and device without any side-effects
This register is used to peek at the values in Receive Data Register without affecting the FULL/EMPTY flag.

Table 5-22. RXDPEEK Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R	0h	Receive Data Peek Register. SACI command response word. RXCTL.RXDSTA not affected by read of RXDPEEK

5.10.15 SPECIAL_AUTH Register (Offset = 200h) [Reset = 00000013h]

SPECIAL_AUTH is shown in Table 5-23.

Return to the Summary Table.

This register indicates the status of different AP firewalls.

Table 5-23. SPECIAL_AUTH Register Field Descriptions

Bit	Field	Type	Reset	Description
31-7	RESERVED	R	0h	Reserved
6	DBGDIS	R	0h	Indicates status of DBGDIS. 0h = Enables debugging capability. 1h = Disables debugging capability
5	AHBAPEN	R	0h	Indicates status of AHBAPEN 0h = Disable AHB-AP 1h = Enable AHB-AP
4	CFGAPEN	R	1h	Indicates status of CFGAPEN 0h = Disable CFG-AP 1h = Enable CFG-AP
3	RESERVED	R	0h	Reserved
2	DFTAPEN	R	0h	Indicates status of DFTAPEN 0h = Disable DFT-TAP 1h = Enable DFT-TAP
1	RESERVED	R	1h	Reserved
0	SECAPEN	R	1h	Indicates status of SECAP 0h = Disable SEC-AP 1h = Enable SEC-AP

5.10.16 SPECIAL_AUTH_SET Register (Offset = 204h) [Reset = 00000013h]

SPECIAL_AUTH_SET is shown in Table 5-24.

Return to the Summary Table.

This register is used for setting bits in SPECIAL_AUTH register.
This register is configured and locked during device boot.

Table 5-24. SPECIAL_AUTH_SET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	KEY	W	0h	This field must be configured with 0xA5 in order to access this register. A5h = This field must be written with 0xA5 to be able to set any of the enable bits
23-7	RESERVED	R	0h	Reserved
6	DBGDIS	W	0h	This bit sets DBGDIS in SPECIAL_AUTH register. 0h = Writing 0 has no effect 1h = SET DBGDIS
5	AHBAPEN	W	0h	This bit sets AHBAPEN in SPECIAL_AUTH register. 0h = Writing 0 has no effect 1h = SET AHB-AP
4	CFGAPEN	W	1h	This bit sets CFGAPEN in SPECIAL_AUTH register. 0h = Writing 0 has no effect 1h = Set CFGAPEN
3	RESERVED	W	0h	Reserved
2	DFTAPEN	W	0h	This bit sets DFTAPEN in SPECIAL_AUTH register. 0h = Writing 0 has no effect 1h = Set DFTAPEN
1	RESERVED	W	1h	Reserved
0	SECAPEN	W	1h	This bit sets SECAPEN bit in SPECIAL_AUTH register. 0h = Writing 0 has no effect 1h = Set SECAPEN

5.10.17 SPECIAL_AUTH_CLR Register (Offset = 208h) [Reset = 00000000h]

SPECIAL_AUTH_CLR is shown in Table 5-25.

Return to the Summary Table.

This register is used for clearing bits in SPECIAL_AUTH register.
This register is configured and locked during device boot.

Table 5-25. SPECIAL_AUTH_CLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	KEY	W	0h	This field must be configured with 0x22 in order to access this register. 22h = This field must be written with 0x22 to be able to clear any of the enable bits
23-7	RESERVED	R	0h	Reserved
6	DBGDIS	W	0h	This bit clears DBGDIS in SPECIAL_AUTH register. 0h = Writing 0 has no effect 1h = Clear DBGDIS
5	AHBAPEN	W	0h	This bit clears AHBAPEN in SPECIAL_AUTH register. 0h = Writing 0 has no effect 1h = Clear AHBAPEN
4	CFGAPEN	W	0h	This bit clears CFGAPEN in SPECIAL_AUTH register. 0h = Writing 0 has no effect 1h = Clear CFGAPEN
3	RESERVED	W	0h	Reserved
2	DFTAPEN	W	0h	This bit clears DFTAPEN in SPECIAL_AUTH register. 0h = Writing 0 has no effect 1h = Clear DFTAPEN
1	RESERVED	W	0h	Reserved
0	SECAPEN	W	0h	This bit clears SECAPEN in SPECIAL_AUTH register. 0h = Writing 0 has no effect 1h = Clear SECAPEN

5.10.18 APP_AUTH Register (Offset = 210h) [Reset = 00000000h]

APP_AUTH is shown in Table 5-26.

Return to the Summary Table.

This register indicates the debug privileges of ARM Cortex CPU.

Table 5-26. APP_AUTH Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1	NIDEN	R	0h	Controls non-invasive debug enable. 0h = Non-invasive debug disabled 1h = Non-invasive debug enabled
0	DBGEN	R	0h	Controls invasive debug enable. 0h = Invasive debug disabled 1h = Invasive debug enabled

5.10.19 APP_AUTH_SET Register (Offset = 214h) [Reset = 00000000h]

APP_AUTH_SET is shown in Table 5-27.

Return to the Summary Table.

This register is used for setting bits in APP_AUTH register.
This register is configured and locked during device boot.

Table 5-27. APP_AUTH_SET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	KEY	W	0h	This field must be configured with 0x39 in order to access this register. 39h = Write this value 0x39 to unlock writing to the APP_AUTH_SET register
23-2	RESERVED	R/W	0h	Reserved
1	NIDEN	W	0h	Sets NIDEN bit in [APP_AUTH]register. 0h = Writing 0 has no effect 1h = Sets NIDEN
0	DBGEN	W	0h	Sets DBGEN bit in APP_AUTH register. 0h = Writing 0 has no effect 1h = Sets DBGEN

5.10.20 APP_AUTH_CLR Register (Offset = 218h) [Reset = 00000000h]

APP_AUTH_CLR is shown in Table 5-28.

Return to the Summary Table.

This register is used for clearing bits in APP_AUTH register.
This register is configured and locked during device boot.

Table 5-28. APP_AUTH_CLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	KEY	W	0h	This field must be configured with 0x7D in order to access this register. 7Dh = Write this value 0x7D to unlock writing to the APP_AUTH_CLR register
23-2	RESERVED	R/W	0h	Reserved
1	NIDEN	W	0h	Clears NIDEN bit in APP_AUTH register. 0h = Writing 0 has no effect 1h = Clears NIDEN
0	DBGEN	W	0h	Clears DBGEN bit in APP_AUTH register. 0h = Writing 0 has no effect 1h = Clears DBGEN

5.10.21 DBGCTL Register (Offset = 21Ch) [Reset = 00000020h]

DBGCTL is shown in Table 5-29.

Return to the Summary Table.

Debug control register. This register is used for controlling debug connection and read out debug status.

Table 5-29. DBGCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Reserved
5	SWDCEN	R/W	1h	This bit is used to enable connection between SWD pads and IceMelter (wakeup circuit used for detecting debug probe) 0h = Connection disabled 1h = Connection enabled
4	DBGPWRUPACK	R	0h	This bit field specifies the status of dbgpwrupack from pmctl. 0h = dbgpwrupreq is not acknowledged 1h = dbgpwrupreq is acknowledged.
3	SYSPWRUPACK	R	0h	This bit field specify the status of syspwrupack from pmctl. 0h = syspwrupreq is not acknowledged 1h = syspwrupreq is acknowledged
2	JTAGSEL	R	0h	This bit field specifies the status of JTAG MODE for TEST TAP. 0h = TEST TAP disabled 1h = TEST TAP enabled
1	SWDSEL	R	0h	This bit field specifies the status of SWD MODE for connection. 0h = debug connection disabled. 1h = debug connection enabled.
0	SWDOVR	R/W	0h	This bit is used for connecting to IO pads to SWCLK/IO on SW-DP through a software request and establish SWD connection without IceMelter trigger for debug purpose. 0h = Transparent mode in which SWD connection is established via IceMelter Sequence. 1h = Force 1 or debug enable mode in which SWD connection is established bypassing IceMelter sequence

Chapter 6

Power, Reset, and Clocking

This chapter describes the systems related to power, resets, and clocking.

6.1 Introduction

Power and clock management is flexible to facilitate low-power applications. The following sections describe details of the clock and power controls.

The features in this chapter are embedded and optimized in TI's Power Manager. Please see the SDK documentation for more details.

Figure 6-1 shows the hierarchy of power-saving features. Low power consumption and cycling time for a power-saving mode have an inverse relationship. The power-saving mode with the lowest power consumption requires the longest time from initiation to power-saving mode, as well as wake-up time back to active mode. Table 6-1 summarizes the power-saving features.

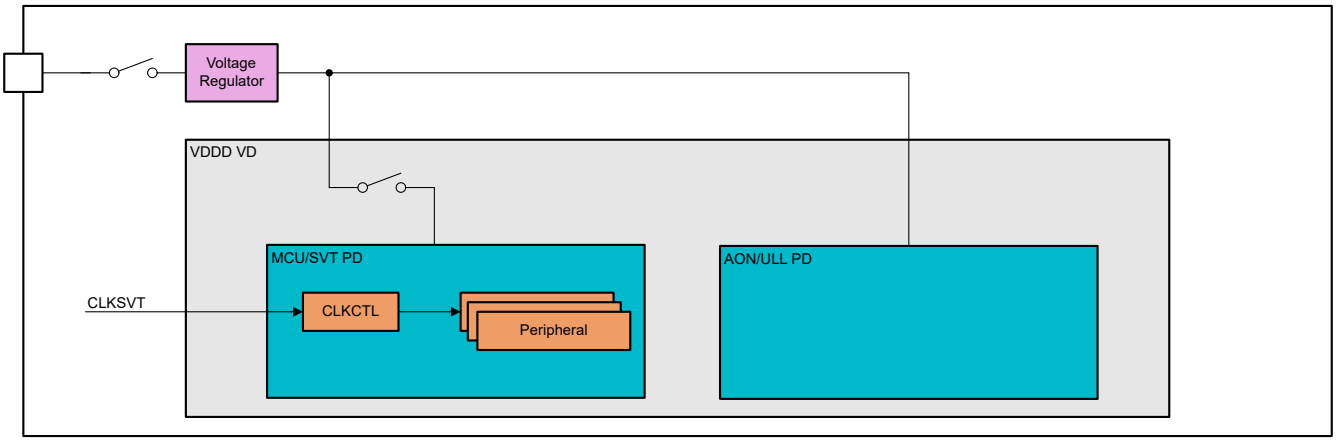


Figure 6-1. Power Hierarchy

Table 6-1. Power Saving Features

Power-Saving Feature	Description
Clock gating	Immediate response—no latency. This feature offers the least amount of power saved.
Power domain off (overrides clock gating)	Power cycling down and up takes longer time than clock gating. Modules in power domains without retention must be reinitialized before functionality can be resumed.
Voltage regulator off	Power cycling down and up takes longer time than power domain cycling. The device loses all configurations and boots at wake-up. This feature offers the least possible current consumption.

Table 6-2 lists the four defined power modes for the power-saving features in TI's Power Manager listed in Table 6-1. Section 6.4 discusses the power modes in detail.

Table 6-2. Power Modes in the TI Power Manager

Power Mode	Description
Active mode	The system CPU is running. System infrastructure components such as system SRAM memory and system NVM memory are also running. Other MCU IP clocks are SW-controlled through CLKCTL.
Idle mode	The system CPU clock is gated. Other MCU IP clocks are SW-controlled through CLKCTL. System Infrastructure components (Flash, ROM, SRAM, bus) clocks are dependent on MCU status.
Standby mode	MCU power domain is powered off, and the VDDD voltage domain is supplied by the micro LDO.
Shutdown mode	Only I/Os maintain their state. All voltage regulators, voltage domains, and power domains are off.

6.2 System CPU Modes

The CPU modes, Run, Sleep, and Deep Sleep, are managed by the TI Power API and cannot be directly manipulated. Table 6-3 shows the CPU mode in each TI-defined power state.

Table 6-3. System CPU Modes

TI Power State	System CPU Mode
Active Mode	Run Mode
Idle Mode	Sleep Mode
Standby mode	Deep sleep Mode

6.3 Supply System

The supply system of the device is complex and controlled by hardware.

6.3.1 Internal DC/DC Converter and Global LDO

Normally, the VDDS supply pins are powered from a 1.8V to 3.8V supply (for example, batteries), and the VDDR supply pins are powered by the internal DC/DC regulator.

Alternatively, the internal Global LDO (GLDO) can be used instead of the DC/DC regulator, but this increases the current consumption of the device. In this mode, disconnect the DCDC pin and remove the DC/DC regulator inductor. The GLDO is connected internally to VDDR. See the device-specific data sheet for the GLDO decoupling capacitor requirements. More information on the DC/DC capability can be found in Chapter 14.

6.4 Power States

The following power states and power state transitions are used in the CC23xx device.

6.4.1 Reset

The lowest power state of the device, reset is entered unconditionally when the reset pin is held low. In this state no oscillators are running, all voltage regulators are disabled and I/Os have both the input and output buffers disabled. There is no internal mechanism to allow software to enter reset, nor any software mechanism to allow exit from reset. Thermal shutdown functionality (where available), after having been explicitly triggered by software, holds the device in a reset state by pulling the RSTN pin low as long as the temperature is above a threshold. See the device-specific data sheet for thermal threshold values.

6.4.2 Shutdown

The lowest power state that can be deliberately entered by software. In this state no oscillators are running, all voltage regulators are disabled, but I/Os retain state. Level-based wake-up can be configured individually for each I/O pin. The device goes through a full boot when exiting shutdown, but I/Os remain latched and the software is able to discern that the reset cause is I/O wake-up from shutdown. Power-on-reset (POR) is active in shutdown but no other voltage supervision is enabled.

6.4.3 Active

Once the reset pin is deasserted and the minimum supply voltage is supplied, the device enters the active power state. HFOSC and the Global LDO are enabled. Once the digital supply is good, the cold boot sequence is performed, applying trims to analog circuitry (including oscillators and voltage regulators) and memories. The CPU boots into the user application at which point the application can configure and enable the DC/DC, low frequency crystal (LFXT) or low frequency oscillator (LFOSC) or high frequency reference clock (HFXT). For more information on the boot process see Chapter 8.

In the active power state, both MCU and AON power domains are powered. Clock gating is used to minimize power consumption. Clock gating to peripherals/subsystems is controlled manually by the CPU.

6.4.4 Idle

Idle is where the CPU is in sleep but selected peripherals and subsystems (such as the radio) can be active. Infrastructure (Flash, ROM, SRAM, bus) clock gating is possible depending on state of the µDMA and debug subsystem.

In Idle mode, all modules are available and power consumption is highly application dependent.

6.4.5 Standby

Standby is the low power state of the device where LFCLK is running and RTC and Watchdog timer can be active. The MCU domain is powered off, but all logic in the AON power domain remains on and clocked by LFCLK. There are up to 6 wake-up sources for standby to active as described in Section 4.3.4.3. When in standby, the DC/DC or GLDO is duty-cycled to periodically recharge VDDR. On standby exit, SVT is powered up again. MCU modules with retention restore the state the modules had before standby entry. See Figure 6-3 to see which modules have retention. Modules without retention are reset and need to be reconfigured when exiting standby.

6.5 Digital Power Partitioning

The device has a single voltage domain, VDDD. Within VDDD there are two power domains. The always-on AON domain and the switchable MCU domain. Throughout this documentation register mnemonics or descriptions can also reference the ULL domain, this is an alternative term for AON and is equivalent in all ways. Similarly the MCU domain can be described in register mnemonics or descriptions as SVT. See Figure 6-2 and Figure 6-3 for an overview of the voltage and power partitioning.

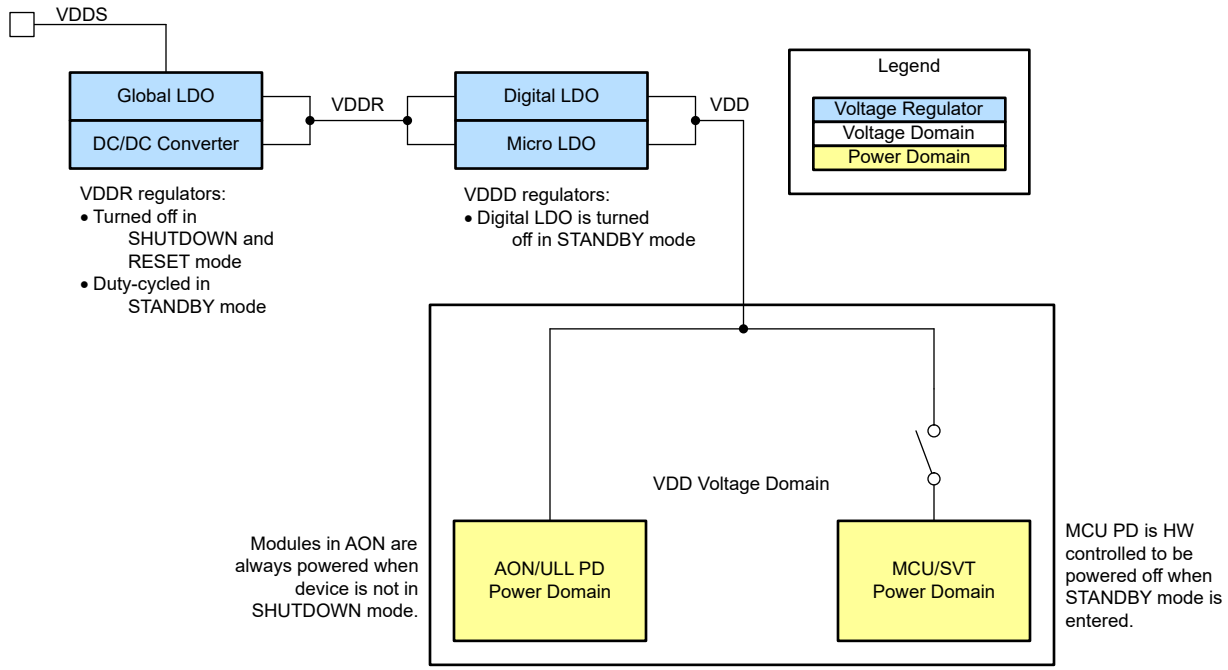


Figure 6-2. Power Supply System

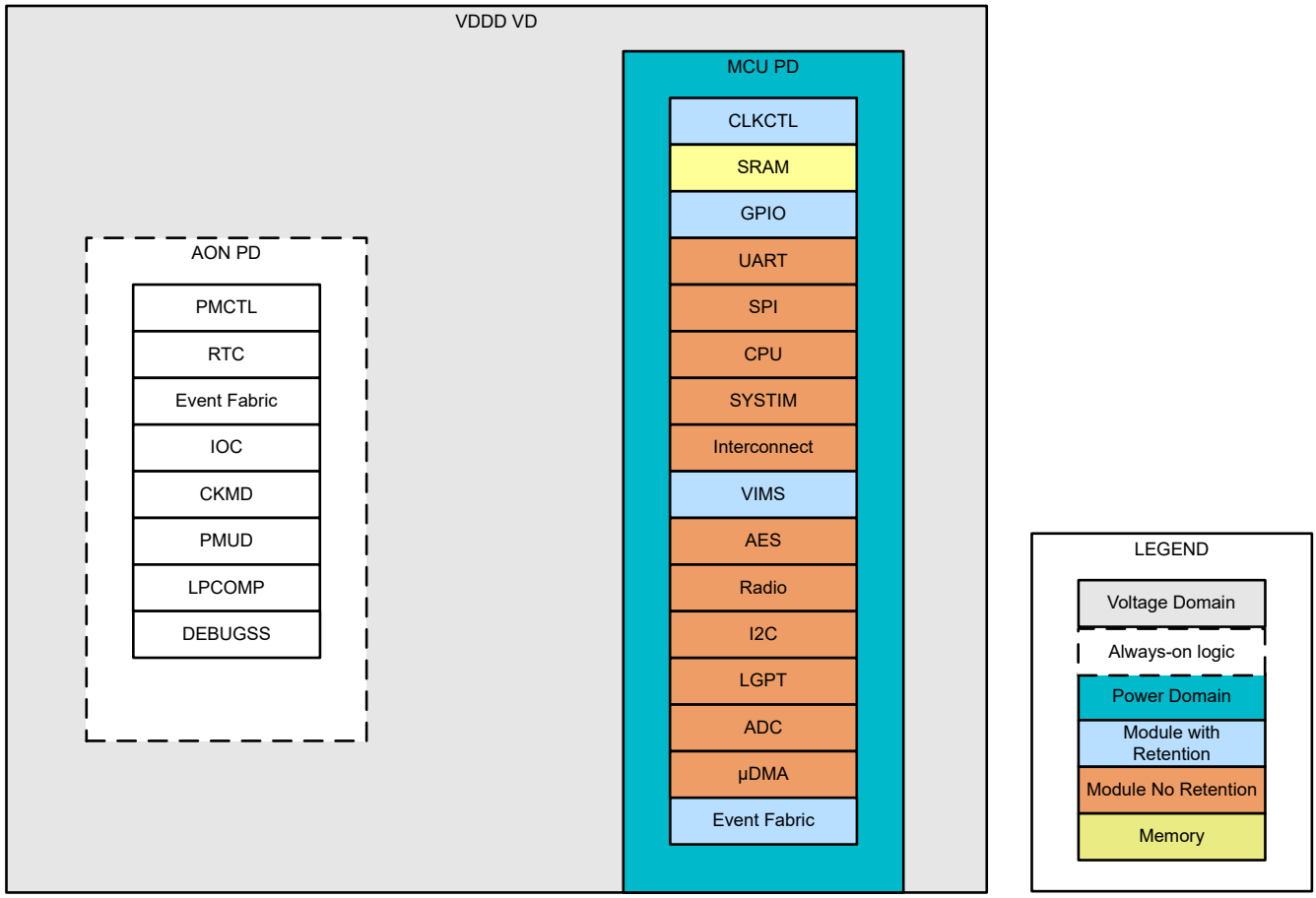


Figure 6-3. Power Partitioning

6.6 Clocks

CC23xx uses a single clock for all of the MCU domain (CLKSVT), and a single clock for the AON domain (CLKULL).

6.6.1 CLKSVT

CLKSVT is a free-running 48MHz clock. Individual peripherals can be clock-gated to reduce power consumption. Individual clock gating settings are configured in CLKCTL.

6.6.2 CLKULL

All AON modules receive a continuously running clock CLKULL, which runs at 24MHz when the CPU domain is powered (active and idle power states) and 32.768kHz when the CPU domain is not powered (standby power state).

6.7 Resets

Only a global asynchronous reset is available, no partial or subsystem reset is supported. Reset can be triggered by:

- Reset pin
- Power-on reset
- VDDS brown-out detector reset
- VDDR brown-out detector reset
- Watchdog reset
- CPU hardfault
- Software reset request

- Debug reset request
- LF clock loss
- Thermal shutdown release

The PMCTL.RSTSTA[2:0] RESETSRC and PMCTL.SYSSRC[7:4] SYSSRC bit fields are populated after reset and report which of the above caused the reset. The PMCTL.RSTSTA register is read by the ROM boot sequence and used to determine which wake-up action to take. The PMCTL.RSTSTA register can also be read by the user application to take appropriate action. The user application can also determine whether the system was woken from shutdown by reading the PMCTL.RSTSTA[17] SDDET bit. If the bit is set, the system has woken from shutdown.

The hierarchy of reset signals is as follows:

- RSTN_POR: Everything is reset, everything is disabled
- RSTN Asserted until all reset sources are released, and all regulated and unregulated voltage supplies are above the minimum required levels

POR is the only reset source that clears the IceMelter and AON/ULL 3P3V REGBANK.

Any reset higher in the hierarchy propagates to everything below. Resets are released at least one clock cycle before any clock starts running or synchronously to the clock edge if that is not possible.

The reset pin on the device serves double duty as an "enable device" signal for a transceiver or network processor (NWP).

6.7.1 Watchdog Timer (WDT)

The Watchdog Timer (WDT) is used to regain control when the system has failed due to a software error or due to the failure of an external device to respond in the expected way specifically during standby. This WDT generates a reset when a time-out value is reached. Writing to CKMD.WDTCNT starts the counter, which starts counting down from the written value on every LFCLK. The WDT relies on a working LFCLK. The WDT will stop working if LFCLK is lost. In the case of a LFCLK loss, the device can be reset by the LF loss detection feature. See Section 6.7.2 for additional details.

If the CKMD.WDTCNT register is written with a new value while the WDT counter is counting, then the counter is loaded with the new value and continues counting. If CKMD.WDTTEST[0] STALLEN is set, the counter can stall when the microcontroller asserts the CPU Halt flag during debug.

To prevent the WDT configuration from being inadvertently altered by software, the write access to the watchdog registers is automatically locked by writing the CKMD.WDTLOCK register to any value. To unlock the WDT, write the CKMD.WDTLOCK register to the value 0x1ACCE551.

The WDT can be configured using the following sequence:

- Unlock the WDT module by writing 0x1ACCE551 to the CKMD.WDTLOCK register.
- Load the CKMD.WDTCNT register with the desired timer load value. The register will be locked upon write.

Note

When this counter is running, there is no way to stop this counter other than a device reset. This is to prevent accidentally disabling the WDT and leaving the device in an unresponsive state.

6.7.2 LF Loss Detection

When the LF clock loss feature is enabled by setting the PMCTL.RSTCTL[2] LFLOSS bit and CKMD.LFMONCTL[0] EN bit, a detected loss of the selected LF source (low frequency crystal (LFXT) or low frequency oscillator (LFOSC)) results in a system reset. After recovery, the PMCTL.RSTSTA[2:0] RESETSRC and PMCTL.RSTSTA[7:4] SYSSRC bit fields show clock loss as the source of reset.

6.8 AON (REG3V3) Register Bank

The device has an AON Register Bank that is directly supplied by VDDS and only reset on POR. The content of this register bank is retained in shutdown and across all resets, except POR. The register bank is accessed through the PMCTL.AONRSET1, PMCTL.AONRCLR1, and PMCTL.AONRSTA1 registers.

6.9 CKMD Registers

Table 6-4 lists the memory-mapped registers for the CKMD registers. All register offset addresses not listed in Table 6-4 should be considered as reserved locations and the register contents should not be modified.

Table 6-4. CKMD Registers

Offset	Acronym	Register Name	Section
0h	DESC	Description Register.	Go
44h	IMASK	Interrupt mask.	Go
48h	RIS	Raw interrupt status.	Go
4Ch	MIS	Masked interrupt status.	Go
50h	ISET	Interrupt set register.	Go
54h	ICLR	Interrupt clear register.	Go
58h	IMSET	Interrupt mask set register.	Go
5Ch	IMCLR	Interrupt mask clear register.	Go
80h	HFOSCCTL	Internal. Only to be used through TI provided API.	Go
84h	HFXTCTL	High frequency crystal control	Go
8Ch	LFOSCCTL	Low frequency oscillator control	Go
90h	LFXTCTL	Low frequency crystal control	Go
94h	LFQUALCTL	Low frequency clock qualification control	Go
98h	LFINCCTL	Low frequency time increment control	Go
9Ch	LFINCOVR	Low frequency time increment override control	Go
A0h	AMPADCCTL	Internal. Only to be used through TI provided API.	Go
A4h	HFTRACKCTL	High frequency tracking loop control	Go
A8h	LDOCTL	Internal. Only to be used through TI provided API.	Go
ACh	NABIASCTL	Nanoamp-bias control	Go
B0h	LFMONCTL	Low-frequency clock-monitor control	Go
C0h	LFCLKSEL	Low frequency clock selection	Go
C4h	TDCCLKSEL	Internal. Only to be used through TI provided API.	Go
C8h	ADCCLKSEL	ADC clock selection	Go
E0h	LFCLKSTAT	Low-frequency clock status	Go
E4h	HFXTSTAT	HFXT status information	Go
E8h	AMPADCSTAT	Internal. Only to be used through TI provided API.	Go
ECh	TRACKSTAT	HFOSC tracking loop status information	Go
F0h	AMPSTAT	HFXT Amplitude Compensation Status	Go
100h	ATBCTL0	Internal. Only to be used through TI provided API.	Go
104h	ATBCTL1	Internal. Only to be used through TI provided API.	Go
108h	DTBCTL	Digital test bus mux control	Go
110h	TRIM0	Internal. Only to be used through TI provided API.	Go
114h	TRIM1	Internal. Only to be used through TI provided API.	Go
118h	HFXTINIT	Initial values for HFXT ramping	Go
11Ch	HFXTTARG	Target values for HFXT ramping	Go
120h	HFXTDYN	Alternative target values for HFXT configuration	Go
124h	AMPCFG0	Amplitude Compensation Configuration 0	Go
128h	AMPCFG1	Amplitude Compensation Configuration 1	Go
12Ch	LOOPCFG	Configuration Register for the Tracking Loop	Go
200h	TDCCTL	Internal. Only to be used through TI provided API.	Go
204h	TDCSTAT	Internal. Only to be used through TI provided API.	Go

Table 6-4. CKMD Registers (continued)

Offset	Acronym	Register Name	Section
208h	TDCRESULT	Internal. Only to be used through TI provided API.	Go
20Ch	TDCSATCFG	Internal. Only to be used through TI provided API.	Go
210h	TDCTRIGSRC	Internal. Only to be used through TI provided API.	Go
214h	TDCTRIGCNT	Internal. Only to be used through TI provided API.	Go
218h	TDCTRIGCNTLOAD	Internal. Only to be used through TI provided API.	Go
21Ch	TDCTRIGCNTCFG	Internal. Only to be used through TI provided API.	Go
220h	TDCPRECTL	Internal. Only to be used through TI provided API.	Go
224h	TDCPRECNTR	Internal. Only to be used through TI provided API.	Go
300h	WDTCNT	WDT counter value register	Go
304h	WDTTEST	WDT test mode register	Go
308h	WDTLOCK	WDT lock register	Go

Complex bit access types are encoded to fit into small table cells. Table 6-5 shows the codes that are used for access types in this section.

Table 6-5. CKMD Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

6.9.1 DESC Register (Offset = 0h) [Reset = 9B4B1000h]

DESC is shown in Table 6-6.

Return to the Summary Table.

Description Register.
This register provides IP module ID, revision information, instance index and standard MMR registers offset.

Table 6-6. DESC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	MODID	R	9B4Bh	Module identifier used to uniquely identify this IP.
15-12	STDIPOFF	R	1h	Standard IP MMR block offset. Standard IP MMRs are the set of from aggregated IRQ registers till DTB. 0: Standard IP MMRs do not exist 0x1-0xF: Standard IP MMRs begin at offset of (64*STDIPOFF from the base IP address) NOTE: This IP does not have DTB as part of the Standard IP MMRs. It uses DTBCTL instead.
11-8	RESERVED	R	0h	Reserved
7-4	MAJREV	R	0h	Major revision of IP (0-15).
3-0	MINREV	R	0h	Minor revision of IP (0-15).

6.9.2 IMASK Register (Offset = 44h) [Reset = 00000000h]

IMASK is shown in Table 6-7.

Return to the Summary Table.

Interrupt mask.
This register selects interrupt sources which are allowed to pass from RIS to MIS when the corresponding bit-fields are set to 1.

Table 6-7. IMASK Register Field Descriptions

Bit	Field	Type	Reset	Description
31-18	RESERVED	R	0h	Reserved
17	LFTICK	R/W	0h	32kHz TICK to RTC and WDT. Either derived from selected LFCLK or generated from CLKULL in absence of LFCLK.
16	LFGEARRSTRT	R/W	0h	LFINC filter gearing restart. Indicates that the LFINC filter restarted gearing. Subsequent LFINC estimates may have higher variation.
15	AMPSETTLED	R/W	0h	HFXT Amplitude compensation - settled Indicates that the amplitude compensation FSM has reached the SETTLED or TCXOMODE state, and the controls configured in HFXTTARG or HFXTDYN are reached.
14	AMPCTRLATTARG	R/W	0h	HFXT Amplitude compensation - controls at target Indicates that the control values configured in HFXTTARG or HFXTDYN are reached. Applies to Q1CAP, Q2CAP and IREF.
13	PRELFEDGE	R/W	0h	Pre-LF clock edge detect. Indicates that a positive edge occured on the selected pre-LF clock LFCLKSEL.PRE. Can be used by software to confirm that a LF clock source is running and within the expected frequency, before selecting it as the main LF clock source.
12	LFCLKLOSS	R/W	0h	LF clock is lost. Indicates that no LF clock edge occured for ~49us (~1.6 times nominal period). The system will automatically fall-back to generating LFTICK based on CLKULL, to avoid timing corruption. Note that this signal is NOT related to the analog LF clock-loss detector which can reset the device during STANDBY.
11	LFCLKOOR	R/W	0h	LF clock period out-of-range. Indicates that a LF clock period was measured to be out-of-range, according to LFQUALCTL.MAXERR.
10	LFCLKGOOD	R/W	0h	LF clock good. Indicates that the LF clock is good, according to the configuration in LFQUALCTL.
9	LFINCUPD	R/W	0h	LFINC updated. Indicates that a new LFINC measurement value is available in LFCLKSTAT.LFINC.
8	TDCDONE	R/W	0h	TDC done event. Indicates that the TDC measurement is done.
7	ADCPEAKUPD	R/W	0h	HFXT-ADC PEAK measurement update event. Indicates that the HFXT-ADC PEAK measurement is done.
6	ADCBIASUPD	R/W	0h	HFXT-ADC BIAS measurement update event. Indicates that the HFXT-ADC BIAS measurement is done.
5	ADCCOMPUPD	R/W	0h	HFXT-ADC comparison update event. Indicates that the HFXT-ADC comparison is done.

Table 6-7. IMASK Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
4	TRACKREFOOR	R/W	0h	Out-of-range indication from the tracking loop. Indicates that the selected reference clock frequency of the tracking loop is out-of-range.
3	TRACKREFLOSS	R/W	0h	Clock loss indication from the tracking loop. Indicates that the selected reference clock of the tracking loop is lost.
2	HFXTAMPGOOD	R/W	0h	HFXT amplitude good indication.
1	HFXTFAULT	R/W	0h	HFXT fault indication. Indicates that HFXT did not start correctly, or its frequency is too low. HFXT will not recover from this fault and has to be restarted. This is only a one-time check at HFXT startup.
0	HFXTGOOD	R/W	0h	HFXT good indication. Indicates that HFXT started correctly. The frequency is not necessarily good enough for radio operation. This is only a one-time check at HFXT startup.

6.9.3 RIS Register (Offset = 48h) [Reset = 00000000h]

RIS is shown in Table 6-8.

Return to the Summary Table.

Raw interrupt status.
This register reflects the state of all pending interrupts, regardless of masking. This register allows the user to implement a poll scheme. A flag set in this register can be cleared by writing 1 to the corresponding ICLR register bit.

Table 6-8. RIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-18	RESERVED	R	0h	Reserved
17	LFTICK	R	0h	32kHz TICK to RTC and WDT. Either derived from selected LFCLK or generated from CLKULL in absence of LFCLK.
16	LFGEARRSTRT	R	0h	LFINC filter gearing restart. Indicates that the LFINC filter restarted gearing. Subsequent LFINC estimates may have higher variation.
15	AMPSETTLED	R	0h	HFXT Amplitude compensation - settled Indicates that the amplitude compensation FSM has reached the SETTLED or TCXOMODE state, and the controls configured in HFXTTARG or HFXTDYN are reached.
14	AMPCTRLATTARG	R	0h	HFXT Amplitude compensation - controls at target Indicates that the control values configured in HFXTTARG or HFXTDYN are reached. Applies to Q1CAP, Q2CAP and IREF.
13	PRELFEDGE	R	0h	Pre-LF clock edge detect. Indicates that a positive edge occured on the selected pre-LF clock LFCLKSEL.PRE. Can be used by software to confirm that a LF clock source is running and within the expected frequency, before selecting it as the main LF clock source.
12	LFCLKLOSS	R	0h	LF clock is lost. Indicates that no LF clock edge occured for ~49us (~1.6 times nominal period). The system will automatically fall-back to generating LFTICK based on CLKULL, to avoid timing corruption. Note that this signal is NOT related to the analog LF clock-loss detector which can reset the device during STANDBY.
11	LFCLKOOR	R	0h	LF clock period out-of-range. Indicates that a LF clock period was measured to be out-of-range, according to LFQUALCTL.MAXERR.
10	LFCLKGOOD	R	0h	LF clock good. Indicates that the LF clock is good, according to the configuration in LFQUALCTL.
9	LFINCUPD	R	0h	LFINC updated. Indicates that a new LFINC measurement value is available in LFCLKSTAT.LFINC.
8	TDCDONE	R	0h	TDC done event. Indicates that the TDC measurement is done.
7	ADCPEAKUPD	R	0h	HFXT-ADC PEAK measurement update event. Indicates that the HFXT-ADC PEAK measurement is done.
6	ADCBIASUPD	R	0h	HFXT-ADC BIAS measurement update event. Indicates that the HFXT-ADC BIAS measurement is done.
5	ADCCOMPUPD	R	0h	HFXT-ADC comparison update event. Indicates that the HFXT-ADC comparison is done.

Table 6-8. RIS Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
4	TRACKREFOOR	R	0h	Out-of-range indication from the tracking loop. Indicates that the selected reference clock frequency of the tracking loop is out-of-range.
3	TRACKREFLOSS	R	0h	Clock loss indication from the tracking loop. Indicates that the selected reference clock of the tracking loop is lost.
2	HFXTAMPGOOD	R	0h	HFXT amplitude good indication.
1	HFXTFAULT	R	0h	HFXT fault indication. Indicates that HFXT did not start correctly, or its frequency is too low. HFXT will not recover from this fault and has to be restarted. This is only a one-time check at HFXT startup.
0	HFXTGOOD	R	0h	HFXT good indication. Indicates that HFXT started correctly. The frequency is not necessarily good enough for radio operation. This is only a one-time check at HFXT startup.

6.9.4 MIS Register (Offset = 4Ch) [Reset = 00000000h]

MIS is shown in Table 6-9.

Return to the Summary Table.

Masked interrupt status.
This register is simply a bitwise AND of the contents of IMASK and RIS.*] registers. A flag set in this register can be cleared by writing 1 to the corresponding ICLR register bit.

Table 6-9. MIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-18	RESERVED	R	0h	Reserved
17	LFTICK	R	0h	32kHz TICK to RTC and WDT. Either derived from selected LFCLK or generated from CLKULL in absence of LFCLK.
16	LFGEARRSTRT	R	0h	LFINC filter gearing restart. Indicates that the LFINC filter restarted gearing. Subsequent LFINC estimates may have higher variation.
15	AMPSETTLED	R	0h	HFXT Amplitude compensation - settled Indicates that the amplitude compensation FSM has reached the SETTLED or TCXOMODE state, and the controls configured in HFXTTARG or HFXTDYN are reached.
14	AMPCTRLATTARG	R	0h	HFXT Amplitude compensation - controls at target Indicates that the control values configured in HFXTTARG or HFXTDYN are reached. Applies to Q1CAP, Q2CAP and IREF.
13	PRELFEDGE	R	0h	Pre-LF clock edge detect. Indicates that a positive edge occured on the selected pre-LF clock LFCLKSEL.PRE. Can be used by software to confirm that a LF clock source is running and within the expected frequency, before selecting it as the main LF clock source.
12	LFCLKLOSS	R	0h	LF clock is lost. Indicates that no LF clock edge occured for ~49us (~1.6 times nominal period). The system will automatically fall-back to generating LFTICK based on CLKULL, to avoid timing corruption. Note that this signal is NOT related to the analog LF clock-loss detector which can reset the device during STANDBY.
11	LFCLKOOR	R	0h	LF clock period out-of-range. Indicates that a LF clock period was measured to be out-of-range, according to LFQUALCTL.MAXERR.
10	LFCLKGOOD	R	0h	LF clock good. Indicates that the LF clock is good, according to the configuration in LFQUALCTL.
9	LFINCUPD	R	0h	LFINC updated. Indicates that a new LFINC measurement value is available in LFCLKSTAT.LFINC.
8	TDCDONE	R	0h	TDC done event. Indicates that the TDC measurement is done.
7	ADCPEAKUPD	R	0h	HFXT-ADC PEAK measurement update event. Indicates that the HFXT-ADC PEAK measurement is done.
6	ADCBIASUPD	R	0h	HFXT-ADC BIAS measurement update event. Indicates that the HFXT-ADC BIAS measurement is done.
5	ADCCOMPUPD	R	0h	HFXT-ADC comparison update event. Indicates that the HFXT-ADC comparison is done.

Table 6-9. MIS Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
4	TRACKREFOOR	R	0h	Out-of-range indication from the tracking loop. Indicates that the selected reference clock frequency of the tracking loop is out-of-range.
3	TRACKREFLOSS	R	0h	Clock loss indication from the tracking loop. Indicates that the selected reference clock of the tracking loop is lost.
2	HFXTAMPGOOD	R	0h	HFXT amplitude good indication.
1	HFXTFAULT	R	0h	HFXT fault indication. Indicates that HFXT did not start correctly, or its frequency is too low. HFXT will not recover from this fault and has to be restarted. This is only a one-time check at HFXT startup.
0	HFXTGOOD	R	0h	HFXT good indication. Indicates that HFXT started correctly. The frequency is not necessarily good enough for radio operation. This is only a one-time check at HFXT startup.

6.9.5 ISET Register (Offset = 50h) [Reset = 00000000h]

ISET is shown in Table 6-10.

Return to the Summary Table.

Interrupt set register.
This register can used by software for diagnostics and safety checking purposes. Writing a 1 to a bit in this register will set the event and the corresponding RIS bit also gets set. If the corresponding IMASK bit is set, then the corresponding MIS register bit also gets set.

Table 6-10. ISET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-18	RESERVED	R	0h	Reserved
17	LFTICK	W	0h	32kHz TICK to RTC and WDT. Either derived from selected LFCLK or generated from CLKULL in absence of LFCLK.
16	LFGEARRSTRT	W	0h	LFINC filter gearing restart. Indicates that the LFINC filter restarted gearing. Subsequent LFINC estimates may have higher variation.
15	AMPSETTLED	W	0h	HFXT Amplitude compensation - settled Indicates that the amplitude compensation FSM has reached the SETTLED or TCXOMODE state, and the controls configured in HFXTTARG or HFXTDYN are reached.
14	AMPCTRLATTARG	W	0h	HFXT Amplitude compensation - controls at target Indicates that the control values configured in HFXTTARG.Q1CAP, HFXTTARG.Q2CAP and HFXTTARG.IREF or HFXTDYN.Q1CAP, HFXTDYN.Q2CAP and HFXTDYN.IREF are reached.
13	PRELFEDGE	W	0h	Pre-LF clock edge detect. Indicates that a positive edge occured on the selected pre-LF clock LFCLKSEL.PRE. Can be used by software to confirm that a LF clock source is running and within the expected frequency, before selecting it as the main LF clock source.
12	LFCLKLOSS	W	0h	LF clock is lost. Indicates that no LF clock edge occured for ~49us (~1.6 times nominal period). The system will automatically fall-back to generating LFTICK based on CLKULL, to avoid timing corruption. Note that this signal is NOT related to the analog LF clock-loss detector which can reset the device during STANDBY.
11	LFCLKOOR	W	0h	LF clock period out-of-range. Indicates that a LF clock period was measured to be out-of-range, according to LFQUALCTL.MAXERR.
10	LFCLKGOOD	W	0h	LF clock good. Indicates that the LF clock is good, according to the configuration in LFQUALCTL.
9	LFINCUPD	W	0h	LFINC updated. Indicates that a new LFINC measurement value is available in LFCLKSTAT.LFINC.
8	TDCDONE	W	0h	TDC done event. Indicates that the TDC measurement is done.
7	ADCPEAKUPD	W	0h	HFXT-ADC PEAK measurement update event. Indicates that the HFXT-ADC PEAK measurement is done.
6	ADCBIASUPD	W	0h	HFXT-ADC BIAS measurement update event. Indicates that the HFXT-ADC BIAS measurement is done.
5	ADCCOMPUPD	W	0h	HFXT-ADC comparison update event. Indicates that the HFXT-ADC comparison is done.

Table 6-10. ISET Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
4	TRACKREFOOR	W	0h	Out-of-range indication from the tracking loop. Indicates that the selected reference clock frequency of the tracking loop is out-of-range.
3	TRACKREFLOSS	W	0h	Clock loss indication from the tracking loop. Indicates that the selected reference clock of the tracking loop is lost.
2	HFXTAMPGOOD	W	0h	HFXT amplitude good indication.
1	HFXTFAULT	W	0h	HFXT fault indication. Indicates that HFXT did not start correctly, or its frequency is too low. HFXT will not recover from this fault and has to be restarted. This is only a one-time check at HFXT startup.
0	HFXTGOOD	W	0h	HFXT good indication. Indicates that HFXT started correctly. The frequency is not necessarily good enough for radio operation. This is only a one-time check at HFXT startup.

6.9.6 ICLR Register (Offset = 54h) [Reset = 00000000h]

ICLR is shown in Table 6-11.

Return to the Summary Table.

Interrupt clear register.
This register allows software to clear interrupts. Writing a 1 to a bit in this register will clear the event and the corresponding RIS bit also gets cleared. If the corresponding IMASK bit is set, then the corresponding MIS register bit also gets cleared.

Table 6-11. ICLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-18	RESERVED	R	0h	Reserved
17	LFTICK	W	0h	32kHz TICK to RTC and WDT. Either derived from selected LFCLK or generated from CLKULL in absence of LFCLK.
16	LFGEARRSTRT	W	0h	LFINC filter gearing restart. Indicates that the LFINC filter restarted gearing. Subsequent LFINC estimates may have higher variation.
15	AMPSETTLED	W	0h	HFXT Amplitude compensation - settled Indicates that the amplitude compensation FSM has reached the SETTLED or TCXOMODE state, and the controls configured in HFXTTARG or HFXTDYN are reached.
14	AMPCTRLATTARG	W	0h	HFXT Amplitude compensation - controls at target Indicates that the control values configured in HFXTTARG or HFXTDYN are reached. Applies to Q1CAP, Q2CAP and IREF.
13	PRELFEDGE	W	0h	Pre-LF clock edge detect. Indicates that a positive edge occured on the selected pre-LF clock LFCLKSEL.PRE. Can be used by software to confirm that a LF clock source is running and within the expected frequency, before selecting it as the main LF clock source.
12	LFCLKLOSS	W	0h	LF clock is lost. Indicates that no LF clock edge occured for ~49us (~1.6 times nominal period). The system will automatically fall-back to generating LFTICK based on CLKULL, to avoid timing corruption. Note that this signal is NOT related to the analog LF clock-loss detector which can reset the device during STANDBY.
11	LFCLKOOR	W	0h	LF clock period out-of-range. Indicates that a LF clock period was measured to be out-of-range, according to LFQUALCTL.MAXERR.
10	LFCLKGOOD	W	0h	LF clock good. Indicates that the LF clock is good, according to the configuration in LFQUALCTL.
9	LFINCUPD	W	0h	LFINC updated. Indicates that a new LFINC measurement value is available in LFCLKSTAT.LFINC.
8	TDCDONE	W	0h	TDC done event. Indicates that the TDC measurement is done.
7	ADCPEAKUPD	W	0h	HFXT-ADC PEAK measurement update event. Indicates that the HFXT-ADC PEAK measurement is done.
6	ADCBIASUPD	W	0h	HFXT-ADC BIAS measurement update event. Indicates that the HFXT-ADC BIAS measurement is done.
5	ADCCOMPUPD	W	0h	HFXT-ADC comparison update event. Indicates that the HFXT-ADC comparison is done.

Table 6-11. ICLR Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
4	TRACKREFOOR	W	0h	Out-of-range indication from the tracking loop. Indicates that the selected reference clock frequency of the tracking loop is out-of-range.
3	TRACKREFLOSS	W	0h	Clock loss indication from the tracking loop. Indicates that the selected reference clock of the tracking loop is lost.
2	HFXTAMPGOOD	W	0h	HFXT amplitude good indication.
1	HFXTFAULT	W	0h	HFXT fault indication. Indicates that HFXT did not start correctly, or its frequency is too low. HFXT will not recover from this fault and has to be restarted. This is only a one-time check at HFXT startup.
0	HFXTGOOD	W	0h	HFXT good indication. Indicates that HFXT started correctly. The frequency is not necessarily good enough for radio operation. This is only a one-time check at HFXT startup.

6.9.7 IMSET Register (Offset = 58h) [Reset = 00000000h]

IMSET is shown in Table 6-12.

Return to the Summary Table.

Interrupt mask set register.
Writing a 1 to a bit in this register will set the corresponding IMASK bit.

Table 6-12. IMSET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-18	RESERVED	R	0h	Reserved
17	LFTICK	W	0h	32kHz TICK to RTC and WDT. Either derived from selected LFCLK or generated from CLKULL in absence of LFCLK.
16	LFGEARRSTRT	W	0h	LFINC filter gearing restart. Indicates that the LFINC filter restarted gearing. Subsequent LFINC estimates may have higher variation.
15	AMPSETTLED	W	0h	HFXT Amplitude compensation - settled Indicates that the amplitude compensation FSM has reached the SETTLED or TCXOMODE state, and the controls configured in HFXTTARG or HFXTDYN are reached.
14	AMPCTRLATTARG	W	0h	HFXT Amplitude compensation - controls at target Indicates that the control values configured in HFXTTARG or HFXTDYN are reached. Applies to Q1CAP, Q2CAP and IREF.
13	PRELFEDGE	W	0h	Pre-LF clock edge detect. Indicates that a positive edge occured on the selected pre-LF clock LFCLKSEL.PRE. Can be used by software to confirm that a LF clock source is running and within the expected frequency, before selecting it as the main LF clock source.
12	LFCLKLOSS	W	0h	LF clock is lost. Indicates that no LF clock edge occured for ~49us (~1.6 times nominal period). The system will automatically fall-back to generating LFTICK based on CLKULL, to avoid timing corruption. Note that this signal is NOT related to the analog LF clock-loss detector which can reset the device during STANDBY.
11	LFCLKOOR	W	0h	LF clock period out-of-range. Indicates that a LF clock period was measured to be out-of-range, according to LFQUALCTL.MAXERR.
10	LFCLKGOOD	W	0h	LF clock good. Indicates that the LF clock is good, according to the configuration in LFQUALCTL.
9	LFINCUPD	W	0h	LFINC updated. Indicates that a new LFINC measurement value is available in LFCLKSTAT.LFINC.
8	TDCDONE	W	0h	TDC done event. Indicates that the TDC measurement is done.
7	ADCPEAKUPD	W	0h	HFXT-ADC PEAK measurement update event. Indicates that the HFXT-ADC PEAK measurement is done.
6	ADCBIASUPD	W	0h	HFXT-ADC BIAS measurement update event. Indicates that the HFXT-ADC BIAS measurement is done.
5	ADCCOMPUPD	W	0h	HFXT-ADC comparison update event. Indicates that the HFXT-ADC comparison is done.
4	TRACKREFOOR	W	0h	Out-of-range indication from the tracking loop. Indicates that the selected reference clock frequency of the tracking loop is out-of-range.

Table 6-12. IMSET Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
3	TRACKREFLOSS	W	0h	Clock loss indication from the tracking loop. Indicates that the selected reference clock of the tracking loop is lost.
2	HFXTAMPGOOD	W	0h	HFXT amplitude good indication.
1	HFXTFAULT	W	0h	HFXT fault indication. Indicates that HFXT did not start correctly, or its frequency is too low. HFXT will not recover from this fault and has to be restarted. This is only a one-time check at HFXT startup.
0	HFXTGOOD	W	0h	HFXT good indication. Indicates that HFXT started correctly. The frequency is not necessarily good enough for radio operation. This is only a one-time check at HFXT startup.

6.9.8 IMCLR Register (Offset = 5Ch) [Reset = 00000000h]

IMCLR is shown in Table 6-13.

Return to the Summary Table.

Interrupt mask clear register.
Writing a 1 to a bit in this register will clear the corresponding IMASK bit.

Table 6-13. IMCLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-18	RESERVED	R	0h	Reserved
17	LFTICK	W	0h	32kHz TICK to RTC and WDT. Either derived from selected LFCLK or generated from CLKULL in absence of LFCLK.
16	LFGEARRSTRT	W	0h	LFINC filter gearing restart. Indicates that the LFINC filter restarted gearing. Subsequent LFINC estimates may have higher variation.
15	AMPSETTLED	W	0h	HFXT Amplitude compensation - settled Indicates that the amplitude compensation FSM has reached the SETTLED or TCXOMODE state, and the controls configured in HFXTTARG or HFXTDYN are reached.
14	AMPCTRLATTARG	W	0h	HFXT Amplitude compensation - controls at target Indicates that the control values configured in HFXTTARG or HFXTDYN are reached. Applies to Q1CAP, Q2CAP and IREF.
13	PRELFEDGE	W	0h	Pre-LF clock edge detect. Indicates that a positive edge occured on the selected pre-LF clock LFCLKSEL.PRE. Can be used by software to confirm that a LF clock source is running and within the expected frequency, before selecting it as the main LF clock source.
12	LFCLKLOSS	W	0h	LF clock is lost. Indicates that no LF clock edge occured for ~49us (~1.6 times nominal period). The system will automatically fall-back to generating LFTICK based on CLKULL, to avoid timing corruption. Note that this signal is NOT related to the analog LF clock-loss detector which can reset the device during STANDBY.
11	LFCLKOOR	W	0h	LF clock period out-of-range. Indicates that a LF clock period was measured to be out-of-range, according to LFQUALCTL.MAXERR.
10	LFCLKGOOD	W	0h	LF clock good. Indicates that the LF clock is good, according to the configuration in LFQUALCTL.
9	LFINCUPD	W	0h	LFINC updated. Indicates that a new LFINC measurement value is available in LFCLKSTAT.LFINC.
8	TDCDONE	W	0h	TDC done event. Indicates that the TDC measurement is done.
7	ADCPEAKUPD	W	0h	HFXT-ADC PEAK measurement update event. Indicates that the HFXT-ADC PEAK measurement is done.
6	ADCBIASUPD	W	0h	HFXT-ADC BIAS measurement update event. Indicates that the HFXT-ADC BIAS measurement is done.
5	ADCCOMPUPD	W	0h	HFXT-ADC comparison update event. Indicates that the HFXT-ADC comparison is done.
4	TRACKREFOOR	W	0h	Out-of-range indication from the tracking loop. Indicates that the selected reference clock frequency of the tracking loop is out-of-range.

Table 6-13. IMCLR Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
3	TRACKREFLOSS	W	0h	Clock loss indication from the tracking loop. Indicates that the selected reference clock of the tracking loop is lost.
2	HFXTAMPGOOD	W	0h	HFXT amplitude good indication.
1	HFXTFAULT	W	0h	HFXT fault indication. Indicates that HFXT did not start correctly, or its frequency is too low. HFXT will not recover from this fault and has to be restarted. This is only a one-time check at HFXT startup.
0	HFXTGOOD	W	0h	HFXT good indication. Indicates that HFXT started correctly. The frequency is not necessarily good enough for radio operation. This is only a one-time check at HFXT startup.

6.9.9 HFOSCCTL Register (Offset = 80h) [Reset = 00000000h]

HFOSCCTL is shown in Table 6-14.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-14. HFOSCCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	PW	W	0h	Internal. Only to be used through TI provided API.
23-9	RESERVED	R	0h	Reserved
8	CLKSVTOVR	R/W	0h	Internal. Only to be used through TI provided API.
7-2	RESERVED	R	0h	Reserved
1	FORCEOFF	R/W	0h	Internal. Only to be used through TI provided API.
0	QUALBYP	R/W	0h	Internal. Only to be used through TI provided API.

6.9.10 HFXTCTL Register (Offset = 84h) [Reset = 00000000h]

HFXTCTL is shown in Table 6-15.

Return to the Summary Table.

High frequency crystal control

Table 6-15. HFXTCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31	AMPOVR	R/W	0h	Internal. Only to be used through TI provided API.
30-27	RESERVED	R	0h	Reserved
26	BIASEN	R/W	0h	Internal. Only to be used through TI provided API.
25	LPBUFEN	R/W	0h	Internal. Only to be used through TI provided API.
24	INJECT	R/W	0h	Internal. Only to be used through TI provided API.
23	QUALBYP	R/W	0h	Internal. Only to be used through TI provided API.
22-20	RESERVED	R	0h	Reserved
19-8	QUALDLY	R/W	0h	Skip potentially unstable clock cycles after enabling HFXT. Number of cycles skipped is 8*QUALDLY.
7	TCXOMODE	R/W	0h	Temperature compensated crystal oscillator mode. Set this bit if a TXCO is connected.
6	TCXOTYPE	R/W	0h	Type of temperature compensated crystal used. Only has effect if TCXOMODE is set. 0h = Use with clipped-sine TCXO 1h = Use with CMOS TCXO
5-3	RESERVED	R	0h	Reserved
2	AUTOEN	R/W	0h	Internal. Only to be used through TI provided API.
1	HPBUFEN	R/W	0h	High performance clock buffer enable. This bit controls the clock output for the RF PLL. It is required for radio operation.
0	EN	R/W	0h	Internal. Only to be used through TI provided API.

6.9.11 LFOSCCTL Register (Offset = 8Ch) [Reset = 00000000h]

LFOSCCTL is shown in Table 6-16.

Return to the Summary Table.

Low frequency oscillator control

Table 6-16. LFOSCCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Reserved
0	EN	R/W	0h	LFOSC enable

6.9.12 LFXTCTL Register (Offset = 90h) [Reset = 00000000h]

LFXTCTL is shown in Table 6-17.

Return to the Summary Table.

Low frequency crystal control

Table 6-17. LFXTCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-15	RESERVED	R	0h	Reserved
14-13	LEAKCOMP	R/W	0h	Leakage compensation control 0h = Full leakage compensation 1h = Half leakage compensation 3h = No leakage compensation
12	BUFBIAS	R/W	0h	Control the BIAS current of the input amp in LP buffer 0h = Minimum bias current: 25nA 1h = Maximum bias current: 50nA
11-8	AMPBIAS	R/W	0h	Adjust current mirror ratio into oscillator core. This value is depending on crystal and is set by FW. This field uses a 2's complement encoding.
7-6	BIASBOOST	R/W	0h	Boost oscillator amplitude This value depends on the crystal and needs to be configured by Firmware.
5-4	REGBIAS	R/W	0h	Regulation loop bias resistor value This value depends on the crystal and needs to be configured by Firmware.
3	RESERVED	R	0h	Reserved
2	HPBUFEN	R/W	0h	Control the buffer used. In normal operation, low-power buffer is used in all device modes. The high-performance buffer is only used for test purposes.
1	AMPREGMODE	R/W	0h	Amplitude regulation mode 0h = Amplitude control loop enabled 1h = Amplitude control loop disabled
0	EN	R/W	0h	LFXT enable

6.9.13 LFQUALCTL Register (Offset = 94h) [Reset = 00002064h]

LFQUALCTL is shown in Table 6-18.

Return to the Summary Table.

Low frequency clock qualification control

Table 6-18. LFQUALCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-14	RESERVED	R	0h	Reserved
13-8	MAXERR	R/W	20h	Maximum LFCLK period error. Value given in microseconds, 3 integer bits + 3 fractional bits.
7-0	CONSEC	R/W	64h	Number of consecutive times the LFCLK period error has to be smaller than MAXERR to be considered "good". Setting this value to 0 will bypass clock qualification, and the "good" indicator will always be 1.

6.9.14 LFINCCTL Register (Offset = 98h) [Reset = 9E848014h]

LFINCCTL is shown in Table 6-19.

Return to the Summary Table.

Low frequency time increment control

Table 6-19. LFINCCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31	PREVENTSTBY	R/W	1h	Controls if the LFINC filter prevents STANBY entry until settled. 0h = Disable. Do not prevent STANDBY entry. 1h = Enable. Prevent STANDBY entry.
30	RESERVED	R	0h	Reserved
29-8	INT	R/W	001E8480h	Integral part of the LFINC filter. This value is updated by Hardware to reflect the current state of the filter. It can also be written to change the current state.
7	STOPGEAR	R/W	0h	Controls the final gear of the LFINC filter. 0h = Lowest final gear. Best settling, but less dynamic frequency tracking. 1h = Highest final gear. Best dynamic frequency tracking, but higher variation in filter value.
6-5	ERRTHR	R/W	0h	Controls the threshold for gearing restart of the LFINC filter. Only effective if GEARRSTRT is not ONETHR or TWOTHR. 0h = Restart gearing on large error. Fewer false restarts, slower response on small frequency shifts. 1h = Middle value towards LARGE. 2h = Middle value towards SMALL. 3h = Restart gearing on small error. Potentially more false restarts, faster response on small frequency shifts.
4-3	GEARRSTRT	R/W	2h	Controls gearing restart of the LFINC filter. 0h = Never restart gearing. Very stable filter value, but very slow response on frequency changes. 1h = Restart gearing when the error accumulator crosses the threshold once. 2h = Restart gearing when the error accumulator crosses the threshold twice in a row.
2	SOFTRSTRT	R/W	1h	Use a higher gear after re-enabling / wakeup. The filter will require 16-24 LFCLK periods to settle (depending on STOPGEAR), but may respond faster to frequency changes during STANDBY. 0h = Don't use soft gearing restarts 1h = Use soft gearing restarts
1-0	RESERVED	R	0h	Reserved

6.9.15 LFINCOVR Register (Offset = 9Ch) [Reset = 00000000h]

LFINCOVR is shown in Table 6-20.

Return to the Summary Table.

Low frequency time increment override control

Table 6-20. LFINCOVR Register Field Descriptions

Bit	Field	Type	Reset	Description
31	OVERRIDE	R/W	0h	Override LF increment Use the value provided in LFINC instead of the value calculated by Hardware.
30-22	RESERVED	R	0h	Reserved
21-0	LFINC	R/W	0h	LF increment value This value is used when OVERRIDE is set to 1. Otherwise the value is calculated automatically.

6.9.16 AMPADCCTL Register (Offset = A0h) [Reset = 00000000h]

AMPADCCTL is shown in Table 6-21.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-21. AMPADCCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31	SWOVR	R/W	0h	Internal. Only to be used through TI provided API.
30-18	RESERVED	R	0h	Reserved
17	PEAKDETEN	R/W	0h	Internal. Only to be used through TI provided API.
16	ADCEN	R/W	0h	Internal. Only to be used through TI provided API.
15	RESERVED	R	0h	Reserved
14-8	COMPVAL	R/W	0h	Internal. Only to be used through TI provided API.
7-5	RESERVED	R	0h	Reserved
4	SRCSEL	R/W	0h	Internal. Only to be used through TI provided API.
3-2	RESERVED	R	0h	Reserved
1	COMPSTRT	R/W	0h	Internal. Only to be used through TI provided API.
0	SARSTRT	R/W	0h	Internal. Only to be used through TI provided API.

6.9.17 HFTRACKCTL Register (Offset = A4h) [Reset = 00400000h]

HFTRACKCTL is shown in Table 6-22.

Return to the Summary Table.

High frequency tracking loop control

Table 6-22. HFTRACKCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31	EN	R/W	0h	Enable tracking loop.
30	DSMBYP	R/W	0h	Bypass Delta-Sigma-Modulation of fine trim.
29-28	RESERVED	R	0h	Reserved
27-26	REFCLK	R/W	0h	Select the reference clock for the tracking loop. Change only while the tracking loop is disabled. 0h = Select HFXT as reference clock. 1h = Select LRF reference clock. 2h = Select GPI as reference clock.
25-0	RATIO	R/W	00400000h	Reference clock ratio. RATIO = 24MHz / (2*reference-frequency) * 2^{24} Commonly used reference clock frequencies are provided as enumerations. 00400000h = Use for 48MHz reference clock 01800000h = Use for 8MHz reference clock 03000000h = Use for 4MHz reference clock

6.9.18 LDOCTL Register (Offset = A8h) [Reset = 00000000h]

LDOCTL is shown in Table 6-23.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-23. LDOCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31	SWOVR	R/W	0h	Internal. Only to be used through TI provided API.
30-5	RESERVED	R	0h	Reserved
4	HFXTLVLEN	R/W	0h	Internal. Only to be used through TI provided API.
3	STARTCTL	R/W	0h	Internal. Only to be used through TI provided API.
2	START	R/W	0h	Internal. Only to be used through TI provided API.
1	BYPASS	R/W	0h	Internal. Only to be used through TI provided API.
0	EN	R/W	0h	Internal. Only to be used through TI provided API.

6.9.19 NABIASCTL Register (Offset = ACh) [Reset = 00000000h]

NABIASCTL is shown in Table 6-24.

Return to the Summary Table.

Nanoamp-bias control

Table 6-24. NABIASCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Reserved
0	EN	R/W	0h	Enable nanoamp-bias

6.9.20 LFMONCTL Register (Offset = B0h) [Reset = 00000000h]

LFMONCTL is shown in Table 6-25.

Return to the Summary Table.

Low-frequency clock-monitor control

Table 6-25. LFMONCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Reserved
0	EN	R/W	0h	Enable LFMONITOR. Enable only after a LF clock source has been selected, enabled and is stable. If LFMONITOR detects a clock loss, the system will be reset.

6.9.21 LFCLKSEL Register (Offset = C0h) [Reset = 00000000h]

LFCLKSEL is shown in Table 6-26.

Return to the Summary Table.

Low frequency clock selection

Table 6-26. LFCLKSEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3-2	PRE	R/W	0h	Select low frequency clock source for the PRELFCLK interrupt. Can be used by Software to confirm that the clock is running and it's frequency is good, before selecting it in MAIN. 0h = No clock. Output will be tied low. 1h = Low frequency on-chip oscillator 2h = Low frequency crystal oscillator 3h = External LF clock through GPI.
1-0	MAIN	R/W	0h	Select the main low frequency clock source. If running, this clock will be used to generate LFTICK and as CLKULL during STANDBY. If not running, LFTICK will be generated from HFOSC and STANDBY entry will be prevented. 0h = No LF clock selected. LFTICK will be generated from HFOSC, STANDBY entry will be prevented. 1h = Low frequency on-chip oscillator 2h = Low frequency crystal oscillator 3h = External LF clock through GPI.

6.9.22 TDCCLKSEL Register (Offset = C4h) [Reset = 00000000h]

TDCCLKSEL is shown in Table 6-27.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-27. TDCCLKSEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1-0	REFCLK	R/W	0h	Internal. Only to be used through TI provided API.

6.9.23 ADCCLKSEL Register (Offset = C8h) [Reset = 00000000h]

ADCCLKSEL is shown in Table 6-28.

Return to the Summary Table.

ADC clock selection

Table 6-28. ADCCLKSEL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1-0	SRC	R/W	0h	Select ADC clock source Change only while ADC is disabled. 0h = 48MHz CLKSVT 1h = 48MHz HFXT

6.9.24 LFCLKSTAT Register (Offset = E0h) [Reset = 00000000h]

LFCLKSTAT is shown in Table 6-29.

Return to the Summary Table.

Low-frequency clock status

Table 6-29. LFCLKSTAT Register Field Descriptions

Bit	Field	Type	Reset	Description
31	GOOD	R	0h	Low frequency clock good Note: This is only a coarse frequency check based on LFQUALCTL. The clock may not be accurate enough for timing purposes.
30-26	RESERVED	R	0h	Reserved
25	FLTSETTLED	R	0h	LFINC filter is running and settled.
24	LFTICKSRC	R	0h	Source of LFTICK. 0h = LFTICK generated from the selected LFCLK 1h = LFTICK generated from CLKULL (LFCLK not available)
23-22	LFINCSRC	R	0h	Source of LFINC used by the RTC. This value depends on LFINCOVR.OVERRIDE, LF clock availability, HF tracking loop status and the device state (ACTIVE/STANDBY). 0h = Using measured value. This value is updated by hardware and can be read from LFINC. 1h = Using filtered / average value. This value is updated by hardware and can be read and updated in LFINCCTL.INT. 2h = Using override value from LFINCOVR.LFINC 3h = Using FAKE LFTICKs with corresponding LFINC value.
21-0	LFINC	R	0h	Measured value of LFINC. Given in microseconds with 16 fractional bits. This value is calculated by Hardware. It is the LFCLK period according to CLKULL cycles.

6.9.25 HFXTSTAT Register (Offset = E4h) [Reset = 00000000h]

HFXTSTAT is shown in Table 6-30.

Return to the Summary Table.

HFXT status information

Table 6-30. HFXTSTAT Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30-16	STARTUPTIME	R	0h	HFXT startup time Can be used by software to plan starting HFXT ahead in time. Measured whenever HFXT is enabled in CLKULL periods (24MHz), from HFXTCTL.EN until the clock is good for radio operation (amplitude compensation is settled).
15-2	RESERVED	R	0h	Reserved
1	FAULT	R	0h	HFXT clock fault Indicates a lower than expected HFXT frequency. HFXT will not recover from this fault, disabling and re-enabling HFXT is required.
0	GOOD	R	0h	HFXT clock available. The frequency is not necessarily good enough for radio operation.

6.9.26 AMPADCSTAT Register (Offset = E8h) [Reset = 00000000h]

AMPADCSTAT is shown in Table 6-31.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-31. AMPADCSTAT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-25	RESERVED	R	0h	Reserved
24	COMPOUT	R	0h	Internal. Only to be used through TI provided API.
23	RESERVED	R	0h	Reserved
22-16	PEAKRAW	R	0h	Internal. Only to be used through TI provided API.
15-8	PEAK	R	0h	Internal. Only to be used through TI provided API.
7	RESERVED	R	0h	Reserved
6-0	BIAS	R	0h	Internal. Only to be used through TI provided API.

6.9.27 TRACKSTAT Register (Offset = ECh) [Reset = 00000000h]

TRACKSTAT is shown in Table 6-32.

Return to the Summary Table.

HFOSC tracking loop status information

Table 6-32. TRACKSTAT Register Field Descriptions

Bit	Field	Type	Reset	Description
31	LOOPERRVLD	R	0h	Current HFOSC tracking error valid This bit is one if the tracking loop is running and the error value is valid.
30	RESERVED	R	0h	Reserved
29-16	LOOPERR	R	0h	Current HFOSC tracking error
15-13	RESERVED	R	0h	Reserved
12-0	FINETRIM	R	0h	Current HFOSC Fine-trim value This field uses the internal fractional representation (sign, 4 integer bits, 8 fractional bits). The actual trim value applied to the oscillator is delta-sigma modulated 5 bits non-signed (inverted sign bit + integer bits).

6.9.28 AMPSTAT Register (Offset = F0h) [Reset = 00000000h]

AMPSTAT is shown in Table 6-33.

Return to the Summary Table.

HFXT Amplitude Compensation Status

Table 6-33. AMPSTAT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-29	RESERVED	R	0h	Reserved
28-25	STATE	R	0h	Current AMPCOMP FSM state. 0h = FSM in idle state 1h = Starting LDO 2h = Second shutdown state 3h = Injecting HFOSC for fast startup 4h = Transition to HFXTTARG values 5h = Initial amplitude ramping with HFXTINIT values 6h = Amplitude down correction 7h = Post injection settle wait Ah = First shutdown state Ch = TCXO settled state Eh = Amplitude up correction Fh = Settled state
24-18	IDAC	R	0h	Current IDAC control value.
17-14	IREF	R	0h	Current IREF control value.
13-8	Q2CAP	R	0h	Current Q2CAP control value.
7-2	Q1CAP	R	0h	Current Q1CAP control value.
1	CTRLATTARGET	R	0h	HFXT control values match target values. This applies to IREF, Q1CAP, Q2CAP values.
0	AMPGOOD	R	0h	HFXT amplitude good

6.9.29 ATBCTL0 Register (Offset = 100h) [Reset = 00000000h]

ATBCTL0 is shown in Table 6-34.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-34. ATBCTL0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-19	RESERVED	R	0h	Reserved
18-0	SEL	R/W	0h	Internal. Only to be used through TI provided API.

6.9.30 ATBCTL1 Register (Offset = 104h) [Reset = 00000000h]

ATBCTL1 is shown in Table 6-35.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-35. ATBCTL1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-15	RESERVED	R	0h	Reserved
14-13	LFOSC	R/W	0h	Internal. Only to be used through TI provided API.
12	NABIAS	R/W	0h	Internal. Only to be used through TI provided API.
11	RESERVED	R	0h	Reserved
10	LFXT	R/W	0h	Internal. Only to be used through TI provided API.
9-8	LFMON	R/W	0h	Internal. Only to be used through TI provided API.
7	HFXT	R/W	0h	Internal. Only to be used through TI provided API.
6-1	RESERVED	R	0h	Reserved
0	HFOSC	R/W	0h	Internal. Only to be used through TI provided API.

6.9.31 DTBCTL Register (Offset = 108h) [Reset = 00000000h]

DTBCTL is shown in Table 6-36.

Return to the Summary Table.

Digital test bus mux control

Table 6-36. DTBCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-23	RESERVED	R	0h	Reserved
22-18	DSEL2	R/W	0h	Internal. Only to be used through TI provided API.
17-13	DSEL1	R/W	0h	Internal. Only to be used through TI provided API.
12-8	DSEL0	R/W	0h	Internal. Only to be used through TI provided API.
7-4	CLKSEL	R/W	0h	Select clock to output on DTB[0] 0h = Select CLKULL (24 MHz during ACTIVE, 32kHz during STANDBY) 1h = Select CLKSVT (48 MHz) 2h = Select CLKADC (48 MHz) 4h = Select tracking loop reference clock 7h = Select LFCLK (selected by LFCLKSEL.MAIN) Ah = Select HFOSC after qualification Ch = Select HFXT divided by 8 Dh = Select HFXT Eh = Select LFOSC Fh = Select LFXT
3-1	RESERVED	R	0h	Reserved
0	EN	R/W	0h	Enable DTB output

6.9.32 TRIM0 Register (Offset = 110h) [Reset = 00000000h]

TRIM0 is shown in Table 6-37.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-37. TRIM0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-9	RESERVED	R	0h	Reserved
8-5	HFOSC_CAP	R/W	0h	Internal. Only to be used through TI provided API.
4-0	HFOSC_COARSE	R/W	0h	Internal. Only to be used through TI provided API.

6.9.33 TRIM1 Register (Offset = 114h) [Reset = 006F94D6h]

TRIM1 is shown in Table 6-38.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-38. TRIM1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-30	HFXTSLICER	R/W	0h	Internal. Only to be used through TI provided API.
29-28	PEAKIBIAS	R/W	0h	Internal. Only to be used through TI provided API.
27	NABIAS_UDIGLDO	R/W	0h	Internal. Only to be used through TI provided API.
26-24	LDOBW	R/W	0h	Internal. Only to be used through TI provided API.
23-20	LDOFB	R/W	6h	Internal. Only to be used through TI provided API.
19-16	LFDLY	R/W	Fh	Internal. Only to be used through TI provided API.
15	NABIAS_LFOSC	R/W	1h	Internal. Only to be used through TI provided API.
14-8	NABIAS_RES	R/W	14h	Internal. Only to be used through TI provided API.
7-0	LFOSC_CAP	R/W	D6h	Internal. Only to be used through TI provided API.

6.9.34 HFXTINIT Register (Offset = 118h) [Reset = 147F8000h]

HFXTINIT is shown in Table 6-39.

Return to the Summary Table.

Initial values for HFXT ramping

Table 6-39. HFXTINIT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-30	RESERVED	R	0h	Reserved
29-23	AMPTHR	R/W	28h	Amplitude threshold during HFXT ramping
22-16	IDAC	R/W	7Fh	Initial HFXT IDAC current
15-12	IREF	R/W	8h	Initial HFXT IREF current
11-6	Q2CAP	R/W	0h	Initial HFXT Q2 cap trim
5-0	Q1CAP	R/W	0h	Initial HFXT Q1 cap trim

6.9.35 HFXTTARG Register (Offset = 11Ch) [Reset = 54464B6Dh]

HFXTTARG is shown in Table 6-40.

Return to the Summary Table.

Target values for HFXT ramping

Table 6-40. HFXTTARG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-30	AMPHYST	R/W	1h	ADC hysteresis used during IDAC updates. Every AMPCFG1.INTERVAL, IDAC will be regulated - up as long as ADC < AMPTHR - down as long as ADC > AMPTHR+AMPHYST
29-23	AMPTHR	R/W	28h	Minimum HFXT amplitude
22-16	IDAC	R/W	46h	Minimum IDAC current
15-12	IREF	R/W	4h	Target HFXT IREF current
11-6	Q2CAP	R/W	2Dh	Target HFXT Q2 cap trim
5-0	Q1CAP	R/W	2Dh	Target HFXT Q1 cap trim

6.9.36 HFXTDYN Register (Offset = 120h) [Reset = 14464B6Dh]

HFXTDYN is shown in Table 6-41.

Return to the Summary Table.

Alternative target values for HFXT configuration
Software can change these values to dynamically transition the HFXT configuration while HFXT is running.
Set SEL to select the alternative set of target values.

Table 6-41. HFXTDYN Register Field Descriptions

Bit	Field	Type	Reset	Description
31	SEL	R/W	0h	Select the dynamic configuration. Amplitude ramping will always happen using the values in HFXTINIT, and HFXTTARG. Afterwards, this bit can be used to select between HFXTTARG and HFXTDYN. Hardware will ensure a smooth transition of analog control signals. 0h = Select configuration in HFXTTARG. 1h = Select configuration in HFXTDYN.
30	RESERVED	R	0h	Reserved
29-23	AMPTHR	R/W	28h	Minimum HFXT amplitude
22-16	IDAC	R/W	46h	Minimum IDAC current
15-12	IREF	R/W	4h	Target HFXT IREF current
11-6	Q2CAP	R/W	2Dh	Target HFXT Q2 cap trim
5-0	Q1CAP	R/W	2Dh	Target HFXT Q1 cap trim

6.9.37 AMPCFG0 Register (Offset = 124h) [Reset = 00348882h]

AMPCFG0 is shown in Table 6-42.

Return to the Summary Table.

Amplitude Compensation Configuration 0

Table 6-42. AMPCFG0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-28	Q2DLY	R/W	0h	Q2CAP change delay. Number of clock cycles to wait before changing Q2CAP by one step. Clock frequency defined in FSMRATE.
27-24	Q1DLY	R/W	0h	Q1CAP change delay. Number of clock cycles to wait before changing Q1CAP by one step. Clock frequency defined in FSMRATE.
23-20	ADCDLY	R/W	3h	ADC and PEAKDET startup time. Number of clock cycles to wait after enabling the PEAKDET and ADC before the first measurement. Clock frequency defined in FSMRATE.
19-15	LDOSTART	R/W	9h	LDO startup time. Number of clock cycles to bypass the LDO resistors for faster startup. Clock frequency defined in FSMRATE.
14-10	INJWAIT	R/W	2h	Inject HFOSC for faster HFXT startup. This value specifies the number of clock cycles to wait after injection is done. The clock speed is defined in FSMRATE.
9-5	INJTIME	R/W	4h	Inject HFOSC for faster HFXT startup. This value specifies the number of clock cycles the injection is enabled. The clock speed is defined in FSMRATE. Set to 0 to disable injection.
4-0	FSMRATE	R/W	2h	Update rate for the AMPCOMP update rate. Also affects the clock rate for the Amplitude ADC. The update rate is 6MHz / (FSMRATE+1). 0h = 6 MHz 1h = 3 MHz 2h = 2 MHz 5h = 1 MHz Bh = 500 kHz 17h = 250 kHz

6.9.38 AMPCFG1 Register (Offset = 128h) [Reset = 260FF0FFh]

AMPCFG1 is shown in Table 6-43.

Return to the Summary Table.

Amplitude Compensation Configuration 1

Table 6-43. AMPCFG1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-28	IDACDLY	R/W	2h	IDAC change delay. Time to wait before changing IDAC by one step. This time needs to be long enough for the crystal to settle. The number of clock cycles to wait is IDACDLY<<4 + 15. Clock frequency defined in AMPCFG0.FSMRATE.
27-24	IREFDLY	R/W	6h	IREF change delay. Number of clock cycles to wait before changing IREF by one step. Clock frequency defined in AMPCFG0.FSMRATE.
23-12	BIASLT	R/W	FFh	Lifetime of the amplitude ADC bias value. This value specifies the number of adjustment intervals, until the ADC bias value has to be measured again. Set to 0 to disable automatic bias measurements.
11-0	INTERVAL	R/W	FFh	Interval for amplitude adjustments. Set to 0 to disable periodic adjustments. This value specifies the number of clock cycles between adjustments. The clock speed is defined in AMPCFG0.FSMRATE.

6.9.39 LOOPCFG Register (Offset = 12Ch) [Reset = 605E33B3h]

LOOPCFG is shown in Table 6-44.

Return to the Summary Table.

Configuration Register for the Tracking Loop

Table 6-44. LOOPCFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-26	FINETRIM_INIT	R/W	18h	Initial value for the resistor fine trim
25-21	BOOST_TARGET	R/W	2h	Number of error-updates using BOOST values, before using KI/KP
20-18	KP_BOOST	R/W	7h	Proportional loop coefficient during BOOST
17-15	KI_BOOST	R/W	4h	Integral loop coefficient during BOOST
14-10	SETTLED_TARGET	R/W	Ch	Number of updates before HFOSC is considered "settled"
9-6	OOR_LIMIT	R/W	Eh	Out-of-range threshold
5-3	KP	R/W	6h	Proportional loop coefficient
2-0	KI	R/W	3h	Integral loop coefficient

6.9.40 TDCCTL Register (Offset = 200h) [Reset = 00000000h]

TDCCTL is shown in Table 6-45.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-45. TDCCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1-0	CMD	W	0h	Internal. Only to be used through TI provided API.

6.9.41 TDCSTAT Register (Offset = 204h) [Reset = 00000006h]

TDCSTAT is shown in Table 6-46.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-46. TDCSTAT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-10	RESERVED	R	0h	Reserved
9	STOP_BF	R	0h	Internal. Only to be used through TI provided API.
8	START_BF	R	0h	Internal. Only to be used through TI provided API.
7	SAT	R	0h	Internal. Only to be used through TI provided API.
6	DONE	R	0h	Internal. Only to be used through TI provided API.
5-0	STATE	R	6h	Internal. Only to be used through TI provided API.

6.9.42 TDCRESULT Register (Offset = 208h) [Reset = 00000002h]

TDCRESULT is shown in Table 6-47.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-47. TDCRESULT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VALUE	R	2h	Internal. Only to be used through TI provided API.

6.9.43 TDCSATCFG Register (Offset = 20Ch) [Reset = 00000000h]

TDCSATCFG is shown in Table 6-48.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-48. TDCSATCFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-5	RESERVED	R	0h	Reserved
4-0	LIMIT	R/W	0h	Internal. Only to be used through TI provided API.

6.9.44 TDCTRIGSRC Register (Offset = 210h) [Reset = 00000000h]

TDCTRIGSRC is shown in Table 6-49.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-49. TDCTRIGSRC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15	STOP_POL	R/W	0h	Internal. Only to be used through TI provided API.
14-13	RESERVED	R	0h	Reserved
12-8	STOP_SRC	R/W	0h	Internal. Only to be used through TI provided API.
7	START_POL	R/W	0h	Internal. Only to be used through TI provided API.
6-5	RESERVED	R	0h	Reserved
4-0	START_SRC	R/W	0h	Internal. Only to be used through TI provided API.

6.9.45 TDCTRIGCNT Register (Offset = 214h) [Reset = 00000000h]

TDCTRIGCNT is shown in Table 6-50.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-50. TDCTRIGCNT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	CNT	R/W	0h	Internal. Only to be used through TI provided API.

6.9.46 TDCTRIGCNTLOAD Register (Offset = 218h) [Reset = 00000000h]

TDCTRIGCNTLOAD is shown in Table 6-51.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-51. TDCTRIGCNTLOAD Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	CNT	R/W	0h	Internal. Only to be used through TI provided API.

6.9.47 TDCTRIGCNTCFG Register (Offset = 21Ch) [Reset = 00000000h]

TDCTRIGCNTCFG is shown in Table 6-52.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-52. TDCTRIGCNTCFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Reserved
0	EN	R/W	0h	Internal. Only to be used through TI provided API.

6.9.48 TDCPRECTL Register (Offset = 220h) [Reset = 00000000h]

TDCPRECTL is shown in Table 6-53.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-53. TDCPRECTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7	RESET_N	R/W	0h	Internal. Only to be used through TI provided API.
6	RATIO	R/W	0h	Internal. Only to be used through TI provided API.
5	RESERVED	R	0h	Reserved
4-0	SRC	R/W	0h	Internal. Only to be used through TI provided API.

6.9.49 TDCPRECNTR Register (Offset = 224h) [Reset = 00000000h]

TDCPRECNTR is shown in Table 6-54.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-54. TDCPRECNTR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-17	RESERVED	R	0h	Reserved
16	CAPT	W	0h	Internal. Only to be used through TI provided API.
15-0	CNT	R	0h	Internal. Only to be used through TI provided API.

6.9.50 WDTCNT Register (Offset = 300h) [Reset = 00000000h]

WDTCNT is shown in Table 6-55.

Return to the Summary Table.

WDT counter value register

Table 6-55. WDTCNT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R/W	0h	Counter value. A write to this field immediately starts (or restarts) the counter. It will count down from the written value. If the counter reaches 0, a reset will be generated. A write value of 0 immediately generates a reset. This field is only writable if not locked. See WDTLOCK register. Writing this field will automatically activate the lock. A read returns the current value of the counter.

6.9.51 WDTTEST Register (Offset = 304h) [Reset = 00000000h]

WDTTEST is shown in Table 6-56.

Return to the Summary Table.

WDT test mode register

Table 6-56. WDTTEST Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Reserved
0	STALLEN	R/W	0h	WDT stall enable This field is only writable if not locked. See WDTLOCK register. 0h = DISABLE WDT continues counting while the CPU is stopped by a debugger. 1h = ENABLE WDT stops counting while the CPU is stopped by a debugger.

6.9.52 WDTLOCK Register (Offset = 308h) [Reset = 00000001h]

WDTLOCK is shown in Table 6-57.

Return to the Summary Table.

WDT lock register

Table 6-57. WDTLOCK Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	STAT	R/W	1h	A write with value 0x1ACCE551 unlocks the watchdog registers for write access. A write with any other value locks the watchdog registers for write access. Writing the WDTCNT register will also lock the watchdog registers. A read of this field returns the state of the lock (0=unlocked, 1=locked).

6.10 CLKCTL Registers

Table 6-58 lists the memory-mapped registers for the CLKCTL registers. All register offset addresses not listed in Table 6-58 should be considered as reserved locations and the register contents should not be modified.

Table 6-58. CLKCTL Registers

Offset	Acronym	Register Name	Section
0h	DESC	Description Register.	Go
4h	DESCEX0	Extended Description Register 0.	Go
8h	DESCEX1	Extended Description Register 1.	Go
Ch	CLKCFG0	Clock Configuration Register 0.	Go
10h	CLKCFG1	Clock Configuration Register 1.	Go
14h	CLKENSET0	Clock Enable Set Register 0.	Go
18h	CLKENSET1	Clock Enable Set Register 1.	Go
20h	CLKENCLR0	Clock Enable Clear Register 0.	Go
24h	CLKENCLR1	Clock Enable Clear Register 1.	Go
3Ch	STBYPTR	Internal. Only to be used through TI provided API.	Go
48h	IDLECFG	IDLE Configuration Register.	Go

Complex bit access types are encoded to fit into small table cells. Table 6-59 shows the codes that are used for access types in this section.

Table 6-59. CLKCTL Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

6.10.1 DESC Register (Offset = 0h) [Reset = 25480000h]

DESC is shown in Table 6-60.

Return to the Summary Table.

Description Register.
This register provides IP module ID, revision information, instance index and standard MMR registers offset.

Table 6-60. DESC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	MODID	R	2548h	Module identifier used to uniquely identify this IP.
15-12	STDIPOFF	R	0h	Standard IP MMR block offset. Standard IP MMRs are the set of from aggregated IRQ registers till DTB. 0: Standard IP MMRs do not exist 0x1-0xF: Standard IP MMRs begin at offset of (64*STDIPOFF from the base IP address)
11-8	INSTIDX	R	0h	IP Instance ID number. If multiple instances of IP exist in the device, this field can identify the instance number (0-15).
7-4	MAJREV	R	0h	Major revision of IP 0-15
3-0	MINREV	R	0h	Minor revision of IP 0-15.

6.10.2 DESCEX0 Register (Offset = 4h) [Reset = 78034447h]

DESCEX0 is shown in Table 6-61.

Return to the Summary Table.

Extended Description Register 0.
This register shows SVT IP availability, HW features and memory size configuration.

Table 6-61. DESCEX0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	LGPT3	R	1h	IP status on device 0h = IP is unavailable 1h = IP is available
29	LGPT2	R	1h	IP status on device 0h = IP is unavailable 1h = IP is available
28	LGPT1	R	1h	IP status on device 0h = IP is unavailable 1h = IP is available
27	LGPT0	R	1h	IP status on device 0h = IP is unavailable 1h = IP is available
26-18	RESERVED	R	0h	Reserved
17	DMA	R	1h	IP status on device 0h = IP is unavailable 1h = IP is available
16	LAES	R	1h	IP status on device 0h = IP is unavailable 1h = IP is available
15	RESERVED	R	0h	Reserved
14	ADC0	R	1h	IP status on device 0h = IP is unavailable 1h = IP is available
13-11	RESERVED	R	0h	Reserved
10	SPI0	R	1h	IP status on device 0h = IP is unavailable 1h = IP is available
9-7	RESERVED	R	0h	Reserved
6	I2C0	R	1h	IP status on device 0h = IP is unavailable 1h = IP is available
5-3	RESERVED	R	0h	Reserved
2	UART0	R	1h	IP status on device 0h = IP is unavailable 1h = IP is available
1	LRFD	R	1h	IP status on device 0h = IP is unavailable 1h = IP is available
0	GPIO	R	1h	IP status on device 0h = IP is unavailable 1h = IP is available

6.10.3 DESCEX1 Register (Offset = 8h) [Reset = F000FF00h]

DESCEX1 is shown in Table 6-62.

Return to the Summary Table.

Extended Description Register 1.
This register shows SVT IP availability, HW features and memory size configuration.

Table 6-62. DESCEX1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-30	FLASHSZ	R	3h	System flash availability 0h = Flash size set to level 0 (Min size) 1h = Flash size set to level 1 2h = Flash size set to level 2 3h = Flash size set to level 3 (Max size)
29-28	SRAMSZ	R	3h	System SRAM availability 0h = SRAM size set to level 0 (Min size) 1h = SRAM size set to level 1 2h = SRAM size set to level 2 3h = SRAM size set to level 3 (Max size)
27-16	RESERVED	R	0h	Reserved
15-8	ROPT	R	FFh	System radio feature availability FFh = All features available
7-0	RESERVED	R	0h	Reserved

6.10.4 CLKCFG0 Register (Offset = Ch) [Reset = 00000001h]

CLKCFG0 is shown in Table 6-63.

Return to the Summary Table.

Clock Configuration Register 0.
This register shows the IP clock configuration for the system.
The configuration is updated through CLKENSET0 and CLKENCLR0.

Table 6-63. CLKCFG0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	LGPT3	R	0h	IP clock configuration 0h = Clock is disabled 1h = Clock is enabled
29	LGPT2	R	0h	IP clock configuration 0h = Clock is disabled 1h = Clock is enabled
28	LGPT1	R	0h	IP clock configuration 0h = Clock is disabled 1h = Clock is enabled
27	LGPT0	R	0h	IP clock configuration 0h = Clock is disabled 1h = Clock is enabled
26-18	RESERVED	R	0h	Reserved
17	DMA	R	0h	IP clock configuration 0h = Clock is disabled 1h = Clock is enabled
16	LAES	R	0h	IP clock configuration 0h = Clock is disabled 1h = Clock is enabled
15	RESERVED	R	0h	Reserved
14	ADC0	R	0h	IP clock configuration 0h = Clock is disabled 1h = Clock is enabled
13-11	RESERVED	R	0h	Reserved
10	SPI0	R	0h	IP clock configuration 0h = Clock is disabled 1h = Clock is enabled
9-7	RESERVED	R	0h	Reserved
6	I2C0	R	0h	IP clock configuration 0h = Clock is disabled 1h = Clock is enabled
5-3	RESERVED	R	0h	Reserved
2	UART0	R	0h	IP clock configuration 0h = Clock is disabled 1h = Clock is enabled
1	LRFD	R	0h	IP clock configuration 0h = Clock is disabled 1h = Clock is enabled
0	GPIO	R	1h	IP clock configuration 0h = Clock is disabled 1h = Clock is enabled

6.10.5 CLKCFG1 Register (Offset = 10h) [Reset = 00000000h]

CLKCFG1 is shown in Table 6-64.

Return to the Summary Table.

Clock Configuration Register 1.
This register shows the IP clock configuration for the system.
The configuration is updated through CLKENSET1 and CLKENCLR1.

Table 6-64. CLKCFG1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	RESERVED	R	0h	Reserved

6.10.6 CLKENSET0 Register (Offset = 14h) [Reset = 00000000h]

CLKENSET0 is shown in Table 6-65.

Return to the Summary Table.

Clock Enable Set Register 0.
This register enables IP clocks in the system.
Used to set the corresponding fields in CLKCFG0 to 1.

Table 6-65. CLKENSET0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	LGPT3	W	0h	Configure IP clock enable 0h = IP clock enable is unchanged 1h = Set IP clock enable
29	LGPT2	W	0h	Configure IP clock enable 0h = IP clock enable is unchanged 1h = Set IP clock enable
28	LGPT1	W	0h	Configure IP clock enable 0h = IP clock enable is unchanged 1h = Set IP clock enable
27	LGPT0	W	0h	Configure IP clock enable 0h = IP clock enable is unchanged 1h = Set IP clock enable
26-18	RESERVED	R	0h	Reserved
17	DMA	W	0h	Configure IP clock enable 0h = IP clock enable is unchanged 1h = Set IP clock enable
16	LAES	W	0h	Configure IP clock enable 0h = IP clock enable is unchanged 1h = Set IP clock enable
15	RESERVED	R	0h	Reserved
14	ADC0	W	0h	Configure IP clock enable 0h = IP clock enable is unchanged 1h = Set IP clock enable
13-11	RESERVED	R	0h	Reserved
10	SPI0	W	0h	Configure IP clock enable 0h = IP clock enable is unchanged 1h = Set IP clock enable
9-7	RESERVED	R	0h	Reserved
6	I2C0	W	0h	Configure IP clock enable 0h = IP clock enable is unchanged 1h = Set IP clock enable
5-3	RESERVED	R	0h	Reserved
2	UART0	W	0h	Configure IP clock enable 0h = IP clock enable is unchanged 1h = Set IP clock enable
1	LRFD	W	0h	Configure IP clock enable 0h = IP clock enable is unchanged 1h = Set IP clock enable
0	GPIO	W	0h	Configure IP clock enable 0h = IP clock enable is unchanged 1h = Set IP clock enable

6.10.7 CLKENSET1 Register (Offset = 18h) [Reset = 00000000h]

CLKENSET1 is shown in Table 6-66.

Return to the Summary Table.

Clock Enable Set Register 1.
This register enables IP clocks in the system.
Used to set the corresponding fields in CLKCFG1 to 1.

Table 6-66. CLKENSET1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	RESERVED	R	0h	Reserved

6.10.8 CLKENCLR0 Register (Offset = 20h) [Reset = 00000000h]

CLKENCLR0 is shown in Table 6-67.

Return to the Summary Table.

Clock Enable Clear Register 0.
This register disables IP clocks in the system.
Used to clear the corresponding fields in CLKCFG0 to 0.

Table 6-67. CLKENCLR0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	LGPT3	W	0h	Configure IP clock enable 0h = IP clock enable is unchanged 1h = Clear IP clock enable
29	LGPT2	W	0h	Configure IP clock enable 0h = IP clock enable is unchanged 1h = Clear IP clock enable
28	LGPT1	W	0h	Configure IP clock enable 0h = IP clock enable is unchanged 1h = Clear IP clock enable
27	LGPT0	W	0h	Configure IP clock enable 0h = IP clock enable is unchanged 1h = Clear IP clock enable
26-18	RESERVED	R	0h	Reserved
17	DMA	W	0h	Configure IP clock enable 0h = IP clock enable is unchanged 1h = Clear IP clock enable
16	LAES	W	0h	Configure IP clock enable 0h = IP clock enable is unchanged 1h = Clear IP clock enable
15	RESERVED	R	0h	Reserved
14	ADC0	W	0h	Configure IP clock enable 0h = IP clock enable is unchanged 1h = Clear IP clock enable
13-11	RESERVED	R	0h	Reserved
10	SPI0	W	0h	Configure IP clock enable 0h = IP clock enable is unchanged 1h = Clear IP clock enable
9-7	RESERVED	R	0h	Reserved
6	I2C0	W	0h	Configure IP clock enable 0h = IP clock enable is unchanged 1h = Clear IP clock enable
5-3	RESERVED	R	0h	Reserved
2	UART0	W	0h	Configure IP clock enable 0h = IP clock enable is unchanged 1h = Clear IP clock enable
1	LRFD	W	0h	Configure IP clock enable 0h = IP clock enable is unchanged 1h = Clear IP clock enable
0	GPIO	W	0h	Configure IP clock enable 0h = IP clock enable is unchanged 1h = Clear IP clock enable

6.10.9 CLKENCLR1 Register (Offset = 24h) [Reset = 00000000h]

CLKENCLR1 is shown in Table 6-68.

Return to the Summary Table.

Clock Enable Clear Register 1.
This register disables IP clocks in the system.
Used to clear the corresponding fields in CLKCFG1 to 0.

Table 6-68. CLKENCLR1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	RESERVED	R	0h	Reserved

6.10.10 STBYPTR Register (Offset = 3Ch) [Reset = 00000000h]

STBYPTR is shown in Table 6-69.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-69. STBYPTR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R/W	0h	Internal. Only to be used through TI provided API.

6.10.11 IDLECFG Register (Offset = 48h) [Reset = 00000000h]

IDLECFG is shown in Table 6-70.

Return to the Summary Table.

IDLE Configuration Register.
This register contains flash LDO configuration for IDLE mode.

Table 6-70. IDLECFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Reserved
0	MODE	R/W	0h	Flash LDO configuration in SLEEP/IDLE mode. 0h = Flash LDO is on in SLEEP/IDLE mode. Gives fast wake up time from SLEEP/IDLE mode, but increased power consumption. 1h = Flash LDO is off in SLEEP/IDLE mode. Decreases power consumption in SLEEP/IDLE mode, but gives longer wake up time. Note: NVM clock is turned off independent of DMA status. Therefore SW must ensure that DMA never access NVM in this mode.

6.11 PMCTL Registers

Table 6-71 lists the memory-mapped registers for the PMCTL registers. All register offset addresses not listed in Table 6-71 should be considered as reserved locations and the register contents should not be modified.

Table 6-71. PMCTL Registers

Offset	Acronym	Register Name	Section
0h	DESC	Description Register.	Go
4h	DESCEX	Extended Description Register.	Go
8h	SHTDWN	Shutdown Register.	Go
Ch	SLPCTL	Sleep Control Register.	Go
10h	WUSTA	Wakeup Status Register	Go
14h	VDDRCTL	VDDR Control Register.	Go
20h	SYSFSET	Internal. Only to be used through TI provided API.	Go
24h	SYSFCLR	Internal. Only to be used through TI provided API.	Go
28h	SYSFSTA	Internal. Only to be used through TI provided API.	Go
2Ch	RSTCTL	Reset Control Register.	Go
30h	RSTSTA	Reset Status.	Go
34h	BOOTSTA	Internal. Only to be used through TI provided API.	Go
3Ch	AONRSTA1	AON Register Status 1.	Go
40h	AONRSET1	AON Register Set 1.	Go
44h	AONRCLR1	AON Register Clear 1.	Go
64h	ETPP	Internal. Only to be used through TI provided API.	Go
7Ch	RETCFG0	Internal. Only to be used through TI provided API.	Go
80h	RETCFG1	Internal. Only to be used through TI provided API.	Go
84h	RETCFG2	Internal. Only to be used through TI provided API.	Go
88h	RETCFG3	Internal. Only to be used through TI provided API.	Go
8Ch	RETCFG4	Internal. Only to be used through TI provided API.	Go
90h	RETCFG5	Internal. Only to be used through TI provided API.	Go
94h	RETCFG6	Internal. Only to be used through TI provided API.	Go
98h	RETCFG7	Internal. Only to be used through TI provided API.	Go

Complex bit access types are encoded to fit into small table cells. Table 6-72 shows the codes that are used for access types in this section.

Table 6-72. PMCTL Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

6.11.1 DESC Register (Offset = 0h) [Reset = D7410010h]

DESC is shown in Table 6-73.

Return to the Summary Table.

Description Register.
This register provides IP module ID, revision information, instance index and standard MMR registers offset.

Table 6-73. DESC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	MODID	R	D741h	Module identifier used to uniquely identify this IP.
15-12	STDIPOFF	R	0h	Standard IP MMR block offset. Standard IP MMRs are the set of from aggregated IRQ registers till DTB. 0: Standard IP MMRs do not exist 0x1-0xF: Standard IP MMRs begin at offset of (64*STDIPOFF from the base IP address)
11-8	INSTIDX	R	0h	IP Instance ID number. If multiple instances of IP exist in the device, this field can identify the instance number (0-15).
7-4	MAJREV	R	1h	Major revision of IP (0-15).
3-0	MINREV	R	0h	Minor revision of IP (0-15).

6.11.2 DESCEX Register (Offset = 4h) [Reset = FC000000h]

DESCEX is shown in Table 6-74.

Return to the Summary Table.

Extended Description Register.
This register shows ULL IP availability and memory size configuration.

Table 6-74. DESCEX Register Field Descriptions

Bit	Field	Type	Reset	Description
31-30	FLASHSZ	R	3h	System flash availability 0h = Flash size set to level 0 (Min size) 1h = Flash size set to level 1 2h = Flash size set to level 2 3h = Flash size set to level 3 (Max size)
29-28	SRAMSZ	R	3h	System SRAM availability 0h = SRAM size set to level 0 (Min size) 1h = SRAM size set to level 1 2h = SRAM size set to level 2 3h = SRAM size set to level 3 (Max size)
27	TSD	R	1h	TSD (thermal shutdown) IP status on device 0h = IP is unavailable 1h = IP is available
26	LPCMP	R	1h	LPCMP (low power comparator) IP status on device 0h = IP is unavailable 1h = IP is available
25-0	RESERVED	R	0h	Reserved

6.11.3 SHTDWN Register (Offset = 8h) [Reset = 00000000h]

SHTDWN is shown in Table 6-75.

Return to the Summary Table.

Shutdown Register.
This register controls SHUTDOWN mode entry.

Table 6-75. SHTDWN Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	KEY	W	0h	Setting a valid key will trigger the device to enter SHUTDOWN mode. A5A5h = This is the only valid key value that will trigger SHUTDOWN mode. All other values are invalid and will have no effect.

6.11.4 SLPCTL Register (Offset = Ch) [Reset = 00000000h]

SLPCTL is shown in Table 6-76.

Return to the Summary Table.

Sleep Control Register.
This register controls I/O pad sleep mode. When I/O pad sleep mode is enabled all I/O pad outputs and I/O pad configurations are latched. Inputs are transparent if I/O pad is configured as input.

Table 6-76. SLPCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R/W	0h	Reserved
0	SLPN	R/W	0h	The boot code will set this bit field and disable sleep mode, automatically unless waking up from a SHUTDOWN RSTSTA.SDDET is set. Application software must reconfigure the state for all IO's before setting this bit field upon waking up from a SHUTDOWN to avoid glitches on pins. 0h = I/O pad sleep mode is enabled 1h = I/O pad sleep mode is disabled

6.11.5 WUSTA Register (Offset = 10h) [Reset = 00000001h]

WUSTA is shown in Table 6-77.

Return to the Summary Table.

Wakeup Status Register
This register shows the device wakeup source. Used to distinguish between wakeup from STANDBY, SHUTDOWN and reset.

Table 6-77. WUSTA Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1-0	SRC	R	1h	This field shows the device wakeup source. 1h = Wakeup from system reset / SHUTDOWN mode. See RSTSTA for more status information. 2h = Wakeup from STANDBY mode.

6.11.6 VDDRCTL Register (Offset = 14h) [Reset = 00000000h]

VDDRCTL is shown in Table 6-78.

Return to the Summary Table.

VDDR Control Register.
This register contains VDDR regulator settings for the device.

Table 6-78. VDDRCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1	STBY	R/W	0h	Select between continuous or duty-cycled VDDR regulation in STANDBY mode. 0h = Duty-cycled VDDR regulation in STANDBY mode. 1h = Continuous VDDR regulation in STANDBY mode.
0	SELECT	R/W	0h	Select between GLDO and DCDC as VDDR regulator (in ACTIVE, IDLE and STANDBY mode). 0h = GLDO enabled for regulation of VDDR voltage 1h = DCDC enabled for regulation of VDDR voltage

6.11.7 SYSFSET Register (Offset = 20h) [Reset = 00000000h]

SYSFSET is shown in Table 6-79.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-79. SYSFSET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-3	RESERVED	R	0h	Reserved
2	FLAG2	W	0h	Internal. Only to be used through TI provided API.
1	FLAG1	W	0h	Internal. Only to be used through TI provided API.
0	FLAG0	W	0h	Internal. Only to be used through TI provided API.

6.11.8 SYSFCLR Register (Offset = 24h) [Reset = 00000000h]

SYSFCLR is shown in Table 6-80.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-80. SYSFCLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-3	RESERVED	R	0h	Reserved
2	FLAG2	W	0h	Internal. Only to be used through TI provided API.
1	FLAG1	W	0h	Internal. Only to be used through TI provided API.
0	FLAG0	W	0h	Internal. Only to be used through TI provided API.

6.11.9 SYSFSTA Register (Offset = 28h) [Reset = 00000000h]

SYSFSTA is shown in Table 6-81.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-81. SYSFSTA Register Field Descriptions

Bit	Field	Type	Reset	Description
31-3	RESERVED	R	0h	Reserved
2	FLAG2	R	0h	Internal. Only to be used through TI provided API.
1	FLAG1	R	0h	Internal. Only to be used through TI provided API.
0	FLAG0	R	0h	Internal. Only to be used through TI provided API.

6.11.10 RSTCTL Register (Offset = 2Ch) [Reset = 00000000h]

RSTCTL is shown in Table 6-82.

Return to the Summary Table.

Reset Control Register.
This register configures and controls system reset.

Table 6-82. RSTCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-3	RESERVED	R	0h	Reserved
2	LFLOSS	R/W	0h	LF clock loss reset enable. Trigger system reset when LF clock loss is detected, which reset the entire device and causes a reboot of the system. The system reset event is captured as RSTSTA.RESETSRC set to SYSRESET and RSTSTA.SYSSRC set to LFLOSSEV. 0h = LF clock loss detection will not trigger a system reset. 1h = LF clock loss detection will trigger a system reset.
1	TSDEN	R/W	0h	TSD (Thermal Shutdown) enable. TSD will trigger an immediate system reset, which reset the entire device and causes a reboot of the system. The device will be in reset until released by the TSD IP. The system reset event is captured as RSTSTA.TSDEV flag set. 0h = No effect 1h = Temperature shutdown comparator enable. Note: If TSD IP not present, see DESCEX.TSD, enable will have no effect.
0	SYSRST	R/W	0h	Trigger system reset, which will reset the entire device and causes a reboot of the system. The system reset event is captured as RSTSTA.RESETSRC set to SYSRESET and RSTSTA.SYSSRC set to SYSRSTEV. 0h = No effect 1h = Trigger a system reset.

6.11.11 RSTSTA Register (Offset = 30h) [Reset = 00000000h]

RSTSTA is shown in Table 6-83.

Return to the Summary Table.

Reset Status.
This register contains the reset source and SHUTDOWN wakeup source for the system.
Check WUSTA.SRC first to ensure that wakeup from STANDBY is not set.
The capture feature is not rearmed until all of the possible reset sources have been released and the result has been copied to this register.
During the copy and rearm process it is one 24MHz period in which an eventual new system reset will be reported as Power on reset regardless of the root cause.

Table 6-83. RSTSTA Register Field Descriptions

Bit	Field	Type	Reset	Description
31-18	RESERVED	R	0h	Reserved
17	SDDET	R	0h	Wakeup from SHUTDOWN flag. Note: This flag will be cleared when SLPCTL.SLPN is asserted. 0h = Wakeup from SHUTDOWN mode not triggered 1h = Wakeup from SHUTDOWN mode
16	IOWUSD	R	0h	Wakeup from SHUTDOWN on an I/O event flag. Note: This flag will be cleared when SLPCTL.SLPN is asserted. 0h = Wakeup from SHUTDOWN not triggered by an I/O event. 1h = Wakeup from SHUTDOWN triggered by an I/O event.
15-8	RESERVED	R	0h	Reserved
7-4	SYSSRC	R	0h	Shows which reset event that triggered SYSRESET in RESETSRC 0h = LF clock loss event 1h = CPU reset event 2h = CPU LOCKUP event 3h = Watchdog timeout event 4h = System reset event 5h = Serial Wire Debug reset event 6h = Analog FSM timeout event Eh = Analog Error reset event Fh = Digital Error reset event
3	TSDEV	R	0h	System reset triggered by TSD event 0h = TSD event not triggered 1h = System reset triggered by TSD event
2-0	RESETSRC	R	0h	Shows the root cause of the last system reset. More than one reported reset source can have been active during the last system reset, but only the root cause is reported. If reset cause is SYSRESET or PINRESET, the other reset flags must be read to determine actual root cause. 0h = Power on reset 1h = Reset pin. TSD will also trigger a pin reset, so actual root cause is given by TSDEV reset flag status. 2h = Brown out detect on VDDS 4h = Brown out detect on VDDR 6h = Digital system reset. Actual root cause is given by SYSSRC.

6.11.12 BOOTSTA Register (Offset = 34h) [Reset = 00000000h]

BOOTSTA is shown in Table 6-84.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-84. BOOTSTA Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	FLAG	R/W	0h	Internal. Only to be used through TI provided API.

6.11.13 AONRSTA1 Register (Offset = 3Ch) [Reset = 00000000h]

AONRSTA1 is shown in Table 6-85.

Return to the Summary Table.

AON Register Status 1.
This register contains the general purpose AON flags for SW, and is updated through AONRSET1.FLAG and AONRCLR1.FLAG.
The register is only reset on a POR event.

Table 6-85. AONRSTA1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-18	RESERVED	R	0h	Reserved
17-0	FLAG	R	0h	State of the AON register flags

6.11.14 AONRSET1 Register (Offset = 40h) [Reset = 00000000h]

AONRSET1 is shown in Table 6-86.

Return to the Summary Table.

AON Register Set 1.
This register sets the AON flags that can be read through AONRSTA1.FLAG.

Table 6-86. AONRSET1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-18	RESERVED	R/W	0h	Reserved
17-0	FLAG	W	0h	Write 1 to set AONRSTA1.FLAG 0h = No flags changed status 0003FFFFh = Set all flags

6.11.15 AONRCLR1 Register (Offset = 44h) [Reset = 00000000h]

AONRCLR1 is shown in Table 6-87.

Return to the Summary Table.

AON Register Clear 1.
This register clears the AON flags that can be read through AONRSTA1.FLAG.

Table 6-87. AONRCLR1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-18	RESERVED	R/W	0h	Reserved
17-0	FLAG	W	0h	Write 1 to clear AONRSTA1.FLAG 0h = No flags changed status 0003FFFFh = Clear all flags

6.11.16 ETPP Register (Offset = 64h) [Reset = 00000000h]

ETPP is shown in Table 6-88.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-88. ETPP Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	RESERVED	R	0h	Reserved

6.11.17 RETCFG0 Register (Offset = 7Ch) [Reset = 00000001h]

RETCFG0 is shown in Table 6-89.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-89. RETCFG0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Reserved
0	VAL	R/W	1h	Internal. Only to be used through TI provided API.

6.11.18 RETCFG1 Register (Offset = 80h) [Reset = 00000000h]

RETCFG1 is shown in Table 6-90.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-90. RETCFG1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Reserved
0	VAL	R/W	0h	Internal. Only to be used through TI provided API.

6.11.19 RETCFG2 Register (Offset = 84h) [Reset = 00000002h]

RETCFG2 is shown in Table 6-91.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-91. RETCFG2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-3	RESERVED	R	0h	Reserved
2-0	VAL	R/W	2h	Internal. Only to be used through TI provided API.

6.11.20 RETCFG3 Register (Offset = 88h) [Reset = 00000000h]

RETCFG3 is shown in Table 6-92.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-92. RETCFG3 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	RESERVED	R	0h	Reserved

6.11.21 RETCFG4 Register (Offset = 8Ch) [Reset = 00000000h]

RETCFG4 is shown in Table 6-93.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-93. RETCFG4 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	RESERVED	R	0h	Reserved

6.11.22 RETCFG5 Register (Offset = 90h) [Reset = 00000000h]

RETCFG5 is shown in Table 6-94.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-94. RETCFG5 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	RESERVED	R	0h	Reserved

6.11.23 RETCFG6 Register (Offset = 94h) [Reset = 00000000h]

RETCFG6 is shown in Table 6-95.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-95. RETCFG6 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	RESERVED	R	0h	Reserved

6.11.24 RETCFG7 Register (Offset = 98h) [Reset = 00000000h]

RETCFG7 is shown in Table 6-96.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 6-96. RETCFG7 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	RESERVED	R	0h	Reserved

Chapter 7

Internal Memory

This chapter presents the versatile instruction memory system (VIMS) and related memories including flash, SRAM, and bootloader ROM.

7.1 SRAM

The CC23xx provides ultra low leakage system RAM consisting of single-cycle on-chip SRAM. The SRAM supports 48MHz (CLKSVT) single cycle read/write access and power optimization by automatic clock gating when idle.

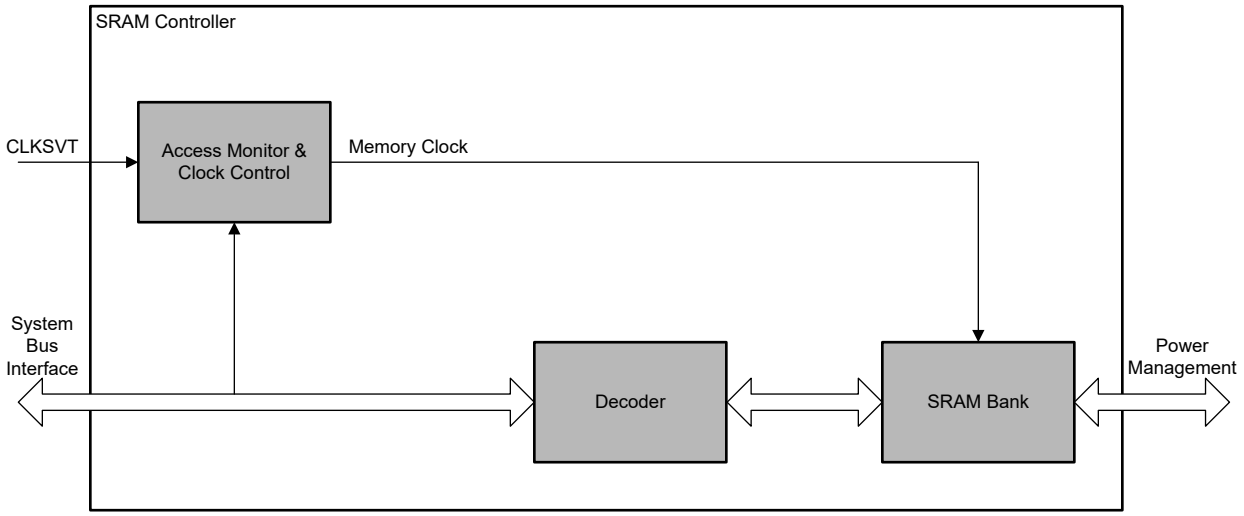


Figure 7-1. SRAM Block Diagram

The entire SRAM is retained in all power modes, except shutdown. There is no configuration option to retain/not-retain certain subsections of the SRAM.

The internal system SRAM is located at address 0x20000000.

7.2 VIMS

7.2.1 Introduction

The main instruction memories are encapsulated in a versatile instruction memory system (VIMS) module, which includes the following memories:

- FLASH/NVM (non-volatile memory)
- Boot ROM

VIMS services the instruction and data fetch requests by the Code Bus and System Bus ports. There is a cache present in VIMS between the Code Bus and FLASH.

7.2.2 Block Diagram

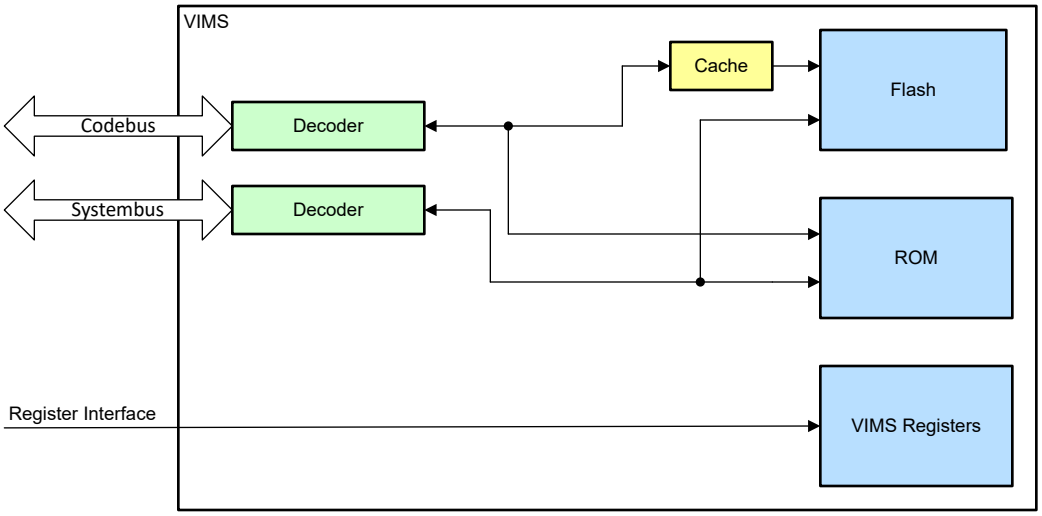


Figure 7-2. VIMS Block Diagram

7.2.3 Cache

The cache is implemented between the Code Bus and the flash. The cache also supports instruction prefetch and branch prediction. The cache control register, VIMS.CCHCTRL, can enable or disable each of these modes. The modes are discussed in the following sections.

7.2.3.1 Basic Cache Mechanism

This is the cache logic with micro-prediction and prefetch disabled. When VIMS receives a 32-bit instruction or data fetch request from CPUSS, VIMS fetches data corresponding to the current address and also the next address, a total of 64 bits of data. This happens whether the cache mechanism is enabled or disabled. When the cache is enabled, the entire 64 bits of data fetched are stored in the line buffer. When the cache is enabled, if the next instruction matches with any one of the instructions stored in the line buffer, the instruction can be served from the cache block itself.

7.2.3.2 Cache Prefetch Mechanism

This is the cache logic with prefetch enabled and micro-prediction disabled.

The prefetch mechanism works by fetching the next 64 bits of data from flash and is activated in two cases. The first case is when there is an idle cycle at CPUSS, there is no pending or new request from CPUSS. The second case is when there is a cache hit.

The prefetch mechanism only works for instruction fetch requests, not data fetch requests. The prefetch mechanism doesn't work if the cache mechanism has marked the present address as a branch address. Note that the cache block can mark only one address as branch address.

Note

If cache is disabled, the prefetch mechanism is disabled automatically.

7.2.3.3 Cache Micro-Prediction Mechanism

This is the cache logic with prefetch and micro-prediction enabled.

When only prefetch is enabled, a branch address is marked when the address is detected. The next time the same branch address is encountered, prefetch is disabled. If both micro-predictor and prefetch are enabled, the branch address is still marked as before but the destination address is also saved. In this case, the next time the same branch address is encountered, the branch address' corresponding destination address is fetched by the prefetch mechanism.

Note

If either cache or prefetch is disabled, micro-prediction is automatically disabled.

7.2.4 Flash

The flash memory consists of a large MAIN region and several smaller regions, including Factory Configuration (FCFG), Customer Configuration (CCFG), TRIM, and ENGR. The FCFG and CCFG regions can also be referred to as NONMAIN regions. See the device-specific data sheet for flash size information.

Each region of the flash memory is organized as a set of 2KB sectors that can be individually erased. Programming the flash by changing bits from 1 to 0 can be done in increments of one 128-bit word.

Erasing a sector causes the entire contents of the sector to be reset to all 1s. The 2KB sectors are paired with sets of other 2KB sectors that can be individually protected by being marked as read-only. Read-only sectors cannot be erased or programmed, which protects the contents of those blocks from being modified. For information on marking blocks as read-only see Section 7.2.4.1

The TRIM and ENGR regions of the flash are locked and inaccessible to the application for program and erase.

7.2.4.1 Flash Read-Only Protection

Flash read-only protection is controlled with the VIMS.WEPRA and VIMS.WEPRB registers. Flash sector size is 2KB so for 512KB of flash there are 256 sectors. Read-only protection has different granularity depending on the sectors being protected.

Each bit in VIMS.WEPRA is responsible for one sector as shown in Table 7-1.

Table 7-1. VIMS.WEPRA Bit to Sector Mapping

WEPRA Bit no.	31	30	...	2	1	0
Protected Sector	31	30	...	2	1	0

Each bit in VIMS.WEPRB is responsible for 8 sectors as shown in Table 7-2.

Table 7-2. VIMS.WEPRB Bit to Sector Mapping

WEPRB Bit no.	27	26	...	2	1	0
Protected Sector	248-255	240-247	...	48-55	40-47	32-39

7.2.4.2 Flash Memory Programming

Memory programming is done using TI-provided API. When calling the API functions, disable all interrupts that trigger access to the flash memory bank being written/erased.

Do not read the flash memory bank being written/erased during a flash memory write or erase operation. If instruction execution is required during a flash memory operation, the executing code must be placed in SRAM (and executed from SRAM) while the flash write/erase operation is in progress.

7.2.5 ROM

Below is an overview of the main ROM functionality. For more details on ROM functions available, see the DriverLib documentation in the CC23xx SDK. For ROM size information, see the device-specific data sheet.

- Life cycle transition support used in TI production
- Device trimming and configuration during boot based on flash configuration contents
- Serial bootloader accessible on UART/SPI
- API functions for flash erase and program operations

The ROM supports a serial bootloader with a SPI or UART interface. For applications that require in-production or in-field programmability, the royalty-free bootloader acts as an application loader and supports firmware updates over a serial bootloader interface. The bootloader either executes automatically if no valid image has

been written to the flash, or the bootloader can be started through a configurable GPIO. The bootloader cannot be called from application code. For more information on the bootloader, see Chapter 8.

7.3 VIMS Registers

Table 7-3 lists the memory-mapped registers for the VIMS registers. All register offset addresses not listed in Table 7-3 should be considered as reserved locations and the register contents should not be modified.

Table 7-3. VIMS Registers

Offset	Acronym	Register Name	Section
0h	DESC	Module Description	Go
4h	DESCEX	Extended Module Description	Go
8h	FLWS1T	Internal. Only to be used through TI provided API.	Go
Ch	FLWS2T	Internal. Only to be used through TI provided API.	Go
18h	PTRMC0	Internal. Only to be used through TI provided API.	Go
1Ch	B0TRMC1	Internal. Only to be used through TI provided API.	Go
20h	B0TRMC0	Internal. Only to be used through TI provided API.	Go
100h	FLBLCK	Internal. Only to be used through TI provided API.	Go
3FCh	CFG	Internal. Only to be used through TI provided API.	Go
410h	WEPRA	Flash main region write/erase protection configuration 1	Go
414h	WEPRB	Flash main region write/erase protection configuration 2	Go
41Ch	WEPRAUX	Flash write/erase protection configuration for other regions	Go
420h	FLBSTAT	Flash status	Go
424h	CCHCTRL	Cache control	Go

Complex bit access types are encoded to fit into small table cells. Table 7-4 shows the codes that are used for access types in this section.

Table 7-4. VIMS Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

7.3.1 DESC Register (Offset = 0h) [Reset = D1400010h]

DESC is shown in Table 7-5.

Return to the Summary Table.

Description Register. This register provides IP module ID, revision information, instance index and standard MMR registers offset.

Table 7-5. DESC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	MODID	R	D140h	Module identifier used to uniquely identify this IP.
15-12	STDIPOFF	R	0h	Standard IP MMR block offset. Standard IP MMRs are the set of from aggregated IRQ registers till DTB. 0: Standard IP MMRs do not exist 0x1-0xF: Standard IP MMRs begin at offset of (64*STDIPOFF from the base IP address)
11-8	INSTIDX	R	0h	IP Instance ID number. If multiple instances of IP exist in the device, this field can identify the instance number (0-15).
7-4	MAJREV	R	1h	Major revision of IP (0-15).
3-0	MINREV	R	0h	Minor revision of IP (0-15).

7.3.2 DESCEX Register (Offset = 4h) [Reset = 08FFB000h]

DESCEX is shown in Table 7-6.

Return to the Summary Table.

Extended Description Register. This register provides configuration details of the IP to software drivers and end users.

Table 7-6. DESCEX Register Field Descriptions

Bit	Field	Type	Reset	Description
31-28	RESERVED	R	0h	Reserved
27	NBANK	R	1h	Provides the FLASH Bank count
26-15	FLSZ	R	1FFh	This provides the total FLASH size in Kilo Bytes. The total FLASH size is (FLSZ + 1)KB
14-0	ROMSZ	R	3000h	Provides the size of ROM in Bytes.

7.3.3 FLWS1T Register (Offset = 8h) [Reset = 00000007h]

FLWS1T is shown in Table 7-7.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 7-7. FLWS1T Register Field Descriptions

Bit	Field	Type	Reset	Description
31-3	RESERVED	R	0h	Reserved
2-0	VAL	R/W	7h	Internal. Only to be used through TI provided API.

7.3.4 FLWS2T Register (Offset = Ch) [Reset = 00000007h]

FLWS2T is shown in Table 7-8.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 7-8. FLWS2T Register Field Descriptions

Bit	Field	Type	Reset	Description
31-3	RESERVED	R	0h	Reserved
2-0	VAL	R/W	7h	Internal. Only to be used through TI provided API.

7.3.5 PTRMC0 Register (Offset = 18h) [Reset = 131A0000h]

PTRMC0 is shown in Table 7-9.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 7-9. PTRMC0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R/W	131A0000h	Internal. Only to be used through TI provided API.

7.3.6 B0TRMC1 Register (Offset = 1Ch) [Reset = 00000000h]

B0TRMC1 is shown in Table 7-10.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 7-10. B0TRMC1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R/W	0h	Internal. Only to be used through TI provided API.

7.3.7 B0TRMC0 Register (Offset = 20h) [Reset = 00000000h]

B0TRMC0 is shown in Table 7-11.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 7-11. B0TRMC0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R/W	0h	Internal. Only to be used through TI provided API.

7.3.8 FLBLCK Register (Offset = 100h) [Reset = 00000000h]

FLBLCK is shown in Table 7-12.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 7-12. FLBLCK Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Reserved
0	VAL	R/W	0h	Internal. Only to be used through TI provided API.

7.3.9 CFG Register (Offset = 3FCh) [Reset = 00000001h]

CFG is shown in Table 7-13.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 7-13. CFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R/W	0h	Reserved
1	TRMVLID	R/W	0h	Internal. Only to be used through TI provided API.
0	WEPRTRM	R/W	1h	Internal. Only to be used through TI provided API.

7.3.10 WEPRA Register (Offset = 410h) [Reset = FFFFFFFFh]

WEPRA is shown in Table 7-14.

Return to the Summary Table.

Flash main region write/erase protection for first 32 sectors. Nth bit corresponds to the Nth sector. This register is sticky when written with value 0.

Table 7-14. WEPRA Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R/W	FFFFFFFFh	Flash write/erase protection configuration value.

7.3.11 WEPRB Register (Offset = 414h) [Reset = 0FFFFFFFh]

WEPRB is shown in Table 7-15.

Return to the Summary Table.

Flash main region write/erase protection for remaining sectors. Each bit corresponds to 8 sectors. Bit 0 corresponds to sector 32-39, bit 1 corresponds to sector 40-47 and so on. This register is sticky when written with value 0.

Table 7-15. WEPRB Register Field Descriptions

Bit	Field	Type	Reset	Description
31-28	RESERVED	R	0h	Reserved
27-0	VAL	R/W	0FFFFFFFh	Flash write/erase protection configuration value.

7.3.12 WEPRAUX Register (Offset = 41Ch) [Reset = 00000007h]

WEPRAUX is shown in Table 7-16.

Return to the Summary Table.

Flash Write/Erase protection for Non-Main, TRIM and ENGR Regions. This register is sticky when written with value 0.

Table 7-16. WEPRAUX Register Field Descriptions

Bit	Field	Type	Reset	Description
31-3	RESERVED	R	0h	Reserved
2	WEPREGR	R/W	1h	Flash engr region write/erase protection configuration value.
1	WEPRTRM	R/W	1h	Flash trim region write/erase protection configuration value.
0	WEPRNMN	R/W	1h	Flash non main region write/erase protection configuration value.

7.3.13 FLBSTAT Register (Offset = 420h) [Reset = 00000000h]

FLBSTAT is shown in Table 7-17.

Return to the Summary Table.

This register is used to indicate status of flash. This register is not retained.

Table 7-17. FLBSTAT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3	PARERR	R	0h	This bit indicates parity error on write/erase and read protection MMRs. This bit is sticky when set to 1 by hardware. 0h = No Error 1h = Error
2	B0BSY	R	0h	This bit indicates if flash is busy. 0h = Idle 1h = Busy
1	B2TRDY	R	0h	This bit indicates if flash is ready in 2T mode. 0h = Not Ready 1h = Ready
0	B1TRDY	R	0h	This bit indicates if flash is ready in 1T mode. 0h = Not Ready 1h = Ready

7.3.14 CCHCTRL Register (Offset = 424h) [Reset = 00000007h]

CCHCTRL is shown in Table 7-18.

Return to the Summary Table.

This register is used for enabling cache, prefetch and micropredictor units.

Table 7-18. CCHCTRL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-3	RESERVED	R	0h	Reserved
2	CCHMPEN	R/W	1h	This bit is used to enable the micropredictor unit. 0h = Disable 1h = Enable
1	CCHPFEN	R/W	1h	This bit is used to enable the prefetch unit. 0h = Disable 1h = Enable
0	CCHEN	R/W	1h	This bit is used to enable the cache. 0h = Disable 1h = Enable

7.4 FLASH Registers

Table 7-19 lists the memory-mapped registers for the FLASH registers. All register offset addresses not listed in Table 7-19 should be considered as reserved locations and the register contents should not be modified.

Table 7-19. FLASH Registers

Offset	Acronym	Register Name	Section
28h	IMASK	Interrupt Mask Register	Go
30h	RIS	Raw Interrupt Status Register	Go
38h	MIS	Masked Interrupt Status Register	Go
40h	ISET	Interrupt Set Register	Go
48h	ICLR	Interrupt Clear Register	Go
FCh	DESC	Hardware Version Description Register	Go
100h	CMDEXEC	Command Execute Register	Go
104h	CMDTYPE	Command Type Register	Go
108h	CMDCTL	Command Control Register	Go
120h	CMDADDR	Command Address Register	Go
124h	CMDBYTEN	Command Program Byte Enable Register	Go
130h	CMDDATA0	Command Data Register 0	Go
134h	CMDDATA1	Command Data Register 1	Go
138h	CMDDATA2	Command Data Register 2	Go
13Ch	CMDDATA3	Command Data Register Bits 127:96	Go
1D0h	CMDWEPROTA	Command Write Erase Protect A Register	Go
1D4h	CMDWEPROTB	Command Write Erase Protect B Register	Go
210h	CMDWEPROTNM	Command Write Erase Protect Non-Main Register	Go
214h	CMDWEPROTTR	Command Write Erase Protect Trim Register	Go
218h	CMDWEPROTEN	Command Write Erase Protect Engr Register	Go
3B0h	CFGCMD	Command Configuration Register	Go
3B4h	CFGPCNT	Pulse Counter Configuration Register	Go
3D0h	STATCMD	Command Status Register	Go
3D4h	STATADDR	Address Status Register	Go
3D8h	STATPCNT	Pulse Count Status Register	Go
3DCh	STATMODE	Mode Status Register	Go
3F0h	GBLINFO0	Global Information Register 0	Go
3F4h	GBLINFO1	Global Information Register 1	Go
3F8h	GBLINFO2	Global Information Register 2	Go
400h	BANK0INFO0	Bank Information Register 0 for Bank 0	Go
404h	BANK0INFO1	Bank Information Register 1 for Bank 0	Go

Complex bit access types are encoded to fit into small table cells. Table 7-20 shows the codes that are used for access types in this section.

Table 7-20. FLASH Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		

Table 7-20. FLASH Access Type Codes (continued)

Access Type	Code	Description
-n		Value after reset or the default value

7.4.1 IMASK Register (Offset = 28h) [Reset = 00000000h]

IMASK is shown in Table 7-21.

Return to the Summary Table.

Interrupt Mask Register:
The IMASK register holds the current interrupt mask settings. Masked interrupts are read in the MIS register. PSD compliant register.

Table 7-21. IMASK Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R/W	0h	Reserved
0	DONE	R/W	0h	Interrupt mask for DONE: 0: Interrupt is disabled in MIS register 1: Interrupt is enabled in MIS register 0h = Interrupt is masked out 1h = Interrupt will request an interrupt service routine and corresponding bit in IPSTANDARD.MIS will be set

7.4.2 RIS Register (Offset = 30h) [Reset = 00000000h]

RIS is shown in Table 7-22.

Return to the Summary Table.

Raw Interrupt Status Register:
The RIS register reflects all pending interrupts, regardless of masking.
The RIS register allows the user to implement a poll scheme. A flag set in this register can be cleared by writing a 1 to the ICLR register bit even if the corresponding IMASK bit is not enabled. A flag can be set by software by writing a 1 to the ISET register. Reading the IIDX register will also clear the corresponding bit in RIS. PSD compliant register.

Table 7-22. RIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Reserved
0	DONE	R	0h	Flash wrapper operation completed. This interrupt bit is set by firmware or the corresponding bit in the ISET register. It is cleared by the corresponding bit in in the ICLR register or reading the IIDX register when this interrupt is the highest priority. 0h = Interrupt did not occur 1h = Interrupt occurred

7.4.3 MIS Register (Offset = 38h) [Reset = 00000000h]

MIS is shown in Table 7-23.

Return to the Summary Table.

Masked Interrupt Status Register:
The MIS register is a bit-wise AND of the contents of the IMASK and RIS registers. This is kept mainly for ARM compatibility, and has limited use since the highest priority interrupt index is returned through the IIDX register.
PSD
compliant register.

Table 7-23. MIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Reserved
0	DONE	R	0h	Flash wrapper operation completed. This masked interrupt bit reflects the bitwise AND of the corresponding RIS and IMASK bits. 0h = Masked interrupt did not occur 1h = Masked interrupt occurred

7.4.4 ISET Register (Offset = 40h) [Reset = 00000000h]

ISET is shown in Table 7-24.

Return to the Summary Table.

Interrupt Set Register:
The ISET register allows software to write a 1 to set corresponding interrupt.
Safety:
This meets a safety requirement to allow software diagnostics to trigger interrupts.
PSD compliant register.

Table 7-24. ISET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	W	0h	Reserved
0	DONE	W	0h	0: No effect 1: Set the DONE interrupt in the RIS register 0h = Writing a 0 has no effect 1h = Set IPSTANDARD.RIS bit

7.4.5 ICLR Register (Offset = 48h) [Reset = 00000000h]

ICLR is shown in Table 7-25.

Return to the Summary Table.

Interrupt Clear Register.
The ICLR register allows allows software to write a 1 to clear corresponding interrupt.
PSD compliant register.

Table 7-25. ICLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	W	0h	Reserved
0	DONE	W	0h	0: No effect 1: Clear the DONE interrupt in the RIS register 0h = Writing a 0 has no effect 1h = Clear IPSTANDARD.RIS bit

7.4.6 DESC Register (Offset = FCh) [Reset = 0B401010h]

DESC is shown in Table 7-26.

Return to the Summary Table.

Hardware Version Description Register:
This register identifies the flash wrapper hardware version and feature set used.

Table 7-26. DESC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	MODULEID	R	B40h	Module ID 0h = Smallest value FFFFh = Highest possible value
15-12	FEATUREVER	R	1h	Feature set 0h = Minimum Value Fh = Maximum Value
11-8	INSTNUM	R	0h	Instance number 0h = Smallest value Fh = Highest possible value
7-4	MAJREV	R	1h	Major Revision 0h = Smallest value Fh = Highest possible value
3-0	MINREV	R	0h	Minor Revision 0h = Smallest value Fh = Highest possible value

7.4.7 CMDEXEC Register (Offset = 100h) [Reset = 00000000h]

CMDEXEC is shown in Table 7-27.

Return to the Summary Table.

Command Execute Register:
Initiates execution of the command specified in the CMDTYPE register.
This register is blocked for writes after being written to 1 and prior to
STATCMD.DONE being set by the flash wrapper hardware.
flash wrapper hardware clears this register after the processing of the command
has completed.

Table 7-27. CMDEXEC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R/W	0h	Reserved
0	VAL	R/W	0h	Command Execute value Initiates execution of the command specified in the CMDTYPE register. 0h = Command will not execute or is not executing in flash wrapper 1h = Command will execute or is executing in flash wrapper

7.4.8 CMDTYPE Register (Offset = 104h) [Reset = 00000000h]

CMDTYPE is shown in Table 7-28.

Return to the Summary Table.

Command Type Register
This register specifies the type of command to be executed by the flash wrapper hardware.
This register is blocked for writes after CMDEXEC is written to a 1 and prior to STATCMD.DONE being set by the hardware to indicate that command execution has completed.

Table 7-28. CMDTYPE Register Field Descriptions

Bit	Field	Type	Reset	Description
31-7	RESERVED	R/W	0h	Reserved
6-4	SIZE	R/W	0h	Command size 0h = Operate on 1 flash word 1h = Operate on 2 flash words 2h = Operate on 4 flash words 3h = Operate on 8 flash words 4h = Operate on a flash sector 5h = Operate on an entire flash bank
3	RESERVED	R/W	0h	Reserved
2-0	COMMAND	R/W	0h	Command type 0h = No Operation 1h = Program 2h = Erase 4h = Mode Change - Perform a mode change only, no other operation. 5h = Clear Status - Clear status bits in FW_SMSTAT only. 6h = Blank Verify - Check whether a flash word is in the erased state. This command may only be used with CMDTYPE.SIZE = ONEWORD

7.4.9 CMDCTL Register (Offset = 108h) [Reset = 00000000h]

CMDCTL is shown in Table 7-29.

Return to the Summary Table.

Command Control Register
This register configures specific capabilities of the state machine for related to the execution of a command.
This register is blocked for writes after CMDEXEC is written to a 1 and prior to STATCMD.DONE being set by the hardware to indicate that command execution has completed.

Table 7-29. CMDCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-22	RESERVED	R/W	0h	Reserved
21	DATAVEREN	R/W	0h	Enable invalid data verify. This checks for 0->1 transitions in the memory when a program operation is initiated. If such a transition is found, the program will fail with an error without executing the program. 0h = Disable 1h = Enable
20	SSERASEDIS	R/W	0h	Disable Stair-Step Erase. If set, the default VHV trim voltage setting will be used for all erase pulses. By default, this bit is reset, meaning that the VHV voltage will be stepped during successive erase pulses. The step count, step voltage, begin and end voltages are all hard-wired. 0h = Enable 1h = Disable
19-17	RESERVED	R	0h	Reserved
16	ADDRXLATEOVR	R/W	0h	Override hardware address translation of address in CMDADDR from a system address to a bank address and bank ID. Use data written to CMDADDR directly as the bank address. Use the value written to CMDCTL.BANKSEL directly as the bank ID. Use the value written to CMDCTL.REGIONSEL directly as the region ID. 0h = Do not override 1h = Override
15-14	RESERVED	R	0h	Reserved
13	RESERVED	R/W	0h	Reserved
12-9	REGIONSEL	R/W	0h	Bank Region A specific region ID can be written to this field to indicate to which region an operation is to be applied if CMDCTL.ADDRXLATEOVR is set. 1h = Main Region 2h = Non-Main Region 4h = Trim Region 8h = Engr Region
8-4	RESERVED	R	0h	Reserved

Table 7-29. CMDCTL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
3-0	MODESEL	R/W	0h	Mode This field is only used for the Mode Change command type. Otherwise, bank and pump modes are set automaticlly through the NW hardware. 0h = Read Mode 2h = Read Margin 0 Mode 4h = Read Margin 1 Mode 6h = Read Margin 0B Mode 7h = Read Margin 1B Mode 9h = Program Verify Mode Ah = Program Single Word Bh = Erase Verify Mode Ch = Erase Sector Eh = Program Multiple Word Fh = Erase Bank

7.4.10 CMDADDR Register (Offset = 120h) [Reset = 00000000h]

CMDADDR is shown in Table 7-30.

Return to the Summary Table.

Command Address Register:
This register forms the target address of a command. The use cases are as follows:
1) For single-word program, this address indicates the flash bank word to be programmed.
2) For multi-word program, this address indicates the first flash bank address for the program. The address will be incremented for further words.
3) For sector erase, this address indicates the sector to be erased.
4) For bank erase, the address indicates the bank to be erased.
Note the address written to this register will be submitted for translation to the flash wrapper address translation interface, and the translated address will be used to access the bank. However, if the CMDCTL.ADDRXLATEOVR bit is set, then the address written to this register will be used directly as the bank address.
This register is blocked for writes after a 1 is written to the CMDEXEC register and prior to STATCMD.DONE being set by the flash wrapper hardware.

Table 7-30. CMDADDR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R/W	0h	Address value 0h = Minimum value of VAL FFFFFFFFh = Maximum value of VAL

7.4.11 CMDBYTEN Register (Offset = 124h) [Reset = 00000000h]

CMDBYTEN is shown in Table 7-31.

Return to the Summary Table.

Command Program Byte Enable Register:
This register forms a per-byte enable for programming data. For data bytes to be programmed, a 1 must be written to the corresponding bit in this register. Normally, all bits are written to 1, allowing program of full flash words. However, leaving some bits 0 allows programming of 8-bit, 16-bit, 32-bit or 64-bit portions of a flash word.
During verify, data bytes read from the flash will not be checked if the corresponding CMDBYTEN bit is 0.
ECC data bytes are protected by the 1-2 MSB bits in this register, depending on the presence of ECC and the flash word data width.
This register is blocked for writes after a 1 is written to the CMDEXEC register and prior to STATCMD.DONE being set by the flash wrapper hardware.
This register is written to all 0 after the completion of all flash wrapper commands.

Table 7-31. CMDBYTEN Register Field Descriptions

Bit	Field	Type	Reset	Description
31-18	RESERVED	R/W	0h	Reserved
17-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Command Byte Enable value. A 1-bit per flash word byte value is placed in this register. 0h = Minimum value of VAL 0003FFFFh = Maximum value of VAL

7.4.12 CMDDATA0 Register (Offset = 130h) [Reset = FFFFFFFFh]

CMDDATA0 is shown in Table 7-32.

Return to the Summary Table.

Command Data Register 0
This register forms the data for a command.
For DATAWIDTH == 128: This register represents bits 31:0 of flash word data register 0.
For DATAWIDTH == 64: This register represents bits 31:0 of flash word data register 0.
This register is blocked for writes after a 1 is written to the CMDEXEC
register and prior to STATCMD.DONE being set by the flash wrapper
hardware.
This register is used to aggregate masking for bits that do not
require additional program pulses during program operations, and will be
written to all 1 after the completion of all flash wrapper commands.
Use cases for the CMDDATA* registers are as follows:
1) Program - These registers contain the data to be programmed.
2) Erase - These registers are not used.

Table 7-32. CMDDATA0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R/W	FFFFFFFFh	A 32-bit data value is placed in this field. 0h = Minimum value of VAL FFFFFFFFh = Maximum value of VAL

7.4.13 CMDDATA1 Register (Offset = 134h) [Reset = FFFFFFFFh]

CMDDATA1 is shown in Table 7-33.

Return to the Summary Table.

Command Data Register 1
This register forms the data for a command.
For DATAWIDTH == 128: This register represents bits 63:32 of flash word data register 0.
For DATAWIDTH == 64: This register represents bits 63:32 of flash word data register 0.
This register is blocked for writes after a 1 is written to the CMDEXEC
register and prior to CMDSTAT.DONE being set by the flash wrapper
hardware.
This register is used to aggregate masking for bits that do not
require additional program pulses during program operations, and will be
written to all 1 after the completion of all flash wrapper commands.
Use cases for the CMDDATA* registers are as follows:
1) Program - These registers contain the data to be programmed.
2) Erase - These registers are not used.

Table 7-33. CMDDATA1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R/W	FFFFFFFFh	A 32-bit data value is placed in this field. 0h = Minimum value of VAL FFFFFFFFh = Maximum value of VAL

7.4.14 CMDDATA2 Register (Offset = 138h) [Reset = FFFFFFFFh]

CMDDATA2 is shown in Table 7-34.

Return to the Summary Table.

Command Data Register 2
This register forms the data for a command.
For DATAWIDTH == 128: This register represents bits 95:64 of flash word data register 0.
For DATAWIDTH == 64: This register represents bits 31:0 of flash word data register 1.
This register is blocked for writes after a 1 is written to the CMDEXEC
register and prior to STATCMD.DONE being set by the flash wrapper
hardware.
This register is used to aggregate masking for bits that do not
require additional program pulses during program operations, and will be
written to all 1 after the completion of all flash wrapper commands.
Use cases for the CMDDATA* registers are as follows:
1) Program - These registers contain the data to be programmed.
2) Erase - These registers are not used.

Table 7-34. CMDDATA2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R/W	FFFFFFFFh	A 32-bit data value is placed in this field. 0h = Minimum value of VAL FFFFFFFFh = Maximum value of VAL

7.4.15 CMDDATA3 Register (Offset = 13Ch) [Reset = FFFFFFFFh]

CMDDATA3 is shown in Table 7-35.

Return to the Summary Table.

Command Data Register 3
This register forms the data for a command.
For DATAWIDTH == 128: This register represents bits 127:96 of flash word data register 0.
For DATAWIDTH == 64: This register represents bits 63:32 of flash word data register 1.
This register is blocked for writes after a 1 is written to the CMDEXEC register and prior to STATCMD.DONE being set by the flash wrapper hardware.
This register is used to aggregate masking for bits that do not require additional program pulses during program operations, and will be written to all 1 after the completion of all flash wrapper commands.
Use cases for the CMDDATA* registers are as follows:
1) Program - These registers contain the data to be programmed.
2) Erase - These registers are not used.

Table 7-35. CMDDATA3 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R/W	FFFFFFFFh	A 32-bit data value is placed in this field. 0h = Minimum value of VAL FFFFFFFFh = Maximum value of VAL

7.4.16 CMDWEPROTA Register (Offset = 1D0h) [Reset = FFFFFFFFh]

CMDWEPROTA is shown in Table 7-36.

Return to the Summary Table.

Command WriteErase Protect A Register
This register allows the first 32 sectors of the main region to be protected from program or erase, with 1 bit protecting each sector. If the main region size is smaller than 32 sectors, then this register provides protection for the whole region.
This register is blocked for writes after a 1 is written to the CMDEXEC register and prior to STATCMD.DONE being set by the flash wrapper hardware.
In addition, this register is used to aggregate masking for sectors that do not require additional erase pulses during bank erase operations, and will be written to all 1 after the completion of all flash wrapper commands.

Table 7-36. CMDWEPROTA Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R/W	FFFFFFFFh	Each bit protects 1 sector. bit [0]: When 1, sector 0 of the flash memory will be protected from program and erase. bit [1]: When 1, sector 1 of the flash memory will be protected from program and erase. : : bit [31]: When 1, sector 31 of the flash memory will be protected from program and erase. 0h = Minimum value of VAL FFFFFFFFh = Maximum value of VAL

7.4.17 CMDWEPROTB Register (Offset = 1D4h) [Reset = 0FFFFFFFh]

CMDWEPROTB is shown in Table 7-37.

Return to the Summary Table.

Command WriteErase Protect B Register
This register allows main region sectors to be protected from program and erase. Each bit corresponds to a group of 8 sectors.
There are 3 cases for how these protect bits are applied:
1. Single-bank system:
In the case where only a single flash bank is present, the first 32 sectors are protected via the CMDWEPROTA register. Thus, the protection give by the bits in CMDWEPROTB begin with sector 32.
2. Multi-bank system, Bank 0:
When multiple flash banks are present, the first 32 sectors of bank 0 are protected via the CMDWEPROTA register. Thus, only bits 4 and above of CMDWEPROTB would be applicable to bank 0. The protection of bit 4 and above would begin at sector 32. Bits 3:0 of WEPROTB are ignored for bank 0.
3. Multi-bank system, Banks 1-N:
For banks other than bank 0 in a multi-bank system, CMDWEPROTA has no effect, so the bits in CMDWEPROTB will protect these banks starting from sector 0.
This register is blocked for writes after a 1 is written to the CMDEXEC register and prior to STATCMD.DONE being set by the flash wrapper hardware.
In addition, this register is used to aggregate masking for sectors that do not require additional erase pulses during bank erase operations, and will be written to all 1 after the completion of all flash wrapper commands.

Table 7-37. CMDWEPROTB Register Field Descriptions

Bit	Field	Type	Reset	Description
31-28	RESERVED	R	0h	Reserved
27-0	VAL	R/W	0FFFFFFFh	Each bit protects a group of 8 sectors. When a bit is 1, the associated 8 sectors in the flash will be protected from program and erase. A maximum of 256 sectors can be protected with this register. 0h = Minimum value of VAL FFFFFFFFh = Maximum value of VAL

7.4.18 CMDWEPROTNM Register (Offset = 210h) [Reset = 00000001h]

CMDWEPROTNM is shown in Table 7-38.

Return to the Summary Table.

Command WriteErase Protect Non-Main
Register
This register allows non-main region region sectors to be protected
from program and erase. Each bit corresponds to 1 sector.
This register is blocked for writes after a 1 is written to the CMDEXEC
register and prior to STATCMD.DONE being set by the flash wrapper
hardware.
In addition, this register is used to aggregate masking for sectors that do not
require additional erase pulses during bank erase operations, and will be
written to all 1 after the completion of all flash wrapper commands.

Table 7-38. CMDWEPROTNM Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Reserved
0	VAL	R/W	1h	Each bit protects 1 sector. bit [0]: When 1, sector 0 of the non-main region will be protected from program and erase. bit [1]: When 1, sector 1 of the non-main region will be protected from program and erase. : : bit [31]: When 1, sector 31 of the non-main will be protected from program and erase. 0h = Minimum value of VAL FFFFFFFFh = Maximum value of VAL

7.4.19 CMDWEPROTTR Register (Offset = 214h) [Reset = 00000001h]

CMDWEPROTTR is shown in Table 7-39.

Return to the Summary Table.

Command WriteErase Protect Trim
Register
This register allows trim region sectors to be protected
from program and erase. Each bit corresponds to 1 sector.
This register is blocked for writes after a 1 is written to the CMDEXEC
register and prior to STATCMD.DONE being set by the flash wrapper
hardware.
In addition, this register is used to aggregate masking for sectors that do not
require additional erase pulses during bank erase operations, and will be
written to all 1 after the completion of all flash wrapper commands.

Table 7-39. CMDWEPROTTR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Reserved
0	VAL	R/W	1h	Each bit protects 1 sector. bit [0]: When 1, sector 0 of the engr region will be protected from program and erase. bit [1]: When 1, sector 1 of the engr region will be protected from program and erase. : : bit [31]: When 1, sector 31 of the engr region will be protected from program and erase. 0h = Minimum value of VAL FFFFFFFFh = Maximum value of VAL

7.4.20 CMDWEPROTEN Register (Offset = 218h) [Reset = 00000001h]

CMDWEPROTEN is shown in Table 7-40.

Return to the Summary Table.

Command WriteErase Protect Engr
Register
This register allows engr region sectors to be protected
from program and erase. Each bit corresponds to 1 sector.
This register is blocked for writes after a 1 is written to the CMDEXEC
register and prior to STATCMD.DONE being set by the flash wrapper
hardware.
In addition, this register is used to aggregate masking for sectors that do not
require additional erase pulses during bank erase operations, and will be
written to all 1 after the completion of all flash wrapper commands.

Table 7-40. CMDWEPROTEN Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Reserved
0	VAL	R/W	1h	Each bit protects 1 sector. bit [0]: When 1, sector 0 of the engr region will be protected from program and erase. bit [1]: When 1, sector 1 of the engr region will be protected from program and erase. : : bit [31]: When 1, sector 31 of the engr region will be protected from program and erase. 0h = Minimum value of VAL FFFFFFFFh = Maximum value of VAL

7.4.21 CFGCMD Register (Offset = 3B0h) [Reset = 00000002h]

CFGCMD is shown in Table 7-41.

Return to the Summary Table.

Command Configuration Register
This register configures specific capabilities of the state machine for related to the execution of a command.
This register is blocked for writes after CMDEXEC is written to a 1 and prior to STATCMD.DONE being set by the hardware to indicate that command execution has completed.

Table 7-41. CFGCMD Register Field Descriptions

Bit	Field	Type	Reset	Description
31-7	RESERVED	R/W	0h	Reserved
6-4	RESERVED	R	0h	Reserved
3-0	WAITSTATE	R/W	2h	Wait State setting for verify reads 0h = Minimum value Fh = Maximum value

7.4.22 CFGPCNT Register (Offset = 3B4h) [Reset = 00000000h]

CFGPCNT is shown in Table 7-42.

Return to the Summary Table.

Pulse Counter Configuration Register
This register allows further configuration of maximum pulse counts for program and erase operations.
This register is blocked for writes after a 1 is written to the CMDEXEC register and prior to STATCMD.DONE being set by the flash wrapper hardware.

Table 7-42. CFGPCNT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-20	RESERVED	R	0h	Reserved
19-17	RESERVED	R/W	0h	Reserved
16	RESERVED	R	0h	Reserved
15-12	RESERVED	R/W	0h	Reserved
11-4	MAXPCNTVAL	R/W	0h	Override maximum pulse counter with this value. If MAXPCNTOVR = 0, then this field is ignored. If MAXPCNTOVR = 1 and MAXERSPCNTOVR = 0, then this value will be used to override the max pulse count for both program and erase. Full max value will be {4'h0, MAXPCNTVAL} . If MAXPCNTOVR = 1 and MAXERSPCNTOVR = 1, then this value will be used to override the max pulse count for program only. Full max value will be {4'h0, MAXPCNTVAL}. 0h = Minimum value FFh = Maximum value
3-1	RESERVED	R/W	0h	Reserved
0	MAXPCNTOVR	R/W	0h	Override hard-wired maximum pulse count. If MAXERSPCNTOVR is not set, then setting this value alone will override the max pulse count for both program and erase. If MAXERSPCNTOVR is set, then this bit will only control the max pulse count setting for program. By default, this bit is 0, and a hard-wired max pulse count is used. 0h = Use hard-wired (default) value for maximum pulse count 1h = Use value from MAXPCNTVAL field as maximum puse count

7.4.23 STATCMD Register (Offset = 3D0h) [Reset = 00000000h]

STATCMD is shown in Table 7-43.

Return to the Summary Table.

Command Status Register
This register contains status regarding completion and errors of command execution.

Table 7-43. STATCMD Register Field Descriptions

Bit	Field	Type	Reset	Description
31-13	RESERVED	R	0h	Reserved
12	FAILMISC	R	0h	Command failed due to error other than write/erase protect violation or verify error. This is an extra bit in case a new failure mechanism is added which requires a status bit. 0h = No Fail 1h = Fail
11-9	RESERVED	R	0h	Reserved
8	FAILINVDATA	R	0h	Program command failed because an attempt was made to program a stored 0 value to a 1. 0h = No Fail 1h = Fail
7	FAILMODE	R	0h	Command failed because a bank has been set to a mode other than READ. Program and Erase commands cannot be initiated unless all banks are in READ mode. 0h = No Fail 1h = Fail
6	FAILILLADDR	R	0h	Command failed due to the use of an illegal address 0h = No Fail 1h = Fail
5	FAILVERIFY	R	0h	Command failed due to verify error 0h = No Fail 1h = Fail
4	FAILWEPROT	R	0h	Command failed due to Write/Erase Protect Sector Violation 0h = No Fail 1h = Fail
3	RESERVED	R	0h	Reserved
2	CMDINPROGRESS	R	0h	Command In Progress 0h = Complete 1h = In Progress
1	CMDPASS	R	0h	Command Pass - valid when CMD_DONE field is 1 0h = Fail 1h = Pass
0	CMDDONE	R	0h	Command Done 0h = Not Done 1h = Done

7.4.24 STATADDR Register (Offset = 3D4h) [Reset = 00210000h]

STATADDR is shown in Table 7-44.

Return to the Summary Table.

Current Address Counter Value
Read only register giving read access to the state machine current address.
A bank id, region id and address are stored in this register and are incremented as necessary during execution of a command.

Table 7-44. STATADDR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-26	RESERVED	R	0h	Reserved
25-21	BANKID	R	1h	Current Bank ID A bank indicator is stored in this register which represents the current bank on which the state machine is operating. There is 1 bit per bank. 1h = Bank 0 2h = Bank 1 4h = Bank 2 8h = Bank 3 10h = Bank 4
20-16	REGIONID	R	1h	Current Region ID A region indicator is stored in this register which represents the current flash region on which the state machine is operating. 1h = Main Region 2h = Non-Main Region 4h = Trim Region 8h = Engr Region
15-0	BANKADDR	R	0h	Current Bank Address A bank offset address is stored in this register. 0h = Minimum value FFFFh = Maximum value

7.4.25 STATPCNT Register (Offset = 3D8h) [Reset = 00000000h]

STATPCNT is shown in Table 7-45.

Return to the Summary Table.

Current Pulse Count Register:
Read only register giving read access to the state machine current pulse count value for program/erase operations.

Table 7-45. STATPCNT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-12	RESERVED	R	0h	Reserved
11-0	PULSECNT	R	0h	Current Pulse Counter Value 0h = Minimum value FFFh = Maximum value

7.4.26 STATMODE Register (Offset = 3DCh) [Reset = 00000000h]

STATMODE is shown in Table 7-46.

Return to the Summary Table.

Mode Status Register
Indicates one or more banks which not in READ mode, and it indicates the mode which the bank(s) are in.

Table 7-46. STATMODE Register Field Descriptions

Bit	Field	Type	Reset	Description
31-18	RESERVED	R	0h	Reserved
17	BANK1TRDY	R	0h	Bank 1T Ready. Bank(s) are ready for 1T access. This is accomplished when the bank and pump have been trimmed. 0h = Not ready 1h = Ready
16	BANK2TRDY	R	0h	Bank 2T Ready. Bank(s) are ready for 2T access. This is accomplished when the pump has fully driven power rails to the bank(s). 0h = Not ready 1h = Ready
15-12	RESERVED	R	0h	Reserved
11-8	BANKMODE	R	0h	Indicates mode of bank(s) that are not in READ mode 0h = Read Mode 2h = Read Margin 0 Mode 4h = Read Margin 1 Mode 6h = Read Margin 0B Mode 7h = Read Margin 1B Mode 9h = Program Verify Mode Ah = Program Single Word Bh = Erase Verify Mode Ch = Erase Sector Eh = Program Multiple Word Fh = Erase Bank
7-5	RESERVED	R	0h	Reserved
4-1	RESERVED	R	0h	Reserved
0	BANKNOTINRD	R	0h	Bank not in read mode. Indicates which banks are not in READ mode. There is 1 bit per bank. 1h = Bank 0 2h = Bank 1 4h = Bank 2 8h = Bank 3 10h = Bank 4

7.4.27 GBLINFO0 Register (Offset = 3F0h) [Reset = 00010800h]

GBLINFO0 is shown in Table 7-47.

Return to the Summary Table.

Global Info 0 Register
Read only register detailing information about sector size and number of banks present.

Table 7-47. GBLINFO0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-19	RESERVED	R	0h	Reserved
18-16	NUMBANKS	R	1h	Number of banks instantiated Minimum: 1 Maximum: 5 1h = Minimum value 5h = Maximum value
15-0	SECTORSIZE	R	800h	Sector size in bytes 400h = Sector size is ONEKB 800h = Sector size is TWOKB

7.4.28 GBLINFO1 Register (Offset = 3F4h) [Reset = 00040080h]

GBLINFO1 is shown in Table 7-48.

Return to the Summary Table.

Global Info 1 Register
Read only register detailing information about data, ecc and redundant data widths in bits.

Table 7-48. GBLINFO1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-19	RESERVED	R	0h	Reserved
18-16	REDWIDTH	R	4h	Redundant data width in bits 0h = Redundant data width is 0. Redundancy/Repair not present. 2h = Redundant data width is 2 bits 4h = Redundant data width is 4 bits
15-13	RESERVED	R	0h	Reserved
12-8	ECCWIDTH	R	0h	ECC data width in bits 0h = ECC data width is 0. ECC not used. 8h = ECC data width is 8 bits 10h = ECC data width is 16 bits
7-0	DATAWIDTH	R	80h	Data width in bits 40h = Data width is 64 bits 80h = Data width is 128 bits

7.4.29 GBLINFO2 Register (Offset = 3F8h) [Reset = 00000001h]

GBLINFO2 is shown in Table 7-49.

Return to the Summary Table.

Global Info 2 Register
Read only register detailing information about the number of data registers present.

Table 7-49. GBLINFO2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3-0	DATAREGISTERS	R	1h	Number of data registers present. 1h = Minimum value of DATAREGISTERS 8h = Maximum value of DATAREGISTERS

7.4.30 BANK0INFO0 Register (Offset = 400h) [Reset = 00000100h]

BANK0INFO0 is shown in Table 7-50.

Return to the Summary Table.

Bank Info 0 Register for bank 0.
Read only register detailing information about Main region size in the bank.

Table 7-50. BANK0INFO0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-12	RESERVED	R	0h	Reserved
11-0	MAINSIZE	R	100h	Main region size in sectors Minimum: 0x8 (8) Maximum: 0x200 (512) 8h = Minimum value of MAINSIZE 200h = Maximum value of MAINSIZE

7.4.31 BANK0INFO1 Register (Offset = 404h) [Reset = 00010101h]

BANK0INFO1 is shown in Table 7-51.

Return to the Summary Table.

Bank Info1 Register for bank 0.
Read only register detailing information about Non-Main, Trim, and Engr region sizes in the bank.

Table 7-51. BANK0INFO1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	RESERVED	R	0h	Reserved
23-16	ENGRSIZE	R	1h	Engr region size in sectors Minimum: 0x0 (0) Maximum: 0x10 (16) 0h = Minimum value of ENGRSIZE 20h = Maximum value of ENGRSIZE
15-8	TRIMSIZE	R	1h	Trim region size in sectors Minimum: 0x0 (0) Maximum: 0x10 (16) 0h = Minimum value of TRIMSIZE 20h = Maximum value of TRIMSIZE
7-0	NONMAINSIZE	R	1h	Non-main region size in sectors Minimum: 0x0 (0) Maximum: 0x10 (16) 0h = Minimum value of NONMAINSIZE 20h = Maximum value of NONMAINSIZE

Chapter 8

Device Boot and Bootloader

This section describes the device boot flow coming out of reset state and device bootloader options supported in system ROM.

8.1 Device Boot and Programming

Device boot is the process that happens after a device reset and before the first instruction of a user application is executed. This process runs out of code in ROM memory and performs the following tasks:

- Required hardware trim values are copied to hardware registers
- SRAM repair information is applied
- Various permissions and restrictions are applied as defined by FCFG (factory configuration) and CCFG (customer configuration)
- SACI, a device management command interface that uses the Serial wire debug (SWD) secure access point (SECAP) mailbox for communication can be entered. It can be used for:
  - Obtaining device information
  - Device lifecycle management
  - Flash programming / validation
  - Debug authentication
  - Entry into various test modes
- A bootloader can be invoked before application entry, either by:
  - A user-defined bootloader, typically implementing a custom method of performing firmware updates
  - A TI-provided simple serial bootloader in ROM intended to allow programming of blank devices through SPI or UART

Note

The serial ROM bootloader can be used for updating on-chip flash with the SPI or UART interface after production. However, the ROM serial bootloader doesn't support security functions and a separate secure boot or secure FW update software should be used for validating the new FW images updated by the ROM serial bootloader.

There are multiple ways to reset the device and the PMCTL.RSTSTA register reports which type of reset has occurred to boot code (and later to bootloader and application) see Table 8-1.

Table 8-1. Reset Causes

RSTSTA[7:4]SYSSRC	RSTSTA[3]TSDEV	RSTSTA[2:0]RESETSRC	Reset Type	Description	Memory/state retention
x	0	0	Power-on reset	Power on reset circuit has released due to supply voltage VDDS above threshold. Seen when power first applied	None
x	0	1	Pin reset	External reset pin (active low) has been released	Retained SRAM unreliable REG3V3 registers retained
x	1	1	Thermal shutdown	Thermal shutdown reset has released due to temperature being below threshold	None
x	0	2	VDDS brownout	VDDS brownout detector has reset device	None
x	0	4	VDDR brownout	VDDR brownout detector has reset device	Retained SRAM unreliable REG3V3 registers retained

Table 8-1. Reset Causes (continued)

RSTSTA[7:4]SYSSRC	RSTSTA[3]TSDEV	RSTSTA[2:0]RESETSRC	Reset Type	Description	Memory/state retention
0	0	6	LF clock loss	LF clock stopped running while in standby power state.	SRAM retained [1] REG3V3 registers retained
1	0	6	CPU-requested reset	System reset requested through ARM core's AIRCR.SYSRESETREQ flag	SRAM retained [1] REG3V3 registers retained
2	0	6	CPU Lockup	ARM core went into lockup state which triggered system reset	SRAM retained [1] REG3V3 registers retained
3	0	6	Watchdog timeout	Watchdog timer timeout occurred and triggered system reset	SRAM retained [1] REG3V3 registers retained
4	0	6	System reset request	System reset requested through PMCTL.RSTCTL.SYSRST flag	SRAM retained [1] REG3V3 registers retained
5	0	6	Serial Wire Debug reset request	System reset requested by external debug probe by writing value 0x5C to SWD:SECAP.RSTCTL.RST	SRAM retained [1] REG3V3 registers retained
14	0	6	Analog error	TI internal use: should not occur on production devices	None
15	0	6	Digital error	TI internal use: should not occur on production devices	None

(1) SRAM is not normally erased during boot. The contents of SRAM can remain across boots except for the range 0x20000000-0x20000447, which is modified during normal boot. If SACI is entered during boot the entire SRAM is cleared.

8.1.1 Boot Flow

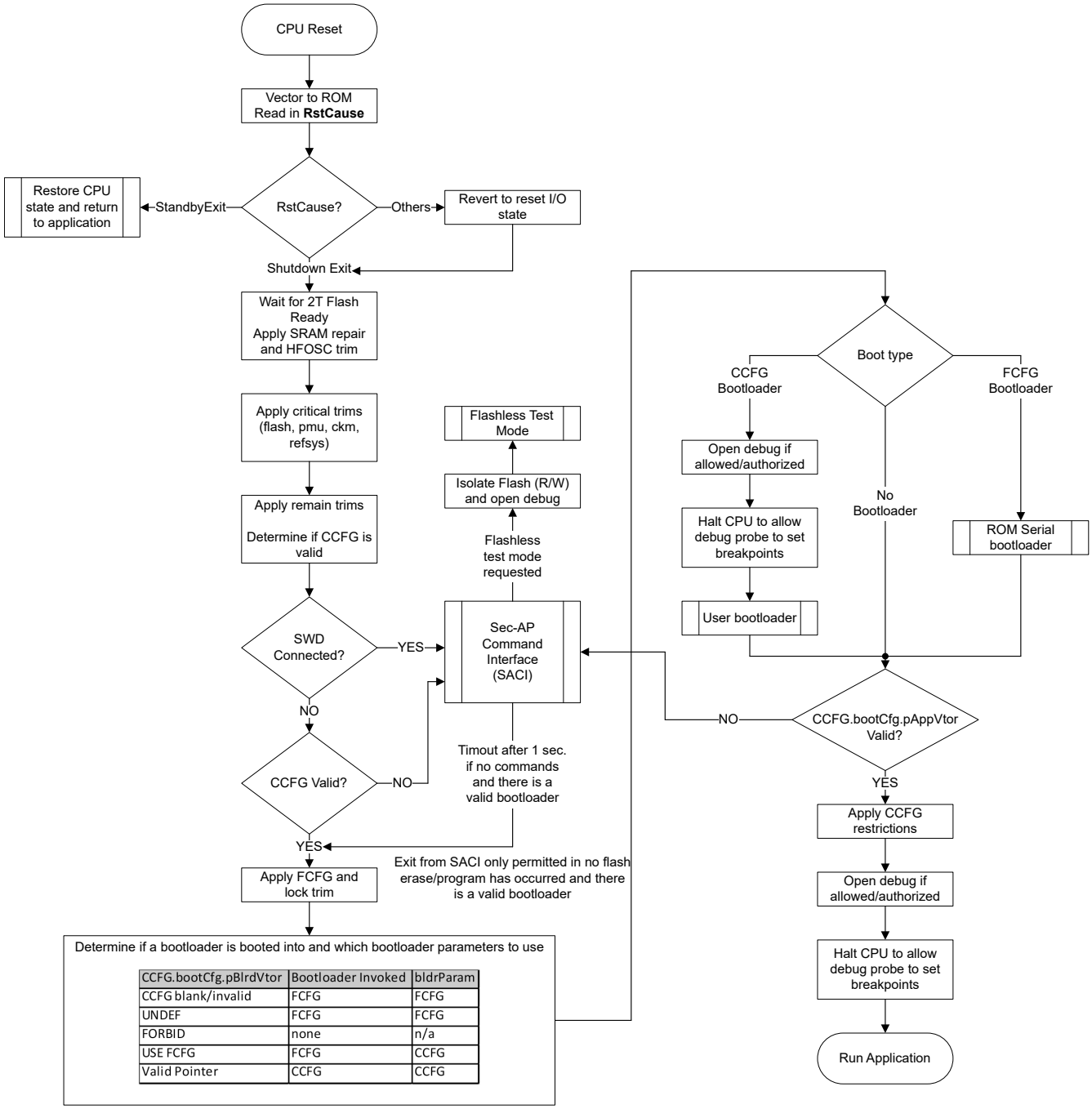


Figure 8-1. Boot Flow

8.1.2 Boot Timing

After reset is released, a normal boot (without SACI invocation and using ROM serial bootloader) typically takes 260 µs. Refer to the data sheet for worst-case boot times across conditions.

8.1.3 Boot Status

An 8-bit value BOOTSTA is updated in the PMCTL.BOOTSTA register during boot to inform about which stage of boot is ongoing or to report errors. A bootloader can use the same mechanism and the mechanism is even

available for user application. BOOTSTA can be read from PMCTL.BOOTSTA using CPU or an external debug probe can read BOOTSTA through SWD:CFGAP.DEVICESTATUS.BOOTSTA.

The BOOTSTA[7:6] bits are sticky, meaning the bits cannot be cleared once set:

BOOTSTA[7:6]	Description
0b00	In boot flow, prior to bootloader invocation. BOOTSTA[5:0] indicates phase of boot or failures (highest numerical values)
0b01	In bootloader. BOOTSTA[5:0] indicates phase of bootloader or failures (highest numerical values)
0b11	Post bootloader boot flow or in application. Some of the lower values of BOOTSTA[5:0] and highest values of BOOTSTA[5:0] are reserved for reporting boot flow phase or failures. The remaining values can be used by an application to report its state in a similar way, and are accessible from SWD:CFGAP even if debugging is not allowed.

The BOOTSTA values used by boot flow and the ROM serial bootloader are given in Table 8-2.

Table 8-2. BOOTSTA Values

BOOTSTA	Set by	Name (waiting for external action) (failure reported; reset device)	Description
0x00	Hardware	BOOT_RESET	Initial value after system reset
0x01	Boot flow	BOOT_COLD_BOOT	Determined that this is not a standby exit scenario and normal boot process starts
0x02	Boot flow	BOOT_SRAM_REP_DONE	SRAM repair has completed; continuing with remaining critical trimming
0x03	Boot flow	BOOT_GENERAL_TRIMS	Critical trimming (flash, oscillators, references, power management) completed, starting general trimming
0x04-0x01F	-	*Reserved*	
0x20	Boot flow	BOOT_ENTERED_SACI	SACI entered (SWD secure access point command interface). If SACI timeout is configured the timer has started running
0x21-0x35	-	*Reserved*	
0x36	Boot flow	BOOT_WAIT_SWD_DISCONNECT	SACI reset command issued with option waitForSwdDisconnect set has been issued. In busy loop until SWD disconnect sequence received by Icemelter
0x37	Boot flow	BOOT_EXITED_SACI	Exit from SACI (due to timeout or SACI exit command)
0x38	Boot flow	BOOT_WAITLOOP_DBGPROBE	TI Internal Test Mode
0x3E	Boot flow	BOOT_FAIL_SRAM_REPAIR	SRAM repair failed: device cannot boot. Remain in infinite loop until reset
0x3F	Boot flow	BOOT_FAULT_HANDLER	Uncaught/unspecified error during boot: device cannot boot. Remain in infinite loop until reset
0x40-0x7F	-	*Cannot be used*	
0x80	Boot flow	BLDR_MODE_ENTRY	Entering bootloader mode; checking which bootloader, if any, to invoke next
0x81	Boot flow	BLDR_WAITLOOP_DBGPROBE	Debug of bootloader allowed; busy loop to allow external probe to set breakpoints before single-stepping or running CPU
0x82-0xB9	-	*Reserved*	
0xBA	Serial ROM bootloader	BLDR_STARTED	Serial ROM bootloader entered

Table 8-2. BOOTSTA Values (continued)

BOOTSTA	Set by	Name (waiting for external action) (failure reported; reset device)	Description
0xBB	Serial ROM bootloader	BLDR_CMD_IDLE	Serial boot ROM has triggered and then either UART or SPI interface was selected (through interface activity); currently bootloader is waiting for a command over selected interface.
0xBC	Serial ROM bootloader	BLDR_CMD_PROCESSING	Serial boot ROM has triggered and then either UART or SPI interface was selected (through interface activity); currently bootloader is in the process of receiving or processing a command
0xBD	Serial ROM bootloader	BLDR_FAIL_EXECUTION_CONTEXT	Failure: serial ROM bootloader entry function was invoked from the application rather than from the boot flow. Remain in infinite loop until reset
0xBE	Serial ROM bootloader	BLDR_FAIL_APPTRANSFER	Failure (should never happen): no valid application for bootloader to transfer to. Remain in infinite loop until reset
0xBF	Serial ROM bootloader	BLDR_FAULT_HANDLER	Uncaught/unspecified error during bootlloader execution. Remain in infinite loop until reset
0xC0	Boot flow	APP_MODE_ENTRY	Entering application mode. Exited bootloader if one was invoked. AHB-AP accessible from external debug probe if debug is open or has been authenticated
0xC1	Boot flow	APP_WAITLOOP_DBGPROBE	Debug of application allowed; busy loop to allow external probe to set breakpoints before single-stepping or running CPU
0xC2	Boot flow	BLDR_START_INIT	Boot complete: set immediately before jumping to application entry point
0xC3-0xFC	-	*Available for application usage*	*Suggest 0xC3 set at the very beginning of application startup code*
0xFD	Boot flow	APP_FAIL_NOAPP	Failure: (should never happen) bootloader exited and there is no valid application to boot into. Remain in infinite loop until reset
0xFE	Boot flow	APP_FAIL_APPTRANSFER	Failure: application returned. Should never happen as link register is set to 0xFFFF_FFFF before application entry function is called. A return from application will thus result in a fault (which may or may not be handled by the application)
0xFF	Boot flow	APP_FAULT_HANDLER	Uncaught/unspecified error during transition to application (typically incorrect vector table address provided in CCFG.bootCfg.pAppVtor or incorrect vector table contents).

8.1.4 Boot Protection/Locking Mechanisms

At various phases during boot, protection mechanisms are enabled and certain registers in the design are locked to improve FW security:

- Changes to hardware trims for oscillators, voltage/current references, flash trimming and power management output voltages
  - Wrong hardware trim values can result in the device operating out of spec and thus having unpredictable behavior.
- Flash sector write/erase protection
  - To avoid program errors, malicious attacks or a debug connection from changing contents of certain flash sectors
- SWD port and debug access

- The SWD port gives access to the CFG-AP and SEC-AP access points from boot to allow device IDs and information to be read out through CFG-AP and communication with SACI through SEC-AP
- The AHB-AP access point that us used by debug probe can be enabled before the bootloader is invoked or the application is invoked using CCFG configurations
- Debug access is enabled by default and if specified by CCFG.debugCfg or with CCFG authentication (password) configurations

8.1.5 Debug and Active SWD Connections at Boot

The SWD debug port is activated whenever a long sequence of bits is clocked into the SWDIO pin by edges on the SWDCK pin. The sequence is detected in all device power states except for reset but including shutdown by a module called IceMelter and the SWD connection status remains until:

- An SWD disconnect sequence is detected
- Power is cycled

In boot, if SWD is connected, the device management interface SACI is always entered even if there is a valid bootloader or application to run. SACI has a configurable timeout for when to continue boot if there is no SWD activity, otherwise SACI waits until instructed by a SACI command over SWD before continuing boot. Communication with SACI is done through SWD using the SEC-AP mailbox.

To allow debugging across resets or shutdown scenarios, SACI commands exist for continuing boot and halting CPU right before application (or bootloader if permitted) is entered. At this point hardware debug breakpoints must be resubmitted as these do not survive the reset. Debug (re)authentication SACI commands can also be required if dictated by CCFG.debugCfg.

See Chapter 5 or Section 8.3 for more details.

8.1.6 Flashless Test Mode and Tools Client Mode

There are two special modes that can be entered where the flash is completely isolated from the rest of the design and no reads, writes or erasures can be performed. These modes are designed to not allow the contents of flash to be read out or modified in any way.

8.1.6.1 Flashless Test Mode

Used by Texas Instruments for failure analysis on any parts returned by customers. This mode allows TI to run most production tests anew on the device and to analyze any reported symptoms/failures without gaining access to or modifying the contents of flash. After a reset the device boots into the programmed application again. A 256-bit password is required to gain access to this mode. There is (by design) no way to prevent a device from supporting flashless test mode.

8.1.6.2 Tools Client Mode

Used by Texas Instruments tools such as SmartRF Studio and Packet Sniffer to allow in-situ RF testing on customer boards without disturbing the application stored in Flash. This mode is similar to flashless test mode except that DFT mechanisms (scan test, RAM BIST, and so on) remain unavailable and that FCFG.appTrims (radio trim values) are copied to the beginning of SRAM before the flash is isolated and then the SWD debugging is enabled allowing test programs to be loaded into and run from SRAM. Tools client mode is not password protected but is only allowed if either CCFG is blank or CCFG.permissions.allowToolsClientMode == ALLOWED. Tools client mode can also be used by customers to do end-of-line testing without having to perform multiple flash program cycles.

8.1.7 Retest Mode and Return-to-Factory Procedure

To do full failure analysis (including flash), a return to factory procedure is supported. This procedure is only allowed if CCFG is blank or CCFG.permissions.allowReturnToFactory == ALLOWED. Before changing to the RETEST life cycle::

- All of SRAM is cleared
- All MAIN flash sectors are unconditionally erased
- CCFG is erased

- FCFG.lifeCycle is updated to RETEST
- A reset is required

8.1.8 Disabling SWD Debug Port

If CCFG.permissions.allowDebugPort = CCFG_PERMISSION_FORBID then the SWD port is disabled during boot before invoking either bootloader or application. From this point not even SWD:CFG-AP or SWD:SEC-AP access points are accessible. Before this point, if the SWD connection sequence has been detected by IceMelter, the device enters SACI during boot and device management commands are available. If all debugging access needs to be blocked then both CCFG.permissions.allowDebugPort = CCFG_PERMISSION_FORBID and CCFG.debugCfg.authorization = CCFG_DBGAUTH_DBGFORBID.

8.2 Flash Programming

A flash image can be programmed into the device using one of three methods:

- SWD Device Management Command Interface (Section 8.3)
  - Interface is always accessible unless an image is already programmed where CCFG.permissions.allowDebugPort==FORBID
  - Communication is done over the 2-pin serial wire debug (SWD) port also used for debug
  - Pipelined flash sector transfer and programming implemented to allow highest possible programming speed
  - Used by IDEs to download image and TI's UniFlash tool. Available on TI's evaluation modules.
  - Used by supported 3rd party gang programmers
- ROM Serial Bootloader
  - A simple serial (UART or SPI) bootloader for flash programming that resides in ROM
  - ROM serial bootloader invoked by default on blank devices and customers can choose to invoke the serial bootloader with their applications too
  - Implements flash erase/programming/verification commands similar to what SACI implements
  - Easily integrates into automated or semi-automated end-of-line testing
- User Bootloader (in flash)
  - A user-defined bootloader that resides in flash and can implement custom functionality such as FW image authentication

8.2.1 CCFG

The CCFG flash sector (described in Section 9.2) contains meta-information about/for the application:

- Boot configuration
  - Where the initial vector table of the application is so that the application entry function can be invoked and the application stack pointer set. Application can change vector table location later if needed.
  - Whether to invoke a bootloader and if so which bootloader and parameters to pass to the bootloader. Location of user bootloader.
- Permissions and hardware options
  - Whether various boot operations or non-debug SWD-related features are allowed
  - Ability to lock application out of certain hardware features or IPs (for example, to minimize harm that programming errors can do)
- Flash write/erase protections that apply to application
- Hardware initialization to perform before first application instruction is run
- Debug permissions and optionally (hash of) password
- User record

The user record is a 128-byte record in CCFG that can be written at the same time as CCFG is written or with a separate command later. This allows the user record to be written as part of a commissioning step separate from the application image that has been programmed. In this case the user record typically contains some kind of device unique ID, address or key.

8.2.2 CCFG Permissions/Restrictions that Affect Flash Programming

The CCFG.permissions record contains fields that controls which types of flash programming operations are allowed:

CCFG.permissions	Description
.allowChipErase	Determines whether the Chip Erase command is allowed. *Normally allowed, as otherwise the device cannot be reprogrammed through SACI or ROM serial bootloader*
.allowFlashProgram	Determines whether flash programming commands are allowed. Reset to allowed after a chip erase. *Normally not allowed to avoid changes to flash through flash programming interface after initial programming.*
.allowFlashVerify	Determines whether flash verify commands are allowed. These commands only check integrity against a provided CRC32 value, never return any flash contents. Flash verify commands are always allowed after a chip erase and until the first reset after the CCFG sector has been programmed *Normally allowed for checking integrity or identifying the flash image.*

On a blank device or a device otherwise without a valid CCFG, including after a chip erase command, these all default to *allowed*.

The CCFG.flashprot.writeEraseProt sub-record controls with finer granularity (down to sector or groups of 8 sector level) whether flash programming is allowed through SACI. The same mechanism controls whether the application is allowed to program these sectors. If flash programming operations are done on a blank device or a device otherwise without a valid CCFG, including after a chip erase command, these all default to *unrestricted*.

The CCFG.flashprot.chipEraseRetain sub-record controls with finer granularity (down to sector or groups of 8 sector level) whether a chip erase affects a sector or not. The mechanism is intended to allow flash sectors devoted to logging or runtime state/configuration to survive the chip erase during a FW update. The .chipEraseRetain sub-record can be used to protect a user bootloader as well, but any user bootloader for FW updates is unlikely to allow chip erase through SACI. This retention mechanism is not intended to be bullet-proof: a sequence of (chip erase → reset → chip erase) erases both CCFG and non-retained MAIN sectors the first time around and CCFG and all MAIN sectors the second time around.

8.2.3 SACI Flash Programming Commands

- User record commands
  - SACI_CMD_MISC_GET_CCFG_USER_REC
    - Allows the user record to be read out (only if CCFG is valid)
    - *Typically invoked prior to a chip erase command when the contents of user record need to be programmed back in after image update*
  - SACI_CMD_FLASH_PROG_CCFG_USER_REC
    - Program the user record with provided data (only if the user record is fully blank)
    - *Typically invoked either as part of some commissioning procedure long after the flash image was originally programmed, or as part of a flash image update to restore the previous user record content*
- Flash erasure commands
  - SACI_CMD_FLASH_ERASE_CHIP
    - First invalidates CCFG, then erases all (non-retained) MAIN sectors, then finally erases CCFG fully.
    - CCFG considered as invalid from here on out
    - Boot may not continue after flash programming commands complete: device must be reset and rebooted
    - *Typically invoked at the beginning of a flash programming session*
- Flash programming commands
  - SACI_CMD_FLASH_PROG_CCFG_SECTOR
    - Programs the entire CCFG sector (optionally skipping past the user record)
    - Only allowed if CCFG is already entirely blank

    - Boot may not continue after flash programming commands complete: device must be reset and rebooted
    - *Typically invoked at end of flash image update so that device does not boot into incomplete image if there is a power failure*
  - SACI_CMD_FLASH_PROG_MAIN_SECTOR
    - Program an arbitrary amount of data at an arbitrary address within a MAIN sector, up to and including the whole sector.
    - Considers CCFG.flashProt.writeEraseProt for this sector and allowFlashProgram
  - SACI_CMD_FLASH_PROG_MAIN_PIPELINED
    - Program any number of whole MAIN sectors starting at the beginning of a MAIN sector
    - Program data can be streamed continuously and sector programming happens in the background while data is being streamed
    - Considers CCFG.flashProt.writeEraseProt for each sector and allowFlashProgram
- Flash verification commands
  - *Note that all flash programming commands perform a full integrity check during programming as well, so verification is not strictly required*
  - SACI_CMD_FLASH_VERIFY_CCFG_SECTOR
    - Used for verification of CCFG sector, as one of:
      - Blank check: returns whether CCFG is fully unprogrammed or not
      - Integrity check: checks each part of CCFG against the embedded CRC32 checksums and reports whether all parts are valid or not. User record can be skipped
      - Identity check: computes checksum of each part of CCFG and compares against user provided CRC32 values to report whether all match or not. User record can be skipped
    - Only allowed if allowFlashVerify == ALLOW
    - Designed to prevent an outside attacker from gaining any information about the contents of CCFG
  - SACI_CMD_FLASH_VERIFY_MAIN_SECTORS
    - Used for verification of one or more whole MAIN sectors (optionally -4B to allow a CRC32 to be appended at end), as one of:
      - Blank check: return whether range is fully unprogrammed or not
      - Integrity/identity check: computes CRC32 over range and compares against user provided CRC32 value to report whether the value matches or not.
    - Only allowed if allowFlashVerify == ALLOW
    - Designed to prevent an outside attacker from gaining any information about flash contents

For full details see Section 8.3.

8.2.4 Flash Programming Flows

Some typical flash programming flows are given below:

8.2.4.1 Initial Programming of a New Device

Note

TI ships the devices with a blank CCFG sector but possibly with bit patterns from flash production test still in the main sectors.

- Perform SWD connect and reset device (through SWD reset or pin reset) to enter SACI
- Perform SACI_CMD_FLASH_ERASE_CHIP command
- Program an image using a sequence of SACI_CMD_FLASH_PROG_MAIN_SECTOR commands and one or more SACI_CMD_FLASH_PROG_MAIN_PIPELINED commands
- *Optionally* verify that image is correctly programmed using one or more SACI_CMD_FLASH_VERIFY_MAIN_SECTORS(crc32_calculated_or_extracted_from_image)
- Program in CCFG sector using SACI_CMD_FLASH_PROG_CCFG_SECTOR. If user record is in use either:
  - Program the user record now as part of SACI_CMD_FLASH_PROG_CCFG_SECTOR command
  - Program the user record in separate step using SACI_CMD_FLASH_PROG_CCFG_USER_REC command

  - Leave the user record unprogrammed for later commissioning step
- *Optionally* verify CCFG sector using SACI_CMD_FLASH_VERIFY_CCFG_SECTOR(identity check)
- Device can now be reset with SACI_CMD_BLDR_APP_RESET_DEVICE after which the device boots into application or any configured bootloader

8.2.4.2 Reprogramming of Previously Programmed Device

Note

Requires that CCFG.permissions.allowChipErase == ALLOWED

- Perform SWD connect and reset device (through SWD reset or pin reset) to enter SACI.
- *If a user record needs to be kept*, read out contents using SACI_CMD_MISC_GET_CCFG_USER_REC.
- Perform the SACI_CMD_FLASH_ERASE_CHIP command.
  - If there are main sectors (logging, runtime configuration, and so on) that should not be erased these should have been identified in CCFG.flashProt.chipEraseRetain and the option retainSelMainSectors passed to the SACI_CMD_FLASH_ERASE_CHIP command.
- *If applicable*, write back the user record using SACI_CMD_FLASH_PROG_CCFG_USER_REC.
- Program in the image using a sequence of SACI_CMD_FLASH_PROG_MAIN_SECTOR commands or one or more SACI_CMD_FLASH_PROG_MAIN_PIPELINED commands.
- *Optionally* verify that the image is correctly programmed using one or more SACI_CMD_FLASH_VERIFY_MAIN_SECTORS(crc32_calculated_or_extracted_from_image).
- Program in CCFG sector using SACI_CMD_FLASH_PROG_CCFG_SECTOR(skipUserRec).
- *Optionally,* verify the CCFG sector using SACI_CMD_FLASH_VERIFY_CCFG_SECTOR (identity check). If the user record has CRC32 at the end of the user record, integrity can be checked too.
- The device can now be reset/rebooted with SACI_CMD_BLDR_APP_RESET_DEVICE after which the device boots into the application or any configured bootloader.

8.2.4.3 Add User Record on Already Programmed Device as Part of Commissioning Step

- Perform SWD connect and reset device (through SWD reset or pin reset) to enter SACI
- Write user record using SACI_CMD_FLASH_PROG_CCFG_USER_REC. The command fails if the user record is not already blank.
- *Optionally and if the user record has a CRC32 at the end,* the user record integrity can now be checked with SACI_CMD_FLASH_VERIFY_CCFG_SECTOR(identity check).
- The device can now be reset/rebooted with SACI_CMD_BLDR_APP_RESET_DEVICE after which the device boots into the application or any configured bootloader.

8.2.4.4 Incrementally Program Ancillary Data to MAIN Flash Sectors of a Previously Programmed Device

Note

Requires that CCFG.permissions.allowFlashProgram == ALLOWED and that sectors in question are not write/erase protected by CCFG.flashProt.writeEraseProt

- Perform SWD connect and reset device (through SWD reset or pin reset) to enter SACI
- Program in ancillary data using one or more SACI_CMD_FLASH_PROG_MAIN_SECTOR or SACI_CMD_FLASH_PROG_MAIN_PIPELINED commands
- *Optionally* verify ancillary data that covers whole sectors through SACI_CMD_FLASH_VERIFY_MAIN_SECTORS(crc32_calculated_over_ancillary_data)
- Device can now be reset/rebooted with SACI_CMD_BLDR_APP_RESET_DEVICE after which the device boots into application or any configured bootloader

8.2.4.5 Debug Flow Charts

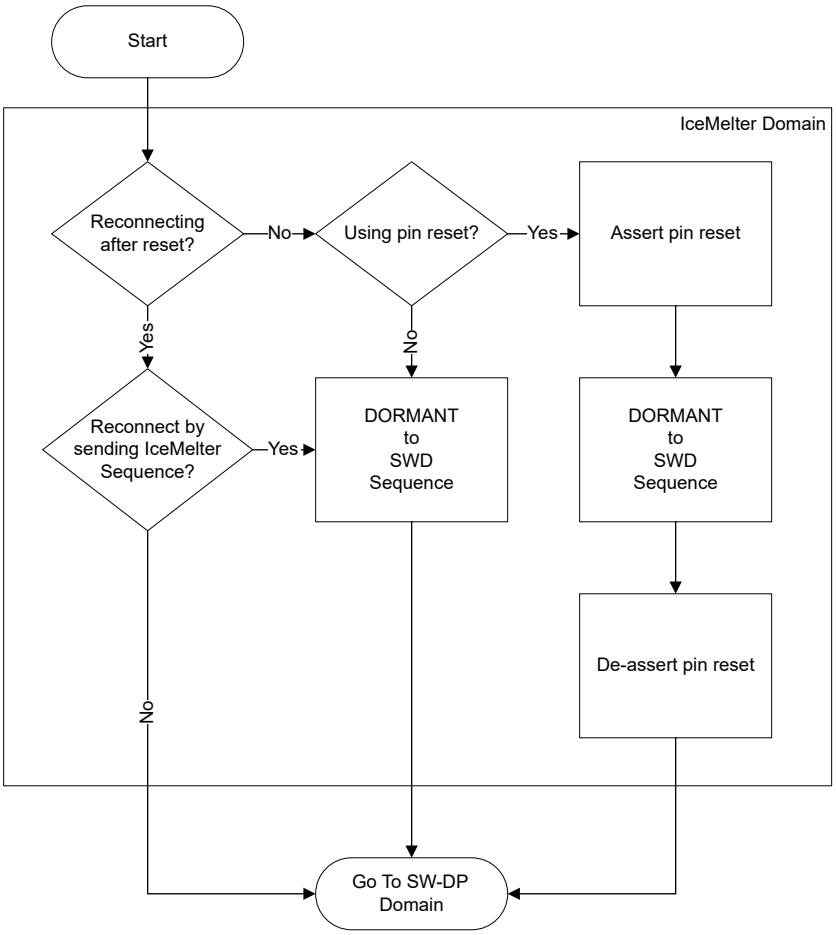


Figure 8-2. Debug - IceMelter Domain

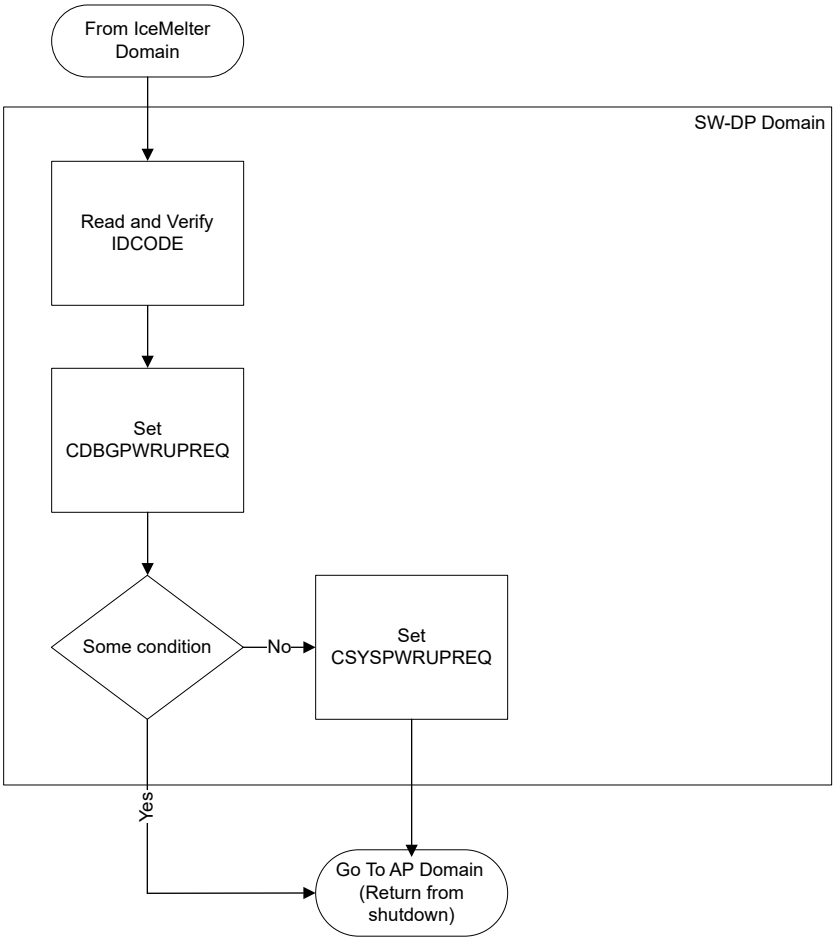


Figure 8-3. Debug - SW DP Domain

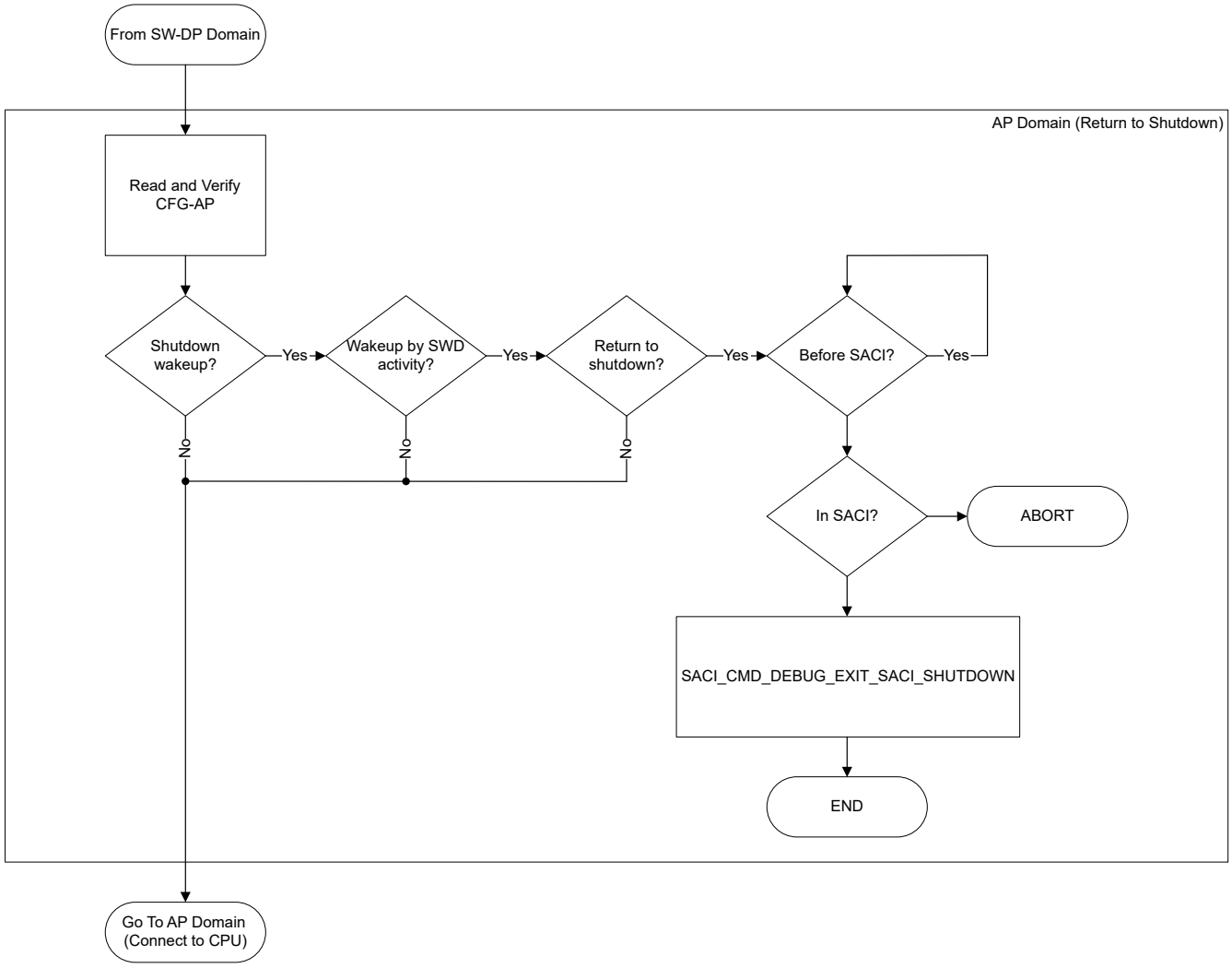


Figure 8-4. Debug - Return to Shutdown

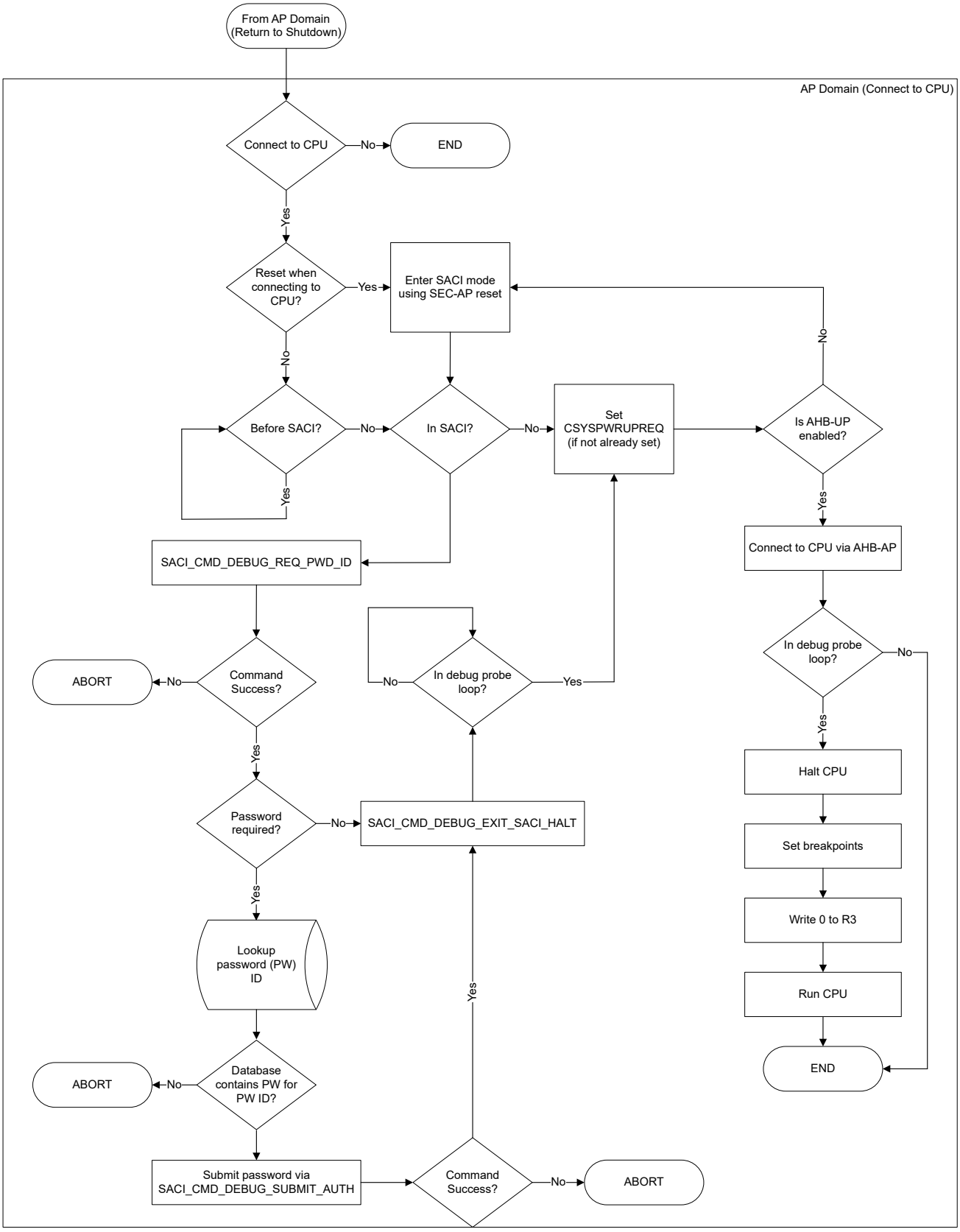


Figure 8-5. Debug - Connect to CPU

8.3 Device Management Command Interface

SACI (Secure-AP Command Interface) is a controlled privilege state the device enters during boot when:

- The device is in certain (manufacturing and failure analysis) lifecycle states
- There is no valid firmware image to boot into (unprogrammed device)
- An active SWD connection exists
- Invalid CCFG detected during boot

SACI implements a set of commands through a hardware mailbox mechanism in SEC-AP that allow an external debug probe or production programmer to:

- Get information about the device and firmware image currently on the device
- Manage device lifecycle state
- Erase device, program a new firmware image and verify image integrity
- Control in-circuit debugging
- Enter test modes

During boot in general and while in SACI, the AHB-AP is not open and thus a debug probe can not access any part of memory or control the CPU. SACI implements a controlled privilege environment and which commands are available depends on CCFG validity/configuration and device lifecycle. When entering/exiting SACI all SRAM memory is cleared to eliminate the risk of any application state to leak out.

If the device has a valid firmware image or bootloader image to boot into, SACI times out if a first command is not received within a configurable timeout (see CCFG.misc.saciTimeoutOverride and CCFG.misc.saciTimeoutExp) and boots normally.

8.3.1 SACI Communication Protocol

A SWD connection must first have been established and the device either halted in SACI during boot or reset (pin reset or SEC_AP reset see Chapter 5). Communication from external host to device uses these SWD registers in SEC-AP:

- DEBUGSS:TXD (data)
- DEBUGSS:TXCTL (flags)
  - Bit 0 / TXD_FULL: Indicates that TXD can be read. Set by hardware when TXD is written, cleared by hardware when TXD is read
  - Bit 1 / CMD_START: Indicates that TXD contains the first word of a command

Communication from device to host uses these registers:

- DEBUGSS:RXD (data)
- DEBUGSS:RXCTL (flags)
  - Bit 0 / RXD_FULL: Set by hardware when RXD is written, cleared by hardware when RXD is read
  - Bit 1 / CMD_ABORTED: Indicates that the previous command was aborted, meaning a new command was started before:
    - All parameter words for the previous command were received, or
    - The last response word of the previous command could be written to RXD
  - Bit 2 / CMD_WORKING: Indicates that SACI is working on a command after receiving all parameters
  - Bit 3 / CMD_ERROR: Indicates any type of error (invalid SACI command ID, prohibited operation, invalid parameters)

8.3.1.1 Host Side Protocol

The external host must follow these steps to execute any SACI command:

- Wait until TXD_FULL = 0
- Set CMD_START
- Write first parameter word to TXD
- If there are more parameter words:
  - Wait until TXD_FULL = 0

  - Clear CMD_START
  - Write second parameter word to TXD
  - Wait until TXD_FULL = 0
  - For each additional word:
    - Write the parameter word to TXD
    - There is no need to check TXD_FULL
- Wait until TXD_FULL = 0
- For commands with returned response, if relevant:
  - Wait until RXD_FULL = 1
  - Read RXD
  - For each response data word:
    - Wait until RXD_FULL = 1
    - Read RXD

The host must implement a timeout while waiting for TXD_FULL = 0:

- If the timeout occurs, the host should assume that the target is in an unknown state and abort or restart the session.
- This timeout can be set relatively high (for example, 1 second), since the timeout should not occur frequently and only one time per session.

8.3.1.2 Command Format

Bits 15:0 of the first SACI command parameter word have fixed formatting. Bits 31:16 of the first parameter word, and later parameter words, if any, are command specific.

Words	Bits	Field	Value	Description
0	7:0	cmdId	-	Command ID
	15:8	respSeqNumber	User defined	Optional response sequence number, included in the response header
	31:16	< command specific >	-	Command specific parameters, if any

8.3.1.3 Response Format

The first SACI command response word has fixed formatting.

Note the extended usage of the "first response word" for SACI_CMD_FLASH_PROG_MAIN_PIPELINED.

Words	Bits	Field	Value	Description
0	7:0	cmdId	-	Command ID, copied from the first command parameter word
	15:8	respSeqNumber	User defined	Optional sequence number, copied from the first command parameter word
	23:16	result	-	Command result
	31:24	dataWordCount	-	Size of additional response data, in number of 32-bit words

8.3.1.4 Response Result Field

The result field of the first response word can have the values in the table below:

- Value 0x80 or higher indicates some type an error that is also reflected by CMD_ERROR.

Result	Value	Description
SUCCESS	0x00	Command executed successfully
INVALID_CMD_ID	0x80	Invalid command ID
INVALID_ADDRESS_PARAM	0x81	Invalid address parameter
INVALID_SIZE_PARAM	0x82	Invalid size parameter
INVALID_KEY_PARAM	0x83	Invalid key parameter
FLASH_FSM_ERROR	0x84	Flash hardware FSM error

Result	Value	Description
PARAM_BUFFER_OVERFLOW	0x85	Parameter data buffer overflow (host must slow down or implement flow control)
NOT_ALLOWED	0x86	Command is not allowed due to restrictions (the command had no effect)
CRC32_MISMATCH	0x87	Calculated CRC32 does not match embedded/expected CRC32
INVALID_PWD_PARAM	0x88	Invalid password parameter
BLANK_CHECK_FAILED	0x89	Blank check detected one or more flash bits that were zero
CMD_FAILED	0xFF	Command failed, non-specific error (should not be used)

8.3.1.5 Command Sequence Tag

The host can optionally increment the response sequence number in the first parameter word of each command sent to SACI. This can be used to identify the corresponding response.

The host should read and check the response for each command.

8.3.1.6 Host Side Timeout

The host must implement a response timeout in case:

- Parameter words are lost due to electrical noise
- Parameter words (for example, a length-related field) are incorrectly received due to electrical noise
- The device fails to complete the operation for some other reason (for example, due to electrical noise on the reset pin)

Note

Some commands take more time to complete than others.

Note

Flash-related commands take more and more time as flash wear increases.

8.3.2 SACI Commands

SACI Command	cmdId	Category	Parameters	Restrictions	Description
SACI_CMD_MISC_ NO_OPERATION	0x01	-	-	-	Performs no operation.
TI Internal Command	0x02	-	-	-	-
SACI_CMD_MISC_ GET_DIE_ID	0x03	Information	-	Tested device only	Get the 128-bit die ID which identifies uniquely the die on the wafer
SACI_CMD_MISC_ GET_CCFG_USER_REC	0x04	Information	-	CCFG valid	Get the user record in CCFG, up to 128 bytes (in increments of 16 bytes)
SACI_CMD_DEBUG_ REQ_PWD_ID	0x05	Debug/control	-	Tested device only	Request password ID for debug authentication
SACI_CMD_DEBUG_ SUBMIT_AUTH	0x06	Debug/control	wordCount, pwd	CCFG valid and debug allowed	The host uses this command to submit the debug authentication password. If correct, debugging is enabled after exiting SACI

SACI Command	cmdId	Category	Parameters	Restrictions	Description
SACI_CMD_DEBUG_ EXIT_SACI_HALT	0x07	Debug/control	-	Debugging allowed, valid user bootloader or application	Exit SACI for debug mode, and wait for the host to setup breakpoints or run/single-step CPU through AHB-AP at the first instruction of the bootloader/application
SACI_CMD_DEBUG_ EXIT_SACI_SHUTDOWN	0x08	Debug/control	-	Wakeup from shutdown power state due to SWD activity	Exit SACI, and re-enter shutdown power state
SACI_CMD_FLASH_ ERASE_CHIP	0x09	Flash programming	retainSelMainSectors, key	Chip erase allowed	Perform chip erase: erase CCFG and main sectors. Optionally a set of main sectors indicated by CCFG can be retained
SACI_CMD_FLASH_ PROG_CCFG_SECTOR	0x0C	Flash Programming	skipUserRec, key, data	CCFG erased	Program the entire CCFG sector, with option to leave the user record part unprogrammed
SACI_CMD_FLASH_ PROG_CCFG_USER_REC	0x0D	Flash programming	key, data	CCFG.userRecord unprogrammed	Program the user record part of the CCFG sector
SACI_CMD_FLASH_ PROG_MAIN_SECTOR	0x0E	Flash programming	key, firstByteAddr, data	Flash erased or CCFG allows flash programming	Program all or a part of one MAIN sector
SACI_CMD_FLASH_ PROG_MAIN_PIPELINED	0x0F	Flash programming	key, firstSectorAddr	Flash erased or CCFG allows flash programming	Program multiple whole, back-to-back MAIN sectors of the flash (full programming speed)
SACI_CMD_FLASH_ VERIFY_MAIN_SECTORS	0x10	Flash programming	doBlankCheck, firstSectorAddr, byteCount,expCrc32	Flash erased (blank check only) or CCFG allows flash verify	Verify the contents of one or more whole (with option to exclude the last 4 bytes) flash MAIN sectors against supplied CRC32 or check that the sectors are blank (all bytes are 0xFF)
SACI_CMD_FLASH_ VERIFY_CCFG_SECTOR	0x11	Flash programming	checkExpCrcs, skipUserRec, doBlankCheck, expBootCfgCrc32, expCentralCrc32, expUserRecCrc32, expDebugCfgCrc32	Flash erased (blank check only) or CCFG allows flash verify	Verify the contents of records in CCFG sectors against supplied CRC32 values or check that CCFG sector is blank (all bytes are 0xFF)
TI Internal Command	0x12				
TI Internal Command	0x13				
SACI_CMD_BLDR_ APP_RESET_DEVICE	0x14	Debug/control	waitForSwdDisconnect	-	Reset the device and reboot (possibly into SACI again)
SACI_CMD_BLDR_ APP_EXIT_SACI_RUN	ox15	Debug/control	-	Valid bootloader or application present, no flash programming commands in current SACI session	Exit SACI, and run bootloader or application
TI Internal Command	0x16				
SACI_CMD_MODE_ REQ_TOOLS_CLIENT	0x17	Test Mode	-	Flash erased or CCFG allows tool client mode	Locks all accesses to flash, erase SRAM and then open up device for debug access. Used by TI development tools

SACI Command	cmdId	Category	Parameters	Restrictions	Description
TI Internal Command	0x18	-	-	-	-

8.3.2.1 Miscellaneous Commands

8.3.2.1.1 SACI_CMD_MISC_NO_OPERATION

Performs no operation.

This command can be used to:

- Disable the inactivity timeout after reset
- Check SACI connection

Restrictions

None.

Table 8-3. Parameter Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x01	Command ID
	15:8	respSeqNumber	User defined	Optional response sequence number, included in the response header
	31:16	reserved0	0x0000	Reserved

Table 8-4. Response Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x01	Command ID, copied from the first command parameter word
	15:8	respSeqNumber	User defined	Optional sequence number, copied from the first command parameter word
	23:16	result	See desc.	Command result (no special result codes for this command)
	31:24	dataWordCount	0	Size of additional response data, in number of 32-bit words

8.3.2.1.2 SACI_CMD_MISC_GET_DIE_ID

Get the 128-bit die ID from FCFG, which identifies uniquely the die on the wafer.

The die ID is fetched from Fcfg.deviceInfo.dieId.

Restrictions

None.

Table 8-5. Parameter Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x03	Command ID
	15:8	respSeqNumber	User defined	Optional response sequence number, included in the response header
	31:16	reserved0	0x0000	Reserved

Table 8-6. Response Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x03	Command ID, copied from the first command parameter word
	15:8	respSeqNumber	User defined	Optional sequence number, copied from the first command parameter word
	23:16	result	-	Command result (no special result codes for this command)
	31:24	dataWordCount	4	Size of additional response data, in number of 32-bit words
4:1	31:0	dieId	Fcfg.deviceInfo.dieId	The die ID, as 4 x 32-bit words: • The first byte of the die ID is in bits 7:0 of word 1 • The last byte of the die ID is in bits 31:24 of word 4

8.3.2.1.3 SACI_CMD_MISC_GET_CCFG_USER_REC

Get the user record in CCFG, up to 128 bytes (in increments of 16 bytes).

Restrictions

CCFG must be valid (Ccfg.bootCfg.crc32 is valid).

Table 8-7. Parameter Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x04	Command ID
	15:8	respSeqNumber	User defined	Optional response sequence number, included in the response header
	31:16	reserved0	0x0000	Reserved

Table 8-8. Response Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x04	Command ID, copied from the first command parameter word
	15:8	respSeqNumber	User defined	Optional sequence number, copied from the first command parameter word
	23:16	result	-	Command result (no special result codes for this command)
	31:24	dataWordCount	0-32	Size of additional response data, in number of 32-bit words: • 32 words = 128 bytes • 0 = 0 bytes if restrictions are not met
1+	31:0	data	-	User record data.

Check Sequence

These checks are performed before command execution, in the indicated order:

- If any restriction is violated: Fail with result NOT_ALLOWED.

8.3.2.2 Debug Commands

Debug authentication and SACI exit to debug mode functionality.

8.3.2.2.1 SACI_CMD_DEBUG_REQ_PWD_ID

Request password ID for debug authentication.

This command must be used to check whether debugging is allowed, and obtain the password ID for SACI_CMD_DEBUG_SUBMIT_AUTH, if required. As indicated, there are three possible states:

Ccfg.debugCfg.authorization	Description	Indicated by
0xA5	Debug authentication is required	Response contains a 64-bit debug password ID
0x5A	Debug authentication is not required	Response contains no password ID. Debugging is enabled after exiting SACI.
Any other value	Debugging is not allowed	Response result is NOT_ALLOWED

The password ID is fetched from Ccfg.debugCfg.pwdId.

Restrictions

CCFG must be valid (Ccfg.bootCfg.crc32 is valid)..

Ccfg.debugCfg.authorization = 0xA5 or 0x5A.

Table 8-9. Parameter Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x05	Command ID
	15:8	respSeqNumber	User defined	Optional response sequence number, included in the response header
	31:16	reserved0	0x0000	Reserved

Table 8-10. Response Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x05	Command ID, copied from the first command parameter word
	15:8	respSeqNumber	User defined	Optional sequence number, copied from the first command parameter word
	23:16	result	-	Command result. One of the common results, or: • NOT_ALLOWED if restrictions are not met
	31:24	dataWordCount	0 or 2	Size of additional response data, in number of 32-bit words: • 2 if debug authentication is required • 0 if debug authentication is not required, or debugging is not allowed
2:1	31:0	pwdId	Ccfg.debugCfg.pwdId	The 64-bit debug password ID: • The first byte of the password ID is in bits 7:0 of word 1 • The last byte of the password ID is in bits 31:24 of word 2

Check Sequence

These checks are performed before command execution, in the indicated order:

- If any restriction is violated: Fail with result NOT_ALLOWED

8.3.2.2.2 SACI_CMD_DEBUG_SUBMIT_AUTH

If the SACI_CMD_DEBUG_REQ_PWD_ID command returns a 64-bit password ID, the host must use this command to submit the password that corresponds to this ID. The host must know the password length.

If the submitted password is correct, debugging is enabled after exiting SACI.

The password check calculates SHA-256 of the submitted password, and compares the resulting 256b hash with Ccfg.debugCfg.pwdHash. The password check takes constant time.

The password is only checked if debug authentication is required (see SACI_CMD_DEBUG_REQ_PWD_ID). If not required, the result is always SUCCESS.

Restrictions

CCFG must be valid (Ccfg.bootCfg.crc32 is valid)..

Ccfg.debugCfg.authorization = 0xA5 or 0x5A.

Table 8-11. Parameter Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x06	Command ID
	15:8	respSeqNumber	User defined	Optional response sequence number, included in the response header
	31:16	wordCount	3 - 16	Size of the password, in number of 32-bit words (min = 3, max = 16)
N:1	31:0	pwd	Device specific	The password, N = wordCount

Table 8-12. Response Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x06	Command ID, copied from the first command parameter word
	15:8	respSeqNumber	User defined	Optional sequence number, copied from the first command parameter word
	23:16	result	-	Command result. One of the common results, or: • NOT_ALLOWED if restrictions are not met • INVALID_SIZE_PARAM if the wordCount parameter is invalid – For wordCount > 16, the response will come after reception of the 16th word • INVALID_PWD_PARAM if the pwd parameter is required and invalid
	31:24	dataWordCount	0	Size of additional response data, in number of 32-bit words

Check Sequence

These checks are performed before command execution, in the indicated order:

- If any restriction is violated: Fail with result NOT_ALLOWED

8.3.2.2.3 SACI_CMD_DEBUG_EXIT_SACI_HALT

Exit SACI for debug mode, and wait for the host to setup a breakpoint at the first instruction of the bootloader/application.

Before this command, the host must use SACI_CMD_DEBUG_REQ_PWD_ID, and if needed SACI_CMD_DEBUG_SUBMIT_AUTH, to enable access to AHB-AP. Otherwise the debug session fails.

Host Follow-Up Actions

After executing this command, the host must:

- Wait until CFGAP.DEVICESTATUS.BOOTSTA equals one of these values:
  - 0x81 (BLDR_WAITLOOP_DBGPROBE) = The boot code is waiting before bootloader entry (CCFG restrictions not yet applied)
  - 0xC1 (APP_WAITLOOP_DBGPROBE) = The boot code is waiting before application entry
- Halt the CPU
- Optional: Configure breakpoint at start of main() or at the start of the application reset vector
- If debugging an application that runs in flash:
  - Write CPU register R3 = 0x00000000
  - Run the CPU
- Otherwise, if debugging an application that runs in SRAM:
  - Write the application image to SRAM
  - Load CPU register SP
  - Load CPU register PC
  - Run the CPU
- Wait until the CPU has reached the breakpoint, if any

Restrictions

SACI_CMD_FLASH_ERASE_CHIP or SACI_CMD_FLASH_PROG_CCFG_SECTOR must not have been used during the current SACI session.

Bootloader must be specified in CCFG or FCFG, or there must be a valid application.

Debugging must be allowed (see SACI_CMD_DEBUG_REQ_PWD_ID for details).

Table 8-13. Parameter Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x07	Command ID
	15:8	respSeqNumber	User defined	Optional response sequence number, included in the response header
	31:16	reserved0	0x0000	Reserved

Table 8-14. Response Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x07	Command ID, copied from the first command parameter word
	15:8	respSeqNumber	User defined	Optional sequence number, copied from the first command parameter word
	23:16	result	-	Command result. One of the common results, or: • NOT_ALLOWED if restrictions are not met
	31:24	dataWordCount	0	Size of additional response data, in number of 32-bit words

Check Sequence

These checks are performed before command execution, in the indicated order:

- If any restriction is violated: Fail with result NOT_ALLOWED

8.3.2.2.4 SACI_CMD_DEBUG_EXIT_SACI_SHUTDOWN

Exit SACI, and enter shutdown mode.

Restrictions

The device must have woken up from shutdown due to SWDCK activity.

Table 8-15. Parameter Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x08	Command ID
	15:8	respSeqNumber	User defined	Optional response sequence number, included in the response header
	31:16	reserved0	0x0000	Reserved

Table 8-16. Response Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x07	Command ID, copied from the first command parameter word
	15:8	respSeqNumber	User defined	Optional sequence number, copied from the first command parameter word
	23:16	result	-	Command result. (1) • NOT_ALLOWED if restrictions are not met
	31:24	dataWordCount	0	Size of additional response data, in number of 32-bit words

(1) Normally none: the device immediately returns to shutdown when the command is processed if restrictions are met. A response is only generated if the command fails to meet the restrictions.

Check Sequence

These checks are performed before command execution, in the indicated order:

- If any restriction is violated: Fail with result NOT_ALLOWED

8.3.2.2.5 SACI_CMD_BLDR_APP_RESET_DEVICE

Reset the device, with or without reentering SACI.

The command has an option to wait for an SWD disconnection sequence (DBGSS:DBGBCTRL.SWDSEL = 0) before the reset is triggered.

- If used, the boot code will not enter SACI after the reset
  - CFGAP.DEVICESTATUS.BOOTSTA is changed to 0x36 (WAIT_SWD_DISCONNECT) when starting to wait for the disconnect sequence
- If not used, the boot code will reenter SACI after the reset (or move on to the bootloader/application after the SACI inactivity timeout)

There is a 1 ms delay immediately before the reset is triggered.

The command uses PMCTL.RSTCTL.SYSRSTREQ to trigger a system reset. This is indicated by PMCTL.RSTSTA.SYSRSTEV = 1 after the reset.

Restrictions

None.

Table 8-17. Parameter Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x14	Command ID
	15:8	respSeqNumber	User defined	Optional response sequence number, included in the response header
	16	waitForSwdDisconnect	-	Specify whether to wait for SWD disconnection before the device reset occurs: • 0 to reset and re-enter SACI • 1 to wait for SWD disconnection sequence, and then reset without re-entering SACI
	31:17	reserved0	0x0000	Reserved

Table 8-18. Response Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x14	Command ID, copied from the first command parameter word
	15:8	respSeqNumber	User defined	Optional sequence number, copied from the first command parameter word
	23:16	result	-	Command result (no special result codes for this command) [1]
	31:24	dataWordCount	0	Size of additional response data, in number of 32-bit words

(1) The command only generates a response if waitForSwdDisconnect = 1. The response is generated before the SWD disconnect sequence detection.

8.3.2.2.6 SACI_CMD_BLDR_APP_EXIT_SACI_RUN

Exit SACI, and run bootloader or application.

If a bootloader is specified in CCFG or FCFG, the device enters the bootloader. Otherwise the device enterts the application.

The device enables access to AHB-AP if Ccfg.debugCfg.authorization = 0x5A, or if SACI_CMD_DEBUG_SUBMIT_AUTH has been performed successfully. This allows a debugger to "attach to running target".

Restrictions

SACI_CMD_FLASH_ERASE_CHIP or SACI_CMD_FLASH_PROG_CCFG_SECTOR must not have been used during the current SACI session.

Bootloader must be specified in CCFG or FCFG, or there must be a valid application.

Table 8-19. Parameter Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x15	Command ID
	15:8	respSeqNumber	User defined	Optional response sequence number, included in the response header
	31:16	reserved0	0x0000	Reserved

Table 8-20. Response Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x15	Command ID, copied from the first command parameter word
	15:8	respSeqNumber	User defined	Optional sequence number, copied from the first command parameter word
	23:16	result	-	Command result. One of the common results, or: • NOT_ALLOWED if restrictions are not met
	31:24	dataWordCount	0	Size of additional response data, in number of 32-bit words

Check Sequence

These checks are performed before command execution, in the indicated order:

- If any restriction is violated: Fail with result NOT_ALLOWED

8.3.2.3 Flash Programming Commands

8.3.2.3.1 SACI_CMD_FLASH_ERASE_CHIP

Perform chip erase.

This erases the CCFG sector and MAIN sectors of the flash, by:

- Invalidating CCFG (all fields in Ccfg.bootCfg changed to 0)
- Erasing all MAIN sectors, except retained sectors (if any)
- Erasing the CCFG sector

The command has an option to retain selected MAIN sectors:

- Specified by the existing CCFG (before the chip erase) in:
  - Ccfg.chipEraseRetain.mainSectors0_31 (1 sector per bit) and Ccfg.chipEraseRetain.mainSectors32_255 (8 sectors per bit)
- The option relies on sticky write/erase protection in the VIMS module. Therefore, after use:
  - SACI_CMD_FLASH_ERASE_CHIP cannot be used again during the current SACI session
  - The retained MAIN sectors are write protected during the current SACI session

If successful, CCFG becomes invalid:

- CCFG restrictions do no longer apply
- Commands that require valid CCFG are prohibited

Restrictions

These conditions must be true:

- Fcfg.permissions.allowChipErase = 0xA (FCFG_PERMISSION_ALLOW)
- If CCFG is valid:
  - Ccfg.permissions.allowChipErase = 0xA (CCFG_PERMISSION_ALLOW)
- If the "retain selected MAIN sectors" option is used:
  - CCFG must be valid (Ccfg.bootCfg.crc32 is valid).
  - For flash sector not valid for the device type, the value of these invalid sectors within Ccfg.flashProt.chipEraseRetain.mainSectors0_31 and Ccfg.flashProt.chipEraseRetain.mainSectors32_255, must be set to the register reset value of the corresponding sectors within the VIMS:WEPRA and VIMS:WEPRB protection registers
- SACI_CMD_FLASH_ERASE_CHIP must not have executed previously in the current SACI session with the "retain selected MAIN sectors" option

Considerations

This command modifies CCFG. Certain commands are not allowed after chip erase:

- SACI_CMD_DEBUG_EXIT_SACI_HALT
- SACI_CMD_BLDR_EXIT_SACI_RUN
- In some cases SACI_CMD_FLASH_ERASE_CHIP (see restrictions)

Reseting the device and reentering SACI is required to reenable those commands.

Table 8-21. Parameter Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x09	Command ID
	15:8	respSeqNumber	User defined	Optional response sequence number, included in the response header
	16	retainSelMainSectors	-	Retain MAIN sectors, as specified by • Ccfg.chipEraseRetain.mainSectors0_31 (1 sector per bit) • Ccfg.chipEraseRetain.mainSectors32_255 (8 sectors per bit)
	31:17	reserved0	0x0000	Reserved
1	31:0	key	0xB7E3A08F	Flash operation key (magic number)

Table 8-22. Response Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x09	Command ID, copied from the first command parameter word
	15:8	respSeqNumber	User defined	Optional sequence number, copied from the first command parameter word
	23:16	result	-	Command result. One of the common results, or: • INVALID_KEY_PARAM if the key parameter is incorrect • FLASH_FSM_ERROR if the flash hardware FSM reported an error • NOT_ALLOWED if restrictions are not met (the command has no effect)
	31:24	dataWordCount	0 or 2	Size of additional response data, in number of 32-bit words: • 0 if retainSelMainSectors = 0 • 2 if retainSelMainSectors = 1
1	31:0	retainedMain0_31	-	The initial value of Ccfg.chipEraseRetain.mainSectors0_31 (1 sector per bit)
2	31:0	retainedMain32_255	-	The initial value of Ccfg.chipEraseRetain.mainSectors32_255 (8 sectors per bit)

Check Sequence

These checks are performed before command execution, in the indicated order:

- If any restriction is violated: Fail with result NOT_ALLOWED
- If key is invalid: Fail with result INVALID_KEY_PARAM

8.3.2.3.2 SACI_CMD_FLASH_PROG_CCFG_SECTOR

Program the entire CCFG sector, with option to skip the user record part.

If skipped, the user record (Ccfg.userRecord) can be programmed later, using SACI_CMD_FLASH_PROG_CCFG_USER_REC.

The CCFG is still considered invalid after executing this command. The new CCFG sector contents take effect after reset.

Restrictions

All bytes in the CCFG sector must be 0xFF before the CCFG sector programming begins (a chip erase must have been performed since the previous CCFG programming).

Considerations

This command modifies CCFG. Certain commands are not allowed after CCFG sector programming:

- SACI_CMD_MISC_GET_CCFG_USER_REC
- SACI_CMD_DEBUG_EXIT_SACI_HALT
- SACI_CMD_BLDR_EXIT_SACI_RUN

Reseting the device and reentering SACI is required to reenable those commands.

Table 8-23. Parameter Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x0C	Command ID
	15:8	respSeqNumber	User defined	Optional response sequence number, included in the response header
	16	skipUserRec	-	Skip user record part
	31:17	reserved0	0x0000	Reserved
1	31:0	key	0xB7E3A08F	Flash operation key (magic number)
513:2	31:0	data	-	Bytes to be programmed: • The first byte is in bits 7:0 of word 2 • The last byte is in bits 31:24 of word 513 If skipUserRec = 1, the user record part of data is don't care.

Table 8-24. Response Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x0C	Command ID, copied from the first command parameter word
	15:8	respSeqNumber	User defined	Optional sequence number, copied from the first command parameter word
	23:16	result	-	Command result. One of the common results, or: • INVALID_KEY_PARAM if the key parameter is incorrect • FLASH_FSM_ERROR if the flash hardware FSM reported an error • NOT_ALLOWED if restrictions are not met (the command has no effect)
	31:24	dataWordCount	0	Size of additional response data, in number of 32-bit words

Check Sequence

These checks are performed before command execution, in the indicated order:

- If any restriction is violated: Fail with result NOT_ALLOWED
- If key is invalid: Fail with result INVALID_KEY_PARAM

8.3.2.3.3 SACI_CMD_FLASH_PROG_CCFG_USER_REC

Program the user record part of the CCFG sector, Ccfg.userRecord.

This command can be used **after** SACI_CMD_FLASH_PROG_CCFG_SECTOR has executed with the option to skip the user record.

Restrictions

All bytes in the CCFG user record must be 0xFF before the CCFG user record programming begins.

Table 8-25. Parameter Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x0D	Command ID
	15:8	respSeqNumber	User defined	Optional response sequence number, included in the response header
	31:16	reserved0	0x0000	Reserved
1	31:0	key	0xB7E3A08F	Flash operation key (magic number)
33:2	31:0	data	-	Bytes to be programmed: • The first byte is in bits 7:0 of word 2 • The last byte is in bits 31:24 of word 33

Table 8-26. Response Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x0D	Command ID, copied from the first command parameter word
	15:8	respSeqNumber	User defined	Optional sequence number, copied from the first command parameter word
	23:16	result	-	Command result. One of the common results, or: • INVALID_KEY_PARAM if the key parameter is incorrect • FLASH_FSM_ERROR if the flash hardware FSM reported an error • NOT_ALLOWED if restrictions are not met (the command has no effect)
	31:24	dataWordCount	0	Size of additional response data, in number of 32-bit words

Check Sequence

These checks are performed before command execution, in the indicated order:

- If any restriction is violated: Fail with result NOT_ALLOWED
- If key is invalid: Fail with result INVALID_KEY_PARAM

8.3.2.3.4 SACI_CMD_FLASH_PROG_MAIN_SECTOR

Program all or a part of one MAIN sector (see Chapter 7).

The programming starts when the specified number of bytes (padded to a whole number of words) has been received by SACI.

Restrictions

These conditions must be true:

- Fcfg.permissions.allowFlashProgram = 0xA (FCFG_PERMISSION_ALLOW)
- If CCFG is valid:
  - Ccfg.permissions.allowFlashProgram= 0xA (CCFG_PERMISSION_ALLOW)

The sector can be protected by:

- Fcfg.flashProt.writeEraseProt.mainSectors0_31 and Ccfg.flashProt.writeEraseProt.mainSectors0_31 (1 sector per bit)
- Fcfg.flashProt.writeEraseProt.mainSectors32_255 and Ccfg.flashProt.writeEraseProt.mainSectors32_255 (8 sectors per bit)

Table 8-27. Parameter Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x0E	Command ID
	15:8	respSeqNumber	User defined	Optional response sequence number, included in the response header
	31:16	byteCount	-	Number of bytes to program. If 0, the command has no effect.
1	31:0	key	0xB7E3A08F	Flash operation key (magic number)
2	31:0	firstByteAddr	-	Address of the first byte to be programmed
3+	31:0	data	User defined	Bytes to be programmed: • The first byte is in bits 7:0 of word 3 • Zero-padding must be added to the most significant part of the last word if byteCount is not a multiple of 4.

Table 8-28. Response Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x0E	Command ID, copied from the first command parameter word
	15:8	respSeqNumber	User defined	Optional sequence number, copied from the first command parameter word
	23:16	result	-	Command result. One of the common results, or: • INVALID_KEY_PARAM if the key parameter is incorrect • INVALID_ADDRESS_PARAM if the firstByteAddr parameter is invalid • INVALID_SIZE_PARAM if the byteCount parameter is invalid (address range spans over multiple MAIN sectors) • FLASH_FSM_ERROR if the flash hardware FSM reported an error – May indicate that the sector was retained by chip erase during the current SACI session • NOT_ALLOWED if restrictions are not met (the command has no effect)
	31:24	dataWordCount	0	Size of additional response data, in number of 32-bit words

Check Sequence

These checks are performed before command execution, in the indicated order:

- If firstByteAddr is invalid: Fail with result INVALID_ADDRESS_PARAM
- If byteCount = 0: Succeed with result SUCCESS
- If byteCount is invalid: Fail with result INVALID_SIZE_PARAM
- If any restriction is violated: Fail with result NOT_ALLOWED
- If key is invalid: Fail with result INVALID_KEY_PARAM

8.3.2.3.5 SACI_CMD_FLASH_PROG_MAIN_PIPELINED

Program multiple whole, back-to-back MAIN sectors of the flash.

This allows the entire MAIN bank to be programmed with only one SACI command, with high performance.

For this command, SACI has data buffers for two MAIN sectors. This enables pipelined programming:

- SACI starts programming a flash sector when all data for that sector has been received.
- During this programming, the host can send the data for the next sector.

Flow Control Mechanism and Response Handling

When SACI receives the last word of a sector, this triggers or pends programming of that sector. When SACI finishes programming of a sector, a response is generated for that sector. This can overwrite the response for the previous sector if the host has not yet read that.

If SACI fails to program a sector (e.g. due to protection), the command ends, and the response for that sector indicates the error. If the host skips/misses reading the response for a sector, the host can assume that the sector was successfully programmed if the host sees the response for a later sector.

The host must not write data for another sector while one sector is being programmed, and another sector is pending. This means:

- The host does not need to check the response until the host has written the first two sectors.
- After the two first sectors, the host can only write data for sector N after the host has read the response for either sector N-2 (one free buffer) or sector N-1 (two free buffers).

The host should always wait for and check the response for the last sector.

Restrictions

These conditions must be true:

- Fcfg.permissions.allowFlashProgram = 0xA (FCFG_PERMISSION_ALLOW)
- If CCFG is valid:
  - Ccfg.permissions.allowFlashProgram= 0xA (CCFG_PERMISSION_ALLOW)

Each sector can be protected by:

- Fcfg.flashProt.writeEraseProt.mainSectors0_31 and Ccfg.flashProt.writeEraseProt.mainSectors0_31 (1 sector per bit)
- Fcfg.flashProt.writeEraseProt.mainSectors32_255 and Ccfg.flashProt.writeEraseProt.mainSectors32_255 (8 sectors per bit)

Table 8-29. Parameter Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x0F	Command ID
	15:8	respSeqNumber	User defined	Base response sequence number
	31:16	reserved0	0x0000	Reserved
1	31:0	key	0xB7E3A08F	Flash operation key (magic number)
2	31:0	firstSectorAddr	-	Address of the first byte of the first sector to be programmed

Then, for each flash sector:

Words	Bits	Field	Value	Description
511:0	31:0	data	User defined	Sector data: • The first byte of the sector is in bits 7:0 of word 0 • The last byte of the sector is in bits 31:24 of word 511

This response is generated after each sector has been programmed:

Table 8-30. Response Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x0F	Command ID, copied from the first command parameter word
	15:8	respSeqNumber	User defined	Response sequence number. For the first sector, the number is equal to the respSeqNumber in the first parameter. For later sectors, the number increments by 1 for each sector.
	23:16	result	-	Command result. One of the common results, or: • INVALID_KEY_PARAM if the key parameter is incorrect • INVALID_ADDRESS_PARAM if the firstSectorAddr parameter is invalid • PARAM_BUFFER_OVERFLOW if the host sends sector data too quickly, without implementing the described response-based flow control mechanism • FLASH_FSM_ERROR if the operation failed (not due to restrictions) – Can indicate that the sector was retained by chip erase during the current SACI session • NOT_ALLOWED if restrictions are not met (the command has no effect)
	31:24	dataWordCount	0	Size of additional response data, in number of 32-bit words

Checks

These checks are performed before command execution, in the indicated order:

- If firstSectorAddr is invalid: Fail with result INVALID_ADDRESS_PARAM

These checks are performed before programming of each sector, in indicated order:

- If any restriction is violated: Fail with result NOT_ALLOWED
- If key is invalid: Fail with result INVALID_KEY_PARAM

The command finishes prematurely when a buffer overflow condition occurs, with result PARAM_BUFFER_OVERFLOW.

8.3.2.3.6 SACI_CMD_FLASH_VERIFY_MAIN_SECTORS

Verify the contents of one or more whole flash MAIN sectors, using CRC32, with option to exclude the last 4 bytes, or check that the sectors are blank (all bytes are 0xFF).

The command perform one of these checks:

- Check that the specified address range is blank (all bytes are 0xFF).
- Compute and check CRC32 over a whole number of back-to-back flash MAIN sectors
  - This can be used when the flash programming tool is able to calculate CRC32
- Compute and check CRC32 over a whole number of back-to-back flash MAIN sectors, minus 4 bytes
  - This can be used when the flash programming tools is unable to calculate CRC32, but the expected CRC32 is stored in the last 4 bytes of the last sector

The command result is SUCCESS if the performed check is successful. The command result is also SUCCESS if the number of bytes to check is 0.

Restrictions

These conditions must be true:

- Fcfg.permissions.allowFlashVerify = 0xA (FCFG_PERMISSION_ALLOW)
- If CCFG is valid:
  - Ccfg.permissions.allowFlashVerify= 0xA (CCFG_PERMISSION_ALLOW)

Table 8-31. Parameter Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x10	Command ID
	15:8	respSeqNumber	User defined	Optional response sequence number, included in the response header
	30:16	reserved0	0	Reserved
	31	doBlankCheck	-	Check if the entire address range is blank (all bytes are 0xFF), instead of CRC check
1	31:0	firstSectorAddr	-	Address of the first byte of the first sector
2	31:0	byteCount	-	Number of bytes to calculate CRC32 over, one of the following: • Whole number of sectors • Whole number of sectors minus 4 bytes
3	31:0	expCrc32	-	Expected CRC32

Table 8-32. Response Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x10	Command ID, copied from the first command parameter word
	15:8	respSeqNumber	User defined	Optional sequence number, copied from the first command parameter word
	23:16	result	-	Command result. One of the common results, or: • INVALID_ADDRESS_PARAM if the firstSectorAddr parameter is invalid • INVALID_SIZE_PARAM if the byteCount parameter is invalid • For doBlankCheck = 0: CRC32_MISMATCH if the calculated CRC32 does not match expCrc32 • For doBlankCheck = 1: BLANK_CHECK_FAILED if one or more bits in the address range are 0 • NOT_ALLOWED if restrictions are not met
	31:24	dataWordCount	0	Size of additional response data, in number of 32-bit words

Check Sequence

These checks are performed before command execution, in the indicated order:

- If firstSectorAddr is invalid: Fail with result INVALID_ADDRESS_PARAM
- If byteCount = 0: Succeed with result SUCCESS
- If byteCount is invalid: Fail with result INVALID_SIZE_PARAM
- If any restriction is violated: Fail with result NOT_ALLOWED

8.3.2.3.7 SACI_CMD_FLASH_VERIFY_CCFG_SECTOR

Verify the specified parts of the flash CCFG sector, using CRC32, or check that the entire sector is blank (all bytes are 0xFF).

CCFG is divided into four parts, each with an embedded CRC32 that covers the data:

- Boot configuration part
  - Data:
    - Ccfg.bootCfg, excluding Ccfg.bootCfg.crc32
  - Embedded CRC32:
    - Ccfg.bootCfg.crc32
- Central part
  - Data:
    - Ccfg.hwOpts
    - Ccfg.misc
    - Ccfg.flashProt
    - Ccfg.permissions
    - Ccfg.hwInitCopyList
  - Embedded CRC32:
    - Ccfg.crc32
- User record part
  - Data:
    - Ccfg.userRecord, excluding Ccfg.userRecord.crc32
  - Embedded CRC32:
    - Ccfg.userRecord.crc32
- Debug configuration part
  - Data:
    - Ccfg.debugCfg, excluding Ccfg.debugCfg.crc32
  - Embedded CRC32:
    - Ccfg.debugCfg.crc32

The command performs one of these checks:

- Check that CCFG is blank (all bytes are 0xFF).
- Compute and check CRC32 of data, using only embedded CRCs:
  - For all parts of CCFG, or
  - For all parts of CCFG except the user record part
- Compute and check CRC32 of data, using both embedded CRCs and expected CRCs (provided in the command):
  - For all parts of CCFG, or
  - For all parts of CCFG except the user record part

The command result is SUCCESS if all performed checks are successful.

Restrictions

These conditions must be true:

- Fcfg.permissions.allowFlashVerify = 0xA (FCFG_PERMISSION_ALLOW)
- If CCFG is valid:
  - Ccfg.permissions.allowFlashVerify= 0xA (CCFG_PERMISSION_ALLOW)

Table 8-33. Parameter Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x11	Command ID
	15:8	respSeqNumber	User defined	Optional response sequence number, included in the response header
	16	checkExpCrcs	-	Check CCFG data also against the expected CRC32 fields the command: • 0: Use only embedded CRCs • 1: Use both embedded CRCs and the expected CRCs
	17	skipUserRec	-	Skip CRC check of the user record part of CCFG
	30:18	reserved0	0b000000	Reserved
	31	doBlankCheck	-	Check if the entire CCFG sector is blank (all bytes are 0xFF), instead of CRC checks
1	31:0	expBootCfgCrc32	-	Expected CRC32 of the boot configuration part: • If checkExpCrcs = 1: Same value as in Ccfg.bootCfg.crc32 • Otherwise: Don't care
2	31:0	expCentralCrc32	-	Expected CRC32 of the central part: • If checkExpCrcs = 1: Same value as in Ccfg.crc32 • Otherwise: Don't care
3	31:0	expUserRecCrc32	-	Expected CRC32 of the user record part: • If skipUserRec = 0 and checkExpCrcs = 1: Same value as in Ccfg.userRecord.crc32 • Otherwise: Don't care
4	31:0	expDebugCfgCrc32	-	Expected CRC32 of the debug configuration part: • If checkExpCrcs = 1: Same value as in Ccfg.debugCfg.crc32 • Otherwise: Don't care

Table 8-34. Response Words

Words	Bits	Field	Value	Description
0	7:0	cmdId	0x11	Command ID, copied from the first command parameter word
	15:8	respSeqNumber	User defined	Optional sequence number, copied from the first command parameter word
	23:16	result	-	Command result. One of the common results, or: • For doBlankCheck = 0: CRC32_MISMATCH if any of the performed CRC checks failed • For doBlankCheck = 1: BLANK_CHECK_FAILED if one or more bits in the CCFG sector are 0 • NOT_ALLOWED if restrictions are not met
	31:24	dataWordCount	0	Size of additional response data, in number of 32-bit words

Check Sequence

These checks are performed before command execution, in the indicated order:

- If any restriction is violated: Fail with result NOT_ALLOWED

8.4 Bootloader Support

Whether a bootloader is run or not and where bootloader parameters are picked from depends on CCFG.

Table 8-35. Bootloader CCFG Configurations

CCFG valid	CCFG.bootCfg.pBldrVtor	Bootloader Invoked	Bootloader parameters selected
No	-	ROM Serial bootloader	FCFG.bootCfg.bldrParam
Yes	USE_FCFG (0xFFFFFFF0)	ROM Serial bootloader	FCFG.bootCfg.bldrParam
Yes	FORBID (0xFFFFFFFC)	None	-
Yes	UNDEF (0xFFFFFFFF)	ROM Serial bootloader	CCFG.bootCfg.bldrParam
Yes	Valid MAIN flash address	CCFG.bootCfg.pBldrVtor[1] (SP: CCFG.bootCfg.pBldrVtor[0])	CCFG.bootCfg.bldrParam

If a bootloader is selected, then that bootloader is always invoked as part of the boot process. The bootloader must thus quickly determine whether to activate or not, and if not pass control back to the boot process by calling the API function *HapiEnterApplication()*. For more information, see hapi.h in the SimpleLink™ CC23xx Software Development Kit (SDK). The ROM serial bootloader determines whether the serial bootloader is triggered (or not) within a few microseconds.

8.4.1 Bootloader Parameters

The bootloader entry function (function pointer in xCFG.bootCfg.pBldrVtor[1]) has the following function prototype:

```
__noreturn void BootloaderEntryFunction(uint32_t bldrParam, uint32_t bldrFlags)
```

bldrParam is passed on from either FCFG.bootCfg.bldrParam or CCFG.bootCfg.bldrParam according to Table 8-35. bldrFlags contains useful information from the boot flow that the bootloader can use. See Table 8-36.

Table 8-36. Bootloader Flags

Flag	Name	Description
bldrFlags[0]	bCcfgValid	1 if CCFG is valid, else 0
bldrFlags[1]	bAppCanBoot	1 if there exists an application that will be booted into once HapiEnterApplication() is called. *Typically if there is no application to boot into remain in the bootloader indefinitely*
bldrFlags[2]	bChipEraseAllowed	1 if chip erase is allowed by CCFG, 0 if not. Any bootloader should for consistency adhere to this restriction
bldrFlags[3]	bParamsFromCcfg	1 if bldrParam is from CCFG, 0 if bldrPAram is from FCFG.
bldrFlags[4]	bBldrAllowDbg	1 if debugging of bootloader is allowed, 0 if not. If 1, the boot flow may already have opened for debugging prior to invoking bootloader.
bldrFlags[31:5]	*Reserved*	Do not assume any specific value

8.4.2 Persistent State

The application may communicate with the bootloader through the REG3V3 register, the contents of which survive a reset and any power-mode transition (including prolonged shutdown and reset power state). The typical use case is a bootloader that performs an FW image update (from the image in internal or external SPI flash) when instructed by the application. The bootloader may also use REG3V3 to keep track of the state and progress of the firmware update process across multiple resets.

The REG3V3 registers:

- 18b assigned to user
- Reset to value zero by power-on-reset
- Can be read through the PMCTL.AONRSTA1.FLAG field
- Individual bits can be set through the PMCTL.AONRSET1.FLAG field.
- Individual bits can be cleared through the PMCTL.AONRCLR1.FLAG field.

A bootloader must remember to clear flags/fields passed from the application as the request has been dealt with so that it doesn't occur multiple times.

8.4.3 User-Defined Bootloader Guidelines

When writing a user-defined bootloader keep the following points in mind:

- Placement
  - Normally a user bootloader resides at the beginning of MAIN flash (sector 0, 1, and so on) to allow per-sector write/erase protection (from the application) and chip erase protection (from the programmer).
  - The bootloader can use all of SRAM except the area 0x20000400-0x2000047F which contains boot flow state variables.
- Entering and exiting the bootloader
  - The bootloader is always invoked as part of any boot flow, so to optimize boot time the bootloader should quickly determine whether it has been triggered or not (typical mechanisms are I/O-level on some pin, a flag from an application somewhere in flash, a flag from application in the REG3V3 registers).
  - The entry function must never return and should accept two 32-bit arguments, bldrParam, and bldrFlags.
  - If not triggered the bootloader shall return control back to the boot flow through the Hard API function *HapiEnterApplication()*, see SimpleLink™ CC23xx Software Development Kit (SDK) documentation.
    - Between the return from bootloader and the invocation of the application boot flow applies CCFG restrictions, opens up for (application-only) debug, and processes the application's HW initialization copy list.
    - Ensure that SP is unwound to the top of the stack (or that at least 256 bytes are available downwards until the start of SRAM) before calling *HapiEnterApplication().*
- CCFG bootloader configuration
  - CCFG.bootCfg.pBldrVtor is a pointer to the bootloader's ISR vector table which has to contain at least two entries (initial-stack-pointer, function-pointer-to-bootloader-entry-function)
    - The bootloader should also populate the VTOR entries for CPU exceptions (HardFault, UserFault, NMI, and so on) or immediately change to a RAM-based VTOR that does
    - Bootloader can reuse the space used by boot flow for call stack by placing initial SP at 0x20000400.
  - Bootloader entry function arguments
    - The first is copied from CCFG.bldrParam and typically contains some bootloader-specific configuration options (for example, trigger pin, serial baud rate, timeout, and so on).
    - The second is bldrFlags, which contains useful hints from the boot flow. At a minimum, bAppCanBoot and bChipEraseAllowed should be heeded.
- General
  - To keep the size of the bootloader small it is advantageous to reuse some of the functionality already in ROM exposed through the Hard API (CRC32, SHA2-256, Flash programming functions, and so on).

Note

The bootloader has an elevated privilege level compared to even the OS kernel. Among other things, the bootloader can program the CCFG sector or program/erase sectors that later have hardware write/erase protection.

8.5 ROM Serial Bootloader

The ROM contains a simple serial bootloader that by default is run on a blank device. Users can elect to run the serial bootloader before their application once one is programmed (unless using their own user bootloader or not using a bootloader). The ROM serial bootloader:

- Implements a set of commands for programming/erasing/verifying flash (similar to SACI but slightly simplified)
- These commands can be accessed over UART or SPI. Whichever interface sees activity first is selected from there on out. UART has an autobaud feature and can detect the baud rate within the range from 9600 to 1.6Mbaud.

- Supports multiple different UART/SPI pin-mapping options for increased flexibility (default is available on all package options, this may not be true for all mapping options)
- Supports selecting any DIO pin as a trigger pin and selecting which logic level triggers the bootloader. Otherwise, the bootloader continues to the application (if one exists) immediately.

8.5.1 ROM Serial Bootloader Interfaces

The bootloader communicates with an external device over a 2-pin UART or a 4-pin SPI interface. The communication protocol and transport layers are described in the following subsections.

8.5.1.1 Packet Handling

The bootloader uses well-defined packets to ensure reliable communications with the external communicating program. All communications (with the exception of the UART automatic baud [see UART Transport Section 8.5.1.2.1]) use these well-defined packets. The packets are always acknowledged or not acknowledged by the communicating devices with defined ACK or NACK bytes.

The packets use the same format for receiving and sending packets. This format includes the method to acknowledge successful or unsuccessful reception of a packet.

While the actual signaling on the serial ports is different, the packet format remains the same for supported UART and SPI interfaces.

Packet send and packet receive must adhere to the simple protocol shown in Figure 8-6.

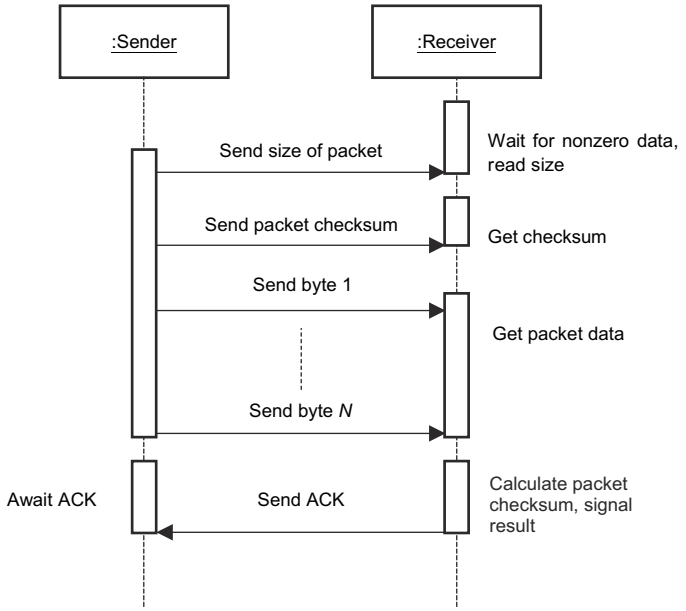


Figure 8-6. Sequence Diagram for Send and Receive Protocol

Perform the following steps to successfully send a packet:

1. Send the size of the packet to be transmitted to the device. The size is always the size of the data + 2 with truncation to 8 bits.
2. Send the checksum of the data buffer to ensure proper transmission of the command. The checksum algorithm is a sum of the data bytes.
3. Send the actual data bytes.
4. Wait for a single-byte acknowledgment from the device that the data was properly received or that a transmission error was detected.

Perform the following steps to successfully receive a packet:

1. Wait for nonzero data to be returned from the device. This is important as the device may send zero bytes between a sent and a received data packet. The first nonzero byte received is the size of the packet that is being received.
2. Read the next byte, which is the checksum for the packet.
3. Read the data bytes from the device. During the data phase, a packet size minus 2 bytes is sent. For example, if the packet size was 3, then there is only 1 byte of data to be received.
4. Calculate the checksum of the data bytes and verify it matches the checksum received in the packet.
5. Send an acknowledge byte or a not-acknowledge byte to the device to indicate the successful or unsuccessful reception of the packet.

Acknowledge (ACK) bytes are sent out whenever a packet is successfully received and verified by the receiver. A not-acknowledge (NACK) byte is sent out whenever a sent packet is detected to have an error, usually as a result of a checksum error or just malformed data in the packet, which allows the sender to retransmit the previous packet.

To illustrate packet handling, the basic packet format is shown in Figure 8-7.

In Figure 8-7, the top line shows the device that is transmitting data; the bottom line is the response from the other device.

In this case, a 6-byte packet is sent with the data shown in Figure 8-7. This data results in a checksum of 0x48+0x6f+0x6c+0x61 which, when truncated to 8 bits, is 0x84. The first byte transmitted holds the size of the packet in a number of bytes. Then the checksum byte is transmitted. The next bytes to go out are the 4 data bytes in this packet. The transmitter is allowed to send zeros until a nonzero response is received, which is necessary for SPI and is allowed by the UART. The receiver is allowed to return zeros until it is ready to ACK or NACK the packet that is being sent. Neither device transfers a nonzero byte until it has received a response after transmitting a packet.

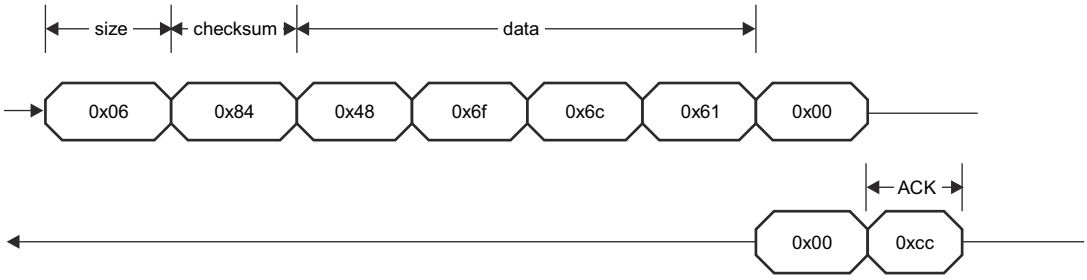


Figure 8-7. Serial Bus Packet Format

8.5.1.1.1 Packet Acknowledge and Not-Acknowledge Bytes

Table 8-37 shows the defined values for packet acknowledge (ACK) and not-acknowledge (NACK) bytes.

Table 8-37. Protocol Acknowledge and Not-Acknowledge Bytes

Protocol Byte	Value
ACK	0xCC
NACK	0x33

8.5.1.2 Transport Layer

The bootloader supports updating through the UART and SPI ports. The SPI port has the advantage of supporting higher and more flexible data rates, but it also requires more connections. The UART has the disadvantage of having slightly lower and possibly less flexible rates. However, the UART requires fewer pins and can be easily implemented with any standard UART connection.

Table 8-38 specifies which serial interface signals are configured to specific DIOs. There are three possible configurations for the serial interfaces. Configuration of the ROM Bootloader is done between the FCFG and the

CCFG. There are defaults that are set in the FCFG that will take effect if a valid CCFG is not present on start-up. If the user wants to alter the defaults set by the FCFG they can update their CCFG to provide the behavior they desire. See Chapter 9.

Table 8-38. Configuration of Serial Interfaces

Signal	serialIoCfgIndex == 0	serialIoCfgIndex == 1	serialIoCfgIndex == 2
UART_RX	DIO20	DIO12	DIO22
UART_TX	DIO6	DIO13	DIO20
SPI_CLK	DIO8	DIO24	DIO24
SPI_CS	DIO11	DIO11	DIO11
SPI_POCI	DIO12	DIO21	DIO12
SPI_PICO	DIO13	DIO13	DIO13

The bootloader initially configures only the input pins on the two serial interfaces. By default, all I/O pins have their input buffers disabled, so the bootloader configures the required pins to be input pins so that the bootloader interface is not accessible from a host before this point in time. For this initial configuration of input pins, the firmware configures the IOC to route the input signals listed in Table 8-38 to their corresponding peripheral signals.

The bootloader selects the interface that is the first to be accessed by the external device. Once selected, the TX output pin for the selected interface is configured; the module on the inactive interface (UART or SPI) is disabled. To switch to the other interface, the device must be reset. The delayed configuration of the TX pin imposes special consideration on an SPI controller device regarding the transfer of the first byte of the first packet (see Section 8.5.1.2.2).

8.5.1.2.1 UART Transport

The connections required to use the UART port are the following two pins: UART0 TX and UART0 RX. The device communicating with the bootloader drives the UART0 RX pin on the CC23xx, while the CC23xx drives the UART0 TX pin.

While the baud rate is flexible, the UART serial format is fixed at 8 data bits, no parity, and 1 stop bit. The bootloader automatically detects the baud rate for communication.

8.5.1.2.1.1 UART Baud Rate Automatic Detection

The bootloader provides a method to automatically detect the UART baud rate used to communicate with it.

To synchronize with the host, the bootloader must receive 2 bytes with the value of 0x55. If synchronization succeeds, the bootloader returns an acknowledge consisting of 2 bytes with the values of 0x00 and 0xCC.

If synchronization fails, the bootloader waits for synchronization attempts.

In the automatic-detection function, the UART0 RX pin is monitored for edges using GPIO interrupts. When enough edges are detected, the bootloader determines the ratio of baud rate and frequency needed to program the UART.

The UART module system clock must be at least 16 times the baud rate; thus, the maximum baud rate can be no higher than 3Mbaud (48MHz divided by 16). The maximum baud rate is restricted to 1.6Mbaud because of the firmware function that detects the transfer rate of the host.

8.5.1.2.2 SPI Transport

The connections required to use the SPI port are the following four pins:

- SPI_POCI
- SPI_PICO
- SPI_CLK
- SPI_CS

The device communicating with the bootloader drives the SPI_PICO, SPI_SCLK, and SPI_CS, while the CC23xx drives the SPI_POCI pin.

The format used for SPI communications is the Motorola format with SPH set to 1 and SPO set to 1 (see Figure 20-5 for more information on this format). Refer to the device-specific data sheet for the maximum rate supported on the SPI clock in the peripheral mode.

The controller must take special consideration (regarding the use of the SPI interface) due to the functionality of not configuring any output pins before the external controller device has selected a serial interface.

Note

On the first packet transferred by the controller, no data is received from the bootloader while the bootloader clocks out the bits in the first byte of the packet.

When the bootloader detects that 1 byte has been received on SPI_PICO, the bootloader configures the SPI_POCI output pin.

Before transmitting the next byte in the first packet, the controller must include a small delay to ensure that the bootloader has completed the configuration of the SPI_POCI output pin.

8.5.2 ROM Serial Bootloader Parameters

The ROM serial bootloader parameters are as follows:

CCFG.bootCfg.bldrParam	Name	Description	Blank device / FCFG default value
[0]	bldrEnabled	If 0, bootloader ignores all commands except BLDR_CMD_GET_STATUS, else normal operation	1
[1]	pinTriggerEnabled	If 0, bootloader unconditionally triggers, else normal pin trigger check is performed	1
[2]	pinTriggerLevel	If 0, a low level on trigger pin triggers bootloader, else a high level	0
[15:3]	*Reserved*		
[21:16]	pinTriggerDio	Index of DIO pin to use for pin trigger check	21
[23:22]	*Reserved*		
[26:24]	serialIoCfgIndex	Index of which I/O mapping to use for UART/SPI:	2 (QFN-40 package) 0 (all other packages)
[31:27]	*Reserved*		

UART	SPI	Index 0	Index 1	Index 2	Index 3...7
RXD		DIO20	DIO12	DIO22	*Reserved*
TXD		DIO6	DIO13	DIO20	*Reserved*
	POCI	DIO12	DIO21	DIO12	*Reserved*
	PICO	DIO13	DIO13	DIO13	*Reserved*
	SCLK	DIO8	DIO24	DIO24	*Reserved*
	CS	DIO11	DIO11	DIO11	*Reserved*

8.5.3 ROM Serial Bootloader Commands

Table 8-39. Command IDs

CMD ID	Value
BLDR_CMD_PING	0x20
BLDR_CMD_GET_STATUS	0x21

Table 8-39. Command IDs (continued)

CMD ID	Value
BLDR_CMD_GET_PART_ID	0x22
BLDR_CMD_RESET	0x23
BLDR_CMD_CHIP_ERASE	0x24
BLDR_CMD_CRC32	0x25
BLDR_CMD_DOWNLOAD	0x26
BLDR_CMD_DOWNLOAD_CRC	0x27
BLDR_CMD_SEND_DATA	0x28

8.5.3.1 BLDR_CMD_PING

This command is used to receive an acknowledge from the bootloader proving that communication has been established. This command is a single byte.

Byte	Field	Value	Description
0	cmdId	BLDR_CMD_PING	Ping Command ID

8.5.3.2 BLDR_CMD_GET_STATUS

This command returns the status of the last command that was issued. Typically, this command should be received after every command is sent to ensure that the previous command was successful or, if unsuccessful, to properly respond to a failure. The command requires one don't care byte in the data of the packet and the bootloader will respond by sending a packet with one byte of data that contains the current status code.

Byte	Field	Value	Description
0	cmdId	BLDR_CMD_GET_STATUS	Get StatusCommand ID

Table 8-40. Return Status Codes

Return Name	Value	Description
BLDR_CMD_RET_SUCCESS	0x40	This is returned in response to a **BLDR_CMD_GET_STATUS** command and indicates that the previous command completed successfully.
BLDR_CMD_RET_UNKNOWN_CMD	0x41	This is returned in response to a **BLDR_CMD_GET_STATUS** command and indicates that the command sent was an unknown command.
BLDR_CMD_RET_INVALID_CMD	0x42	This is returned in response to a **BLDR_CMD_GET_STATUS** command and indicates that the previous command was formatted incorrectly.
BLDR_CMD_RET_INVALID_ADR	0x43	This is returned in response to a **BLDR_CMD_GET_STATUS** command and indicates that the previous download command contained an invalid address value.
BLDR_CMD_RET_FLASH_FAIL	0x44	This is returned in response to a **BLDR_CMD_GET_STATUS** command and indicates that an attempt to program or erase the flash has failed.
BLDR_CMD_RET_CRC_FAIL	0x45	This is returned in response to a **BLDR_CMD_GET_STATUS** command and indicates that the previous CRC32 command match failed.

Table 8-40. Return Status Codes (continued)

Return Name	Value	Description
BLDR_CMD_RET_NEEDS_CHIP_ERASE	0x46	This is returned in response to a **BLDR_CMD_GET_STATUS** command and indicates that the previous Download command failed because a **BLDR_CMD_CHIP_ERASE** command must be run first.

8.5.3.3 BLDR_CMD_GET_PART_ID

This command is sent to the bootloader to get the Part ID of the device.

Byte	Field	Description
0	cmdId	BLDR_CMD_GET_PART_ID command ID

8.5.3.4 BLDR_CMD_RESET

This command is used to tell the bootloader to reset. This is used after downloading a new image to the device to cause the new application to start from a reset. The normal boot sequence occurs and the image runs as if from a hardware reset. This command can also be used to reset the bootloader if a critical error occurs and the host device wants to restart communication with the bootloader.

The bootloader responds with an ACK signal to the host device before actually executing the system reset of the device running the bootloader. This informs the updater host device that the command was received successfully and the part will be reset.

Byte	Field	Description
0	cmdId	BLDR_CMD_RESET command ID

8.5.3.5 BLDR_CMD_CHIP_ERASE

This command is used to perform a chip erase of the device. All main flash bank sectors not protected by FCFG and CCFG protect bits are erased. The CCFG is erased once the bank erase has completed.

This command first invalidates the CCFG and then begin erasing all unprotected sectors in the main flash bank. Once the flash sectors have been erased, the command finally erases the contents of the CCFG.

If the CCFG permissions disallow a chip erase, the command responds with CMD_INVALID_CMD.

The write/erase flash protections applied in the CCFG (see Table 9-3) won't affect the ROM bootloader. If write/erase flash protection is required, consider implementing a 'User-Defined' Bootloader (see User-Defined Bootloader Guidelines)

Byte	Field	Description
0	cmdId	BLDR_CMD_CHIP_ERASE command ID

8.5.3.6 BLDR_CMD_CRC32

This command is sent to the bootloader to calculate a CRC32 for a specified memory area. The command consists of three 32-bit values that are each transferred MSB first.

The Memory address must be sector aligned. Only memory addresses within the main flash region or the CCFG_BASE address itself are valid.

The Size must be sector aligned or (sector - 4 bytes) aligned.

The combination of memory address and size cannot go outside of either the main flash region or the CCFG region. If the parameters are valid, the command only reports if the expected CRC matches the calculated CRC. Follow up this command with the **BLDR_CMD_GET_STATUS** to read the result of the CRC comparison.

Byte	Field	Description
0	cmdId	BLDR_CMD_CRC32 command ID
1	Memory Address [31:24]	Memory Address to start the CRC calculation
2	Memory Address [23:16]	
3	Memory Address [15:8]	
4	Memory Address [7:0]	
5	Memory Area Size [31:24]	Number of bytes to run the CRC calculation over
6	Memory Area Size [23:16]	
7	Memory Area Size [15:8]	
8	Memory Area Size [7:0]	
9	Expected CRC [31:24]	The CRC value the host is expecting the CRC calculation results in
10	Expected CRC [23:16]	
11	Expected CRC [15:8]	
12	Expected CRC [7:0]	

8.5.3.7 BLDR_CMD_DOWNLOAD

This command is sent to the bootloader to indicate where to store data and how many bytes are sent by the **BLDR_CMD_SEND_DATA** commands that follow. The command consists of two 32-bit values that are both transferred MSB first.

The first 32-bit value is the address to start programming data into, while the second is the 32-bit size of the data that is to be sent.

This command should be followed by **BLDR_CMD_GET_STATUS** to ensure that the program address and program size were valid for the device running the bootloader.

Byte	Field	Description
0	cmdId	BLDR_CMD_DOWNLOAD command ID
1	Program Address [31:24]	Start address of the download
2	Program Address [23:16]	
3	Program Address [15:8]	
4	Program Address [7:0]	
5	Program Size [31:24]	Number of bytes (length of the download)
6	Program Size [23:16]	
7	Program Size [15:8]	
8	Program Size [7:0]	

8.5.3.8 BLDR_CMD_DOWNLOAD_CRC

This command is sent to the bootloader to indicate where to store data, how many bytes are to be sent by the **BLDR_CMD_SEND_DATA** commands that follow and the CRC32 value covering all the bytes.

The command consists of three 32-bit values that are all transferred MSB first. The first 32-bit value is the address to start programming data into, the second is the 32-bit size of the data that is to be sent and the third is the 32-bit CRC expected value.

This command should be followed by a **BLDR_CMD_GET_STATUS** to ensure that the program address and program size were valid for the device running the bootloader.

Byte	Field	Description
0	cmdId	BLDR_CMD_DOWNLOAD_CRC command ID

Byte	Field	Description
1	Program Address [31:24]	Start address of the download
2	Program Address [23:16]	
3	Program Address [15:8]	
4	Program Address [7:0]	
5	Program Size [31:24]	Number of bytes (length of the download)
6	Program Size [23:16]	
7	Program Size [15:8]	
8	Program Size [7:0]	
9	Expected CRC [31:24]	The expected CRC calculation over the completed download image. If the CRC calculation fails, the download is considered a failure and the content that was downloaded is immediately erased.
10	Expected CRC [23:16]	
11	Expected CRC [15:8]	
12	Expected CRC [7:0]	

8.5.3.9 BLDR_CMD_SEND_DATA

This command should only follow a **BLDR_CMD_DOWNLOAD** command or another **BLDR_CMD_SEND_DATA** command, if more data is needed.

Consecutive send data commands automatically increment the address and continue programming from the previous location. The caller should allow the device to finish the flash programming before issuing another command in order to avoid overflowing the input buffers of the serial interface. The command terminates programming once the number of bytes indicated by the **BLDR_CMD_DOWNLOAD** command has been received. Each time this function is called, it should be followed by a **BLDR_CMD_GET_STATUS** command to ensure that the data was successfully programmed into the flash.

If the bootloader responds with a NACK to this command, the bootloader does not increment the current address to allow re-transmission of the previous data.

A maximum of 253 bytes of data can be sent per **BLDR_CMD_SEND_DATA** command.

Byte	Field	Description
0	cmdId	BLDR_CMD_SEND_DATA command ID
1	download_image[0]	Consecutive bytes of the image to be downloaded
....	....	
X <= 252	download_image[X <= 252]	

8.5.4 Bootloader Firmware Update Example

The following steps can be followed to perform a FW image update to a device enabled to run the ROM Bootloader.

For this example, let's assume we have an updated application which begins at address 0x00000000, has a length of 0x28000 and a CRC over all of the bytes has a value of 0xFACEFACE.

1. The device needs to bootup into the ROM Bootloader. This can be done either by setting the pAppVtor to an invalid value OR by setting the pinTriggerDio to the specified pinTriggerLevel.
2. The bootloader needs to know which serial interface is being used.
   a. UART - send the AutoBaud sequence as described in Section 8.5.1.2.1.1
   b. SPI - send any ROM Bootloader cmd (ping command is a good suggestion)
3. Now that the bootloader is triggered and communicating correctly, prepare the device for the FW image update
   a. Send the BLDR_CMD_CHIP_ERASE command
   b. Wait for the ACK/NACK and then send the BLDR_CMD_GET_STATUS

   c. Wait for the ACK/NACK and ensure the previous command was completed successfully with the BLDR_CMD_RET_SUCCESS value
4. Start the Application Download
   a. Send the BLDR_CMD_DOWNLOAD_CRC command, passing as input the startAddress=(0x00000000), length=(0x28000), CRC=(0xFACEFACE)
   b. wait for the ACK/NACK and then send the BLDR_CMD_GET_STATUS
   c. wait for the ACK/NACK and ensure the previous command was completed successfully with the BLDR_CMD_RET_SUCCESS value
5. Loop over the bytes of the image and send the data to the ROM Bootloader
   a. Send the BLDR_CMD_SEND_DATA command, passing as input the next 252 bytes of the application image.
   b. wait for the ACK/NACK and then send the BLDR_CMD_GET_STATUS
   c. wait for the ACK/NACK and ensure the previous command was completed successfully with the BLDR_CMD_RET_SUCCESS value
   d. Repeat steps 5a-5c until all 0x28000 bytes have been transferred
6. Start the CCFG Download
   a. Send the BLDR_CMD_DOWNLOAD_CRC command passing as input the startAddress=(0x4E020000), length=(2048), CRC=(calc_crc(ccfg_contents))
   b. wait for the ACK/NACK and then send the BLDR_CMD_GET_STATUS
   c. wait for the ACK/NACK and ensure the previous command was completed successfully with the BLDR_CMD_RET_SUCCESS value
7. Loop over the bytes of the CCFG contents and send the data to the ROM Bootloader
   a. Send the BLDR_CMD_SEND_DATA command, passing as input the next 252 bytes of the CCFG content.
   b. wait for the ACK/NACK and then send the BLDR_CMD_GET_STATUS
   c. wait for the ACK/NACK and ensure the previous command was completed successfully with the BLDR_CMD_RET_SUCCESS value
   d. Repeat steps 5a-5c until all 2048 bytes have been transferred
8. Reset the device either by pulling the RST pin externally or by sending the BLDR_CMD_RESET command.
   a. Keeping in mind that the triggerPin should now be inverted so that the bootcode/bootlaoder can freely pass execution onto the application this time around
9. DONE! The device will now bootup into the new flash content that has been programmed to it.

Chapter 9

Device Configuration

This chapter describes the device configuration areas. The factory configuration (FCFG) and customer configuration (CCFG) areas are located in Flash. The FCFG is set by Texas Instruments during device production and contains device-specific trim values and configuration. The CCFG must be set by the application and contains configuration parameters for the ROM boot code, device hardware, and device firmware.

9.1 Factory Configuration (FCFG)

The FCFG flash sector is written during TI manufacturing and is write and erase protected out of device boot.

FCFG contains the following:

- Unique device identifiers and MAC addresses
- Device configurations
- Device trims
- Default bootloader definition

This section lists only a subset of the defined fields within the FCFG flash sector.

Fields listed are either referenced by other chapters in the TRM or can be accessed by a non-TI part of an application executing in flash.

For a detailed list of fields, refer to the hw_fcfg.h file found in the DriverLib part of the SimpleLink™ CC23xx Software Development Kit (SDK).

This C-header file holds a struct defining the complete layout of the FCFG flash sector.

Please note that minor updates of the FCFG field description (hw_fcfg.h) can occur as part of a Product SDK release.

The FCFG flash sector is split in sections as listed in the table below.

Table 9-1. FCFG Structure

FCFG hierarchy/field				Description
.deviceInfo				**Device information**
	.uuid[8]			64-bit device-unique UUID (non-sequential across parts)
	.bleAddr[6]			48-bit device-unique BLE address
	.macAddr[8]			64-bit device-unique IEEE MAC address
	.dieId[16]			128-bit die identifier (lot #, wafer #, die X/Y, date, etc).This ID is reported by the SACI_MISC_GET_DIE_ID SACI command.
	.partId			**Identification information specific to this orderable part number as reported in CFG-AP:PARTID**
		.part		Random bit pattern to uniquely identify numeric TI part number
		.variant		Random bit pattern to uniquely identify package/ memory variant suffix to TI part number
		.majorRev		Major revision for orderable part starting at 1
		.minorRev		Minor revision for orderable part starting at 0
.bootCfg				**Bootloader configuration**
	.pBldrVtor			Pointer to ISR vector table of bootloader (default: vector table for ROM serial bootloader)
	.bldrParam			Parameters passed to bootloader controlling bootloader behavior. For the ROM serial bootloader this controls the I/O pin used to trigger bootloader and which I/O pins are used for SPI/UART interfaces. (default: see Chapter 8)

9.2 Customer Configuration (CCFG)

Introduction

The CCFG flash sector contains meta-information about or for the application:

- Boot configuration:
  - Where the initial vector table of the application is so that the application's entry function can be invoked and the stack pointer set. Application can change the vector table location later if needed.
  - Whether to invoke a bootloader and if so which bootloader and parameters to pass to the bootloader. See Chapter 8 for more information.
  - The location of user bootloader.
- Permissions and hardware options:
  - Whether various boot operations or non-debug Serial Wire Debug (SWD) related features are allowed.
  - The ability to lock application out of certain hardware features or peripherals (for example, to minimize harm that programming errors can do)
- Flash write and erase protections that apply to the application
- Hardware initialization to perform before first application instruction is run
- Debug permissions and optional authorization options
- User record

For regular software application development, the TI supported SysConfig tool is used to create the contents of the CCFG.

This section covers only the main parts of the CCFG. For a detailed view of the CCFG structure please refer to the hw_ccfg.h file provided by the DriverLib part of the SimpleLink™ CC23xx Software Development Kit (SDK) or the SysConfig tool.

The C header file, hw_ccfg.h, provides a struct defining the complete CCFG layout.

Please note that minor updates of the CCFG field description (hw_fcfg.h) can occur as part of a Product SDK release.

The CCFG structure is split into sections, each having multiple fields. Table 9-3 presents a high level view of CCFG sections.

Most sections contain multiple fields. Only a subset of the fields are listed.

CRC Calculation and Location

The CRCs use CRC-32, which has the following implementation:

- The polynomial is $x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$
- The hexadecimal representation of the polynomial is `0x04C11DB7`.
- The initial value is `0xFFFFFFFF`.

There are four different CRCs used to validate the CCFG data. One of the CRCs, the user record CRC, is optional and is the last four bytes of the 128B user record. The data over which the CRC is calculated starts at “Data Start Offset” from Table 9-2 and ends at the "CRC Offset". CRC field width is 4 bytes.

Table 9-2. CRC Locations

CCFG Section	Data Start Offset	CRC Offset
`.bootCfg`	0x00	0xC
`.hwOpts` through `.hwInitCopyList[]`	0x10	0x74C
`.userRecord`	0x750	0x7CC
`.debugCfg`	0x7D0	0x7FC

Preparing for Production

When preparing the software image for production, review the following CCFG fields and set according to your system requirements

CCFG.permissions covers a number of restrictions on the flashing and debugging of the device. See Table 9-3 for details on each of the permission settings. See Chapter 8 for more detailed descriptions on boot behavior based on the CCFG configuration.

CCFG.flashProt covers the write/erase protection for the device. If certain areas of flash need to be restricted from erasing or writing then the appropriate fields are be written here. See Flash Read-Only Protection for details on which sectors are protected by these settings.

CCFG.debugCfg covers enabling debug features and the password protection of those features.

Table 9-3. CCFG Structure

CCFG hierarchy/field				Description
.bootCfg				**Contains function pointer that defines how to enter bootloader/application and also bootloader parameters.**
	.pBldrVtor			Pointer to user bootloader vector table
	.bldrParam			Parameter passed to bootloader
	.pAppVtor			Pointer to application VTOR table
	.crc32			CRC32 integrity checksum for CCFG
.hwOpts[2]				**Bitmask defining which peripherals/features and how much memory is accessible**

Table 9-3. CCFG Structure (continued)

CCFG hierarchy/field				Description
.permissions				**Device permission fields. This is maximally-restrictive combined with similar fields in FCFG.permissions.**
	...			*Misc unused/reserved permissions*
	.allowReturnToFactory			Allow Return-To-Factory procedure by SACI. Refer to Section 8.1.7 for details. Defined options are: • CCFG_PERMISSION_ALLOW (default) • CCFG_PERMISSION_FORBID
	.allowToolsClientMode			Allow tools client mode to be enabled by SACI. Defined options are: • CCFG_PERMISSION_ALLOW (default) • CCFG_PERMISSION_FORBID
	.allowChipErase			Allow chip erase by SACI or bootloader. Defined options are: • CCFG_PERMISSION_ALLOW (default) • CCFG_PERMISSION_FORBID
	.allowFlashProgram			Allow flash program by SACI. Defined options are: • CCFG_PERMISSION_ALLOW (default) • CCFG_PERMISSION_FORBID
	.allowFlashVerify			Allow flash verify by SACI. Defined options are: • CCFG_PERMISSION_ALLOW (default) • CCFG_PERMISSION_FORBID
	...			*Misc unused/reserved permissions*
	.allowDebugPort			Allow enabling of SWD port. Defined options are: • CCFG_PERMISSION_ALLOW (default) • CCFG_PERMISSION_FORBID
.misc				**Misc. boot-related fields**
	.saciTimeoutExp			Configures the SACI timeout when there is a bootloader or application to boot into: SACI timeout is infinite when 0, else (2^saciTimeoutExp)*64 ms. (default: 0x4 = 1s)
	.saciTimeoutOverride			Determines if SACI timeout defined in FCFG is overridden by SACI timeout defined in CCFG

Table 9-3. CCFG Structure (continued)

CCFG hierarchy/field				Description
.flashProt				**Flash write/erase protection fields. Protection is applied before entering bootloader/ application during boot**
	.writeEraseProt			**Write/erase protection fields**
		.mainSectors0_31		Bitmask for write/erase protection of individual sectors in sector range [0, 31]. 0 = protected (default: 1)
		mainSectors32_255		Bitmask for write/erase protection of groups of 8 sectors. Bit i protects sectors [32+8i, 39+8i]. 0 = protected (default: 1)
		.auxSectors		**Bitmasks for write/erase protection of auxilliary sectors**
			.ccfgSector	Protect CCFG sector 0 = protected (default: 1)
			.fcfgSector	Protect FCFG sector 0 = protected (default: 0)
			...	*Misc internal flash sector protections*
.hwInitCopyList[]				**Remaining hardware trims applied during boot. Stored in a flexible copy list format.**
.userRecord				**User record (programmable also through separate SACI command).**
.debugCfg				**Bootloader configuration**
	.authorization			Debug authorization requirements. Defined options are: CCFG_DBGAUTH_REQPWD: Require debug authentication (as per other fields in section) CCFG_DBGAUTH_DBGOPEN: Debug always allowed (AHB-AP opened upon bootloader/ application entry). CCFG_DBGAUTH_DBGFORBID: Debug not allowed.
	.allowBldr			Whether debugging of bootloader is allowed or not. Defined options are: CCFG_DBGBLDR_ALLOW: Bootloader debugging allowed. CCFG_DBGBLDR_FORBID: bootloader debugging not allowed.
	.pwdId[8]			64-bit customer-defined password ID readable through SACI command (can be used by user to calculate or look up debug password).
	.pwdHash[32]			256 bit of SHA256 hash of user-supplied response to password ID (through SACI command)
	.crc32			Integrity check of debugCfg section, Any integrity check error of debugCfg section is interpreted as debugCfg.authorization=Debug not allowed.

Chapter 10

General Purpose Timers (LGPT)

This section describes the General Purpose Timer (LGPT) module and provides example use case scenarios.

10.1 Overview

The General Purpose Timer (LGPT) is used to count or time external or internal events, generate pulse-width modulation (PWM) signals and generate IR-modulated codes.

There are up to four general-purpose timers available. See the device-specific data sheet for available timers and features.

Features

- General timing features, capture and compare
  - 3 Capture/Compare channels per timer
  - 16-bit counter width or
  - 24-bit counter width
- 8-bit prescaler
  - Configurable counter rate
  - Count from an external event
- Different counter modes
  - Count up once
  - Count up repeatedly
  - Count up and down repeatedly
  - Start counting on the configurable event
  - Quadrature decoding (QDEC)
- 15 different channel Capture/Compare actions
  - Period and pulse width measurement
  - Three capture actions
  - 12 compare actions
- Filtering on capture inputs
- Generate PWM
  - Complementary PWM outputs
  - Programmable deadband insertion
  - Park Mode on fault, sets the DIO to a predetermined state upon fault
- IR signal generation
- Generate interrupts, DMA requests, and ADC triggers
- Possible to chain the timers together and synchronize them

10.2 Block Diagram

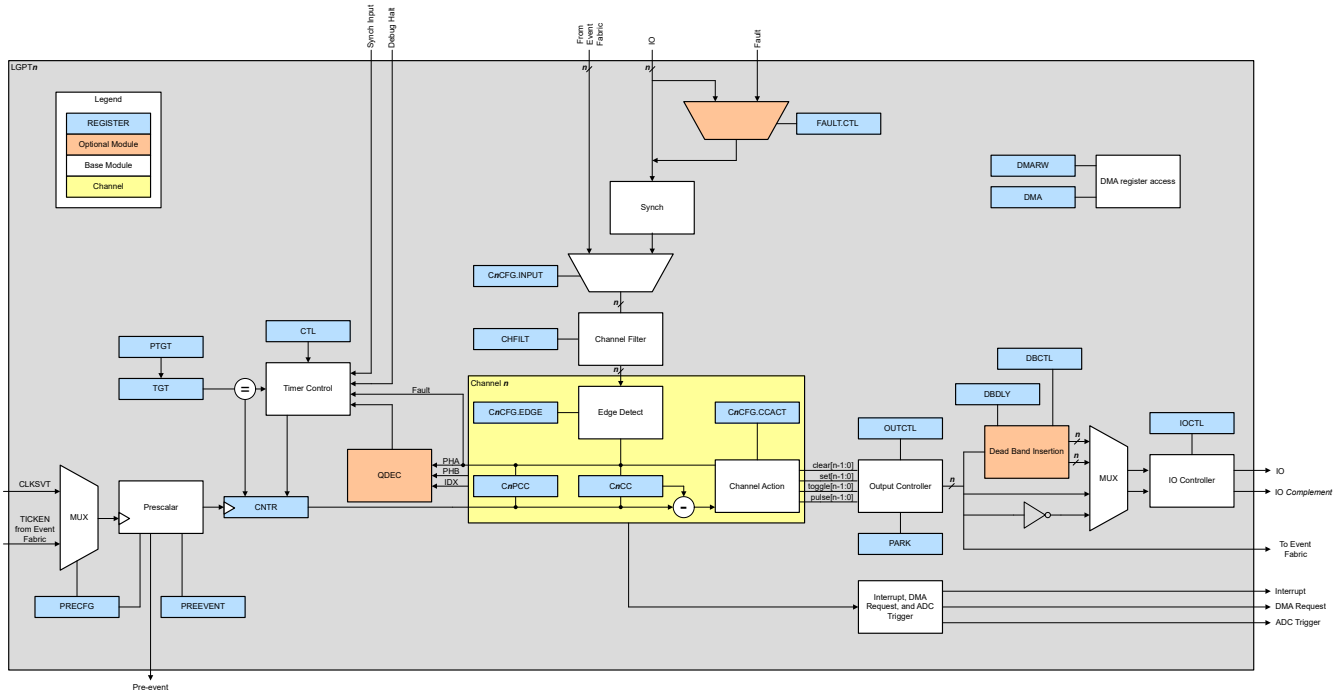


Figure 10-1. Single Timer Block Diagram

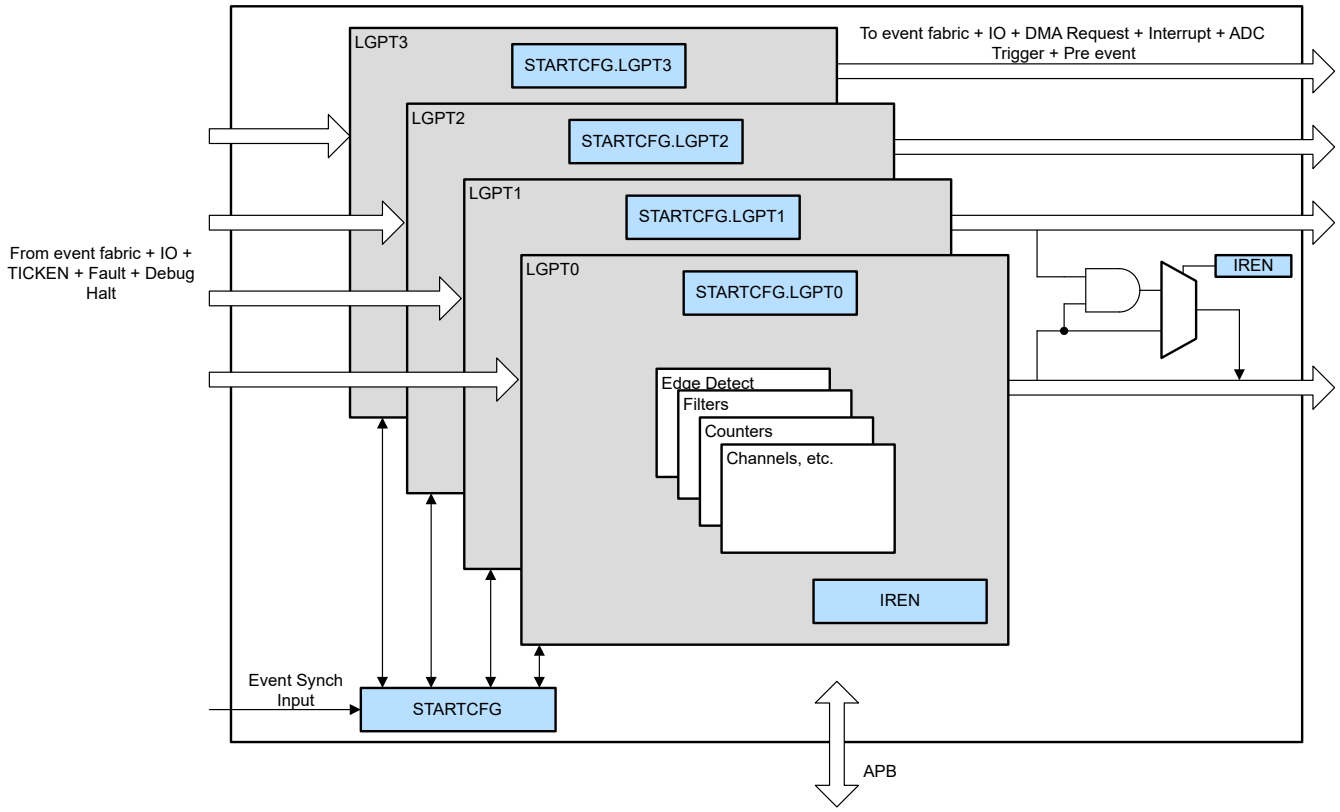


Figure 10-2. Multiple Timer Block Diagram

10.3 Functional Description

10.3.1 Prescaler

The prescaler is an 8-bit counter which counts down from the value PRECFG.TICKDIV to zero repeatedly. The rate of the down count is referred to as the prescaler clock. When the prescaler counter reaches zero, CNTR is updated. The rate of the CNTR update is referred to as the timer clock.

The prescaler can optionally run on the system clock (48MHz) or the TICKEN signal from the event fabric. This can be configured in the PRECFG.TICKSRC register field.

If the prescaler clock is configured to be the TICKEN event signal from the Event fabric, the source of the event must be subscribed by the EVTSVT.LGPT*x*TENSEL register.

The timer clock and prescaler clock determine the following:

- **Prescaler clock**
  - Timer clock
  - Prescaler event output update
  - Sampling of channel filter (optionally)
- **Timer clock**
  - CNTR rate
  - Channels update event outputs on this clock
  - Sampling of channel filter (optionally)
  - QDEC sampling

10.3.2 Counter

The value written to CTL.MODE[0:2] determines the counter mode as follows:

- UP_ONCE: The timer counts from 0 to the selected target. The timer then becomes disabled.
- UP_PER: The timer counts from 0 to the selected target, repeatedly.
- UPDWN_PER: The timer counts from 0 to the selected target and decrements back to 0, repeatedly.
- QDEC: The timer functions as a quadrature decoder. IOC input 0, IOC input 1, and IOC input 2 are used respectively as PHA, PHB, and IDX inputs. IDX can be turned off by setting CH2CCFG.EDGE = NONE.
- SYNC_UP_ONCE: Same as UP_ONCE but the timer synchronizes to another timer. Choose which timer in STARTCFG.
- SYNC_UP_PER: Same as UP_PER but the timer synchronizes to another timer. Choose which timer in STARTCFG.
- SYNC_UPDWN_PER: Same as SYNC_UPDWN_PER but the timer synchronizes to another timer. Choose which timer in STARTCFG.

Note

While the counter can be written, the intent is only to support the setting of an initial position in QDEC mode. The ability to write the counter in other modes while the timer is running is possible, but the resulting behavior is unpredictable.

10.3.3 Target

The Target TGT register sets the target value for the counter.

The Pipline Target PTGT register, if written to, is loaded into TGT on counter zero crossing.

The QDEC mode and the SYNC modes are further described in Section 10.4.1 and Section 10.3.9.

10.3.4 Channel Input Logic

Every channel has an input which is used when the channel is configured in a capture action. The channel input can come from different sources, can be filtered and goes through an edge detection logic before triggering the channel capture. See Figure 10-3 detailing the input logic.

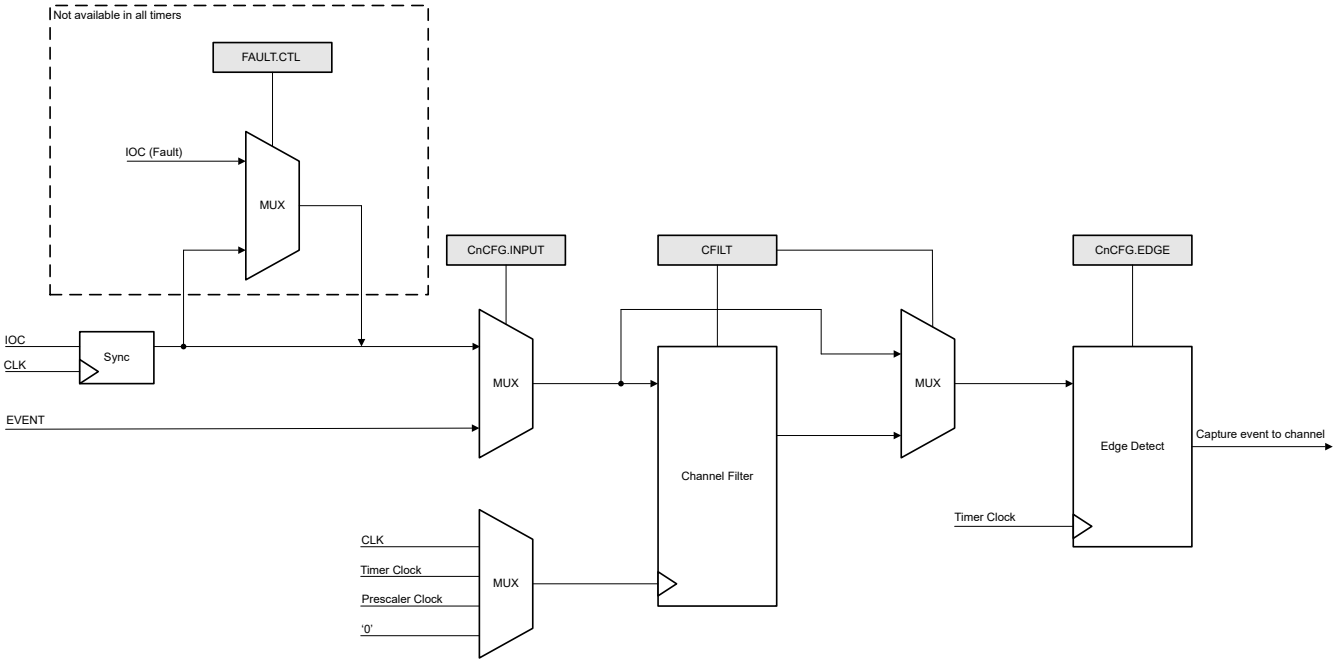


Figure 10-3. Channel Input Logic

If the FAULT register is present, then enabling the FAULT logic uses channel 0 as fault input. The CnCFG.INPUT field configures if the input comes from the IOC or Event logic. The channel filter can be configured to require up to CHFILT.LOAD +1 consecutive input samples before the input is propagated to the edge detection logic. This can typically be used to avoid capturing glitches. After the channel filter, the input goes into the edge detection logic. This is configured in the CnCFG.EDGE field.

The different components of the capture data path are clocked as follows:

- The asynchronous IOC inputs are synchronized at the system clock (48MHz).
- The *channel filter*, if used, is either clocked at the system clock, timer clock, or prescaler clock.
- The *edge detect* logic is always clocked at the timer clock.

10.3.5 Channel Output Logic

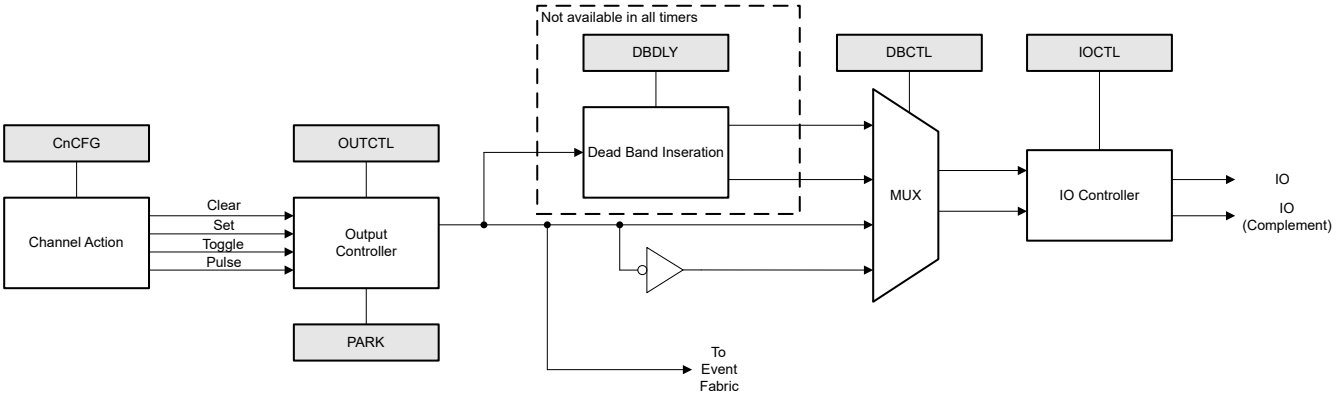


Figure 10-4. Channel Output Logic

Each timer has the same number of outputs as channels, but each channel does not control a dedicated output. Instead, every channel can control every output. Which output each channel controls is configured in the CnCFG.OUTn fields.

The user can set and clear timer outputs manually by writing OUTCTL. Manual update of an output takes priority over automatic channel updates of the same output. Listed in decreasing order of priority, each output can:

1. Clear
2. Set
3. Toggle
4. Pulse (The output remains high for two counter clock periods, then goes low.)

An output can receive update requests from several channels at the same time. In this case, the output is updated according to the priority list. The output updated from a channel is decided by the channel action, CnCFG.CCACT.

10.3.6 Channel Actions

Each channel implements 15 different channel actions. Configured in CnCFG.CCACT, actions are categorized as one-shot and continuous:

- A one-shot channel action performs a function only once before the timer disables the channel.
- A continuous channel action performs a function until the user disables the channel.

Table 10-1 lists the 15 channel actions.

Table 10-1. Channel Actions

LGPT.CnCFG[3:0] CCACT bit field	**Action**	**One-shot or Continuous**
0	Disable channel	One-shot
1	Set on capture, and then disable channel	One-shot
2	Clear on zero, toggle on compare, and then disable channel	One-shot
3	Set on zero, toggle on compare, and then disable channel	One-shot
4	Clear on compare, and then disable channel	One-shot
5	Set on compare, and then disable channel	One-shot
6	Toggle on compare, and then disable channel	One-shot
7	Pulse on compare, and then disable channel	One-shot
8	Period and pulse width measurement	Continuous
9	Set on capture repeatedly	Continuous
10	Clear on zero, toggle on compare repeatedly	Continuous
11	Set on zero, toggle on compare repeatedly	Continuous
12	Clear on compare repeatedly	Continuous
13	Set on compare repeatedly	Continuous
14	Toggle on compare repeatedly	Continuous
15	Pulse on compare repeatedly	Continuous

After configuration, the channel requests updates of enabled event outputs (set by CnCFG.OUT fields) according to the channel action description in the table above. There are three channel actions that require further description.

10.3.6.1 Period and Pulse Width Measurement

This channel action continuously captures the period and pulse width of the channel's input signal relative to the signal edge given by CnCFG.EDGE. The channel requests to set enabled events when CnCC.VALUE contains the signal period and PCnCC.VALUE contains signal pulse width. The channel function synchronizes the timer counter to the selected signal edge of the incoming signal. Hence:

- The counter restarts regularly, so other channel actions must be chosen with this in mind.

- The channels configured for this channel action cannot perform measurements simultaneously. The measurements are done in a time-interleaved manner.

Example: Two channels in Timer Period and Pulse Width Capture

The timer measures the signal period and pulse width of two different signals A (From IO Controller or Event Fabric) and B (From IO Controller or Event Fabric). See Chapter 18 and Section 4.3 for more information on configuring the I/O Controller and Event Fabric.

In this example, both signals have periods less than the counter range. Hence, time-out detection as described in the register documentation is not required. Configure as follows:

- Channel 0:
  - C0CFG. CCACT = PER_PULSE_WIDTH_MEAS
  - C0CFG. OUT0 = 1
  - C0CFG.INPUT = EVT/IO (Signal A)
  - C0CFG.EDGE = RISING
- Channel 1:
  - C1CFG. CCACT = PER_PULSE_WIDTH_MEAS
  - C1CFG. OUT1 = 1
  - C1CFG.INPUT = EVT/IO (Signal B)
  - C1CFG.EDGE = FALLING
- Timer:
  - CTL.MODE = UP_PER

Figure 10-5 shows how the timer counter first synchronizes to signal A. Channel 0 then captures the high phase of signal A into PC0CC at time t_0.The period of signal A is captured in C0CC at time t_1. At the same time, Channel 0 sets the event output 0 high, and the timer counter starts to synchronize with signal B. Channel 1 then captures the low phase of signal B into PC1CC at time t_2. Finally, the period of signal B is captured in C1CC at time t_3. At the same time, channel 1 sets the event output 1 high, and the timer counter starts to synchronize to signal A. The sequence then repeats itself until stopped by the user.

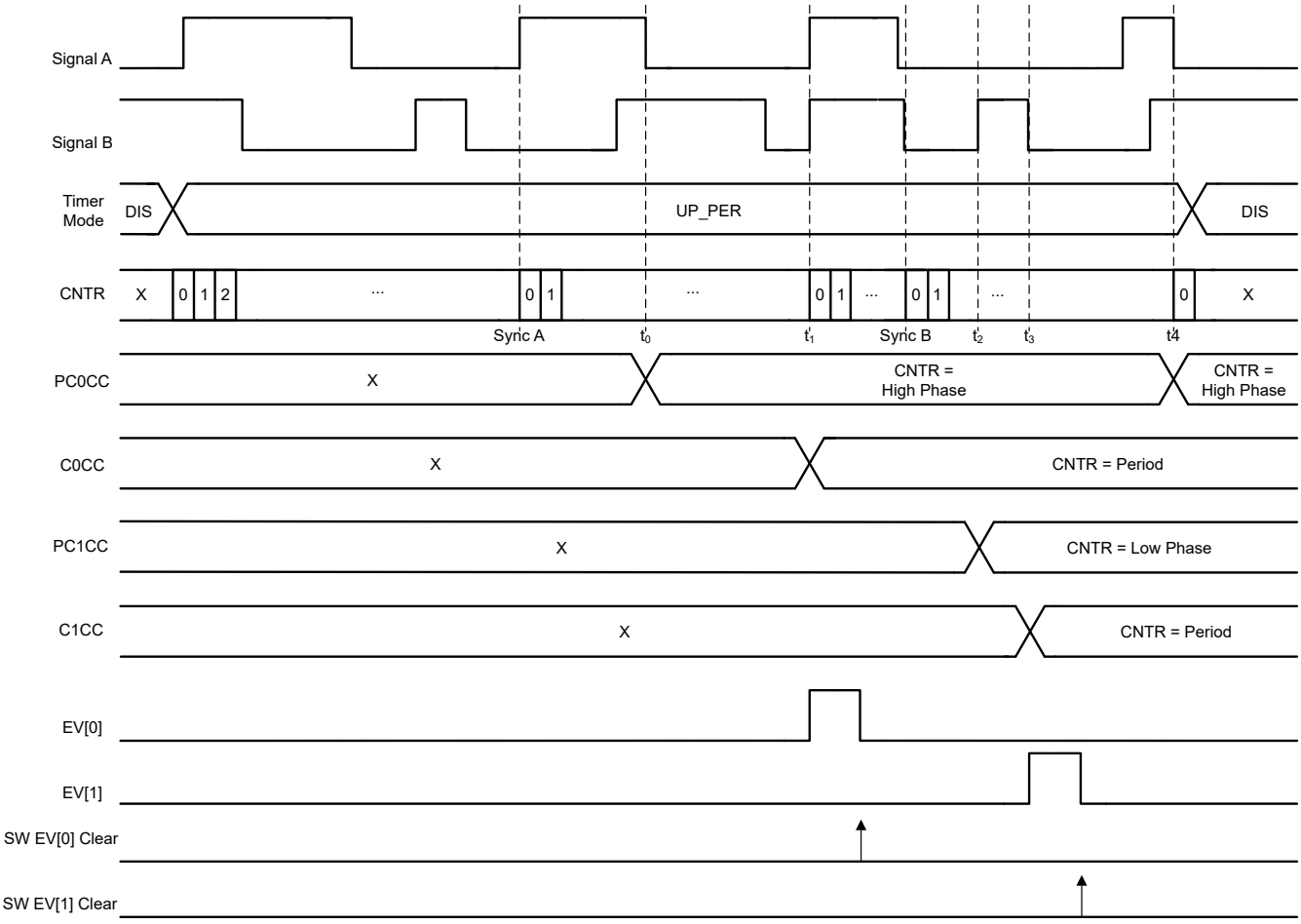


Figure 10-5. Period Pulse Width Measurement

10.3.6.2 Clear on Zero, Toggle on Compare Repeatedly

This channel action continuously:

- Clears the enabled output events when CNTR = 0
- Toggles the enabled output events when CNTR = CnCC

The channel generates a center-aligned PWM waveform when CTL.MODE = UPDWN_PER. The channel copies a new value written in PCnCC to CnCC when CNTR becomes 0. This action prevents jitter on the edges of the generated PWM signal. Similarly, the timer copies a new value written in PTGT to TGT when CNTR becomes 0. This action avoids period-jitter in PWM applications with time-varying periods.

Example: Center-Aligned PWM Generation by Channel 0

This example illustrates center-aligned PWM generation by channel 0. The waveform is synthesized on output 0. The timer period is kept static, and the target value is set to half the period. Configure as follows:

- Channel 0:
  - C0CFG.CCACT = CLR_ON_0_TGL_ON_CMP
  - C0CFG.OUT0 = 1
  - C0CC = C0
- Timer:
  - TARGET = PERIOD / 2
  - CTL.MODE = UPDWN_PER

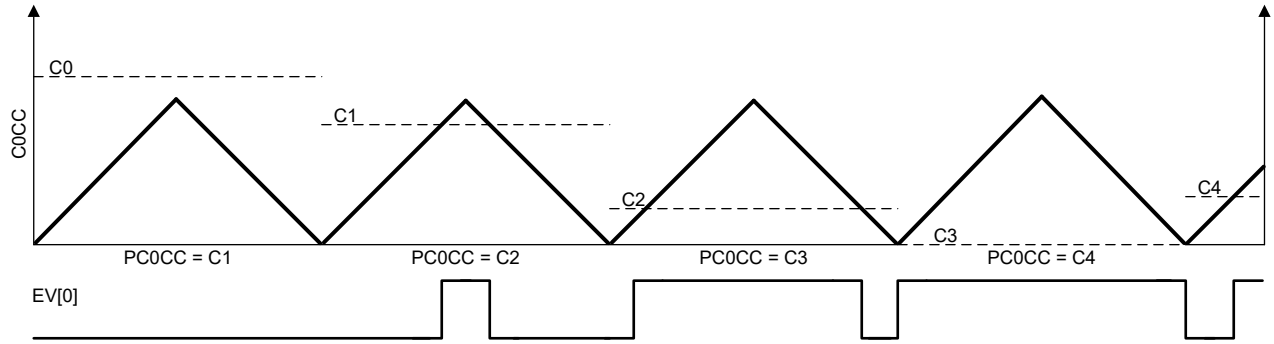


Figure 10-6. Center-Aligned PWM

The duty-cycle of the generated PWM waveform is controlled by PC0CC updates. Waveform generation on event output 0 continues until aborted by the user.

If the user wants to stop/pause the PWM generation in a controlled manner, the following procedure is recommended:

- Set FAULT.CTL = ZEROCOND
- Then set ISET.FAULT. The counter then halts at CNTR = 0.
- The counter can then either be started again by clearing RIS.FAULT, or turned completely off by setting CTL.MODE = DIS.

10.3.6.3 Set on Zero, Toggle on Compare Repeatedly

This channel action continuously does the following:

- Sets the enabled output events when CNTR = 0
- Toggles the enabled output events when CNTR = CnCC

The channel generates an edge-aligned PWM waveform when CTL.MODE = UP_PER. The channel copies a new value written in PCnCC to CnCC when CNTR becomes 0. This prevents jitter on the edges of the generated PWM signal. Similarly, the timer copies a new value written in PTGT to TGT when CNTR becomes 0. This avoids period-jitter in PWM applications with a time-varying period.

Example: Edge-Aligned PWM Generation by Channel 0

This example illustrates edge-aligned PWM generation by channel 0 (see below). The waveform is synthesized on event output 0. The timer period is kept static, and the target value is set to period minus 1. Configure as follows:

- Channel 0:
  - C0CFG.CCACT = SET_ON_0_TGL_ON_CMP
  - C0CFG. OUT0 = 1
  - C0CC = C0
- Timer:
  - TGT = PERIOD - 1
  - CTL.MODE = UP_PER

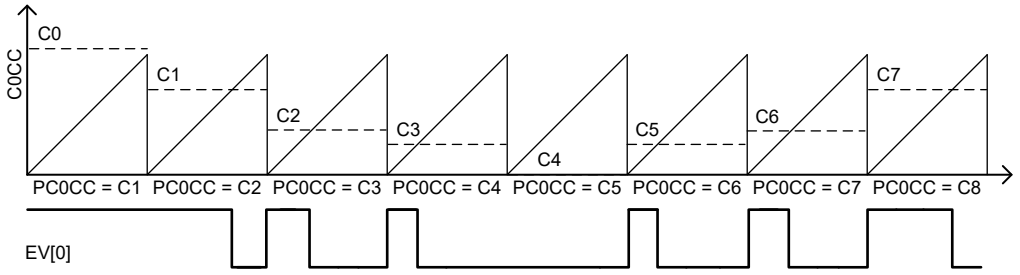


Figure 10-7. Edge-Aligned PWM

The duty-cycle of the generated PWM waveform is controlled by PC0CC updates. Waveform generation on event output 0 continues until aborted by the user.

10.3.7 Channel Capture Configuration

The channel's capture source can be set by the LGPT.CnCFG[6] INPUT bit field. Here the bit field INPUT sets either the synchronous event from the event fabric or the asynchronous IOC inputs as the capture source. See Chapter 18 and Section 4.3 for more information on configuring the I/O controller and event fabric.

10.3.8 Channel Filters

The channel filter sets a window within which the input must remain stable; otherwise, the transition is not passed to the edge detection logic.

The channel filter counts down from CHFILT.LOAD[15:8] while two consecutive input samples are equal. If two consecutive input samples are unequal the filter counter is reloaded with LOAD. If the channel filter reaches zero the input is passed to the edge detection logic. The filter delays the input signal by at least LOAD + 1 filter clock cycles.

When writing CTL.MODE to any other value than disabled (0x0) the internal channel filter counter is loaded with the LOAD value. Do not change the CHFILT register while the timer is running and CTL.MODE[2:0] is not disabled (0x0).

10.3.8.1 Setting up the Channel Filters

To set up the channel filters in LGPT follow these steps:

- Set the channel filter sample period by setting CHFILT.MODE:
  - CHFILT.MODE = BYPASS: No filter is used.
  - CHFILT.MODE = CLK: 48MHz sample rate (CLKSVT)
  - CHFILT.MODE = TICKCLK: Sample same as PRECFG.TICKSRC
  - CHFILT.MODE = TIMERCLK: The sample rate is the same as the counter rate.
- Set the CHFILT.LOAD value.
- Start LGPT.

When configuring the channel filter make sure the channel filter period is not shorter than the timer clock period. That is,

(Channel filter clock period) × (CHFILT.LOAD) > timer clock period.

If this is not satisfied capture events can be missed.

10.3.9 Synchronize Multiple LGPT Timers

The STARTCFG register can be used to synchronize multiple timers. The LGPT timers are configured in the following manner to synch to LGPT0:

- Configure LGPT1.STARTCFG = 1 and LGPT1.CTL.MODE in one of the SYNC modes.
- Configure LGPT2.STARTCFG = 1 and LGPT2.CTL.MODE in one of the SYNC modes.
- Configure LGPT3.STARTCFG = 1 and LGPT3.CTL.MODE in one of the SYNC modes.
- Start LGTP0 in the wanted mode.

Table 10-2. Sync Modes

LGPTn.CTL[2:0] MODE bit field	SYNC MODE
0	Disable
1	Not Synchronous
2	Not Synchronous
3	Not Synchronous
4	Not Synchronous
5	Count Up Once - Synchronous to another LGPT
6	Count Up Periodically - Synchronous to another LGPT
7	Count Up and Down Periodically - Synchronous to another LGPT

The timers are started when LGPT0 starts by changing the CTL.MODE field from SYNC mode to the respective active mode.

10.3.10 Interrupts, ADC Trigger, and DMA Request

Each timer can generate an Interrupt request, ADC trigger, and DMA request output event. These output events can be triggered on different internal timer events. The different internal events can be viewed in the RIS register.

If one or more of the fields in the IMASK register are set high, the timer sends out an interrupt event when the internal timer event corresponding to the fields set in IMASK occurs.

These internal timer events can also set the ADC trigger or DMA request depending on the configuration of the ADCTRG.SRC and DMA.REQ field. That is, the ADC trigger and DMA request output events are generated when the corresponding interrupt is set in the RIS register.

Below are some important side effects regarding the Interrupts, ADC trigger, and DMA request.

- Reading/writing to the CnCC or PCnCC register shall clear the corresponding channel interrupt.
- Reading/writing to NC (No Clear) registers does not have any side effects on interrupts.
- Reading/writing to PTGT or TGT clears both RIS.ZERO and RIS.TGT.
- The CMP interrupts are updated on the timer clock. These interrupts trigger at the same time as a CMP event.
- The ZERO and TGT interrupts trigger after one system clock cycle when CNTR = ZERO/TARGET.
- The field CTL.INTP gets set when the ZERO and TGT interrupts are set. This field decides if the interrupts are set at the beginning of the timer period or at the end of the timer period.
- The RIS.ZERO interrupt is not set when starting the timer.
- Note that if you have a short timer period, and you have configured the LGPT to set the interrupt output on both ZERO and TGT, you can accidentally clear both ZERO and TGT when reading/writing to PTGT/TGT. This depends on the CPU's response time. To ensure the ZERO and TGT interrupts are received, use PTGTNC/TGTNC and clear the corresponding interrupt by writing to RIS.ICLR.
- If the µDMA request is used in addition to the interrupt make sure that a write/read does not clear unhandled interrupt requests. As an example, if the µDMA updates PTGT on ZERO interrupt and the CPU does some external handling on TGT interrupt, then if the timer period is short, the µDMA write to PTGT can clear the unhandled TGT interrupt. This can be avoided by letting the µDMA write to PTGTNC and letting the CPU clear both TGT and ZERO.

10.4 Timer Modes

10.4.1 Quadrature Decoder

The Phase A (PHA), Phase B (PHB) and IDX signals are input events of channel 0, channel 1, and channel 2, respectively. PHA and PHB are required and IDX is optional. The signals are typically provided from an incremental encoder. An incremental encoder can provide two outputs which indicate a linear or a rotary motion. The output of the incremental encoder is typically a 90° shifted square wave and is provided as an input to the LGPT. When enabled in QDEC mode, the LGPT is used to decode the quadrature-encoded data to provide

information on the relative positioning and movement of a linear or rotary motion. The accumulation of the counter value in LGPT with respect to PHA/PHB follows the following table:

Table 10-3. Counter Accumulation Based on QDEC Inputs

Previous Pin Event	Current Pin Event	Counter (+ or -)	Direction
PHA Falling	PHB Rising	-	Down
PHA Falling	PHB Falling	+	Up
PHA Falling	PHA Rising	+ if new direction is up, - if new direction is down	Toggle
PHA Rising	PHB Rising	+	Up
PHA Rising	PHB Falling	-	Down
PHA Rising	PHA Falling	+ if new direction is up, - if new direction is down	Toggle
PHB Falling	PHA Rising	+	Up
PHB Falling	PHA Falling	-	Down
PHB Falling	PHB Rising	+ if new direction is up, - if new direction is down	Toggle
PHB Rising	PHA Rising	-	Down
PHB Rising	PHA Falling	+	Up
PHB Rising	PHB Falling	+ if new direction is up, - if new direction is down	Toggle

To setup LGPT in QDEC mode follow these steps:

- Configure where the input is from, IOC or event fabric. Normally this is IOC:
  - PHA: C0CFG.INPUT = 1 (IOC)
  - PHB: C1CFG.INPUT = 1 (IOC)
- (Optional) IDX: C2CFG.INPUT = 1 (IOC)
- (Optional) If IDX is used set C2CFG.EDGE != 0.
- Set the sample rate. Use PRECFG.TICKDIV/TICKSRC to set the sample rate. The sample rate is:
  - TICKSRC = 0: 48MHz / (PRECFG.TICKDIV + 1)
  - TICKSRC !=0: (ticken freq) / (PREVFG.TICKDIV + 1)
- Configure TGT and PTGT.
- (Optional) Enable channel filters. See Section 10.3.8

  .
- (Optional) Enable QDEC interrupts:
  - (Optional) Set IMASK.DBLTRANS to get the interrupt when a double transition occurs. This indicates that the sampling rate is too low.
  - (Optional) Set IMASK.CNTRCHNG to get the interrupt when the counter changes. This indicates a movement of the measuring device.
  - (Optional) Set IMASK.DIRCHNG to get the interrupt when the direction of the counter changes.
- (Optional) Configure PREEVENT if you want a high output signal just before the sampling; for example, turn on an LED.
- Start timer in QDEC by writing CTL.MODE= QDEC_MODE.

Figure 10-8 shows the QDEC-related signals with PREEVENT.

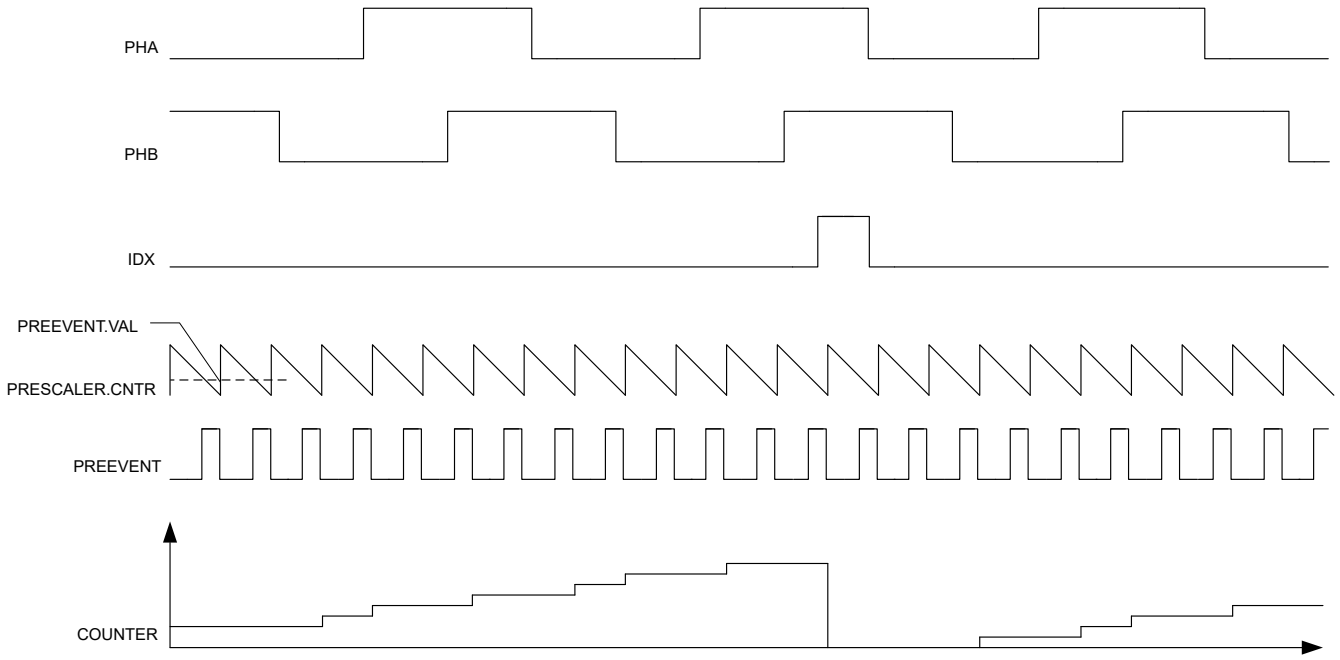


Figure 10-8. QDEC Example with PREEVENT

10.4.2 DMA

The register DMA is used to enable DMA requests. The DMA.REQ field sets which interrupt event generates a DMA request. The request is a pulse (one system clock period) which is generated when the corresponding interrupt is set in the RIS register. Upon a DMA request defined by REQ an internal address pointer is set to RWADDR*4. Every access to DMARW increments the internal pointer by 4 such that the next DMA access is to the next register.

The internal pointer stops after RWCNTR increments. Further access is ignored.

Below is an example of how to setup DMA and DMARW for updating PTGT, PC0CC, PC1CC when the counter hits zero.

- Set DMA.REQ = 2
- Set DMA.RWADDR = 63 (The PTGT address is 63×4=0xFC)
- Set DMA.RWCNTR = 2 (The PC1CC address is 63×4 + 2×4 = 0x104)
- Start timer in required mode.
- Upon receiving the DMA request the DMA should make at least three writes to DMARW, which will be as if writing to PTGT, PC0CC then PC1CC. Any other read/writes to DMARW will be ignored.

10.4.3 IR Generation

By using LGPT0 and LGPT1 the timers can generate IR codes with minimal software interaction. By enabling IRGEN.CTL in LGPT0 an AND gate is set between T0C0 and T1C0, the output of this AND gate replaces T0C0. In IR generation mode LGPT0 generates the carrier wave, while LGP1 works as the modulator.

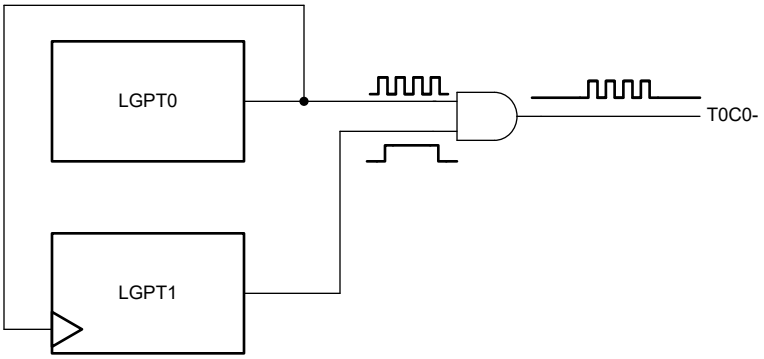


Figure 10-9. LGPT IR Generation

Below is an example on how to set up the timers for IR with a 38 kHz carrier wave.

LGPT0 (Carrier wave generation):

- Set the PRECFG.TICKDIV and TGT to fulfill the following equation:

(48MHz)/((PRECFG.TICKDIV+1) × (TGT+1)) = Wanted Frequency (for example, 38kHz)

For example, PRECFG.TICKDIV = 2 and TGT 420. This gives a carrier wave of approximately 38kHz (Theoretically 38.00475kHz).

- Set C0CC = 210 to give roughly 50% duty cycle.
- Set C0CFG.CCACT = TGL_ON_CMP.
- Set IRGEN.CTL = 1.
- Start timer in up-periodic. CTL.MODE = 2

Configure EVTSVT

- Set EVTSVT.LGPT1TENSEL = LGPT0C0.

LGPT1 (Modulator, example for the NEC protocol)

- Set PRECFG.TICKSRC = FALL_TICK
- Set C0CFG.CCACT = SET_ON_0_TGL_ON_CMP, C0CFG.OUT0 = 1
- Set TGT to the length of the symbol.
- Set C0CC to the number of 38kHz pulses in the symbol.
- Update PTGT on RIS.ZERO interrupt to send a new symbol.

10.4.4 Fault and Park

If LGPTn.DESCEX[18] HBDF (Has deadband, Fault, and Park logic) bit is set to one, the LGPT implements the FAULT and PARK registers. The FAULT register is used to stop the timer upon an active fault input signal from the IOC. The PARK register can be used to set the IOC outputs of the LGPT to a given state when the timer has stopped as a result of fault.

The fault input overrides channel 0 IOC input when FAULT.CTL != DIS. See Figure 10-3.

This means that channel 0 receives fault as input signal when C0CFG.INPUT = IO and FAULT.CTL != DIS. CHFILT can be used to avoid glitching on the fault input. Fault is level triggered, the polarity is set by the C0CFG.EDGE field. Here C0CFG.EDGE = RISE gives active high and C0CFG.EDGE = FALL gives active low polarity.

Set the Fault mode by setting FAULT.CTL. There are four different modes:

- **DIS**: The counter ignores the fault input.
- **IMMEDIATE**: In this mode the counter stops immediately on an active fault input (2 system clock cycles of synch delay is expected). This is done by hardware by setting CTL.MODE = DIS. To start the counter software must set CTL.MODE != DIS. The RIS.FAULT interrupt is also set immediately on active fault input. If

the RIS.FAULT input is cleared, it will not be set again while CTL.MODE == DIS even though the fault input is active. This is because the 2 stage synchronizers and the channel filter is not active while CTL.MODE == DIS. If the counter is started by setting CTL.MODE != DIS when the fault input is active it will immediately stop the counter and set RIS.FAULT.

- **ZEROCOND**: In this mode the counter stops when CNTR = 0 after an active fault input. If the RIS.FAULT flag has been cleared by software before CNTR= 0, and the fault input is inactive, the counter will continue as normal. When the counter stops on zero, it can be started again by clearing the RIS.FAULT flag. If you want to change the counter mode you should set CTL.MODE = DIS, clear the RIS.FAULT interrupt, then start timer in required mode. The channel filter will keep running while the CNTR is halted at zero. This ensures an up-to-date fault input.
- **IRQ**: In this mode only the RIS.FAULT interrupt is set on active fault.

General notes on Fault:

- When the channel filter is used together with Fault, and the timer is restarted, the fault input will not be evaluated before CHFILT.LOAD number of consecutive equal samples. This means that an active fault could have been present when the timer started (by setting CTL.MODE != DIS), but the timer will not stop before CHFILT.LOAD samples.
- In IMMEDIATE mode clear RIS.FAULT to start the timer again. Not doing so will immediately stop the timer upon start.

Example setup of Fault and Park.

- Set FAULT.CTL to wanted mode.
- Set C0CFG.EDGE = RISE/FALL. Here RISE = active high, and FALL = active low.
- Set C0CFG.INPUT = IO.
- (Optional) Config CHFILT
- (Optional) Config IMASK to set CPU interrupt on fault. Typically required.
- (Optional) Config PARK.
- Start timer in wanted mode.

Figure 10-10 and Figure 10-11 illustrate the consequences of some of the possible Fault and Park configurations.

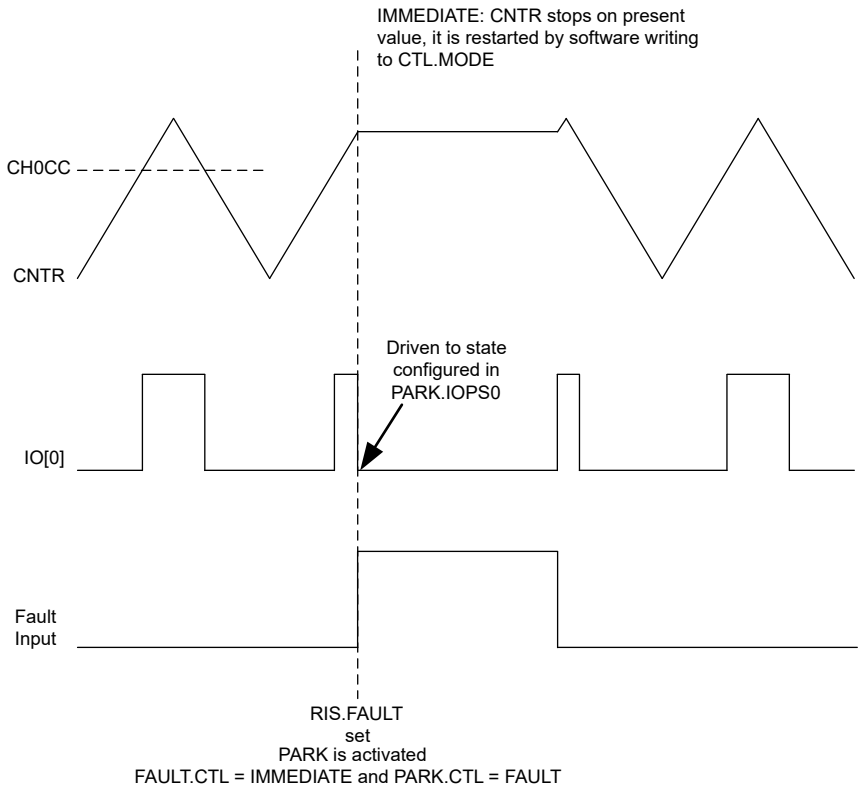


Figure 10-10. Fault and Park - Immediate Mode

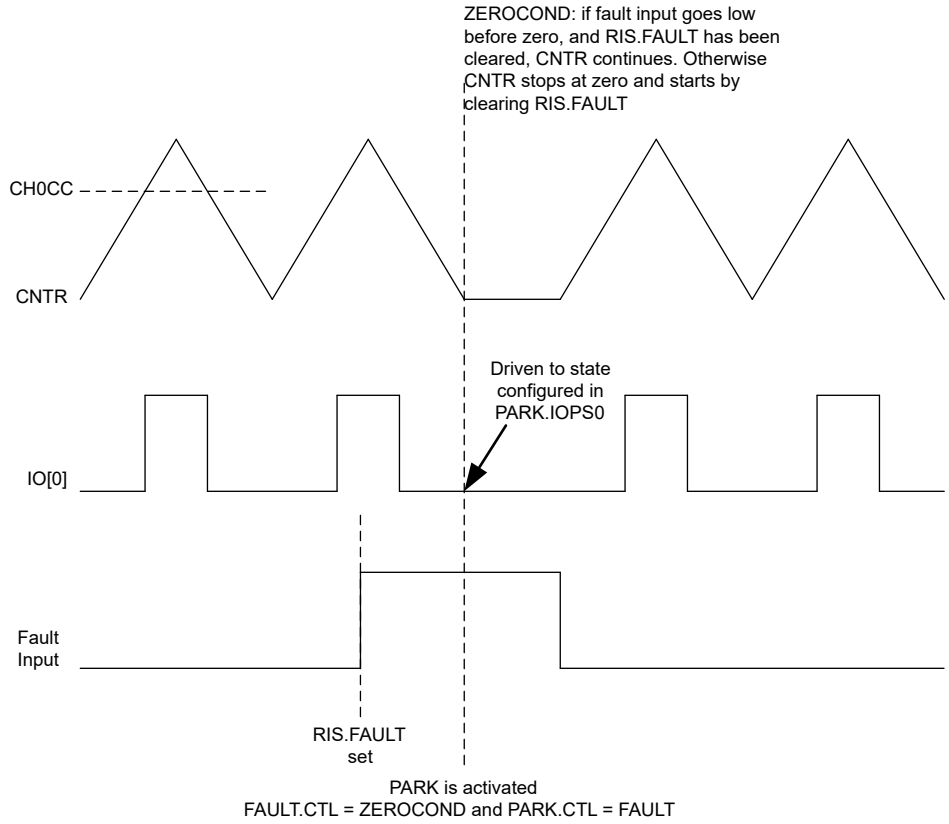


Figure 10-11. Fault and Park - Zero Condition

10.4.5 Deadband

If the SYS_HDBF=1 the LGPT can optionally insert a deadband transition in a reference PWM signal. Deadband insertion is accomplished by taking a reference pulse width modulated signal and generating two pulse width modulated signals (IO*n* and IO_C*n*) of the same frequency but with a deadband period inserted between the signals. This is shown in Figure 10-12.

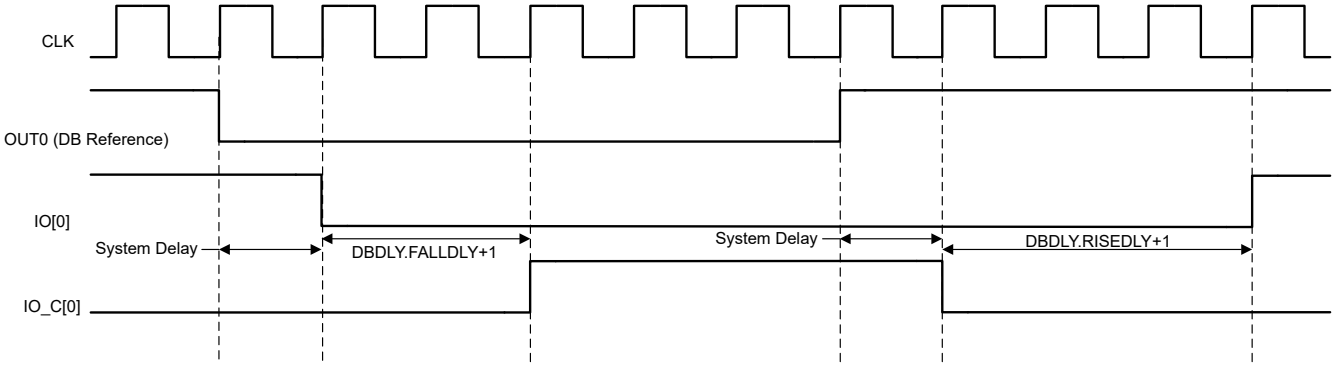


Figure 10-12. Deadband Insertion

As shown in Figure 10-12 RISEDLY and FALLDLY fields from the DBDLY register are added with a value of 1 during deadband insertion. Both IO and IO_C signals are also one system clock cycled delayed due to the deadband insertion logic.

Note

- Configuring RISEDLY longer than or equal to the pulse width of the reference signal results in a constant low IO output.
- Configuring FALLDLY longer than or equal to the low pulse width of the reference signal results in a constant low IO_C output.

Example Setup of Deadband on IO0 and IO_C0

- Configure PWM output as required on channel 0. See Section 10.3.6.2
- Set PWM output to be generated on output 0 by setting C0CFG.OUT0 = 1
- Set wanted fall/rise delay by setting DBDLY.RISEDLY and DBDLY.FALLDLY.
- Set DBCTL to generate a deadband on IO0 and IO_C0 by setting DBCTL.IO0 = 1.
- Start the timer by writing to CTL.

10.4.6 Deadband, Fault, and Park

An important feature to maintain when a deadband is used together with fault and park is to never make a switch on the IO outputs without deadband insertion. When the park values for IO and IO_C are opposite this switch is trivial, as this corresponds to a switch done under normal PWM deadband operation. See Figure 10-13.

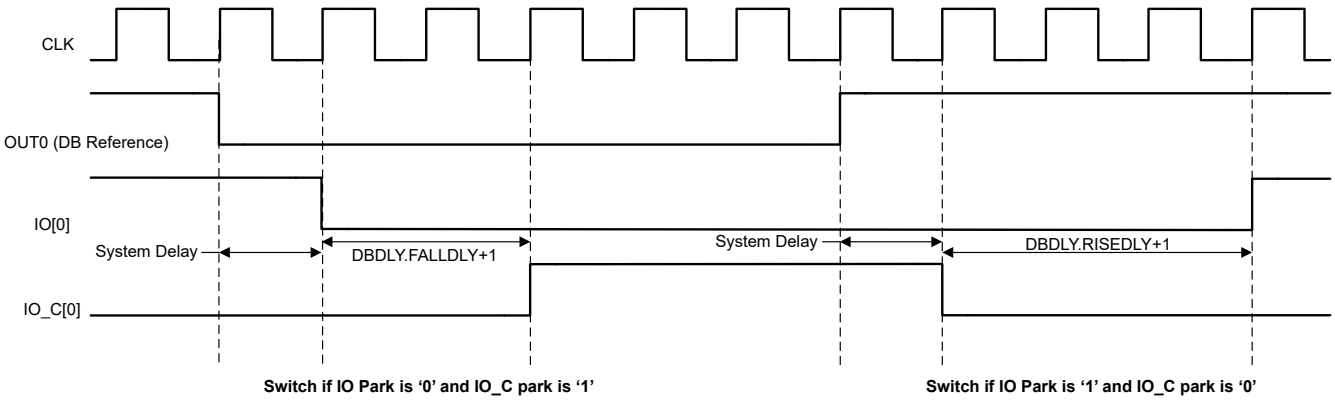


Figure 10-13. Deadband Switch to Opposite Park States

If the IO and IO_C park values are equal the switch is not trivial. To maintain a deadband insertion between switches, the implementation principle is as follows:

- When park is activated (either from fault or debug) the deadband reference input is set to IO park state immediately.
- This will ensure that the IO output gets set to park state after a fall/rise delay.
- When the delay is finished, meaning that the IO output is in the park state, this output is "locked" to the park state. The deadband logic does not control this output anymore.
- At the same time as the IO output is locked, the deadband reference signal is switched to the inverse of the IO_C park state. This then sets IO_C to the IO_C park state after a fall/rise delay.
- As the IO output is locked, the output does not change as the deadband reference signal changes.

When using this method, the IO and IO_C outputs can use (FALLDLY+RISEDLY+2) cycles before settling in the park state when the IO and IO_C outputs are equal. The IO and IO_C outputs use FALLDLY+1 or RISEDLY+1 to settle when the IO and IO_C outputs are opposite.

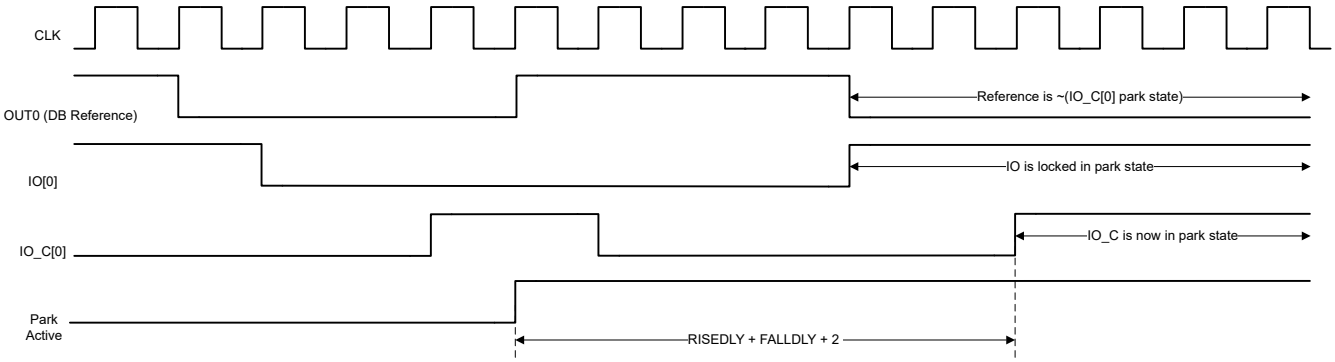


Figure 10-14. Deadband Switch to Equal Park States

In Figure 10-14 Park active is an internal signal set by either Fault or Debug depending on the configuration. Park active is set one clock cycle after Fault or Debug is registered.

10.4.7 Example Application: Brushless DC (BLDC) Motor

The LGPT can be used to drive a BLDC motor. Consider a BLDC motor with three half bridges, this requires three or more channels. In this scenario, the three IOC and the three IOC complementary outputs of one LGPT are connected to a motor driver or directly to the half bridges. See Figure 10-15.

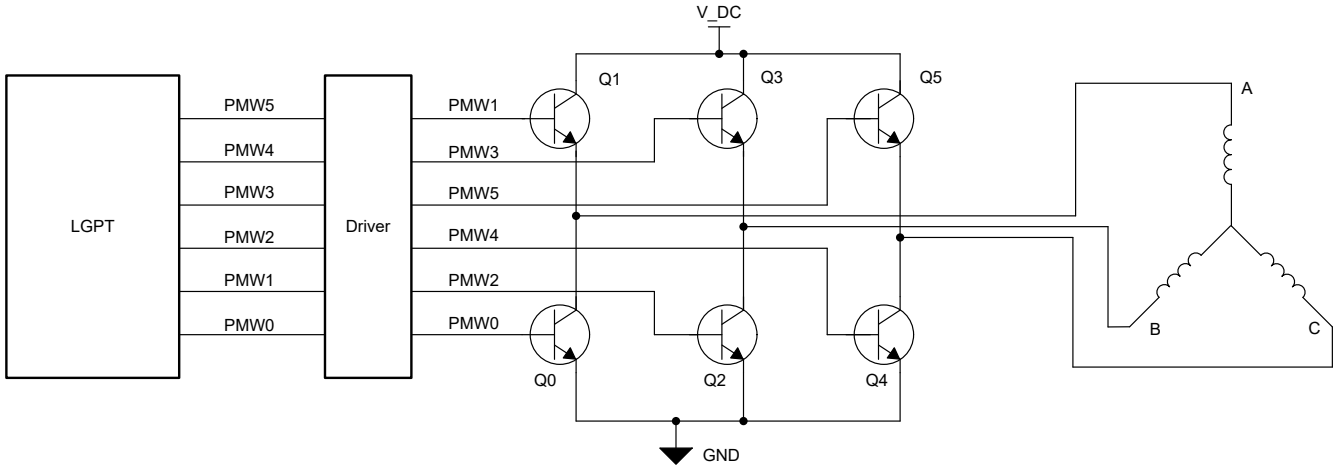


Figure 10-15. BLDC Application Example

Here each half bridge is controlled by an IOC and IOC complementary pair, that is, PWM0 and PWM1 correspond to IOC[0] and IOC_C[0], PWM2 and PWM3 to IOC[1] and IOC_C[1], and so on. To operate the motor in a basic fashion current is driven through two of the inductors at a time in a sequential pattern. This is done by switching the transistors in a distinct pattern, see Figure 10-16.

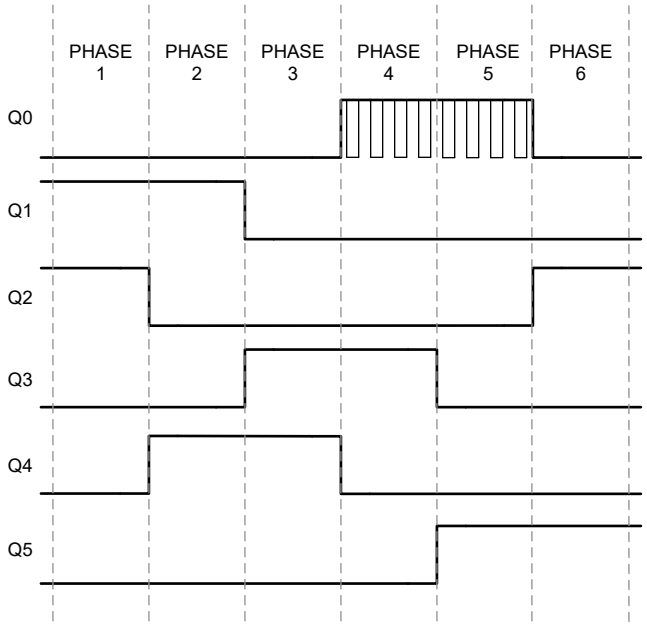


Figure 10-16. Example of Transition Phases to Drive a BLDC Motor

Notice that the high time of each transistor consists of a PWM signal (as illustrated in Q0). The duty cycle of the PWM signal corresponds to the current that is driven through the inductors and consequently the motor speed/load. The PWM can be generated as mentioned in Section 10.3.6.2. The software interactions required to operate the motor (assuming Q0 is connected to IOC[0] and Q1 is connected to IOC_C[0], and so on) are as follows:

SW operations:

- Phase 1: Configure IOCTL to invert IOC_C[0] (Q1) and let IOC[1] (Q2) out. All other outputs are configured low.
- Phase 2: Configure IOCTL to invert IOC_C[0] (Q1) and let IOC[2] (Q4) out. All other outputs are configured low.

- Phase 3: Configure IOCTL to invert IOC_C[1] (Q3) and let IOC[2] (Q4) out. All other outputs are configured low.
- Phase 4: Configure IOCTL to invert IOC_C[1] (Q3) and let IOC[0] (Q0) out. All other outputs are configured low.
- Phase 5: Configure IOCTL to invert IOC_C[2] (Q5) and let IOC[0] (Q0) out. All other outputs are configured low.
- Phase 6: Configure IOCTL to invert IOC_C[2] (Q5) and let IOC[1] (Q2) out. All other outputs are configured low.

Software needs a signal to determine when to change between the different phases, this can for example be done by an ADC measuring the back (EMF) Electromotive Force at the inactive inductor. The software can also change phases only on a ZERO interrupt from LGPT to ensure complete PWM pulses during phase changes.

10.5 LGPT0 Registers

Table 10-4 lists the memory-mapped registers for the LGPT0 registers. All register offset addresses not listed in Table 10-4 should be considered as reserved locations and the register contents should not be modified.

Table 10-4. LGPT0 Registers

Offset	Acronym	Register Name	Section
0h	DESC	Description Register.	Go
4h	DESCEX	Description Extended	Go
8h	STARTCFG	Start Configuration	Go
Ch	CTL	Timer Control	Go
10h	OUTCTL	Output Control	Go
14h	CNTR	Counter	Go
18h	PRECFG	Clock Prescaler Configuration	Go
1Ch	PREEVENT	Prescaler Event	Go
20h	CHFILT	Channel Input Filter	Go
34h	QDECSTAT	Quadrature Decoder Status	Go
38h	IRGEN	IR Generation	Go
3Ch	DMA	Direct Memory Accsess	Go
40h	DMARW	Direct Memory Access	Go
44h	ADCTRG	ADC Trigger	Go
48h	IOCTL	IO Controller	Go
68h	IMASK	Interrupt mask.	Go
6Ch	RIS	Raw interrupt status.	Go
70h	MIS	Masked interrupt status.	Go
74h	ISET	Interrupt set register.	Go
78h	ICLR	Interrupt clear register.	Go
7Ch	IMSET	Interrupt mask set register.	Go
80h	IMCLR	Interrupt mask clear register.	Go
84h	EMU	Debug control	Go
C0h	C0CFG	Channel 0 Configuration	Go
C4h	C1CFG	Channel 1 Configuration	Go
C8h	C2CFG	Channel 2 Configuration	Go
FCh	PTGT	Pipeline Target	Go
100h	PC0CC	Pipeline Channel 0 Capture Compare	Go
104h	PC1CC	Pipeline Channel 1 Capture Compare	Go
108h	PC2CC	Pipeline Channel 2 Capture Compare	Go
13Ch	TGT	Target	Go
140h	C0CC	Channel 0 Capture Compare	Go
144h	C1CC	Channel 1 Capture Compare	Go
148h	C2CC	Channel 2 Capture Compare	Go
17Ch	PTGTNC	Pipeline Target No Clear	Go
180h	PC0CCNC	Pipeline Channel 0 Capture Compare No Clear	Go
184h	PC1CCNC	Pipeline Channel 1 Capture Compare No Clear	Go
188h	PC2CCNC	Pipeline Channel 2 Capture Compare No Clear	Go
1BCh	TGTNC	Target No Clear	Go
1C0h	C0CCNC	Channel 0 Capture Compare No Clear	Go
1C4h	C1CCNC	Channel 1 Capture Compare No Clear	Go

Table 10-4. LGPT0 Registers (continued)

Offset	Acronym	Register Name	Section
1C8h	C2CCNC	Channel 2 Capture Compare No Clear	Go

Complex bit access types are encoded to fit into small table cells. Table 10-5 shows the codes that are used for access types in this section.

Table 10-5. LGPT0 Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

10.5.1 DESC Register (Offset = 0h) [Reset = DE491010h]

DESC is shown in Table 10-6.

Return to the Summary Table.

Description Register.
This register provides IP module ID, revision information, instance index and standard MMR registers offset.

Table 10-6. DESC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	MODID	R	DE49h	Module identifier used to uniquely identify this IP.
15-12	STDIPOFF	R	1h	Standard IP MMR block offset. Standard IP MMRs are the set of from aggregated IRQ registers till DTB. 0: Standard IP MMRs do not exist 0x1-0xF: Standard IP MMRs begin at offset of (64*STDIPOFF from the base IP address)
11-8	INSTIDX	R	0h	IP Instance ID number. If multiple instances of IP exist in the device, this field can identify the instance number.
7-4	MAJREV	R	1h	Major revision of IP.
3-0	MINREV	R	0h	Minor revision of IP.

10.5.2 DESCEX Register (Offset = 4h) [Reset = 000A38C3h]

DESCEX is shown in Table 10-7.

Return to the Summary Table.

Description Extended
This register describes the parameters of the LGPT.

Table 10-7. DESCEX Register Field Descriptions

Bit	Field	Type	Reset	Description
31-20	RESERVED	R	0h	Reserved
19	HIR	R	1h	Has IR logic.
18	HDBF	R	0h	Has deadband, Fault, and Park logic.
17-14	PREW	R	8h	Prescale width. The prescaler can maximum be configured to 2^{PREW}-1.
13	HQDEC	R	1h	Has Quadrature Decoder.
12	HCIF	R	1h	Has channel input filter.
11-8	CIFS	R	8h	Channel input filter size. The prevailing state filter can maximum be configured to 2^{CIFS}-1.
7	HDMA	R	1h	Has uDMA output and logic.
6	HINT	R	1h	Has interrupt output and logic.
5-4	CNTRW	R	0h	Counter bit-width. The maximum counter value is equal to 2^{CNTRW}-1. 0h = 16-bit counter. 1h = 24-bit counter. 2h = 32-bit counter. 3h = RESERVED
3-0	NCH	R	3h	Number of channels.

10.5.3 STARTCFG Register (Offset = 8h) [Reset = 00000000h]

STARTCFG is shown in Table 10-8.

Return to the Summary Table.

Start Configuration
This register is only for when CTL.MODE is configured to one of the SYNC modes.
This register defines when this LGPT starts.

Table 10-8. STARTCFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1-0	LGPT0	R/W	0h	LGPT start

10.5.4 CTL Register (Offset = Ch) [Reset = 00000000h]

CTL is shown in Table 10-9.

Return to the Summary Table.

Timer Control

Table 10-9. CTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2RST	W	0h	Channel 2 reset. 0h = No effect. 1h = Reset C2CC, PC2CC, and C2CFG.
9	C1RST	W	0h	Channel 1 reset. 0h = No effect. 1h = Reset C1CC, PC1CC, and C1CFG.
8	C0RST	W	0h	Channel 0 reset. 0h = No effect. 1h = Reset C0CC, PC0CC, and C0CFG.
7-6	RESERVED	R	0h	Reserved
5	INTP	R/W	0h	Interrupt Phase. This bit field controls when the RIS.TGT and RIS.ZERO interrupts are set. 0h = RIS.TGT and RIS.ZERO are set one system clock cycle after CNTR = TARGET/ZERO. 1h = RIS.TGT and RIS.ZERO are set one timer clock cycle after CNTR = TARGET/ZERO.
4-3	CMPDIR	R/W	0h	Compare direction. This bit field controls the direction the counter must have in order to set the RIS.CnCC channel interrupts. This bitfield is only relevant if [CnCFG.CCACT] is configured to a compare action. 0h = Compare RIS fields are set on up count and down count. 1h = Compare RIS fields are only set on up count. 2h = Compare RIS fields are only set on down count. 3h = RESERVED

Table 10-9. CTL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
2-0	MODE	R/W	0h	Timer mode control The CNTR restarts from 0 when MODE is written to UP_ONCE, UP_PER, UPDWN_PER, QDEC, SYNC_UP_ONCE, SYNC_UP_PER or SYNC_UPDWN_PER. When writing MODE all internally queued updates to the channels and TGT is cleared. When configuring the timer, MODE should be the last thing to configure. If changing timer configuration after MODE has been set is necessary, instructions, if any, given in the configuration registers should be followed. See for example C0CFG. 0h = Disable timer. Updates to counter, channels, and events stop. 1h = Count up once. The timer increments from 0 to target value, then stops and sets MODE to DIS. 2h = Count up periodically. The timer increments from 0 to target value, repeatedly. Period = (target value + 1) * timer clock period 3h = Count up and down periodically. The timer counts from 0 to target value and back to 0, repeatedly. Period = (target value * 2) * timer clock period 4h = The timer functions as a quadrature decoder. IOC input 0, IOC input 1 and IOC input 2 are used respectivly as PHA, PHB and IDX inputs. IDX can be turned off by setting C2CFG.EDGE = NONE. The timer clock frequency sets the sample rate of the QDEC logic. This frequency can be configured in PRECFG. 5h = Start counting up once synchronous to another LGPT, selected within STARTCFG. The timer is started by setting CTL.MODE = UP_ONCE automatically. It then functions as a normal timer in CTL.MODE = UP_ONCE, incrementing from 0 to target value, then stops and sets MODE to DIS. 6h = Start counting up periodically synchronous to another LGPT, selected within STARTCFG. The timer is started by setting CTL.MODE = UP_PER automatically. It then operates as a normal timer in CTL.MODE = UP_PER, incrementing from 0 to target value, repeatedly. Period = (target value * 2) * timer clock period 7h = Start counting up and down periodically synchronous to another LGPT, selected within STARTCFG. The timer is started by setting CTL.MODE = UPDWN_PER automatically. It then operates as a normal timer in CTL.MODE = UPDWN_PER, counting from 0 to target value and back to 0, repeatedly. Period = (target value * 2) * timer clock period

10.5.5 OUTCTL Register (Offset = 10h) [Reset = 00000000h]

OUTCTL is shown in Table 10-10.

Return to the Summary Table.

Output Control
Set and clear individual outputs manually. Manual update of an output takes priority over automatic channel updates to the same output. It is not possible to set and clear an output at the same time, such requests will be neglected.
An output can be automatically cleared, set, toggled, or pulsed by each channel, listed in decreasing order of priority. The action with highest priority happens when multiple channels want to update an output at the same time.
All outputs are connected to the event fabric and the IO controller. The outputs going to the IO controller have an aditional complementary output, this output is the inverted IO output. Both the IO and the IO complementary outputs are passed through an IO Controller, see IOCTL.

Table 10-10. OUTCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Reserved
5	SETOUT2	W	0h	Set output 2. Write 1 to set output 2.
4	CLROUT2	W	0h	Clear output 2. Write 1 to clear output 2.
3	SETOUT1	W	0h	Set output 1. Write 1 to set output 1.
2	CLROUT1	W	0h	Clear output 1. Write 1 to clear output 1.
1	SETOUT0	W	0h	Set output 0. Write 1 to set output 0.
0	CLROUT0	W	0h	Clear output 0. Write 1 to clear output 0.

10.5.6 CNTR Register (Offset = 14h) [Reset = 00000000h]

CNTR is shown in Table 10-11.

Return to the Summary Table.

Counter
The counter of this timer. After CTL.MODE is set the counter updates at the rate specified in PRECFG.

Table 10-11. CNTR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Current counter value. If CTL.MODE = QDEC this can be used to set the initial counter value during QDEC. Writing to CNTR in other modes than QDEC is possible, but may result in unpredictable behavior.

10.5.7 PRECFG Register (Offset = 18h) [Reset = 00000000h]

PRECFG is shown in Table 10-12.

Return to the Summary Table.

Clock Prescaler Configuration
This register is used to set the timer clock period. The prescaler is a counter which counts down from the value TICKDIV. When the prescaler counter reaches zero, CNTR is updated. The field TICKDIV effectively divides the prescaler tick source. The timer clock frequency can be calculated as TICKSRC/(TICKDIV+1).

Table 10-12. PRECFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-8	TICKDIV	R/W	0h	Tick division. TICKDIV determines the timer clock frequency for the counter, and timer output updates. The timer clock frequency is the clock selected by TICKSRC divided by (TICKDIV + 1). This inverse is the timer clock period. 0x00: Divide by 1. 0x01: Divide by 2. ... 0xFF: Divide by 256.
7-2	RESERVED	R	0h	Reserved
1-0	TICKSRC	R/W	0h	Prescaler tick source. TICKSRC determines the source which decrements the prescaler. 0h = Prescaler is updated at the system clock. 1h = Prescaler is updated at the rising edge of TICKEN. 2h = Prescaler is updated at the falling edge of TICKEN. 3h = Prescaler is updated at both edges of TICKEN.

10.5.8 PREEVENT Register (Offset = 1Ch) [Reset = 00000000h]

PREEVENT is shown in Table 10-13.

Return to the Summary Table.

Prescaler Event
This register is used to output a logic high signal before the zero crossing of the prescaler counter. The output is routed to the IOC.

Table 10-13. PREEVENT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	VAL	R/W	0h	Sets the HIGH time of the prescaler event output. Event goes high when the prescaler counter equals VAL. Event goes low when prescaler counter is 0. Note: - Can be used to precharge or turn an external component on for a short time before sampling, like in QDEC. - If there is a requirement to create such events that have very short periods compared to timer clock period, use two timers. One timer acts as prescaler and event generator for another timer.

10.5.9 CHFILT Register (Offset = 20h) [Reset = 00000000h]

CHFILT is shown in Table 10-14.

Return to the Summary Table.

Channel Input Filter
This register is used to configure the filter on the channel inputs. The configuration is for all inputs.
The filter is enabled when a channel is in capture mode.
The input to the filter is passed to the edge detection logic if LOAD + 1 consecutive input samples are equal. The filter functions as a down counter, counting down every input sample.
If two consecutive samples are unequal, the filter counter restarts from LOAD.
If the filter counter reaches zero, the input signal is valid and passed to the edge detection logic.
The channel filter should only be configured while the CTL.MODE = DIS. Configuring the filter while the timer is running can result in unexpected behavior.

Table 10-14. CHFILT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-8	LOAD	R/W	0h	The input of the channel filter is passed to the edge detection logic after LOAD + 1 consecutive equal samples.
7-2	RESERVED	R	0h	Reserved
1-0	MODE	R/W	0h	Channel filter mode 0h = Filter is bypassed. No Filter is used. 1h = Filter is clocked by system clock. 2h = Filter is clocked by PRECFG.TICKSRC. 3h = Filter is clocked by timer clock.

10.5.10 QDECSTAT Register (Offset = 34h) [Reset = 00000000h]

QDECSTAT is shown in Table 10-15.

Return to the Summary Table.

Quadrature Decoder Status
This register can be used during QDEC mode to check the status of the quadrature decoder.

Table 10-15. QDECSTAT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1	DBLTRANS	R	0h	Double transition 0h = Single or no transition on phase inputs. 1h = Double transition on phase inputs.
0	QDIR	R	0h	Direction of count during QDEC mode. 0h = Up (PHA leads PHB) 1h = Down (PHB leads PHA)

10.5.11 IRGEN Register (Offset = 38h) [Reset = 00000000h]

IRGEN is shown in Table 10-16.

Return to the Summary Table.

IR Generation
Use this register to generate IR codes. When CTL = 1, an AND gate is enabled between IO output 0 in LGPT0 and IC output 0 in LGPT1. The output of the gate overrides IO output 0 in LGPT0. See OUTCTL for explanation of outputs.
To generate IR codes let LGPT0 generate the carrier wave on output 0. Set this output as tick input of LGPT1, with PRECFG.TICKSRC = FALL_TICK.
Generate wanted IR codes by adjusting LGPT1 PTGT and PC0CC.

Table 10-16. IRGEN Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Reserved
0	CTL	R/W	0h	Control 0h = Disable. 1h = Enable.

10.5.12 DMA Register (Offset = 3Ch) [Reset = 00000000h]

DMA is shown in Table 10-17.

Return to the Summary Table.

Direct Memory Accsess
This register is used to enable DMA requests from the timer and set the register addresses which the DMA will access (read/write).
Choose DMA request source by setting the REQ field. The setting of the corresponding interrupt in the RIS registers also sets the DMA request.
Upon a DMA request defined by REQ an internal address pointer is set to RWADDR*4. Every access to DMARW will increment the internal pointer by 4 such that the next DMA access will be to the next register.
The internal pointer will stop after RWCNTR increments. Further access will be ignored.

Table 10-17. DMA Register Field Descriptions

Bit	Field	Type	Reset	Description
31-20	RESERVED	R	0h	Reserved
19-16	RWCNTR	R/W	0h	The read/write counter. RWCNTR+1 is the number of times the DMA can access (read/write) the DMARW register. For each DMA access to DMARW an internal counter is incremented, writing to the next address field. RWADDR + 4*RWCNTR is the final register address which can be accessed by the DMA.
15	RESERVED	R	0h	Reserved
14-8	RWADDR	R/W	0h	The base address which the DMA access when reading/writing DMARW. The base address is set by taking the 9 LSB of the physical address and divide by 4. For example, if you wanted the RWADDR to point to the PTGT register you should set RWADDR = 0x0FC/4.
7-4	RESERVED	R	0h	Reserved
3-0	REQ	R/W	0h	0h = Disabled 1h = Setting of RIS.TGT generates a DMA request. 2h = Setting of RIS.ZERO generates a DMA request. 3h = Setting of RIS.FAULT generates a DMA request. 4h = Setting of RIS.C0CC generates a DMA request. 5h = Setting of RIS.C1CC generates a DMA request. 6h = Setting of RIS.C2CC generates a DMA request. 7h = Setting of RIS.C3CC generates a DMA request. 8h = Setting of RIS.C4CC generates a DMA request. 9h = Setting of RIS.C5CC generates a DMA request. Ah = Setting of RIS.C6CC generates a DMA request. Bh = Setting of RIS.C7CC generates a DMA request. Ch = Setting of RIS.C8CC generates a DMA request. Dh = Setting of RIS.C9CC generates a DMA request. Eh = Setting of RIS.C10CC generates a DMA request. Fh = Setting of RIS.C11CC generates a DMA request.

10.5.13 DMARW Register (Offset = 40h) [Reset = 00000000h]

DMARW is shown in Table 10-18.

Return to the Summary Table.

Direct Memory Access
This register is used by the DMA to access (read/write) register inside this LGPT module.
Each access to this register will increment the internal DMA address counter. See DMA for description.

Table 10-18. DMARW Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	DMA read write value. The value that is read/written from/to the registers.

10.5.14 ADCTRG Register (Offset = 44h) [Reset = 00000000h]

ADCTRG is shown in Table 10-19.

Return to the Summary Table.

ADC Trigger
This register is used to enable ADC trigger from the timer.
Choose ADC trigger source by setting the SRC field. The setting of the corresponding interrupt in the RIS registers also sets the ADC trigger.

Table 10-19. ADCTRG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3-0	SRC	R/W	0h	0h = Disabled 1h = Setting of RIS.TGT generates an ADC trigger. 2h = Setting of RIS.ZERO generates an ADC trigger. 3h = Setting of RIS.FAULT generates an ADC trigger. 4h = Setting of RIS.C0CC generates an ADC trigger. 5h = Setting of RIS.C1CC generates an ADC trigger. 6h = Setting of RIS.C2CC generates an ADC trigger. 7h = Setting of RIS.C3CC generates an ADC trigger. 8h = Setting of RIS.C4CC generates an ADC trigger. 9h = Setting of RIS.C5CC generates an ADC trigger. Ah = Setting of RIS.C6CC generates an ADC trigger. Bh = Setting of RIS.C7CC generates an ADC trigger. Ch = Setting of RIS.C8CC generates an ADC trigger. Dh = Setting of RIS.C9CC generates an ADC trigger. Eh = Setting of RIS.C10CC generates an ADC trigger. Fh = Setting of RIS.C11CC generates an ADC trigger.

10.5.15 IOCTL Register (Offset = 48h) [Reset = 00000000h]

IOCTL is shown in Table 10-20.

Return to the Summary Table.

IO Controller
This register overrides the IO outputs.

Table 10-20. IOCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-12	RESERVED	R	0h	Reserved
11-10	COUT2	R/W	0h	IO complementary output 2 control This bit field controls IO complementary output 2. 0h = Normal output. The IO complementary output is not changed. 1h = Driven low. The IO complementary output is driven low. 2h = Driven high. The IO complementary output is driven high. 3h = Inverted value. The IO complementary output is inverted.
9-8	OUT2	R/W	0h	IO output 2 control This bit field controls IO output 2. 0h = Normal output. The IO output is not changed. 1h = Driven low. The IO output is driven low. 2h = Driven high. The IO output is driven high. 3h = Inverted value. The IO output is inverted.
7-6	COUT1	R/W	0h	IO complementary output 1 control This bit field controls IO complementary output 1. 0h = Normal output. The IO complementary output is not changed. 1h = Driven low. The IO complementary output is driven low. 2h = Driven high. The IO complementary output is driven high. 3h = Inverted value. The IO complementary output is inverted.
5-4	OUT1	R/W	0h	IO output 1 control This bit field controls IO output 1. 0h = Normal output. The IO output is not changed. 1h = Driven low. The IO output is driven low. 2h = Driven high. The IO output is driven high. 3h = Inverted value. The IO output is inverted.
3-2	COUT0	R/W	0h	IO complementary output 0 control This bit field controls IO complementary output 0. 0h = Normal output. The IO complementary output is not changed. 1h = Driven low. The IO complementary output is driven low. 2h = Driven high. The IO complementary output is driven high. 3h = Inverted value. The IO complementary output is inverted.
1-0	OUT0	R/W	0h	IO output 0 control This bit field controls IO output 0. 0h = Normal output. The IO output is not changed. 1h = Driven low. The IO output is driven low. 2h = Driven high. The IO output is driven high. 3h = Inverted value. The IO output is inverted.

10.5.16 IMASK Register (Offset = 68h) [Reset = 00000000h]

IMASK is shown in Table 10-21.

Return to the Summary Table.

Interrupt mask.
This register selects interrupt sources which are allowed to pass from RIS to MIS when the corresponding bit-fields are set to 1.

Table 10-21. IMASK Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	R/W	0h	Enable RIS.C2CC interrupt. 0h = Disable 1h = Enable
9	C1CC	R/W	0h	Enable RIS.C1CC interrupt. 0h = Disable 1h = Enable
8	C0CC	R/W	0h	Enable RIS.C0CC interrupt. 0h = Disable 1h = Enable
7	RESERVED	R	0h	Reserved
6	FAULT	R/W	0h	Enable RIS.FAULT interrupt. 0h = Disable 1h = Enable
5	IDX	R/W	0h	Enable RIS.IDX interrupt. 0h = Disable 1h = Enable
4	DIRCHNG	R/W	0h	Enable RIS.DIRCHNG interrupt. 0h = Disable 1h = Enable
3	CNTRCHNG	R/W	0h	Enable RIS.CNTRCHNG interrupt. 0h = Disable 1h = Enable
2	DBLTRANS	R/W	0h	Enable RIS.DBLTRANS interrupt. 0h = Disable 1h = Enable
1	ZERO	R/W	0h	Enable RIS.ZERO interrupt. 0h = Disable 1h = Enable
0	TGT	R/W	0h	Enable RIS.TGT interrupt. 0h = Disable 1h = Enable

10.5.17 RIS Register (Offset = 6Ch) [Reset = 00000000h]

RIS is shown in Table 10-22.

Return to the Summary Table.

Raw interrupt status.
This register reflects the state of all pending interrupts, regardless of masking. This register allows the user to implement a poll scheme. A flag set in this register can be cleared by writing 1 to the corresponding ICLR register bit.

Table 10-22. RIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	R	0h	Status of the C2CC interrupt. The interrupt is set when C2CC has capture or compare event. 0h = Cleared 1h = Set
9	C1CC	R	0h	Status of the C1CC interrupt. The interrupt is set when C1CC has capture or compare event. 0h = Cleared 1h = Set
8	C0CC	R	0h	Status of the C0CC interrupt. The interrupt is set when C0CC has capture or compare event. 0h = Cleared 1h = Set
7	RESERVED	R	0h	Reserved
6	FAULT	R	0h	Status of the FAULT interrupt. The interrupt is set immediately on active fault input. 0h = Cleared 1h = Set
5	IDX	R	0h	Status of the IDX interrupt. The interrupt is set when IDX is active. 0h = Cleared 1h = Set
4	DIRCHNG	R	0h	Status of the DIRCHNG interrupt. The interrupt is set when the direction of the counter changes. 0h = Cleared 1h = Set
3	CNTRCHNG	R	0h	Status of the CNTRCHNG interrupt. The interrupt is set when the counter increments or decrements. 0h = Cleared 1h = Set
2	DBLTRANS	R	0h	Status of the DBLTRANS interrupt. The interrupt is set when a double transition has happened during QDEC mode. 0h = Cleared 1h = Set
1	ZERO	R	0h	Status of the ZERO interrupt. The interrupt is set when CNTR = 0. 0h = Cleared 1h = Set
0	TGT	R	0h	Status of the TGT interrupt. The interrupt is set when CNTR = TGT. 0h = Cleared 1h = Set

10.5.18 MIS Register (Offset = 70h) [Reset = 00000000h]

MIS is shown in Table 10-23.

Return to the Summary Table.

Masked interrupt status.
This register is simply a bitwise AND of the contents of IMASK and RIS.*] registers. A flag set in this register can be cleared by writing 1 to the corresponding ICLR register bit.

Table 10-23. MIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	R	0h	Masked status of the RIS.C2CC interrupt. 0h = Cleared 1h = Set
9	C1CC	R	0h	Masked status of the RIS.C1CC interrupt. 0h = Cleared 1h = Set
8	C0CC	R	0h	Masked status of the RIS.C0CC interrupt. 0h = Cleared 1h = Set
7	RESERVED	R	0h	Reserved
6	FAULT	R	0h	Masked status of the RIS.FAULT interrupt. 0h = Cleared 1h = Set
5	IDX	R	0h	Masked status of the RIS.IDX interrupt. 0h = Cleared 1h = Set
4	DIRCHNG	R	0h	Masked status of the RIS.DIRCHNG interrupt. 0h = Cleared 1h = Set
3	CNTRCHNG	R	0h	Masked status of the RIS.CNTRCHNG interrupt. 0h = Cleared 1h = Set
2	DBLTRANS	R	0h	Masked status of the RIS.DBLTRANS interrupt. 0h = Cleared 1h = Set
1	ZERO	R	0h	Masked status of the RIS.ZERO interrupt. 0h = Cleared 1h = Set
0	TGT	R	0h	Masked status of the RIS.TGT interrupt. 0h = Cleared 1h = Set

10.5.19 ISET Register (Offset = 74h) [Reset = 00000000h]

ISET is shown in Table 10-24.

Return to the Summary Table.

Interrupt set register.
This register can used by software for diagnostics and safety checking purposes. Writing a 1 to a bit in this register will set the event and the corresponding RIS bit also gets set. If the corresponding IMASK bit is set, then the corresponding MIS register bit also gets set.

Table 10-24. ISET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	W	0h	Set the RIS.C2CC interrupt. 0h = No effect 1h = Set
9	C1CC	W	0h	Set the RIS.C1CC interrupt. 0h = No effect 1h = Set
8	C0CC	W	0h	Set the RIS.C0CC interrupt. 0h = No effect 1h = Set
7	RESERVED	R	0h	Reserved
6	FAULT	W	0h	Set the RIS.FAULT interrupt. 0h = No effect 1h = Set
5	IDX	W	0h	Set the RIS.IDX interrupt. 0h = No effect 1h = Set
4	DIRCHNG	W	0h	Set the RIS.DIRCHNG interrupt. 0h = No effect 1h = Set
3	CNTRCHNG	W	0h	Set the RIS.CNTRCHNG interrupt. 0h = No effect 1h = Set
2	DBLTRANS	W	0h	Set the RIS.DBLTRANS interrupt. 0h = No effect 1h = Set
1	ZERO	W	0h	Set the RIS.ZERO interrupt. 0h = No effect 1h = Set
0	TGT	W	0h	Set the RIS.TGT interrupt. 0h = No effect 1h = Set

10.5.20 ICLR Register (Offset = 78h) [Reset = 00000000h]

ICLR is shown in Table 10-25.

Return to the Summary Table.

Interrupt clear register.
This register allows software to clear interrupts. Writing a 1 to a bit in this register will clear the event and the corresponding RIS bit also gets cleared. If the corresponding IMASK bit is set, then the corresponding MIS register bit also gets cleared.

Table 10-25. ICLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	W	0h	Clear the RIS.C2CC interrupt. 0h = No effect 1h = Clear
9	C1CC	W	0h	Clear the RIS.C1CC interrupt. 0h = No effect 1h = Clear
8	C0CC	W	0h	Clear the RIS.C0CC interrupt. 0h = No effect 1h = Clear
7	RESERVED	R	0h	Reserved
6	FAULT	W	0h	Clear the RIS.FAULT interrupt. 0h = No effect 1h = Clear
5	IDX	W	0h	Clear the RIS.IDX interrupt. 0h = No effect 1h = Clear
4	DIRCHNG	W	0h	Clear the RIS.DIRCHNG interrupt. 0h = No effect 1h = Clear
3	CNTRCHNG	W	0h	Clear the RIS.CNTRCHNG interrupt. 0h = No effect 1h = Clear
2	DBLTRANS	W	0h	Clear the RIS.DBLTRANS interrupt. 0h = No effect 1h = Clear
1	ZERO	W	0h	Clear the RIS.ZERO interrupt. 0h = No effect 1h = Clear
0	TGT	W	0h	Clear the RIS.TGT interrupt. 0h = No effect 1h = Clear

10.5.21 IMSET Register (Offset = 7Ch) [Reset = 00000000h]

IMSET is shown in Table 10-26.

Return to the Summary Table.

Interrupt mask set register.
Writing a 1 to a bit in this register will set the corresponding IMASK bit.

Table 10-26. IMSET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	W	0h	Set the MIS.C2CC mask. 0h = No effect 1h = Set
9	C1CC	W	0h	Set the MIS.C1CC mask. 0h = No effect 1h = Set
8	C0CC	W	0h	Set the MIS.C0CC mask. 0h = No effect 1h = Set
7	RESERVED	R	0h	Reserved
6	FAULT	W	0h	Set the MIS.FAULT mask. 0h = No effect 1h = Set
5	IDX	W	0h	Set the MIS.IDX mask. 0h = No effect 1h = Set
4	DIRCHNG	W	0h	Set the MIS.DIRCHNG mask. 0h = No effect 1h = Set
3	CNTRCHNG	W	0h	Set the MIS.CNTRCHNG mask. 0h = No effect 1h = Set
2	DBLTRANS	W	0h	Set the MIS.DBLTRANS mask. 0h = No effect 1h = Set
1	ZERO	W	0h	Set the MIS.ZERO mask. 0h = No effect 1h = Set
0	TGT	W	0h	Set the MIS.TGT mask. 0h = No effect 1h = Set

10.5.22 IMCLR Register (Offset = 80h) [Reset = 00000000h]

IMCLR is shown in Table 10-27.

Return to the Summary Table.

Interrupt mask clear register.
Writing a 1 to a bit in this register will clear the corresponding IMASK bit.

Table 10-27. IMCLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	W	0h	Clear the MIS.C2CC mask. 0h = No effect 1h = Clear
9	C1CC	W	0h	Clear the MIS.C1CC mask. 0h = No effect 1h = Clear
8	C0CC	W	0h	Clear the MIS.C0CC mask. 0h = No effect 1h = Clear
7	RESERVED	R	0h	Reserved
6	FAULT	W	0h	Clear the MIS.FAULT mask. 0h = No effect 1h = Clear
5	IDX	W	0h	Clear the MIS.IDX mask. 0h = No effect 1h = Clear
4	DIRCHNG	W	0h	Clear the MIS.DIRCHNG mask. 0h = No effect 1h = Clear
3	CNTRCHNG	W	0h	Clear the MIS.CNTRCHNG mask. 0h = No effect 1h = Clear
2	DBLTRANS	W	0h	Clear the MIS.DBLTRANS mask. 0h = No effect 1h = Clear
1	ZERO	W	0h	Clear the MIS.ZERO mask. 0h = No effect 1h = Clear
0	TGT	W	0h	Clear the MIS.TGT mask. 0h = No effect 1h = Clear

10.5.23 EMU Register (Offset = 84h) [Reset = 00000000h]

EMU is shown in Table 10-28.

Return to the Summary Table.

Debug control
This register can be used to freeze the timer when CPU halts when HALT is set to 1. When HALT is set to 0, or when the CPU releases debug halt, the filters and edge detection logic is flushed and the timer starts. For setting a predefined output value during a CPU debug halt, PARK, if the timer has this register, should be configured additionally. If this timer does not have the PARK register a predefined output value during CPU halt is not possible.

Table 10-28. EMU Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1	CTL	R/W	0h	Halt control. Configure when the counter shall stop upon CPU halt. This bitfield only applies if HALT = 1. 0h = Immediate reaction. The counter stops immediately on debug halt. 1h = Zero condition. The counter stops when CNTR = 0.
0	HALT	R/W	0h	Halt LGPT when CPU is halted in debug. 0h = Disable. 1h = Enable.

10.5.24 C0CFG Register (Offset = C0h) [Reset = 00000000h]

C0CFG is shown in Table 10-29.

Return to the Summary Table.

Channel 0 Configuration
This register configures channel function and enables outputs.
Each channel has an edge-detection circuit. The the edge-detection circuit is:
- enabled while CCACT selects a capture function and CTL.MODE is different from DIS.
- flushed while CCACT selects a capture function and CTL.MODE is changed from DIS to another mode.
The flush action uses two system clock periods. It prevents capture events caused by expired signal values stored in the edge-detection circuit.
The channel input signal enters the edge-detection circuit. False capture events can occur when:
- the edge-detection circuit contains expired signal samples and the circuit is enabled without flush as described above.
- the CCACT field is reconfigured while CTL.MODE is different from DIS.
Primary use scenario is to select CCACT before starting the timer. Follow these steps to configure CCACT to a capture action while CTL.MODE is different from DIS:
- Set EDGE to NONE.
- Configure CCACT.
- Wait for three system clock periods before setting EDGE different from NONE.
These steps prevent capture events caused by expired signal values in edge-detection circuit.

Table 10-29. C0CFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	OUT2	R/W	0h	Output 2 enable. When 0 < CCACT < 8, OUT2 becomes zero after a capture or compare event. 0h = Channel 0 does not control output 2. 1h = Channel 0 controls output 2.
9	OUT1	R/W	0h	Output 1 enable. When 0 < CCACT < 8, OUT1 becomes zero after a capture or compare event. 0h = Channel 0 does not control output 1. 1h = Channel 0 controls output 1.
8	OUT0	R/W	0h	Output 0 enable. When 0 < CCACT < 8, OUT0 becomes zero after a capture or compare event. 0h = Channel 0 does not control output 0. 1h = Channel 0 controls output 0.
7	RESERVED	R	0h	Reserved
6	INPUT	R/W	0h	Select channel input. 0h = Event fabric 1h = IO controller
5-4	EDGE	R/W	0h	Determines the edge that triggers the channel input event. This happens post filter. 0h = Input is turned off. 1h = Input event is triggered at rising edge. 2h = Input event is triggered at falling edge. 3h = Input event is triggered at both edges.

Table 10-29. C0CFG Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
3-0	CCACT	R/W	0h	Capture-Compare action. Capture-Compare action defines 15 different channel functions that utilize capture, compare, and zero events. In every compare event the timer looks at the current value of CNTR. The corresponding output event will be set 1 timer period after CNTR = C0CC. 0h = Disable channel. 1h = Set on capture, and then disable channel. Channel function sequence: - Set enabled outputs on capture event and copy CNTR.VAL to C0CC.VAL. - Disable channel. Primary use scenario is to select this function before starting the timer. Follow these steps to select this function while CTL.MODE is different from DIS: - Set CCACT to SET_ON_CAPT with no output enable. - Configure INPUT (optional). - Wait for three timer clock periods as defined in PRECFG before setting CCACT to SET_ON_CAPT_DIS. Output enable is optional. These steps prevent capture events caused by expired signal values in edge-detection circuit. 2h = Clear on zero, toggle on compare, and then disable channel. Channel function sequence: - Clear enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C0CC.VAL = CNTR.VAL. - Disable channel. Enabled outputs are set when C0CC.VAL = 0 and CNTR.VAL = 0. 3h = Set on zero, toggle on compare, and then disable channel. Channel function sequence: - Set enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C0CC.VAL = CNTR.VAL. - Disable channel. Enabled outputs are cleared when C0CC.VAL = 0 and CNTR.VAL = 0. 4h = Clear on compare, and then disable channel. Channel function sequence: - Clear enabled outputs when C0CC.VAL = CNTR.VAL. - Disable channel. 5h = Set on compare, and then disable channel. Channel function sequence: - Set enabled outputs when C0CC.VAL = CNTR.VAL. - Disable channel. 6h = Toggle on compare, and then disable channel. Channel function sequence: - Toggle enabled outputs when C0CC.VAL = CNTR.VAL. - Disable channel. 7h = Pulse on compare, and then disable channel. Channel function sequence: - Pulse enabled outputs when C0CC.VAL = CNTR.VAL. - Disable channel. The output is high for two timer clock periods. 8h = Period and pulse width measurement. Continuously capture period and pulse width of the signal selected by INPUT relative to the signal edge given by EDGE. Set enabled outputs and RIS.C0CC when C0CC.VAL contains signal period and PC0CC.VAL contains signal pulse width. Notes: - Make sure to configure INPUT and CCACT when CTL.MODE equals DIS, then set CTL.MODE to UP_ONCE or UP_PER. - The counter restarts in the selected timer mode when C0CC.VAL contains the signal period. - If more than one channel uses this function, the channels will perform this function one at a time. The channel with lowest number has priority and performs the function first. Next measurement starts

Table 10-29. C0CFG Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
				when current measurement completes successfully or times out. A timeout occurs when counter equals target. - To observe a timeout event the RIS.TGT interrupt can be used, or another channel can be configured to SET_ON_CMP with compare value equal TGT. Signal property requirements: - Signal Period >= 2 * (1 + PRECFG.TICKDIV) * timer clock period. - Signal Period <= MAX(CNTR) * (1 + PRECFG.TICKDIV) * timer clock period. - Signal low and high phase >= (1 + PRECFG.TICKDIV) * timer clock period. 9h = Set on capture repeatedly. Channel function sequence: - Set enabled outputs on capture event and copy CNTR.VAL to C0CC.VAL. Ah = Clear on zero, toggle on compare repeatedly. Channel function sequence: - Clear enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C0CC.VAL = CNTR.VAL. Set CTL.MODE to UPDWN_PER for center-aligned PWM generation. Duty cycle is given by: When C0CC.VAL <= TGT.VAL: Duty cycle = 1 - (C0CC.VAL / TGT.VAL). When C0CC.VAL > TGT.VAL: Duty cycle = 0. Enabled outputs are set when C0CC.VAL = 0 and CNTR.VAL = 0. Bh = Set on zero, toggle on compare repeatedly. Channel function sequence: - Set enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C0CC.VAL = CNTR.VAL. Set CTL.MODE to UP_PER for edge-aligned PWM generation. Duty cycle is given by: When C0CC.VAL <= TGT.VAL: Duty cycle = C0CC.VAL / (TGT.VAL + 1). When C0CC.VAL > TGT.VAL: Duty cycle = 1. Enabled outputs are cleared when C0CC.VAL = 0 and CNTR.VAL = 0. Ch = Clear on compare repeatedly. Channel function sequence: - Clear enabled outputs when C0CC.VAL = CNTR.VAL. Dh = Set on compare repeatedly. Channel function sequence: - Set enabled outputs when C0CC.VAL = CNTR.VAL. Eh = Toggle on compare repeatedly. Channel function sequence: - Toggle enabled outputs when C0CC.VAL = CNTR.VAL. Fh = Pulse on compare repeatedly. Channel function sequence: - Pulse enabled outputs when C0CC.VAL = CNTR.VAL. The output is high for two timer clock periods.

10.5.25 C1CFG Register (Offset = C4h) [Reset = 00000000h]

C1CFG is shown in Table 10-30.

Return to the Summary Table.

Channel 1 Configuration
This register configures channel function and enables outputs.
Each channel has an edge-detection circuit. The the edge-detection circuit is:
- enabled while CCACT selects a capture function and CTL.MODE is different from DIS.
- flushed while CCACT selects a capture function and CTL.MODE is changed from DIS to another mode.
The flush action uses two system clock periods. It prevents capture events caused by expired signal values stored in the edge-detection circuit.
The channel input signal enters the edge-detection circuit. False capture events can occur when:
- the edge-detection circuit contains expired signal samples and the circuit is enabled without flush as described above.
- the CCACT field is reconfigured while CTL.MODE is different from DIS.
Primary use scenario is to select CCACT before starting the timer. Follow these steps to configure CCACT to a capture action while CTL.MODE is different from DIS:
- Set EDGE to NONE.
- Configure CCACT.
- Wait for three system clock periods before setting EDGE different from NONE.
These steps prevent capture events caused by expired signal values in edge-detection circuit.

Table 10-30. C1CFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	OUT2	R/W	0h	Output 2 enable. When 0 < CCACT < 8, OUT2 becomes zero after a capture or compare event. 0h = Channel 1 does not control output 2. 1h = Channel 1 controls output 2.
9	OUT1	R/W	0h	Output 1 enable. When 0 < CCACT < 8, OUT1 becomes zero after a capture or compare event. 0h = Channel 1 does not control output 1. 1h = Channel 1 controls output 1.
8	OUT0	R/W	0h	Output 0 enable. When 0 < CCACT < 8, OUT0 becomes zero after a capture or compare event. 0h = Channel 1 does not control output 0. 1h = Channel 1 controls output 0.
7	RESERVED	R	0h	Reserved
6	INPUT	R/W	0h	Select channel input. 0h = Event fabric 1h = IO controller
5-4	EDGE	R/W	0h	Determines the edge that triggers the channel input event. This happens post filter. 0h = Input is turned off. 1h = Input event is triggered at rising edge. 2h = Input event is triggered at falling edge. 3h = Input event is triggered at both edges.

Table 10-30. C1CFG Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
3-0	CCACT	R/W	0h	Capture-Compare action. Capture-Compare action defines 15 different channel functions that utilize capture, compare, and zero events. In every compare event the timer looks at the current value of CNTR. The corresponding output event will be set 1 timer period after CNTR = C1CC. 0h = Disable channel. 1h = Set on capture, and then disable channel. Channel function sequence: - Set enabled outputs on capture event and copy CNTR.VAL to C1CC.VAL. - Disable channel. Primary use scenario is to select this function before starting the timer. Follow these steps to select this function while CTL.MODE is different from DIS: - Set CCACT to SET_ON_CAPT with no output enable. - Configure INPUT (optional). - Wait for three timer clock periods as defined in PRECFG before setting CCACT to SET_ON_CAPT_DIS. Output enable is optional. These steps prevent capture events caused by expired signal values in edge-detection circuit. 2h = Clear on zero, toggle on compare, and then disable channel. Channel function sequence: - Clear enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C1CC.VAL = CNTR.VAL. - Disable channel. Enabled outputs are set when C1CC.VAL = 0 and CNTR.VAL = 0. 3h = Set on zero, toggle on compare, and then disable channel. Channel function sequence: - Set enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C1CC.VAL = CNTR.VAL. - Disable channel. Enabled outputs are cleared when C1CC.VAL = 0 and CNTR.VAL = 0. 4h = Clear on compare, and then disable channel. Channel function sequence: - Clear enabled outputs when C1CC.VAL = CNTR.VAL. - Disable channel. 5h = Set on compare, and then disable channel. Channel function sequence: - Set enabled outputs when C1CC.VAL = CNTR.VAL. - Disable channel. 6h = Toggle on compare, and then disable channel. Channel function sequence: - Toggle enabled outputs when C1CC.VAL = CNTR.VAL. - Disable channel. 7h = Pulse on compare, and then disable channel. Channel function sequence: - Pulse enabled outputs when C1CC.VAL = CNTR.VAL. - Disable channel. The output is high for two timer clock periods. 8h = Period and pulse width measurement. Continuously capture period and pulse width of the signal selected by INPUT relative to the signal edge given by EDGE. Set enabled outputs and RIS.C1CC when C1CC.VAL contains signal period and PC1CC.VAL contains signal pulse width. Notes: - Make sure to configure INPUT and CCACT when CTL.MODE equals DIS, then set CTL.MODE to UP_ONCE or UP_PER. - The counter restarts in the selected timer mode when C1CC.VAL contains the signal period. - If more than one channel uses this function, the channels will perform this function one at a time. The channel with lowest number has priority and performs the function first. Next measurement starts

Table 10-30. C1CFG Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
				when current measurement completes successfully or times out. A timeout occurs when counter equals target. - To observe a timeout event the RIS.TGT interrupt can be used, or another channel can be configured to SET_ON_CMP with compare value equal TGT. Signal property requirements: - Signal Period >= 2 * (1 + PRECFG.TICKDIV) * timer clock period. - Signal Period <= MAX(CNTR) * (1 + PRECFG.TICKDIV) * timer clock period. - Signal low and high phase >= (1 + PRECFG.TICKDIV) * timer clock period. 9h = Set on capture repeatedly. Channel function sequence: - Set enabled outputs on capture event and copy CNTR.VAL to C1CC.VAL. Ah = Clear on zero, toggle on compare repeatedly. Channel function sequence: - Clear enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C1CC.VAL = CNTR.VAL. Set CTL.MODE to UPDWN_PER for center-aligned PWM generation. Duty cycle is given by: When C1CC.VAL <= TGT.VAL: Duty cycle = 1 - (C1CC.VAL / TGT.VAL). When C1CC.VAL > TGT.VAL: Duty cycle = 0. Enabled outputs are set when C1CC.VAL = 0 and CNTR.VAL = 0. Bh = Set on zero, toggle on compare repeatedly. Channel function sequence: - Set enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C1CC.VAL = CNTR.VAL. Set CTL.MODE to UP_PER for edge-aligned PWM generation. Duty cycle is given by: When C1CC.VAL <= TGT.VAL: Duty cycle = C1CC.VAL / (TGT.VAL + 1). When C1CC.VAL > TGT.VAL: Duty cycle = 1. Enabled outputs are cleared when C1CC.VAL = 0 and CNTR.VAL = 0. Ch = Clear on compare repeatedly. Channel function sequence: - Clear enabled outputs when C1CC.VAL = CNTR.VAL. Dh = Set on compare repeatedly. Channel function sequence: - Set enabled outputs when C1CC.VAL = CNTR.VAL. Eh = Toggle on compare repeatedly. Channel function sequence: - Toggle enabled outputs when C1CC.VAL = CNTR.VAL. Fh = Pulse on compare repeatedly. Channel function sequence: - Pulse enabled outputs when C1CC.VAL = CNTR.VAL. The output is high for two timer clock periods.

10.5.26 C2CFG Register (Offset = C8h) [Reset = 00000000h]

C2CFG is shown in Table 10-31.

Return to the Summary Table.

Channel 2 Configuration
This register configures channel function and enables outputs.
Each channel has an edge-detection circuit. The the edge-detection circuit is:
- enabled while CCACT selects a capture function and CTL.MODE is different from DIS.
- flushed while CCACT selects a capture function and CTL.MODE is changed from DIS to another mode.
The flush action uses two system clock periods. It prevents capture events caused by expired signal values stored in the edge-detection circuit.
The channel input signal enters the edge-detection circuit. False capture events can occur when:
- the edge-detection circuit contains expired signal samples and the circuit is enabled without flush as described above.
- the CCACT field is reconfigured while CTL.MODE is different from DIS.
Primary use scenario is to select CCACT before starting the timer. Follow these steps to configure CCACT to a capture action while CTL.MODE is different from DIS:
- Set EDGE to NONE.
- Configure CCACT.
- Wait for three system clock periods before setting EDGE different from NONE.
These steps prevent capture events caused by expired signal values in edge-detection circuit.

Table 10-31. C2CFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	OUT2	R/W	0h	Output 2 enable. When 0 < CCACT < 8, OUT2 becomes zero after a capture or compare event. 0h = Channel 2 does not control output 2. 1h = Channel 2 controls output 2.
9	OUT1	R/W	0h	Output 1 enable. When 0 < CCACT < 8, OUT1 becomes zero after a capture or compare event. 0h = Channel 2 does not control output 1. 1h = Channel 2 controls output 1.
8	OUT0	R/W	0h	Output 0 enable. When 0 < CCACT < 8, OUT0 becomes zero after a capture or compare event. 0h = Channel 2 does not control output 0. 1h = Channel 2 controls output 0.
7	RESERVED	R	0h	Reserved
6	INPUT	R/W	0h	Select channel input. 0h = Event fabric 1h = IO controller
5-4	EDGE	R/W	0h	Determines the edge that triggers the channel input event. This happens post filter. 0h = Input is turned off. 1h = Input event is triggered at rising edge. 2h = Input event is triggered at falling edge. 3h = Input event is triggered at both edges.

Table 10-31. C2CFG Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
3-0	CCACT	R/W	0h	Capture-Compare action. Capture-Compare action defines 15 different channel functions that utilize capture, compare, and zero events. In every compare event the timer looks at the current value of CNTR. The corresponding output event will be set 1 timer period after CNTR = C2CC. 0h = Disable channel. 1h = Set on capture, and then disable channel. Channel function sequence: - Set enabled outputs on capture event and copy CNTR.VAL to C2CC.VAL. - Disable channel. Primary use scenario is to select this function before starting the timer. Follow these steps to select this function while CTL.MODE is different from DIS: - Set CCACT to SET_ON_CAPT with no output enable. - Configure INPUT (optional). - Wait for three timer clock periods as defined in PRECFG before setting CCACT to SET_ON_CAPT_DIS. Output enable is optional. These steps prevent capture events caused by expired signal values in edge-detection circuit. 2h = Clear on zero, toggle on compare, and then disable channel. Channel function sequence: - Clear enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C2CC.VAL = CNTR.VAL. - Disable channel. Enabled outputs are set when C2CC.VAL = 0 and CNTR.VAL = 0. 3h = Set on zero, toggle on compare, and then disable channel. Channel function sequence: - Set enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C2CC.VAL = CNTR.VAL. - Disable channel. Enabled outputs are cleared when C2CC.VAL = 0 and CNTR.VAL = 0. 4h = Clear on compare, and then disable channel. Channel function sequence: - Clear enabled outputs when C2CC.VAL = CNTR.VAL. - Disable channel. 5h = Set on compare, and then disable channel. Channel function sequence: - Set enabled outputs when C2CC.VAL = CNTR.VAL. - Disable channel. 6h = Toggle on compare, and then disable channel. Channel function sequence: - Toggle enabled outputs when C2CC.VAL = CNTR.VAL. - Disable channel. 7h = Pulse on compare, and then disable channel. Channel function sequence: - Pulse enabled outputs when C2CC.VAL = CNTR.VAL. - Disable channel. The output is high for two timer clock periods. 8h = Period and pulse width measurement. Continuously capture period and pulse width of the signal selected by INPUT relative to the signal edge given by EDGE. Set enabled outputs and RIS.C2CC when C2CC.VAL contains signal period and PC2CC.VAL contains signal pulse width. Notes: - Make sure to configure INPUT and CCACT when CTL.MODE equals DIS, then set CTL.MODE to UP_ONCE or UP_PER. - The counter restarts in the selected timer mode when C2CC.VAL contains the signal period. - If more than one channel uses this function, the channels will perform this function one at a time. The channel with lowest number has priority and performs the function first. Next measurement starts

Table 10-31. C2CFG Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
				when current measurement completes successfully or times out. A timeout occurs when counter equals target. - To observe a timeout event the RIS.TGT interrupt can be used, or another channel can be configured to SET_ON_CMP with compare value equal TGT. Signal property requirements: - Signal Period >= 2 * (1 + PRECFG.TICKDIV) * timer clock period. - Signal Period <= MAX(CNTR) * (1 + PRECFG.TICKDIV) * timer clock period. - Signal low and high phase >= (1 + PRECFG.TICKDIV) * timer clock period. 9h = Set on capture repeatedly. Channel function sequence: - Set enabled outputs on capture event and copy CNTR.VAL to C2CC.VAL. Ah = Clear on zero, toggle on compare repeatedly. Channel function sequence: - Clear enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C2CC.VAL = CNTR.VAL. Set CTL.MODE to UPDWN_PER for center-aligned PWM generation. Duty cycle is given by: When C2CC.VAL <= TGT.VAL: Duty cycle = 1 - (C2CC.VAL / TGT.VAL). When C2CC.VAL > TGT.VAL: Duty cycle = 0. Enabled outputs are set when C2CC.VAL = 0 and CNTR.VAL = 0. Bh = Set on zero, toggle on compare repeatedly. Channel function sequence: - Set enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C2CC.VAL = CNTR.VAL. Set CTL.MODE to UP_PER for edge-aligned PWM generation. Duty cycle is given by: When C2CC.VAL <= TGT.VAL: Duty cycle = C2CC.VAL / (TGT.VAL + 1). When C2CC.VAL > TGT.VAL: Duty cycle = 1. Enabled outputs are cleared when C2CC.VAL = 0 and CNTR.VAL = 0. Ch = Clear on compare repeatedly. Channel function sequence: - Clear enabled outputs when C2CC.VAL = CNTR.VAL. Dh = Set on compare repeatedly. Channel function sequence: - Set enabled outputs when C2CC.VAL = CNTR.VAL. Eh = Toggle on compare repeatedly. Channel function sequence: - Toggle enabled outputs when C2CC.VAL = CNTR.VAL. Fh = Pulse on compare repeatedly. Channel function sequence: - Pulse enabled outputs when C2CC.VAL = CNTR.VAL. The output is high for two timer clock periods.

10.5.27 PTGT Register (Offset = FCh) [Reset = 00000000h]

PTGT is shown in Table 10-32.

Return to the Summary Table.

Pipeline Target
A read or write to this register will clear the RIS.ZERO and RIS.TGT interrupt.
If CTL.MODE != QDEC.
Target value for next counter period.
The timer will copy PTGT.VAL to TGT.VAL on the upcoming CNTR zero crossing only if PTGT.VAL has been written. The copy does not happen when restarting the timer.
This is useful to avoid period jitter in PWM applications with time-varying period, sometimes referenced as phase corrected PWM.
If CTL.MODE = QDEC
The CNTR value is updated with VALUE on IDX if the counter is counting down. If the counter is counting up, CNTR is loaded with zero on IDX.
In this mode the VALUE is not loaded into TGT on zero crossing.

Table 10-32. PTGT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	The pipleline target value.

10.5.28 PC0CC Register (Offset = 100h) [Reset = 00000000h]

PC0CC is shown in Table 10-33.

Return to the Summary Table.

Pipeline Channel 0 Capture Compare

Table 10-33. PC0CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Pipeline Capture Compare value. User defined pipeline compare value or channel-updated capture value. A read or write to this register will clear the RIS.C0CC interrupt. Compare mode: An update of VAL will be transferred to C0CC.VAL when the next CNTR.VAL is zero and CTL.MODE is different from DIS. This is useful for PWM generation and prevents jitter on the edges of the generated signal. Capture mode: When C0CFG.CCACT equals PER_PULSE_WIDTH_MEAS then VAL contains the width of the low or high phase of the selected signal. This is specified by C0CFG.EDGE.

10.5.29 PC1CC Register (Offset = 104h) [Reset = 00000000h]

PC1CC is shown in Table 10-34.

Return to the Summary Table.

Pipeline Channel 1 Capture Compare

Table 10-34. PC1CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Pipeline Capture Compare value. User defined pipeline compare value or channel-updated capture value. A read or write to this register will clear the RIS.C1CC interrupt. Compare mode: An update of VAL will be transferred to C1CC.VAL when the next CNTR.VAL is zero and CTL.MODE is different from DIS. This is useful for PWM generation and prevents jitter on the edges of the generated signal. Capture mode: When C1CFG.CCACT equals PER_PULSE_WIDTH_MEAS then VAL contains the width of the low or high phase of the selected signal. This is specified by C1CFG.EDGE.

10.5.30 PC2CC Register (Offset = 108h) [Reset = 00000000h]

PC2CC is shown in Table 10-35.

Return to the Summary Table.

Pipeline Channel 2 Capture Compare

Table 10-35. PC2CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Pipeline Capture Compare value. User defined pipeline compare value or channel-updated capture value. A read or write to this register will clear the RIS.C2CC interrupt. Compare mode: An update of VAL will be transferred to C2CC.VAL when the next CNTR.VAL is zero and CTL.MODE is different from DIS. This is useful for PWM generation and prevents jitter on the edges of the generated signal. Capture mode: When C2CFG.CCACT equals PER_PULSE_WIDTH_MEAS then VAL contains the width of the low or high phase of the selected signal. This is specified by C2CFG.EDGE.

10.5.31 TGT Register (Offset = 13Ch) [Reset = 0000FFFFh]

TGT is shown in Table 10-36.

Return to the Summary Table.

Target
User defined counter target.
A read or write to this register will clear the RIS.ZERO and RIS.TGT interrupt.

Table 10-36. TGT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	FFFFh	User defined counter target value.

10.5.32 C0CC Register (Offset = 140h) [Reset = 00000000h]

C0CC is shown in Table 10-37.

Return to the Summary Table.

Channel 0 Capture Compare

Table 10-37. C0CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Capture Compare value. User defined compare value or channel-updated capture value. A read or write to this register will clear the RIS.C0CC interrupt. Compare mode: VAL is compared against CNTR.VAL and an event is generated as specified by C0CFG.CCACT when these are equal. Capture mode: The current counter value is stored in VAL when a capture event occurs. C0CFG.CCACT determines if VAL is a signal period or a regular capture value.

10.5.33 C1CC Register (Offset = 144h) [Reset = 00000000h]

C1CC is shown in Table 10-38.

Return to the Summary Table.

Channel 1 Capture Compare

Table 10-38. C1CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Capture Compare value. User defined compare value or channel-updated capture value. A read or write to this register will clear the RIS.C1CC interrupt. Compare mode: VAL is compared against CNTR.VAL and an event is generated as specified by C1CFG.CCACT when these are equal. Capture mode: The current counter value is stored in VAL when a capture event occurs. C1CFG.CCACT determines if VAL is a signal period or a regular capture value.

10.5.34 C2CC Register (Offset = 148h) [Reset = 00000000h]

C2CC is shown in Table 10-39.

Return to the Summary Table.

Channel 2 Capture Compare

Table 10-39. C2CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Capture Compare value. User defined compare value or channel-updated capture value. A read or write to this register will clear the RIS.C2CC interrupt. Compare mode: VAL is compared against CNTR.VAL and an event is generated as specified by C2CFG.CCACT when these are equal. Capture mode: The current counter value is stored in VAL when a capture event occurs. C2CFG.CCACT determines if VAL is a signal period or a regular capture value.

10.5.35 PTGTNC Register (Offset = 17Ch) [Reset = 00000000h]

PTGTNC is shown in Table 10-40.

Return to the Summary Table.

Pipeline Target No Clear
Use this register to read or write to PTGT without clearing the RIS.ZERO and RIS.TGT interrupt.

Table 10-40. PTGTNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	A read or write to this register will not clear the RIS.TGT interrupt. If CTL.MODE != QDEC. Target value for next counter period. The timer copies VAL to TGT.VAL when CNTR.VAL becomes 0. The copy does not happen when restarting the timer. This is useful to avoid period jitter in PWM applications with time-varying period, sometimes referenced as phase corrected PWM. If CTL.MODE = QDEC. The CNTR.VAL is updated with VAL on IDX. VAL is not loaded into TGT.VAL when CNTR.VAL becomes 0.

10.5.36 PC0CCNC Register (Offset = 180h) [Reset = 00000000h]

PC0CCNC is shown in Table 10-41.

Return to the Summary Table.

Pipeline Channel 0 Capture Compare No Clear

Table 10-41. PC0CCNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Pipeline Capture Compare value. User defined pipeline compare value or channel-updated capture value. A read or write to this register will not clear the RIS.C0CC interrupt. Compare mode: An update of VAL will be transferred to C0CC.VAL when the next CNTR.VAL is zero and CTL.MODE is different from DIS. This is useful for PWM generation and prevents jitter on the edges of the generated signal. Capture mode: When C0CFG.CCACT equals PER_PULSE_WIDTH_MEAS then VAL contains the width of the low or high phase of the selected signal. This is specified by C0CFG.EDGE.

10.5.37 PC1CCNC Register (Offset = 184h) [Reset = 00000000h]

PC1CCNC is shown in Table 10-42.

Return to the Summary Table.

Pipeline Channel 1 Capture Compare No Clear

Table 10-42. PC1CCNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Pipeline Capture Compare value. User defined pipeline compare value or channel-updated capture value. A read or write to this register will not clear the RIS.C1CC interrupt. Compare mode: An update of VAL will be transferred to C1CC.VAL when the next CNTR.VAL is zero and CTL.MODE is different from DIS. This is useful for PWM generation and prevents jitter on the edges of the generated signal. Capture mode: When C1CFG.CCACT equals PER_PULSE_WIDTH_MEAS then VAL contains the width of the low or high phase of the selected signal. This is specified by C1CFG.EDGE.

10.5.38 PC2CCNC Register (Offset = 188h) [Reset = 00000000h]

PC2CCNC is shown in Table 10-43.

Return to the Summary Table.

Pipeline Channel 2 Capture Compare No Clear

Table 10-43. PC2CCNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Pipeline Capture Compare value. User defined pipeline compare value or channel-updated capture value. A read or write to this register will not clear the RIS.C2CC interrupt. Compare mode: An update of VAL will be transferred to C2CC.VAL when the next CNTR.VAL is zero and CTL.MODE is different from DIS. This is useful for PWM generation and prevents jitter on the edges of the generated signal. Capture mode: When C2CFG.CCACT equals PER_PULSE_WIDTH_MEAS then VAL contains the width of the low or high phase of the selected signal. This is specified by C2CFG.EDGE.

10.5.39 TGTNC Register (Offset = 1BCh) [Reset = 0000FFFFh]

TGTNC is shown in Table 10-44.

Return to the Summary Table.

Target No Clear
Use this register to read or write to TGT without clearing the RIS.ZERO and RIS.TGT interrupt.

Table 10-44. TGTNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	FFFFh	User defined counter target value.

10.5.40 C0CCNC Register (Offset = 1C0h) [Reset = 00000000h]

C0CCNC is shown in Table 10-45.

Return to the Summary Table.

Channel 0 Capture Compare No Clear

Table 10-45. C0CCNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Capture Compare value. User defined compare value or channel-updated capture value. A read or write to this register will not clear the RIS.C0CC interrupt. Compare mode: VAL is compared against CNTR.VAL and an event is generated as specified by C0CFG.CCACT when these are equal. Capture mode: The current counter value is stored in VAL when a capture event occurs. C0CFG.CCACT determines if VAL is a signal period or a regular capture value.

10.5.41 C1CCNC Register (Offset = 1C4h) [Reset = 00000000h]

C1CCNC is shown in Table 10-46.

Return to the Summary Table.

Channel 1 Capture Compare No Clear

Table 10-46. C1CCNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Capture Compare value. User defined compare value or channel-updated capture value. A read or write to this register will not clear the RIS.C1CC interrupt. Compare mode: VAL is compared against CNTR.VAL and an event is generated as specified by C1CFG.CCACT when these are equal. Capture mode: The current counter value is stored in VAL when a capture event occurs. C1CFG.CCACT determines if VAL is a signal period or a regular capture value.

10.5.42 C2CCNC Register (Offset = 1C8h) [Reset = 00000000h]

C2CCNC is shown in Table 10-47.

Return to the Summary Table.

Channel 2 Capture Compare No Clear

Table 10-47. C2CCNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Capture Compare value. User defined compare value or channel-updated capture value. A read or write to this register will not clear the RIS.C2CC interrupt. Compare mode: VAL is compared against CNTR.VAL and an event is generated as specified by C2CFG.CCACT when these are equal. Capture mode: The current counter value is stored in VAL when a capture event occurs. C2CFG.CCACT determines if VAL is a signal period or a regular capture value.

10.6 LGPT1 Registers

Table 10-48 lists the memory-mapped registers for the LGPT1 registers. All register offset addresses not listed in Table 10-48 should be considered as reserved locations and the register contents should not be modified.

Table 10-48. LGPT1 Registers

Offset	Acronym	Register Name	Section
0h	DESC	Description Register.	Go
4h	DESCEX	Description Extended	Go
8h	STARTCFG	Start Configuration	Go
Ch	CTL	Timer Control	Go
10h	OUTCTL	Output Control	Go
14h	CNTR	Counter	Go
18h	PRECFG	Clock Prescaler Configuration	Go
1Ch	PREEVENT	Prescaler Event	Go
20h	CHFILT	Channel Input Filter	Go
24h	FAULT	Fault	Go
28h	PARK	Park	Go
2Ch	DBDLY	Deadband Delay	Go
30h	DBCTL	Deadband Control	Go
3Ch	DMA	Direct Memory Accsess	Go
40h	DMARW	Direct Memory Access	Go
44h	ADCTRG	ADC Trigger	Go
48h	IOCTL	IO Controller	Go
68h	IMASK	Interrupt mask.	Go
6Ch	RIS	Raw interrupt status.	Go
70h	MIS	Masked interrupt status.	Go
74h	ISET	Interrupt set register.	Go
78h	ICLR	Interrupt clear register.	Go
7Ch	IMSET	Interrupt mask set register.	Go
80h	IMCLR	Interrupt mask clear register.	Go
84h	EMU	Debug control	Go
C0h	C0CFG	Channel 0 Configuration	Go
C4h	C1CFG	Channel 1 Configuration	Go
C8h	C2CFG	Channel 2 Configuration	Go
FCh	PTGT	Pipeline Target	Go
100h	PC0CC	Pipeline Channel 0 Capture Compare	Go
104h	PC1CC	Pipeline Channel 1 Capture Compare	Go
108h	PC2CC	Pipeline Channel 2 Capture Compare	Go
13Ch	TGT	Target	Go
140h	C0CC	Channel 0 Capture Compare	Go
144h	C1CC	Channel 1 Capture Compare	Go
148h	C2CC	Channel 2 Capture Compare	Go
17Ch	PTGTNC	Pipeline Target No Clear	Go
180h	PC0CCNC	Pipeline Channel 0 Capture Compare No Clear	Go
184h	PC1CCNC	Pipeline Channel 1 Capture Compare No Clear	Go
188h	PC2CCNC	Pipeline Channel 2 Capture Compare No Clear	Go
1BCh	TGTNC	Target No Clear	Go

Table 10-48. LGPT1 Registers (continued)

Offset	Acronym	Register Name	Section
1C0h	C0CCNC	Channel 0 Capture Compare No Clear	Go
1C4h	C1CCNC	Channel 1 Capture Compare No Clear	Go
1C8h	C2CCNC	Channel 2 Capture Compare No Clear	Go

Complex bit access types are encoded to fit into small table cells. Table 10-49 shows the codes that are used for access types in this section.

Table 10-49. LGPT1 Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

10.6.1 DESC Register (Offset = 0h) [Reset = DE491010h]

DESC is shown in Table 10-50.

Return to the Summary Table.

Description Register.
This register provides IP module ID, revision information, instance index and standard MMR registers offset.

Table 10-50. DESC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	MODID	R	DE49h	Module identifier used to uniquely identify this IP.
15-12	STDIPOFF	R	1h	Standard IP MMR block offset. Standard IP MMRs are the set of from aggregated IRQ registers till DTB. 0: Standard IP MMRs do not exist 0x1-0xF: Standard IP MMRs begin at offset of (64*STDIPOFF from the base IP address)
11-8	INSTIDX	R	0h	IP Instance ID number. If multiple instances of IP exist in the device, this field can identify the instance number.
7-4	MAJREV	R	1h	Major revision of IP.
3-0	MINREV	R	0h	Minor revision of IP.

10.6.2 DESCEX Register (Offset = 4h) [Reset = 000618C3h]

DESCEX is shown in Table 10-51.

Return to the Summary Table.

Description Extended
This register describes the parameters of the LGPT.

Table 10-51. DESCEX Register Field Descriptions

Bit	Field	Type	Reset	Description
31-20	RESERVED	R	0h	Reserved
19	HIR	R	0h	Has IR logic.
18	HDBF	R	1h	Has deadband, Fault, and Park logic.
17-14	PREW	R	8h	Prescale width. The prescaler can maximum be configured to 2^{PREW}-1.
13	HQDEC	R	0h	Has Quadrature Decoder.
12	HCIF	R	1h	Has channel input filter.
11-8	CIFS	R	8h	Channel input filter size. The prevailing state filter can maximum be configured to 2^{CIFS}-1.
7	HDMA	R	1h	Has uDMA output and logic.
6	HINT	R	1h	Has interrupt output and logic.
5-4	CNTRW	R	0h	Counter bit-width. The maximum counter value is equal to 2^{CNTRW}-1. 0h = 16-bit counter. 1h = 24-bit counter. 2h = 32-bit counter. 3h = RESERVED
3-0	NCH	R	3h	Number of channels.

10.6.3 STARTCFG Register (Offset = 8h) [Reset = 00000000h]

STARTCFG is shown in Table 10-52.

Return to the Summary Table.

Start Configuration
This register is only for when CTL.MODE is configured to one of the SYNC modes.
This register defines when this LGPT starts.

Table 10-52. STARTCFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1-0	LGPT0	R/W	0h	LGPT start

10.6.4 CTL Register (Offset = Ch) [Reset = 00000000h]

CTL is shown in Table 10-53.

Return to the Summary Table.

Timer Control

Table 10-53. CTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2RST	W	0h	Channel 2 reset. 0h = No effect. 1h = Reset C2CC, PC2CC, and C2CFG.
9	C1RST	W	0h	Channel 1 reset. 0h = No effect. 1h = Reset C1CC, PC1CC, and C1CFG.
8	C0RST	W	0h	Channel 0 reset. 0h = No effect. 1h = Reset C0CC, PC0CC, and C0CFG.
7-6	RESERVED	R	0h	Reserved
5	INTP	R/W	0h	Interrupt Phase. This bit field controls when the RIS.TGT and RIS.ZERO interrupts are set. 0h = RIS.TGT and RIS.ZERO are set one system clock cycle after CNTR = TARGET/ZERO. 1h = RIS.TGT and RIS.ZERO are set one timer clock cycle after CNTR = TARGET/ZERO.
4-3	CMPDIR	R/W	0h	Compare direction. This bit field controls the direction the counter must have in order to set the RIS.CnCC channel interrupts. This bitfield is only relevant if [CnCFG.CCACT] is configured to a compare action. 0h = Compare RIS fields are set on up count and down count. 1h = Compare RIS fields are only set on up count. 2h = Compare RIS fields are only set on down count. 3h = RESERVED

Table 10-53. CTL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
2-0	MODE	R/W	0h	Timer mode control The CNTR restarts from 0 when MODE is written to UP_ONCE, UP_PER, UPDWN_PER, QDEC, SYNC_UP_ONCE, SYNC_UP_PER or SYNC_UPDWN_PER. When writing MODE all internally queued updates to the channels and TGT is cleared. When configuring the timer, MODE should be the last thing to configure. If changing timer configuration after MODE has been set is necessary, instructions, if any, given in the configuration registers should be followed. See for example C0CFG. 0h = Disable timer. Updates to counter, channels, and events stop. 1h = Count up once. The timer increments from 0 to target value, then stops and sets MODE to DIS. 2h = Count up periodically. The timer increments from 0 to target value, repeatedly. Period = (target value + 1) * timer clock period 3h = Count up and down periodically. The timer counts from 0 to target value and back to 0, repeatedly. Period = (target value * 2) * timer clock period 4h = The timer functions as a quadrature decoder. IOC input 0, IOC input 1 and IOC input 2 are used respectivly as PHA, PHB and IDX inputs. IDX can be turned off by setting C2CFG.EDGE = NONE. The timer clock frequency sets the sample rate of the QDEC logic. This frequency can be configured in PRECFG. 5h = Start counting up once synchronous to another LGPT, selected within STARTCFG. The timer is started by setting CTL.MODE = UP_ONCE automatically. It then functions as a normal timer in CTL.MODE = UP_ONCE, incrementing from 0 to target value, then stops and sets MODE to DIS. 6h = Start counting up periodically synchronous to another LGPT, selected within STARTCFG. The timer is started by setting CTL.MODE = UP_PER automatically. It then operates as a normal timer in CTL.MODE = UP_PER, incrementing from 0 to target value, repeatedly. Period = (target value * 2) * timer clock period 7h = Start counting up and down periodically synchronous to another LGPT, selected within STARTCFG. The timer is started by setting CTL.MODE = UPDWN_PER automatically. It then operates as a normal timer in CTL.MODE = UPDWN_PER, counting from 0 to target value and back to 0, repeatedly. Period = (target value * 2) * timer clock period

10.6.5 OUTCTL Register (Offset = 10h) [Reset = 00000000h]

OUTCTL is shown in Table 10-54.

Return to the Summary Table.

Output Control
Set and clear individual outputs manually. Manual update of an output takes priority over automatic channel updates to the same output. It is not possible to set and clear an output at the same time, such requests will be neglected.
An output can be automatically cleared, set, toggled, or pulsed by each channel, listed in decreasing order of priority. The action with highest priority happens when multiple channels want to update an output at the same time.
All outputs are connected to the event fabric and the IO controller. The outputs going to the IO controller have an aditional complementary output, this output is the inverted IO output. Both the IO and the IO complementary outputs are passed through an IO Controller, see IOCTL.

Table 10-54. OUTCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Reserved
5	SETOUT2	W	0h	Set output 2. Write 1 to set output 2.
4	CLROUT2	W	0h	Clear output 2. Write 1 to clear output 2.
3	SETOUT1	W	0h	Set output 1. Write 1 to set output 1.
2	CLROUT1	W	0h	Clear output 1. Write 1 to clear output 1.
1	SETOUT0	W	0h	Set output 0. Write 1 to set output 0.
0	CLROUT0	W	0h	Clear output 0. Write 1 to clear output 0.

10.6.6 CNTR Register (Offset = 14h) [Reset = 00000000h]

CNTR is shown in Table 10-55.

Return to the Summary Table.

Counter
The counter of this timer. After CTL.MODE is set the counter updates at the rate specified in PRECFG.

Table 10-55. CNTR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Current counter value. If CTL.MODE = QDEC this can be used to set the initial counter value during QDEC. Writing to CNTR in other modes than QDEC is possible, but may result in unpredictable behavior.

10.6.7 PRECFG Register (Offset = 18h) [Reset = 00000000h]

PRECFG is shown in Table 10-56.

Return to the Summary Table.

Clock Prescaler Configuration
This register is used to set the timer clock period. The prescaler is a counter which counts down from the value TICKDIV. When the prescaler counter reaches zero, CNTR is updated. The field TICKDIV effectively divides the prescaler tick source. The timer clock frequency can be calculated as TICKSRC/(TICKDIV+1).

Table 10-56. PRECFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-8	TICKDIV	R/W	0h	Tick division. TICKDIV determines the timer clock frequency for the counter, and timer output updates. The timer clock frequency is the clock selected by TICKSRC divided by (TICKDIV + 1). This inverse is the timer clock period. 0x00: Divide by 1. 0x01: Divide by 2. ... 0xFF: Divide by 256.
7-2	RESERVED	R	0h	Reserved
1-0	TICKSRC	R/W	0h	Prescaler tick source. TICKSRC determines the source which decrements the prescaler. 0h = Prescaler is updated at the system clock. 1h = Prescaler is updated at the rising edge of TICKEN. 2h = Prescaler is updated at the falling edge of TICKEN. 3h = Prescaler is updated at both edges of TICKEN.

10.6.8 PREEVENT Register (Offset = 1Ch) [Reset = 00000000h]

PREEVENT is shown in Table 10-57.

Return to the Summary Table.

Prescaler Event
This register is used to output a logic high signal before the zero crossing of the prescaler counter. The output is routed to the IOC.

Table 10-57. PREEVENT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	VAL	R/W	0h	Sets the HIGH time of the prescaler event output. Event goes high when the prescaler counter equals VAL. Event goes low when prescaler counter is 0. Note: - Can be used to precharge or turn an external component on for a short time before sampling, like in QDEC. - If there is a requirement to create such events that have very short periods compared to timer clock period, use two timers. One timer acts as prescaler and event generator for another timer.

10.6.9 CHFILT Register (Offset = 20h) [Reset = 00000000h]

CHFILT is shown in Table 10-58.

Return to the Summary Table.

Channel Input Filter
This register is used to configure the filter on the channel inputs. The configuration is for all inputs.
The filter is enabled when a channel is in capture mode.
The input to the filter is passed to the edge detection logic if LOAD + 1 consecutive input samples are equal. The filter functions as a down counter, counting down every input sample.
If two consecutive samples are unequal, the filter counter restarts from LOAD.
If the filter counter reaches zero, the input signal is valid and passed to the edge detection logic.
The channel filter should only be configured while the CTL.MODE = DIS. Configuring the filter while the timer is running can result in unexpected behavior.

Table 10-58. CHFILT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-8	LOAD	R/W	0h	The input of the channel filter is passed to the edge detection logic after LOAD + 1 consecutive equal samples.
7-2	RESERVED	R	0h	Reserved
1-0	MODE	R/W	0h	Channel filter mode 0h = Filter is bypassed. No Filter is used. 1h = Filter is clocked by system clock. 2h = Filter is clocked by PRECFG.TICKSRC. 3h = Filter is clocked by timer clock.

10.6.10 FAULT Register (Offset = 24h) [Reset = 00000000h]

FAULT is shown in Table 10-59.

Return to the Summary Table.

Fault
This register is used to configure the fault input logic.
Primary use scenario is to select CTL before starting the timer. Follow these steps to configure CTL while CTL.MODE is different from DIS:
- Set C0CFG.EDGE to NONE.
- Configure CTL.
- Wait for three system clock periods before setting C0CFG.EDGE different from NONE.
These steps prevent fault detection caused by expired signal values in synchronizers and edge-detection circuit.

Table 10-59. FAULT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1-0	CTL	R/W	0h	Fault control On active fault input the counter can optionally stop. If the counter stops this is done by hardware, software must then restart the timer if wanted. The fault input overrides channel 0 IOC input when CTL != DIS. This means that channel 0 receives fault as input signal when C0CFG.INPUT = IO and CTL != DIS. CHFILT can be used to avoid glitching on the fault input. Fault is level triggered, the polarity is set by the C0CFG.EDGE field. Here C0CFG.EDGE = RISE gives active high and C0CFG.EDGE = FALL gives active low polarity. Fault is typically used together with PARK to stop the PWM signal to an external motor control circuit safely. Configure PARK to ensure predefined values of the PWM outputs. If CTL != DIS the RIS.FAULT interrupt is set immediately when the fault input is active while CTL.MODE != DIS. The three modes of fault is described below: CTL = IMMEDIATE In this mode the counter stops immediately on an active fault input. This is done by hardware by setting CTL.MODE = DIS. To start the counter software must set CTL.MODE != DIS. When the counter has stopped, the input synchronizers and the channel filter is not running. This means that if RIS.FAULT is cleared it will not be set again while CTL.MODE = DIS. CTL = ZEROCOND In this mode the counter stops when CNTR = 0 after an active fault input. If the RIS.FAULT interrupt has been cleared by software before CNTR = 0, and the fault input is inactive, the counter will continue as normal. When the counter stops on zero, it can be started again by clearing the RIS.FAULT interrupt if the fault input is inactive. To change the counter mode set CTL.MODE = DIS, clear the RIS.FAULT interrupt, then start timer in wanted mode. CTL = IRQ In this mode only the RIS.FAULT flag is set on an active fault input. 0h = Disable. The timer ignores fault. 1h = Immediate reaction. The counter stops immediately on fault. 2h = Zero condition. The counter stops when CNTR = 0. 3h = Interrupt request. Only set RIS.FAULT on active fault.

10.6.11 PARK Register (Offset = 28h) [Reset = 00000000h]

PARK is shown in Table 10-60.

Return to the Summary Table.

Park
This register configures how the outputs should be set in Park mode. Park mode is either entered by debug halt or fault. Park mode is activated when the counter stops. Park mode is inactive when the counter starts. When park mode is active all outputs are set to their predefined states.
For IO output signals which have enabled deadband, a deadband insertion will be done before switching to the predefined state.

Table 10-60. PARK Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7	IOCPS2	R/W	0h	IO Complementary Park State 2 Park state for IO Complementary output 2. 0h = Output is set low in park mode. 1h = Output is set high in park mode.
6	IOPS2	R/W	0h	IO Park State 2 Park state for IO output 2. 0h = Output is set low in park mode. 1h = Output is set high in park mode.
5	IOCPS1	R/W	0h	IO Complementary Park State 1 Park state for IO Complementary output 1. 0h = Output is set low in park mode. 1h = Output is set high in park mode.
4	IOPS1	R/W	0h	IO Park State 1 Park state for IO output 1. 0h = Output is set low in park mode. 1h = Output is set high in park mode.
3	IOCPS0	R/W	0h	IO Complementary Park State 0 Park state for IO Complementary output 0. 0h = Output is set low in park mode. 1h = Output is set high in park mode.
2	IOPS0	R/W	0h	IO Park State 0 Park state for IO output 0. 0h = Output is set low in park mode. 1h = Output is set high in park mode.
1-0	CTL	R/W	0h	Park Control. 0h = Disable park mode. 1h = Enter park mode on fault. 2h = Enter park mode on debug. 3h = Enter parkmode on fault or debug.

10.6.12 DBDLY Register (Offset = 2Ch) [Reset = 00000000h]

DBDLY is shown in Table 10-61.

Return to the Summary Table.

Deadband Delay
This register is used to insert a deadband delay when generating complementary PWM signals. To enable deadband, on for example IO output 0, create a reference PWM signal on Output 0, then set DBCTL.IOC0 = EN. TBD: 12-bit width fall delay and rise delay may be excessive, if 8-bits are enough we can join DBDLY and DBCTL.

Table 10-61. DBDLY Register Field Descriptions

Bit	Field	Type	Reset	Description
31-28	RESERVED	R	0h	Reserved
27-16	FALLDLY	R/W	0h	Fall delay. The number of system clock periods inserted between the fall of the deadband reference signal and the rise of the inverted output signal.
15-12	RESERVED	R	0h	Reserved
11-0	RISEDLY	R/W	0h	Rise delay. The number of system clock periods inserted between the rise of the deadband reference signal and the rise of the output signal.

10.6.13 DBCTL Register (Offset = 30h) [Reset = 00000000h]

DBCTL is shown in Table 10-62.

Return to the Summary Table.

Deadband Control
This register is used to enable deadband for IOC outputs.

Table 10-62. DBCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-3	RESERVED	R	0h	Reserved
2	IO2	R/W	0h	Enable deadband on IO and IO complementary output 2. 0h = Disable 1h = Enable
1	IO1	R/W	0h	Enable deadband on IO and IO complementary output 1. 0h = Disable 1h = Enable
0	IO0	R/W	0h	Enable deadband on IO and IO complementary output 0. 0h = Disable 1h = Enable

10.6.14 DMA Register (Offset = 3Ch) [Reset = 00000000h]

DMA is shown in Table 10-63.

Return to the Summary Table.

Direct Memory Accsess
This register is used to enable DMA requests from the timer and set the register addresses which the DMA will access (read/write).
Choose DMA request source by setting the REQ field. The setting of the corresponding interrupt in the RIS registers also sets the DMA request.
Upon a DMA request defined by REQ an internal address pointer is set to RWADDR*4. Every access to DMARW will increment the internal pointer by 4 such that the next DMA access will be to the next register.
The internal pointer will stop after RWCNTR increments. Further access will be ignored.

Table 10-63. DMA Register Field Descriptions

Bit	Field	Type	Reset	Description
31-20	RESERVED	R	0h	Reserved
19-16	RWCNTR	R/W	0h	The read/write counter. RWCNTR+1 is the number of times the DMA can access (read/write) the DMARW register. For each DMA access to DMARW an internal counter is incremented, writing to the next address field. RWADDR + 4*RWCNTR is the final register address which can be accessed by the DMA.
15	RESERVED	R	0h	Reserved
14-8	RWADDR	R/W	0h	The base address which the DMA access when reading/writing DMARW. The base address is set by taking the 9 LSB of the physical address and divide by 4. For example, if you wanted the RWADDR to point to the PTGT register you should set RWADDR = 0x0FC/4.
7-4	RESERVED	R	0h	Reserved
3-0	REQ	R/W	0h	0h = Disabled 1h = Setting of RIS.TGT generates a DMA request. 2h = Setting of RIS.ZERO generates a DMA request. 3h = Setting of RIS.FAULT generates a DMA request. 4h = Setting of RIS.C0CC generates a DMA request. 5h = Setting of RIS.C1CC generates a DMA request. 6h = Setting of RIS.C2CC generates a DMA request. 7h = Setting of RIS.C3CC generates a DMA request. 8h = Setting of RIS.C4CC generates a DMA request. 9h = Setting of RIS.C5CC generates a DMA request. Ah = Setting of RIS.C6CC generates a DMA request. Bh = Setting of RIS.C7CC generates a DMA request. Ch = Setting of RIS.C8CC generates a DMA request. Dh = Setting of RIS.C9CC generates a DMA request. Eh = Setting of RIS.C10CC generates a DMA request. Fh = Setting of RIS.C11CC generates a DMA request.

10.6.15 DMARW Register (Offset = 40h) [Reset = 00000000h]

DMARW is shown in Table 10-64.

Return to the Summary Table.

Direct Memory Access
This register is used by the DMA to access (read/write) register inside this LGPT module.
Each access to this register will increment the internal DMA address counter. See DMA for description.

Table 10-64. DMARW Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	DMA read write value. The value that is read/written from/to the registers.

10.6.16 ADCTRG Register (Offset = 44h) [Reset = 00000000h]

ADCTRG is shown in Table 10-65.

Return to the Summary Table.

ADC Trigger
This register is used to enable ADC trigger from the timer.
Choose ADC trigger source by setting the SRC field. The setting of the corresponding interrupt in the RIS registers also sets the ADC trigger.

Table 10-65. ADCTRG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3-0	SRC	R/W	0h	0h = Disabled 1h = Setting of RIS.TGT generates an ADC trigger. 2h = Setting of RIS.ZERO generates an ADC trigger. 3h = Setting of RIS.FAULT generates an ADC trigger. 4h = Setting of RIS.C0CC generates an ADC trigger. 5h = Setting of RIS.C1CC generates an ADC trigger. 6h = Setting of RIS.C2CC generates an ADC trigger. 7h = Setting of RIS.C3CC generates an ADC trigger. 8h = Setting of RIS.C4CC generates an ADC trigger. 9h = Setting of RIS.C5CC generates an ADC trigger. Ah = Setting of RIS.C6CC generates an ADC trigger. Bh = Setting of RIS.C7CC generates an ADC trigger. Ch = Setting of RIS.C8CC generates an ADC trigger. Dh = Setting of RIS.C9CC generates an ADC trigger. Eh = Setting of RIS.C10CC generates an ADC trigger. Fh = Setting of RIS.C11CC generates an ADC trigger.

10.6.17 IOCTL Register (Offset = 48h) [Reset = 00000000h]

IOCTL is shown in Table 10-66.

Return to the Summary Table.

IO Controller
This register overrides the IO outputs.

Table 10-66. IOCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-12	RESERVED	R	0h	Reserved
11-10	COUT2	R/W	0h	IO complementary output 2 control This bit field controls IO complementary output 2. 0h = Normal output. The IO complementary output is not changed. 1h = Driven low. The IO complementary output is driven low. 2h = Driven high. The IO complementary output is driven high. 3h = Inverted value. The IO complementary output is inverted.
9-8	OUT2	R/W	0h	IO output 2 control This bit field controls IO output 2. 0h = Normal output. The IO output is not changed. 1h = Driven low. The IO output is driven low. 2h = Driven high. The IO output is driven high. 3h = Inverted value. The IO output is inverted.
7-6	COUT1	R/W	0h	IO complementary output 1 control This bit field controls IO complementary output 1. 0h = Normal output. The IO complementary output is not changed. 1h = Driven low. The IO complementary output is driven low. 2h = Driven high. The IO complementary output is driven high. 3h = Inverted value. The IO complementary output is inverted.
5-4	OUT1	R/W	0h	IO output 1 control This bit field controls IO output 1. 0h = Normal output. The IO output is not changed. 1h = Driven low. The IO output is driven low. 2h = Driven high. The IO output is driven high. 3h = Inverted value. The IO output is inverted.
3-2	COUT0	R/W	0h	IO complementary output 0 control This bit field controls IO complementary output 0. 0h = Normal output. The IO complementary output is not changed. 1h = Driven low. The IO complementary output is driven low. 2h = Driven high. The IO complementary output is driven high. 3h = Inverted value. The IO complementary output is inverted.
1-0	OUT0	R/W	0h	IO output 0 control This bit field controls IO output 0. 0h = Normal output. The IO output is not changed. 1h = Driven low. The IO output is driven low. 2h = Driven high. The IO output is driven high. 3h = Inverted value. The IO output is inverted.

10.6.18 IMASK Register (Offset = 68h) [Reset = 00000000h]

IMASK is shown in Table 10-67.

Return to the Summary Table.

Interrupt mask.
This register selects interrupt sources which are allowed to pass from RIS to MIS when the corresponding bit-fields are set to 1.

Table 10-67. IMASK Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	R/W	0h	Enable RIS.C2CC interrupt. 0h = Disable 1h = Enable
9	C1CC	R/W	0h	Enable RIS.C1CC interrupt. 0h = Disable 1h = Enable
8	C0CC	R/W	0h	Enable RIS.C0CC interrupt. 0h = Disable 1h = Enable
7	RESERVED	R	0h	Reserved
6	FAULT	R/W	0h	Enable RIS.FAULT interrupt. 0h = Disable 1h = Enable
5	IDX	R/W	0h	Enable RIS.IDX interrupt. 0h = Disable 1h = Enable
4	DIRCHNG	R/W	0h	Enable RIS.DIRCHNG interrupt. 0h = Disable 1h = Enable
3	CNTRCHNG	R/W	0h	Enable RIS.CNTRCHNG interrupt. 0h = Disable 1h = Enable
2	DBLTRANS	R/W	0h	Enable RIS.DBLTRANS interrupt. 0h = Disable 1h = Enable
1	ZERO	R/W	0h	Enable RIS.ZERO interrupt. 0h = Disable 1h = Enable
0	TGT	R/W	0h	Enable RIS.TGT interrupt. 0h = Disable 1h = Enable

10.6.19 RIS Register (Offset = 6Ch) [Reset = 00000000h]

RIS is shown in Table 10-68.

Return to the Summary Table.

Raw interrupt status.
This register reflects the state of all pending interrupts, regardless of masking. This register allows the user to implement a poll scheme. A flag set in this register can be cleared by writing 1 to the corresponding ICLR register bit.

Table 10-68. RIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	R	0h	Status of the C2CC interrupt. The interrupt is set when C2CC has capture or compare event. 0h = Cleared 1h = Set
9	C1CC	R	0h	Status of the C1CC interrupt. The interrupt is set when C1CC has capture or compare event. 0h = Cleared 1h = Set
8	C0CC	R	0h	Status of the C0CC interrupt. The interrupt is set when C0CC has capture or compare event. 0h = Cleared 1h = Set
7	RESERVED	R	0h	Reserved
6	FAULT	R	0h	Status of the FAULT interrupt. The interrupt is set immediately on active fault input. 0h = Cleared 1h = Set
5	IDX	R	0h	Status of the IDX interrupt. The interrupt is set when IDX is active. 0h = Cleared 1h = Set
4	DIRCHNG	R	0h	Status of the DIRCHNG interrupt. The interrupt is set when the direction of the counter changes. 0h = Cleared 1h = Set
3	CNTRCHNG	R	0h	Status of the CNTRCHNG interrupt. The interrupt is set when the counter increments or decrements. 0h = Cleared 1h = Set
2	DBLTRANS	R	0h	Status of the DBLTRANS interrupt. The interrupt is set when a double transition has happened during QDEC mode. 0h = Cleared 1h = Set
1	ZERO	R	0h	Status of the ZERO interrupt. The interrupt is set when CNTR = 0. 0h = Cleared 1h = Set
0	TGT	R	0h	Status of the TGT interrupt. The interrupt is set when CNTR = TGT. 0h = Cleared 1h = Set

10.6.20 MIS Register (Offset = 70h) [Reset = 00000000h]

MIS is shown in Table 10-69.

Return to the Summary Table.

Masked interrupt status.
This register is simply a bitwise AND of the contents of IMASK and RIS.*] registers. A flag set in this register can be cleared by writing 1 to the corresponding ICLR register bit.

Table 10-69. MIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	R	0h	Masked status of the RIS.C2CC interrupt. 0h = Cleared 1h = Set
9	C1CC	R	0h	Masked status of the RIS.C1CC interrupt. 0h = Cleared 1h = Set
8	C0CC	R	0h	Masked status of the RIS.C0CC interrupt. 0h = Cleared 1h = Set
7	RESERVED	R	0h	Reserved
6	FAULT	R	0h	Masked status of the RIS.FAULT interrupt. 0h = Cleared 1h = Set
5	IDX	R	0h	Masked status of the RIS.IDX interrupt. 0h = Cleared 1h = Set
4	DIRCHNG	R	0h	Masked status of the RIS.DIRCHNG interrupt. 0h = Cleared 1h = Set
3	CNTRCHNG	R	0h	Masked status of the RIS.CNTRCHNG interrupt. 0h = Cleared 1h = Set
2	DBLTRANS	R	0h	Masked status of the RIS.DBLTRANS interrupt. 0h = Cleared 1h = Set
1	ZERO	R	0h	Masked status of the RIS.ZERO interrupt. 0h = Cleared 1h = Set
0	TGT	R	0h	Masked status of the RIS.TGT interrupt. 0h = Cleared 1h = Set

10.6.21 ISET Register (Offset = 74h) [Reset = 00000000h]

ISET is shown in Table 10-70.

Return to the Summary Table.

Interrupt set register.
This register can used by software for diagnostics and safety checking purposes. Writing a 1 to a bit in this register will set the event and the corresponding RIS bit also gets set. If the corresponding IMASK bit is set, then the corresponding MIS register bit also gets set.

Table 10-70. ISET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	W	0h	Set the RIS.C2CC interrupt. 0h = No effect 1h = Set
9	C1CC	W	0h	Set the RIS.C1CC interrupt. 0h = No effect 1h = Set
8	C0CC	W	0h	Set the RIS.C0CC interrupt. 0h = No effect 1h = Set
7	RESERVED	R	0h	Reserved
6	FAULT	W	0h	Set the RIS.FAULT interrupt. 0h = No effect 1h = Set
5	IDX	W	0h	Set the RIS.IDX interrupt. 0h = No effect 1h = Set
4	DIRCHNG	W	0h	Set the RIS.DIRCHNG interrupt. 0h = No effect 1h = Set
3	CNTRCHNG	W	0h	Set the RIS.CNTRCHNG interrupt. 0h = No effect 1h = Set
2	DBLTRANS	W	0h	Set the RIS.DBLTRANS interrupt. 0h = No effect 1h = Set
1	ZERO	W	0h	Set the RIS.ZERO interrupt. 0h = No effect 1h = Set
0	TGT	W	0h	Set the RIS.TGT interrupt. 0h = No effect 1h = Set

10.6.22 ICLR Register (Offset = 78h) [Reset = 00000000h]

ICLR is shown in Table 10-71.

Return to the Summary Table.

Interrupt clear register.
This register allows software to clear interrupts. Writing a 1 to a bit in this register will clear the event and the corresponding RIS bit also gets cleared. If the corresponding IMASK bit is set, then the corresponding MIS register bit also gets cleared.

Table 10-71. ICLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	W	0h	Clear the RIS.C2CC interrupt. 0h = No effect 1h = Clear
9	C1CC	W	0h	Clear the RIS.C1CC interrupt. 0h = No effect 1h = Clear
8	C0CC	W	0h	Clear the RIS.C0CC interrupt. 0h = No effect 1h = Clear
7	RESERVED	R	0h	Reserved
6	FAULT	W	0h	Clear the RIS.FAULT interrupt. 0h = No effect 1h = Clear
5	IDX	W	0h	Clear the RIS.IDX interrupt. 0h = No effect 1h = Clear
4	DIRCHNG	W	0h	Clear the RIS.DIRCHNG interrupt. 0h = No effect 1h = Clear
3	CNTRCHNG	W	0h	Clear the RIS.CNTRCHNG interrupt. 0h = No effect 1h = Clear
2	DBLTRANS	W	0h	Clear the RIS.DBLTRANS interrupt. 0h = No effect 1h = Clear
1	ZERO	W	0h	Clear the RIS.ZERO interrupt. 0h = No effect 1h = Clear
0	TGT	W	0h	Clear the RIS.TGT interrupt. 0h = No effect 1h = Clear

10.6.23 IMSET Register (Offset = 7Ch) [Reset = 00000000h]

IMSET is shown in Table 10-72.

Return to the Summary Table.

Interrupt mask set register.
Writing a 1 to a bit in this register will set the corresponding IMASK bit.

Table 10-72. IMSET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	W	0h	Set the MIS.C2CC mask. 0h = No effect 1h = Set
9	C1CC	W	0h	Set the MIS.C1CC mask. 0h = No effect 1h = Set
8	C0CC	W	0h	Set the MIS.C0CC mask. 0h = No effect 1h = Set
7	RESERVED	R	0h	Reserved
6	FAULT	W	0h	Set the MIS.FAULT mask. 0h = No effect 1h = Set
5	IDX	W	0h	Set the MIS.IDX mask. 0h = No effect 1h = Set
4	DIRCHNG	W	0h	Set the MIS.DIRCHNG mask. 0h = No effect 1h = Set
3	CNTRCHNG	W	0h	Set the MIS.CNTRCHNG mask. 0h = No effect 1h = Set
2	DBLTRANS	W	0h	Set the MIS.DBLTRANS mask. 0h = No effect 1h = Set
1	ZERO	W	0h	Set the MIS.ZERO mask. 0h = No effect 1h = Set
0	TGT	W	0h	Set the MIS.TGT mask. 0h = No effect 1h = Set

10.6.24 IMCLR Register (Offset = 80h) [Reset = 00000000h]

IMCLR is shown in Table 10-73.

Return to the Summary Table.

Interrupt mask clear register.
Writing a 1 to a bit in this register will clear the corresponding IMASK bit.

Table 10-73. IMCLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	W	0h	Clear the MIS.C2CC mask. 0h = No effect 1h = Clear
9	C1CC	W	0h	Clear the MIS.C1CC mask. 0h = No effect 1h = Clear
8	C0CC	W	0h	Clear the MIS.C0CC mask. 0h = No effect 1h = Clear
7	RESERVED	R	0h	Reserved
6	FAULT	W	0h	Clear the MIS.FAULT mask. 0h = No effect 1h = Clear
5	IDX	W	0h	Clear the MIS.IDX mask. 0h = No effect 1h = Clear
4	DIRCHNG	W	0h	Clear the MIS.DIRCHNG mask. 0h = No effect 1h = Clear
3	CNTRCHNG	W	0h	Clear the MIS.CNTRCHNG mask. 0h = No effect 1h = Clear
2	DBLTRANS	W	0h	Clear the MIS.DBLTRANS mask. 0h = No effect 1h = Clear
1	ZERO	W	0h	Clear the MIS.ZERO mask. 0h = No effect 1h = Clear
0	TGT	W	0h	Clear the MIS.TGT mask. 0h = No effect 1h = Clear

10.6.25 EMU Register (Offset = 84h) [Reset = 00000000h]

EMU is shown in Table 10-74.

Return to the Summary Table.

Debug control
This register can be used to freeze the timer when CPU halts when HALT is set to 1. When HALT is set to 0, or when the CPU releases debug halt, the filters and edge detection logic is flushed and the timer starts. For setting a predefined output value during a CPU debug halt, PARK, if the timer has this register, should be configured additionally. If this timer does not have the PARK register a predefined output value during CPU halt is not possible.

Table 10-74. EMU Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1	CTL	R/W	0h	Halt control. Configure when the counter shall stop upon CPU halt. This bitfield only applies if HALT = 1. 0h = Immediate reaction. The counter stops immediately on debug halt. 1h = Zero condition. The counter stops when CNTR = 0.
0	HALT	R/W	0h	Halt LGPT when CPU is halted in debug. 0h = Disable. 1h = Enable.

10.6.26 C0CFG Register (Offset = C0h) [Reset = 00000000h]

C0CFG is shown in Table 10-75.

Return to the Summary Table.

Channel 0 Configuration
This register configures channel function and enables outputs.
Each channel has an edge-detection circuit. The the edge-detection circuit is:
- enabled while CCACT selects a capture function and CTL.MODE is different from DIS.
- flushed while CCACT selects a capture function and CTL.MODE is changed from DIS to another mode.
The flush action uses two system clock periods. It prevents capture events caused by expired signal values stored in the edge-detection circuit.
The channel input signal enters the edge-detection circuit. False capture events can occur when:
- the edge-detection circuit contains expired signal samples and the circuit is enabled without flush as described above.
- the CCACT field is reconfigured while CTL.MODE is different from DIS.
Primary use scenario is to select CCACT before starting the timer. Follow these steps to configure CCACT to a capture action while CTL.MODE is different from DIS:
- Set EDGE to NONE.
- Configure CCACT.
- Wait for three system clock periods before setting EDGE different from NONE.
These steps prevent capture events caused by expired signal values in edge-detection circuit.

Table 10-75. C0CFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	OUT2	R/W	0h	Output 2 enable. When 0 < CCACT < 8, OUT2 becomes zero after a capture or compare event. 0h = Channel 0 does not control output 2. 1h = Channel 0 controls output 2.
9	OUT1	R/W	0h	Output 1 enable. When 0 < CCACT < 8, OUT1 becomes zero after a capture or compare event. 0h = Channel 0 does not control output 1. 1h = Channel 0 controls output 1.
8	OUT0	R/W	0h	Output 0 enable. When 0 < CCACT < 8, OUT0 becomes zero after a capture or compare event. 0h = Channel 0 does not control output 0. 1h = Channel 0 controls output 0.
7	RESERVED	R	0h	Reserved
6	INPUT	R/W	0h	Select channel input. 0h = Event fabric 1h = IO controller
5-4	EDGE	R/W	0h	Determines the edge that triggers the channel input event. This happens post filter. 0h = Input is turned off. 1h = Input event is triggered at rising edge. 2h = Input event is triggered at falling edge. 3h = Input event is triggered at both edges.

Table 10-75. C0CFG Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
3-0	CCACT	R/W	0h	Capture-Compare action. Capture-Compare action defines 15 different channel functions that utilize capture, compare, and zero events. In every compare event the timer looks at the current value of CNTR. The corresponding output event will be set 1 timer period after CNTR = C0CC. 0h = Disable channel. 1h = Set on capture, and then disable channel. Channel function sequence: - Set enabled outputs on capture event and copy CNTR.VAL to C0CC.VAL. - Disable channel. Primary use scenario is to select this function before starting the timer. Follow these steps to select this function while CTL.MODE is different from DIS: - Set CCACT to SET_ON_CAPT with no output enable. - Configure INPUT (optional). - Wait for three timer clock periods as defined in PRECFG before setting CCACT to SET_ON_CAPT_DIS. Output enable is optional. These steps prevent capture events caused by expired signal values in edge-detection circuit. 2h = Clear on zero, toggle on compare, and then disable channel. Channel function sequence: - Clear enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C0CC.VAL = CNTR.VAL. - Disable channel. Enabled outputs are set when C0CC.VAL = 0 and CNTR.VAL = 0. 3h = Set on zero, toggle on compare, and then disable channel. Channel function sequence: - Set enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C0CC.VAL = CNTR.VAL. - Disable channel. Enabled outputs are cleared when C0CC.VAL = 0 and CNTR.VAL = 0. 4h = Clear on compare, and then disable channel. Channel function sequence: - Clear enabled outputs when C0CC.VAL = CNTR.VAL. - Disable channel. 5h = Set on compare, and then disable channel. Channel function sequence: - Set enabled outputs when C0CC.VAL = CNTR.VAL. - Disable channel. 6h = Toggle on compare, and then disable channel. Channel function sequence: - Toggle enabled outputs when C0CC.VAL = CNTR.VAL. - Disable channel. 7h = Pulse on compare, and then disable channel. Channel function sequence: - Pulse enabled outputs when C0CC.VAL = CNTR.VAL. - Disable channel. The output is high for two timer clock periods. 8h = Period and pulse width measurement. Continuously capture period and pulse width of the signal selected by INPUT relative to the signal edge given by EDGE. Set enabled outputs and RIS.C0CC when C0CC.VAL contains signal period and PC0CC.VAL contains signal pulse width. Notes: - Make sure to configure INPUT and CCACT when CTL.MODE equals DIS, then set CTL.MODE to UP_ONCE or UP_PER. - The counter restarts in the selected timer mode when C0CC.VAL contains the signal period. - If more than one channel uses this function, the channels will perform this function one at a time. The channel with lowest number has priority and performs the function first. Next measurement starts

Table 10-75. C0CFG Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
				when current measurement completes successfully or times out. A timeout occurs when counter equals target. - To observe a timeout event the RIS.TGT interrupt can be used, or another channel can be configured to SET_ON_CMP with compare value equal TGT. Signal property requirements: - Signal Period >= 2 * (1 + PRECFG.TICKDIV) * timer clock period. - Signal Period <= MAX(CNTR) * (1 + PRECFG.TICKDIV) * timer clock period. - Signal low and high phase >= (1 + PRECFG.TICKDIV) * timer clock period. 9h = Set on capture repeatedly. Channel function sequence: - Set enabled outputs on capture event and copy CNTR.VAL to C0CC.VAL. Ah = Clear on zero, toggle on compare repeatedly. Channel function sequence: - Clear enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C0CC.VAL = CNTR.VAL. Set CTL.MODE to UPDWN_PER for center-aligned PWM generation. Duty cycle is given by: When C0CC.VAL <= TGT.VAL: Duty cycle = 1 - (C0CC.VAL / TGT.VAL). When C0CC.VAL > TGT.VAL: Duty cycle = 0. Enabled outputs are set when C0CC.VAL = 0 and CNTR.VAL = 0. Bh = Set on zero, toggle on compare repeatedly. Channel function sequence: - Set enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C0CC.VAL = CNTR.VAL. Set CTL.MODE to UP_PER for edge-aligned PWM generation. Duty cycle is given by: When C0CC.VAL <= TGT.VAL: Duty cycle = C0CC.VAL / (TGT.VAL + 1). When C0CC.VAL > TGT.VAL: Duty cycle = 1. Enabled outputs are cleared when C0CC.VAL = 0 and CNTR.VAL = 0. Ch = Clear on compare repeatedly. Channel function sequence: - Clear enabled outputs when C0CC.VAL = CNTR.VAL. Dh = Set on compare repeatedly. Channel function sequence: - Set enabled outputs when C0CC.VAL = CNTR.VAL. Eh = Toggle on compare repeatedly. Channel function sequence: - Toggle enabled outputs when C0CC.VAL = CNTR.VAL. Fh = Pulse on compare repeatedly. Channel function sequence: - Pulse enabled outputs when C0CC.VAL = CNTR.VAL. The output is high for two timer clock periods.

10.6.27 C1CFG Register (Offset = C4h) [Reset = 00000000h]

C1CFG is shown in Table 10-76.

Return to the Summary Table.

Channel 1 Configuration
This register configures channel function and enables outputs.
Each channel has an edge-detection circuit. The the edge-detection circuit is:
- enabled while CCACT selects a capture function and CTL.MODE is different from DIS.
- flushed while CCACT selects a capture function and CTL.MODE is changed from DIS to another mode.
The flush action uses two system clock periods. It prevents capture events caused by expired signal values stored in the edge-detection circuit.
The channel input signal enters the edge-detection circuit. False capture events can occur when:
- the edge-detection circuit contains expired signal samples and the circuit is enabled without flush as described above.
- the CCACT field is reconfigured while CTL.MODE is different from DIS.
Primary use scenario is to select CCACT before starting the timer. Follow these steps to configure CCACT to a capture action while CTL.MODE is different from DIS:
- Set EDGE to NONE.
- Configure CCACT.
- Wait for three system clock periods before setting EDGE different from NONE.
These steps prevent capture events caused by expired signal values in edge-detection circuit.

Table 10-76. C1CFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	OUT2	R/W	0h	Output 2 enable. When 0 < CCACT < 8, OUT2 becomes zero after a capture or compare event. 0h = Channel 1 does not control output 2. 1h = Channel 1 controls output 2.
9	OUT1	R/W	0h	Output 1 enable. When 0 < CCACT < 8, OUT1 becomes zero after a capture or compare event. 0h = Channel 1 does not control output 1. 1h = Channel 1 controls output 1.
8	OUT0	R/W	0h	Output 0 enable. When 0 < CCACT < 8, OUT0 becomes zero after a capture or compare event. 0h = Channel 1 does not control output 0. 1h = Channel 1 controls output 0.
7	RESERVED	R	0h	Reserved
6	INPUT	R/W	0h	Select channel input. 0h = Event fabric 1h = IO controller
5-4	EDGE	R/W	0h	Determines the edge that triggers the channel input event. This happens post filter. 0h = Input is turned off. 1h = Input event is triggered at rising edge. 2h = Input event is triggered at falling edge. 3h = Input event is triggered at both edges.

Table 10-76. C1CFG Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
3-0	CCACT	R/W	0h	Capture-Compare action. Capture-Compare action defines 15 different channel functions that utilize capture, compare, and zero events. In every compare event the timer looks at the current value of CNTR. The corresponding output event will be set 1 timer period after CNTR = C1CC. 0h = Disable channel. 1h = Set on capture, and then disable channel. Channel function sequence: - Set enabled outputs on capture event and copy CNTR.VAL to C1CC.VAL. - Disable channel. Primary use scenario is to select this function before starting the timer. Follow these steps to select this function while CTL.MODE is different from DIS: - Set CCACT to SET_ON_CAPT with no output enable. - Configure INPUT (optional). - Wait for three timer clock periods as defined in PRECFG before setting CCACT to SET_ON_CAPT_DIS. Output enable is optional. These steps prevent capture events caused by expired signal values in edge-detection circuit. 2h = Clear on zero, toggle on compare, and then disable channel. Channel function sequence: - Clear enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C1CC.VAL = CNTR.VAL. - Disable channel. Enabled outputs are set when C1CC.VAL = 0 and CNTR.VAL = 0. 3h = Set on zero, toggle on compare, and then disable channel. Channel function sequence: - Set enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C1CC.VAL = CNTR.VAL. - Disable channel. Enabled outputs are cleared when C1CC.VAL = 0 and CNTR.VAL = 0. 4h = Clear on compare, and then disable channel. Channel function sequence: - Clear enabled outputs when C1CC.VAL = CNTR.VAL. - Disable channel. 5h = Set on compare, and then disable channel. Channel function sequence: - Set enabled outputs when C1CC.VAL = CNTR.VAL. - Disable channel. 6h = Toggle on compare, and then disable channel. Channel function sequence: - Toggle enabled outputs when C1CC.VAL = CNTR.VAL. - Disable channel. 7h = Pulse on compare, and then disable channel. Channel function sequence: - Pulse enabled outputs when C1CC.VAL = CNTR.VAL. - Disable channel. The output is high for two timer clock periods. 8h = Period and pulse width measurement. Continuously capture period and pulse width of the signal selected by INPUT relative to the signal edge given by EDGE. Set enabled outputs and RIS.C1CC when C1CC.VAL contains signal period and PC1CC.VAL contains signal pulse width. Notes: - Make sure to configure INPUT and CCACT when CTL.MODE equals DIS, then set CTL.MODE to UP_ONCE or UP_PER. - The counter restarts in the selected timer mode when C1CC.VAL contains the signal period. - If more than one channel uses this function, the channels will perform this function one at a time. The channel with lowest number has priority and performs the function first. Next measurement starts

Table 10-76. C1CFG Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
				when current measurement completes successfully or times out. A timeout occurs when counter equals target. - To observe a timeout event the RIS.TGT interrupt can be used, or another channel can be configured to SET_ON_CMP with compare value equal TGT. Signal property requirements: - Signal Period >= 2 * (1 + PRECFG.TICKDIV) * timer clock period. - Signal Period <= MAX(CNTR) * (1 + PRECFG.TICKDIV) * timer clock period. - Signal low and high phase >= (1 + PRECFG.TICKDIV) * timer clock period. 9h = Set on capture repeatedly. Channel function sequence: - Set enabled outputs on capture event and copy CNTR.VAL to C1CC.VAL. Ah = Clear on zero, toggle on compare repeatedly. Channel function sequence: - Clear enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C1CC.VAL = CNTR.VAL. Set CTL.MODE to UPDWN_PER for center-aligned PWM generation. Duty cycle is given by: When C1CC.VAL <= TGT.VAL: Duty cycle = 1 - (C1CC.VAL / TGT.VAL). When C1CC.VAL > TGT.VAL: Duty cycle = 0. Enabled outputs are set when C1CC.VAL = 0 and CNTR.VAL = 0. Bh = Set on zero, toggle on compare repeatedly. Channel function sequence: - Set enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C1CC.VAL = CNTR.VAL. Set CTL.MODE to UP_PER for edge-aligned PWM generation. Duty cycle is given by: When C1CC.VAL <= TGT.VAL: Duty cycle = C1CC.VAL / (TGT.VAL + 1). When C1CC.VAL > TGT.VAL: Duty cycle = 1. Enabled outputs are cleared when C1CC.VAL = 0 and CNTR.VAL = 0. Ch = Clear on compare repeatedly. Channel function sequence: - Clear enabled outputs when C1CC.VAL = CNTR.VAL. Dh = Set on compare repeatedly. Channel function sequence: - Set enabled outputs when C1CC.VAL = CNTR.VAL. Eh = Toggle on compare repeatedly. Channel function sequence: - Toggle enabled outputs when C1CC.VAL = CNTR.VAL. Fh = Pulse on compare repeatedly. Channel function sequence: - Pulse enabled outputs when C1CC.VAL = CNTR.VAL. The output is high for two timer clock periods.

10.6.28 C2CFG Register (Offset = C8h) [Reset = 00000000h]

C2CFG is shown in Table 10-77.

Return to the Summary Table.

Channel 2 Configuration
This register configures channel function and enables outputs.
Each channel has an edge-detection circuit. The the edge-detection circuit is:
- enabled while CCACT selects a capture function and CTL.MODE is different from DIS.
- flushed while CCACT selects a capture function and CTL.MODE is changed from DIS to another mode.
The flush action uses two system clock periods. It prevents capture events caused by expired signal values stored in the edge-detection circuit.
The channel input signal enters the edge-detection circuit. False capture events can occur when:
- the edge-detection circuit contains expired signal samples and the circuit is enabled without flush as described above.
- the CCACT field is reconfigured while CTL.MODE is different from DIS.
Primary use scenario is to select CCACT before starting the timer. Follow these steps to configure CCACT to a capture action while CTL.MODE is different from DIS:
- Set EDGE to NONE.
- Configure CCACT.
- Wait for three system clock periods before setting EDGE different from NONE.
These steps prevent capture events caused by expired signal values in edge-detection circuit.

Table 10-77. C2CFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	OUT2	R/W	0h	Output 2 enable. When 0 < CCACT < 8, OUT2 becomes zero after a capture or compare event. 0h = Channel 2 does not control output 2. 1h = Channel 2 controls output 2.
9	OUT1	R/W	0h	Output 1 enable. When 0 < CCACT < 8, OUT1 becomes zero after a capture or compare event. 0h = Channel 2 does not control output 1. 1h = Channel 2 controls output 1.
8	OUT0	R/W	0h	Output 0 enable. When 0 < CCACT < 8, OUT0 becomes zero after a capture or compare event. 0h = Channel 2 does not control output 0. 1h = Channel 2 controls output 0.
7	RESERVED	R	0h	Reserved
6	INPUT	R/W	0h	Select channel input. 0h = Event fabric 1h = IO controller
5-4	EDGE	R/W	0h	Determines the edge that triggers the channel input event. This happens post filter. 0h = Input is turned off. 1h = Input event is triggered at rising edge. 2h = Input event is triggered at falling edge. 3h = Input event is triggered at both edges.

Table 10-77. C2CFG Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
3-0	CCACT	R/W	0h	Capture-Compare action. Capture-Compare action defines 15 different channel functions that utilize capture, compare, and zero events. In every compare event the timer looks at the current value of CNTR. The corresponding output event will be set 1 timer period after CNTR = C2CC. 0h = Disable channel. 1h = Set on capture, and then disable channel. Channel function sequence: - Set enabled outputs on capture event and copy CNTR.VAL to C2CC.VAL. - Disable channel. Primary use scenario is to select this function before starting the timer. Follow these steps to select this function while CTL.MODE is different from DIS: - Set CCACT to SET_ON_CAPT with no output enable. - Configure INPUT (optional). - Wait for three timer clock periods as defined in PRECFG before setting CCACT to SET_ON_CAPT_DIS. Output enable is optional. These steps prevent capture events caused by expired signal values in edge-detection circuit. 2h = Clear on zero, toggle on compare, and then disable channel. Channel function sequence: - Clear enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C2CC.VAL = CNTR.VAL. - Disable channel. Enabled outputs are set when C2CC.VAL = 0 and CNTR.VAL = 0. 3h = Set on zero, toggle on compare, and then disable channel. Channel function sequence: - Set enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C2CC.VAL = CNTR.VAL. - Disable channel. Enabled outputs are cleared when C2CC.VAL = 0 and CNTR.VAL = 0. 4h = Clear on compare, and then disable channel. Channel function sequence: - Clear enabled outputs when C2CC.VAL = CNTR.VAL. - Disable channel. 5h = Set on compare, and then disable channel. Channel function sequence: - Set enabled outputs when C2CC.VAL = CNTR.VAL. - Disable channel. 6h = Toggle on compare, and then disable channel. Channel function sequence: - Toggle enabled outputs when C2CC.VAL = CNTR.VAL. - Disable channel. 7h = Pulse on compare, and then disable channel. Channel function sequence: - Pulse enabled outputs when C2CC.VAL = CNTR.VAL. - Disable channel. The output is high for two timer clock periods. 8h = Period and pulse width measurement. Continuously capture period and pulse width of the signal selected by INPUT relative to the signal edge given by EDGE. Set enabled outputs and RIS.C2CC when C2CC.VAL contains signal period and PC2CC.VAL contains signal pulse width. Notes: - Make sure to configure INPUT and CCACT when CTL.MODE equals DIS, then set CTL.MODE to UP_ONCE or UP_PER. - The counter restarts in the selected timer mode when C2CC.VAL contains the signal period. - If more than one channel uses this function, the channels will perform this function one at a time. The channel with lowest number has priority and performs the function first. Next measurement starts

Table 10-77. C2CFG Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
				when current measurement completes successfully or times out. A timeout occurs when counter equals target. - To observe a timeout event the RIS.TGT interrupt can be used, or another channel can be configured to SET_ON_CMP with compare value equal TGT. Signal property requirements: - Signal Period >= 2 * (1 + PRECFG.TICKDIV) * timer clock period. - Signal Period <= MAX(CNTR) * (1 + PRECFG.TICKDIV) * timer clock period. - Signal low and high phase >= (1 + PRECFG.TICKDIV) * timer clock period. 9h = Set on capture repeatedly. Channel function sequence: - Set enabled outputs on capture event and copy CNTR.VAL to C2CC.VAL. Ah = Clear on zero, toggle on compare repeatedly. Channel function sequence: - Clear enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C2CC.VAL = CNTR.VAL. Set CTL.MODE to UPDWN_PER for center-aligned PWM generation. Duty cycle is given by: When C2CC.VAL <= TGT.VAL: Duty cycle = 1 - (C2CC.VAL / TGT.VAL). When C2CC.VAL > TGT.VAL: Duty cycle = 0. Enabled outputs are set when C2CC.VAL = 0 and CNTR.VAL = 0. Bh = Set on zero, toggle on compare repeatedly. Channel function sequence: - Set enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C2CC.VAL = CNTR.VAL. Set CTL.MODE to UP_PER for edge-aligned PWM generation. Duty cycle is given by: When C2CC.VAL <= TGT.VAL: Duty cycle = C2CC.VAL / (TGT.VAL + 1). When C2CC.VAL > TGT.VAL: Duty cycle = 1. Enabled outputs are cleared when C2CC.VAL = 0 and CNTR.VAL = 0. Ch = Clear on compare repeatedly. Channel function sequence: - Clear enabled outputs when C2CC.VAL = CNTR.VAL. Dh = Set on compare repeatedly. Channel function sequence: - Set enabled outputs when C2CC.VAL = CNTR.VAL. Eh = Toggle on compare repeatedly. Channel function sequence: - Toggle enabled outputs when C2CC.VAL = CNTR.VAL. Fh = Pulse on compare repeatedly. Channel function sequence: - Pulse enabled outputs when C2CC.VAL = CNTR.VAL. The output is high for two timer clock periods.

10.6.29 PTGT Register (Offset = FCh) [Reset = 00000000h]

PTGT is shown in Table 10-78.

Return to the Summary Table.

Pipeline Target
A read or write to this register will clear the RIS.ZERO and RIS.TGT interrupt.
If CTL.MODE != QDEC.
Target value for next counter period.
The timer will copy PTGT.VAL to TGT.VAL on the upcoming CNTR zero crossing only if PTGT.VAL has been written. The copy does not happen when restarting the timer.
This is useful to avoid period jitter in PWM applications with time-varying period, sometimes referenced as phase corrected PWM.
If CTL.MODE = QDEC
The CNTR value is updated with VALUE on IDX if the counter is counting down. If the counter is counting up, CNTR is loaded with zero on IDX.
In this mode the VALUE is not loaded into TGT on zero crossing.

Table 10-78. PTGT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	The pipleline target value.

10.6.30 PC0CC Register (Offset = 100h) [Reset = 00000000h]

PC0CC is shown in Table 10-79.

Return to the Summary Table.

Pipeline Channel 0 Capture Compare

Table 10-79. PC0CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Pipeline Capture Compare value. User defined pipeline compare value or channel-updated capture value. A read or write to this register will clear the RIS.C0CC interrupt. Compare mode: An update of VAL will be transferred to C0CC.VAL when the next CNTR.VAL is zero and CTL.MODE is different from DIS. This is useful for PWM generation and prevents jitter on the edges of the generated signal. Capture mode: When C0CFG.CCACT equals PER_PULSE_WIDTH_MEAS then VAL contains the width of the low or high phase of the selected signal. This is specified by C0CFG.EDGE.

10.6.31 PC1CC Register (Offset = 104h) [Reset = 00000000h]

PC1CC is shown in Table 10-80.

Return to the Summary Table.

Pipeline Channel 1 Capture Compare

Table 10-80. PC1CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Pipeline Capture Compare value. User defined pipeline compare value or channel-updated capture value. A read or write to this register will clear the RIS.C1CC interrupt. Compare mode: An update of VAL will be transferred to C1CC.VAL when the next CNTR.VAL is zero and CTL.MODE is different from DIS. This is useful for PWM generation and prevents jitter on the edges of the generated signal. Capture mode: When C1CFG.CCACT equals PER_PULSE_WIDTH_MEAS then VAL contains the width of the low or high phase of the selected signal. This is specified by C1CFG.EDGE.

10.6.32 PC2CC Register (Offset = 108h) [Reset = 00000000h]

PC2CC is shown in Table 10-81.

Return to the Summary Table.

Pipeline Channel 2 Capture Compare

Table 10-81. PC2CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Pipeline Capture Compare value. User defined pipeline compare value or channel-updated capture value. A read or write to this register will clear the RIS.C2CC interrupt. Compare mode: An update of VAL will be transferred to C2CC.VAL when the next CNTR.VAL is zero and CTL.MODE is different from DIS. This is useful for PWM generation and prevents jitter on the edges of the generated signal. Capture mode: When C2CFG.CCACT equals PER_PULSE_WIDTH_MEAS then VAL contains the width of the low or high phase of the selected signal. This is specified by C2CFG.EDGE.

10.6.33 TGT Register (Offset = 13Ch) [Reset = 0000FFFFh]

TGT is shown in Table 10-82.

Return to the Summary Table.

Target
User defined counter target.
A read or write to this register will clear the RIS.ZERO and RIS.TGT interrupt.

Table 10-82. TGT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	FFFFh	User defined counter target value.

10.6.34 C0CC Register (Offset = 140h) [Reset = 00000000h]

C0CC is shown in Table 10-83.

Return to the Summary Table.

Channel 0 Capture Compare

Table 10-83. C0CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Capture Compare value. User defined compare value or channel-updated capture value. A read or write to this register will clear the RIS.C0CC interrupt. Compare mode: VAL is compared against CNTR.VAL and an event is generated as specified by C0CFG.CCACT when these are equal. Capture mode: The current counter value is stored in VAL when a capture event occurs. C0CFG.CCACT determines if VAL is a signal period or a regular capture value.

10.6.35 C1CC Register (Offset = 144h) [Reset = 00000000h]

C1CC is shown in Table 10-84.

Return to the Summary Table.

Channel 1 Capture Compare

Table 10-84. C1CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Capture Compare value. User defined compare value or channel-updated capture value. A read or write to this register will clear the RIS.C1CC interrupt. Compare mode: VAL is compared against CNTR.VAL and an event is generated as specified by C1CFG.CCACT when these are equal. Capture mode: The current counter value is stored in VAL when a capture event occurs. C1CFG.CCACT determines if VAL is a signal period or a regular capture value.

10.6.36 C2CC Register (Offset = 148h) [Reset = 00000000h]

C2CC is shown in Table 10-85.

Return to the Summary Table.

Channel 2 Capture Compare

Table 10-85. C2CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Capture Compare value. User defined compare value or channel-updated capture value. A read or write to this register will clear the RIS.C2CC interrupt. Compare mode: VAL is compared against CNTR.VAL and an event is generated as specified by C2CFG.CCACT when these are equal. Capture mode: The current counter value is stored in VAL when a capture event occurs. C2CFG.CCACT determines if VAL is a signal period or a regular capture value.

10.6.37 PTGTNC Register (Offset = 17Ch) [Reset = 00000000h]

PTGTNC is shown in Table 10-86.

Return to the Summary Table.

Pipeline Target No Clear
Use this register to read or write to PTGT without clearing the RIS.ZERO and RIS.TGT interrupt.

Table 10-86. PTGTNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	A read or write to this register will not clear the RIS.TGT interrupt. If CTL.MODE != QDEC. Target value for next counter period. The timer copies VAL to TGT.VAL when CNTR.VAL becomes 0. The copy does not happen when restarting the timer. This is useful to avoid period jitter in PWM applications with time-varying period, sometimes referenced as phase corrected PWM. If CTL.MODE = QDEC. The CNTR.VAL is updated with VAL on IDX. VAL is not loaded into TGT.VAL when CNTR.VAL becomes 0.

10.6.38 PC0CCNC Register (Offset = 180h) [Reset = 00000000h]

PC0CCNC is shown in Table 10-87.

Return to the Summary Table.

Pipeline Channel 0 Capture Compare No Clear

Table 10-87. PC0CCNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Pipeline Capture Compare value. User defined pipeline compare value or channel-updated capture value. A read or write to this register will not clear the RIS.C0CC interrupt. Compare mode: An update of VAL will be transferred to C0CC.VAL when the next CNTR.VAL is zero and CTL.MODE is different from DIS. This is useful for PWM generation and prevents jitter on the edges of the generated signal. Capture mode: When C0CFG.CCACT equals PER_PULSE_WIDTH_MEAS then VAL contains the width of the low or high phase of the selected signal. This is specified by C0CFG.EDGE.

10.6.39 PC1CCNC Register (Offset = 184h) [Reset = 00000000h]

PC1CCNC is shown in Table 10-88.

Return to the Summary Table.

Pipeline Channel 1 Capture Compare No Clear

Table 10-88. PC1CCNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Pipeline Capture Compare value. User defined pipeline compare value or channel-updated capture value. A read or write to this register will not clear the RIS.C1CC interrupt. Compare mode: An update of VAL will be transferred to C1CC.VAL when the next CNTR.VAL is zero and CTL.MODE is different from DIS. This is useful for PWM generation and prevents jitter on the edges of the generated signal. Capture mode: When C1CFG.CCACT equals PER_PULSE_WIDTH_MEAS then VAL contains the width of the low or high phase of the selected signal. This is specified by C1CFG.EDGE.

10.6.40 PC2CCNC Register (Offset = 188h) [Reset = 00000000h]

PC2CCNC is shown in Table 10-89.

Return to the Summary Table.

Pipeline Channel 2 Capture Compare No Clear

Table 10-89. PC2CCNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Pipeline Capture Compare value. User defined pipeline compare value or channel-updated capture value. A read or write to this register will not clear the RIS.C2CC interrupt. Compare mode: An update of VAL will be transferred to C2CC.VAL when the next CNTR.VAL is zero and CTL.MODE is different from DIS. This is useful for PWM generation and prevents jitter on the edges of the generated signal. Capture mode: When C2CFG.CCACT equals PER_PULSE_WIDTH_MEAS then VAL contains the width of the low or high phase of the selected signal. This is specified by C2CFG.EDGE.

10.6.41 TGTNC Register (Offset = 1BCh) [Reset = 0000FFFFh]

TGTNC is shown in Table 10-90.

Return to the Summary Table.

Target No Clear
Use this register to read or write to TGT without clearing the RIS.ZERO and RIS.TGT interrupt.

Table 10-90. TGTNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	FFFFh	User defined counter target value.

10.6.42 C0CCNC Register (Offset = 1C0h) [Reset = 00000000h]

C0CCNC is shown in Table 10-91.

Return to the Summary Table.

Channel 0 Capture Compare No Clear

Table 10-91. C0CCNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Capture Compare value. User defined compare value or channel-updated capture value. A read or write to this register will not clear the RIS.C0CC interrupt. Compare mode: VAL is compared against CNTR.VAL and an event is generated as specified by C0CFG.CCACT when these are equal. Capture mode: The current counter value is stored in VAL when a capture event occurs. C0CFG.CCACT determines if VAL is a signal period or a regular capture value.

10.6.43 C1CCNC Register (Offset = 1C4h) [Reset = 00000000h]

C1CCNC is shown in Table 10-92.

Return to the Summary Table.

Channel 1 Capture Compare No Clear

Table 10-92. C1CCNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Capture Compare value. User defined compare value or channel-updated capture value. A read or write to this register will not clear the RIS.C1CC interrupt. Compare mode: VAL is compared against CNTR.VAL and an event is generated as specified by C1CFG.CCACT when these are equal. Capture mode: The current counter value is stored in VAL when a capture event occurs. C1CFG.CCACT determines if VAL is a signal period or a regular capture value.

10.6.44 C2CCNC Register (Offset = 1C8h) [Reset = 00000000h]

C2CCNC is shown in Table 10-93.

Return to the Summary Table.

Channel 2 Capture Compare No Clear

Table 10-93. C2CCNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Capture Compare value. User defined compare value or channel-updated capture value. A read or write to this register will not clear the RIS.C2CC interrupt. Compare mode: VAL is compared against CNTR.VAL and an event is generated as specified by C2CFG.CCACT when these are equal. Capture mode: The current counter value is stored in VAL when a capture event occurs. C2CFG.CCACT determines if VAL is a signal period or a regular capture value.

10.7 LGPT2 Registers

Table 10-94 lists the memory-mapped registers for the LGPT2 registers. All register offset addresses not listed in Table 10-94 should be considered as reserved locations and the register contents should not be modified.

Table 10-94. LGPT2 Registers

Offset	Acronym	Register Name	Section
0h	DESC	Description Register.	Go
4h	DESCEX	Description Extended	Go
8h	STARTCFG	Start Configuration	Go
Ch	CTL	Timer Control	Go
10h	OUTCTL	Output Control	Go
14h	CNTR	Counter	Go
18h	PRECFG	Clock Prescaler Configuration	Go
1Ch	PREEVENT	Prescaler Event	Go
20h	CHFILT	Channel Input Filter	Go
34h	QDECSTAT	Quadrature Decoder Status	Go
3Ch	DMA	Direct Memory Accsess	Go
40h	DMARW	Direct Memory Access	Go
44h	ADCTRG	ADC Trigger	Go
48h	IOCTL	IO Controller	Go
68h	IMASK	Interrupt mask.	Go
6Ch	RIS	Raw interrupt status.	Go
70h	MIS	Masked interrupt status.	Go
74h	ISET	Interrupt set register.	Go
78h	ICLR	Interrupt clear register.	Go
7Ch	IMSET	Interrupt mask set register.	Go
80h	IMCLR	Interrupt mask clear register.	Go
84h	EMU	Debug control	Go
C0h	C0CFG	Channel 0 Configuration	Go
C4h	C1CFG	Channel 1 Configuration	Go
C8h	C2CFG	Channel 2 Configuration	Go
FCh	PTGT	Pipeline Target	Go
100h	PC0CC	Pipeline Channel 0 Capture Compare	Go
104h	PC1CC	Pipeline Channel 1 Capture Compare	Go
108h	PC2CC	Pipeline Channel 2 Capture Compare	Go
13Ch	TGT	Target	Go
140h	C0CC	Channel 0 Capture Compare	Go
144h	C1CC	Channel 1 Capture Compare	Go
148h	C2CC	Channel 2 Capture Compare	Go
17Ch	PTGTNC	Pipeline Target No Clear	Go
180h	PC0CCNC	Pipeline Channel 0 Capture Compare No Clear	Go
184h	PC1CCNC	Pipeline Channel 1 Capture Compare No Clear	Go
188h	PC2CCNC	Pipeline Channel 2 Capture Compare No Clear	Go
1BCh	TGTNC	Target No Clear	Go
1C0h	C0CCNC	Channel 0 Capture Compare No Clear	Go
1C4h	C1CCNC	Channel 1 Capture Compare No Clear	Go
1C8h	C2CCNC	Channel 2 Capture Compare No Clear	Go

Complex bit access types are encoded to fit into small table cells. Table 10-95 shows the codes that are used for access types in this section.

Table 10-95. LGPT2 Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

10.7.1 DESC Register (Offset = 0h) [Reset = DE491010h]

DESC is shown in Table 10-96.

Return to the Summary Table.

Description Register.
This register provides IP module ID, revision information, instance index and standard MMR registers offset.

Table 10-96. DESC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	MODID	R	DE49h	Module identifier used to uniquely identify this IP.
15-12	STDIPOFF	R	1h	Standard IP MMR block offset. Standard IP MMRs are the set of from aggregated IRQ registers till DTB. 0: Standard IP MMRs do not exist 0x1-0xF: Standard IP MMRs begin at offset of (64*STDIPOFF from the base IP address)
11-8	INSTIDX	R	0h	IP Instance ID number. If multiple instances of IP exist in the device, this field can identify the instance number.
7-4	MAJREV	R	1h	Major revision of IP.
3-0	MINREV	R	0h	Minor revision of IP.

10.7.2 DESCEX Register (Offset = 4h) [Reset = 000238C3h]

DESCEX is shown in Table 10-97.

Return to the Summary Table.

Description Extended
This register describes the parameters of the LGPT.

Table 10-97. DESCEX Register Field Descriptions

Bit	Field	Type	Reset	Description
31-20	RESERVED	R	0h	Reserved
19	HIR	R	0h	Has IR logic.
18	HDBF	R	0h	Has deadband, fault, and park logic.
17-14	PREW	R	8h	Prescale width. The prescaler can maximum be configured to 2^{PREW}-1.
13	HQDEC	R	1h	Has Quadrature Decoder.
12	HCIF	R	1h	Has channel input filter.
11-8	CIFS	R	8h	Channel input filter size. The prevailing state filter can maximum be configured to 2^{CIFS}-1.
7	HDMA	R	1h	Has uDMA output and logic.
6	HINT	R	1h	Has interrupt output and logic.
5-4	CNTRW	R	0h	Counter bit-width. The maximum counter value is equal to 2^{CNTRW}-1. 0h = 16-bit counter. 1h = 24-bit counter. 2h = 32-bit counter. 3h = RESERVED
3-0	NCH	R	3h	Number of channels.

10.7.3 STARTCFG Register (Offset = 8h) [Reset = 00000000h]

STARTCFG is shown in Table 10-98.

Return to the Summary Table.

Start Configuration
This register is only for when CTL.MODE is configured to one of the SYNC modes.
This register defines when this LGPT starts.

Table 10-98. STARTCFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1-0	LGPT0	R/W	0h	LGPT start

10.7.4 CTL Register (Offset = Ch) [Reset = 00000000h]

CTL is shown in Table 10-99.

Return to the Summary Table.

Timer Control

Table 10-99. CTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2RST	W	0h	Channel 2 reset. 0h = No effect. 1h = Reset C2CC, PC2CC, and C2CFG.
9	C1RST	W	0h	Channel 1 reset. 0h = No effect. 1h = Reset C1CC, PC1CC, and C1CFG.
8	C0RST	W	0h	Channel 0 reset. 0h = No effect. 1h = Reset C0CC, PC0CC, and C0CFG.
7-6	RESERVED	R	0h	Reserved
5	INTP	R/W	0h	Interrupt Phase. This bit field controls when the RIS.TGT and RIS.ZERO interrupts are set. 0h = RIS.TGT and RIS.ZERO are set one system clock cycle after CNTR = TARGET/ZERO. 1h = RIS.TGT and RIS.ZERO are set one timer clock cycle after CNTR = TARGET/ZERO.
4-3	CMPDIR	R/W	0h	Compare direction. This bit field controls the direction the counter must have in order to set the RIS.CnCC channel interrupts. This bitfield is only relevant if [CnCFG.CCACT] is configured to a compare action. 0h = Compare RIS fields are set on up count and down count. 1h = Compare RIS fields are only set on up count. 2h = Compare RIS fields are only set on down count. 3h = RESERVED

Table 10-99. CTL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
2-0	MODE	R/W	0h	Timer mode control The CNTR restarts from 0 when MODE is written to UP_ONCE, UP_PER, UPDWN_PER, QDEC, SYNC_UP_ONCE, SYNC_UP_PER or SYNC_UPDWN_PER. When writing MODE all internally queued updates to the channels and TGT is cleared. When configuring the timer, MODE should be the last thing to configure. If changing timer configuration after MODE has been set is necessary, instructions, if any, given in the configuration registers should be followed. See for example C0CFG. 0h = Disable timer. Updates to counter, channels, and events stop. 1h = Count up once. The timer increments from 0 to target value, then stops and sets MODE to DIS. 2h = Count up periodically. The timer increments from 0 to target value, repeatedly. Period = (target value + 1) * timer clock period 3h = Count up and down periodically. The timer counts from 0 to target value and back to 0, repeatedly. Period = (target value * 2) * timer clock period 4h = The timer functions as a quadrature decoder. IOC input 0, IOC input 1 and IOC input 2 are used respectivly as PHA, PHB and IDX inputs. IDX can be turned off by setting C2CFG.EDGE = NONE. The timer clock frequency sets the sample rate of the QDEC logic. This frequency can be configured in PRECFG. 5h = Start counting up once synchronous to another LGPT, selected within STARTCFG. The timer is started by setting CTL.MODE = UP_ONCE automatically. It then functions as a normal timer in CTL.MODE = UP_ONCE, incrementing from 0 to target value, then stops and sets MODE to DIS. 6h = Start counting up periodically synchronous to another LGPT, selected within STARTCFG. The timer is started by setting CTL.MODE = UP_PER automatically. It then operates as a normal timer in CTL.MODE = UP_PER, incrementing from 0 to target value, repeatedly. Period = (target value * 2) * timer clock period 7h = Start counting up and down periodically synchronous to another LGPT, selected within STARTCFG. The timer is started by setting CTL.MODE = UPDWN_PER automatically. It then operates as a normal timer in CTL.MODE = UPDWN_PER, counting from 0 to target value and back to 0, repeatedly. Period = (target value * 2) * timer clock period

10.7.5 OUTCTL Register (Offset = 10h) [Reset = 00000000h]

OUTCTL is shown in Table 10-100.

Return to the Summary Table.

Output Control
Set and clear individual outputs manually. Manual update of an output takes priority over automatic channel updates to the same output. It is not possible to set and clear an output at the same time, such requests will be neglected.
An output can be automatically cleared, set, toggled, or pulsed by each channel, listed in decreasing order of priority. The action with highest priority happens when multiple channels want to update an output at the same time.
All outputs are connected to the event fabric and the IO controller. The outputs going to the IO controller have an aditional complementary output, this output is the inverted IO output. Both the IO and the IO complementary outputs are passed through an IO Controller, see IOCTL.

Table 10-100. OUTCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Reserved
5	SETOUT2	W	0h	Set output 2. Write 1 to set output 2.
4	CLROUT2	W	0h	Clear output 2. Write 1 to clear output 2.
3	SETOUT1	W	0h	Set output 1. Write 1 to set output 1.
2	CLROUT1	W	0h	Clear output 1. Write 1 to clear output 1.
1	SETOUT0	W	0h	Set output 0. Write 1 to set output 0.
0	CLROUT0	W	0h	Clear output 0. Write 1 to clear output 0.

10.7.6 CNTR Register (Offset = 14h) [Reset = 00000000h]

CNTR is shown in Table 10-101.

Return to the Summary Table.

Counter
The counter of this timer. After CTL.MODE is set the counter updates at the rate specified in PRECFG.

Table 10-101. CNTR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Current counter value. If CTL.MODE = QDEC this can be used to set the initial counter value during QDEC. Writing to CNTR in other modes than QDEC is possible, but may result in unpredictable behavior.

10.7.7 PRECFG Register (Offset = 18h) [Reset = 00000000h]

PRECFG is shown in Table 10-102.

Return to the Summary Table.

Clock Prescaler Configuration
This register is used to set the timer clock period. The prescaler is a counter which counts down from the value TICKDIV. When the prescaler counter reaches zero, CNTR is updated. The field TICKDIV effectively divides the prescaler tick source. The timer clock frequency can be calculated as TICKSRC/(TICKDIV+1).

Table 10-102. PRECFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-8	TICKDIV	R/W	0h	Tick division. TICKDIV determines the timer clock frequency for the counter, and timer output updates. The timer clock frequency is the clock selected by TICKSRC divided by (TICKDIV + 1). This inverse is the timer clock period. 0x00: Divide by 1. 0x01: Divide by 2. ... 0xFF: Divide by 256.
7-2	RESERVED	R	0h	Reserved
1-0	TICKSRC	R/W	0h	Prescaler tick source. TICKSRC determines the source which decrements the prescaler. 0h = Prescaler is updated at the system clock. 1h = Prescaler is updated at the rising edge of TICKEN. 2h = Prescaler is updated at the falling edge of TICKEN. 3h = Prescaler is updated at both edges of TICKEN.

10.7.8 PREEVENT Register (Offset = 1Ch) [Reset = 00000000h]

PREEVENT is shown in Table 10-103.

Return to the Summary Table.

Prescaler Event
This register is used to output a logic high signal before the zero crossing of the prescaler counter. The output is routed to the IOC.

Table 10-103. PREEVENT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	VAL	R/W	0h	Sets the HIGH time of the prescaler event output. Event goes high when the prescaler counter equals VAL. Event goes low when prescaler counter is 0. Note: - Can be used to precharge or turn an external component on for a short time before sampling, like in QDEC. - If there is a requirement to create such events that have very short periods compared to timer clock period, use two timers. One timer acts as prescaler and event generator for another timer.

10.7.9 CHFILT Register (Offset = 20h) [Reset = 00000000h]

CHFILT is shown in Table 10-104.

Return to the Summary Table.

Channel Input Filter
This register is used to configure the filter on the channel inputs. The configuration is for all inputs.
The filter is enabled when a channel is in capture mode.
The input to the filter is passed to the edge detection logic if LOAD + 1 consecutive input samples are equal. The filter functions as a down counter, counting down every input sample.
If two consecutive samples are unequal, the filter counter restarts from LOAD.
If the filter counter reaches zero, the input signal is valid and passed to the edge detection logic.
The channel filter should only be configured while the CTL.MODE = DIS. Configuring the filter while the timer is running can result in unexpected behavior.

Table 10-104. CHFILT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-8	LOAD	R/W	0h	The input of the channel filter is passed to the edge detection logic after LOAD + 1 consecutive equal samples.
7-2	RESERVED	R	0h	Reserved
1-0	MODE	R/W	0h	Channel filter mode 0h = Filter is bypassed. No Filter is used. 1h = Filter is clocked by system clock. 2h = Filter is clocked by PRECFG.TICKSRC. 3h = Filter is clocked by timer clock.

10.7.10 QDECSTAT Register (Offset = 34h) [Reset = 00000000h]

QDECSTAT is shown in Table 10-105.

Return to the Summary Table.

Quadrature Decoder Status
This register can be used during QDEC mode to check the status of the quadrature decoder.

Table 10-105. QDECSTAT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1	DBLTRANS	R	0h	Double transition 0h = Single or no transition on phase inputs. 1h = Double transition on phase inputs.
0	QDIR	R	0h	Direction of count during QDEC mode. 0h = Up (PHA leads PHB) 1h = Down (PHB leads PHA)

10.7.11 DMA Register (Offset = 3Ch) [Reset = 00000000h]

DMA is shown in Table 10-106.

Return to the Summary Table.

Direct Memory Accsess
This register is used to enable DMA requests from the timer and set the register addresses which the DMA will access (read/write).
Choose DMA request source by setting the REQ field. The setting of the corresponding interrupt in the RIS registers also sets the DMA request.
Upon a DMA request defined by REQ an internal address pointer is set to RWADDR*4. Every access to DMARW will increment the internal pointer by 4 such that the next DMA access will be to the next register.
The internal pointer will stop after RWCNTR increments. Further access will be ignored.

Table 10-106. DMA Register Field Descriptions

Bit	Field	Type	Reset	Description
31-20	RESERVED	R	0h	Reserved
19-16	RWCNTR	R/W	0h	The read/write counter. RWCNTR+1 is the number of times the DMA can access (read/write) the DMARW register. For each DMA access to DMARW an internal counter is incremented, writing to the next address field. RWADDR + 4*RWCNTR is the final register address which can be accessed by the DMA.
15	RESERVED	R	0h	Reserved
14-8	RWADDR	R/W	0h	The base address which the DMA access when reading/writing DMARW. The base address is set by taking the 9 LSB of the physical address and divide by 4. For example, if you wanted the RWADDR to point to the PTGT register you should set RWADDR = 0x0FC/4.
7-4	RESERVED	R	0h	Reserved
3-0	REQ	R/W	0h	0h = Disabled 1h = Setting of RIS.TGT generates a DMA request. 2h = Setting of RIS.ZERO generates a DMA request. 3h = Setting of RIS.FAULT generates a DMA request. 4h = Setting of RIS.C0CC generates a DMA request. 5h = Setting of RIS.C1CC generates a DMA request. 6h = Setting of RIS.C2CC generates a DMA request. 7h = Setting of RIS.C3CC generates a DMA request. 8h = Setting of RIS.C4CC generates a DMA request. 9h = Setting of RIS.C5CC generates a DMA request. Ah = Setting of RIS.C6CC generates a DMA request. Bh = Setting of RIS.C7CC generates a DMA request. Ch = Setting of RIS.C8CC generates a DMA request. Dh = Setting of RIS.C9CC generates a DMA request. Eh = Setting of RIS.C10CC generates a DMA request. Fh = Setting of RIS.C11CC generates a DMA request.

10.7.12 DMARW Register (Offset = 40h) [Reset = 00000000h]

DMARW is shown in Table 10-107.

Return to the Summary Table.

Direct Memory Access
This register is used by the DMA to access (read/write) register inside this LGPT module.
Each access to this register will increment the internal DMA address counter. See DMA for description.

Table 10-107. DMARW Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	DMA read write value. The value that is read/written from/to the registers.

10.7.13 ADCTRG Register (Offset = 44h) [Reset = 00000000h]

ADCTRG is shown in Table 10-108.

Return to the Summary Table.

ADC Trigger
This register is used to enable ADC trigger from the timer.
Choose ADC trigger source by setting the SRC field. The setting of the corresponding interrupt in the RIS registers also sets the ADC trigger.

Table 10-108. ADCTRG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3-0	SRC	R/W	0h	0h = Disabled 1h = Setting of RIS.TGT generates an ADC trigger. 2h = Setting of RIS.ZERO generates an ADC trigger. 3h = Setting of RIS.FAULT generates an ADC trigger. 4h = Setting of RIS.C0CC generates an ADC trigger. 5h = Setting of RIS.C1CC generates an ADC trigger. 6h = Setting of RIS.C2CC generates an ADC trigger. 7h = Setting of RIS.C3CC generates an ADC trigger. 8h = Setting of RIS.C4CC generates an ADC trigger. 9h = Setting of RIS.C5CC generates an ADC trigger. Ah = Setting of RIS.C6CC generates an ADC trigger. Bh = Setting of RIS.C7CC generates an ADC trigger. Ch = Setting of RIS.C8CC generates an ADC trigger. Dh = Setting of RIS.C9CC generates an ADC trigger. Eh = Setting of RIS.C10CC generates an ADC trigger. Fh = Setting of RIS.C11CC generates an ADC trigger.

10.7.14 IOCTL Register (Offset = 48h) [Reset = 00000000h]

IOCTL is shown in Table 10-109.

Return to the Summary Table.

IO Controller
This register overrides the IO outputs.

Table 10-109. IOCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-12	RESERVED	R	0h	Reserved
11-10	COUT2	R/W	0h	IO complementary output 2 control This bit field controls IO complementary output 2. 0h = Normal output. The IO complementary output is not changed. 1h = Driven low. The IO complementary output is driven low. 2h = Driven high. The IO complementary output is driven high. 3h = Inverted value. The IO complementary output is inverted.
9-8	OUT2	R/W	0h	IO output 2 control This bit field controls IO output 2. 0h = Normal output. The IO output is not changed. 1h = Driven low. The IO output is driven low. 2h = Driven high. The IO output is driven high. 3h = Inverted value. The IO output is inverted.
7-6	COUT1	R/W	0h	IO complementary output 1 control This bit field controls IO complementary output 1. 0h = Normal output. The IO complementary output is not changed. 1h = Driven low. The IO complementary output is driven low. 2h = Driven high. The IO complementary output is driven high. 3h = Inverted value. The IO complementary output is inverted.
5-4	OUT1	R/W	0h	IO output 1 control This bit field controls IO output 1. 0h = Normal output. The IO output is not changed. 1h = Driven low. The IO output is driven low. 2h = Driven high. The IO output is driven high. 3h = Inverted value. The IO output is inverted.
3-2	COUT0	R/W	0h	IO complementary output 0 control This bit field controls IO complementary output 0. 0h = Normal output. The IO complementary output is not changed. 1h = Driven low. The IO complementary output is driven low. 2h = Driven high. The IO complementary output is driven high. 3h = Inverted value. The IO complementary output is inverted.
1-0	OUT0	R/W	0h	IO output 0 control This bit field controls IO output 0. 0h = Normal output. The IO output is not changed. 1h = Driven low. The IO output is driven low. 2h = Driven high. The IO output is driven high. 3h = Inverted value. The IO output is inverted.

10.7.15 IMASK Register (Offset = 68h) [Reset = 00000000h]

IMASK is shown in Table 10-110.

Return to the Summary Table.

Interrupt mask.
This register selects interrupt sources which are allowed to pass from RIS to MIS when the corresponding bit-fields are set to 1.

Table 10-110. IMASK Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	R/W	0h	Enable RIS.C2CC interrupt. 0h = Disable 1h = Enable
9	C1CC	R/W	0h	Enable RIS.C1CC interrupt. 0h = Disable 1h = Enable
8	C0CC	R/W	0h	Enable RIS.C0CC interrupt. 0h = Disable 1h = Enable
7	RESERVED	R	0h	Reserved
6	FAULT	R/W	0h	Enable RIS.FAULT interrupt. 0h = Disable 1h = Enable
5	IDX	R/W	0h	Enable RIS.IDX interrupt. 0h = Disable 1h = Enable
4	DIRCHNG	R/W	0h	Enable RIS.DIRCHNG interrupt. 0h = Disable 1h = Enable
3	CNTRCHNG	R/W	0h	Enable RIS.CNTRCHNG interrupt. 0h = Disable 1h = Enable
2	DBLTRANS	R/W	0h	Enable RIS.DBLTRANS interrupt. 0h = Disable 1h = Enable
1	ZERO	R/W	0h	Enable RIS.ZERO interrupt. 0h = Disable 1h = Enable
0	TGT	R/W	0h	Enable RIS.TGT interrupt. 0h = Disable 1h = Enable

10.7.16 RIS Register (Offset = 6Ch) [Reset = 00000000h]

RIS is shown in Table 10-111.

Return to the Summary Table.

Raw interrupt status.
This register reflects the state of all pending interrupts, regardless of masking. This register allows the user to implement a poll scheme. A flag set in this register can be cleared by writing 1 to the corresponding ICLR register bit.

Table 10-111. RIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	R	0h	Status of the C2CC interrupt. The interrupt is set when C2CC has capture or compare event. 0h = Cleared 1h = Set
9	C1CC	R	0h	Status of the C1CC interrupt. The interrupt is set when C1CC has capture or compare event. 0h = Cleared 1h = Set
8	C0CC	R	0h	Status of the C0CC interrupt. The interrupt is set when C0CC has capture or compare event. 0h = Cleared 1h = Set
7	RESERVED	R	0h	Reserved
6	FAULT	R	0h	Status of the FAULT interrupt. The interrupt is set immediately on active fault input. 0h = Cleared 1h = Set
5	IDX	R	0h	Status of the IDX interrupt. The interrupt is set when IDX is active. 0h = Cleared 1h = Set
4	DIRCHNG	R	0h	Status of the DIRCHNG interrupt. The interrupt is set when the direction of the counter changes. 0h = Cleared 1h = Set
3	CNTRCHNG	R	0h	Status of the CNTRCHNG interrupt. The interrupt is set when the counter increments or decrements. 0h = Cleared 1h = Set
2	DBLTRANS	R	0h	Status of the DBLTRANS interrupt. The interrupt is set when a double transition has happened during QDEC mode. 0h = Cleared 1h = Set
1	ZERO	R	0h	Status of the ZERO interrupt. The interrupt is set when CNTR = 0. 0h = Cleared 1h = Set
0	TGT	R	0h	Status of the TGT interrupt. The interrupt is set when CNTR = TGT. 0h = Cleared 1h = Set

10.7.17 MIS Register (Offset = 70h) [Reset = 00000000h]

MIS is shown in Table 10-112.

Return to the Summary Table.

Masked interrupt status.
This register is simply a bitwise AND of the contents of IMASK and RIS.*] registers. A flag set in this register can be cleared by writing 1 to the corresponding ICLR register bit.

Table 10-112. MIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	R	0h	Masked status of the RIS.C2CC interrupt. 0h = Cleared 1h = Set
9	C1CC	R	0h	Masked status of the RIS.C1CC interrupt. 0h = Cleared 1h = Set
8	C0CC	R	0h	Masked status of the RIS.C0CC interrupt. 0h = Cleared 1h = Set
7	RESERVED	R	0h	Reserved
6	FAULT	R	0h	Masked status of the RIS.FAULT interrupt. 0h = Cleared 1h = Set
5	IDX	R	0h	Masked status of the RIS.IDX interrupt. 0h = Cleared 1h = Set
4	DIRCHNG	R	0h	Masked status of the RIS.DIRCHNG interrupt. 0h = Cleared 1h = Set
3	CNTRCHNG	R	0h	Masked status of the RIS.CNTRCHNG interrupt. 0h = Cleared 1h = Set
2	DBLTRANS	R	0h	Masked status of the RIS.DBLTRANS interrupt. 0h = Cleared 1h = Set
1	ZERO	R	0h	Masked status of the RIS.ZERO interrupt. 0h = Cleared 1h = Set
0	TGT	R	0h	Masked status of the RIS.TGT interrupt. 0h = Cleared 1h = Set

10.7.18 ISET Register (Offset = 74h) [Reset = 00000000h]

ISET is shown in Table 10-113.

Return to the Summary Table.

Interrupt set register.
This register can used by software for diagnostics and safety checking purposes. Writing a 1 to a bit in this register will set the event and the corresponding RIS bit also gets set. If the corresponding IMASK bit is set, then the corresponding MIS register bit also gets set.

Table 10-113. ISET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	W	0h	Set the RIS.C2CC interrupt. 0h = No effect 1h = Set
9	C1CC	W	0h	Set the RIS.C1CC interrupt. 0h = No effect 1h = Set
8	C0CC	W	0h	Set the RIS.C0CC interrupt. 0h = No effect 1h = Set
7	RESERVED	R	0h	Reserved
6	FAULT	W	0h	Set the RIS.FAULT interrupt. 0h = No effect 1h = Set
5	IDX	W	0h	Set the RIS.IDX interrupt. 0h = No effect 1h = Set
4	DIRCHNG	W	0h	Set the RIS.DIRCHNG interrupt. 0h = No effect 1h = Set
3	CNTRCHNG	W	0h	Set the RIS.CNTRCHNG interrupt. 0h = No effect 1h = Set
2	DBLTRANS	W	0h	Set the RIS.DBLTRANS interrupt. 0h = No effect 1h = Set
1	ZERO	W	0h	Set the RIS.ZERO interrupt. 0h = No effect 1h = Set
0	TGT	W	0h	Set the RIS.TGT interrupt. 0h = No effect 1h = Set

10.7.19 ICLR Register (Offset = 78h) [Reset = 00000000h]

ICLR is shown in Table 10-114.

Return to the Summary Table.

Interrupt clear register.
This register allows software to clear interrupts. Writing a 1 to a bit in this register will clear the event and the corresponding RIS bit also gets cleared. If the corresponding IMASK bit is set, then the corresponding MIS register bit also gets cleared.

Table 10-114. ICLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	W	0h	Clear the RIS.C2CC interrupt. 0h = No effect 1h = Clear
9	C1CC	W	0h	Clear the RIS.C1CC interrupt. 0h = No effect 1h = Clear
8	C0CC	W	0h	Clear the RIS.C0CC interrupt. 0h = No effect 1h = Clear
7	RESERVED	R	0h	Reserved
6	FAULT	W	0h	Clear the RIS.FAULT interrupt. 0h = No effect 1h = Clear
5	IDX	W	0h	Clear the RIS.IDX interrupt. 0h = No effect 1h = Clear
4	DIRCHNG	W	0h	Clear the RIS.DIRCHNG interrupt. 0h = No effect 1h = Clear
3	CNTRCHNG	W	0h	Clear the RIS.CNTRCHNG interrupt. 0h = No effect 1h = Clear
2	DBLTRANS	W	0h	Clear the RIS.DBLTRANS interrupt. 0h = No effect 1h = Clear
1	ZERO	W	0h	Clear the RIS.ZERO interrupt. 0h = No effect 1h = Clear
0	TGT	W	0h	Clear the RIS.TGT interrupt. 0h = No effect 1h = Clear

10.7.20 IMSET Register (Offset = 7Ch) [Reset = 00000000h]

IMSET is shown in Table 10-115.

Return to the Summary Table.

Interrupt mask set register.
Writing a 1 to a bit in this register will set the corresponding IMASK bit.

Table 10-115. IMSET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	W	0h	Set the MIS.C2CC mask. 0h = No effect 1h = Set
9	C1CC	W	0h	Set the MIS.C1CC mask. 0h = No effect 1h = Set
8	C0CC	W	0h	Set the MIS.C0CC mask. 0h = No effect 1h = Set
7	RESERVED	R	0h	Reserved
6	FAULT	W	0h	Set the MIS.FAULT mask. 0h = No effect 1h = Set
5	IDX	W	0h	Set the MIS.IDX mask. 0h = No effect 1h = Set
4	DIRCHNG	W	0h	Set the MIS.DIRCHNG mask. 0h = No effect 1h = Set
3	CNTRCHNG	W	0h	Set the MIS.CNTRCHNG mask. 0h = No effect 1h = Set
2	DBLTRANS	W	0h	Set the MIS.DBLTRANS mask. 0h = No effect 1h = Set
1	ZERO	W	0h	Set the MIS.ZERO mask. 0h = No effect 1h = Set
0	TGT	W	0h	Set the MIS.TGT mask. 0h = No effect 1h = Set

10.7.21 IMCLR Register (Offset = 80h) [Reset = 00000000h]

IMCLR is shown in Table 10-116.

Return to the Summary Table.

Interrupt mask clear register.
Writing a 1 to a bit in this register will clear the corresponding IMASK bit.

Table 10-116. IMCLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	W	0h	Clear the MIS.C2CC mask. 0h = No effect 1h = Clear
9	C1CC	W	0h	Clear the MIS.C1CC mask. 0h = No effect 1h = Clear
8	C0CC	W	0h	Clear the MIS.C0CC mask. 0h = No effect 1h = Clear
7	RESERVED	R	0h	Reserved
6	FAULT	W	0h	Clear the MIS.FAULT mask. 0h = No effect 1h = Clear
5	IDX	W	0h	Clear the MIS.IDX mask. 0h = No effect 1h = Clear
4	DIRCHNG	W	0h	Clear the MIS.DIRCHNG mask. 0h = No effect 1h = Clear
3	CNTRCHNG	W	0h	Clear the MIS.CNTRCHNG mask. 0h = No effect 1h = Clear
2	DBLTRANS	W	0h	Clear the MIS.DBLTRANS mask. 0h = No effect 1h = Clear
1	ZERO	W	0h	Clear the MIS.ZERO mask. 0h = No effect 1h = Clear
0	TGT	W	0h	Clear the MIS.TGT mask. 0h = No effect 1h = Clear

10.7.22 EMU Register (Offset = 84h) [Reset = 00000000h]

EMU is shown in Table 10-117.

Return to the Summary Table.

Debug control
This register can be used to freeze the timer when CPU halts when HALT is set to 1. When HALT is set to 0, or when the CPU releases debug halt, the filters and edge detection logic is flushed and the timer starts. For setting a predefined output value during a CPU debug halt, PARK, if the timer has this register, should be configured additionally. If this timer does not have the PARK register a predefined output value during CPU halt is not possible.

Table 10-117. EMU Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1	CTL	R/W	0h	Halt control. Configure when the counter shall stop upon CPU halt. This bitfield only applies if HALT = 1. 0h = Immediate reaction. The counter stops immediately on debug halt. 1h = Zero condition. The counter stops when CNTR = 0.
0	HALT	R/W	0h	Halt LGPT when CPU is halted in debug. 0h = Disable. 1h = Enable.

10.7.23 C0CFG Register (Offset = C0h) [Reset = 00000000h]

C0CFG is shown in Table 10-118.

Return to the Summary Table.

Channel 0 Configuration
This register configures channel function and enables outputs.
Each channel has an edge-detection circuit. The the edge-detection circuit is:
- enabled while CCACT selects a capture function and CTL.MODE is different from DIS.
- flushed while CCACT selects a capture function and CTL.MODE is changed from DIS to another mode.
The flush action uses two system clock periods. It prevents capture events caused by expired signal values stored in the edge-detection circuit.
The channel input signal enters the edge-detection circuit. False capture events can occur when:
- the edge-detection circuit contains expired signal samples and the circuit is enabled without flush as described above.
- the CCACT field is reconfigured while CTL.MODE is different from DIS.
Primary use scenario is to select CCACT before starting the timer. Follow these steps to configure CCACT to a capture action while CTL.MODE is different from DIS:
- Set EDGE to NONE.
- Configure CCACT.
- Wait for three system clock periods before setting EDGE different from NONE.
These steps prevent capture events caused by expired signal values in edge-detection circuit.

Table 10-118. C0CFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	OUT2	R/W	0h	Output 2 enable. When 0 < CCACT < 8, OUT2 becomes zero after a capture or compare event. 0h = Channel 0 does not control output 2. 1h = Channel 0 controls output 2.
9	OUT1	R/W	0h	Output 1 enable. When 0 < CCACT < 8, OUT1 becomes zero after a capture or compare event. 0h = Channel 0 does not control output 1. 1h = Channel 0 controls output 1.
8	OUT0	R/W	0h	Output 0 enable. When 0 < CCACT < 8, OUT0 becomes zero after a capture or compare event. 0h = Channel 0 does not control output 0. 1h = Channel 0 controls output 0.
7	RESERVED	R	0h	Reserved
6	INPUT	R/W	0h	Select channel input. 0h = Event fabric 1h = IO controller
5-4	EDGE	R/W	0h	Determines the edge that triggers the channel input event. This happens post filter. 0h = Input is turned off. 1h = Input event is triggered at rising edge. 2h = Input event is triggered at falling edge. 3h = Input event is triggered at both edges.

Table 10-118. C0CFG Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
3-0	CCACT	R/W	0h	Capture-Compare action. Capture-Compare action defines 15 different channel functions that utilize capture, compare, and zero events. In every compare event the timer looks at the current value of CNTR. The corresponding output event will be set 1 timer period after CNTR = C0CC. 0h = Disable channel. 1h = Set on capture, and then disable channel. Channel function sequence: - Set enabled outputs on capture event and copy CNTR.VAL to C0CC.VAL. - Disable channel. Primary use scenario is to select this function before starting the timer. Follow these steps to select this function while CTL.MODE is different from DIS: - Set CCACT to SET_ON_CAPT with no output enable. - Configure INPUT (optional). - Wait for three timer clock periods as defined in PRECFG before setting CCACT to SET_ON_CAPT_DIS. Output enable is optional. These steps prevent capture events caused by expired signal values in edge-detection circuit. 2h = Clear on zero, toggle on compare, and then disable channel. Channel function sequence: - Clear enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C0CC.VAL = CNTR.VAL. - Disable channel. Enabled outputs are set when C0CC.VAL = 0 and CNTR.VAL = 0. 3h = Set on zero, toggle on compare, and then disable channel. Channel function sequence: - Set enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C0CC.VAL = CNTR.VAL. - Disable channel. Enabled outputs are cleared when C0CC.VAL = 0 and CNTR.VAL = 0. 4h = Clear on compare, and then disable channel. Channel function sequence: - Clear enabled outputs when C0CC.VAL = CNTR.VAL. - Disable channel. 5h = Set on compare, and then disable channel. Channel function sequence: - Set enabled outputs when C0CC.VAL = CNTR.VAL. - Disable channel. 6h = Toggle on compare, and then disable channel. Channel function sequence: - Toggle enabled outputs when C0CC.VAL = CNTR.VAL. - Disable channel. 7h = Pulse on compare, and then disable channel. Channel function sequence: - Pulse enabled outputs when C0CC.VAL = CNTR.VAL. - Disable channel. The output is high for two timer clock periods. 8h = Period and pulse width measurement. Continuously capture period and pulse width of the signal selected by INPUT relative to the signal edge given by EDGE. Set enabled outputs and RIS.C0CC when C0CC.VAL contains signal period and PC0CC.VAL contains signal pulse width. Notes: - Make sure to configure INPUT and CCACT when CTL.MODE equals DIS, then set CTL.MODE to UP_ONCE or UP_PER. - The counter restarts in the selected timer mode when C0CC.VAL contains the signal period. - If more than one channel uses this function, the channels will perform this function one at a time. The channel with lowest number has priority and performs the function first. Next measurement starts

Table 10-118. C0CFG Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
				when current measurement completes successfully or times out. A timeout occurs when counter equals target. - To observe a timeout event the RIS.TGT interrupt can be used, or another channel can be configured to SET_ON_CMP with compare value equal TGT. Signal property requirements: - Signal Period >= 2 * (1 + PRECFG.TICKDIV) * timer clock period. - Signal Period <= MAX(CNTR) * (1 + PRECFG.TICKDIV) * timer clock period. - Signal low and high phase >= (1 + PRECFG.TICKDIV) * timer clock period. 9h = Set on capture repeatedly. Channel function sequence: - Set enabled outputs on capture event and copy CNTR.VAL to C0CC.VAL. Ah = Clear on zero, toggle on compare repeatedly. Channel function sequence: - Clear enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C0CC.VAL = CNTR.VAL. Set CTL.MODE to UPDWN_PER for center-aligned PWM generation. Duty cycle is given by: When C0CC.VAL <= TGT.VAL: Duty cycle = 1 - (C0CC.VAL / TGT.VAL). When C0CC.VAL > TGT.VAL: Duty cycle = 0. Enabled outputs are set when C0CC.VAL = 0 and CNTR.VAL = 0. Bh = Set on zero, toggle on compare repeatedly. Channel function sequence: - Set enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C0CC.VAL = CNTR.VAL. Set CTL.MODE to UP_PER for edge-aligned PWM generation. Duty cycle is given by: When C0CC.VAL <= TGT.VAL: Duty cycle = C0CC.VAL / (TGT.VAL + 1). When C0CC.VAL > TGT.VAL: Duty cycle = 1. Enabled outputs are cleared when C0CC.VAL = 0 and CNTR.VAL = 0. Ch = Clear on compare repeatedly. Channel function sequence: - Clear enabled outputs when C0CC.VAL = CNTR.VAL. Dh = Set on compare repeatedly. Channel function sequence: - Set enabled outputs when C0CC.VAL = CNTR.VAL. Eh = Toggle on compare repeatedly. Channel function sequence: - Toggle enabled outputs when C0CC.VAL = CNTR.VAL. Fh = Pulse on compare repeatedly. Channel function sequence: - Pulse enabled outputs when C0CC.VAL = CNTR.VAL. The output is high for two timer clock periods.

10.7.24 C1CFG Register (Offset = C4h) [Reset = 00000000h]

C1CFG is shown in Table 10-119.

Return to the Summary Table.

Channel 1 Configuration
This register configures channel function and enables outputs.
Each channel has an edge-detection circuit. The the edge-detection circuit is:
- enabled while CCACT selects a capture function and CTL.MODE is different from DIS.
- flushed while CCACT selects a capture function and CTL.MODE is changed from DIS to another mode.
The flush action uses two system clock periods. It prevents capture events caused by expired signal values stored in the edge-detection circuit.
The channel input signal enters the edge-detection circuit. False capture events can occur when:
- the edge-detection circuit contains expired signal samples and the circuit is enabled without flush as described above.
- the CCACT field is reconfigured while CTL.MODE is different from DIS.
Primary use scenario is to select CCACT before starting the timer. Follow these steps to configure CCACT to a capture action while CTL.MODE is different from DIS:
- Set EDGE to NONE.
- Configure CCACT.
- Wait for three system clock periods before setting EDGE different from NONE.
These steps prevent capture events caused by expired signal values in edge-detection circuit.

Table 10-119. C1CFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	OUT2	R/W	0h	Output 2 enable. When 0 < CCACT < 8, OUT2 becomes zero after a capture or compare event. 0h = Channel 1 does not control output 2. 1h = Channel 1 controls output 2.
9	OUT1	R/W	0h	Output 1 enable. When 0 < CCACT < 8, OUT1 becomes zero after a capture or compare event. 0h = Channel 1 does not control output 1. 1h = Channel 1 controls output 1.
8	OUT0	R/W	0h	Output 0 enable. When 0 < CCACT < 8, OUT0 becomes zero after a capture or compare event. 0h = Channel 1 does not control output 0. 1h = Channel 1 controls output 0.
7	RESERVED	R	0h	Reserved
6	INPUT	R/W	0h	Select channel input. 0h = Event fabric 1h = IO controller
5-4	EDGE	R/W	0h	Determines the edge that triggers the channel input event. This happens post filter. 0h = Input is turned off. 1h = Input event is triggered at rising edge. 2h = Input event is triggered at falling edge. 3h = Input event is triggered at both edges.

Table 10-119. C1CFG Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
3-0	CCACT	R/W	0h	Capture-Compare action. Capture-Compare action defines 15 different channel functions that utilize capture, compare, and zero events. In every compare event the timer looks at the current value of CNTR. The corresponding output event will be set 1 timer period after CNTR = C1CC. 0h = Disable channel. 1h = Set on capture, and then disable channel. Channel function sequence: - Set enabled outputs on capture event and copy CNTR.VAL to C1CC.VAL. - Disable channel. Primary use scenario is to select this function before starting the timer. Follow these steps to select this function while CTL.MODE is different from DIS: - Set CCACT to SET_ON_CAPT with no output enable. - Configure INPUT (optional). - Wait for three timer clock periods as defined in PRECFG before setting CCACT to SET_ON_CAPT_DIS. Output enable is optional. These steps prevent capture events caused by expired signal values in edge-detection circuit. 2h = Clear on zero, toggle on compare, and then disable channel. Channel function sequence: - Clear enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C1CC.VAL = CNTR.VAL. - Disable channel. Enabled outputs are set when C1CC.VAL = 0 and CNTR.VAL = 0. 3h = Set on zero, toggle on compare, and then disable channel. Channel function sequence: - Set enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C1CC.VAL = CNTR.VAL. - Disable channel. Enabled outputs are cleared when C1CC.VAL = 0 and CNTR.VAL = 0. 4h = Clear on compare, and then disable channel. Channel function sequence: - Clear enabled outputs when C1CC.VAL = CNTR.VAL. - Disable channel. 5h = Set on compare, and then disable channel. Channel function sequence: - Set enabled outputs when C1CC.VAL = CNTR.VAL. - Disable channel. 6h = Toggle on compare, and then disable channel. Channel function sequence: - Toggle enabled outputs when C1CC.VAL = CNTR.VAL. - Disable channel. 7h = Pulse on compare, and then disable channel. Channel function sequence: - Pulse enabled outputs when C1CC.VAL = CNTR.VAL. - Disable channel. The output is high for two timer clock periods. 8h = Period and pulse width measurement. Continuously capture period and pulse width of the signal selected by INPUT relative to the signal edge given by EDGE. Set enabled outputs and RIS.C1CC when C1CC.VAL contains signal period and PC1CC.VAL contains signal pulse width. Notes: - Make sure to configure INPUT and CCACT when CTL.MODE equals DIS, then set CTL.MODE to UP_ONCE or UP_PER. - The counter restarts in the selected timer mode when C1CC.VAL contains the signal period. - If more than one channel uses this function, the channels will perform this function one at a time. The channel with lowest number has priority and performs the function first. Next measurement starts

Table 10-119. C1CFG Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
				when current measurement completes successfully or times out. A timeout occurs when counter equals target. - To observe a timeout event the RIS.TGT interrupt can be used, or another channel can be configured to SET_ON_CMP with compare value equal TGT. Signal property requirements: - Signal Period >= 2 * (1 + PRECFG.TICKDIV) * timer clock period. - Signal Period <= MAX(CNTR) * (1 + PRECFG.TICKDIV) * timer clock period. - Signal low and high phase >= (1 + PRECFG.TICKDIV) * timer clock period. 9h = Set on capture repeatedly. Channel function sequence: - Set enabled outputs on capture event and copy CNTR.VAL to C1CC.VAL. Ah = Clear on zero, toggle on compare repeatedly. Channel function sequence: - Clear enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C1CC.VAL = CNTR.VAL. Set CTL.MODE to UPDWN_PER for center-aligned PWM generation. Duty cycle is given by: When C1CC.VAL <= TGT.VAL: Duty cycle = 1 - (C1CC.VAL / TGT.VAL). When C1CC.VAL > TGT.VAL: Duty cycle = 0. Enabled outputs are set when C1CC.VAL = 0 and CNTR.VAL = 0. Bh = Set on zero, toggle on compare repeatedly. Channel function sequence: - Set enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C1CC.VAL = CNTR.VAL. Set CTL.MODE to UP_PER for edge-aligned PWM generation. Duty cycle is given by: When C1CC.VAL <= TGT.VAL: Duty cycle = C1CC.VAL / (TGT.VAL + 1). When C1CC.VAL > TGT.VAL: Duty cycle = 1. Enabled outputs are cleared when C1CC.VAL = 0 and CNTR.VAL = 0. Ch = Clear on compare repeatedly. Channel function sequence: - Clear enabled outputs when C1CC.VAL = CNTR.VAL. Dh = Set on compare repeatedly. Channel function sequence: - Set enabled outputs when C1CC.VAL = CNTR.VAL. Eh = Toggle on compare repeatedly. Channel function sequence: - Toggle enabled outputs when C1CC.VAL = CNTR.VAL. Fh = Pulse on compare repeatedly. Channel function sequence: - Pulse enabled outputs when C1CC.VAL = CNTR.VAL. The output is high for two timer clock periods.

10.7.25 C2CFG Register (Offset = C8h) [Reset = 00000000h]

C2CFG is shown in Table 10-120.

Return to the Summary Table.

Channel 2 Configuration
This register configures channel function and enables outputs.
Each channel has an edge-detection circuit. The the edge-detection circuit is:
- enabled while CCACT selects a capture function and CTL.MODE is different from DIS.
- flushed while CCACT selects a capture function and CTL.MODE is changed from DIS to another mode.
The flush action uses two system clock periods. It prevents capture events caused by expired signal values stored in the edge-detection circuit.
The channel input signal enters the edge-detection circuit. False capture events can occur when:
- the edge-detection circuit contains expired signal samples and the circuit is enabled without flush as described above.
- the CCACT field is reconfigured while CTL.MODE is different from DIS.
Primary use scenario is to select CCACT before starting the timer. Follow these steps to configure CCACT to a capture action while CTL.MODE is different from DIS:
- Set EDGE to NONE.
- Configure CCACT.
- Wait for three system clock periods before setting EDGE different from NONE.
These steps prevent capture events caused by expired signal values in edge-detection circuit.

Table 10-120. C2CFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	OUT2	R/W	0h	Output 2 enable. When 0 < CCACT < 8, OUT2 becomes zero after a capture or compare event. 0h = Channel 2 does not control output 2. 1h = Channel 2 controls output 2.
9	OUT1	R/W	0h	Output 1 enable. When 0 < CCACT < 8, OUT1 becomes zero after a capture or compare event. 0h = Channel 2 does not control output 1. 1h = Channel 2 controls output 1.
8	OUT0	R/W	0h	Output 0 enable. When 0 < CCACT < 8, OUT0 becomes zero after a capture or compare event. 0h = Channel 2 does not control output 0. 1h = Channel 2 controls output 0.
7	RESERVED	R	0h	Reserved
6	INPUT	R/W	0h	Select channel input. 0h = Event fabric 1h = IO controller
5-4	EDGE	R/W	0h	Determines the edge that triggers the channel input event. This happens post filter. 0h = Input is turned off. 1h = Input event is triggered at rising edge. 2h = Input event is triggered at falling edge. 3h = Input event is triggered at both edges.

Table 10-120. C2CFG Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
3-0	CCACT	R/W	0h	Capture-Compare action. Capture-Compare action defines 15 different channel functions that utilize capture, compare, and zero events. In every compare event the timer looks at the current value of CNTR. The corresponding output event will be set 1 timer period after CNTR = C2CC. 0h = Disable channel. 1h = Set on capture, and then disable channel. Channel function sequence: - Set enabled outputs on capture event and copy CNTR.VAL to C2CC.VAL. - Disable channel. Primary use scenario is to select this function before starting the timer. Follow these steps to select this function while CTL.MODE is different from DIS: - Set CCACT to SET_ON_CAPT with no output enable. - Configure INPUT (optional). - Wait for three timer clock periods as defined in PRECFG before setting CCACT to SET_ON_CAPT_DIS. Output enable is optional. These steps prevent capture events caused by expired signal values in edge-detection circuit. 2h = Clear on zero, toggle on compare, and then disable channel. Channel function sequence: - Clear enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C2CC.VAL = CNTR.VAL. - Disable channel. Enabled outputs are set when C2CC.VAL = 0 and CNTR.VAL = 0. 3h = Set on zero, toggle on compare, and then disable channel. Channel function sequence: - Set enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C2CC.VAL = CNTR.VAL. - Disable channel. Enabled outputs are cleared when C2CC.VAL = 0 and CNTR.VAL = 0. 4h = Clear on compare, and then disable channel. Channel function sequence: - Clear enabled outputs when C2CC.VAL = CNTR.VAL. - Disable channel. 5h = Set on compare, and then disable channel. Channel function sequence: - Set enabled outputs when C2CC.VAL = CNTR.VAL. - Disable channel. 6h = Toggle on compare, and then disable channel. Channel function sequence: - Toggle enabled outputs when C2CC.VAL = CNTR.VAL. - Disable channel. 7h = Pulse on compare, and then disable channel. Channel function sequence: - Pulse enabled outputs when C2CC.VAL = CNTR.VAL. - Disable channel. The output is high for two timer clock periods. 8h = Period and pulse width measurement. Continuously capture period and pulse width of the signal selected by INPUT relative to the signal edge given by EDGE. Set enabled outputs and RIS.C2CC when C2CC.VAL contains signal period and PC2CC.VAL contains signal pulse width. Notes: - Make sure to configure INPUT and CCACT when CTL.MODE equals DIS, then set CTL.MODE to UP_ONCE or UP_PER. - The counter restarts in the selected timer mode when C2CC.VAL contains the signal period. - If more than one channel uses this function, the channels will perform this function one at a time. The channel with lowest number has priority and performs the function first. Next measurement starts

Table 10-120. C2CFG Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
				when current measurement completes successfully or times out. A timeout occurs when counter equals target. - To observe a timeout event the RIS.TGT interrupt can be used, or another channel can be configured to SET_ON_CMP with compare value equal TGT. Signal property requirements: - Signal Period >= 2 * (1 + PRECFG.TICKDIV) * timer clock period. - Signal Period <= MAX(CNTR) * (1 + PRECFG.TICKDIV) * timer clock period. - Signal low and high phase >= (1 + PRECFG.TICKDIV) * timer clock period. 9h = Set on capture repeatedly. Channel function sequence: - Set enabled outputs on capture event and copy CNTR.VAL to C2CC.VAL. Ah = Clear on zero, toggle on compare repeatedly. Channel function sequence: - Clear enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C2CC.VAL = CNTR.VAL. Set CTL.MODE to UPDWN_PER for center-aligned PWM generation. Duty cycle is given by: When C2CC.VAL <= TGT.VAL: Duty cycle = 1 - (C2CC.VAL / TGT.VAL). When C2CC.VAL > TGT.VAL: Duty cycle = 0. Enabled outputs are set when C2CC.VAL = 0 and CNTR.VAL = 0. Bh = Set on zero, toggle on compare repeatedly. Channel function sequence: - Set enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C2CC.VAL = CNTR.VAL. Set CTL.MODE to UP_PER for edge-aligned PWM generation. Duty cycle is given by: When C2CC.VAL <= TGT.VAL: Duty cycle = C2CC.VAL / (TGT.VAL + 1). When C2CC.VAL > TGT.VAL: Duty cycle = 1. Enabled outputs are cleared when C2CC.VAL = 0 and CNTR.VAL = 0. Ch = Clear on compare repeatedly. Channel function sequence: - Clear enabled outputs when C2CC.VAL = CNTR.VAL. Dh = Set on compare repeatedly. Channel function sequence: - Set enabled outputs when C2CC.VAL = CNTR.VAL. Eh = Toggle on compare repeatedly. Channel function sequence: - Toggle enabled outputs when C2CC.VAL = CNTR.VAL. Fh = Pulse on compare repeatedly. Channel function sequence: - Pulse enabled outputs when C2CC.VAL = CNTR.VAL. The output is high for two timer clock periods.

10.7.26 PTGT Register (Offset = FCh) [Reset = 00000000h]

PTGT is shown in Table 10-121.

Return to the Summary Table.

Pipeline Target
A read or write to this register will clear the RIS.ZERO and RIS.TGT interrupt.
If CTL.MODE != QDEC.
Target value for next counter period.
The timer will copy PTGT.VAL to TGT.VAL on the upcoming CNTR zero crossing only if PTGT.VAL has been written. The copy does not happen when restarting the timer.
This is useful to avoid period jitter in PWM applications with time-varying period, sometimes referenced as phase corrected PWM.
If CTL.MODE = QDEC
The CNTR value is updated with VALUE on IDX if the counter is counting down. If the counter is counting up, CNTR is loaded with zero on IDX.
In this mode the VALUE is not loaded into TGT on zero crossing.

Table 10-121. PTGT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	The pipleline target value.

10.7.27 PC0CC Register (Offset = 100h) [Reset = 00000000h]

PC0CC is shown in Table 10-122.

Return to the Summary Table.

Pipeline Channel 0 Capture Compare

Table 10-122. PC0CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Pipeline Capture Compare value. User defined pipeline compare value or channel-updated capture value. A read or write to this register will clear the RIS.C0CC interrupt. Compare mode: An update of VAL will be transferred to C0CC.VAL when the next CNTR.VAL is zero and CTL.MODE is different from DIS. This is useful for PWM generation and prevents jitter on the edges of the generated signal. Capture mode: When C0CFG.CCACT equals PER_PULSE_WIDTH_MEAS then VAL contains the width of the low or high phase of the selected signal. This is specified by C0CFG.EDGE.

10.7.28 PC1CC Register (Offset = 104h) [Reset = 00000000h]

PC1CC is shown in Table 10-123.

Return to the Summary Table.

Pipeline Channel 1 Capture Compare

Table 10-123. PC1CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Pipeline Capture Compare value. User defined pipeline compare value or channel-updated capture value. A read or write to this register will clear the RIS.C1CC interrupt. Compare mode: An update of VAL will be transferred to C1CC.VAL when the next CNTR.VAL is zero and CTL.MODE is different from DIS. This is useful for PWM generation and prevents jitter on the edges of the generated signal. Capture mode: When C1CFG.CCACT equals PER_PULSE_WIDTH_MEAS then VAL contains the width of the low or high phase of the selected signal. This is specified by C1CFG.EDGE.

10.7.29 PC2CC Register (Offset = 108h) [Reset = 00000000h]

PC2CC is shown in Table 10-124.

Return to the Summary Table.

Pipeline Channel 2 Capture Compare

Table 10-124. PC2CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Pipeline Capture Compare value. User defined pipeline compare value or channel-updated capture value. A read or write to this register will clear the RIS.C2CC interrupt. Compare mode: An update of VAL will be transferred to C2CC.VAL when the next CNTR.VAL is zero and CTL.MODE is different from DIS. This is useful for PWM generation and prevents jitter on the edges of the generated signal. Capture mode: When C2CFG.CCACT equals PER_PULSE_WIDTH_MEAS then VAL contains the width of the low or high phase of the selected signal. This is specified by C2CFG.EDGE.

10.7.30 TGT Register (Offset = 13Ch) [Reset = 0000FFFFh]

TGT is shown in Table 10-125.

Return to the Summary Table.

Target
User defined counter target.
A read or write to this register will clear the RIS.ZERO and RIS.TGT interrupt.

Table 10-125. TGT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	FFFFh	User defined counter target value.

10.7.31 C0CC Register (Offset = 140h) [Reset = 00000000h]

C0CC is shown in Table 10-126.

Return to the Summary Table.

Channel 0 Capture Compare

Table 10-126. C0CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Capture Compare value. User defined compare value or channel-updated capture value. A read or write to this register will clear the RIS.C0CC interrupt. Compare mode: VAL is compared against CNTR.VAL and an event is generated as specified by C0CFG.CCACT when these are equal. Capture mode: The current counter value is stored in VAL when a capture event occurs. C0CFG.CCACT determines if VAL is a signal period or a regular capture value.

10.7.32 C1CC Register (Offset = 144h) [Reset = 00000000h]

C1CC is shown in Table 10-127.

Return to the Summary Table.

Channel 1 Capture Compare

Table 10-127. C1CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Capture Compare value. User defined compare value or channel-updated capture value. A read or write to this register will clear the RIS.C1CC interrupt. Compare mode: VAL is compared against CNTR.VAL and an event is generated as specified by C1CFG.CCACT when these are equal. Capture mode: The current counter value is stored in VAL when a capture event occurs. C1CFG.CCACT determines if VAL is a signal period or a regular capture value.

10.7.33 C2CC Register (Offset = 148h) [Reset = 00000000h]

C2CC is shown in Table 10-128.

Return to the Summary Table.

Channel 2 Capture Compare

Table 10-128. C2CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Capture Compare value. User defined compare value or channel-updated capture value. A read or write to this register will clear the RIS.C2CC interrupt. Compare mode: VAL is compared against CNTR.VAL and an event is generated as specified by C2CFG.CCACT when these are equal. Capture mode: The current counter value is stored in VAL when a capture event occurs. C2CFG.CCACT determines if VAL is a signal period or a regular capture value.

10.7.34 PTGTNC Register (Offset = 17Ch) [Reset = 00000000h]

PTGTNC is shown in Table 10-129.

Return to the Summary Table.

Pipeline Target No Clear
Use this register to read or write to PTGT without clearing the RIS.ZERO and RIS.TGT interrupt.

Table 10-129. PTGTNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	A read or write to this register will not clear the RIS.TGT interrupt. If CTL.MODE != QDEC. Target value for next counter period. The timer copies VAL to TGT.VAL when CNTR.VAL becomes 0. The copy does not happen when restarting the timer. This is useful to avoid period jitter in PWM applications with time-varying period, sometimes referenced as phase corrected PWM. If CTL.MODE = QDEC. The CNTR.VAL is updated with VAL on IDX. VAL is not loaded into TGT.VAL when CNTR.VAL becomes 0.

10.7.35 PC0CCNC Register (Offset = 180h) [Reset = 00000000h]

PC0CCNC is shown in Table 10-130.

Return to the Summary Table.

Pipeline Channel 0 Capture Compare No Clear

Table 10-130. PC0CCNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Pipeline Capture Compare value. User defined pipeline compare value or channel-updated capture value. A read or write to this register will not clear the RIS.C0CC interrupt. Compare mode: An update of VAL will be transferred to C0CC.VAL when the next CNTR.VAL is zero and CTL.MODE is different from DIS. This is useful for PWM generation and prevents jitter on the edges of the generated signal. Capture mode: When C0CFG.CCACT equals PER_PULSE_WIDTH_MEAS then VAL contains the width of the low or high phase of the selected signal. This is specified by C0CFG.EDGE.

10.7.36 PC1CCNC Register (Offset = 184h) [Reset = 00000000h]

PC1CCNC is shown in Table 10-131.

Return to the Summary Table.

Pipeline Channel 1 Capture Compare No Clear

Table 10-131. PC1CCNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Pipeline Capture Compare value. User defined pipeline compare value or channel-updated capture value. A read or write to this register will not clear the RIS.C1CC interrupt. Compare mode: An update of VAL will be transferred to C1CC.VAL when the next CNTR.VAL is zero and CTL.MODE is different from DIS. This is useful for PWM generation and prevents jitter on the edges of the generated signal. Capture mode: When C1CFG.CCACT equals PER_PULSE_WIDTH_MEAS then VAL contains the width of the low or high phase of the selected signal. This is specified by C1CFG.EDGE.

10.7.37 PC2CCNC Register (Offset = 188h) [Reset = 00000000h]

PC2CCNC is shown in Table 10-132.

Return to the Summary Table.

Pipeline Channel 2 Capture Compare No Clear

Table 10-132. PC2CCNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Pipeline Capture Compare value. User defined pipeline compare value or channel-updated capture value. A read or write to this register will not clear the RIS.C2CC interrupt. Compare mode: An update of VAL will be transferred to C2CC.VAL when the next CNTR.VAL is zero and CTL.MODE is different from DIS. This is useful for PWM generation and prevents jitter on the edges of the generated signal. Capture mode: When C2CFG.CCACT equals PER_PULSE_WIDTH_MEAS then VAL contains the width of the low or high phase of the selected signal. This is specified by C2CFG.EDGE.

10.7.38 TGTNC Register (Offset = 1BCh) [Reset = 0000FFFFh]

TGTNC is shown in Table 10-133.

Return to the Summary Table.

Target No Clear
Use this register to read or write to TGT without clearing the RIS.ZERO and RIS.TGT interrupt.

Table 10-133. TGTNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	FFFFh	User defined counter target value.

10.7.39 C0CCNC Register (Offset = 1C0h) [Reset = 00000000h]

C0CCNC is shown in Table 10-134.

Return to the Summary Table.

Channel 0 Capture Compare No Clear

Table 10-134. C0CCNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Capture Compare value. User defined compare value or channel-updated capture value. A read or write to this register will not clear the RIS.C0CC interrupt. Compare mode: VAL is compared against CNTR.VAL and an event is generated as specified by C0CFG.CCACT when these are equal. Capture mode: The current counter value is stored in VAL when a capture event occurs. C0CFG.CCACT determines if VAL is a signal period or a regular capture value.

10.7.40 C1CCNC Register (Offset = 1C4h) [Reset = 00000000h]

C1CCNC is shown in Table 10-135.

Return to the Summary Table.

Channel 1 Capture Compare No Clear

Table 10-135. C1CCNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Capture Compare value. User defined compare value or channel-updated capture value. A read or write to this register will not clear the RIS.C1CC interrupt. Compare mode: VAL is compared against CNTR.VAL and an event is generated as specified by C1CFG.CCACT when these are equal. Capture mode: The current counter value is stored in VAL when a capture event occurs. C1CFG.CCACT determines if VAL is a signal period or a regular capture value.

10.7.41 C2CCNC Register (Offset = 1C8h) [Reset = 00000000h]

C2CCNC is shown in Table 10-136.

Return to the Summary Table.

Channel 2 Capture Compare No Clear

Table 10-136. C2CCNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	Capture Compare value. User defined compare value or channel-updated capture value. A read or write to this register will not clear the RIS.C2CC interrupt. Compare mode: VAL is compared against CNTR.VAL and an event is generated as specified by C2CFG.CCACT when these are equal. Capture mode: The current counter value is stored in VAL when a capture event occurs. C2CFG.CCACT determines if VAL is a signal period or a regular capture value.

10.8 LGPT3 Registers

Table 10-137 lists the memory-mapped registers for the LGPT3 registers. All register offset addresses not listed in Table 10-137 should be considered as reserved locations and the register contents should not be modified.

Table 10-137. LGPT3 Registers

Offset	Acronym	Register Name	Section
0h	DESC	Description Register.	Go
4h	DESCEX	Description Extended	Go
8h	STARTCFG	Start Configuration	Go
Ch	CTL	Timer Control	Go
10h	OUTCTL	Output Control	Go
14h	CNTR	Counter	Go
18h	PRECFG	Clock Prescaler Configuration	Go
1Ch	PREEVENT	Prescaler Event	Go
20h	CHFILT	Channel Input Filter	Go
3Ch	DMA	Direct Memory Accsess	Go
40h	DMARW	Direct Memory Access	Go
44h	ADCTRG	ADC Trigger	Go
48h	IOCTL	IO Controller	Go
68h	IMASK	Interrupt mask.	Go
6Ch	RIS	Raw interrupt status.	Go
70h	MIS	Masked interrupt status.	Go
74h	ISET	Interrupt set register.	Go
78h	ICLR	Interrupt clear register.	Go
7Ch	IMSET	Interrupt mask set register.	Go
80h	IMCLR	Interrupt mask clear register.	Go
84h	EMU	Debug control	Go
C0h	C0CFG	Channel 0 Configuration	Go
C4h	C1CFG	Channel 1 Configuration	Go
C8h	C2CFG	Channel 2 Configuration	Go
FCh	PTGT	Pipeline Target	Go
100h	PC0CC	Pipeline Channel 0 Capture Compare	Go
104h	PC1CC	Pipeline Channel 1 Capture Compare	Go
108h	PC2CC	Pipeline Channel 2 Capture Compare	Go
13Ch	TGT	Target	Go
140h	C0CC	Channel 0 Capture Compare	Go
144h	C1CC	Channel 1 Capture Compare	Go
148h	C2CC	Channel 2 Capture Compare	Go
17Ch	PTGTNC	Pipeline Target No Clear	Go
180h	PC0CCNC	Pipeline Channel 0 Capture Compare No Clear	Go
184h	PC1CCNC	Pipeline Channel 1 Capture Compare No Clear	Go
188h	PC2CCNC	Pipeline Channel 2 Capture Compare No Clear	Go
1BCh	TGTNC	Target No Clear	Go
1C0h	C0CCNC	Channel 0 Capture Compare No Clear	Go
1C4h	C1CCNC	Channel 1 Capture Compare No Clear	Go
1C8h	C2CCNC	Channel 2 Capture Compare No Clear	Go

Complex bit access types are encoded to fit into small table cells. Table 10-138 shows the codes that are used for access types in this section.

Table 10-138. LGPT3 Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

10.8.1 DESC Register (Offset = 0h) [Reset = DE491010h]

DESC is shown in Table 10-139.

Return to the Summary Table.

Description Register.
This register provides IP module ID, revision information, instance index and standard MMR registers offset.

Table 10-139. DESC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	MODID	R	DE49h	Module identifier used to uniquely identify this IP.
15-12	STDIPOFF	R	1h	Standard IP MMR block offset. Standard IP MMRs are the set of from aggregated IRQ registers till DTB. 0: Standard IP MMRs do not exist 0x1-0xF: Standard IP MMRs begin at offset of (64*STDIPOFF from the base IP address)
11-8	INSTIDX	R	0h	IP Instance ID number. If multiple instances of IP exist in the device, this field can identify the instance number.
7-4	MAJREV	R	1h	Major revision of IP.
3-0	MINREV	R	0h	Minor revision of IP.

10.8.2 DESCEX Register (Offset = 4h) [Reset = 000218D3h]

DESCEX is shown in Table 10-140.

Return to the Summary Table.

Description Extended
This register describes the parameters of the LGPT.

Table 10-140. DESCEX Register Field Descriptions

Bit	Field	Type	Reset	Description
31-20	RESERVED	R	0h	Reserved
19	HIR	R	0h	Has IR logic.
18	HDBF	R	0h	Has deadband, fault, and park logic.
17-14	PREW	R	8h	Prescale width. The prescaler can maximum be configured to 2^{PREW}-1.
13	HQDEC	R	0h	Has Quadrature Decoder.
12	HCIF	R	1h	Has channel input filter.
11-8	CIFS	R	8h	Channel input filter size. The prevailing state filter can maximum be configured to 2^{CIFS}-1.
7	HDMA	R	1h	Has uDMA output and logic.
6	HINT	R	1h	Has interrupt output and logic.
5-4	CNTRW	R	1h	Counter bit-width. The maximum counter value is equal to 2^{CNTRW}-1. 0h = 16-bit counter. 1h = 24-bit counter. 2h = 32-bit counter. 3h = RESERVED
3-0	NCH	R	3h	Number of channels.

10.8.3 STARTCFG Register (Offset = 8h) [Reset = 00000000h]

STARTCFG is shown in Table 10-141.

Return to the Summary Table.

Start Configuration
This register is only for when CTL.MODE is configured to one of the SYNC modes.
This register defines when this LGPT starts.

Table 10-141. STARTCFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1-0	LGPT0	R/W	0h	LGPT start

10.8.4 CTL Register (Offset = Ch) [Reset = 00000000h]

CTL is shown in Table 10-142.

Return to the Summary Table.

Timer Control

Table 10-142. CTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2RST	W	0h	Channel 2 reset. 0h = No effect. 1h = Reset C2CC, PC2CC, and C2CFG.
9	C1RST	W	0h	Channel 1 reset. 0h = No effect. 1h = Reset C1CC, PC1CC, and C1CFG.
8	C0RST	W	0h	Channel 0 reset. 0h = No effect. 1h = Reset C0CC, PC0CC, and C0CFG.
7-6	RESERVED	R	0h	Reserved
5	INTP	R/W	0h	Interrupt Phase. This bit field controls when the RIS.TGT and RIS.ZERO interrupts are set. 0h = RIS.TGT and RIS.ZERO are set one system clock cycle after CNTR = TARGET/ZERO. 1h = RIS.TGT and RIS.ZERO are set one timer clock cycle after CNTR = TARGET/ZERO.
4-3	CMPDIR	R/W	0h	Compare direction. This bit field controls the direction the counter must have in order to set the RIS.CnCC channel interrupts. This bitfield is only relevant if [CnCFG.CCACT] is configured to a compare action. 0h = Compare RIS fields are set on up count and down count. 1h = Compare RIS fields are only set on up count. 2h = Compare RIS fields are only set on down count. 3h = RESERVED

Table 10-142. CTL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
2-0	MODE	R/W	0h	Timer mode control The CNTR restarts from 0 when MODE is written to UP_ONCE, UP_PER, UPDWN_PER, QDEC, SYNC_UP_ONCE, SYNC_UP_PER or SYNC_UPDWN_PER. When writing MODE all internally queued updates to the channels and TGT is cleared. When configuring the timer, MODE should be the last thing to configure. If changing timer configuration after MODE has been set is necessary, instructions, if any, given in the configuration registers should be followed. See for example C0CFG. 0h = Disable timer. Updates to counter, channels, and events stop. 1h = Count up once. The timer increments from 0 to target value, then stops and sets MODE to DIS. 2h = Count up periodically. The timer increments from 0 to target value, repeatedly. Period = (target value + 1) * timer clock period 3h = Count up and down periodically. The timer counts from 0 to target value and back to 0, repeatedly. Period = (target value * 2) * timer clock period 4h = The timer functions as a quadrature decoder. IOC input 0, IOC input 1 and IOC input 2 are used respectivly as PHA, PHB and IDX inputs. IDX can be turned off by setting C2CFG.EDGE = NONE. The timer clock frequency sets the sample rate of the QDEC logic. This frequency can be configured in PRECFG. 5h = Start counting up once synchronous to another LGPT, selected within STARTCFG. The timer is started by setting CTL.MODE = UP_ONCE automatically. It then functions as a normal timer in CTL.MODE = UP_ONCE, incrementing from 0 to target value, then stops and sets MODE to DIS. 6h = Start counting up periodically synchronous to another LGPT, selected within STARTCFG. The timer is started by setting CTL.MODE = UP_PER automatically. It then operates as a normal timer in CTL.MODE = UP_PER, incrementing from 0 to target value, repeatedly. Period = (target value * 2) * timer clock period 7h = Start counting up and down periodically synchronous to another LGPT, selected within STARTCFG. The timer is started by setting CTL.MODE = UPDWN_PER automatically. It then operates as a normal timer in CTL.MODE = UPDWN_PER, counting from 0 to target value and back to 0, repeatedly. Period = (target value * 2) * timer clock period

10.8.5 OUTCTL Register (Offset = 10h) [Reset = 00000000h]

OUTCTL is shown in Table 10-143.

Return to the Summary Table.

Output Control
Set and clear individual outputs manually. Manual update of an output takes priority over automatic channel updates to the same output. It is not possible to set and clear an output at the same time, such requests will be neglected.
An output can be automatically cleared, set, toggled, or pulsed by each channel, listed in decreasing order of priority. The action with highest priority happens when multiple channels want to update an output at the same time.
All outputs are connected to the event fabric and the IO controller. The outputs going to the IO controller have an aditional complementary output, this output is the inverted IO output. Both the IO and the IO complementary outputs are passed through an IO Controller, see IOCTL.

Table 10-143. OUTCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Reserved
5	SETOUT2	W	0h	Set output 2. Write 1 to set output 2.
4	CLROUT2	W	0h	Clear output 2. Write 1 to clear output 2.
3	SETOUT1	W	0h	Set output 1. Write 1 to set output 1.
2	CLROUT1	W	0h	Clear output 1. Write 1 to clear output 1.
1	SETOUT0	W	0h	Set output 0. Write 1 to set output 0.
0	CLROUT0	W	0h	Clear output 0. Write 1 to clear output 0.

10.8.6 CNTR Register (Offset = 14h) [Reset = 00000000h]

CNTR is shown in Table 10-144.

Return to the Summary Table.

Counter
The counter of this timer. After CTL.MODE is set the counter updates at the rate specified in PRECFG.

Table 10-144. CNTR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	RESERVED	R	0h	Reserved
23-0	VAL	R/W	0h	Current counter value. If CTL.MODE = QDEC this can be used to set the initial counter value during QDEC. Writing to CNTR in other modes than QDEC is possible, but may result in unpredictable behavior.

10.8.7 PRECFG Register (Offset = 18h) [Reset = 00000000h]

PRECFG is shown in Table 10-145.

Return to the Summary Table.

Clock Prescaler Configuration
This register is used to set the timer clock period. The prescaler is a counter which counts down from the value TICKDIV. When the prescaler counter reaches zero, CNTR is updated. The field TICKDIV effectively divides the prescaler tick source. The timer clock frequency can be calculated as TICKSRC/(TICKDIV+1).

Table 10-145. PRECFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-8	TICKDIV	R/W	0h	Tick division. TICKDIV determines the timer clock frequency for the counter, and timer output updates. The timer clock frequency is the clock selected by TICKSRC divided by (TICKDIV + 1). This inverse is the timer clock period. 0x00: Divide by 1. 0x01: Divide by 2. ... 0xFF: Divide by 256.
7-2	RESERVED	R	0h	Reserved
1-0	TICKSRC	R/W	0h	Prescaler tick source. TICKSRC determines the source which decrements the prescaler. 0h = Prescaler is updated at the system clock. 1h = Prescaler is updated at the rising edge of TICKEN. 2h = Prescaler is updated at the falling edge of TICKEN. 3h = Prescaler is updated at both edges of TICKEN.

10.8.8 PREEVENT Register (Offset = 1Ch) [Reset = 00000000h]

PREEVENT is shown in Table 10-146.

Return to the Summary Table.

Prescaler Event
This register is used to output a logic high signal before the zero crossing of the prescaler counter. The output is routed to the IOC.

Table 10-146. PREEVENT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	VAL	R/W	0h	Sets the HIGH time of the prescaler event output. Event goes high when the prescaler counter equals VAL. Event goes low when prescaler counter is 0. Note: - Can be used to precharge or turn an external component on for a short time before sampling, like in QDEC. - If there is a requirement to create such events that have very short periods compared to timer clock period, use two timers. One timer acts as prescaler and event generator for another timer.

10.8.9 CHFILT Register (Offset = 20h) [Reset = 00000000h]

CHFILT is shown in Table 10-147.

Return to the Summary Table.

Channel Input Filter
This register is used to configure the filter on the channel inputs. The configuration is for all inputs.
The filter is enabled when a channel is in capture mode.
The input to the filter is passed to the edge detection logic if LOAD + 1 consecutive input samples are equal. The filter functions as a down counter, counting down every input sample.
If two consecutive samples are unequal, the filter counter restarts from LOAD.
If the filter counter reaches zero, the input signal is valid and passed to the edge detection logic.
The channel filter should only be configured while the CTL.MODE = DIS. Configuring the filter while the timer is running can result in unexpected behavior.

Table 10-147. CHFILT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-8	LOAD	R/W	0h	The input of the channel filter is passed to the edge detection logic after LOAD + 1 consecutive equal samples.
7-2	RESERVED	R	0h	Reserved
1-0	MODE	R/W	0h	Channel filter mode 0h = Filter is bypassed. No Filter is used. 1h = Filter is clocked by system clock. 2h = Filter is clocked by PRECFG.TICKSRC. 3h = Filter is clocked by timer clock.

10.8.10 DMA Register (Offset = 3Ch) [Reset = 00000000h]

DMA is shown in Table 10-148.

Return to the Summary Table.

Direct Memory Accsess
This register is used to enable DMA requests from the timer and set the register addresses which the DMA will access (read/write).
Choose DMA request source by setting the REQ field. The setting of the corresponding interrupt in the RIS registers also sets the DMA request.
Upon a DMA request defined by REQ an internal address pointer is set to RWADDR*4. Every access to DMARW will increment the internal pointer by 4 such that the next DMA access will be to the next register.
The internal pointer will stop after RWCNTR increments. Further access will be ignored.

Table 10-148. DMA Register Field Descriptions

Bit	Field	Type	Reset	Description
31-20	RESERVED	R	0h	Reserved
19-16	RWCNTR	R/W	0h	The read/write counter. RWCNTR+1 is the number of times the DMA can access (read/write) the DMARW register. For each DMA access to DMARW an internal counter is incremented, writing to the next address field. RWADDR + 4*RWCNTR is the final register address which can be accessed by the DMA.
15	RESERVED	R	0h	Reserved
14-8	RWADDR	R/W	0h	The base address which the DMA access when reading/writing DMARW. The base address is set by taking the 9 LSB of the physical address and divide by 4. For example, if you wanted the RWADDR to point to the PTGT register you should set RWADDR = 0x0FC/4.
7-4	RESERVED	R	0h	Reserved
3-0	REQ	R/W	0h	0h = Disabled 1h = Setting of RIS.TGT generates a DMA request. 2h = Setting of RIS.ZERO generates a DMA request. 3h = Setting of RIS.FAULT generates a DMA request. 4h = Setting of RIS.C0CC generates a DMA request. 5h = Setting of RIS.C1CC generates a DMA request. 6h = Setting of RIS.C2CC generates a DMA request. 7h = Setting of RIS.C3CC generates a DMA request. 8h = Setting of RIS.C4CC generates a DMA request. 9h = Setting of RIS.C5CC generates a DMA request. Ah = Setting of RIS.C6CC generates a DMA request. Bh = Setting of RIS.C7CC generates a DMA request. Ch = Setting of RIS.C8CC generates a DMA request. Dh = Setting of RIS.C9CC generates a DMA request. Eh = Setting of RIS.C10CC generates a DMA request. Fh = Setting of RIS.C11CC generates a DMA request.

10.8.11 DMARW Register (Offset = 40h) [Reset = 00000000h]

DMARW is shown in Table 10-149.

Return to the Summary Table.

Direct Memory Access
This register is used by the DMA to access (read/write) register inside this LGPT module.
Each access to this register will increment the internal DMA address counter. See DMA for description.

Table 10-149. DMARW Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	RESERVED	R	0h	Reserved
23-0	VAL	R/W	0h	DMA read write value. The value that is read/written from/to the registers.

10.8.12 ADCTRG Register (Offset = 44h) [Reset = 00000000h]

ADCTRG is shown in Table 10-150.

Return to the Summary Table.

ADC Trigger
This register is used to enable ADC trigger from the timer.
Choose ADC trigger source by setting the SRC field. The setting of the corresponding interrupt in the RIS registers also sets the ADC trigger.

Table 10-150. ADCTRG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3-0	SRC	R/W	0h	0h = Disabled 1h = Setting of RIS.TGT generates an ADC trigger. 2h = Setting of RIS.ZERO generates an ADC trigger. 3h = Setting of RIS.FAULT generates an ADC trigger. 4h = Setting of RIS.C0CC generates an ADC trigger. 5h = Setting of RIS.C1CC generates an ADC trigger. 6h = Setting of RIS.C2CC generates an ADC trigger. 7h = Setting of RIS.C3CC generates an ADC trigger. 8h = Setting of RIS.C4CC generates an ADC trigger. 9h = Setting of RIS.C5CC generates an ADC trigger. Ah = Setting of RIS.C6CC generates an ADC trigger. Bh = Setting of RIS.C7CC generates an ADC trigger. Ch = Setting of RIS.C8CC generates an ADC trigger. Dh = Setting of RIS.C9CC generates an ADC trigger. Eh = Setting of RIS.C10CC generates an ADC trigger. Fh = Setting of RIS.C11CC generates an ADC trigger.

10.8.13 IOCTL Register (Offset = 48h) [Reset = 00000000h]

IOCTL is shown in Table 10-151.

Return to the Summary Table.

IO Controller
This register overrides the IO outputs.

Table 10-151. IOCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-12	RESERVED	R	0h	Reserved
11-10	COUT2	R/W	0h	IO complementary output 2 control This bit field controls IO complementary output 2. 0h = Normal output. The IO complementary output is not changed. 1h = Driven low. The IO complementary output is driven low. 2h = Driven high. The IO complementary output is driven high. 3h = Inverted value. The IO complementary output is inverted.
9-8	OUT2	R/W	0h	IO output 2 control This bit field controls IO output 2. 0h = Normal output. The IO output is not changed. 1h = Driven low. The IO output is driven low. 2h = Driven high. The IO output is driven high. 3h = Inverted value. The IO output is inverted.
7-6	COUT1	R/W	0h	IO complementary output 1 control This bit field controls IO complementary output 1. 0h = Normal output. The IO complementary output is not changed. 1h = Driven low. The IO complementary output is driven low. 2h = Driven high. The IO complementary output is driven high. 3h = Inverted value. The IO complementary output is inverted.
5-4	OUT1	R/W	0h	IO output 1 control This bit field controls IO output 1. 0h = Normal output. The IO output is not changed. 1h = Driven low. The IO output is driven low. 2h = Driven high. The IO output is driven high. 3h = Inverted value. The IO output is inverted.
3-2	COUT0	R/W	0h	IO complementary output 0 control This bit field controls IO complementary output 0. 0h = Normal output. The IO complementary output is not changed. 1h = Driven low. The IO complementary output is driven low. 2h = Driven high. The IO complementary output is driven high. 3h = Inverted value. The IO complementary output is inverted.
1-0	OUT0	R/W	0h	IO output 0 control This bit field controls IO output 0. 0h = Normal output. The IO output is not changed. 1h = Driven low. The IO output is driven low. 2h = Driven high. The IO output is driven high. 3h = Inverted value. The IO output is inverted.

10.8.14 IMASK Register (Offset = 68h) [Reset = 00000000h]

IMASK is shown in Table 10-152.

Return to the Summary Table.

Interrupt mask.
This register selects interrupt sources which are allowed to pass from RIS to MIS when the corresponding bit-fields are set to 1.

Table 10-152. IMASK Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	R/W	0h	Enable RIS.C2CC interrupt. 0h = Disable 1h = Enable
9	C1CC	R/W	0h	Enable RIS.C1CC interrupt. 0h = Disable 1h = Enable
8	C0CC	R/W	0h	Enable RIS.C0CC interrupt. 0h = Disable 1h = Enable
7	RESERVED	R	0h	Reserved
6	FAULT	R/W	0h	Enable RIS.FAULT interrupt. 0h = Disable 1h = Enable
5	IDX	R/W	0h	Enable RIS.IDX interrupt. 0h = Disable 1h = Enable
4	DIRCHNG	R/W	0h	Enable RIS.DIRCHNG interrupt. 0h = Disable 1h = Enable
3	CNTRCHNG	R/W	0h	Enable RIS.CNTRCHNG interrupt. 0h = Disable 1h = Enable
2	DBLTRANS	R/W	0h	Enable RIS.DBLTRANS interrupt. 0h = Disable 1h = Enable
1	ZERO	R/W	0h	Enable RIS.ZERO interrupt. 0h = Disable 1h = Enable
0	TGT	R/W	0h	Enable RIS.TGT interrupt. 0h = Disable 1h = Enable

10.8.15 RIS Register (Offset = 6Ch) [Reset = 00000000h]

RIS is shown in Table 10-153.

Return to the Summary Table.

Raw interrupt status.
This register reflects the state of all pending interrupts, regardless of masking. This register allows the user to implement a poll scheme. A flag set in this register can be cleared by writing 1 to the corresponding ICLR register bit.

Table 10-153. RIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	R	0h	Status of the C2CC interrupt. The interrupt is set when C2CC has capture or compare event. 0h = Cleared 1h = Set
9	C1CC	R	0h	Status of the C1CC interrupt. The interrupt is set when C1CC has capture or compare event. 0h = Cleared 1h = Set
8	C0CC	R	0h	Status of the C0CC interrupt. The interrupt is set when C0CC has capture or compare event. 0h = Cleared 1h = Set
7	RESERVED	R	0h	Reserved
6	FAULT	R	0h	Status of the FAULT interrupt. The interrupt is set immediately on active fault input. 0h = Cleared 1h = Set
5	IDX	R	0h	Status of the IDX interrupt. The interrupt is set when IDX is active. 0h = Cleared 1h = Set
4	DIRCHNG	R	0h	Status of the DIRCHNG interrupt. The interrupt is set when the direction of the counter changes. 0h = Cleared 1h = Set
3	CNTRCHNG	R	0h	Status of the CNTRCHNG interrupt. The interrupt is set when the counter increments or decrements. 0h = Cleared 1h = Set
2	DBLTRANS	R	0h	Status of the DBLTRANS interrupt. The interrupt is set when a double transition has happened during QDEC mode. 0h = Cleared 1h = Set
1	ZERO	R	0h	Status of the ZERO interrupt. The interrupt is set when CNTR = 0. 0h = Cleared 1h = Set
0	TGT	R	0h	Status of the TGT interrupt. The interrupt is set when CNTR = TGT. 0h = Cleared 1h = Set

10.8.16 MIS Register (Offset = 70h) [Reset = 00000000h]

MIS is shown in Table 10-154.

Return to the Summary Table.

Masked interrupt status.
This register is simply a bitwise AND of the contents of IMASK and RIS.*] registers. A flag set in this register can be cleared by writing 1 to the corresponding ICLR register bit.

Table 10-154. MIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	R	0h	Masked status of the RIS.C2CC interrupt. 0h = Cleared 1h = Set
9	C1CC	R	0h	Masked status of the RIS.C1CC interrupt. 0h = Cleared 1h = Set
8	C0CC	R	0h	Masked status of the RIS.C0CC interrupt. 0h = Cleared 1h = Set
7	RESERVED	R	0h	Reserved
6	FAULT	R	0h	Masked status of the RIS.FAULT interrupt. 0h = Cleared 1h = Set
5	IDX	R	0h	Masked status of the RIS.IDX interrupt. 0h = Cleared 1h = Set
4	DIRCHNG	R	0h	Masked status of the RIS.DIRCHNG interrupt. 0h = Cleared 1h = Set
3	CNTRCHNG	R	0h	Masked status of the RIS.CNTRCHNG interrupt. 0h = Cleared 1h = Set
2	DBLTRANS	R	0h	Masked status of the RIS.DBLTRANS interrupt. 0h = Cleared 1h = Set
1	ZERO	R	0h	Masked status of the RIS.ZERO interrupt. 0h = Cleared 1h = Set
0	TGT	R	0h	Masked status of the RIS.TGT interrupt. 0h = Cleared 1h = Set

10.8.17 ISET Register (Offset = 74h) [Reset = 00000000h]

ISET is shown in Table 10-155.

Return to the Summary Table.

Interrupt set register.
This register can used by software for diagnostics and safety checking purposes. Writing a 1 to a bit in this register will set the event and the corresponding RIS bit also gets set. If the corresponding IMASK bit is set, then the corresponding MIS register bit also gets set.

Table 10-155. ISET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	W	0h	Set the RIS.C2CC interrupt. 0h = No effect 1h = Set
9	C1CC	W	0h	Set the RIS.C1CC interrupt. 0h = No effect 1h = Set
8	C0CC	W	0h	Set the RIS.C0CC interrupt. 0h = No effect 1h = Set
7	RESERVED	R	0h	Reserved
6	FAULT	W	0h	Set the RIS.FAULT interrupt. 0h = No effect 1h = Set
5	IDX	W	0h	Set the RIS.IDX interrupt. 0h = No effect 1h = Set
4	DIRCHNG	W	0h	Set the RIS.DIRCHNG interrupt. 0h = No effect 1h = Set
3	CNTRCHNG	W	0h	Set the RIS.CNTRCHNG interrupt. 0h = No effect 1h = Set
2	DBLTRANS	W	0h	Set the RIS.DBLTRANS interrupt. 0h = No effect 1h = Set
1	ZERO	W	0h	Set the RIS.ZERO interrupt. 0h = No effect 1h = Set
0	TGT	W	0h	Set the RIS.TGT interrupt. 0h = No effect 1h = Set

10.8.18 ICLR Register (Offset = 78h) [Reset = 00000000h]

ICLR is shown in Table 10-156.

Return to the Summary Table.

Interrupt clear register.
This register allows software to clear interrupts. Writing a 1 to a bit in this register will clear the event and the corresponding RIS bit also gets cleared. If the corresponding IMASK bit is set, then the corresponding MIS register bit also gets cleared.

Table 10-156. ICLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	W	0h	Clear the RIS.C2CC interrupt. 0h = No effect 1h = Clear
9	C1CC	W	0h	Clear the RIS.C1CC interrupt. 0h = No effect 1h = Clear
8	C0CC	W	0h	Clear the RIS.C0CC interrupt. 0h = No effect 1h = Clear
7	RESERVED	R	0h	Reserved
6	FAULT	W	0h	Clear the RIS.FAULT interrupt. 0h = No effect 1h = Clear
5	IDX	W	0h	Clear the RIS.IDX interrupt. 0h = No effect 1h = Clear
4	DIRCHNG	W	0h	Clear the RIS.DIRCHNG interrupt. 0h = No effect 1h = Clear
3	CNTRCHNG	W	0h	Clear the RIS.CNTRCHNG interrupt. 0h = No effect 1h = Clear
2	DBLTRANS	W	0h	Clear the RIS.DBLTRANS interrupt. 0h = No effect 1h = Clear
1	ZERO	W	0h	Clear the RIS.ZERO interrupt. 0h = No effect 1h = Clear
0	TGT	W	0h	Clear the RIS.TGT interrupt. 0h = No effect 1h = Clear

10.8.19 IMSET Register (Offset = 7Ch) [Reset = 00000000h]

IMSET is shown in Table 10-157.

Return to the Summary Table.

Interrupt mask set register.
Writing a 1 to a bit in this register will set the corresponding IMASK bit.

Table 10-157. IMSET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	W	0h	Set the MIS.C2CC mask. 0h = No effect 1h = Set
9	C1CC	W	0h	Set the MIS.C1CC mask. 0h = No effect 1h = Set
8	C0CC	W	0h	Set the MIS.C0CC mask. 0h = No effect 1h = Set
7	RESERVED	R	0h	Reserved
6	FAULT	W	0h	Set the MIS.FAULT mask. 0h = No effect 1h = Set
5	IDX	W	0h	Set the MIS.IDX mask. 0h = No effect 1h = Set
4	DIRCHNG	W	0h	Set the MIS.DIRCHNG mask. 0h = No effect 1h = Set
3	CNTRCHNG	W	0h	Set the MIS.CNTRCHNG mask. 0h = No effect 1h = Set
2	DBLTRANS	W	0h	Set the MIS.DBLTRANS mask. 0h = No effect 1h = Set
1	ZERO	W	0h	Set the MIS.ZERO mask. 0h = No effect 1h = Set
0	TGT	W	0h	Set the MIS.TGT mask. 0h = No effect 1h = Set

10.8.20 IMCLR Register (Offset = 80h) [Reset = 00000000h]

IMCLR is shown in Table 10-158.

Return to the Summary Table.

Interrupt mask clear register.
Writing a 1 to a bit in this register will clear the corresponding IMASK bit.

Table 10-158. IMCLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	C2CC	W	0h	Clear the MIS.C2CC mask. 0h = No effect 1h = Clear
9	C1CC	W	0h	Clear the MIS.C1CC mask. 0h = No effect 1h = Clear
8	C0CC	W	0h	Clear the MIS.C0CC mask. 0h = No effect 1h = Clear
7	RESERVED	R	0h	Reserved
6	FAULT	W	0h	Clear the MIS.FAULT mask. 0h = No effect 1h = Clear
5	IDX	W	0h	Clear the MIS.IDX mask. 0h = No effect 1h = Clear
4	DIRCHNG	W	0h	Clear the MIS.DIRCHNG mask. 0h = No effect 1h = Clear
3	CNTRCHNG	W	0h	Clear the MIS.CNTRCHNG mask. 0h = No effect 1h = Clear
2	DBLTRANS	W	0h	Clear the MIS.DBLTRANS mask. 0h = No effect 1h = Clear
1	ZERO	W	0h	Clear the MIS.ZERO mask. 0h = No effect 1h = Clear
0	TGT	W	0h	Clear the MIS.TGT mask. 0h = No effect 1h = Clear

10.8.21 EMU Register (Offset = 84h) [Reset = 00000000h]

EMU is shown in Table 10-159.

Return to the Summary Table.

Debug control
This register can be used to freeze the timer when CPU halts when HALT is set to 1. When HALT is set to 0, or when the CPU releases debug halt, the filters and edge detection logic is flushed and the timer starts. For setting a predefined output value during a CPU debug halt, PARK, if the timer has this register, should be configured additionally. If this timer does not have the PARK register a predefined output value during CPU halt is not possible.

Table 10-159. EMU Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1	CTL	R/W	0h	Halt control. Configure when the counter shall stop upon CPU halt. This bitfield only applies if HALT = 1. 0h = Immediate reaction. The counter stops immediately on debug halt. 1h = Zero condition. The counter stops when CNTR = 0.
0	HALT	R/W	0h	Halt LGPT when CPU is halted in debug. 0h = Disable. 1h = Enable.

10.8.22 C0CFG Register (Offset = C0h) [Reset = 00000000h]

C0CFG is shown in Table 10-160.

Return to the Summary Table.

Channel 0 Configuration
This register configures channel function and enables outputs.
Each channel has an edge-detection circuit. The the edge-detection circuit is:
- enabled while CCACT selects a capture function and CTL.MODE is different from DIS.
- flushed while CCACT selects a capture function and CTL.MODE is changed from DIS to another mode.
The flush action uses two system clock periods. It prevents capture events caused by expired signal values stored in the edge-detection circuit.
The channel input signal enters the edge-detection circuit. False capture events can occur when:
- the edge-detection circuit contains expired signal samples and the circuit is enabled without flush as described above.
- the CCACT field is reconfigured while CTL.MODE is different from DIS.
Primary use scenario is to select CCACT before starting the timer. Follow these steps to configure CCACT to a capture action while CTL.MODE is different from DIS:
- Set EDGE to NONE.
- Configure CCACT.
- Wait for three system clock periods before setting EDGE different from NONE.
These steps prevent capture events caused by expired signal values in edge-detection circuit.

Table 10-160. C0CFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	OUT2	R/W	0h	Output 2 enable. When 0 < CCACT < 8, OUT2 becomes zero after a capture or compare event. 0h = Channel 0 does not control output 2. 1h = Channel 0 controls output 2.
9	OUT1	R/W	0h	Output 1 enable. When 0 < CCACT < 8, OUT1 becomes zero after a capture or compare event. 0h = Channel 0 does not control output 1. 1h = Channel 0 controls output 1.
8	OUT0	R/W	0h	Output 0 enable. When 0 < CCACT < 8, OUT0 becomes zero after a capture or compare event. 0h = Channel 0 does not control output 0. 1h = Channel 0 controls output 0.
7	RESERVED	R	0h	Reserved
6	INPUT	R/W	0h	Select channel input. 0h = Event fabric 1h = IO controller
5-4	EDGE	R/W	0h	Determines the edge that triggers the channel input event. This happens post filter. 0h = Input is turned off. 1h = Input event is triggered at rising edge. 2h = Input event is triggered at falling edge. 3h = Input event is triggered at both edges.

Table 10-160. C0CFG Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
3-0	CCACT	R/W	0h	Capture-Compare action. Capture-Compare action defines 15 different channel functions that utilize capture, compare, and zero events. In every compare event the timer looks at the current value of CNTR. The corresponding output event will be set 1 timer period after CNTR = C0CC. 0h = Disable channel. 1h = Set on capture, and then disable channel. Channel function sequence: - Set enabled outputs on capture event and copy CNTR.VAL to C0CC.VAL. - Disable channel. Primary use scenario is to select this function before starting the timer. Follow these steps to select this function while CTL.MODE is different from DIS: - Set CCACT to SET_ON_CAPT with no output enable. - Configure INPUT (optional). - Wait for three timer clock periods as defined in PRECFG before setting CCACT to SET_ON_CAPT_DIS. Output enable is optional. These steps prevent capture events caused by expired signal values in edge-detection circuit. 2h = Clear on zero, toggle on compare, and then disable channel. Channel function sequence: - Clear enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C0CC.VAL = CNTR.VAL. - Disable channel. Enabled outputs are set when C0CC.VAL = 0 and CNTR.VAL = 0. 3h = Set on zero, toggle on compare, and then disable channel. Channel function sequence: - Set enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C0CC.VAL = CNTR.VAL. - Disable channel. Enabled outputs are cleared when C0CC.VAL = 0 and CNTR.VAL = 0. 4h = Clear on compare, and then disable channel. Channel function sequence: - Clear enabled outputs when C0CC.VAL = CNTR.VAL. - Disable channel. 5h = Set on compare, and then disable channel. Channel function sequence: - Set enabled outputs when C0CC.VAL = CNTR.VAL. - Disable channel. 6h = Toggle on compare, and then disable channel. Channel function sequence: - Toggle enabled outputs when C0CC.VAL = CNTR.VAL. - Disable channel. 7h = Pulse on compare, and then disable channel. Channel function sequence: - Pulse enabled outputs when C0CC.VAL = CNTR.VAL. - Disable channel. The output is high for two timer clock periods. 8h = Period and pulse width measurement. Continuously capture period and pulse width of the signal selected by INPUT relative to the signal edge given by EDGE. Set enabled outputs and RIS.C0CC when C0CC.VAL contains signal period and PC0CC.VAL contains signal pulse width. Notes: - Make sure to configure INPUT and CCACT when CTL.MODE equals DIS, then set CTL.MODE to UP_ONCE or UP_PER. - The counter restarts in the selected timer mode when C0CC.VAL contains the signal period. - If more than one channel uses this function, the channels will perform this function one at a time. The channel with lowest number has priority and performs the function first. Next measurement starts

Table 10-160. C0CFG Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
				when current measurement completes successfully or times out. A timeout occurs when counter equals target. - To observe a timeout event the RIS.TGT interrupt can be used, or another channel can be configured to SET_ON_CMP with compare value equal TGT. Signal property requirements: - Signal Period >= 2 * (1 + PRECFG.TICKDIV) * timer clock period. - Signal Period <= MAX(CNTR) * (1 + PRECFG.TICKDIV) * timer clock period. - Signal low and high phase >= (1 + PRECFG.TICKDIV) * timer clock period. 9h = Set on capture repeatedly. Channel function sequence: - Set enabled outputs on capture event and copy CNTR.VAL to C0CC.VAL. Ah = Clear on zero, toggle on compare repeatedly. Channel function sequence: - Clear enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C0CC.VAL = CNTR.VAL. Set CTL.MODE to UPDWN_PER for center-aligned PWM generation. Duty cycle is given by: When C0CC.VAL <= TGT.VAL: Duty cycle = 1 - (C0CC.VAL / TGT.VAL). When C0CC.VAL > TGT.VAL: Duty cycle = 0. Enabled outputs are set when C0CC.VAL = 0 and CNTR.VAL = 0. Bh = Set on zero, toggle on compare repeatedly. Channel function sequence: - Set enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C0CC.VAL = CNTR.VAL. Set CTL.MODE to UP_PER for edge-aligned PWM generation. Duty cycle is given by: When C0CC.VAL <= TGT.VAL: Duty cycle = C0CC.VAL / (TGT.VAL + 1). When C0CC.VAL > TGT.VAL: Duty cycle = 1. Enabled outputs are cleared when C0CC.VAL = 0 and CNTR.VAL = 0. Ch = Clear on compare repeatedly. Channel function sequence: - Clear enabled outputs when C0CC.VAL = CNTR.VAL. Dh = Set on compare repeatedly. Channel function sequence: - Set enabled outputs when C0CC.VAL = CNTR.VAL. Eh = Toggle on compare repeatedly. Channel function sequence: - Toggle enabled outputs when C0CC.VAL = CNTR.VAL. Fh = Pulse on compare repeatedly. Channel function sequence: - Pulse enabled outputs when C0CC.VAL = CNTR.VAL. The output is high for two timer clock periods.

10.8.23 C1CFG Register (Offset = C4h) [Reset = 00000000h]

C1CFG is shown in Table 10-161.

Return to the Summary Table.

Channel 1 Configuration
This register configures channel function and enables outputs.
Each channel has an edge-detection circuit. The the edge-detection circuit is:
- enabled while CCACT selects a capture function and CTL.MODE is different from DIS.
- flushed while CCACT selects a capture function and CTL.MODE is changed from DIS to another mode.
The flush action uses two system clock periods. It prevents capture events caused by expired signal values stored in the edge-detection circuit.
The channel input signal enters the edge-detection circuit. False capture events can occur when:
- the edge-detection circuit contains expired signal samples and the circuit is enabled without flush as described above.
- the CCACT field is reconfigured while CTL.MODE is different from DIS.
Primary use scenario is to select CCACT before starting the timer. Follow these steps to configure CCACT to a capture action while CTL.MODE is different from DIS:
- Set EDGE to NONE.
- Configure CCACT.
- Wait for three system clock periods before setting EDGE different from NONE.
These steps prevent capture events caused by expired signal values in edge-detection circuit.

Table 10-161. C1CFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	OUT2	R/W	0h	Output 2 enable. When 0 < CCACT < 8, OUT2 becomes zero after a capture or compare event. 0h = Channel 1 does not control output 2. 1h = Channel 1 controls output 2.
9	OUT1	R/W	0h	Output 1 enable. When 0 < CCACT < 8, OUT1 becomes zero after a capture or compare event. 0h = Channel 1 does not control output 1. 1h = Channel 1 controls output 1.
8	OUT0	R/W	0h	Output 0 enable. When 0 < CCACT < 8, OUT0 becomes zero after a capture or compare event. 0h = Channel 1 does not control output 0. 1h = Channel 1 controls output 0.
7	RESERVED	R	0h	Reserved
6	INPUT	R/W	0h	Select channel input. 0h = Event fabric 1h = IO controller
5-4	EDGE	R/W	0h	Determines the edge that triggers the channel input event. This happens post filter. 0h = Input is turned off. 1h = Input event is triggered at rising edge. 2h = Input event is triggered at falling edge. 3h = Input event is triggered at both edges.

Table 10-161. C1CFG Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
3-0	CCACT	R/W	0h	Capture-Compare action. Capture-Compare action defines 15 different channel functions that utilize capture, compare, and zero events. In every compare event the timer looks at the current value of CNTR. The corresponding output event will be set 1 timer period after CNTR = C1CC. 0h = Disable channel. 1h = Set on capture, and then disable channel. Channel function sequence: - Set enabled outputs on capture event and copy CNTR.VAL to C1CC.VAL. - Disable channel. Primary use scenario is to select this function before starting the timer. Follow these steps to select this function while CTL.MODE is different from DIS: - Set CCACT to SET_ON_CAPT with no output enable. - Configure INPUT (optional). - Wait for three timer clock periods as defined in PRECFG before setting CCACT to SET_ON_CAPT_DIS. Output enable is optional. These steps prevent capture events caused by expired signal values in edge-detection circuit. 2h = Clear on zero, toggle on compare, and then disable channel. Channel function sequence: - Clear enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C1CC.VAL = CNTR.VAL. - Disable channel. Enabled outputs are set when C1CC.VAL = 0 and CNTR.VAL = 0. 3h = Set on zero, toggle on compare, and then disable channel. Channel function sequence: - Set enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C1CC.VAL = CNTR.VAL. - Disable channel. Enabled outputs are cleared when C1CC.VAL = 0 and CNTR.VAL = 0. 4h = Clear on compare, and then disable channel. Channel function sequence: - Clear enabled outputs when C1CC.VAL = CNTR.VAL. - Disable channel. 5h = Set on compare, and then disable channel. Channel function sequence: - Set enabled outputs when C1CC.VAL = CNTR.VAL. - Disable channel. 6h = Toggle on compare, and then disable channel. Channel function sequence: - Toggle enabled outputs when C1CC.VAL = CNTR.VAL. - Disable channel. 7h = Pulse on compare, and then disable channel. Channel function sequence: - Pulse enabled outputs when C1CC.VAL = CNTR.VAL. - Disable channel. The output is high for two timer clock periods. 8h = Period and pulse width measurement. Continuously capture period and pulse width of the signal selected by INPUT relative to the signal edge given by EDGE. Set enabled outputs and RIS.C1CC when C1CC.VAL contains signal period and PC1CC.VAL contains signal pulse width. Notes: - Make sure to configure INPUT and CCACT when CTL.MODE equals DIS, then set CTL.MODE to UP_ONCE or UP_PER. - The counter restarts in the selected timer mode when C1CC.VAL contains the signal period. - If more than one channel uses this function, the channels will perform this function one at a time. The channel with lowest number has priority and performs the function first. Next measurement starts

Table 10-161. C1CFG Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
				when current measurement completes successfully or times out. A timeout occurs when counter equals target. - To observe a timeout event the RIS.TGT interrupt can be used, or another channel can be configured to SET_ON_CMP with compare value equal TGT. Signal property requirements: - Signal Period >= 2 * (1 + PRECFG.TICKDIV) * timer clock period. - Signal Period <= MAX(CNTR) * (1 + PRECFG.TICKDIV) * timer clock period. - Signal low and high phase >= (1 + PRECFG.TICKDIV) * timer clock period. 9h = Set on capture repeatedly. Channel function sequence: - Set enabled outputs on capture event and copy CNTR.VAL to C1CC.VAL. Ah = Clear on zero, toggle on compare repeatedly. Channel function sequence: - Clear enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C1CC.VAL = CNTR.VAL. Set CTL.MODE to UPDWN_PER for center-aligned PWM generation. Duty cycle is given by: When C1CC.VAL <= TGT.VAL: Duty cycle = 1 - (C1CC.VAL / TGT.VAL). When C1CC.VAL > TGT.VAL: Duty cycle = 0. Enabled outputs are set when C1CC.VAL = 0 and CNTR.VAL = 0. Bh = Set on zero, toggle on compare repeatedly. Channel function sequence: - Set enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C1CC.VAL = CNTR.VAL. Set CTL.MODE to UP_PER for edge-aligned PWM generation. Duty cycle is given by: When C1CC.VAL <= TGT.VAL: Duty cycle = C1CC.VAL / (TGT.VAL + 1). When C1CC.VAL > TGT.VAL: Duty cycle = 1. Enabled outputs are cleared when C1CC.VAL = 0 and CNTR.VAL = 0. Ch = Clear on compare repeatedly. Channel function sequence: - Clear enabled outputs when C1CC.VAL = CNTR.VAL. Dh = Set on compare repeatedly. Channel function sequence: - Set enabled outputs when C1CC.VAL = CNTR.VAL. Eh = Toggle on compare repeatedly. Channel function sequence: - Toggle enabled outputs when C1CC.VAL = CNTR.VAL. Fh = Pulse on compare repeatedly. Channel function sequence: - Pulse enabled outputs when C1CC.VAL = CNTR.VAL. The output is high for two timer clock periods.

10.8.24 C2CFG Register (Offset = C8h) [Reset = 00000000h]

C2CFG is shown in Table 10-162.

Return to the Summary Table.

Channel 2 Configuration
This register configures channel function and enables outputs.
Each channel has an edge-detection circuit. The the edge-detection circuit is:
- enabled while CCACT selects a capture function and CTL.MODE is different from DIS.
- flushed while CCACT selects a capture function and CTL.MODE is changed from DIS to another mode.
The flush action uses two system clock periods. It prevents capture events caused by expired signal values stored in the edge-detection circuit.
The channel input signal enters the edge-detection circuit. False capture events can occur when:
- the edge-detection circuit contains expired signal samples and the circuit is enabled without flush as described above.
- the CCACT field is reconfigured while CTL.MODE is different from DIS.
Primary use scenario is to select CCACT before starting the timer. Follow these steps to configure CCACT to a capture action while CTL.MODE is different from DIS:
- Set EDGE to NONE.
- Configure CCACT.
- Wait for three system clock periods before setting EDGE different from NONE.
These steps prevent capture events caused by expired signal values in edge-detection circuit.

Table 10-162. C2CFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10	OUT2	R/W	0h	Output 2 enable. When 0 < CCACT < 8, OUT2 becomes zero after a capture or compare event. 0h = Channel 2 does not control output 2. 1h = Channel 2 controls output 2.
9	OUT1	R/W	0h	Output 1 enable. When 0 < CCACT < 8, OUT1 becomes zero after a capture or compare event. 0h = Channel 2 does not control output 1. 1h = Channel 2 controls output 1.
8	OUT0	R/W	0h	Output 0 enable. When 0 < CCACT < 8, OUT0 becomes zero after a capture or compare event. 0h = Channel 2 does not control output 0. 1h = Channel 2 controls output 0.
7	RESERVED	R	0h	Reserved
6	INPUT	R/W	0h	Select channel input. 0h = Event fabric 1h = IO controller
5-4	EDGE	R/W	0h	Determines the edge that triggers the channel input event. This happens post filter. 0h = Input is turned off. 1h = Input event is triggered at rising edge. 2h = Input event is triggered at falling edge. 3h = Input event is triggered at both edges.

Table 10-162. C2CFG Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
3-0	CCACT	R/W	0h	Capture-Compare action. Capture-Compare action defines 15 different channel functions that utilize capture, compare, and zero events. In every compare event the timer looks at the current value of CNTR. The corresponding output event will be set 1 timer period after CNTR = C2CC. 0h = Disable channel. 1h = Set on capture, and then disable channel. Channel function sequence: - Set enabled outputs on capture event and copy CNTR.VAL to C2CC.VAL. - Disable channel. Primary use scenario is to select this function before starting the timer. Follow these steps to select this function while CTL.MODE is different from DIS: - Set CCACT to SET_ON_CAPT with no output enable. - Configure INPUT (optional). - Wait for three timer clock periods as defined in PRECFG before setting CCACT to SET_ON_CAPT_DIS. Output enable is optional. These steps prevent capture events caused by expired signal values in edge-detection circuit. 2h = Clear on zero, toggle on compare, and then disable channel. Channel function sequence: - Clear enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C2CC.VAL = CNTR.VAL. - Disable channel. Enabled outputs are set when C2CC.VAL = 0 and CNTR.VAL = 0. 3h = Set on zero, toggle on compare, and then disable channel. Channel function sequence: - Set enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C2CC.VAL = CNTR.VAL. - Disable channel. Enabled outputs are cleared when C2CC.VAL = 0 and CNTR.VAL = 0. 4h = Clear on compare, and then disable channel. Channel function sequence: - Clear enabled outputs when C2CC.VAL = CNTR.VAL. - Disable channel. 5h = Set on compare, and then disable channel. Channel function sequence: - Set enabled outputs when C2CC.VAL = CNTR.VAL. - Disable channel. 6h = Toggle on compare, and then disable channel. Channel function sequence: - Toggle enabled outputs when C2CC.VAL = CNTR.VAL. - Disable channel. 7h = Pulse on compare, and then disable channel. Channel function sequence: - Pulse enabled outputs when C2CC.VAL = CNTR.VAL. - Disable channel. The output is high for two timer clock periods. 8h = Period and pulse width measurement. Continuously capture period and pulse width of the signal selected by INPUT relative to the signal edge given by EDGE. Set enabled outputs and RIS.C2CC when C2CC.VAL contains signal period and PC2CC.VAL contains signal pulse width. Notes: - Make sure to configure INPUT and CCACT when CTL.MODE equals DIS, then set CTL.MODE to UP_ONCE or UP_PER. - The counter restarts in the selected timer mode when C2CC.VAL contains the signal period. - If more than one channel uses this function, the channels will perform this function one at a time. The channel with lowest number has priority and performs the function first. Next measurement starts

Table 10-162. C2CFG Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
				when current measurement completes successfully or times out. A timeout occurs when counter equals target. - To observe a timeout event the RIS.TGT interrupt can be used, or another channel can be configured to SET_ON_CMP with compare value equal TGT. Signal property requirements: - Signal Period >= 2 * (1 + PRECFG.TICKDIV) * timer clock period. - Signal Period <= MAX(CNTR) * (1 + PRECFG.TICKDIV) * timer clock period. - Signal low and high phase >= (1 + PRECFG.TICKDIV) * timer clock period. 9h = Set on capture repeatedly. Channel function sequence: - Set enabled outputs on capture event and copy CNTR.VAL to C2CC.VAL. Ah = Clear on zero, toggle on compare repeatedly. Channel function sequence: - Clear enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C2CC.VAL = CNTR.VAL. Set CTL.MODE to UPDWN_PER for center-aligned PWM generation. Duty cycle is given by: When C2CC.VAL <= TGT.VAL: Duty cycle = 1 - (C2CC.VAL / TGT.VAL). When C2CC.VAL > TGT.VAL: Duty cycle = 0. Enabled outputs are set when C2CC.VAL = 0 and CNTR.VAL = 0. Bh = Set on zero, toggle on compare repeatedly. Channel function sequence: - Set enabled outputs when CNTR.VAL = 0. - Toggle enabled outputs when C2CC.VAL = CNTR.VAL. Set CTL.MODE to UP_PER for edge-aligned PWM generation. Duty cycle is given by: When C2CC.VAL <= TGT.VAL: Duty cycle = C2CC.VAL / (TGT.VAL + 1). When C2CC.VAL > TGT.VAL: Duty cycle = 1. Enabled outputs are cleared when C2CC.VAL = 0 and CNTR.VAL = 0. Ch = Clear on compare repeatedly. Channel function sequence: - Clear enabled outputs when C2CC.VAL = CNTR.VAL. Dh = Set on compare repeatedly. Channel function sequence: - Set enabled outputs when C2CC.VAL = CNTR.VAL. Eh = Toggle on compare repeatedly. Channel function sequence: - Toggle enabled outputs when C2CC.VAL = CNTR.VAL. Fh = Pulse on compare repeatedly. Channel function sequence: - Pulse enabled outputs when C2CC.VAL = CNTR.VAL. The output is high for two timer clock periods.

10.8.25 PTGT Register (Offset = FCh) [Reset = 00000000h]

PTGT is shown in Table 10-163.

Return to the Summary Table.

Pipeline Target
A read or write to this register will clear the RIS.ZERO and RIS.TGT interrupt.
If CTL.MODE != QDEC.
Target value for next counter period.
The timer will copy PTGT.VAL to TGT.VAL on the upcoming CNTR zero crossing only if PTGT.VAL has been written. The copy does not happen when restarting the timer.
This is useful to avoid period jitter in PWM applications with time-varying period, sometimes referenced as phase corrected PWM.
If CTL.MODE = QDEC
The CNTR value is updated with VALUE on IDX if the counter is counting down. If the counter is counting up, CNTR is loaded with zero on IDX.
In this mode the VALUE is not loaded into TGT on zero crossing.

Table 10-163. PTGT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	RESERVED	R	0h	Reserved
23-0	VAL	R/W	0h	The pipleline target value.

10.8.26 PC0CC Register (Offset = 100h) [Reset = 00000000h]

PC0CC is shown in Table 10-164.

Return to the Summary Table.

Pipeline Channel 0 Capture Compare

Table 10-164. PC0CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	RESERVED	R	0h	Reserved
23-0	VAL	R/W	0h	Pipeline Capture Compare value. User defined pipeline compare value or channel-updated capture value. A read or write to this register will clear the RIS.C0CC interrupt. Compare mode: An update of VAL will be transferred to C0CC.VAL when the next CNTR.VAL is zero and CTL.MODE is different from DIS. This is useful for PWM generation and prevents jitter on the edges of the generated signal. Capture mode: When C0CFG.CCACT equals PER_PULSE_WIDTH_MEAS then VAL contains the width of the low or high phase of the selected signal. This is specified by C0CFG.EDGE.

10.8.27 PC1CC Register (Offset = 104h) [Reset = 00000000h]

PC1CC is shown in Table 10-165.

Return to the Summary Table.

Pipeline Channel 1 Capture Compare

Table 10-165. PC1CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	RESERVED	R	0h	Reserved
23-0	VAL	R/W	0h	Pipeline Capture Compare value. User defined pipeline compare value or channel-updated capture value. A read or write to this register will clear the RIS.C1CC interrupt. Compare mode: An update of VAL will be transferred to C1CC.VAL when the next CNTR.VAL is zero and CTL.MODE is different from DIS. This is useful for PWM generation and prevents jitter on the edges of the generated signal. Capture mode: When C1CFG.CCACT equals PER_PULSE_WIDTH_MEAS then VAL contains the width of the low or high phase of the selected signal. This is specified by C1CFG.EDGE.

10.8.28 PC2CC Register (Offset = 108h) [Reset = 00000000h]

PC2CC is shown in Table 10-166.

Return to the Summary Table.

Pipeline Channel 2 Capture Compare

Table 10-166. PC2CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	RESERVED	R	0h	Reserved
23-0	VAL	R/W	0h	Pipeline Capture Compare value. User defined pipeline compare value or channel-updated capture value. A read or write to this register will clear the RIS.C2CC interrupt. Compare mode: An update of VAL will be transferred to C2CC.VAL when the next CNTR.VAL is zero and CTL.MODE is different from DIS. This is useful for PWM generation and prevents jitter on the edges of the generated signal. Capture mode: When C2CFG.CCACT equals PER_PULSE_WIDTH_MEAS then VAL contains the width of the low or high phase of the selected signal. This is specified by C2CFG.EDGE.

10.8.29 TGT Register (Offset = 13Ch) [Reset = 00FFFFFFh]

TGT is shown in Table 10-167.

Return to the Summary Table.

Target
User defined counter target.
A read or write to this register will clear the RIS.ZERO and RIS.TGT interrupt.

Table 10-167. TGT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	RESERVED	R	0h	Reserved
23-0	VAL	R/W	00FFFFFFh	User defined counter target value.

10.8.30 C0CC Register (Offset = 140h) [Reset = 00000000h]

C0CC is shown in Table 10-168.

Return to the Summary Table.

Channel 0 Capture Compare

Table 10-168. C0CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	RESERVED	R	0h	Reserved
23-0	VAL	R/W	0h	Capture Compare value. User defined compare value or channel-updated capture value. A read or write to this register will clear the RIS.C0CC interrupt. Compare mode: VAL is compared against CNTR.VAL and an event is generated as specified by C0CFG.CCACT when these are equal. Capture mode: The current counter value is stored in VAL when a capture event occurs. C0CFG.CCACT determines if VAL is a signal period or a regular capture value.

10.8.31 C1CC Register (Offset = 144h) [Reset = 00000000h]

C1CC is shown in Table 10-169.

Return to the Summary Table.

Channel 1 Capture Compare

Table 10-169. C1CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	RESERVED	R	0h	Reserved
23-0	VAL	R/W	0h	Capture Compare value. User defined compare value or channel-updated capture value. A read or write to this register will clear the RIS.C1CC interrupt. Compare mode: VAL is compared against CNTR.VAL and an event is generated as specified by C1CFG.CCACT when these are equal. Capture mode: The current counter value is stored in VAL when a capture event occurs. C1CFG.CCACT determines if VAL is a signal period or a regular capture value.

10.8.32 C2CC Register (Offset = 148h) [Reset = 00000000h]

C2CC is shown in Table 10-170.

Return to the Summary Table.

Channel 2 Capture Compare

Table 10-170. C2CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	RESERVED	R	0h	Reserved
23-0	VAL	R/W	0h	Capture Compare value. User defined compare value or channel-updated capture value. A read or write to this register will clear the RIS.C2CC interrupt. Compare mode: VAL is compared against CNTR.VAL and an event is generated as specified by C2CFG.CCACT when these are equal. Capture mode: The current counter value is stored in VAL when a capture event occurs. C2CFG.CCACT determines if VAL is a signal period or a regular capture value.

10.8.33 PTGTNC Register (Offset = 17Ch) [Reset = 00000000h]

PTGTNC is shown in Table 10-171.

Return to the Summary Table.

Pipeline Target No Clear
Use this register to read or write to PTGT without clearing the RIS.ZERO and RIS.TGT interrupt.

Table 10-171. PTGTNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	RESERVED	R	0h	Reserved
23-0	VAL	R/W	0h	A read or write to this register will not clear the RIS.TGT interrupt. If CTL.MODE != QDEC. Target value for next counter period. The timer copies VAL to TGT.VAL when CNTR.VAL becomes 0. The copy does not happen when restarting the timer. This is useful to avoid period jitter in PWM applications with time-varying period, sometimes referenced as phase corrected PWM. If CTL.MODE = QDEC. The CNTR.VAL is updated with VAL on IDX. VAL is not loaded into TGT.VAL when CNTR.VAL becomes 0.

10.8.34 PC0CCNC Register (Offset = 180h) [Reset = 00000000h]

PC0CCNC is shown in Table 10-172.

Return to the Summary Table.

Pipeline Channel 0 Capture Compare No Clear

Table 10-172. PC0CCNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	RESERVED	R	0h	Reserved
23-0	VAL	R/W	0h	Pipeline Capture Compare value. User defined pipeline compare value or channel-updated capture value. A read or write to this register will not clear the RIS.C0CC interrupt. Compare mode: An update of VAL will be transferred to C0CC.VAL when the next CNTR.VAL is zero and CTL.MODE is different from DIS. This is useful for PWM generation and prevents jitter on the edges of the generated signal. Capture mode: When C0CFG.CCACT equals PER_PULSE_WIDTH_MEAS then VAL contains the width of the low or high phase of the selected signal. This is specified by C0CFG.EDGE.

10.8.35 PC1CCNC Register (Offset = 184h) [Reset = 00000000h]

PC1CCNC is shown in Table 10-173.

Return to the Summary Table.

Pipeline Channel 1 Capture Compare No Clear

Table 10-173. PC1CCNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	RESERVED	R	0h	Reserved
23-0	VAL	R/W	0h	Pipeline Capture Compare value. User defined pipeline compare value or channel-updated capture value. A read or write to this register will not clear the RIS.C1CC interrupt. Compare mode: An update of VAL will be transferred to C1CC.VAL when the next CNTR.VAL is zero and CTL.MODE is different from DIS. This is useful for PWM generation and prevents jitter on the edges of the generated signal. Capture mode: When C1CFG.CCACT equals PER_PULSE_WIDTH_MEAS then VAL contains the width of the low or high phase of the selected signal. This is specified by C1CFG.EDGE.

10.8.36 PC2CCNC Register (Offset = 188h) [Reset = 00000000h]

PC2CCNC is shown in Table 10-174.

Return to the Summary Table.

Pipeline Channel 2 Capture Compare No Clear

Table 10-174. PC2CCNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	RESERVED	R	0h	Reserved
23-0	VAL	R/W	0h	Pipeline Capture Compare value. User defined pipeline compare value or channel-updated capture value. A read or write to this register will not clear the RIS.C2CC interrupt. Compare mode: An update of VAL will be transferred to C2CC.VAL when the next CNTR.VAL is zero and CTL.MODE is different from DIS. This is useful for PWM generation and prevents jitter on the edges of the generated signal. Capture mode: When C2CFG.CCACT equals PER_PULSE_WIDTH_MEAS then VAL contains the width of the low or high phase of the selected signal. This is specified by C2CFG.EDGE.

10.8.37 TGTNC Register (Offset = 1BCh) [Reset = 00FFFFFFh]

TGTNC is shown in Table 10-175.

Return to the Summary Table.

Target No Clear
Use this register to read or write to TGT without clearing the RIS.ZERO and RIS.TGT interrupt.

Table 10-175. TGTNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	RESERVED	R	0h	Reserved
23-0	VAL	R/W	00FFFFFFh	User defined counter target value.

10.8.38 C0CCNC Register (Offset = 1C0h) [Reset = 00000000h]

C0CCNC is shown in Table 10-176.

Return to the Summary Table.

Channel 0 Capture Compare No Clear

Table 10-176. C0CCNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	RESERVED	R	0h	Reserved
23-0	VAL	R/W	0h	Capture Compare value. User defined compare value or channel-updated capture value. A read or write to this register will not clear the RIS.C0CC interrupt. Compare mode: VAL is compared against CNTR.VAL and an event is generated as specified by C0CFG.CCACT when these are equal. Capture mode: The current counter value is stored in VAL when a capture event occurs. C0CFG.CCACT determines if VAL is a signal period or a regular capture value.

10.8.39 C1CCNC Register (Offset = 1C4h) [Reset = 00000000h]

C1CCNC is shown in Table 10-177.

Return to the Summary Table.

Channel 1 Capture Compare No Clear

Table 10-177. C1CCNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	RESERVED	R	0h	Reserved
23-0	VAL	R/W	0h	Capture Compare value. User defined compare value or channel-updated capture value. A read or write to this register will not clear the RIS.C1CC interrupt. Compare mode: VAL is compared against CNTR.VAL and an event is generated as specified by C1CFG.CCACT when these are equal. Capture mode: The current counter value is stored in VAL when a capture event occurs. C1CFG.CCACT determines if VAL is a signal period or a regular capture value.

10.8.40 C2CCNC Register (Offset = 1C8h) [Reset = 00000000h]

C2CCNC is shown in Table 10-178.

Return to the Summary Table.

Channel 2 Capture Compare No Clear

Table 10-178. C2CCNC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	RESERVED	R	0h	Reserved
23-0	VAL	R/W	0h	Capture Compare value. User defined compare value or channel-updated capture value. A read or write to this register will not clear the RIS.C2CC interrupt. Compare mode: VAL is compared against CNTR.VAL and an event is generated as specified by C2CFG.CCACT when these are equal. Capture mode: The current counter value is stored in VAL when a capture event occurs. C2CFG.CCACT determines if VAL is a signal period or a regular capture value.

Chapter 11

System Timer (SYSTIM)

This chapter discusses the features and configurations of the System Timer (SYSTIM) module.

11.1 Overview

SYSTIM is a 34-bit timer running at a resolution of 250ns with a low range (about 1.2hr) but high precision (of 250ns) that can be used by both the RF-Core and the system CPU. SYSTIM follows the RTC (Chapter 12) and can only be used in device active state. The SYSTIM synchronizes with the RTC. Synchronization is done during start-up and continuously during the active state. When the device goes from standby, reset, or shutdown to active the value of the RTC time is loaded to SYSTIM with a resolution of 250ns.

SYSTIM has five channels:

- 3 channels with capture and compare dedicated to RF-Core
  - Channels 2, 3, and 4
  - These channels work with the 250ns compare and capture value.
  - These channels are reserved for radio functions.
- 1 channel with capture and compare for the system software
  - Channel 1
  - Works with the 1µs resolution compare and capture value
  - This channel is available for custom use cases.
- 1 channel in backup with capture and compare
  - Channel 0
  - Configurable for 1µs or 250ns resolution
  - This channel is reserved for TI Software functions.

11.2 Block Diagram

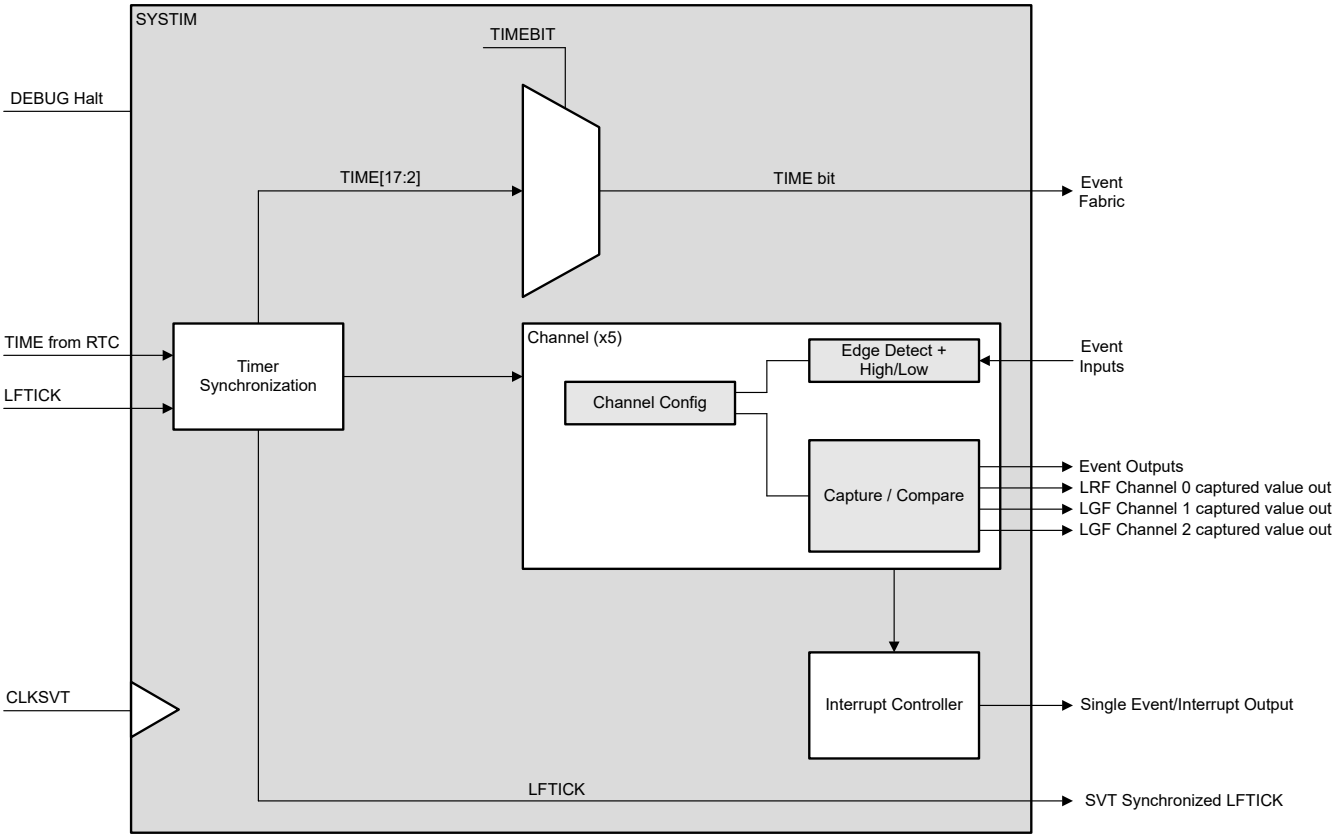


Figure 11-1. SYSTIM Block Diagram

11.3 Functional Description

11.3.1 Common Channel Features

11.3.1.1 Compare Mode

A channel is automatically armed in compare mode by writing any value to the SYSTIM.CH*n*VAL[31:0] DATA bit field.

Once programmed in Compare mode, the channel generates an event when SYSTIM time reaches the programmed value.

The SYSTIM channel generates an event immediately if the value programmed is within a certain limit in the past. This limit is 4 seconds in the past for 1µs resolution channels. The limit is 1 second in the past for 250ns resolution channel.

11.3.1.2 Capture Mode

A channel can be armed in Capture mode by setting the SYSTIM.CH*n*CFG[0] MODE bit to 1.

In Capture mode, the channel captures the 32 bits of SYSTIM time based on the event that the channel receives. The resolution of these 32 bits is based on the type of channel used. The capture can be configured on different edges of the input event. For more details, refer to the SYSTIM.CH*n*CFG register.

By default, a channel is disarmed after capture happens. The SYSTIM.CH*n*CFG[0] MODE bit is cleared to 0.

11.3.1.3 Additional Channel Arming Methods

In addition to the previously mentioned methods for arming a channel, a channel can be armed by programming the ARMSET register. When the SYSTIM.CH*n*CFG[0] MODE bit is 0 and SYSTIM.ARMSET[*n*] CH*n* bit is set for that channel, the channel is armed in compare mode and starts comparison based on the current CH*n*VAL register value.

When SYSTIM.CH*n*CFG[3] REARM is set to 1, re-arm is enabled. The channel remains in continuous capture mode. Otherwise, the channel is in one-shot capture mode. Rearm is only valid for capture mode.

A channel can be disarmed by setting the SYSTIM.ARMCLR[*n*] CH*n* bit to 1. ARMCLR disarms the channel and resets the SYSTIM.CH*n*CFG[0] MODE bit to 0.

The CHnSR register can be written to load the channel with a value without triggering the channel to enter compare mode. Then ARMSET can be used to arm the channel in Compare mode by setting the ARMSET[n], provided the CHnCFG[0] MODE is zero.

11.3.2 Interrupts and Events

- SYSTIM capture/compare event
  - SYSTIM event output:
    - One event output per channel + one combined event output for all the channels routed to the MCU event fabric.
    - Total of 5+1 event outputs. Only the combined event has a standard complement of MIS/RIS/IMASK/ ISET/ICLR/IMCLR/IMSET registers. RIS is automatically cleared when reading capture value or writing compare value.
      - The combined event also includes a timer overflow event. This event is asserted when time overflows and remains asserted till 4s.
    - The event set within RIS can be cleared through ICLR, and also when any of the following occurs:
      - Reading from the capture register (This only occurs if the channel is in Capture mode or disarmed.)
      - Writing to the compare register
      - Arming the channel in capture mode by writing the CH ***n*** CFG.MODE bit to 1
      - Arming the channel in compare mode by writing the ARMSET ***n*** bit to 1, provided CH ***n*** CFG.MODE bit as 0

    - Trigger past event
      - A compare event triggers immediately if 0 <= TIME-CMP < 2^22, in other words, if the compare time is now or up to 1.048576s in the past for the LRF channel, and up to 4.294s in the past for system and backup channels (if the backup channel is configured to 1µs resolution).
- SYSTIM event inputs:
  - One event input per channel
  - Can be configured to capture a configurable condition (rising edge, falling edge, and both edges). This also generates a capture event output on the same channel, setting the RIS interrupt flag.
  - The SYSTIM.CH*n*CFG[2:1] INP bit field can be used to configure the capture condition.
    - If SYSTIM.CH*n*CFG[2:1] INP bit field = 0, then capture on the rising edge.
    - If SYSTIM.CH*n*CFG[2:1] INP bit field = 1, then capture on the falling edge.
    - If SYSTIM.CH*n*CFG[2:1] INP bit field = 2, then capture on both rising and falling edges.
  - The software must arm a channel for capture, and a capture event automatically disarms the channel.

11.4 SYSTIM Registers

Table 11-1 lists the memory-mapped registers for the SYSTIM registers. All register offset addresses not listed in Table 11-1 should be considered as reserved locations and the register contents should not be modified.

Table 11-1. SYSTIM Registers

Offset	Acronym	Register Name	Section
0h	DESC	Module Description	Go
44h	IMASK	Interrupt mask	Go
48h	RIS	Raw interrupt status	Go
4Ch	MIS	Masked interrupt status	Go
50h	ISET	Interrupt set	Go
54h	ICLR	Interrupt clear	Go
58h	IMSET	Interrupt mask set	Go
5Ch	IMCLR	Interrupt mask clear	Go
60h	EMU	Emulation	Go
100h	TIME250N	Systime Count Value [31:0]	Go
104h	TIME1U	Systime Count Value [33:2]	Go
108h	OUT	channel's Ouput Value	Go
10Ch	CH0CFG	channel0 Configuration.	Go
110h	CH1CFG	channel1 Configuration.	Go
114h	CH2CFG	channel2 Configuration.	Go
118h	CH3CFG	channel3 Configuration.	Go
11Ch	CH4CFG	channel4 Configuration.	Go
120h	CH0CC	Channel 0 Capture/Compare Value	Go
124h	CH1CC	Channel 1 Capture/Compare Value	Go
128h	CH2CC	Channel 2 Capture/Compare Value	Go
12Ch	CH3CC	Channel 3 Capture/Compare Value	Go
130h	CH4CC	Channel 4 Capture/Compare Value	Go
134h	TIMEBIT	Systimer's Time bit	Go
140h	STATUS	Timer Status	Go
144h	ARMSET	Channel arming set	Go
148h	ARMCLR	Channel Arming clear	Go
14Ch	CH0CCSR	Channel 0 Save/Restore Value	Go
150h	CH1CCSR	Channel 1 Save/Restore Value	Go
154h	CH2CCSR	Channel 2 Save/Restore Value	Go
158h	CH3CCSR	Channel 3 Save/Restore Value	Go
15Ch	CH4CCSR	Channel 4 Save/Restore Value	Go

Complex bit access types are encoded to fit into small table cells. Table 11-2 shows the codes that are used for access types in this section.

Table 11-2. SYSTIM Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		

Table 11-2. SYSTIM Access Type Codes (continued)

Access Type	Code	Description
-n		Value after reset or the default value

11.4.1 DESC Register (Offset = 0h) [Reset = 94431010h]

DESC is shown in Table 11-3.

Return to the Summary Table.

Description.
This register identifies the peripheral and its exact version.

Table 11-3. DESC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	MODID	R	9443h	Module identifier used to uniquely identify this IP.
15-12	STDIPOFF	R	1h	Standard IP MMR block offset. Standard IP MMRs are the set of from aggregated IRQ registers till DTB. 0: Standard IP MMRs do not exist 0x1-0xF: Standard IP MMRs begin at offset of (64*STDIPOFF from the base IP address)
11-8	INSTIDX	R	0h	IP Instance ID number. If multiple instances of IP exists in SOC, this field can identify the instance number 0-15
7-4	MAJREV	R	1h	Major revision of IP 0-15
3-0	MINREV	R	0h	Minor revision of IP 0-15.

11.4.2 IMASK Register (Offset = 44h) [Reset = 00000000h]

IMASK is shown in Table 11-4.

Return to the Summary Table.

Interrupt mask.
This register selects interrupt sources which are allowed to pass from RIS to MIS when the corresponding bit-fields are set to 1.

Table 11-4. IMASK Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Reserved
5	OVFL	R/W	0h	Systimer counter overflow event interrupt mask. 0h = Disable Interrupt Mask 1h = Enable Interrrupt Mask
4	EV4	R/W	0h	Systimer channel 4 event interrupt mask. 0h = Disable Interrupt Mask 1h = Enable Interrrupt Mask
3	EV3	R/W	0h	Systimer channel 3 event interrupt mask. 0h = Disable Interrupt Mask 1h = Enable Interrrupt Mask
2	EV2	R/W	0h	Systimer channel 2 event interrupt mask. 0h = Disable Interrupt Mask 1h = Enable Interrrupt Mask
1	EV1	R/W	0h	Systimer channel 1 event interrupt mask. 0h = Disable Interrupt Mask 1h = Enable Interrrupt Mask
0	EV0	R/W	0h	Systimer channel 0 event interrupt mask. 0h = Disable Interrupt Mask 1h = Enable Interrrupt Mask

11.4.3 RIS Register (Offset = 48h) [Reset = 00000000h]

RIS is shown in Table 11-5.

Return to the Summary Table.

Raw interrupt status.
This register reflects the state of all pending interrupts, regardless of masking. This register allows the user to implement a poll scheme. A flag set in this register can be cleared by writing 1 to the corresponding ICLR register bit.

Table 11-5. RIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Reserved
5	OVFL	R	0h	Raw interrupt status for Systimer counter overflow event. This bit is set to 1 when an event is received on SysTimer Overflow occurs. 0h = Interrupt did not occur 1h = Interrupt occured
4	EV4	R	0h	Raw interrupt status for channel 4 event. This bit is set to 1 when a CAPTURE or COMPARE event is received on channel 4. 0h = Interrupt did not occur 1h = Interrupt occured
3	EV3	R	0h	Raw interrupt status for channel 3 event. This bit is set to 1 when a CAPTURE or COMPARE event is received on channel 3. 0h = Interrupt did not occur 1h = Interrupt occured
2	EV2	R	0h	Raw interrupt status for channel 2 Event. This bit is set to 1 when a CAPTURE or COMPARE event is received on channel 2. 0h = Interrupt did not occur 1h = Interrupt occured
1	EV1	R	0h	Raw interrupt status for channel 1 event. This bit is set to 1 when a CAPTURE or COMPARE event is received on channel 1. 0h = Interrupt did not occur 1h = Interrupt occured
0	EV0	R	0h	Raw interrupt status for channel 0 event. This bit is set to 1 when a CAPTURE or COMPARE event is received on channel 0. 0h = Interrupt did not occur 1h = Interrupt occured

11.4.4 MIS Register (Offset = 4Ch) [Reset = 00000000h]

MIS is shown in Table 11-6.

Return to the Summary Table.

Masked interrupt status.
This register is simply a bitwise AND of the contents of IMASK and RIS.*] registers. A flag set in this register can be cleared by writing 1 to the corresponding ICLR register bit.

Table 11-6. MIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Reserved
5	OVFL	R	0h	Mask Interrupt status for Systimer counter overflow Event in MIS register. 0h = Interrupt did not occur 1h = Interrupt occured
4	EV4	R	0h	Mask interrupt status for channel 4 event. 0h = Interrupt did not occur 1h = Interrupt occured
3	EV3	R	0h	Mask interrupt status for channel 3 event. 0h = Interrupt did not occur 1h = Interrupt occured
2	EV2	R	0h	Mask interrupt status for channel 2 event. 0h = Interrupt did not occur 1h = Interrupt occured
1	EV1	R	0h	Mask interrupt status for channel 1 event. 0h = Interrupt did not occur 1h = Interrupt occured
0	EV0	R	0h	Mask interrupt status for channel 0 event. 0h = Interrupt did not occur 1h = Interrupt occured

11.4.5 ISET Register (Offset = 50h) [Reset = 00000000h]

ISET is shown in Table 11-7.

Return to the Summary Table.

Interrupt set.
This register can used by software for diagnostics and safety checking purposes. Writing a 1 to a bit in this register will set the event and the corresponding RIS bit also gets set. If the corresponding IMASK bit is set, then the corresponding MIS register bit also gets set.

Table 11-7. ISET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Reserved
5	OVFL	W	0h	Sets Systimer counter overflow interrupt. 0h = Writing 0 has no effect 1h = Set Interrupt
4	EV4	W	0h	Sets channel 4 interrupt. 0h = Writing 0 has no effect 1h = Set Interrupt
3	EV3	W	0h	Sets channel 3 interrupt. 0h = Writing 0 has no effect 1h = Set Interrupt
2	EV2	W	0h	Sets channel 2 interrupt. 0h = Writing 0 has no effect 1h = Set Interrupt
1	EV1	W	0h	Sets channel 1 interrupt. 0h = Writing 0 has no effect 1h = Set Interrupt
0	EV0	W	0h	Sets channel 0 interrupt. 0h = Writing 0 has no effect 1h = Set Interrupt

11.4.6 ICLR Register (Offset = 54h) [Reset = 00000000h]

ICLR is shown in Table 11-8.

Return to the Summary Table.

Interrupt clear.
'This register allows software to clear interrupts. Writing a 1 to a bit in this register will clear the event and the corresponding RIS bit also gets cleared. If the corresponding IMASK bit is set, then the corresponding MIS register bit also gets cleared.

Table 11-8. ICLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Reserved
5	OVFL	W	0h	Clears Systimer counter overflow interrupt. 0h = Writing 0 has no effect 1h = Clear Interrupt
4	EV4	W	0h	Clears channel 4 interrupt. 0h = Writing 0 has no effect 1h = Clear Interrupt
3	EV3	W	0h	Clears channel 3 interrupt. 0h = Writing 0 has no effect 1h = Clear Interrupt
2	EV2	W	0h	Clears channel 2 interrupt. 0h = Writing 0 has no effect 1h = Clear Interrupt
1	EV1	W	0h	Clears channel 1 interrupt. 0h = Writing 0 has no effect 1h = Clear Interrupt
0	EV0	W	0h	Clears channel 0 interrupt. 0h = Writing 0 has no effect 1h = Clear Interrupt

11.4.7 IMSET Register (Offset = 58h) [Reset = 00000000h]

IMSET is shown in Table 11-9.

Return to the Summary Table.

Interrupt mask set.
Writing a 1 to a bit in this register will set the corresponding IMASK bit.

Table 11-9. IMSET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Reserved
5	OVFL	W	0h	Sets Timer Overflow Event Interrupt Mask. 0h = Writing 0 has no effect 1h = Set interrupt mask
4	EV4	W	0h	Sets channel4 Event Interrupt mask 0h = Writing 0 has no effect 1h = Set interrupt mask
3	EV3	W	0h	Sets channel3 Event Interrupt mask 0h = Writing 0 has no effect 1h = Set interrupt mask
2	EV2	W	0h	Sets channel2 Event Interrupt mask 0h = Writing 0 has no effect 1h = Set interrupt mask
1	EV1	W	0h	Sets channel1 Event Interrupt mask 0h = Writing 0 has no effect 1h = Set interrupt mask
0	EV0	W	0h	Sets channel0 Event Interrupt mask 0h = Writing 0 has no effect 1h = Set interrupt mask

11.4.8 IMCLR Register (Offset = 5Ch) [Reset = 00000000h]

IMCLR is shown in Table 11-10.

Return to the Summary Table.

Interrupt mask clear.
Writing a 1 to a bit in this register will clear the corresponding IMASK bit.

Table 11-10. IMCLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Reserved
5	OVFL	W	0h	Clears Timer Overflow Event Interrupt Mask. 0h = Writing 0 has no effect 1h = Clear interrupt mask
4	EV4	W	0h	Clears channel4 Event Interrupt Mask. 0h = Writing 0 has no effect 1h = Clear interrupt mask
3	EV3	W	0h	Clears channel3 Event Interrupt Mask. 0h = Writing 0 has no effect 1h = Clear interrupt mask
2	EV2	W	0h	Clears channel2 Event Interrupt Mask. 0h = Writing 0 has no effect 1h = Clear interrupt mask
1	EV1	W	0h	Clears channel1 Event Interrupt Mask. 0h = Writing 0 has no effect 1h = Clear interrupt mask
0	EV0	W	0h	Clears channel0 Event Interrupt Mask. 0h = Writing 0 has no effect 1h = Clear interrupt mask

11.4.9 EMU Register (Offset = 60h) [Reset = 00000000h]

EMU is shown in Table 11-11.

Return to the Summary Table.

Emulation control.
This register controls the behavior of the IP related to core halted input.

Table 11-11. EMU Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Reserved
0	HALT	R/W	0h	Halt control. 0h = Free run option. The IP ignores the state of the core halted input. 1h = Freeze option. The IP freezes functionality when the core halted input is asserted, and resumes when it is deasserted. The freeze can either be immediate or after the IP has reached a boundary from where it can resume without corruption.

11.4.10 TIME250N Register (Offset = 100h) [Reset = 00000000h]

TIME250N is shown in Table 11-12.

Return to the Summary Table.

Systimer Counter Value - 250ns resolution.
This 32-bit value reads out bits [31:0] of the systimer counter. The counter is 34-bit and runs on CLKSVT/12. It maintains a resolution of 250ns with a range of about 17.9m.

Table 11-12. TIME250N Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R	0h	32-bit counter value [31:0]. This will provide a 250ns resolution and a range of 17.9m.

11.4.11 TIME1U Register (Offset = 104h) [Reset = 00000000h]

TIME1U is shown in Table 11-13.

Return to the Summary Table.

Systimer Counter Value - 1µs resolution
This 32-bit value reads out bits[33:2] of the systimer counter. The counter is 34-bit and runs on CLKSVT/12. It maintains a resolution of 1us with a range of about 1 h 11m.

Table 11-13. TIME1U Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R	0h	32-bit counter value [33:2]. This will provide a resolution of 1us and a range of 1hr and 11m.

11.4.12 OUT Register (Offset = 108h) [Reset = 00000000h]

OUT is shown in Table 11-14.

Return to the Summary Table.

Systimer's channel Output Event Values

Table 11-14. OUT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-5	RESERVED	R	0h	Reserved
4	OUT4	R/W	0h	Output Value of channel 4. 0h = Event did not occur. 1h = Event occured
3	OUT3	R/W	0h	Output Value of channel 3. 0h = Event did not occur. 1h = Event occured
2	OUT2	R/W	0h	Output Value of channel 2. 0h = Event did not occur. 1h = Event occured
1	OUT1	R/W	0h	Output Value of channel 1. 0h = Event did not occur. 1h = Event occured
0	OUT0	R/W	0h	Output Value of channel 0. 0h = Event did not occur. 1h = Event occured

11.4.13 CH0CFG Register (Offset = 10Ch) [Reset = 00000000h]

CH0CFG is shown in Table 11-15.

Return to the Summary Table.

Systimer channel 0 configuration.
This channel has configurability for 250ns and 1us based capture and compare operations.

Table 11-15. CH0CFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-5	RESERVED	R	0h	Reserved
4	RES	R/W	0h	This bit decides the RESOLUTION of the channel that will be used. 0h = channel Works in Timer's 1us Resolution. 1h = channel Works in Timer's 250ns resolution
3	REARM	R/W	0h	When Rearm is enabled the channel remains in continous capture mode. Otherwise it'll be in one shot capture mode. Rearm is only valid for capture mode. 0h = Re Arm is disabled 1h = Re arm is enabled
2-1	INP	R/W	0h	Decides the channel input signal's mode. Setting the Value as 2'b11 selects the Rise Function. 0h = Capture on rising edge 1h = Capture on Falling Edge 2h = Capture on both Edge
0	MODE	R/W	0h	Decides the channel mode. 0h = channel is disabled 1h = channel is in capture mode

11.4.14 CH1CFG Register (Offset = 110h) [Reset = 00000000h]

CH1CFG is shown in Table 11-16.

Return to the Summary Table.

Systimer channel 1 configuration.
This channel works in 1us based capture and compare operations.

Table 11-16. CH1CFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3	REARM	R/W	0h	When Rearm is enabled the channel remains in continous capture mode. Otherwise it'll be in one shot capture mode. Rearm is only valid for capture mode. 0h = Re Arm is disabled 1h = Re arm is enabled
2-1	INP	R/W	0h	Decides the channel input signal's mode. Setting the Value as 2'b11 selects the Rise Function 0h = Capture on rising edge 1h = Capture on Falling Edge 2h = Capture on both Edge
0	MODE	R/W	0h	Decides the channel mode. 0h = channel is disabled 1h = channel is in capture mode

11.4.15 CH2CFG Register (Offset = 114h) [Reset = 00000000h]

CH2CFG is shown in Table 11-17.

Return to the Summary Table.

Systimer channel 2 configuration.
This channel works in 250ns based capture and compare operations.

Table 11-17. CH2CFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3	REARM	R/W	0h	When Rearm is enabled the channel remains in continous capture mode. Otherwise it'll be in one shot capture mode. Rearm is only valid for capture mode. 0h = Re Arm is disabled 1h = Re arm is enabled
2-1	INP	R/W	0h	Decides the channel input signal's mode. Setting the Value as 2'b11 selects the Rise Function 0h = Capture on rising edge 1h = Capture on Falling Edge 2h = Capture on both Edge
0	MODE	R/W	0h	Decides the channel mode. 0h = channel is disabled 1h = channel is in capture mode

11.4.16 CH3CFG Register (Offset = 118h) [Reset = 00000000h]

CH3CFG is shown in Table 11-18.

Return to the Summary Table.

Systimer channel 3 configuration.
This channel works in 250ns based capture and compare operations.

Table 11-18. CH3CFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3	REARM	R/W	0h	When Rearm is enabled the channel remains in continous capture mode. Otherwise it'll be in one shot capture mode. Rearm is only valid for capture mode. 0h = Re Arm is disabled 1h = Re arm is enabled
2-1	INP	R/W	0h	Decides the channel input signal's mode. Setting the Value as 2'b11 selects the Rise Function 0h = Capture on rising edge 1h = Capture on Falling Edge 2h = Capture on both Edge
0	MODE	R/W	0h	Decides the channel mode. 0h = channel is disabled 1h = channel is in capture mode

11.4.17 CH4CFG Register (Offset = 11Ch) [Reset = 00000000h]

CH4CFG is shown in Table 11-19.

Return to the Summary Table.

Systimer channel 4 configuration.
This channel works in 250ns based capture and compare operations.

Table 11-19. CH4CFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3	REARM	R/W	0h	When Rearm is enabled the channel remains in continous capture mode. Otherwise it'll be in one shot capture mode. Rearm is only valid for capture mode. 0h = Re Arm is disabled 1h = Re arm is enabled
2-1	INP	R/W	0h	Decides the channel input signal's mode. Setting the Value as 2'b11 selects the Rise Function 0h = Capture on rising edge 1h = Capture on Falling Edge 2h = Capture on both Edge
0	MODE	R/W	0h	Decides the channel mode. 0h = channel is disabled 1h = channel is in capture mode

11.4.18 CH0CC Register (Offset = 120h) [Reset = 00000000h]

CH0CC is shown in Table 11-20.

Return to the Summary Table.

System Timer channel 0 Capture/Compare register.
This register when written with any compare value will arm the channel to work in compare mode.

Table 11-20. CH0CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R/W	0h	Capture/compare value

11.4.19 CH1CC Register (Offset = 124h) [Reset = 00000000h]

CH1CC is shown in Table 11-21.

Return to the Summary Table.

System Timer channel 1 Capture/Compare register.
This register when written with any compare value will arm the channel to work in compare mode.

Table 11-21. CH1CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R/W	0h	Capture/compare value

11.4.20 CH2CC Register (Offset = 128h) [Reset = 00000000h]

CH2CC is shown in Table 11-22.

Return to the Summary Table.

System Timer channel 2 Capture/Compare register.
This register when written with any compare value will arm the channel to work in compare mode.

Table 11-22. CH2CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R/W	0h	Capture/compare value

11.4.21 CH3CC Register (Offset = 12Ch) [Reset = 00000000h]

CH3CC is shown in Table 11-23.

Return to the Summary Table.

System Timer channel 3 Capture/Compare register.
This register when written with any compare value will arm the channel to work in compare mode.

Table 11-23. CH3CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R/W	0h	Capture/compare value

11.4.22 CH4CC Register (Offset = 130h) [Reset = 00000000h]

CH4CC is shown in Table 11-24.

Return to the Summary Table.

System Timer channel 4 Capture/Compare register.
This register when written with any compare value will arm the channel to work in compare mode.

Table 11-24. CH4CC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R/W	0h	Capture/compare value

11.4.23 TIMEBIT Register (Offset = 134h) [Reset = 00000000h]

TIMEBIT is shown in Table 11-25.

Return to the Summary Table.

Systimer's Time bit.
This Register will be used to specify which TIME bit is required by LGPT to be forwarded from SYSTIMER.

Table 11-25. TIMEBIT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R/W	0h	The corresponding bit will have value '1' rest should be '0'. If more than one bit is asserted, output is "or" of all the bits. 0h = No bit is forwarded to the event fabric. 1h = Bit2 is forwarded to the event fabric. 2h = Bit3 is forwarded to the event fabric. 4h = Bit4 is forwarded to the event fabric. 8h = Bit5 is forwarded to the event fabric. 10h = Bit6 is forwarded to the event fabric. 20h = Bit7 is forwarded to the event fabric. 40h = Bit8 is forwarded to the event fabric. 80h = Bit9 is forwarded to the event fabric. 100h = Bit10 is forwarded to the event fabric. 200h = Bit11 is forwarded to the event fabric. 400h = Bit12 is forwarded to the event fabric. 800h = Bit13 is forwarded to the event fabric. 1000h = Bit14 is forwarded to the event fabric. 2000h = Bit15 is forwarded to the event fabric. 4000h = Bit16 is forwarded to the event fabric. 8000h = Bit17 is forwarded to the event fabric.

11.4.24 STATUS Register (Offset = 140h) [Reset = 00000010h]

STATUS is shown in Table 11-26.

Return to the Summary Table.

Systimer status.
This register can be used to read the running status of the timer and to resync the Systimer with RTC.

Table 11-26. STATUS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-5	RESERVED	R	0h	Reserved
4	SYNCUP	R/W	1h	This bit indicates sync status of Systimer with RTC. The bitfield has a reset value of '1', which gets cleared to '0' after the Systimer synchronizes with RTC on the first LFTICK edge. A write to this bit resynchronizes the Systimer with RTC on the next LFTICK edge. A read value of '1' indicates the synchronization is ongoing and a read of '0' indicates the synchronization is done.
3-1	RESERVED	R	0h	Reserved
0	VAL	R	0h	This bit indicates if the system time is initialized and running. 0h = system timer is not running. 1h = system timer is running

11.4.25 ARMSET Register (Offset = 144h) [Reset = 00000000h]

ARMSET is shown in Table 11-27.

Return to the Summary Table.

ARMSET
Reading this register gives out the status of the 5 channels.
Channel state UNARMED returns 0.
Channel state CAPTURE or COMPARE returns 1.
A write to ARMSET has for each channel the following effect -
If ARMSTA[x]==0 -> no effect
If ARMSTA[x]==1 and channel x is in CAPTURE state then no effect on the channel
Else, set channel in COMPARE mode using existing CHxVAL value

Table 11-27. ARMSET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-5	RESERVED	R	0h	Reserved
4	CH4	R/W	0h	Arming channel 4 for either compare or capture operation. 0h = No effect on the channel 1h = if channel 4 is in CAPTURE state then no effect on the channel else it can set channel in COMPARE mode using existing CH4CC.VAL value.
3	CH3	R/W	0h	Arming channel 3 for either compare or capture operation. 0h = No effect on the channel 1h = if channel 3 is in CAPTURE state then no effect on the channel else it can set channel in COMPARE mode using existing CH3CC.VAL value
2	CH2	R/W	0h	Arming channel 2 for either compare or capture operation. 0h = No effect on the channel 1h = if channel 2 is in CAPTURE state then no effect on the channel else it can set channel in COMPARE mode using existing CH2CC.VAL value
1	CH1	R/W	0h	Arming channel 1 for either compare or capture operation. 0h = No effect on the channel 1h = if channel 1 is in CAPTURE state then no effect on the channel else it can Set channel in COMPARE mode using existing CH1CC.VAL value
0	CH0	R/W	0h	Arming channel 0 for either compare or capture operation. 0h = No effect on the channel 1h = if channel 0 is in CAPTURE state then no effect on the channel else it can set channel in COMPARE mode using existing CH0CC.VAL value

11.4.26 ARMCLR Register (Offset = 148h) [Reset = 00000000h]

ARMCLR is shown in Table 11-28.

Return to the Summary Table.

ARMCLR
Read of this register gives out the status of the 5 channels .
Channel state UNARMED returns 0.
Channel state CAPTURE or COMPARE returns 1.
A write to ARMCLR has for each channel the following effect -
If ARMCLR[x]==0 -> no effect.
Else, set channel in UNARMED state without triggering event unless a compare/capture event happens in the same cycle

Table 11-28. ARMCLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-5	RESERVED	R	0h	Reserved
4	CH4	R/W	0h	Disarming channel 4 0h = No effect on the channel 1h = Set channel in UNARMED state without triggering event unless a compare/capture event happens in the same cycle
3	CH3	R/W	0h	Disarming channel 3 0h = No effect on the channel 1h = Set channel in UNARMED state without triggering event unless a compare/capture event happens in the same cycle
2	CH2	R/W	0h	Disarming channel 2 0h = No effect on the channel 1h = Set channel in UNARMED state without triggering event unless a compare/capture event happens in the same cycle
1	CH1	R/W	0h	Disarming channel 1 0h = No effect on the channel 1h = Set channel in UNARMED state without triggering event unless a compare/capture event happens in the same cycle
0	CH0	R/W	0h	Disarming channel 0 0h = No effect on the channel 1h = Set channel in UNARMED state without triggering event unless a compare/capture event happens in the same cycle

11.4.27 CH0CCSR Register (Offset = 14Ch) [Reset = 00000000h]

CH0CCSR is shown in Table 11-29.

Return to the Summary Table.

Save/restore alias register for channel 0.
A read to this register behaves exactly as a read to CH0CC.
Write to CH0CCSR sets CH0CC.VAL value of register without affecting channel state or configuration

Table 11-29. CH0CCSR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R/W	0h	Capture/compare value

11.4.28 CH1CCSR Register (Offset = 150h) [Reset = 00000000h]

CH1CCSR is shown in Table 11-30.

Return to the Summary Table.

Save/restore alias registers channel 1.
A read to CH1CCSR behaves exactly as a read to CH1VAL.
Write to this register sets CH1CC.VAL without affecting channel state or configuration.

Table 11-30. CH1CCSR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R/W	0h	Capture/compare value

11.4.29 CH2CCSR Register (Offset = 154h) [Reset = 00000000h]

CH2CCSR is shown in Table 11-31.

Return to the Summary Table.

Save/restore alias registers channel 2.
A read to CH2CCSR behaves exactly as a read to CH2CC
Write to CH2CCSR sets CH2CC.VAL value of register without affecting channel state or configuration

Table 11-31. CH2CCSR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R/W	0h	Capture/compare value

11.4.30 CH3CCSR Register (Offset = 158h) [Reset = 00000000h]

CH3CCSR is shown in Table 11-32.

Return to the Summary Table.

Save/restore alias registers channel 3.
A read to CH3CCSR behaves exactly as a read to CH3CC
Write to CH3CCSR sets CH3CC.VAL value of register without affecting channel state or configuration.

Table 11-32. CH3CCSR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R/W	0h	Capture/compare value

11.4.31 CH4CCSR Register (Offset = 15Ch) [Reset = 00000000h]

CH4CCSR is shown in Table 11-33.

Return to the Summary Table.

Save/restore alias registers channel 4.
A read to CH4CCSR behaves exactly as a read to CH4CC
Write to CH4CCSR sets CH4CC.VAL value of register without affecting channel state or configuration.

Table 11-33. CH4CCSR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R/W	0h	Capture/compare value

Chapter 12

Real Time Clock (RTC)

This chapter describes the functionality of the Real Time Clock (RTC) module.

12.1 Introduction

The RTC is a 67-bit, 2-channel timer running on the LFCLK system clock. The RTC is active in the standby and active power states. When the device enters the reset or shutdown state the RTC is reset.

The RTC accumulates time elapsed since reset on each LFCLK. The RTC counter is incremented by LFINC at a rate of 32.768kHz. LFINC indicates the period of LFCLK in µs, with an additional granularity of 16 fractional bits.

The counter can be read from two 32-bit registers. RTC.TIME8U has a range of approximately 9.5 hours with an LSB representing 8 microseconds. RTC.TIME524M has a range of approximately 71.4 years with an LSB representing 524 milliseconds.

There is hardware synchronization between the system timer (SYSTIM) and the RTC so that the multi-channel and higher resolution SYSTIM remains in synchronization with the RTC's time base.

The RTC has two channels: one compare channel and one capture channel and is capable of waking the device out of the standby power state. The RTC compare channel is typically used only by system software and only during the standby power state.

12.2 Block Diagram

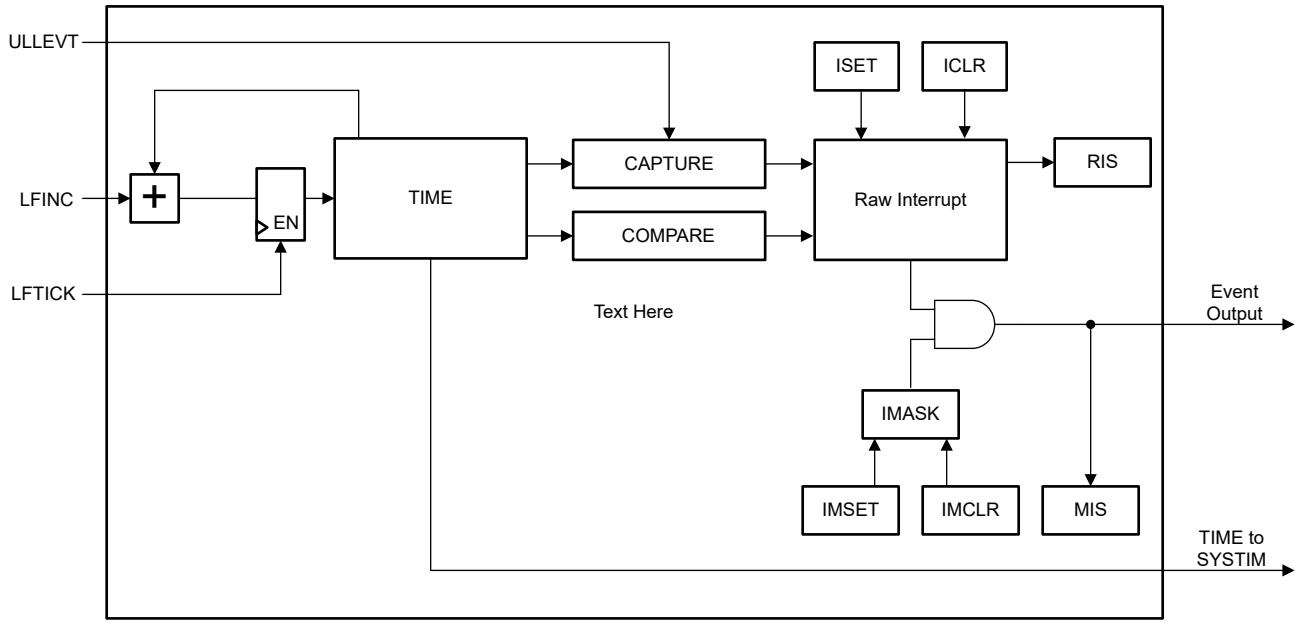


Figure 12-1. RTC Block Diagram

12.3 Interrupts and Events

12.3.1 Input Event

RTC has one capture input event from the AON/ULL event fabric. The capture event is selected by writing EVTULL.RTCCPTSEL[5:0] PUBID bit field. The capture can be on a rising or falling edge. This is configured by writing or clearing the RTC.CH1CFG[0] EDGE bit.

- RTC.CH1CFG[0] EDGE = 0 is rising edge configuration. This is the reset value for this bit.
- RTC.CH1CFG[0] EDGE = 1 is falling edge configuration.

12.3.2 Output Event

The RTC has one combined interrupt request event output. See Section 4.3 for more information on interrupt and event handling.

Interrupt flags for the combined interrupt can be read from the RTC.MIS register. Interrupts can be cleared by writing to the RTC.ICLR register. Interrupt status for the capture channel is cleared by reading the RTC.CH1CC8U[20:0] VAL bit field. Interrupt status for the compare channel is cleared by writing to the RTC.CH0CC8U[31:0] VAL bit field.

12.3.3 Arming and Disarming Channels

RTC.ARMSET and RTC.ARMCLR are provided as additional methods of arming and disarming channels. A read of either the RTC.ARMCLR or RTC.ARMSET register returns the armed status of each channel. If the capture or compare channel is armed, setting the corresponding bit in the RTC.ARMCLR register to 1 sets the channel in the unarmed state without triggering an event (unless a compare or capture event happens in the same cycle).

If the channel is not armed, writing the RTC.ARMSET[1] CH1 bit arms the capture channel. Writing to RTC.ARMSET[0] CH0 has no effect on the compare channel. The compare channel is automatically armed when a value is written to the RTC.CH0COMP register.

12.4 Capture and Compare Configuration

The capture and compare channels use the 32-bit RTC.TIME8U register with a resolution of 8µs and a range of approximately 9.5 hours.

12.4.1 Capture

The capture event is selected by writing EVTULL.RTCCPTSEL[5:0] PUBID bit field. Event capture can occur on either the rising or falling edge of the event by setting or clearing the RTC.CH1CFG[0] EDGE bit. Capture is armed by setting the RTC.ARMSET[1] CH1 bit. Once capture is armed the RTC.CH1CC8U[20:0] bit field is updated with the value from the RTC.TIME8U[31:0] VAL bit field at the time the capture event occurs.

12.4.2 Compare

The compare channel is armed when a compare value is written to the RTC.CH0COMP register. The compare channel is disarmed when a compare event occurs.

A compare event is generated based on the compare value written to the RTC.CH0COMP register.

- RTC generates an immediate event if the compare value is between now and 1 second in the past.
- Otherwise, RTC generates a compare event when the difference between the compare the value and the RTC value is within 32µs.

12.5 RTC Registers

Table 12-1 lists the memory-mapped registers for the RTC registers. All register offset addresses not listed in Table 12-1 should be considered as reserved locations and the register contents should not be modified.

Table 12-1. RTC Registers

Offset	Acronym	Register Name	Section
0h	DESC	Module Description	Go
4h	CTL	RTC control Register	Go
8h	ARMSET	Channel Arming Set	Go
Ch	ARMCLR	Channel Arming Clear	Go
18h	TIME8U	RTC Lower Time Slice	Go
1Ch	TIME524M	RTC Upper Time Slice	Go
28h	CH0CC8U	Channel0 compare value	Go
38h	CH1CC8U	Channel1 capture Value	Go
3Ch	CH1CFG	channel1 Input Configuration	Go
44h	IMASK	Interrupt mask	Go
48h	RIS	Raw interrupt status	Go
4Ch	MIS	Masked interrupt status	Go
50h	ISET	Interrupt set	Go
54h	ICLR	Interrupt clear	Go
58h	IMSET	Interrupt mask set	Go
5Ch	IMCLR	Interrupt clear	Go
60h	EMU	Emulation	Go

Complex bit access types are encoded to fit into small table cells. Table 12-2 shows the codes that are used for access types in this section.

Table 12-2. RTC Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

12.5.1 DESC Register (Offset = 0h) [Reset = 64421010h]

DESC is shown in Table 12-3.

Return to the Summary Table.

Description Register. This register provides IP module ID, revision information, instance index and standard MMR registers offset.

Table 12-3. DESC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	MODID	R/W	6442h	Module identifier used to uniquely identify this IP.
15-12	STDIPOFF	R/W	1h	Standard IP MMR block offset. Standard IP MMRs are the set of from aggregated IRQ registers till DTB. 0: Standard IP MMRs do not exist 0x1-0xF: Standard IP MMRs begin at offset of (64*STDIPOFF from the base IP address)
11-8	INSTIDX	R/W	0h	IP Instance ID number. If multiple instances of IP exist in the device, this field can identify the instance number (0-15).
7-4	MAJREV	R/W	1h	Major revision of IP (0-15).
3-0	MINREV	R/W	0h	Minor revision of IP (0-15).

12.5.2 CTL Register (Offset = 4h) [Reset = 00000000h]

CTL is shown in Table 12-4.

Return to the Summary Table.

RTC Control register. This register controls resetting the of RTC counter

Table 12-4. CTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Reserved
0	RST	W	0h	RTC counter reset. Writing 1 to this bit will reset the RTC counter, and cause it to resume counting from 0x0 0h = No effect 1h = Reset the timer.

12.5.3 ARMSET Register (Offset = 8h) [Reset = 00000000h]

ARMSET is shown in Table 12-5.

Return to the Summary Table.

RTC channel mode set register. Read to each bit field of this register provides the current channel mode.
- Read of 1'b0 indicates the channel is unarmed.
- Read of 1'b1 indicates the channel is either in capture or compare mode.
A write to each bitfield of this register the following effect:
- Write of 1'b0 has no effect on channel mode.
- Write of 1'b1 has no effect on the compare channel. While write of 1'b1 for capture channel will arm it in capture mode if it is disabled.

Table 12-5. ARMSET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1	CH1	R/W	0h	Arming Channel 1 for capture operation. 0h = No effect on the channel 1h = Enable the Channel 1 for capture operation
0	CH0	R/W	0h	No effect on arming the channel. Read will give the status of the Channel 0. 0h = No effect on the channel 1h = No effect on the compare channel

12.5.4 ARMCLR Register (Offset = Ch) [Reset = 00000000h]

ARMCLR is shown in Table 12-6.

Return to the Summary Table.

RTC channel mode clear register. Read to each bit field of this register provides the current channel mode.
- Read of 1'b0 indicates the channel is unarmed.
- Read of 1'b1 indicates the channel is either in capture or compare mode.
A write to each bitfield of this register the following effect:
- Write of 1'b0 has no effect on channel mode.
- Write of 1'b1 for capture/compare channel will disarm it without triggering event unless a compare/capture event happens in the same cycle.

Table 12-6. ARMCLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1	CH1	R/W	0h	Disarming Channel 1 0h = No effect on the channel 1h = Set channel in UNARMED state without triggering event unless a capture event happens in the same cycle
0	CH0	R/W	0h	Disarming Channel 0 0h = No effect on the channel 1h = Set channel in UNARMED state without triggering event unless a compare event happens in the same cycle

12.5.5 TIME8U Register (Offset = 18h) [Reset = 00000000h]

TIME8U is shown in Table 12-7.

Return to the Summary Table.

RTC Time value register. 32-bit unsigned integer representing [34:3] time slice of the real time clock counter. The counter runs on LFCLK. This field has a resolution of 8us, and range of about 9.5 hours.

Table 12-7. TIME8U Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R	0h	Unsigned integer representing [34:3]slice of real time counter.

12.5.6 TIME524M Register (Offset = 1Ch) [Reset = 00000000h]

TIME524M is shown in Table 12-8.

Return to the Summary Table.

RTC time value register. 32-bit unsigned integer representing [50:19] time slice of the real time clock counter. This field has a resolution of about 0.5s and a range of about 71.4 years.

Table 12-8. TIME524M Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R	0h	Unsigned integer representing. [50:19]slice of real time counter.

12.5.7 CH0CC8U Register (Offset = 28h) [Reset = 00000000h]

CH0CC8U is shown in Table 12-9.

Return to the Summary Table.

Channel 0 compare value. A write to this register automatically enables the channel to trigger an event when RTC timer reaches the programmed value or if the programmed value is 1 sec in the past.

Table 12-9. CH0CC8U Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R/W	0h	RTC Channel 0 compare value. This value is compared against TIME8U.VAL. A Channel 0 event is generated when TIME8U.VAL value reaches or exceeds this compare value.

12.5.8 CH1CC8U Register (Offset = 38h) [Reset = 00000000h]

CH1CC8U is shown in Table 12-10.

Return to the Summary Table.

Channel 1 capture value. This register captures the RTC time slice [34:3] on each selected edge of the capture event when the ARMSET.CH1 = 1.

Table 12-10. CH1CC8U Register Field Descriptions

Bit	Field	Type	Reset	Description
31-21	RESERVED	R	0h	Reserved
20-0	VAL	R	0h	TIME8U.VAL captured value at the last selected edge of capture event.

12.5.9 CH1CFG Register (Offset = 3Ch) [Reset = 00000000h]

CH1CFG is shown in Table 12-11.

Return to the Summary Table.

Channel 1 configuration register. This register can be used to select the capture edge for generating the capture event.

Table 12-11. CH1CFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Reserved
0	EDGE	R/W	0h	Edge detect configuration for capture source 0h = Rising Edge. 1h = Falling Edge.

12.5.10 IMASK Register (Offset = 44h) [Reset = 00000000h]

IMASK is shown in Table 12-12.

Return to the Summary Table.

Interrupt Mask. If a bit is set, then corresponding interrupt is un-masked. Un-masking the interrupt causes the raw interrupt to be visible in IIDX, as well as MIS.

Table 12-12. IMASK Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1	EV1	R/W	0h	Channel 1 Event Interrupt Mask. 0h = Clear Interrupt Mask 1h = Enable Interrrupt Mask
0	EV0	R/W	0h	Channel 0 Event Interrupt Mask. 0h = Disable Interrupt Mask 1h = Enable Interrrupt Mask

12.5.11 RIS Register (Offset = 48h) [Reset = 00000000h]

RIS is shown in Table 12-13.

Return to the Summary Table.

Interrupt mask. This register selects interrupt sources which are allowed to pass from RIS to MIS when the corresponding bit-fields are set to 1.

Table 12-13. RIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1	EV1	R	0h	Raw interrupt status for Channel 1 event. This bit is set to 1 when a capture event is received on Channel 1. This bit will be cleared when the bit in ICLR.EV1 is set to 1 or when the captured time value is read from the CH1CC8U register. 0h = Interrupt did not occur 1h = Interrupt occured
0	EV0	R	0h	Raw interrupt status for Channel 0 event. This bit is set to 1 when a compare event occurs on Channel 0. This bit will be cleared. When the corresponding bit in ICLR.EV0 is set to 1. Or when a new compare value is written in CH0CC8U register 0h = Interrupt did not occur 1h = Interrupt occured

12.5.12 MIS Register (Offset = 4Ch) [Reset = 00000000h]

MIS is shown in Table 12-14.

Return to the Summary Table.

Masked interrupt status. This register is simply a bitwise AND of the contents of IMASK and RIS.*] registers. A flag set in this register can be cleared by writing 1 to the corresponding ICLR register bit.

Table 12-14. MIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1	EV1	R	0h	Masked interrupt status for channel 1 event. 0h = Interrupt did not occur 1h = Interrupt occured
0	EV0	R	0h	Masked interrupt status for channel 0 event. 0h = Interrupt did not occur 1h = Interrupt occured

12.5.13 ISET Register (Offset = 50h) [Reset = 00000000h]

ISET is shown in Table 12-15.

Return to the Summary Table.

Interrupt set register. This register can used by software for diagnostics and safety checking purposes. Writing a 1 to a bit in this register will set the event and the corresponding RIS bit also gets set. If the corresponding IMASK bit is set, then the corresponding MIS register bit also gets set.

Table 12-15. ISET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1	EV1	W	0h	Set Channel 1 event Interrupt. 0h = Writing 0 has no effect 1h = Set interrupt
0	EV0	W	0h	Set Channel 0 event Interrupt. 0h = Writing 0 has no effect 1h = Set interrupt

12.5.14 ICLR Register (Offset = 54h) [Reset = 00000000h]

ICLR is shown in Table 12-16.

Return to the Summary Table.

Interrupt clear register. This register allows software to clear interrupts. Writing a 1 to a bit in this register will clear the event and the corresponding RIS bit also gets cleared. If the corresponding IMASK bit is set, then the corresponding MIS register bit also gets cleared.

Table 12-16. ICLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1	EV1	W	0h	Clears channel 1 event interrupt. 0h = Writing 0 has no effect 1h = Clear Interrupt
0	EV0	W	0h	Clears channel 0 event interrupt. 0h = Writing 0 has no effect 1h = Clear Interrupt.

12.5.15 IMSET Register (Offset = 58h) [Reset = 00000000h]

IMSET is shown in Table 12-17.

Return to the Summary Table.

Interrupt mask set register. Writing a 1 to a bit in this register will set the corresponding IMASK bit.

Table 12-17. IMSET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1	EV1	W	0h	Set channel 1 event interrupt mask. 0h = Writing 0 has no effect 1h = Set interrupt mask
0	EV0	W	0h	Set channel 0 event interrupt mask. 0h = Writing 0 has no effect 1h = Set interrupt mask

12.5.16 IMCLR Register (Offset = 5Ch) [Reset = 00000000h]

IMCLR is shown in Table 12-18.

Return to the Summary Table.

Interrupt mask clear register. Writing a 1 to a bit in this register will clear the corresponding IMASK bit.

Table 12-18. IMCLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1	EV1	W	0h	Clears Channel 1 event interrupt mask. 0h = Writing 0 has no effect 1h = Clear Interrupt Mask
0	EV0	W	0h	Clears Channel 0 event interrupt mask. 0h = Writing 0 has no effect 1h = Clear Interrupt Mask

12.5.17 EMU Register (Offset = 60h) [Reset = 00000000h]

EMU is shown in Table 12-19.

Return to the Summary Table.

Emulation control register. This register controls the behavior of the IP related to core halted input.

Table 12-19. EMU Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Reserved
0	HALT	R/W	0h	Halt control. 0h = Free run option. The IP ignores the state of the core halted input. 1h = Freeze option. The IP freezes functionality when the core halted input is asserted, and resumes when it is deasserted. The freeze can either be immediate or after the IP has reached a boundary from where it can resume without corruption.

Chapter 13

Low Power Comparator and SYS0

This chapter discusses the features and functions of the CC23xx low power comparator. The registers of the low power comparator are a part of System Control and Trim (SYS0) register set.

13.1 Introduction

LPCOMP is an ultra-low-power clocked comparator that can be used for medium accuracy, and low-speed operations where power consumption is the main concern. Typical applications are wake-up on analog events like power supply monitoring or external transducers that generate analog output signals. The comparator includes input multiplexers on both the signal (positive) side and the reference (negative) side, a capacitive divider for low-power division of either the reference signal or the input signal, and programmable hysteresis that can be configured to trigger on both low and high comparator output.

Features

- Can be used for voltage monitoring in standby mode with ultra-low power consumption
- Operational across device supply voltage range
- Programmable Voltage Divider—two modes of operation:
  - Voltage Divider on Reference Side: The voltage divider block is configured on the reference side of the amplifier and the inputs can be divided in the ratio of 1/4, 1/3, 1/2, 3/4, and 1/1 with optional hysteresis applied.
  - Voltage Divider on Signal Side: The voltage divider block is configured on the signal side of the amplifier and the inputs can be divided in the ratio of 1/4, 1/3, 1/2, 3/4, and 1/1 with optional hysteresis applied.
- Input multiplexers - The input multiplexers present on the reference and signal side of the amplifier provide greater flexibility in selecting the signal and reference inputs to the amplifier based on system requirements. The multiplexers can pass inputs from external pins or supply voltages. The multiplexer on the reference side is connected to both VDDS and VDDD which can be used as internal references. The multiplexer on the signal side is directly connected to VDDS which can be used for monitoring the supply voltage.
- Hysteresis Polarity - The polarity of the hysteresis can be changed based on whether the voltage divider is present on the signal side or the reference side. This is controlled by the SYS0:LPCMPCFG[30] HYSPOL bit.

13.2 Block Diagram

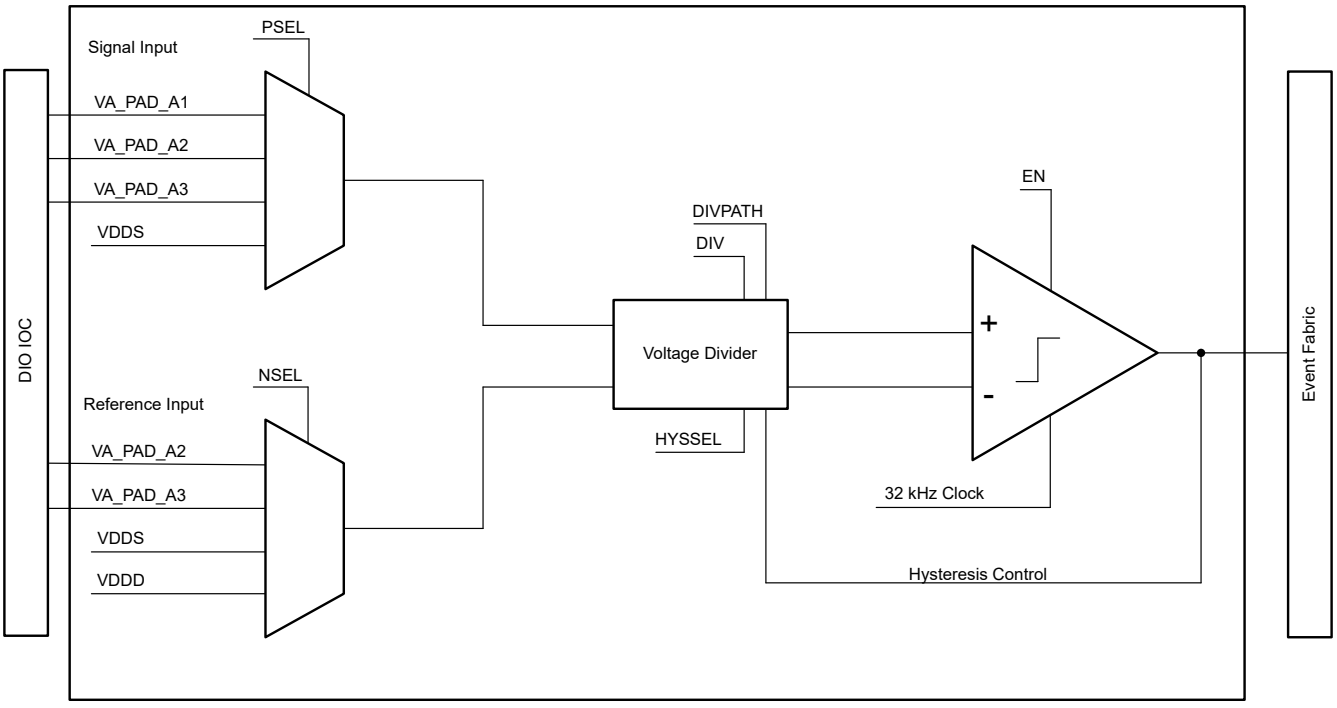


Figure 13-1. LPCOMP Block Diagram

13.3 Functional Description

LPCOMP consists of two input multiplexors that select between various inputs as shown in Figure 13-1. These inputs are routed to a programmable voltage divider. From there the inputs are routed to the comparator. The comparator has a latching output that latches on the 32kHz clock. The comparator result has a 1-3 clock cycle delay. The entire comparator module can be enabled or disabled by setting or clearing the SYS0.LPCMPCFG[0] EN bit. LFOSC must be enabled before LPCOMP is enabled.

13.3.1 Input Selection

Signal Input

The AUX_LP_COMP signal input can connect to:

- VA_PAD_A1
- VA_PAD_A2
- VA_PAD_A3
- VDDS

The selection is made by configuring SYS0.LPCMPCFG[11:8] PSEL bit field.

Reference Input

The AUX_LP_COMP reference input can connect to:

- VA_PAD_A2
- VA_PAD_A3
- VDDS
- VDDD

The selection is made by configuring SYS0.LPCMPCFG NSEL[14:12] bit field.

The mapping between AUX_LP_COMP input and DIO for CC23xx is shown in Table 13-1.

Table 13-1. AUX_LP_COMP Input to DIO Mapping

AUX_LP_COMP Input	DIO
VA_PAD_A1	DIO21_A10
VA_PAD_A2	DIO23_A8
VA_PAD_A3	DIO24_A7

13.3.2 Voltage Divider

The voltage divider is applied to either the signal input or the reference input of the comparator. The voltage divider is capable of dividing the input voltage in the ratio of 1/4, 1/3 1/2, 3/4, and 1/1 with the addition of voltage to the input signal to control hysteresis on the output. Connect the divider on the reference side. The minimum equivalent input resistance of the divider is 30 M-Ohm.

13.3.3 Hysteresis

The purpose of hysteresis is to prevent rapid changes on the comparator output due to noise on the input. This is done by increasing the voltage difference of the signal input compared to the reference input. Because of this, hysteresis will affect the accuracy of the measurement and should be disabled before the measurement. This means that the hysteresis should be enabled by the comparator output going low, if the input signal is above the threshold before the measurement is done (e.g. supply monitoring where the signal is slowly dropping to the reference level). If the signal is rising towards the reference level, hysteresis should be enabled by the comparator output going high. This is accommodated by a control-bit where the user can select if the hysteresis should be enabled by a high or low comparator output signal.

The table below shows the possible permutations and the resulting necessary configuration of the hysteresis polarity and trig-level.

Table 13-2. Hysteresis Configuration

User Case		Hysteresis polarity	Hysteresis trig-level
Divider on reference input	Signal starting high and going low	Positive (1)	Low (0)
Divider on reference input	Signal starting low and going high	Negative (0)	High (1)
Divider on signal input	Signal starting high and going low	Negative (0)	Low (0)
Divider on signal input	Signal starting low and going high	Positive (1)	High (1)

13.3.4 Wake-Up

The LPCOMP can be configured to wake the CC23xx from standby mode. This is enabled by setting the SYS0.LPCMPCFG[18] WUENSB bit.

13.4 SYS0 Registers

Table 13-3 lists the memory-mapped registers for the SYS0 registers. All register offset addresses not listed in Table 13-3 should be considered as reserved locations and the register contents should not be modified.

Table 13-3. SYS0 Registers

Offset	Acronym	Register Name	Section
0h	DESC	Description Register	Go
Ch	MUNLOCK	Mutable section Unlock	Go
100h	ATESTCFG	ATEST Configuration	Go
108h	TSENSCFG	TSENSE Configuration	Go
10Ch	LPCMPCFG	LPCMP configuration	Go
3FCh	DEVICEID	Device ID	Go
7F8h	PARTID	Part ID	Go
800h	TMUTE0	Internal. Only to be used through TI provided API.	Go
804h	TMUTE1	Internal. Only to be used through TI provided API.	Go
808h	TMUTE2	TMUTE2 trim Register	Go
80Ch	TMUTE3	Internal. Only to be used through TI provided API.	Go
810h	TMUTE4	TMUTE4 trim Register	Go
814h	TMUTE5	Internal. Only to be used through TI provided API.	Go

Complex bit access types are encoded to fit into small table cells. Table 13-4 shows the codes that are used for access types in this section.

Table 13-4. SYS0 Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
WCap	W Cap	Write Capture
Reset or Default Value		
-n		Value after reset or the default value

13.4.1 DESC Register (Offset = 0h) [Reset = 6B4E0010h]

DESC is shown in Table 13-5.

Return to the Summary Table.

Description Register
This register provides IP module ID, revision information, instance index and standard MMR registers offset.

Table 13-5. DESC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	MODID	R	6B4Eh	Module Identifier is used to uniquely identify this IP
15-12	STDIPOFF	R	0h	Standard IP MMR block offset. Standard IP MMRs are the set from aggregated IRQ registers till DTB. 0: Standard IP MMRs do not exist 0x1-0xF: Standard IP MMRs begin at offset of (64*STDIPOFF from the base IP address)
11-8	INSTIDX	R	0h	IP Instance ID number. If multiple instances of IP exist in the device, this field can identify the instance number (0-15).
7-4	MAJREV	R	1h	Major revision of IP (0-15).
3-0	MINREV	R	0h	Minor revision of IP (0-15).

13.4.2 MUNLOCK Register (Offset = Ch) [Reset = 00000000h]

MUNLOCK is shown in Table 13-6.

Return to the Summary Table.

Mutable section Unlock
This register unlocks registers in mutable section

Table 13-6. MUNLOCK Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	KEY	WCap	0h	Write the unlock key 0xC5AF_6927 to temporarily unlock registers in mutable section. The lock is set automatically if no write accesses, to the mutable section, are detected for consecutive 32 CLKULL (24MHz) clock cycles. Writing any value other that the unlock key will immediately lock the mutable register space for write access. 0h = Lock registers in the mutable section C5AF6927h = Unlock registers in the mutable section

13.4.3 ATESTCFG Register (Offset = 100h) [Reset = 0000000Fh]

ATESTCFG is shown in Table 13-7.

Return to the Summary Table.

ATEST Configuration
This register is used to configure analog switches in ATEST module.

Table 13-7. ATESTCFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	KEY	W	0h	Key must be written with value 0x5A for successful write to ATESTCFG and to unlock register state. Write with any value other than 0x5A to KEY will be ignored and register content is not updated. It is recommended to write this register with incorrect KEY to lock back register state after necessary ATESTCFG updates are done. Read value of KEY is 0x0.
23-9	RESERVED	R	0h	Reserved
8	VSEL	R/W	0h	Selects supply for ATEST switches. 0h = Selects VDDBOOST 1h = Selects VDDA
7	VA2VA1	R/W	0h	Enables isolation switch between VA_ATEST_A1 and VA_PAD_A1. 0h = Switch is open 1h = Switch is closed
6	VA2VA0	R/W	0h	Enables isolation switch between VA_ATEST_A0 and VA_PAD_A0. 0h = Switch is open 1h = Switch is closed
5	VR2VA1	R/W	0h	Enables isolation switch between VR_ATEST_A1 and VA_ATEST_A1. 0h = Switch is open 1h = Switch is closed
4	VR2VA0	R/W	0h	Enables isolation switch between VR_ATEST_A0 and VA_ATEST_A0. 0h = Switch is open 1h = Switch is closed
3	SHTVA1	R/W	1h	Shorts VA_ATEST_A1 to ground. 0h = Switch is open 1h = Switch is closed
2	SHTVA0	R/W	1h	Shorts VA_ATEST_A0 to ground. 0h = Switch is open 1h = Switch is closed
1	SHTVR1	R/W	1h	Shorts VR_ATEST_A1 to ground. 0h = Switch is open 1h = Switch is closed
0	SHTVR0	R/W	1h	Shorts VR_ATEST_A0 to ground. 0h = Switch is open 1h = Switch is closed

13.4.4 TSENSCFG Register (Offset = 108h) [Reset = 00000000h]

TSENSCFG is shown in Table 13-8.

Return to the Summary Table.

TSENSE Configuration
This register is used to configure temperature sensor module.

Table 13-8. TSENSCFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-12	RESERVED	R	0h	Software should not rely on the value of a reserved field. Writing any other value than the reset value may result in undefined behavior
11-8	SPARE	R/W	0h	Spare bits
7-2	RESERVED	R	0h	Reserved
1-0	SEL	R/W	0h	Used to enable and configure temperature sensor module. Setting the value to 0x3 will disable the temperature sensor. 0h = Temperature sensor is disabled 1h = 20uA current is injected on VR_ATEST_A0 and voltage measured on VR_ATEST_A1 2h = 20uA current is injected on VR_ATEST_A0 and ground measured on VR_ATEST_A1

13.4.5 LPCMPCFG Register (Offset = 10Ch) [Reset = 00000000h]

LPCMPCFG is shown in Table 13-9.

Return to the Summary Table.

LPCMP configuration
This register is used to configure and check the status of low-power comparator (LPCOMP) module.

Table 13-9. LPCMPCFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Software should not rely on the value of a reserved field. Writing any other value than the reset value may result in undefined behavior
30	HYSPOL	R/W	0h	Spare bit for LPCOMP
29-28	ATESTMUX	R/W	0h	Used to configure ATEST mux in comparator module and provides chosen output on VA_ATEST_A0. Note: This bit field is write-protected using global lock indicator on production device. 0h = ATEST mux is off 1h = Selects comparator output 2h = Selects voltage divider output 3h = Selects bias current output
27-25	RESERVED	R	0h	Reserved
24	EVTIFG	R/W	0h	Event flag The event flag is set when the comparator output transition is qualified based on the edge polarity configuration in EDGCFG. 0h = Clear 1h = Set
23-22	RESERVED	R	0h	Reserved
21	COUTEN	R/W	0h	Enables LPCOMP output on device pin. 0h = Disabled 1h = Enabled
20	COUT	R	0h	LPCOMP output status. This bit captures the value LPCOMP raw output. 0h = Output is low 1h = Output is high
19	RESERVED	R	0h	Reserved
18	WUENSB	R/W	0h	Enables lpcmpcfg output to wake device from standby. 0h = Disable 1h = Enable
17	EVTEN	R/W	0h	Enables event generation. Comparator module will produce event on ULL event fabric when EVTIFG is set. 0h = Disable 1h = Enable
16	EDGCFG	R/W	0h	Selects positive edge or negative edge detection on comparator output to set the event flag 0h = Rise edge detection 1h = Fall edge detection
15	RESERVED	R	0h	Reserved
14-12	NSEL	R/W	0h	Negative input selection. Setting values 0x5-0x7 will open all the switches. 0h = All switches are open 1h = Selects VA_PAD_A2 2h = Selects VA_PAD_A3 3h = Selects VDDA 4h = Selects VDDD

Table 13-9. LPCMPCFG Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
11-8	PSEL	R/W	0h	Positive input selection. Setting values 0x9-0xF will open all the switches. 0h = All switches are open 1h = Selects VA_PAD_A1 2h = Selects VA_PAD_A2 3h = Selects VA_PAD_A3 4h = Selects VR_ATEST_A0 5h = Selects VR_ATEST_A1 6h = Selects VA_ATEST_A0 7h = Selects VA_ATEST_A1 8h = Selects VDDA
7-5	HYSSEL	R/W	0h	Used to enable and select hysteresis level Hysteresis is disabled when HYSSEL = 0 and enabled for other values of HYSSEL from 1 to 7. Refer to device specific datasheet for individual hysteresis values. 0h = Hysteresis is disabled 1h = Hysteresis value: TBD 2h = Hysteresis value: TBD 3h = Hysteresis value: TBD 4h = Hysteresis value: TBD 5h = Hysteresis value: TBD 6h = Hysteresis value: TBD 7h = Hysteresis value: TBD
4	DIVPATH	R/W	0h	Used to select the path on which voltage divider is applied 0h = Divider is applied on N-side 1h = Divider is applied on P-side
3-1	DIV	R/W	0h	Used to configure reference divider. Setting values 0x5-0x7 will set the divide value to 1. 0h = Divide value is 1 1h = Divide value is 3/4 2h = Divide value is 1/2 3h = Divide value is 1/3 4h = Divide value is 1/4
0	EN	R/W	0h	Used to enable comparator module. 0h = Disable 1h = Enable

13.4.6 DEVICEID Register (Offset = 3FCh) [Reset = 0BB8402Fh]

DEVICEID is shown in Table 13-10.

Return to the Summary Table.

Device ID
This register provides Device ID information.
Note: This 32-bit register value is provided as output to DEBUGSS.

Table 13-10. DEVICEID Register Field Descriptions

Bit	Field	Type	Reset	Description
31-28	VERSION	R	0h	Monotonic increasing value indicating new hardware revision. A newer hardware revision shall never have a lower version than an older revision of hardware.
27-12	DEVICE	R	BB84h	Value generated by RAMP for the SOC. This value uniquely identifies the die from any other TI device.
11-1	MANFACTURER	R	17h	JEP 106 assigned manufacturer ID. This field identifies the device as a Texas Instruments device.
0	ALWAYSONE	R	1h	Value 1 in this bit field means that a 32-bit scan register exists.

13.4.7 PARTID Register (Offset = 7F8h) [Reset = 00000000h]

PARTID is shown in Table 13-11.

Return to the Summary Table.

Part ID
This register is programmed by boot code with Part ID information. Note: This 32-bit register value is provided as output to DEBUGSS

Table 13-11. PARTID Register Field Descriptions

Bit	Field	Type	Reset	Description
31	START	R/W	0h	Start bit 0h = Clear 1h = Set
30-28	MAJORREV	R/W	0h	Monotonic increasing value indicating a new revision of the SKU significant enough that users of the device may have to revise PCB or software design
27-24	MINORREV	R/W	0h	Monotonic increasing value indicating a new revision of the SKU that preserves compatibility with lesser MINORREV values
23-16	VARIANT	R/W	0h	Bit pattern uniquely identifying a variant of a part
15-0	PART	R/W	0h	Bit pattern uniquely identifying a part

13.4.8 TMUTE0 Register (Offset = 800h) [Reset = 00000000h]

TMUTE0 is shown in Table 13-12.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 13-12. TMUTE0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	CDACL	R/W	0h	Internal. Only to be used through TI provided API.

13.4.9 TMUTE1 Register (Offset = 804h) [Reset = 00000000h]

TMUTE1 is shown in Table 13-13.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 13-13. TMUTE1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	CDACM	R/W	0h	Internal. Only to be used through TI provided API.

13.4.10 TMUTE2 Register (Offset = 808h) [Reset = 00800000h]

TMUTE2 is shown in Table 13-14.

Return to the Summary Table.

TMUTE2 trim Register

Table 13-14. TMUTE2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	RESERVED
30-26	IBTRIM	R/W	0h	LPCOMP: Bias current trim, 250nA to be terminated across I2V, 1M Ω setting. Resulting target trim voltage 250mV.
25-23	TRIM	R/W	1h	ADC REFBUF trim bits.
22-16	LATCH	R/W	0h	SOC ADC: Latch trim bits. These bits are used in the analog IP.
15-4	OFFSET	R/W	0h	SOCADC: Offset trim bits. These bits are used in DTC.
3-2	RES	R/W	0h	SOCADC: Resistor trim bits. These bits are used in the analog IP.
1-0	CDACU	R/W	0h	SOCADC: Upper 2 bits of CDAC trim. These bits are used in DTC.

13.4.11 TMUTE3 Register (Offset = 80Ch) [Reset = 00000000h]

TMUTE3 is shown in Table 13-15.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 13-15. TMUTE3 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-26	BATC1	R/W	0h	Internal. Only to be used through TI provided API.
25-19	BATC0	R/W	0h	Internal. Only to be used through TI provided API.
18-14	TEMPC2	R/W	0h	Internal. Only to be used through TI provided API.
13-8	TEMPC1	R/W	0h	Internal. Only to be used through TI provided API.
7-0	TEMPC0	R/W	0h	Internal. Only to be used through TI provided API.

13.4.12 TMUTE4 Register (Offset = 810h) [Reset = B02E603Fh]

TMUTE4 is shown in Table 13-16.

Return to the Summary Table.

TMUTE4 trim Register

Table 13-16. TMUTE4 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-28	RECHCOMPREFLVL	R/W	Bh	Internal. Only to be used through TI provided API.
27-26	IOSTRCFG2	R/W	0h	Internal. Only to be used through TI provided API.
25-22	IOSTRCFG1	R/W	0h	Internal. Only to be used through TI provided API.
21-19	MAX	R/W	5h	Internal. Only to be used through TI provided API.
18-16	MED	R/W	6h	Internal. Only to be used through TI provided API.
15-13	MIN	R/W	3h	Internal. Only to be used through TI provided API.
12-11	DCDCLOAD	R/W	0h	Internal. Only to be used through TI provided API.
10-8	IPEAK	R/W	0h	DCDC: Set inductor peak current
7-6	DTIME	R/W	0h	Internal. Only to be used through TI provided API.
5-3	LENSEL	R/W	7h	Internal. Only to be used through TI provided API.
2-0	HENSEL	R/W	7h	Internal. Only to be used through TI provided API.

13.4.13 TMUTE5 Register (Offset = 814h) [Reset = 00000000h]

TMUTE5 is shown in Table 13-17.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 13-17. TMUTE5 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-13	RESERVED	R	0h	Internal. Only to be used through TI provided API.
12-10	DCDCDRVDS	R/W	0h	Internal. Only to be used through TI provided API.
9-5	GLDOISCLR	W	0h	Internal. Only to be used through TI provided API.
4-0	GLDOISSET	W	0h	Internal. Only to be used through TI provided API.
4-0	RESERVED	R	0h	Reserved

Chapter 14

Battery Monitor, Temperature Sensor, and DCDC Controller (PMUD)

This chapter describes the function of the Power Management Unit - Digital (PMUD).

14.1 Introduction

PMUD is composed of two components, BATMON (battery monitoring unit) and DCDC control logic.

BATMON monitors both the VDDS supply voltage and the temperature through an on-chip temperature sensor, outputs the voltage-dependent control signals to the I/O pads, gives a coarse temperature measure, and provides an SW interface for accessing battery state and temperature measurements. When enabled, the battery and temperature monitor module is operational in all operation modes except the lowest power mode, shutdown. When the device is in standby, the measurements of the battery monitor module is limited to recharge cycles. At least two measurements are performed in each recharge cycle.

The battery monitor provides voltage and temperature information to several modules. This is done to enable the correct operation and the lowest power consumption. Therefore, the recommendation is to not modify any settings in the battery monitor or turn the battery monitor off.

DCDC outputs the control signals for DCDC switches, measures battery voltage, and also takes care of turning off GLDO when not required.

14.2 Functional Description

14.2.1 BATMON

The battery monitor is a 7-bit SAR-like ADC running at 125kHz that performs alternate measurements of the supply voltage and the temperature. When the battery monitor has settled on the first measurement, the ADC stops working in SAR mode and starts linear tracking of voltage and temperature. A small digital core transforms these measurements to voltage and temperature in °C, which are read directly from the PMUD:BAT and PMUD:TEMP registers.

When a change in supply voltage or temperature is detected, the battery monitor solely tracks the voltage until the voltage has settled on a new constant level. The resolution of the ADC and the 125 kHz clock speed limits the battery monitor's capability of measuring voltage spikes. Due to the battery monitor not only alternating between temperature and battery voltage but also between checking if there has been a positive or negative change since the last read, there can be a delay of 6 clock cycles between a voltage dip and the time when the ADC can detect that the temperature or voltage has changed. Due to the prioritization of voltage tracking upon detection of voltage changes, temperature changes can be detected with more delays if the voltage is also changing at the same time. This is important to keep in mind because the battery monitor is designed to measure the battery voltage; the battery monitor is not designed to measure voltage spurs due to short periods of higher current consumption.

The module also includes an event register, PMUD.EVENT, which includes six event bits:

- PMUD.EVENT[5] TEMP_UPDATE bit: indicates that the temperature has changed.
- PMUD.EVENT[4] BATT_UPDATE bit: indicates that the voltage has changed.
- PMUD.EVENT[3] TEMP_BELOW_LL bit: indicates that the temperature is below the lower limit value that is set in the PMUD.TEMPLL register.
- PMUD.EVENT[2] TEMP_OVER_UL bit: indicates that the temperature is over the upper limit value that is set in the PMUD.TEMPUL register.
- PMUD.EVENT[1] BATT_BELOW_LL bit: indicates that the voltage is below the lower limit value that is set in the PMUD:BATTLL register.
- PMUD.EVENT[0] BATT_OVER_UL bit: indicates that the voltage is over the upper limit value that is set in the PMUD.BATTUL register.

These events must be cleared by writing to the PMUD:EVENT register. The events are asserted again if the conditions for the events are met (assertion of the new events takes precedence over the clearing of the events). In addition to the individual events listed previously, the battery monitor module has a combined event that is connected to the CPU as an interrupt line. The mask register, PMUD.EVENTMASK, can be used to select which of the events in PMUD.EVENT contribute to the combined event. This combined event is connected to the AON event fabric. For details, see Section 4.3.

14.2.2 DCDC

DCDC buck converter is a switch mode power supply that generates regulated output voltage from a higher unregulated supply. Compared to linear regulators, DCDC has a higher power efficiency which reduces total energy consumption of the SoC.

DCDC is connected in parallel with a linear regulator (GLDO), in the CC23xx implementation.

DCDC supports a maximum load current of around 45mA when configured to the maximum value. GLDO can support a peak load current of up to approximately 55mA. In applications where higher load current needs are to be met, GLDO is enabled automatically preventing VDDR output from going low. The best possible power efficiency is achieved across load variations with the parallel operation of DCDC and GLDO.

The input supply of DCDC can range from 2.2V to 3.8V. DCDC regulates the output to 1.5V. But for GLDO, the supply can range from 1.71V to 3.8V to generate a regulated output of 1.5V. DCDC hardware is designed so that when the supply voltage drops below 2.2V, DCDC automatically shuts off and only GLDO generates the regulated VDDR.

GLDO can operate independently, but DCDC needs GLDO to support driving a higher load above the DCDC limit.

DCDC hardware implements these features to provide smooth and parallel operation of DCDC and GLDO:

- Load meter
- GLDO enable toggle logic
- Adaptive peak current control

Load meter:

The load meter is used to measure the load on DCDC output as a percentage of maximum load support by DCDC. Load meter can be enabled by PMUD.DCDCCFG.LM_EN and the output is available on the status register PMUD.DCDCSTAT.LOAD. After the load meter feature is enabled, it takes approximately 500µs to generate the first output, and for any subsequent load change, it can take approximately 250µs to update the register to the accurate load level.

GLDO enable toggle logic:

GLDO enable toggle logic is used for parallel operation of DCDC and GLDO. As the load increases on VDDR above DCDC load support, GLDO is enabled by the analog circuit almost instantly. This feature is used prevent undesirable and repeated enable and disable toggles of GLDO when the load on VDDR is close to maximum DCDC load support, which can result in a higher VDDR ripple.

This feature also enables the load meter. Once GLDO is enabled due to a higher VDDR load, load meter output is continuously compared with an internally programmed DCDC load threshold. Once the load on DCDC is less than the threshold, GLDO is safely disabled. The hardware also provides a minimum enable window for GLDO to reduce the ripples on VDDR.

Adaptive peak current control:

DCDC has a programmable peak current to vary the maximum load support. To support higher load, DCDC can be programmed to higher peak current setting, but then DCDC operates on a reduced power efficiency. With lower peak current DCDC works with higher efficiency but with lower maximum load support. So, for given load on VDDR, there is an optimum peak current setting for DCDC which provides the best power efficiency. Adaptive peak current control is an algorithm that updates peak current dynamically until the optimum value is reached.

For the operation of Adaptive peak current control, load meter output is continuously monitored against two programmable DCDC load thresholds (high threshold and low threshold). If the load on DCDC is more than the high threshold, the algorithm increments the peak current value programmed. If the load on DCDC drops below the low threshold, the programmed peak current value is decremented. For a given load on VDDR, after a few increments and decrements, the final peak current value reaches the optimum setting for the best power

efficiency. Each increment or decrement operation can have a delay of up to 250µs, and to reach the optimum peak current setting, the adaptive peak current control algorithm can take up to 1ms.

Registers used for Adaptive peak current control:

1. DCDCCFG.ADP_IPEAK_EN: Enables Adaptive peak current control hardware algorithm.
2. DCDCCFG.LM_HIGHTH: DCDC load meter high threshold value. DCDC load meter output is in a percentage scale so the valid values are ‘d1 to ‘d100. Values from ‘d101 to ‘d127 are invalid and are not to be used. The recommended value is ‘d80.
3. DCDCCFG.LM_LOWTH: DCDC load meter low threshold value. The recommended value is ‘d50.
4. SYS0.TMUTE4.IPEAK: This field can be used to program the maximum current that has to be supported by DCDC from ‘d0 (approximately 14mA) to ‘d7 (approximately 45mA). When the adaptive algorithm is enabled, then this value acts as the limit up to which the algorithm can increment the DCDC peak current.
5. DCDCSTAT.IPEAK: DCDC peak current value programmed within the SYS0.TMUTE4.IPEAK register when this algorithm is not enabled. When this is enabled, this register outputs the peak current value being driven by the algorithm.

14.3 PMUD Registers

Table 14-1 lists the memory-mapped registers for the PMUD registers. All register offset addresses not listed in Table 14-1 should be considered as reserved locations and the register contents should not be modified.

Table 14-1. PMUD Registers

Offset	Acronym	Register Name	Section
0h	CTL	Control	Go
4h	MEASCFG	Internal. Only to be used through TI provided API.	Go
28h	BAT	Last Measured Battery Voltage	Go
2Ch	BATUPD	Battery Update	Go
30h	TEMP	Last measured Temperature in Degree Celsius	Go
34h	TEMPUPD	Temperature Update	Go
48h	EVENTMASK	Event Mask	Go
4Ch	EVENT	Event	Go
50h	BATTUL	Battery Upper Limit	Go
54h	BATTLL	Battery Lower Limit	Go
58h	TEMPUL	Temperature Upper Limit	Go
5Ch	TEMPLL	Temperature Lower Limit	Go
90h	PREG0	Internal. Only to be used through TI provided API.	Go
94h	PREG1	Internal. Only to be used through TI provided API.	Go
98h	PREG2	Internal. Only to be used through TI provided API.	Go
9Ch	DCDCCFG	DCDC configuration	Go
A0h	DCDCSTAT	DCDC status	Go

Complex bit access types are encoded to fit into small table cells. Table 14-2 shows the codes that are used for access types in this section.

Table 14-2. PMUD Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

14.3.1 CTL Register (Offset = 0h) [Reset = 00000004h]

CTL is shown in Table 14-3.

Return to the Summary Table.

Control
General Configuration of BATMON

Table 14-3. CTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-3	RESERVED	R	0h	Reserved
2	HYST_EN	R/W	1h	Enables hysteresis on both battery and temperature measurements. 0h = Disable 1h = Enable
1	CALC_EN	R/W	0h	Configuration of the calculation block that converts the digital battery/temperature level to a Volt/Celsius value. 0h = Calculation disabled 1h = Calculation enabled
0	MEAS_EN	R/W	0h	Configuration of the measurement block that interfaces with the analog domain. 0h = Measurements disabled 1h = Measurements enabled (battery voltage and temperature)

14.3.2 MEASCFG Register (Offset = 4h) [Reset = 00000000h]

MEASCFG is shown in Table 14-4.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 14-4. MEASCFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1-0	PER	R/W	0h	Internal. Only to be used through TI provided API.

14.3.3 BAT Register (Offset = 28h) [Reset = 00000000h]

BAT is shown in Table 14-5.

Return to the Summary Table.

Last Measured Battery Voltage
This register should be read when BATUPD.STA = 1

Table 14-5. BAT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10-8	INT	R	0h	Integer part: 0x0: Battery voltage = 0V + fractional part ... 0x3: Battery voltage = 3V + fractional part 0x4: Battery voltage = 4V + fractional part
7-0	FRAC	R	0h	Fractional part, standard binary fractional encoding. 0x00: .0V ... 0x20: 1/8 = .125V 0x40: 1/4 = .25V 0x80: 1/2 = .5V ... 0xA0: 1/2 + 1/8 = .625V ... 0xFF: 1/2 + 1/4 + 1/8 + ... + 1/256 = 0.99V

14.3.4 BATUPD Register (Offset = 2Ch) [Reset = 00000000h]

BATUPD is shown in Table 14-6.

Return to the Summary Table.

Battery Update
Indicates BAT Updates

Table 14-6. BATUPD Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Reserved
0	STA	R/W	0h	Battery update status. Write 1 to clear the status. 0h = No update since last clear 1h = New battery voltage present

14.3.5 TEMP Register (Offset = 30h) [Reset = 00000000h]

TEMP is shown in Table 14-7.

Return to the Summary Table.

Last measured Temperature in Degree Celsius
This register should be read when TEMPUPD.STA = 1.

Table 14-7. TEMP Register Field Descriptions

Bit	Field	Type	Reset	Description
31-17	RESERVED	R	0h	Reserved
16-8	INT	R	0h	Integer part of temperature value (signed) Total value = INT + FRAC 2's complement encoding 0x100: Min value (-256°C) 0x1D8: -40°C 0x1FF: -1°C 0x00: 0°C 0x1B: 27°C 0x55: 85°C 0xFF: Max value (255°C)
7-6	FRAC	R	0h	Fractional part of temperature value. Total value = INT + FRAC The encoding is an extension of the 2's complement encoding. 00: 0.0°C 01: 0.25°C 10: 0.5°C 11: 0.75°C For example: 000000001,00 = (1+0,00) = 1,00 000000000,11 = (0+0,75) = 0,75 000000000,10 = (0+0,50) = 0,50 000000000,01 = (0+0,25) = 0,25 000000000,00 = (0+0,00) = 0,00 111111111,11 = (-1+0,75) = -0,25 111111111,10 = (-1+0,50) = -0,50 111111111,01 = (-1+0,25) = -0,75 111111111,00 = (-1+0,00) = -1,00 111111110,11 = (-2+0,75) = -1,25
5-0	RESERVED	R	0h	Reserved

14.3.6 TEMPUPD Register (Offset = 34h) [Reset = 00000000h]

TEMPUPD is shown in Table 14-8.

Return to the Summary Table.

Temperature Update
Indicates TEMP Updates

Table 14-8. TEMPUPD Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Reserved
0	STA	R/W	0h	Temperature update status. Write 1 to clear the status. 0h = No temperature update since last clear 1h = New temperature value present

14.3.7 EVENTMASK Register (Offset = 48h) [Reset = 00000000h]

EVENTMASK is shown in Table 14-9.

Return to the Summary Table.

Event Mask

Table 14-9. EVENTMASK Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Reserved
5	TEMP_UPDATE_MASK	R/W	0h	1: EVENT.TEMP_UPDATE contributes to combined event from BATMON 0: EVENT.TEMP_UPDATE does not contribute to combined event from BATMON
4	BATT_UPDATE_MASK	R/W	0h	1: EVENT.BATT_UPDATE contributes to combined event from BATMON 0: EVENT.BATT_UPDATE does not contribute to combined event from BATMON
3	TEMP_BELOW_LL_MASK	R/W	0h	1: EVENT.TEMP_BELOW_LL contributes to combined event from BATMON 0: EVENT.TEMP_BELOW_LL does not contribute to combined event from BATMON
2	TEMP_OVER_UL_MASK	R/W	0h	1: EVENT.TEMP_OVER_UL contributes to combined event from BATMON 0: EVENT.TEMP_OVER_UL does not contribute to combined event from BATMON
1	BATT_BELOW_LL_MASK	R/W	0h	1: EVENT.BATT_BELOW_LL contributes to combined event from BATMON 0: EVENT.BATT_BELOW_LL does not contribute to combined event from BATMON
0	BATT_OVER_UL_MASK	R/W	0h	1: EVENT.BATT_OVER_UL contributes to combined event from BATMON 0: EVENT.BATT_OVER_UL does not contribute to combined event from BATMON

14.3.8 EVENT Register (Offset = 4Ch) [Reset = 00000000h]

EVENT is shown in Table 14-10.

Return to the Summary Table.

Event

Table 14-10. EVENT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Reserved
5	TEMP_UPDATE	R/W	0h	Alias to TEMPUPD.STA
4	BATT_UPDATE	R/W	0h	Alias to BATUPD.STA
3	TEMP_BELOW_LL	R/W	0h	Read: 1: Temperature level is below the lower limit set by TEMPLL. 0: Temperature level is not below the lower limit set by TEMPLL. Write: 1: Clears the flag 0: No change in the flag
2	TEMP_OVER_UL	R/W	0h	Read: 1: Temperature level is above the upper limit set by TEMPUL. 0: Temperature level is not above the upper limit set by TEMPUL. Write: 1: Clears the flag 0: No change in the flag
1	BATT_BELOW_LL	R/W	0h	Read: 1: Battery level is below the lower limit set by BATTLL. 0: Battery level is not below the lower limit set by BATTLL. Write: 1: Clears the flag 0: No change in the flag
0	BATT_OVER_UL	R/W	0h	Read: 1: Battery level is above the upper limit set by BATTUL. 0: Battery level is not above the upper limit set by BATTUL. Write: 1: Clears the flag 0: No change in the flag

14.3.9 BATTUL Register (Offset = 50h) [Reset = 000007FFh]

BATTUL is shown in Table 14-11.

Return to the Summary Table.

Battery Upper Limit
Total battery voltage = INT + FRAC
It is a sum of integer and fractional parts

Table 14-11. BATTUL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10-8	INT	R/W	7h	Integer part: Total battery voltage = INT + FRAC (integer and fractional part) 0x0: Battery voltage = 0V + fractional part ... 0x3: Battery voltage = 3V + fractional part 0x4: Battery voltage = 4V + fractional part
7-0	FRAC	R/W	FFh	Fractional part, standard binary fractional encoding. 0x00: .0V ... 0x20: 1/8 = .125V 0x40: 1/4 = .25V 0x80: 1/2 = .5V ... 0xA0: 1/2 + 1/8 = .625V ... 0xFF: 1/2 + 1/4 + 1/8 + ... + 1/256 = 0.99V

14.3.10 BATTLL Register (Offset = 54h) [Reset = 00000000h]

BATTLL is shown in Table 14-12.

Return to the Summary Table.

Battery Lower Limit
Total battery voltage = INT + FRAC
It is a sum of integer and fractional parts

Table 14-12. BATTLL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10-8	INT	R/W	0h	Integer part: Total battery voltage = INT + FRAC (integer and fractional part) 0x0: Battery voltage = 0V + fractional part ... 0x3: Battery voltage = 3V + fractional part 0x4: Battery voltage = 4V + fractional part
7-0	FRAC	R/W	0h	Fractional part, standard binary fractional encoding. 0x00: .0V ... 0x20: 1/8 = .125V 0x40: 1/4 = .25V 0x80: 1/2 = .5V ... 0xA0: 1/2 + 1/8 = .625V ... 0xFF: 1/2 + 1/4 + 1/8 + ... + 1/256 = 0.99V

14.3.11 TEMPUL Register (Offset = 58h) [Reset = 0000FFC0h]

TEMPUL is shown in Table 14-13.

Return to the Summary Table.

Temperature Upper Limit

Table 14-13. TEMPUL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-17	RESERVED	R	0h	Reserved
16-8	INT	R/W	FFh	Integer part (signed) of temperature upper limit. Total value = INT + FRAC 2's complement encoding 0x100: Min value (-256°C) 0x1D8: -40°C 0x1FF: -1°C 0x00: 0°C 0x1B: 27°C 0x55: 85°C 0xFF: Max value (255°C)
7-6	FRAC	R/W	3h	Fractional part of temperature upper limit. Total value = INT + FRAC The encoding is an extension of the 2's complement encoding. 00: 0.0°C 01: 0.25°C 10: 0.5°C 11: 0.75°C For example: 000000001,00 = (1+0,00) = 1,00 000000000,11 = (0+0,75) = 0,75 000000000,10 = (0+0,50) = 0,50 000000000,01 = (0+0,25) = 0,25 000000000,00 = (0+0,00) = 0,00 111111111,11 = (-1+0,75) = -0,25 111111111,10 = (-1+0,50) = -0,50 111111111,01 = (-1+0,25) = -0,75 111111111,00 = (-1+0,00) = -1,00 111111110,11 = (-2+0,75) = -1,25
5-0	RESERVED	R	0h	Reserved

14.3.12 TEMPLL Register (Offset = 5Ch) [Reset = 00010000h]

TEMPLL is shown in Table 14-14.

Return to the Summary Table.

Temperature Lower Limit

Table 14-14. TEMPLL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-17	RESERVED	R	0h	Reserved
16-8	INT	R/W	100h	Integer part (signed) of temperature lower limit. Total value = INT + FRAC 2's complement encoding 0x100: Min value (-256°C) 0x1D8: -40°C 0x1FF: -1°C 0x00: 0°C 0x1B: 27°C 0x55: 85°C 0xFF: Max value (255°C)
7-6	FRAC	R/W	0h	Fractional part of temperature lower limit. Total value = INT + FRAC The encoding is an extension of the 2's complement encoding. 00: 0.0°C 01: 0.25°C 10: 0.5°C 11: 0.75°C For example: 000000001,00 = (1+0,00) = 1,00 000000000,11 = (0+0,75) = 0,75 000000000,10 = (0+0,50) = 0,50 000000000,01 = (0+0,25) = 0,25 000000000,00 = (0+0,00) = 0,00 111111111,11 = (-1+0,75) = -0,25 111111111,10 = (-1+0,50) = -0,50 111111111,01 = (-1+0,25) = -0,75 111111111,00 = (-1+0,00) = -1,00 111111110,11 = (-2+0,75) = -1,25
5-0	RESERVED	R	0h	Reserved

14.3.13 PREG0 Register (Offset = 90h) [Reset = 00000000h]

PREG0 is shown in Table 14-15.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 14-15. PREG0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-12	RESERVED	R	0h	Reserved
11	LOW_IPEAK_DIS	R/W	0h	Internal. Only to be used through TI provided API.
10	SOCLDO_ITESTEN	R/W	0h	Internal. Only to be used through TI provided API.
9-7	SOCLDO_ATBSEL	R/W	0h	Internal. Only to be used through TI provided API.
6-5	UDIGLDO_ATBSEL	R/W	0h	Internal. Only to be used through TI provided API.
4-2	DIGLDO_ATBSEL	R/W	0h	Internal. Only to be used through TI provided API.
1	SPARE	R/W	0h	Internal. Only to be used through TI provided API.
0	UDIGLDO_EN	R/W	0h	Internal. Only to be used through TI provided API.

14.3.14 PREG1 Register (Offset = 94h) [Reset = 00000000h]

PREG1 is shown in Table 14-16.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 14-16. PREG1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-20	RESERVED	R	0h	Reserved
19	TEST_DCDC_NMOS	R/W	0h	Internal. Only to be used through TI provided API.
18	TEST_DCDC_PMOS	R/W	0h	Internal. Only to be used through TI provided API.
17	DITHER_EN	R/W	0h	Internal. Only to be used through TI provided API.
16	GLDO_AON	R/W	0h	Internal. Only to be used through TI provided API.
15	RCHG_BLK_VTRIG_EN	R/W	0h	Internal. Only to be used through TI provided API.
14	RCHG_BLK_ATEST_EN	R/W	0h	Internal. Only to be used through TI provided API.
13	RCHG_FORCE_SAMP_VREF	R/W	0h	Internal. Only to be used through TI provided API.
12	RCHG_COMP_CLK_DIS	R/W	0h	Internal. Only to be used through TI provided API.
11-8	RESERVED	R	0h	Reserved
7	SPARE	R/W	0h	Internal. Only to be used through TI provided API.
6	VDDR_ATBSEL	R/W	0h	Internal. Only to be used through TI provided API.
5	GLDO_EA_BIAS_DIS	R/W	0h	Internal. Only to be used through TI provided API.
4-1	GLDO_ATBSEL	R/W	0h	Internal. Only to be used through TI provided API.
0	RESERVED	R	0h	Reserved

14.3.15 PREG2 Register (Offset = 98h) [Reset = 00000000h]

PREG2 is shown in Table 14-17.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 14-17. PREG2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Reserved
5	RSTNMASK	R/W	0h	Internal. Only to be used through TI provided API.
4	DCDC_RCHG_ATBSEL	R/W	0h	Internal. Only to be used through TI provided API.
3-0	PMUREG_ATBSEL	R/W	0h	Internal. Only to be used through TI provided API.

14.3.16 DCDCCFG Register (Offset = 9Ch) [Reset = 00000000h]

DCDCCFG is shown in Table 14-18.

Return to the Summary Table.

DCDC configuration register

Table 14-18. DCDCCFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-23	RESERVED	R	0h	Reserved
22-16	LM_HIGHTH	R/W	0h	DCDC load meter high threshold value for adaptive IPEAK adjustment. DCDC load meter output is in percentage scale so the applicable values are 'd1 to 'd100. Values from 'd101 to 'd127 are invalid and not to be used.
15	RESERVED	R	0h	Reserved
14-8	LM_LOWTH	R/W	0h	DCDC load meter low threshold value for adaptive IPEAK adjustment. DCDC load meter output is in percentage scale so the applicable values are 'd1 to 'd100. Values from 'd101 to 'd127 are invalid and not to be used.
7-5	RESERVED	R	0h	Reserved
4	ADP_IPEAK_EN	R/W	0h	This bit is used to enable adaptive IPEAK adjustment scheme in hardware. When this bit is set, DCDC IPEAK value is automatically adjusted to suitable value by sensing the DCDC load meter output for better DCDC operational efficiency. 0h = Disable 1h = Enable
3-1	RESERVED	R	0h	Reserved
0	LMEN	R/W	0h	This bit is used to enable DCDC load meter. Software can obtain DCDC load meter value from DCDCSTAT regiser and adjust IPEAK setting in SYS0.TDCDC register accordingly. 0h = Disable 1h = Enable

14.3.17 DCDCSTAT Register (Offset = A0h) [Reset = 00000000h]

DCDCSTAT is shown in Table 14-19.

Return to the Summary Table.

DCDC status register

Table 14-19. DCDCSTAT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10-8	IPEAK	R	0h	DCDC IPEAK value. This value is same as what is programmed in SYS0:TMUTE4.IPEAK when adaptive IPEAK adjustment scheme is not enabled, and it shows current IPEAK value applied by hardware when adaptive IPEAK adjustment scheme is enabled. Note: Software can only support IPEAK = 1
7	RESERVED	R	0h	Reserved
6-0	LOAD	R	0h	This indicates DCDC load meter output value in percentage scale. Applicable range is 'd1 to 'd100.

Chapter 15

Micro Direct Memory Access (µDMA)

This chapter describes the direct memory access (DMA) controller, known as µDMA.

15.1 Introduction

The CC23xx includes a direct memory access (DMA) controller, known as µDMA. The µDMA controller provides a way to offload data transfer tasks from the Arm® Cortex®-M0+ processor, allowing for more efficient use of the processor and the available bus bandwidth. The µDMA controller can perform transfers between memory and peripherals. The controller has dedicated channels for each supported on-chip module, and can be programmed to automatically perform transfers between peripherals and memory as the peripheral is ready to transfer more data. The µDMA controller provides the following features:

- 8-channel configurable µDMA controller
- Support for memory-to-memory, memory-to-peripheral, and peripheral-to-memory in multiple transfer modes:
  - Basic for simple transfer scenarios
  - Ping-pong for continuous data flow
  - Scatter-gather for a programmable list of arbitrary transfers initiated from a single request
- Highly flexible and configurable channel operation:
  - Independently configured and operated channels
  - Dedicated channels for supported on-chip modules
  - Primary and secondary channel assignments
  - Flexible channel assignments
  - One channel each for receive and transmit paths for bidirectional modules
  - Dedicated channel for software-initiated transfers
  - Per-channel configurable priority scheme
  - Optional software-initiated requests for any channel
- Two levels of priority
- Data sizes of 8, 16, and 32 bits
- Transfer size is programmable in binary steps from 1 to 1024
- Source and destination address increment size of byte, halfword, word, or no increment
- Maskable peripheral requests
- Interrupt on transfer completion with a separate interrupt per channel

15.2 Block Diagram

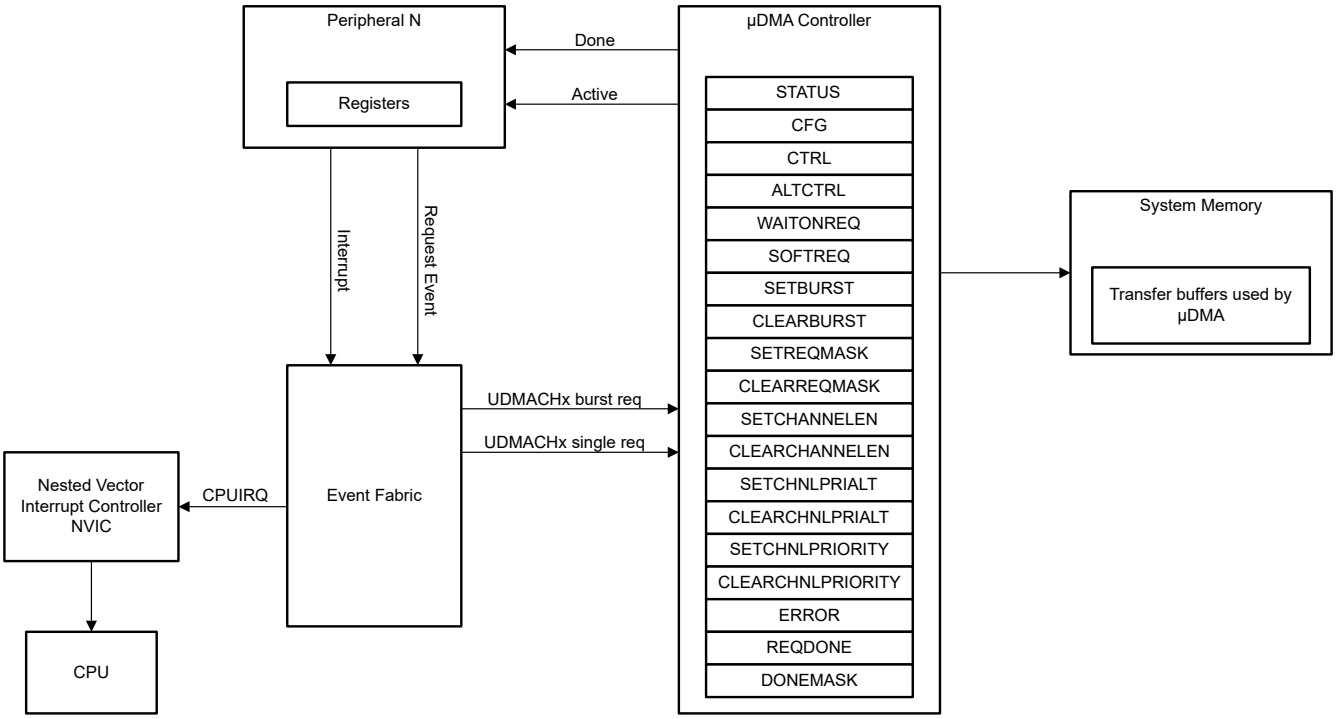


Figure 15-1. µDMA Block Diagram

15.3 Functional Description

The µDMA controller is a flexible and highly configurable DMA controller designed to work efficiently with the Arm Cortex-M0+ processor core of the microcontroller. The controller supports multiple data sizes and address increment schemes, multiple levels of priority among DMA channels, and several transfer modes to allow for sophisticated programmed data transfers.

Each supported peripheral function has a dedicated channel on the µDMA controller that can be configured independently. The µDMA controller implements a configuration method using channel control structures maintained in system memory by the processor. While simple transfer modes are supported, it is also possible to build up sophisticated task lists in memory that allow the µDMA controller to perform arbitrary-sized transfers to and from arbitrary locations as part of a single transfer request. The µDMA controller also supports the use of ping-pong buffering to accommodate constant streaming of data to or from a peripheral.

Each channel also has a configurable arbitration size. The arbitration size is the number of items that are transferred in a burst before the µDMA controller requests channel priority. Using the arbitration size, it is possible to control exactly how many items are transferred to or from a peripheral every time a µDMA service request is made.

15.3.1 Channel Assignments

Table 15-1 lists µDMA channel assignments to peripherals.

Table 15-1. Channel Assignments

µDMA Channel	Type [1] [2]	DTB [3]	Triggers
0	DCH	YES	SPI0TXTRG, UART0RXTRG
1	DCH	YES	SPI0RXTRG, UART0TXTRG
2	DCH	YES	LRFDTRG, UART0TXTRG
3	DCH	YES	ADC0TRG, UART0RXTRG
4	DCH	YES	LAESTRGA, LRFDTRG
5	DCH	YES	LAESTRGB, ADC0TRG
6	ECH	YES	ANY
7	ECH	YES	ANY

(1) The dedicated µDMA Channel (DCH) type can only be connected to peripherals with a dedicated µDMA interface.
(2) Event Publisher Channel (ECH) type can only be connected to publishers.
(3) For DTB output, only SREQ and REQ is considered.

15.3.2 Priority

The µDMA controller assigns priority to each channel based on the channel number and the priority-level bit for the channel. Channel 0 has the highest priority, and as the channel number increases, the priority of a channel decreases. Each channel has a priority-level bit to provide two levels of priority: default priority and high priority. If the priority-level bit is set, then that channel has a higher priority than all other channels at default priority. If multiple channels are set for high priority, then the channel number is used to determine relative priority among all the high-priority channels.

The priority bit for a channel can be set using the DMA.SETCHNLPRIORITY register and cleared with the DMA.CLEARCHNLPRIORITY register.

15.3.3 Arbitration Size

When a µDMA channel requests a transfer, the µDMA controller arbitrates among all the channels making a request and services the µDMA channel with the highest priority. Once a transfer begins, it continues for a selectable number of transfers before re-arbitrating among the requesting channels. The arbitration size can be configured for each channel, ranging from 1 to 1024 item transfers. After the µDMA controller transfers the number of items specified by the arbitration size, the controller then checks among all the channels making a request, and services the channel with the highest priority.

If a lower-priority µDMA channel uses a large arbitration size, the latency for higher-priority channels is increased because the µDMA controller completes the lower-priority burst before checking for higher-priority requests. Therefore, lower-priority channels must not use a large arbitration size for best response on high-priority channels.

The arbitration size can also be thought of as burst size. Arbitration size is the maximum number of items that are transferred at any one time in a burst. Here, the term *arbitration* refers to the determination of the µDMA channel priority, not arbitration for the bus. When the µDMA controller arbitrates for the bus, the processor always takes priority. Furthermore, the µDMA controller is delayed whenever the processor must perform a bus transaction on the same bus, even in the middle of a burst transfer.

15.3.4 Request Types

The µDMA controller responds to two types of requests from a peripheral: single request or burst request. Each peripheral can support either or both types of requests. A single request means that the peripheral is ready to transfer one item, while a burst request means that the peripheral is ready to transfer multiple items.

The µDMA controller responds differently depending on whether the peripheral is making a single request or a burst request. If both types of requests are asserted and the µDMA channel has been set up for a burst transfer, then the burst request takes precedence. Table 15-2 lists how each peripheral supports the two request types.

Table 15-2. Request Type Support

Peripheral	Single Request Signal	Burst Request Signal
ADC	None (FIFO is not empty)	Sequencer IE bit (FIFO is half full)
General-purpose timer	Raw interrupt pulse	None
GPIO	Raw interrupt pulse	None
SPI TX	TXFIFO not full	TXFIFO level (configurable)
SPI RX	RXFIFO not empty	RXFIFO level (configurable)
UART TX	TXFIFO not full	TXFIFO level (configurable)
UART RX	RXFIFO not empty	RXFIFO level (configurable)

15.3.4.1 Single Request

When a single request is detected (not a burst request), the µDMA controller transfers one item and then stops to wait for another request.

15.3.4.2 Burst Request

When a burst request is detected, the µDMA controller transfers the number of items that is the lesser of the arbitration size or the number of items remaining in the transfer. Therefore, the arbitration size must be the same as the number of data items that the peripheral can accommodate when making a burst request. For example, the UART and SPI, which use a mix of single or burst requests, could generate a burst request based on the FIFO trigger level. In this case, the arbitration size must be set to the amount of data that the FIFO can transfer when the trigger level is reached. A burst transfer runs to completion once it starts and cannot be interrupted, even by a higher-priority channel. Burst transfers complete in a shorter time than the same number of non-burst transfers.

It may be desirable to use only burst transfers and not allow single transfers (for example, when the nature of the data is such that it only makes sense when transferred together as a single unit rather than one piece at a time). The single request can be disabled in the DMA.SETBURST register. By setting the bit for a channel in this register, the µDMA controller responds only to burst requests for that channel.

15.3.5 Channel Configuration

The µDMA controller uses an area of system memory to store a set of channel control structures in a table. The control table can have one or two entries for each µDMA channel. Each entry in the table structure contains

source and destination pointers, transfer size, and transfer mode. The control table can be located anywhere in system memory, but the control table must be contiguous and aligned on a 256-byte boundary.

Table 15-3 describes the memory layout of the channel control table. Each channel can have one or two control structures in the control table—a primary control structure and an optional, alternate control structure. The table is organized with all of the primary entries in the first half of the table, and with all the alternate structures in the second half of the table. The primary entry is used for simple transfer modes where transfers can be reconfigured and restarted after each transfer completes. In this case, the alternate control structures are not used and only the first half of the table must be allocated in memory. The rest of the memory can be used for something else. If a more complex transfer mode is used, such as ping-pong or scatter-gather, then the alternate control structure is also used and memory space must be allocated for the entire table.

Any unused memory in the control table can be used by the application, which includes the control structures for any channels that are unused by the application, as well as the unused control word for each channel.

Table 15-3. Control Structure Memory Map

Offset	Channel
0x0	0, Primary
0x10	1, Primary
...	...
0x70	7, Primary
0x80	0, Alternate
0x90	1, Alternate
...	...
0xF00	7, Alternate

Table 15-4 describes an individual control-structure entry in the control table. Each entry is aligned on a 16-byte boundary. The entry contains four 4-byte long words: the source end pointer, the destination end pointer, the control word, and an unused entry. The inclusive end pointers point to the ending address of the transfer. If the source or destination is non-incrementing (as for a peripheral register), then the pointer must point to the transfer address.

Table 15-4. Channel Control Structure

Offset	Description
0x000	Source end pointer
0x004	Destination end pointer
0x008	Control word
0x00C	Unused entry

The control word contains the following fields:

- Source and destination data sizes
- Source and destination address increment size
- Number of transfers before bus arbitration
- Total number of items to transfer
- Useburst flag
- Transfer mode

The control parameters for a channel can be set using the driver library function:

```
void uDMAChannelControlSet();
```

The µDMA controller updates the transfer size and transfer mode fields as the transfer is performed. At the end of a transfer, the transfer size indicates 0, and the transfer mode indicates stopped. Because the control word is

modified by the µDMA controller, the control word must be reconfigured before each new transfer. The source and destination end pointers are not modified, so the source and destination end pointers can be left unchanged if the source or destination addresses remain the same.

Before starting a transfer, a µDMA channel must be enabled by setting the appropriate bit in the DMA.SETCHANNELEN register. A channel can be disabled by setting the channel bit in the DMA.CLEARCHANNELEN register. At the end of a complete µDMA transfer, the controller automatically disables the channel.

15.3.6 Transfer Modes

The µDMA controller supports several transfer modes. Two of the modes support simple, one-time transfers. Several complex modes support a continuous flow of data.

15.3.6.1 Stop Mode

While stop mode is not a transfer mode, stop is a valid value for the *mode* field of the control word. When the mode field has the *stop* value, the µDMA controller does not perform any transfers and disables the channel if enabled. The µDMA controller updates the control word to set the mode to stop at the end of a transfer. This mode can be useful in scatter-gather operations.

15.3.6.2 Basic Mode

In basic mode, the µDMA controller performs transfers as long as there are more items to transfer, and a transfer request is present. This mode is used with peripherals that assert a µDMA request signal whenever the peripheral is ready for a data transfer. Basic mode must not be used in any situation where the request is not present during the entire transfer.

The µDMA controller sets the mode for that channel to stop when all of the items have been transferred using basic mode.

15.3.6.3 Auto Mode

Auto mode is similar to basic mode, except when a transfer request is received, the transfer completes, even if the µDMA request is removed. This mode is suitable for software-triggered transfers. Generally, auto mode is not used with a peripheral.

The µDMA controller sets the mode for that channel to stop when all the items have been transferred using auto mode.

15.3.6.4 Ping-Pong Mode

Ping-pong mode supports a continuous data flow to or from a peripheral. Both the primary and alternate data structures must be implemented to use ping-pong mode. Both structures are set up by the processor for data transfer between memory and a peripheral. The transfer is started using the primary control structure.

When the transfer using the primary control structure completes, the µDMA controller reads the alternate control structure for that channel to continue the transfer. Each time this occurs, an interrupt is generated, and the processor can reload the control structure for the just-completed transfer. Data flow can continue indefinitely this way, using the primary and alternate control structures to switch between buffers as the data flows to or from the peripheral.

Figure 15-2 shows an example operation in ping-pong mode.

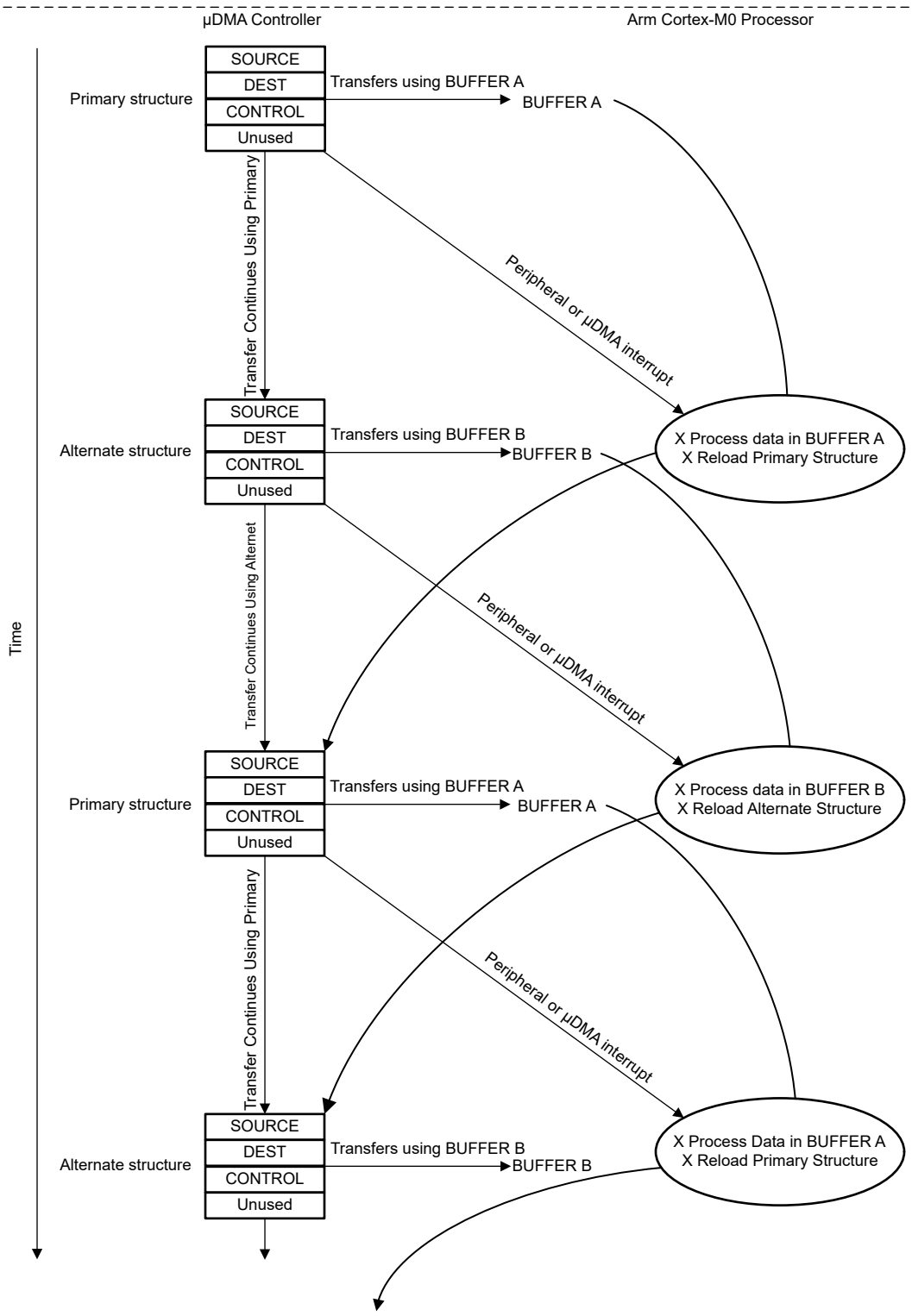


Figure 15-2. Example of Ping-Pong µDMA Transaction

15.3.6.5 Memory Scatter-Gather Mode

Memory scatter-gather mode is a complex mode used when data must be transferred to or from varied locations in memory instead of a set of contiguous locations in a memory buffer. For example, a gather µDMA operation could be used to selectively read the payload of several stored packets of a communication protocol, and store them together in sequence in a memory buffer.

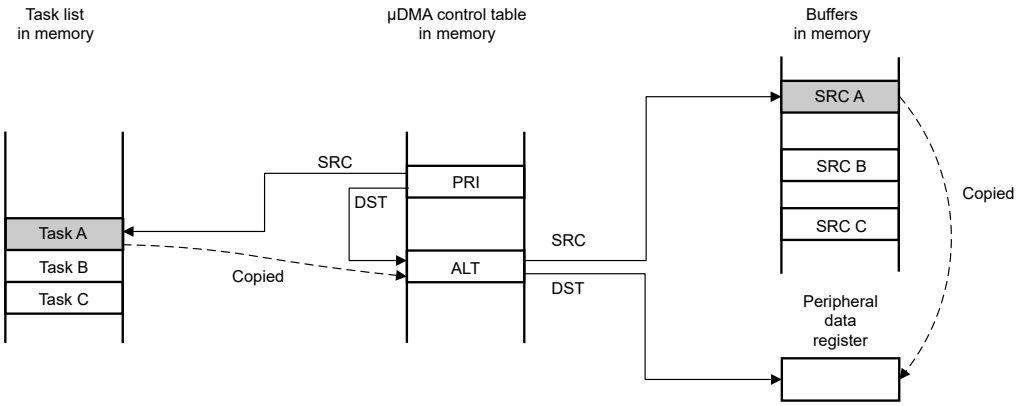
In memory scatter-gather mode, the primary control structure is used to program the alternate control structure from a table in memory. The table is set up by the processor software and contains a list of control structures, each containing the source and destination end pointers, and the control word for a specific transfer. The mode of each control word must be set to memory scatter-gather mode. Each entry in the table is, in turn, copied to the alternate structure where it is then executed. The µDMA controller alternates between using the primary control structure to copy the next transfer instruction from the list, and then executing the new transfer instruction. The end of the list is marked by programming the control word for the last entry to use auto-transfer mode. When the last transfer is performed using auto mode, the µDMA controller stops. A completion interrupt is generated only after the last transfer.

It is possible to loop the list by having the last entry copy the primary control structure to point back to the beginning of the list (or to a new list). It is also possible to trigger a set of other channels to perform a transfer, either directly, by programming a write to the software trigger for another channel, or indirectly, by causing a peripheral action that results in a µDMA request.

By programming the µDMA controller using this method, a set of arbitrary transfers can be performed based on a single µDMA request.

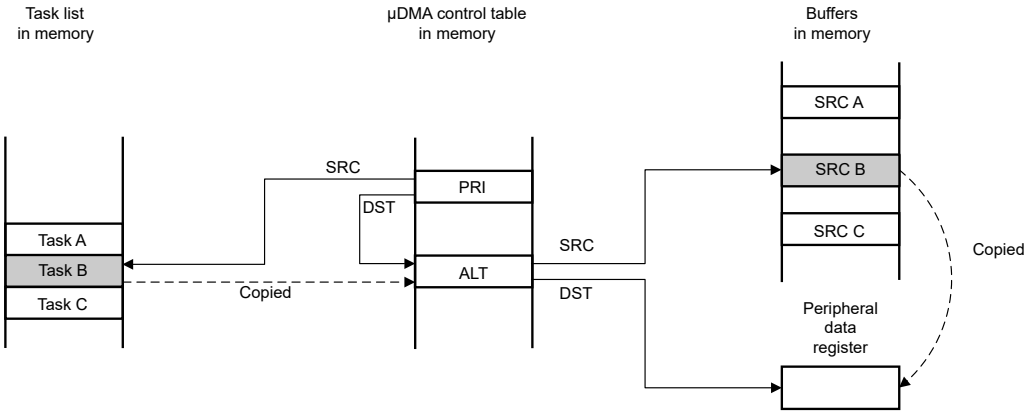
Figure 15-3 shows an example of operation in memory scatter-gather mode. This example shows a gather operation, where data in three separate buffers in memory is copied together into one buffer. Figure 15-3 shows how the application sets up a µDMA task list in memory, which is then used by the controller to perform three sets of copy operations from different locations in memory. The primary control structure for the channel used for the operation is configured to copy from the task list to the alternate control structure.

Figure 15-4 shows the sequence as the µDMA controller performs the three sets of copy operations. First, using the primary control structure, the µDMA controller loads the alternate control structure with Task A. The µDMA controller then performs the copy operation specified by Task A, copying the data from the source buffer A to the destination buffer. Next, the µDMA controller again uses the primary control structure to load Task B into the alternate control structure and then performs the B operation with the alternate control structure. The process is repeated for Task C.



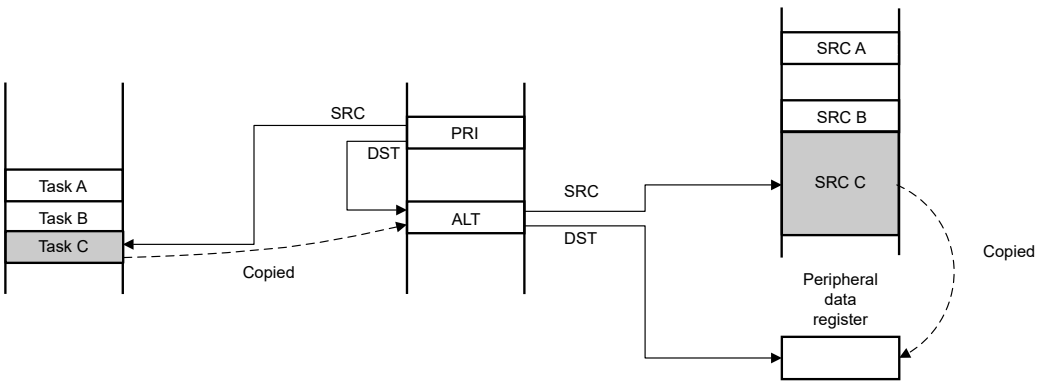
Using the primary control structure of the channel, the µDMA controller copies task A configuration to the alternate control structure of the channel.

Then, using the alternate control structure of the channel, the µDMA controller copies data from source buffer A to the peripheral data register.



Using the primary control structure of the channel, the µDMA controller copies task B configuration to the alternate control structure of the channel.

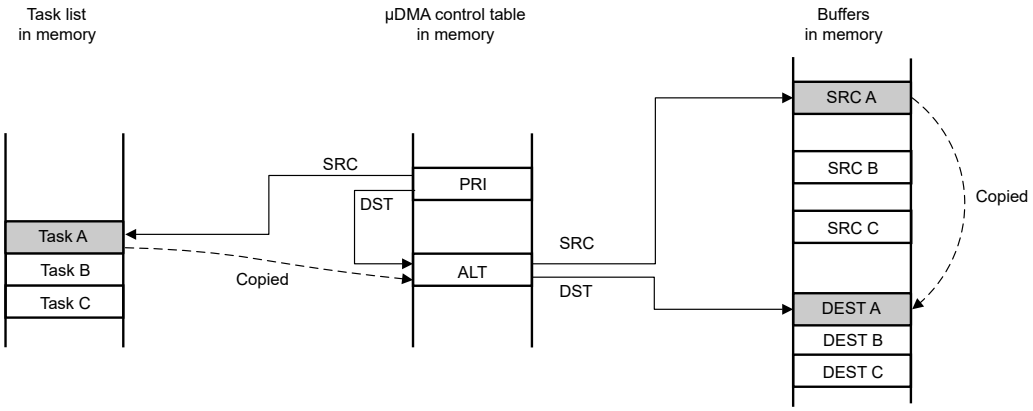
Then, using the alternate control structure of the channel, the µDMA controller copies data from source buffer B to the peripheral data register.



Using the primary control structure of the channel, the µDMA controller copies task C configuration to the alternate control structure of the channel.

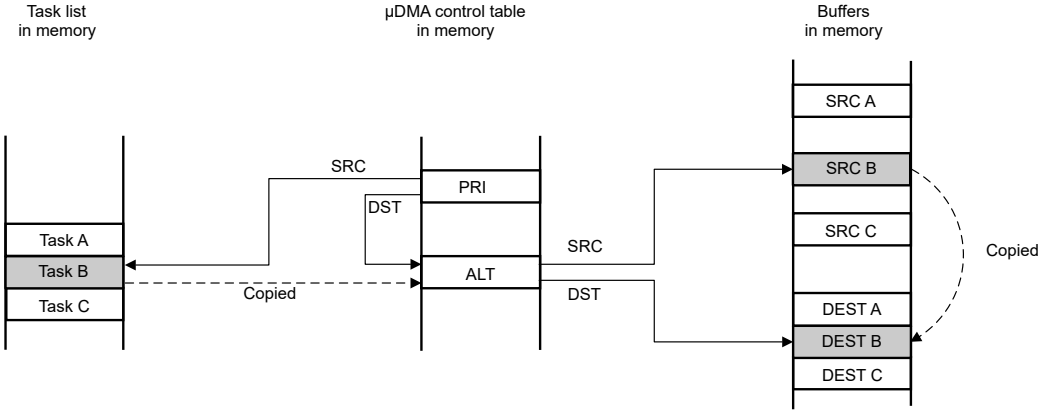
Then, using the alternate control structure of the channel, the µDMA controller copies data from source buffer C to the peripheral data register.

Figure 15-3. Memory Scatter-Gather, Setup, and Configuration



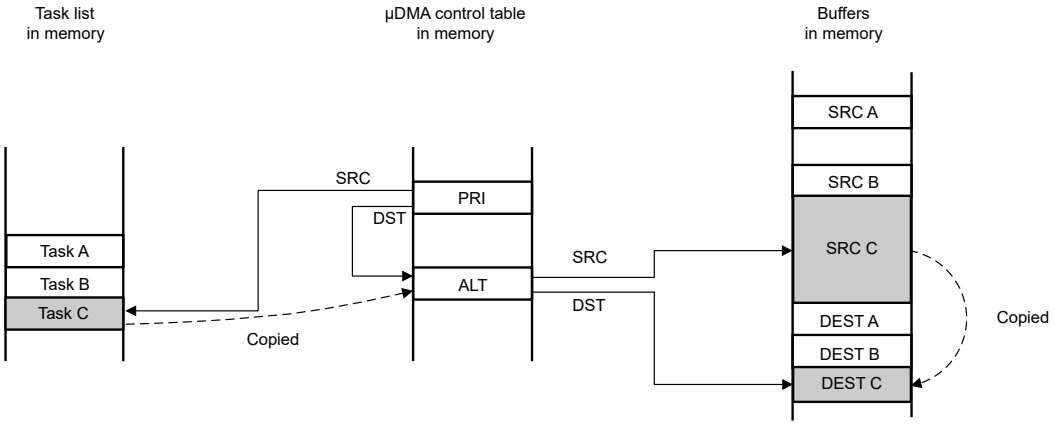
Using the primary control structure of the channel, the µDMA controller copies task A configuration to the alternate control structure of the channel.

Then, using the alternate control structure of the channel, the µDMA controller copies data from source buffer A to the destination buffer.



Using the primary control structure of the channel, the µDMA controller copies task B configuration to the alternate control structure of the channel.

Then, using the alternate control structure of the channel, the µDMA controller copies data from source buffer B to the destination buffer.



Using the primary control structure of the channel, the µDMA controller copies task C configuration to the alternate control structure of the channel.

Then, using the alternate control structure of the channel, the µDMA controller copies data from source buffer C to destination buffer.

Figure 15-4. Memory Scatter-Gather, µDMA Copy Sequence

15.3.6.6 Peripheral Scatter-Gather Mode

Peripheral scatter-gather mode is similar to memory scatter-gather mode, except that the transfers are controlled by a peripheral making a µDMA request. When the µDMA controller detects a request from the peripheral, the µDMA controller uses the primary control structure to copy one entry from the list to the alternate control structure and then performs the transfer. At the end of this transfer, the next transfer is started only if the peripheral again asserts a µDMA request. The µDMA controller continues to perform transfers from the list only when the peripheral makes a request until the last transfer completes. A completion interrupt is generated only after the last transfer.

Using this method, the µDMA controller can transfer data to or from a peripheral from a set of arbitrary locations whenever the peripheral is ready to transfer data.

Figure 15-5 shows an example of operation in peripheral scatter-gather mode. This example shows a gather operation where data from three separate buffers in memory is copied to a single peripheral data register. Figure 15-5 shows how the application sets up a µDMA task list in memory, which is then used by the controller to perform three sets of copy operations from different locations in memory. The primary control structure for the channel used for the operation is configured to copy from the task list to the alternate control structure.

Figure 15-6 shows the sequence as the µDMA controller performs the three sets of copy operations. First, using the primary control structure, the µDMA controller loads the alternate control structure with Task A. The µDMA controller then performs the copy operation specified by Task A, copying the data from the source buffer A to the peripheral data register. Next, the µDMA controller again uses the primary control structure to load Task B into the alternate control structure and then performs the B operation with the alternate control structure. The process is repeated for Task C.

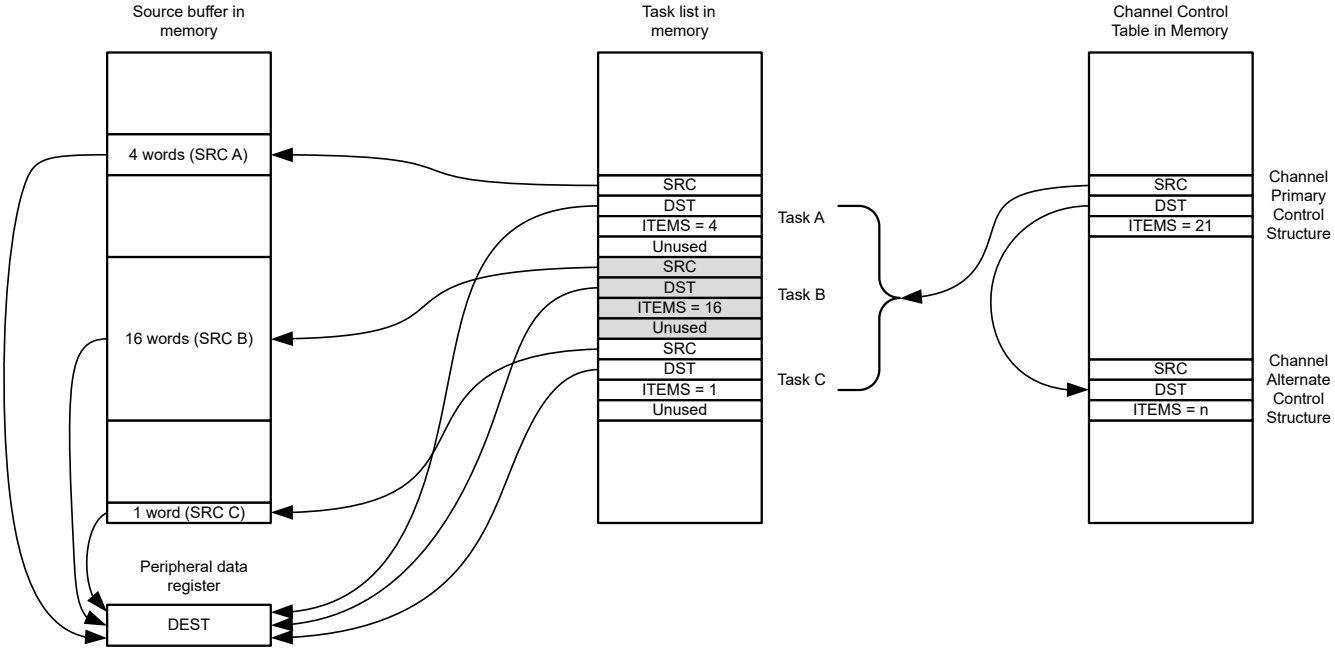
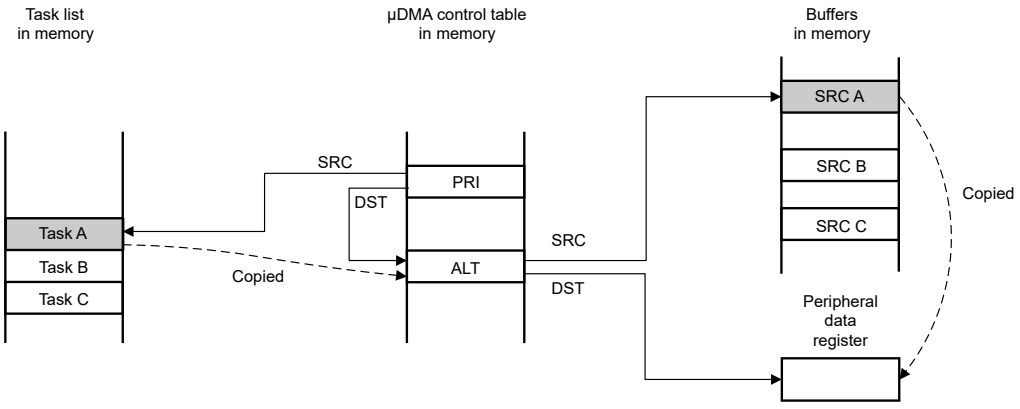
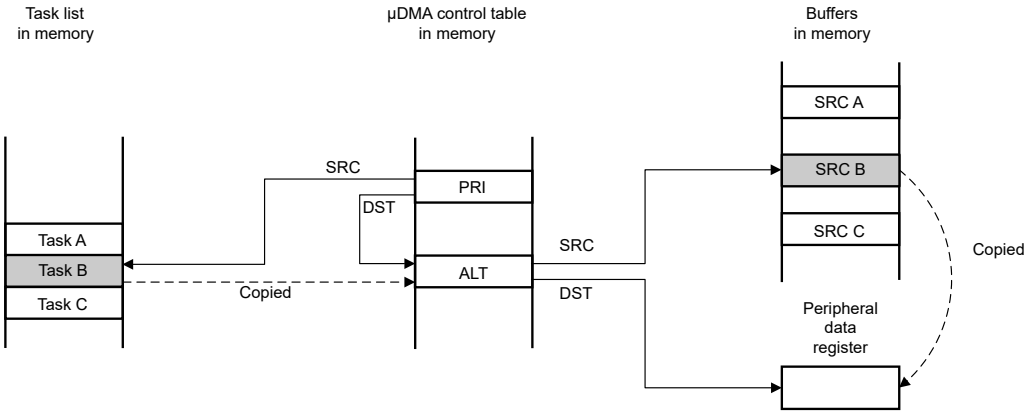


Figure 15-5. Peripheral Scatter-Gather, Setup, and Configuration



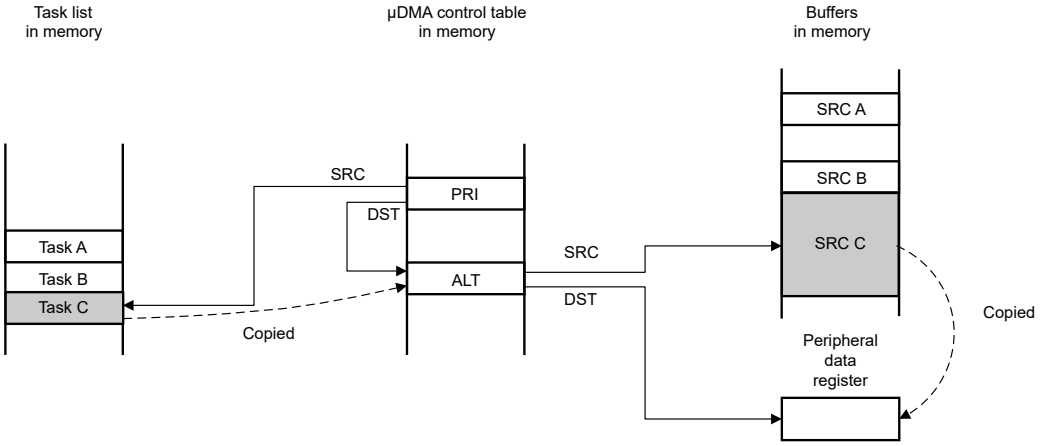
Using the primary control structure of the channel, the µDMA controller copies task A configuration to the alternate control structure of the channel.

Then, using the alternate control structure of the channel, the µDMA controller copies data from source buffer A to the peripheral data register.



Using the primary control structure of the channel, the µDMA controller copies task B configuration to the alternate control structure of the channel.

Then, using the alternate control structure of the channel, the µDMA controller copies data from source buffer B to the peripheral data register.



Using the primary control structure of the channel, the µDMA controller copies task C configuration to the alternate control structure of the channel.

Then, using the alternate control structure of the channel, the µDMA controller copies data from source buffer C to the peripheral data register.

Figure 15-6. Peripheral Scatter-Gather, µDMA Copy Sequence

15.3.7 Transfer Size and Increments

The µDMA controller supports transfer data sizes of 8, 16, or 32 bits. The source and destination data size must be the same for any given transfer. The source and destination address can be automatically incremented by bytes, half-words, words, or set to no increment. The source and destination address increment values can be set independently; it is not necessary for the address increment to match the data size, as long as the increment is the same or larger than the data size. For example, it is possible to perform a transfer using 8-bit data size

by using an address increment of full words (4 bytes). The data to be transferred must be aligned in memory according to the data size (8, 16, or 32 bits).

Table 15-5 provides the configuration to read from a peripheral that supplies 8-bit data.

Table 15-5. µDMA Read Example: 8-Bit Peripheral

Field	Configuration
Source data size	8 bits
Destination data size	8 bits
Source address increment	No increment
Destination address increment	Byte
Source end pointer	Peripheral read FIFO register
Destination end pointer	End of the data buffer in memory

15.3.8 Peripheral Interface

Each peripheral that supports µDMA has a single request or burst request signal that is asserted when the peripheral is ready to transfer data (see Table 15-2). The request signal can be disabled or enabled using the DMA.SETREQMASK and DMA.CLEARREQMASK registers, respectively. The µDMA request signal is disabled or masked when the channel request mask bit is set. When the request is not masked and the µDMA channel is configured correctly and enabled, the peripheral asserts the request signal and the µDMA controller begins the transfer.

Note

The peripheral must disable all interrupts to the event fabric when using µDMA to transfer data to and from a peripheral.

When a µDMA transfer is complete, the µDMA controller generates an interrupt; for more information, see Section 15.3.10.

For more information on how a specific peripheral interacts with the µDMA controller, refer to the DMA Operation section in the chapter that discusses that peripheral.

15.3.9 Software Request

Channels can be set up to perform software transfers through the DMA.SOFTREQ register. If the channel used for software is also tied to a specific peripheral, the dma_done/interrupt signal is provided directly to the Arm Cortex-M0+ CPU instead of sending the signal to the peripheral. The interrupt used is a combined interrupt, number 46 – software µDMA interrupt, for all software transfers.

If software uses a µDMA channel of the peripheral to initiate a request, then the completion interrupt occurs on the interrupt vector for the peripheral instead of occurring on the software interrupt vector.

Note

DMA software requests are specified on DMA channels 7 and 8.

15.3.10 Interrupts and Errors

The µDMA controller generates a completion interrupt on the interrupt vector of the peripheral when a µDMA transfer completes. Therefore, if µDMA is used to transfer data for a peripheral and interrupts are used, then the

interrupt handler for that peripheral must be designed to handle the µDMA transfer completion interrupt. If the transfer uses the software µDMA channel, then the completion interrupt occurs on the dedicated software µDMA interrupt vector (see Table 15-6).

When µDMA is enabled for a peripheral, the µDMA controller stops the normal transfer interrupts for a peripheral from reaching the interrupt controller (INTC). The interrupts are still reported in the interrupt registers of the peripheral. Thus, when a large amount of data is transferred using µDMA, instead of receiving multiple interrupts from the peripheral as data flows, the INTC receives only one interrupt when the transfer completes. Unmasked peripheral error interrupts continue to be sent to the INTC.

When a µDMA channel generates a completion interrupt, the CHNLS bit corresponding to the peripheral channel is set in the DMA Channel Request Done register, DMA.REQDONE. This register can be used by the interrupt handler code of the peripheral to determine if the interrupt was caused by the µDMA channel or an error event reported by the interrupt registers of the peripheral. The completion interrupt request from the µDMA controller is automatically cleared when the interrupt handler is activated.

If the µDMA controller encounters a bus or memory protection error when trying to perform a data transfer, the controller disables the µDMA channel that caused the error and generates an interrupt on the µDMA error interrupt vector. The processor can read the DMA Clear Bus Error register, DMA.ERROR[0] STATUS bit to determine if an error is pending. The STATUS bit is set if an error occurs. The error can be cleared by setting the STATUS bit to 1.

Note

The µDMA error event is connected as an interrupt to Arm Cortex-M0+ processor through the event fabric.

Table 15-6 lists the dedicated interrupt assignments for the µDMA controller.

Table 15-6. µDMA Interrupt Assignments

Interrupt	Assignment
20	µDMA software channel transfer done
21	µDMA error

15.3.11 Initialization and Configuration

15.3.11.1 Module Initialization

The µDMA controller resides in the peripheral domain, which must be powered up to enable the µDMA controller. The following steps are necessary:

1. Enable the µDMA controller by setting the CLKCTL.CLKENSET0[17] DMA bit or by using the driver library function:

```
Power_enable_DMA()
```

2. Enable the µDMA controller by setting the DMA Configuration register, DMA.CFG[0] MASTERENABLE bit.
3. Program the location of the channel control table by writing the base address of the table to the DMA Channel Control Base Pointer register, DMA.CTRL. The base address must be aligned on a 256-byte boundary.

15.3.11.2 Configuring a Memory-to-Memory Transfer

The µDMA channels 6 and 7 are dedicated for software-initiated transfers. This specific example uses channel 6. No attributes must be set for a software-based transfer. The attributes are cleared by default, but are explicitly cleared as shown in the following sections.

15.3.11.3 Configure the Channel Attributes

Configure the channel attributes as follows, or use the following driver library function:

```
uDMAChannelAttributeDisable(uint32_t ui32Base, uint32_t ui32ChannelNum, uint32_t ui32Attr)
```

1. Program bit 0 of the DMA Set Channel Priority register, DMA.SETCHNLPRIORITY, or the DMA Clear Channel Priority register, DMA.CLEARCHNLPRIORITY, to set the channel to high priority or default priority.
2. Set bit 6 of the DMA Clear Channel Primary Alternate register, DMA.CLEARCHNLPRIALT, to select the primary channel control structure for this transfer.
3. Set bit 6 of the DMA Channel Clear Useburst register, DMA.CLEARBURST, to allow the µDMA controller to respond to single requests and burst requests.
4. Set bit 6 of the DMA Clear Channel Request Mask register, DMA.CLEARREQMASK, to allow the µDMA controller to recognize requests for this channel.

15.3.11.4 Configure the Channel Control Structure

This example transfers 256 words from one memory buffer to another. Channel 6 is used for a software transfer, and the control structure for channel 6 must be configured to transfer 8-bit data with source and destination increments in bytes and byte-wise buffer copy. A bus arbitration size of eight can be used here.

The transfer buffer and transfer size are now configured. The transfer uses auto mode, which means that the transfer automatically runs to completion after the first request.

15.3.11.5 Start the Transfer

Finally, the channel must be enabled. A request must also be made because this is a software-initiated transfer. The request starts the transfer.

1. Enable global interrupts:

   ```
   IntMasterEnable()
   ```

   and enable interrupt for µDMA:

   ```
   IntEnable(uint32_t ui32Interrupt)
   ```

2. Enable the channel by setting bit 0 of the µDMA Set Channel Enable register, DMA.SETCHANNELEN.
3. Issue a transfer request by setting bit 0 of the µDMA Channel Software Request register, DMA.SOFTREQ.
4. The µDMA transfer begins. If the interrupt is enabled, then the processor is notified by an interrupt when the transfer completes.

If needed, the status can be checked by reading the DMA.SETCHANNELEN register bit 0. This bit is automatically cleared when the transfer completes.

15.3.11.6 Software Considerations

The µDMA driver should not disable µDMA when it has active transactions. The behavior when µDMA starts up again is not defined and by the time µDMA is reenabled, peripheral state and memory contents might have changed.

Software must also ensure that all µDMA channels from peripherals are disabled before entering standby, so no new requests are generated when µDMA is being disabled.

If the software doesn't take care of turning off µDMA only after all ongoing µDMA transactions are completed, and instead turns the clocks off in the middle of a transfer sequence—µDMA only ensures that the current transaction is cleanly terminated and the bus does not hang—but it does not wait until all transactions within the ongoing sequence are completed (which might take an arbitrarily large amount of time).

15.4 DMA Registers

Table 15-7 lists the memory-mapped registers for the DMA registers. All register offset addresses not listed in Table 15-7 should be considered as reserved locations and the register contents should not be modified.

Table 15-7. DMA Registers

Offset	Acronym	Register Name	Section
0h	STATUS	Status Register.	Go
4h	CFG	Configuration Register.	Go
8h	CTRL	Channel Control Data Base Pointer Register.	Go
Ch	ALTCTRL	Channel Alternate Control Data Base Pointer Register.	Go
10h	WAITONREQ	Channel Wait On Request Status Register.	Go
14h	SOFTREQ	Channel Software Request Register.	Go
18h	SETBURST	Channel Set UseBurst Register.	Go
1Ch	CLEARBURST	Channel Clear UseBurst Register.	Go
20h	SETREQMASK	Channel Set Request Mask Register.	Go
24h	CLEARREQMASK	Clear Channel Request Mask Register.	Go
28h	SETCHANNELEN	Set Channel Enable Register.	Go
2Ch	CLEARCHANNELEN	Clear Channel Enable Register.	Go
30h	SETCHNLPRIALT	Channel Set Primary-Alternate Register.	Go
34h	CLEARCHNLPRIALT	Channel Clear Primary-Alternate Register.	Go
38h	SETCHNLPRIORITY	Set Channel Priority Register.	Go
3Ch	CLEARCHNLPRIORITY	Clear Channel Priority Register.	Go
4Ch	ERROR	Error Status and Clear Register.	Go
504h	REQDONE	Channel Request Done Register.	Go
520h	DONEMASK	Channel Request Done Mask Register.	Go

Complex bit access types are encoded to fit into small table cells. Table 15-8 shows the codes that are used for access types in this section.

Table 15-8. DMA Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

15.4.1 STATUS Register (Offset = 0h) [Reset = 00070000h]

STATUS is shown in Table 15-9.

Return to the Summary Table.

Status Register.

Table 15-9. STATUS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-28	TEST	R	0h	0x0: Controller does not include the integration test logic 0x1: Controller includes the integration test logic 0x2: Undefined ... 0xF: Undefined
27-21	RESERVED	R	0h	Reserved
20-16	TOTALCHANNELS	R	7h	Register value returns number of available uDMA channels minus one. For example a read out value of: 0x00: Show that the controller is configured to use 1 uDMA channel 0x01: Shows that the controller is configured to use 2 uDMA channels ... 0x1F: Shows that the controller is configured to use 32 uDMA channels (32-1=31=0x1F)
15-8	RESERVED	R	0h	Reserved
7-4	STATE	R	0h	Current state of the control state machine. State can be one of the following: 0x0: Idle 0x1: Reading channel controller data 0x2: Reading source data end pointer 0x3: Reading destination data end pointer 0x4: Reading source data 0x5: Writing destination data 0x6: Waiting for uDMA request to clear 0x7: Writing channel controller data 0x8: Stalled 0x9: Done 0xA: Peripheral scatter-gather transition 0xB: Undefined ... 0xF: Undefined.
3-1	RESERVED	R	0h	Reserved
0	MASTERENABLE	R	0h	Shows the enable status of the controller as configured by CFG.MASTERENABLE: 0h = Controller is disabled 1h = Controller is enabled

15.4.2 CFG Register (Offset = 4h) [Reset = 00000000h]

CFG is shown in Table 15-10.

Return to the Summary Table.

Configuration Register.

Table 15-10. CFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	W	0h	Reserved
7-5	PRTOCTRL	W	0h	Sets the AHB-Lite bus protocol protection state by controlling the AHB signal HProt[3:1] as follows: Bit [7] Controls HProt[3] to indicate if a cacheable access is occurring. Bit [6] Controls HProt[2] to indicate if a bufferable access is occurring. Bit [5] Controls HProt[1] to indicate if a privileged access is occurring. When bit [n] = 1 then the corresponding HProt bit is high. When bit [n] = 0 then the corresponding HProt bit is low. This field controls HProt[3:1] signal for all transactions initiated by uDMA except two transactions below: - the read from the address indicated by source address pointer - the write to the address indicated by destination address pointer HProt[3:1] for these two exceptions can be controlled by dedicated fields in the channel configutation descriptor.
4-1	RESERVED	W	0h	Reserved
0	MASTERENABLE	W	0h	Enables the controller. 0h = Disables the controller 1h = Enables the controller

15.4.3 CTRL Register (Offset = 8h) [Reset = 00000000h]

CTRL is shown in Table 15-11.

Return to the Summary Table.

Channel Control Data Base Pointer Register.

Table 15-11. CTRL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	BASEPTR	R/W	0h	This register point to the base address for the primary data structures of each uDMA channel. This is not stored in module, but in system memory, thus space must be allocated for this usage when uDMA is in usage
7-0	RESERVED	R	0h	Reserved

15.4.4 ALTCTRL Register (Offset = Ch) [Reset = 00000080h]

ALTCTRL is shown in Table 15-12.

Return to the Summary Table.

Channel Alternate Control Data Base Pointer Register.

Table 15-12. ALTCTRL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	BASEPTR	R	80h	This register shows the base address for the alternate data structures and is calculated by module, thus read only

15.4.5 WAITONREQ Register (Offset = 10h) [Reset = 000000FFh]

WAITONREQ is shown in Table 15-13.

Return to the Summary Table.

Channel Wait On Request Status Register.

Table 15-13. WAITONREQ Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	CHNLSTATUS	R	FFh	Channel wait on request status: Bit [Ch] = 0: Once uDMA receives a single or burst request on channel Ch, this channel may come out of active state even if request is still present. Bit [Ch] = 1: Once uDMA receives a single or burst request on channel Ch, it keeps channel Ch in active state until the requests are deasserted. This handshake is necessary for channels where the requester is in an asynchronous domain or can run at slower clock speed than uDMA

15.4.6 SOFTREQ Register (Offset = 14h) [Reset = 00000000h]

SOFTREQ is shown in Table 15-14.

Return to the Summary Table.

Channel Software Request Register.

Table 15-14. SOFTREQ Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	CHNLS	W	0h	Set the appropriate bit to generate a software uDMA request on the corresponding uDMA channel Bit [Ch] = 0: Does not create a uDMA request for channel Ch Bit [Ch] = 1: Creates a uDMA request for channel Ch Writing to a bit where a uDMA channel is not implemented does not create a uDMA request for that channel

15.4.7 SETBURST Register (Offset = 18h) [Reset = 00000000h]

SETBURST is shown in Table 15-15.

Return to the Summary Table.

Channel Set UseBurst Register.

Table 15-15. SETBURST Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	CHNLS	R/W	0h	Returns the useburst status, or disables individual channels from generating single uDMA requests. The value R is the arbitration rate and stored in the controller data structure. Read as: Bit [Ch] = 0: uDMA channel Ch responds to both burst and single requests on channel Ch. The controller performs 2^R, or single, bus transfers. Bit [Ch] = 1: uDMA channel Ch does not respond to single transfer requests. The controller only responds to burst transfer requests and performs 2^R transfers. Write as: Bit [Ch] = 0: No effect. Use the CLEARBURST.CHNLS to set bit [Ch] to 0. Bit [Ch] = 1: Disables single transfer requests on channel Ch. The controller performs 2^R transfers for burst requests. Writing to a bit where a uDMA channel is not implemented has no effect

15.4.8 CLEARBURST Register (Offset = 1Ch) [Reset = 00000000h]

CLEARBURST is shown in Table 15-16.

Return to the Summary Table.

Channel Clear UseBurst Register.

Table 15-16. CLEARBURST Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	CHNLS	W	0h	Set the appropriate bit to enable single transfer requests. Write as: Bit [Ch] = 0: No effect. Use the SETBURST.CHNLS to disable single transfer requests. Bit [Ch] = 1: Enables single transfer requests on channel Ch. Writing to a bit where a uDMA channel is not implemented has no effect.

15.4.9 SETREQMASK Register (Offset = 20h) [Reset = 00000000h]

SETREQMASK is shown in Table 15-17.

Return to the Summary Table.

Channel Set Request Mask Register.

Table 15-17. SETREQMASK Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	CHNLS	R/W	0h	Returns the burst and single request mask status, or disables the corresponding channel from generating uDMA requests. Read as: Bit [Ch] = 0: External requests are enabled for channel Ch. Bit [Ch] = 1: External requests are disabled for channel Ch. Write as: Bit [Ch] = 0: No effect. Use the CLEARREQMASK.CHNLS to enable uDMA requests. Bit [Ch] = 1: Disables uDMA burst request channel [Ch] and uDMA single request channel [Ch] input from generating uDMA requests. Writing to a bit where a uDMA channel is not implemented has no effect

15.4.10 CLEARREQMASK Register (Offset = 24h) [Reset = 00000000h]

CLEARREQMASK is shown in Table 15-18.

Return to the Summary Table.

Clear Channel Request Mask Register.

Table 15-18. CLEARREQMASK Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	CHNLS	W	0h	Set the appropriate bit to enable uDMA request for the channel. Write as: Bit [Ch] = 0: No effect. Use the SETREQMASK.CHNLS to disable channel Ch from generating requests. Bit [Ch] = 1: Enables channel [Ch] to generate uDMA requests. Writing to a bit where a uDMA channel is not implemented has no effect.

15.4.11 SETCHANNELEN Register (Offset = 28h) [Reset = 00000000h]

SETCHANNELEN is shown in Table 15-19.

Return to the Summary Table.

Set Channel Enable Register.

Table 15-19. SETCHANNELEN Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	CHNLS	R/W	0h	Returns the enable status of the channels, or enable the corresponding channels. Read as: Bit [Ch] = 0: Channel Ch is disabled. Bit [Ch] = 1: Channel Ch is enabled. Write as: Bit [Ch] = 0: No effect. Use the CLEARCHANNELEN.CHNLS to disable a channel Bit [Ch] = 1: Enables channel Ch Writing to a bit where a uDMA channel is not implemented has no effect

15.4.12 CLEARCHANNELEN Register (Offset = 2Ch) [Reset = 00000000h]

CLEARCHANNELEN is shown in Table 15-20.

Return to the Summary Table.

Clear Channel Enable Register.

Table 15-20. CLEARCHANNELEN Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	CHNLS	W	0h	Set the appropriate bit to disable the corresponding uDMA channel. Write as: Bit [Ch] = 0: No effect. Use the SETCHANNELEN.CHNLS to enable uDMA channels. Bit [Ch] = 1: Disables channel Ch Writing to a bit where a uDMA channel is not implemented has no effect

15.4.13 SETCHNLPRIALT Register (Offset = 30h) [Reset = 00000000h]

SETCHNLPRIALT is shown in Table 15-21.

Return to the Summary Table.

Channel Set Primary-Alternate Register.

Table 15-21. SETCHNLPRIALT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	CHNLS	R/W	0h	Returns the channel control data structure status, or selects the alternate data structure for the corresponding uDMA channel. Read as: Bit [Ch] = 0: uDMA channel Ch is using the primary data structure. Bit [Ch] = 1: uDMA channel Ch is using the alternate data structure. Write as: Bit [Ch] = 0: No effect. Use the CLEARCHNLPRIALT.CHNLS to disable a channel Bit [Ch] = 1: Selects the alternate data structure for channel Ch Writing to a bit where a uDMA channel is not implemented has no effect

15.4.14 CLEARCHNLPRIALT Register (Offset = 34h) [Reset = 00000000h]

CLEARCHNLPRIALT is shown in Table 15-22.

Return to the Summary Table.

Channel Clear Primary-Alternate Register.

Table 15-22. CLEARCHNLPRIALT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	CHNLS	W	0h	Clears the appropriate bit to select the primary data structure for the corresponding uDMA channel. Write as: Bit [Ch] = 0: No effect. Use the SETCHNLPRIALT.CHNLS to select the alternate data structure. Bit [Ch] = 1: Selects the primary data structure for channel Ch. Writing to a bit where a uDMA channel is not implemented has no effect

15.4.15 SETCHNLPRIORITY Register (Offset = 38h) [Reset = 00000000h]

SETCHNLPRIORITY is shown in Table 15-23.

Return to the Summary Table.

Set Channel Priority Register.

Table 15-23. SETCHNLPRIORITY Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	CHNLS	R/W	0h	Returns the channel priority mask status, or sets the channel priority to high. Read as: Bit [Ch] = 0: uDMA channel Ch is using the default priority level. Bit [Ch] = 1: uDMA channel Ch is using a high priority level. Write as: Bit [Ch] = 0: No effect. Use the CLEARCHNLPRIORITY.CHNLS to set channel Ch to the default priority level. Bit [Ch] = 1: Channel Ch uses the high priority level. Writing to a bit where a uDMA channel is not implemented has no effect

15.4.16 CLEARCHNLPRIORITY Register (Offset = 3Ch) [Reset = 00000000h]

CLEARCHNLPRIORITY is shown in Table 15-24.

Return to the Summary Table.

Clear Channel Priority Register.

Table 15-24. CLEARCHNLPRIORITY Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	CHNLS	W	0h	Clear the appropriate bit to select the default priority level for the specified uDMA channel. Write as: Bit [Ch] = 0: No effect. Use the SETCHNLPRIORITY.CHNLS to set channel Ch to the high priority level. Bit [Ch] = 1: Channel Ch uses the default priority level. Writing to a bit where a uDMA channel is not implemented has no effect

15.4.17 ERROR Register (Offset = 4Ch) [Reset = 00000000h]

ERROR is shown in Table 15-25.

Return to the Summary Table.

Error Status and Clear Register.

Table 15-25. ERROR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Reserved
0	STATUS	R/W	0h	Returns the status of bus error flag in uDMA, or clears this bit Read as: 0: No bus error detected 1: Bus error detected Write as: 0: No effect, status of bus error flag is unchanged. 1: Clears the bus error flag.

15.4.18 REQDONE Register (Offset = 504h) [Reset = 00000000h]

REQDONE is shown in Table 15-26.

Return to the Summary Table.

Channel Request Done Register.

Table 15-26. REQDONE Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	CHNLS	R/W	0h	Reflects the uDMA done status for the given channel, channel [Ch]. It's a sticky done bit. Unless cleared by writing a 1, it holds the value of 1. Read as: Bit [Ch] = 0: Request has not completed for channel Ch Bit [Ch] = 1: Request has completed for the channel Ch Writing a 1 to individual bits would clear the corresponding bit. Write as: Bit [Ch] = 0: No effect. Bit [Ch] = 1: The corresponding [Ch] bit is cleared and is set to 0

15.4.19 DONEMASK Register (Offset = 520h) [Reset = 00000000h]

DONEMASK is shown in Table 15-27.

Return to the Summary Table.

Channel Request Done Mask Register.

Table 15-27. DONEMASK Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7-0	CHNLS	R/W	0h	Controls the propagation of the uDMA done and active state to the assigned peripheral. Specifically used for software channels. Read as: Bit [Ch] = 0: uDMA done and active state for channel Ch is not blocked from reaching to the peripherals. Note that the uDMA done state for channel [Ch] is blocked from contributing to generation of combined uDMA done signal Bit [Ch] = 1: uDMA done and active state for channel Ch is blocked from reaching to the peripherals. Note that the uDMA done state for channel [Ch] is not blocked from contributing to generation of combined uDMA done signal Write as: Bit [Ch] = 0: Allows uDMA done and active state to propagate to the peripherals. Note that this disables uDMA done state for channel [Ch] from contributing to generation of combined uDMA done signal Bit [Ch] = 1: Blocks uDMA done and active state to propagate to the peripherals. Note that this enables uDMA done for channel [Ch] to contribute to generation of combined uDMA done signal.

Chapter 16

Advanced Encryption Standard (AES)

This chapter describes the functionality of the advanced encryption standard (AES) system.

16.1 Introduction

The AES accelerator module performs encryption and decryption of 128-bit data blocks with a 128-bit key in hardware according to the Advanced Encryption Standard (AES). AES is a symmetric-key block cipher algorithm specified in FIPS PUB 197. Encryption converts data to an unintelligible form called ciphertext. Decryption converts the ciphertext back into the original form called plaintext.

Features

- The AES supports AES-128 block cipher encryption.
- The AES supports the following cipher modes for encryption: ECB, CBC-MAC, CBC, CTR, CFB, OFB, and PCBC.
- CCM can also be accelerated by the software configuring CBC-MAC and CTR modes accordingly.
- Cipher mode decryption is supported, except for ECB, CBC, and PCBC.
- The module supports AES CTR-DRBG acceleration.
- Data can be routed to and from the module through CPU and µDMA.
- The µDMA interface includes two channels to enable parallel execution of writing and reading data.
- µDMA is capable of feeding the module in all listed cipher modes.
- Capable of completing 256 B CCM using the AES module within 60µs.
- Completes a single AES-128 ECB encryption within 23 CLKSVT clock cycles (approximately 50µs).
- The AES implements hardware features to minimize bus traffic and disturbances to running code.
- The AES key is stored within the AES module, avoiding the need to reload the key for every encryption operation.
- Implements hardware acceleration and configurable increment option for AES-CTR modes
- Supports writing the next block of plaintext during ongoing encryption, which enables parallel processing

16.1.1 AES Performance

The finite State machine (FSM) processes the data in a column-fashioned way, processing two columns/cycle, completing 10 rounds in 20 cycles. With three cycles of preprocessing, the execution/encryption time is 23 cycles or 5.56 bits a cycle.

16.2 Functional Description

The AES accelerator consists of the register interface and the finite state machine (FSM).

The register bank provides various options to the user to configure the plaintext source, encryption triggers, µDMA channel triggers, counter size, endianness, alignment, and actions that clear status events and IRQs. In addition to the key storage register, there are registers for plaintext and buffer. Users can also configure side effects such as XORing, clearing of events or IRQs, and generation of µDMA and AES triggers. Due to these options, external intervention by the CPU or µDMA is kept to a minimum, thereby significantly increasing throughput.

The FSM operates on the input block, performing the required substitution, shift, and mix operations. A new subkey is generated and XORed with the data each round. Round keys are generated on the fly and parallel to data processing. To accommodate CTR cipher mode, the IP offers a 128-bit register acting either as a counter in CTR cipher mode or as a pipeline buffer to enable the update of the next plaintext/ciphertext while AES-128 encryption is ongoing.

Data blocks can be transferred to and from AES through µDMA or CPU.

16.2.1 Reset Considerations

A hardware reset will reset all the registers. After reset, KEY, µDMA, and AUTOCFG registers should be reinitialized.

16.2.2 Interrupt and Event Support

16.2.2.1 Interrupt Events and Requests

The following are the interrupt sources

- Channel A µDMA done
- Channel B µDMA done
- Start of AES operation
- End of AES operation

16.2.2.2 Connection to Event Fabric

The combined OR of the AES.MIS register bits generates an event, which is connected to the SVT/MCU Event Fabric module. See Chapter 4 for more information.

16.2.3 µDMA

µDMA channel numbers four and five are assigned to AES Channels A and B. Each channel has an interface consisting of µDMA request and µDMA done signals. Configure µDMA channels to specify the total transfers. For multi-block encryption, specify the total transfers required for all the blocks. The µDMA request signal is generated by AES to alert µDMA for data transfer. The µDMA done signal from the µDMA indicates the completion of all data transfers.

Note

Channel A has higher priority over Channel B.

16.2.3.1 µDMA Example

Consider AES encryption has to be performed on 4 blocks of 128-bit plaintext which is stored in SRAM. The primary control structure of µDMA Channel 4 which is assigned to AES channel A, can be configured as below, to transfer the plaintext from SRAM to AES.

Transfer size	16 words
Source end pointer	Plaintext memory location in SRAM
Destination end pointer	[DMA.DMACHA]
Source Increment	32 bits
Destination Increment	None
Arbitration size	4 words
Transfer Mode	Basic

µDMA channel 5 which is assigned to AES channel B, can be configured as below, to transfer ciphertext from AES to SRAM

Transfer size	16 words
Source end pointer	[DMA.DMACHB]
Destination end pointer	SRAM memory location to store ciphertext
Source Increment	None
Destination Increment	32 bits
Arbitration size	4 words
Transfer Mode	Basic

Note

The address pointed by [DMA.DMACHA] and [DMA.DMACHB] is automatically incremented within the design after every transfer. Thus, the Destination Increment and Source Increment for AES channel A and channel B, respectively, is configured as None.

Refer to Chapter 15 for more details on configuring µDMA channels.

16.3 Encryption and Decryption Configuration

Each of the following modes can be accelerated either through µDMA or CPU. This section provides sequence of operations which includes initialization and separate steps for µDMA and CPU data transfer.

16.3.1 CBC-MAC (Cipher Block Chaining-Message Authentication Code)

The fastest way to accelerate CBC-MAC is to upload the next plaintext during the current encryption.

Assuming initialization from reset state and denoting plaintext as an array, m[1:x]:

Initialization:

- Write KEY0-KEY3
- AUTOCFG:
  - AESSRC = TXTXBUF
  - TRGAES = WRBUF3 (Write to BUF3 triggers AES)
  - BUSHALT = EN
- Write 0 to TXT0-TXT3

Cipher mode with µDMA:

- µDMA channel A moves m[1:x] into BUF when a new AES operation starts:
  - ADRCHA = BUF0
  - TRGCHA = AESSTART
- START: CPU writes 0x1 to TRG.DMACHA to start CBC-MAC
- END : CPU waits for µDMA done and STA.STATE = IDLE, then reads result/TAG/MIC from TXT0-TXT3

Cipher mode with CPU:

- for i = 1; i<=x, increment i:
  - Write m[i] to BUF
- Wait until STA.STATE = IDLE
- Read result/TAG/MIC from TXT0-TXT3

16.3.2 CBC (Cipher Block Chaining) Encryption

The fastest way to accelerate this cipher mode is to upload the next plaintext during the current encryption.

An Initialization Vector (IV) is used to randomize the encryption so that distinct ciphertexts are produced even if the same plaintext is encrypted multiple times. Assuming initialization from reset state and denoting plaintext as an array, m[1:x]:

Initialization:

- Write KEY0-KEY3
- AUTOCFG
  - AESSRC = TXTXBUF
  - TRGAES = RDTXT3 | WRBUF3S (the first encryption starts by writing BUF3, the successive ones by reading TXT3)
- Write IV to TXT0-TXT3

Cipher mode with µDMA:

Assumes CHA has written new plaintext to BUF before reading ciphertext.

- µDMA channel A moves m[1:x] into BUF when a new encryption starts
  - ADRCHA = BUF0
  - TRGCHA = AESSTART
  - DONEACT = GATE_TRGAES_ON_CHA_DEL (to avoid spurious last AES operation).In case of single block encryption, configure DONEACT = GATE_TRGAES_ON_CHA
- µDMA channel B moves ciphertext[1:x] to memory when AES completes

  - ADRCHB = TXT0
  - TRGCHB = AESDONE
- START: CPU writes 0x1 to TRG.DMACHA to start CBC encryption. µDMA moves N x 16B.
- END : CPU waits for µDMA to signal 'done' and STA.STATE = IDLE, then reads result/TAG/MIC from TXT0-TXT3.

Cipher mode with CPU:

- Writes m[1] to BUF (triggers AES)
- for i = 1; i<x, increment i:
  - Prepares m[i+1]
  - Wait until STA.STATE = BUSY || use interrupt (This can be skipped since AES is immediately triggered)
  - Writes m[i+1] to BUF
  - Waits for STA.STATE = IDLE || use interrupt
  - Reads ciphertext[i] from TXT
- Last m(x):
  - Set AUTOCFG.TRGAES = DISABLE
  - Waits for STA.STATE = IDLE || use interrupt
  - Reads ciphertext[x] from TXT

Note

The loop can run over i=1:x, but then an additional AES encryption is triggered on the last read of TXT3. This can be aborted immediately after completing the x-th iteration of the for-loop.

16.3.3 CBC Decryption

CBC decryption is not supported.

16.3.4 CTR (Counter) Encryption/Decryption

Assuming initialization from reset state and denoting plaintext as an array, m[1:x]:

Initialization:

- Write KEY0-KEY3
- AUTOCFG
  - AESSRC = BUF
  - TRGAES = RDTXT3 | WRBUF3S (the first encryption starts by writing BUF3, the successive ones by reading TXT3)
  - CTRSIZE = CTR8/CTR16/CTR32/CTR64/CTR128

Cipher mode with µDMA:

- µDMA channel A moves m[1:x] into TXTX when AES completes
  - ADRCHA = TXTX0
  - TRGCHA = AESDONE
  - DONEACT = GATE_TRGAES_ON_CHA (to avoid spurious last AES using µDMA)
- µDMA channel B moves ciphertext[1:x] to memory after channel A has written TXTX3
  - ADRCHB = TXT0
  - TRGCHB = WRTXT3
- START: SW initializes BUF with data corresponding to nonce, flags, and counter. This triggers first AES operation
- END : CPU waits for µDMA to signal 'done'

Cipher mode with CPU:

- Initializes BUF with data corresponding to nonce, flags, and counter. This triggers first AES operation
- for i=1; i<x; increment i:
  - Waits for STA.STATE = IDLE || use interrupt

  - Writes plaintext[i] to TXTX0-TXTX3
  - Reads ciphertext[i]
- Set AUTOCFG.TRGAES =DISABLE
- Waits for STA.STATE = IDLE || use interrupt
- Writes plaintext[i] to TXTX0-TXTX3
- Reads ciphertext[i]

Note

The loop can run over i=1:x, but then an additional AES operation is triggered on the last read of TXT3. This can be aborted immediately after completing the x-th iteration of the for-loop.

16.3.5 ECB (Electronic Code Book) Encryption

Assuming initialization from reset state and that CHA has written new plaintext to BUF before reading ciphertext, denoting plaintext as an array, m[1:x]:

Initialization:

- Write KEY0-KEY3
- Set AUTOCFG:
  - AESSRC = BUF
  - TRGAES = RDTXT3 | WRBUF3S

Cipher mode with µDMA:

- µDMA channel A moves m[1:x] into BUF when AES operation starts
  - ADRCHA = BUF0
  - TRGCHA = AESSTART
  - DONEACT = GATE_TRGAES_ON_CHA_DEL (to avoid spurious last AES operation). In case of single block encryption, configure DONEACT = GATE_TRGAES_ON_CHA.
- µDMA channel B moves ciphertext[1:x] to memory when AES completes.
  - ADRCHB = TXT0
  - TRGCHB = AESDONE
- START: CPU writes 0x1 to TRG.DMACHA
- END : CPU waits for µDMA 'done'

Cipher mode with CPU:

- Writes m[1] to BUF (triggers AES operation)
- for i = 1; i<x, increment i:
  - Prepares m[i+1]
  - Wait until STA.STATE = BUSY || use interrupt (This can be skipped since AES is immediately triggered)
  - Writes m[i+1] to BUF
  - Waits for STA.STATE = IDLE || use interrupt
  - Reads ciphertext[i] from TXT
- Last m(x):
  - Set AUTOCFG.TRGAES = DISABLE
  - Waits for STA.STATE = IDLE || use interrupt
  - Reads ciphertext[x] from TXT

Note

The loop can run over i=1:x, but then an additional AES operation is triggered on the last read of TXT3. This can be aborted immediately after completing the x-th iteration of the for-loop.

16.3.6 ECB Decryption

ECB decryption is not supported.

16.3.7 CFB (Cipher Feedback) Encryption

Below are the steps for CFB encryption with 16-byte block size.

Assuming initialization from reset state and denoting plaintext as an array, m[1:x]:

Initialization:

- Write KEY0-KEY3
- SW initializes TXT with (Initialization Vector) IV
- AUTOCFG
  - AESSRC = TXT
  - TRGAES = RDTXT3 (reading TXTXBUF3 also causes trigger, see register information)
  - TRGTXT = RDTXTXBUF3

Cipher mode with µDMA:

Assumes CHA has written new plaintext to BUF before reading ciphertext:

- µDMA channel A moves m[1:x] into BUF when AES starts
  - ADRCHA = BUF0
  - TRGCHA = AESSTART
  - DONEACT = GATE_TRGAES_ON_CHA(to avoid spurious last AES using µDMA)
- µDMA channel B gets ciphertext[1:x] by reading TXTXBUF when AES completes.
  - ADRCHB = TXTXBUF0
  - TRGCHB = AESDONE
- START: CPU writes 'TXT' to TRG.OP to start first AES.
- END: CPU waits for µDMA to signal 'done'.

Cipher mode with CPU:

- Writes 'TXT' to TRG.OP to start first AES.
- for i = 1; i<x, increment i:
  - Waits until STA.STATE = BUSY || use interrupt (This can be skipped since AES is immediately triggered)
  - Writes m[i] to BUF
  - Waits for STA.STATE = IDLE || use interrupt
  - Reads ciphertext[i] from TXTXBUF (TXT <= TXT XOR BUF (what was just read), triggers new encryption)
- Last m(x):
  - Set AUTOCFG.TRGAES = DISABLE
  - Waits until STA.STATE = BUSY || use interrupt (This can be skipped since AES is immediately triggered)
  - Writes m[i] to BUF
  - Waits for STA.STATE = IDLE || use interrupt
  - Reads ciphertext[x] from TXTXBUF

Note

The loop can run over i=1:x, but then an additional AES encryption is triggered on the last read of TXT3. This can be aborted immediately after completing the x-th iteration of the for-loop.

16.3.8 CFB Decryption

Below are the steps for CFB decryption with 16-byte block size.

Assuming initialization from reset state and denoting plaintext as an array, m[1:x]:

Initialization:

- Write KEY0-KEY3
- SW initializes TXT
- AUTOCFG
  - AESSRC = BUF (automatically triggered AES operations use the BUF content as plaintext)

  - TRGAES = RDTXT3 (reading TXTXBUF3 also causes trigger, see register information)

Cipher mode with µDMA:

Assumes CHA has written new plaintext to BUF before reading ciphertext:

- µDMA channel A moves m[1:x] into BUF when AES starts
  - ADRCHA = BUF0
  - TRGCHA = AESSTART
  - DONEACT = GATE_TRGAES_ON_CHA (to avoid spurious last AES using µDMA)
- µDMA channel B gets ciphertext[1:x] by reading TXTXBUF when AES completes
  - ADRCHB = TXTXBUF0
  - TRGCHB = AESDONE
- START: CPU write 'TXT' to TRG.AESOP to trigger first AES(K,TXT)
- END:
  - CPU waits for µDMA to signal 'done'
  - CPU optionally aborts the spurious AES by writing 0x1 to ABORT.AES

Cipher mode with CPU:

CPU implements the µDMA behavior. CPU needs to wait for STA.STATE = BUSY before CPU can update BUF with ciphertext[i].

16.3.9 OFB (Open Feedback) Encryption

Assuming initialization from reset state and denoting plaintext as an array, m[1:x]:

Initialization:

- Write KEY0-KEY3
- SW initializes TXT with IV
- AUTOCFG
  - AESSRC = TXT
  - TRGAES = RDTXT3

Cipher mode with µDMA:

Assumes CHA has written new plaintext to BUF before reading ciphertext:

- µDMA channel A moves m[1:x] into BUF when AES starts
  - ADRCHA = BUF0
  - TRGCHA = AESSTART
  - DONEACT = GATE_TRGAES_ON_CHA (to avoid spurious last AES using µDMA)
- µDMA channel B gets ciphertext[1:x] by reading TXTXBUF when AES completes
  - ADRCHB = TXTXBUF0
  - TRGCHB = AESDONE
- START: SW write 'TXT' to TRG.AESOP to trigger first AES(K,TXT)
- END :
  - CPU waits for µDMA to signal 'done'
  - CPU optionally aborts the spurious AES by writing 0x1 to ABORT.AES

Cipher mode with CPU:

CPU implements the µDMA behavior. CPU needs to wait for STA.STATE = BUSY before CPU can update BUF with ciphertext[i].

16.3.10 OFB Decryption

Assuming initialization from reset state and denoting plaintext as an array, m[1:x].

The same steps are carried out as for encryption, with the exception that plaintext and ciphertext are swapped.

16.3.11 PCBC (Propagating Cipher Block Chaining) Encryption

Assuming initialization from reset state and denoting plaintext as an array, m[1:x]:

Initialization:

- Write KEY0-KEY3
- AUTOCFG
  - AESSRC = TXTXBUF
  - TRGAES = WRBUF3
  - TRGTXT = RDTXT3
- Write IV to TXT

Cipher mode with µDMA:

- µDMA channel A moves m[1:x] into BUF on TXT3 read.
  - ADRCHA = BUF0
  - TRGCHA = RDTXT3
  - DONEACT = GATE_TRGAES_ON_CHA (to avoid spurious last AES using µDMA)
- µDMA channel B gets ciphertext[1:x] by reading TXT when AES completes.
  - ADRCHB = TXT0
  - TRGCHB = AESDONE (Completion of each burst updates TXT <= TXT XOR BUF, and triggers channel A)
- START: SW starts the process by writing 0x1 to TRG.DMACHA
- END: SW waits for µDMA to signal 'done'.

Cipher mode with CPU:

- for i = 1; i<x, increment i:
  - Write m[i] to BUF
  - Waits for STA.STATE = IDLE || use interrupt
  - Reads ciphertext[i] from TXT
- Last m(x):
  - Set AUTOCFG.TRGAES = DISABLE
  - Writes m[x] to BUF
  - Waits for STA.STATE = IDLE || use interrupt
  - Reads ciphertext[x] from TXT

Note

The loop can run over i=1:x, but then an additional AES encryption is triggered on the last read of TXT3. This can be aborted immediately after completing the x-th iteration of the for-loop.

16.3.12 PCBC Decryption

PCBC decryption is not supported.

16.3.13 CTR-DRBG (Counter-Deterministic Random Bit Generator)

Assuming initialization from reset state and denoting plaintext as an array, m[1:x]:

Initialization:

- Write KEY0-KEY3
- Write BUF with data corresponding to nonce, flags, and counter.

Cipher mode with µDMA:

The following can be useful if many random numbers need to be generated and stored:

- AUTOCFG
  - CTRSIZE = CTR8/CTR16/CTR32/CTR64/CTR128

- µDMA channel A triggers blockcipher.
  - µDMA configuration: µDMA CH A shall write TRG.AESOP = BUF all the times, single transfer per arbitration cycle, R=0.
  - TRGCHA = RDTXT3
- µDMA channel B moves ciphertext[1:x]/random numbers to memory after channel A has triggered AES.
  - ADRCHB = TXT0
  - TRGCHB = AESDONE
- START: CPU writes 1 to TRG.DMACHA
- END: CPU waits for RIS.CHBDONE

Cipher mode with CPU:

The following is useful if a single or few random numbers are needed:

Initialization:

AUTOCFG

- CTRSIZE = CTR8/CTR16/CTR32/CTR64/CTR128

Operation:

- CPU writes 1 to TRG.AESOP = BUF
- CPU waits for AES to complete.
- CPU reads the result

16.3.14 CCM

CCM can be realized using 16 + 2(Nonce + Packet and Payload header) blocks of CBC-MAC to calculate authentication block and 16 + 1(Authentication Tag) blocks of CTR to perform encryption/decryption. CTR uses the same 16 blocks of plaintext as CBC-MAC.

16.4 AES Registers

Table 16-1 lists the memory-mapped registers for the AES registers. All register offset addresses not listed in Table 16-1 should be considered as reserved locations and the register contents should not be modified.

Table 16-1. AES Registers

Offset	Acronym	Register Name	Section
0h	DESC	Description Register.	Go
10h	TRG	Trigger	Go
14h	ABORT	Abort	Go
18h	CLR	Clear	Go
1Ch	STA	Status	Go
20h	DMA	Direct Memory Access	Go
24h	DMACHA	DMA Channel A data transfer	Go
28h	DMACHB	DMA Channel B data transfer	Go
2Ch	AUTOCFG	Automatic Configuration	Go
50h	KEY0	Key Word 0	Go
54h	KEY1	Key Word 1	Go
58h	KEY2	Key Word 2	Go
5Ch	KEY3	Key Word 3	Go
70h	TXT0	Text Word 0	Go
74h	TXT1	Text Word 1	Go
78h	TXT2	Text Word 2	Go
7Ch	TXT3	Text Word 3	Go
80h	TXTX0	Text Word 0 XOR	Go
84h	TXTX1	Text Word 1 XOR	Go
88h	TXTX2	Text Word 2 XOR	Go
8Ch	TXTX3	Text Word 3 XOR	Go
90h	BUF0	Buffer Word 0	Go
94h	BUF1	Buffer Word 1	Go
98h	BUF2	Buffer Word 2	Go
9Ch	BUF3	Buffer Word 3	Go
A0h	TXTXBUF0	Text Word 0 XOR Buffer Word 0	Go
A4h	TXTXBUF1	Text Word 1 XOR Buffer Word 1	Go
A8h	TXTXBUF2	Text Word 2 XOR Buffer Word 2	Go
ACh	TXTXBUF3	Text Word 3 XOR Buffer Word3	Go
104h	IMASK	Interrupt Mask register	Go
108h	RIS	Raw Interrupt Status register	Go
10Ch	MIS	Masked Interrupt Status register	Go
110h	ISET	Interrupt Set register	Go
114h	ICLR	Interrupt Clear register	Go
118h	IMSET	Interrupt Mask Set register	Go
11Ch	IMCLR	Interrupt Mask Clear register	Go

Complex bit access types are encoded to fit into small table cells. Table 16-2 shows the codes that are used for access types in this section.

Table 16-2. AES Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
RH	R H	Read Set or cleared by hardware
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

16.4.1 DESC Register (Offset = 0h) [Reset = 6B424010h]

DESC is shown in Table 16-3.

Return to the Summary Table.

Description Register.
This register provides IP module ID, revision information, instance index and standard MMR registers offset.

Table 16-3. DESC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	MODID	R	6B42h	Module Identifier This register is used to uniquely identify this IP.
15-12	STDIPOFF	R	4h	Standard IP MMR block offset Standard IP MMRs are the set from aggregated IRQ registers till DTB. 0: Standard IP MMRs do not exist. 0x1-0xF: Standard IP MMRs begin at offset of 64*STDIPOFF from the base IP address.
11-8	INSTIDX	R	0h	IP Instance ID number If multiple instances of IP exist in the device, this field can identify the instance number (0-15).
7-4	MAJREV	R	1h	Major revision of IP (0-15)
3-0	MINREV	R	0h	Minor Revision of IP(0-15)

16.4.2 TRG Register (Offset = 10h) [Reset = 00000000h]

TRG is shown in Table 16-4.

Return to the Summary Table.

Trigger
This register is used to manually trigger operations.

Table 16-4. TRG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3	DMACHA	W	0h	Manually trigger channel A request 0h = Writing 0 has no effect 1h = Triggers channel A request
2	DMACHB	W	0h	Manually trigger channel B request 0h = Writing 0 has no effect 1h = Triggers channel B request
1-0	AESOP	W	0h	AES Operation Write an enumerated value to this field when STA.STATE = IDLE to manually trigger an AES operation. If condition is not met, the trigger is ignored. Non-enumerated values are ignored. Enumerated value indicates source of AES operation 1h = TXT = AES(KEY,TXT) 2h = TXT = AES(KEY,BUF) 3h = TXT = AES(KEY, TXT XOR BUF)

16.4.3 ABORT Register (Offset = 14h) [Reset = 00000000h]

ABORT is shown in Table 16-5.

Return to the Summary Table.

Abort
This register is used to abort current AES operation.

Table 16-5. ABORT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Reserved
0	ABORTAES	W	0h	Abort AES operation Abort an ongoing AES operation. An abort will clear TXT, BUF, DMA, AUTOCFG registers 0h = Writing 0 has no effect 1h = Aborts an ongoing AES operation

16.4.4 CLR Register (Offset = 18h) [Reset = 00000000h]

CLR is shown in Table 16-6.

Return to the Summary Table.

Clear
This register is used to clear contents of TXT and BUF when STA.STATE = IDLE. If condition is not met, the contents remain unchanged.

Table 16-6. CLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1	TXT	W	0h	Clear TXT 0h = Writing 0 has no effect 1h = Clears TXT
0	BUF	W	0h	Clear BUF 0h = Writing 0 has no effect 1h = Clears BUF

16.4.5 STA Register (Offset = 1Ch) [Reset = 00000000h]

STA is shown in Table 16-7.

Return to the Summary Table.

Status
This register provides information on AES accellerator state and BUF status.

Table 16-7. STA Register Field Descriptions

Bit	Field	Type	Reset	Description
31-2	RESERVED	R	0h	Reserved
1	BUFSTA	R	0h	BUF Status Field gives the status of BUF, indicating EMPTY or FULL, when AUTOCFG.TRGAES = WRBUF3. If AUTOCFG.TRGAES != WRBUF3, then STA.BUFSTA will hold the value 0. Note : Useful for CBC-MAC 0h = Data stored in BUF is already consumed by the AES engine and next block of data can be written in BUF. 1h = Data stored in BUF is not yet consumed by the AES engine. Next block of data cannot be written into BUF until STA.STATE = IDLE.
0	STATE	R	0h	State Field gives the state of the AES engine. 0h = AES engine is IDLE 1h = AES operation active

16.4.6 DMA Register (Offset = 20h) [Reset = 00000000h]

DMA is shown in Table 16-8.

Return to the Summary Table.

Direct Memory Access
This register controls the conditions that will generate burst requests on each DMA channel.

Table 16-8. DMA Register Field Descriptions

Bit	Field	Type	Reset	Description
31-20	RESERVED	R	0h	Reserved
19-16	DONEACT	R/W	0h	Done Action This field determines the side effects of DMA done. It is allowed to configure this field with an OR-combination of supported enums, with the exception that GATE_TRGAES_ON_CHA and GATE_TRGAES_ON_CHA_DEL must be mutually exclusive 0h = DMA done has no side effect 1h = Triggers defined in AUTOCFG.TRGAES are gated when RIS.CHADONE = SET 2h = Delayed gating of triggers defined in AUTOCFG.TRGAES Due to the pipelining of BUF writes, in certain modes, DMA CHA Done appears before the last but one AES operation has completed. Setting this bit, will gate the triggers defined in AUTOCFG.TRGAES only after the last write by CHA is consumed by AES FSM. Used in ECB,CBC,CBC-MAC modes (having multiple blocks encryption/ decryption) to avoid spurious AES operation triggered on last read by CHB. For single mode operation, DMA.GATE_TRGAES_ON_CHA must be used. 4h = DMA channel A done event clears TXT0 thru TXT3 if STA.STATE = IDLE. Event is ignored if condition is not met. 8h = DMA channel B done event clears TXT0 thru TXT3 if STA.STATE = IDLE. Event is ignored if condition is not met.
15-14	RESERVED	R	0h	Reserved
13-12	ADRCHB	R/W	0h	Channel B Read Write Address The DMA accesses DMACHB to read or write contents of TXT and BUF as a response to a burst request. This field specifes the start address of the first DMA transfer that follows the burst request. The internal address gets incremented automatically for subsequent accesses. The DMA can transfer 8-bit, 16-bit, or 32-bit words, and must always complete a 16-byte transfer before re-arbitration. 0h = Start address is TXT0 1h = Start address is TXTX0 2h = Start address is BUF0 3h = Start address is TXTXBUF0
11	RESERVED	R	0h	Reserved
10-8	TRGCHB	R/W	0h	Channel B Trigger Select the condition that triggers DMA channel B request. Non-enumerated values are not supported and ignored. 0h = DMA requests are disabled 1h = Start of AES operation triggers request 2h = Completion of AES operation triggers request 3h = Writes to TXT3, TXTX3, or TXTXBUF3 trigger request 4h = Reads of TXT3, or TXTXBUF3 trigger request
7-6	RESERVED	R	0h	Reserved

Table 16-8. DMA Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
5-4	ADRCHA	R/W	0h	Channel A Read Write Address The DMA accesses DMACHA to read or write contents of TXT and BUF as a response to a burst request. This field specifes the start address of the first DMA transfer that follows the burst request. The internal address gets incremented automatically for subsequent accesses. The DMA can transfer 8-bit, 16-bit, or 32-bit words, and must always complete a 16-byte transfer before re-arbitration. 0h = Start address is TXT0 1h = Start address is TXTX0 2h = Start address is BUF0 3h = Start address is TXTXBUF0
3	RESERVED	R	0h	Reserved
2-0	TRGCHA	R/W	0h	Channel A Trigger Select the condition that triggers DMA channel A request. Non-enumerated values are not supported and ignored. 0h = DMA requests are disabled 1h = Start of AES operation triggers request 2h = Completion of AES operation triggers request 3h = Writes to TXT3 or TXTX3 trigger request 4h = Reads of TXT3 or TXTXBUF3 trigger request

16.4.7 DMACHA Register (Offset = 24h) [Reset = 00000000h]

DMACHA is shown in Table 16-9.

Return to the Summary Table.

DMA Channel A data transfer
DMA accesses this register to read or write contents from sequential addresses specifed by DMA.ADRCHA.

Table 16-9. DMACHA Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R/W	0h	Value transferred through DMA Channel A

16.4.8 DMACHB Register (Offset = 28h) [Reset = 00000000h]

DMACHB is shown in Table 16-10.

Return to the Summary Table.

DMA Channel B data transfer
DMA accesses this register to read or write contents from sequential addresses specifed by DMA.ADRCHB.

Table 16-10. DMACHB Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R/W	0h	Value transferred through DMA Channel B

16.4.9 AUTOCFG Register (Offset = 2Ch) [Reset = 00000000h]

AUTOCFG is shown in Table 16-11.

Return to the Summary Table.

Automatic Configuration
This register configures automatic hardware updates to TXT and BUF. Configure this register to reduce software overhead during cipher modes.

Table 16-11. AUTOCFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-29	RESERVED	R	0h	Reserved
28	CHBDNCLR	R/W	0h	This field enable auto-clear of RIS.CHBDONE interrupt on read/write of TXT3/BUF3/TXTX3/TXTXBUF3 . 0h = Disable auto-clear of RIS.CHBDONE interrupt 1h = Enable auto-clear of RIS.CHBDONE interrupt
27	CHADNCLR	R/W	0h	This field enables auto-clear of RIS.CHADONE interrupt on read/ write of TXT3/BUF3/TXTX3/TXTXBUF3 . 0h = Disable auto-clear of RIS.CHADONE interrupt 1h = Enable auto-clear of RIS.CHADONE interrupt
26	CLRAESST	R/W	0h	Clear AES Start This field enables auto-clear of RIS.AESSTART interrupt on read/ write of TXT3/BUF3/TXTX3/TXTXBUF3 . 0h = Disable auto-clear of RIS.AESSTART interrupt 1h = Enable auto-clear of RIS.AESSTART interrupt
25	CLRAESDN	R/W	0h	Clear AES Done This field enables auto-clear of RIS.AESDONE interrupt on read/ write of TXT3/BUF3/TXTX3/TXTXBUF3 . 0h = Disable auto-clear of RIS.AESDONE interrupt 1h = Enable auto-clear of RIS.AESDONE interrupt
24	BUSHALT	R/W	0h	Bus Halt This field decides if bus halts on access to KEY, TXT, BUF, TXTX and TXTXBUF when STA.STATE = BUSY. 0h = Disable bus halt When STA.STATE = BUSY, writes to KEY, TXT, TXTX are ignored, reads from TXT, TXTXBUF return zero. When STA.STATE = BUSY and if either STA.BUFSTA = FULL or AUTOCFG.CTRSIZE != DISABLE, writes to BUF are ignored, reads return zero. 1h = Enable bus halt When STA.STATE = BUSY, access to KEY, TXT, TXTX, TXTXBUF halt the bus until STA.STATE = IDLE. When STA.STATE = BUSY and if either STA.BUFSTA = FULL or AUTOCFG.CTRSIZE != DISABLE, access to BUF halts the bus until STA.STATE = IDLE.
23-22	RESERVED	R	0h	Reserved
21-19	CTRSIZE	R/W	0h	Counter Size Configures size of counter as either 8,16,32,64 or 128 Non-enumerated values are not supported and ignored 0h = Disable CTR operation 1h = Configures counter size as 8-bit 2h = Configures counter size as 16-bit 3h = Configures counter size as 32-bit 4h = Configures counter size as 64-bit 5h = Configures counter size as 128-bit

Table 16-11. AUTOCFG Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
18	CTRALIGN	R/W	0h	Counter Alignment Specifies alignment of counter 0h = Indicates Left Aligned Counter Not applicable for 128-bit counter size. For 128-bit counter, all octets will be considered When left aligned,,octet 0-7 will be considered , based on counter size and endianness 1h = Indicates right aligned counter Not applicable when counter size is 128-bit For 128-bit counter, all octets will be considered If right aligned, octet 8-15 will be considered based on endianness and counter size
17	CTRENDN	R/W	0h	Counter Endianness Specifies Endianness of counter 0h = Specifies Little Endian Counter Carry will flow from octet 'n' to octet 'n+1' 1h = Specifies Big Endian Counter Carry will flow from octet 'n' to octet 'n-1'
16-10	RESERVED	R	0h	Reserved
9-8	TRGTXT	R/W	0h	Trigger for TXT This field determines if and when hardware automatically XORs BUF into TXT. Non-enumerated values are not supported and ignored. It is allowed to configure this field with an OR-combination of supported enums. 0h = No hardware update of TXT 1h = Hardware XORs content of BUF into TXT upon read of TXT3 2h = Hardware XORs content of BUF into TXT upon read of TXTXBUF3
7-6	RESERVED	R	0h	Reserved
5-4	AESSRC	R/W	0h	AES Source This field specifies the data source to hardware-triggered AES operations. Non-enumerated values are not supported and ignored. 1h = TXT = AES(KEY,TXT) 2h = TXT = AES(KEY,BUF) 3h = TXT = AES(KEY, TXT XOR BUF)
3-0	TRGAES	RH/W	0h	Trigger Electronic Codebook This field specifies one or more actions that indirectly trigger AES operation. It is allowed to configure this field with an OR-combination of supported enums. 0h = No user action indirectly triggers AES operation 1h = All writes to TXT3 or TXTX3 trigger action, only when STA.STATE = IDLE 2h = All reads of TXT3 or TXTXBUF3 trigger action, only when STA.STATE = IDLE 4h = All writes to BUF3 will schedule to trigger action once STA.STATE is or becomes IDLE, only when AUTOCFG.CTRSIZE = DIS 8h = Write to BUF3 will schedule to trigger single action once STA.STATE is or becomes IDLE. Subsequent writes do not trigger action unless this setting is written again to this field.

16.4.10 KEY0 Register (Offset = 50h) [Reset = 00000000h]

KEY0 is shown in Table 16-12.

Return to the Summary Table.

Key Word 0
Write KEY0 through KEY3 to populate the 128-bit key. The key is not consumed by the hardware. It is hence not required to reload the key for subsequent block encryptions/decryptions unless required by the application.

Table 16-12. KEY0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	W	0h	Value of KEY[31:0]

16.4.11 KEY1 Register (Offset = 54h) [Reset = 00000000h]

KEY1 is shown in Table 16-13.

Return to the Summary Table.

Key Word 1
Write KEY0 through KEY3 to populate the 128-bit key. The key is not consumed by the hardware. It is hence not required to reload the key for subsequent block encryptions/decryptions unless required by the application.

Table 16-13. KEY1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	W	0h	Value of KEY[63:32]

16.4.12 KEY2 Register (Offset = 58h) [Reset = 00000000h]

KEY2 is shown in Table 16-14.

Return to the Summary Table.

Key Word 2
Write KEY0 through KEY3 to populate the 128-bit key. The key is not consumed by the hardware. It is hence not required to reload the key for subsequent block encryptions/decryptions unless required by the application.

Table 16-14. KEY2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	W	0h	Value of KEY[95:64]

16.4.13 KEY3 Register (Offset = 5Ch) [Reset = 00000000h]

KEY3 is shown in Table 16-15.

Return to the Summary Table.

Key Word 3
Write KEY0 through KEY3 to populate the 128-bit key. The key is not consumed by the hardware. It is hence not required to reload the key for subsequent block encryptions/decryptions unless required by the application.

Table 16-15. KEY3 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	W	0h	Value of KEY[127:96]

16.4.14 TXT0 Register (Offset = 70h) [Reset = 00000000h]

TXT0 is shown in Table 16-16.

Return to the Summary Table.

Text Word 0
TXT is the 128-bit buffer, the AES-128 algorithm performs its operations on. AES input can be written to TXT, and ciphertext can be read from TXT.

Table 16-16. TXT0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	RH/W	0h	Value of TXT[31:0]

16.4.15 TXT1 Register (Offset = 74h) [Reset = 00000000h]

TXT1 is shown in Table 16-17.

Return to the Summary Table.

Text Word 1
TXT is the 128-bit buffer, the AES-128 algorithm performs its operations on. AES input can be written to TXT, and ciphertext can be read from TXT.

Table 16-17. TXT1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	RH/W	0h	Value of TXT[63:32]

16.4.16 TXT2 Register (Offset = 78h) [Reset = 00000000h]

TXT2 is shown in Table 16-18.

Return to the Summary Table.

Text Word 2
TXT is the 128-bit buffer, the AES-128 algorithm performs its operations on. AES input can be written to TXT, and ciphertext can be read from TXT.

Table 16-18. TXT2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	RH/W	0h	Value of TXT[95:64]

16.4.17 TXT3 Register (Offset = 7Ch) [Reset = 00000000h]

TXT3 is shown in Table 16-19.

Return to the Summary Table.

Text Word 3
TXT is the 128-bit buffer, the AES-128 algorithm performs its operations on. AES input can be written to TXT, and ciphertext can be read from TXT.

Table 16-19. TXT3 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	RH/W	0h	Value of TXT[127:96] AUTOCFG.TRGAES decides if a write to or a read of this field triggers an AES operation.

16.4.18 TXTX0 Register (Offset = 80h) [Reset = 00000000h]

TXTX0 is shown in Table 16-20.

Return to the Summary Table.

Text Word 0 XOR
Write data to this register to XOR data with contents in TXT0.VAL.

Table 16-20. TXTX0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	W	0h	Value in TXT0 will be TXT0.VAL = VAL XOR TXT0.VAL

16.4.19 TXTX1 Register (Offset = 84h) [Reset = 00000000h]

TXTX1 is shown in Table 16-21.

Return to the Summary Table.

Text Word 1 XOR
Write data to this register to XOR data with contents in TXT1.VAL.

Table 16-21. TXTX1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	W	0h	Value in TXT1 will be TXT1.VAL = VAL XOR TXT1.VAL

16.4.20 TXTX2 Register (Offset = 88h) [Reset = 00000000h]

TXTX2 is shown in Table 16-22.

Return to the Summary Table.

Text Word 2 XOR
Write data to this register to XOR data with contents in TXT2.VAL.

Table 16-22. TXTX2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	W	0h	Value in TXT2 will be TXT2.VAL = VAL XOR TXT2.VAL

16.4.21 TXTX3 Register (Offset = 8Ch) [Reset = 00000000h]

TXTX3 is shown in Table 16-23.

Return to the Summary Table.

Text Word 3 XOR
Write data to this register to XOR data with contents in TXT3.VAL.
AUTOCFG.TRGAES decides if a write to or a read of this field triggers an AES operation.

Table 16-23. TXTX3 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	W	0h	Value in TXT3 will be TXT3.VAL = VAL XOR TXT3.VAL

16.4.22 BUF0 Register (Offset = 90h) [Reset = 00000000h]

BUF0 is shown in Table 16-24.

Return to the Summary Table.

Buffer Word 0
BUF is a 128-bit auxiliary register that functions as a buffer, counter, or storage of operations in cipher modes.

Table 16-24. BUF0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	RH/W	0h	Value of BUF[31:0]

16.4.23 BUF1 Register (Offset = 94h) [Reset = 00000000h]

BUF1 is shown in Table 16-25.

Return to the Summary Table.

Buffer Word 1
BUF is a 128-bit auxiliary register that functions as a buffer, counter, or storage of operations in cipher modes.

Table 16-25. BUF1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	RH/W	0h	Value of BUF[63:32]

16.4.24 BUF2 Register (Offset = 98h) [Reset = 00000000h]

BUF2 is shown in Table 16-26.

Return to the Summary Table.

Buffer Word 2
BUF is a 128-bit auxiliary register that functions as a buffer, counter, or storage of operations in cipher modes.

Table 16-26. BUF2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	RH/W	0h	Value of BUF[95:64]

16.4.25 BUF3 Register (Offset = 9Ch) [Reset = 00000000h]

BUF3 is shown in Table 16-27.

Return to the Summary Table.

Buffer Word 3
BUF is a 128-bit auxiliary register that functions as a buffer, counter, or storage of operations in cipher modes. AUTOCFG.TRGAES decides if a write to this field triggers an AES operation.

Table 16-27. BUF3 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	RH/W	0h	Value of BUF[127:96]

16.4.26 TXTXBUF0 Register (Offset = A0h) [Reset = 00000000h]

TXTXBUF0 is shown in Table 16-28.

Return to the Summary Table.

Text Word 0 XOR Buffer Word 0
Read this register to obtain plaintext during CFB decryption.

Table 16-28. TXTXBUF0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R	0h	Value read will be TXT0.VAL XOR BUF0.VAL

16.4.27 TXTXBUF1 Register (Offset = A4h) [Reset = 00000000h]

TXTXBUF1 is shown in Table 16-29.

Return to the Summary Table.

Text Word 1 XOR Buffer Word 1
Read this register to obtain plaintext during CFB decryption.

Table 16-29. TXTXBUF1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R	0h	Value read will be TXT1.VAL XOR BUF1.VAL

16.4.28 TXTXBUF2 Register (Offset = A8h) [Reset = 00000000h]

TXTXBUF2 is shown in Table 16-30.

Return to the Summary Table.

Text Word 2 XOR Buffer Word 2
Read this register to obtain plaintext during CFB decryption.

Table 16-30. TXTXBUF2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R	0h	Value read will be TXT2.VAL XOR BUF2.VAL

16.4.29 TXTXBUF3 Register (Offset = ACh) [Reset = 00000000h]

TXTXBUF3 is shown in Table 16-31.

Return to the Summary Table.

Text Word 3 XOR Buffer Word3
Read this register to obtain plaintext during CFB decryption.

Table 16-31. TXTXBUF3 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	VAL	R	0h	Value read will be TXT3.VAL XOR BUF3.VAL

16.4.30 IMASK Register (Offset = 104h) [Reset = 00000000h]

IMASK is shown in Table 16-32.

Return to the Summary Table.

Interrupt Mask register

Table 16-32. IMASK Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3	CHBDONE	R/W	0h	DMA Channel B Done interrupt mask 0h = Disable interrupt mask 1h = Enable interrupt mask
2	CHADONE	R/W	0h	DMA Channel A Done interrupt mask 0h = Disable interrupt mask 1h = Enable interrupt mask
1	AESSTART	R/W	0h	AES Start interrupt mask 0h = Disable interrupt mask 1h = Enable interrupt mask
0	AESDONE	R/W	0h	AES Done interrupt mask 0h = Disable interrupt mask 1h = Enable interrupt mask

16.4.31 RIS Register (Offset = 108h) [Reset = 00000000h]

RIS is shown in Table 16-33.

Return to the Summary Table.

Raw Interrupt Status register

Table 16-33. RIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3	CHBDONE	R	0h	Raw Interrupt Status for DMA Channel B Done 0h = Interrupt did not occur 1h = Interrupt occurred
2	CHADONE	R	0h	Raw Interrupt Status for DMA Channel A Done 0h = Interrupt did not occur 1h = Interrupt occurred
1	AESSTART	R	0h	Raw Interrupt Status for AES Start 0h = Interrupt did not occur 1h = Interrupt occurred
0	AESDONE	R	0h	Raw Interrupt Status for AES Done 0h = Interrupt did not occur 1h = Interrupt occurred

16.4.32 MIS Register (Offset = 10Ch) [Reset = 00000000h]

MIS is shown in Table 16-34.

Return to the Summary Table.

Masked Interrupt Status register

Table 16-34. MIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3	CHBDONE	R	0h	Masked Interrupt Status for DMA Channel B Done 0h = Interrupt did not occur 1h = Interrupt occurred
2	CHADONE	R	0h	Masked Interrupt Status for DMA Channel A Done 0h = Interrupt did not occur 1h = Interrupt occurred
1	AESSTART	R	0h	Masked Interrupt Status for AES Start 0h = Interrupt did not occur 1h = Interrupt occurred
0	AESDONE	R	0h	Masked Interrupt Status for AES Done 0h = Interrupt did not occur 1h = Interrupt occurred

16.4.33 ISET Register (Offset = 110h) [Reset = 00000000h]

ISET is shown in Table 16-35.

Return to the Summary Table.

Interrupt Set register

Table 16-35. ISET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3	CHBDONE	W	0h	Set DMA Channel B Done interrupt 0h = Writing 0 has no effect 1h = Set interrupt
2	CHADONE	W	0h	Set DMA Channel A Done interrupt 0h = Writing 0 has no effect 1h = Set interrupt
1	AESSTART	W	0h	Set AES Start interrupt 0h = Writing 0 has no effect 1h = Set interrupt
0	AESDONE	W	0h	Set AES Done interrupt 0h = Writing 0 has no effect 1h = Set interrupt

16.4.34 ICLR Register (Offset = 114h) [Reset = 00000000h]

ICLR is shown in Table 16-36.

Return to the Summary Table.

Interrupt Clear register

Table 16-36. ICLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3	CHBDONE	W	0h	Clear DMA Channel B Done interrupt 0h = Writing 0 has no effect 1h = Clear interrupt
2	CHADONE	W	0h	Clear DMA Channel A Done interrupt 0h = Writing 0 has no effect 1h = Clear interrupt
1	AESSTART	W	0h	Clear AES Start interrupt 0h = Writing 0 has no effect 1h = Clear interrupt
0	AESDONE	W	0h	Clear AES Done interrupt 0h = Writing 0 has no effect 1h = Clear interrupt

16.4.35 IMSET Register (Offset = 118h) [Reset = 00000000h]

IMSET is shown in Table 16-37.

Return to the Summary Table.

Interrupt Mask Set register

Table 16-37. IMSET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3	CHBDONE	W	0h	Set DMA Channel B Done interrupt mask 0h = Writing 0 has no effect 1h = Set interrupt mask
2	CHADONE	W	0h	Set DMA Channel A Done interrupt mask 0h = Writing 0 has no effect 1h = Set interrupt mask
1	AESSTART	W	0h	Set AES Start interrupt mask 0h = Writing 0 has no effect 1h = Set interrupt mask
0	AESDONE	W	0h	Set AES Done interrupt mask 0h = Writing 0 has no effect 1h = Set interrupt mask

16.4.36 IMCLR Register (Offset = 11Ch) [Reset = 00000000h]

IMCLR is shown in Table 16-38.

Return to the Summary Table.

Interrupt Mask Clear register

Table 16-38. IMCLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3	CHBDONE	W	0h	Clear DMA Channel B Done interrupt mask 0h = Writing 0 has no effect 1h = Clear interrupt mask
2	CHADONE	W	0h	Clear DMA Channel A Done interrupt mask 0h = Writing 0 has no effect 1h = Clear interrupt mask
1	AESSTART	W	0h	Clear AES Start interrupt mask 0h = Writing 0 has no effect 1h = Clear interrupt mask
0	AESDONE	W	0h	Clear AES Done interrupt mask 0h = Writing 0 has no effect 1h = Clear interrupt mask

Chapter 17

Analog to Digital Converter (ADC)

This chapter describes the functionality of the analog-to-digital converter (ADC) module.

17.1 Overview

The purpose of the ADC is to measure analog signals and convert them to a digital representation with minimal CPU intervention providing for lower power and greater task integration.

The ADC supports fast 12-, 10-, and 8-bit analog-to-digital conversions. The ADC implements a 12-bit Successive Approximation Register (SAR) core, sample/conversion mode control, and up to 4 independent conversion-and-control buffers. This means the ADC allows up to four independent analog-to-digital converter (ADC) samples to be converted and stored without any CPU intervention.

ADC features include:

- 1Msps conversion rate at a resolution of 12 bits when reference is external or supply (VDDS)
- 200ksps sampling rate with internal reference
- Full scale ADC operating voltage range
- 12-bit max resolution with support for 10-bit and 8-bit lower resolution modes
- Sample-and-hold with programmable sampling periods controlled by software or timers
- Two sampling trigger sources: software trigger and event trigger
- Software-selectable on-chip reference voltage of 1.4V or 2.5V
- Configurable ADC reference source: VDDS, internal reference (VREF), or external reference (VREF+/-)
- Up to 16 individually configurable analog input channels
- Internal conversion channels for temperature sensing, supply monitoring, and analog signal chain (see device-specific data sheet for availability and channel mapping)
- Configurable ADC clock source
- Different conversion modes: Single-channel, repeat-single-channel, sequence, repeat-sequence, and software requested ad-hoc single conversion modes
- Four 16-bit conversion-result storage registers (MEMRES0:3)
- Support for FIFO and non-FIFO modes for CPU and µDMA
- Data compaction within FIFO for 32-bit reads
- Window comparator with provision to configure low and high threshold values for low-power monitoring of input signals from conversion-result registers
- µDMA support with interrupt event generation on completion of transfer
- Automatic and manual power down schemes
- Unsigned binary and two's complement data format
- 10-bit sample timer with two independent sample time compare registers
- Sample time compare value selection in each memory control register
- Provision to enable window comparator in each memory control register
- Auto-next or trigger-next configuration for sequence or repeated sequence of channels operation
- Different event sources with single event output
- µDMA trigger logic and interface to work with µDMA

Figure 17-1 shows the functional block diagram of the ADC peripheral.

17.2 Block Diagram

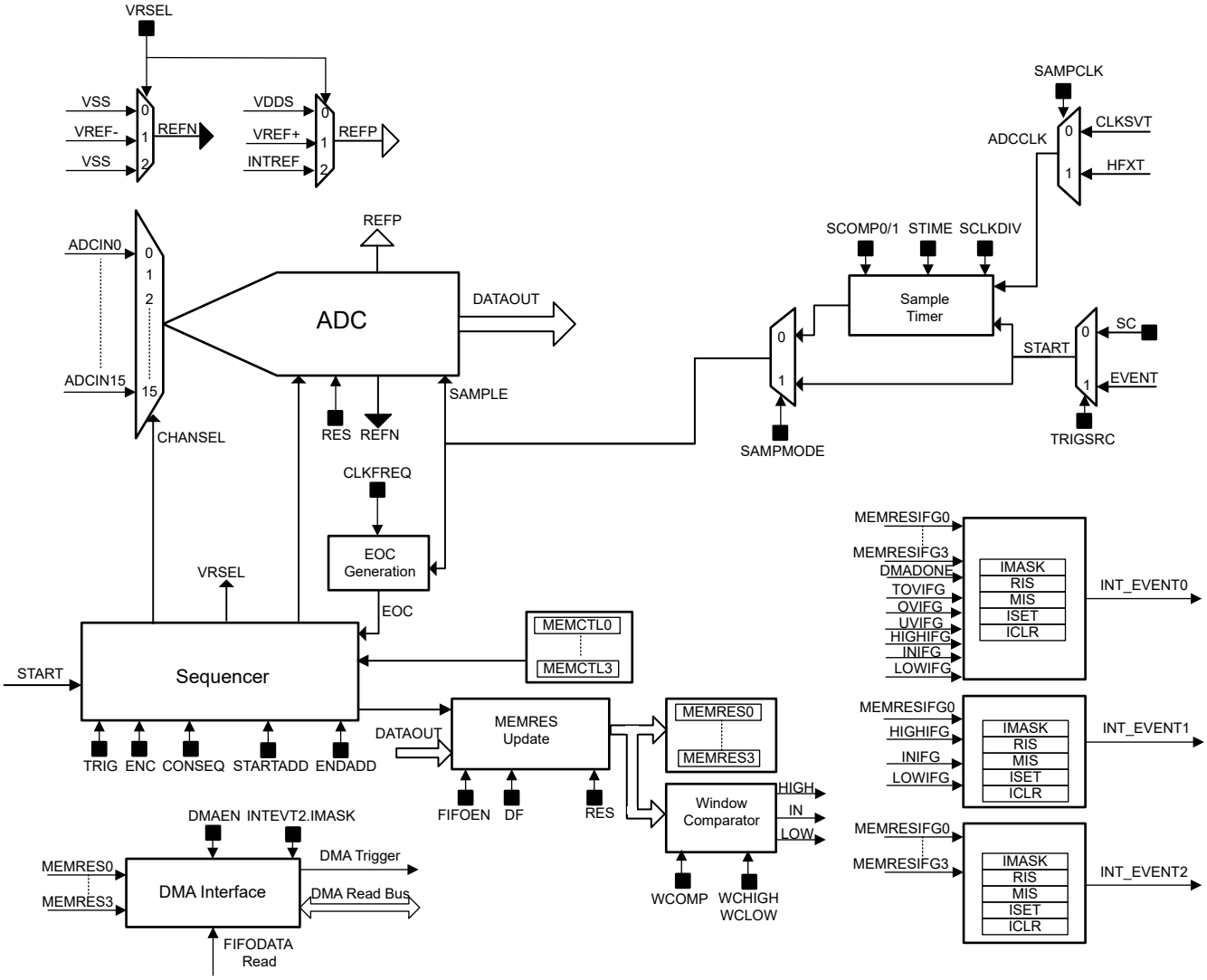


Figure 17-1. ADC Block Diagram

17.3 Functional Description

The ADC is configured with user software. The following sections describe the setup and operation of the ADC.

17.3.1 ADC Core

The ADC core converts an analog input to a digital representation. The core uses two voltage levels (V_{R+} and V_{R-}) to define the upper and lower limits of the conversion. The digital output (N_{ADC}) is full scale when the input signal is equal to or higher than V_{R+}, and is zero when the input signal is equal to or lower than V_{R-}. The input channel and the positive reference voltage level (V_{R+}) are defined in the conversion-control memory.

Equation 1 shows the conversion formula for the ADC result, N_{ADC}, for n-bit resolution mode.

$$N_{ADC} = \left(2^n - 1\right) \times \frac{(V_{in} + 0.5LSB) - V_{R-}}{V_{R+} - V_{R-}} \qquad Where\ LSB = \frac{V_{R+} - V_{R-}}{2^n} \tag{1}$$

Given that V_{R-} is 0 V in this ADC, the equation for N_{ADC} becomes:

$$N_{ADC} = \left(2^n - 1\right) \times \frac{V_{in} + 0.5LSB}{V_{R+}}, \qquad Where\ LSB = \frac{V_{R+}}{2^n} \tag{2}$$

Equation 3 describes the input voltage at which the ADC output saturates:

$$V_{in} = V_{R+} - 1.5LSB \tag{3}$$

Note

The ADC is not functional in standby or shutdown modes.

17.3.2 Voltage Reference Options

The ADC voltage reference (V_{R+}) can be configured through the VRSEL bits in the MEMCTL register. Different reference sources can be selected for conversion on different channels. There are three options available for supplying a reference voltage to the ADC:

1. External reference supplied to the ADC through the AREF+/- pins
2. Supply voltage (VDDS)
3. Configurable internal reference voltage of 1.4V and 2.5V (VREF)

When supplying an external reference to the ADC, the AREF+ pin is connected to the reference source with a 2.2µF decoupling capacitor and the AREF- pin is connected to ground.

17.3.3 Resolution Modes

The ADC supports operation in 12-bit (default), 10-bit, and 8-bit resolution modes. The resolution mode is configured using the ADC.CTL2[2:1] RES bit field.

- When 12-bit mode is selected, the conversion phase requires a total of 14 conversion clock cycles.
- When 10-bit mode is selected, the conversion phase requires a total of 12 conversion clock cycles.
- When 8-bit mode is selected, the conversion phase requires a total of nine conversion clock cycles.

The conversion window is based on the resolution mode and the frequency of ADCCLK. For more details, refer to Figure 17-2 and Figure 17-3.

17.3.4 ADC Clocking

The ADC peripheral clock (ADCCLK) is provided by CKMD and is used for the sampling clock (SAMPCLK). CLKSVT and HFXT are the clock sources available for ADCCLK, which can support up to 48MHz. Refer to the device-specific data sheet for supported ADCCLK frequencies. Using CLKSVT, which is the bus clock for all peripherals, is very useful for the deterministic start of sampling and simultaneous sampling. Using the HFXT as the clock source for ADCCLK is useful for when a very accurate, low-jitter, sampling period is needed. The ADC clock source can be selected by writing the CKMD.ADCCLKSEL[1:0] SRC bit field. The conversion clock is sourced from the selected ADCCLK within the digital hardware.

17.3.5 Power-Down Behavior

To save power, disable the ADC when not in use. The PWRDN bit in the CTL0 register selects the ADC power-down policy between AUTO and MANUAL.

Configure PWRDN based on the max ADC sampling rate required and the operational needs in different power modes.

The reset value of PWRDN is ‘0,’ which has the default behavior of automatic power down of the ADC peripheral at the end of a conversion and when the next sample signal is not required to be asserted immediately. When the PWRDN bit is set to ‘1’ the bit selects manual power-down behavior. In this setting, the ADC is not powered down at the end of a conversion and remains enabled.

Refer to the device-specific data sheet for specifications on the ADC wakeup and enable time.

17.3.6 Sampling Trigger Sources and Sampling Modes

Sample Triggers

There are two sampling trigger sources available that can be selected through the TRIGSRC bit in the CTL1 register; one is a software trigger and the other is an event trigger.

When the software trigger is selected as the source, the application software can set the Start Conversion (SC) bit in the CTL1 register to initiate the sample phase. When the event trigger is selected as the source, a rising edge on the selected event from the event manager initiates the sample phase. An event is always edge-triggered.

Sampling Modes

There are two sampling modes available, AUTO and MANUAL, which are selected through the SAMPMODE bit in the CTL1 register.

17.3.6.1 AUTO Sampling Mode

In AUTO mode, the sample signal is generated synchronous to the sampling clock (SAMPCLK) and can be programmed using an internal sampling timer to determine the duration of the sampling window. The sample timer is 10-bit wide and there are two sample time compare registers (SCOMPx) available to account for various source impedances to measure signals from. One of these two SCOMP registers can be selected using the STIME bit in the MEMCTL register.

Figure 17-2 shows the ADC sample and conversion timing diagram when the ADC is configured in AUTO sampling mode.

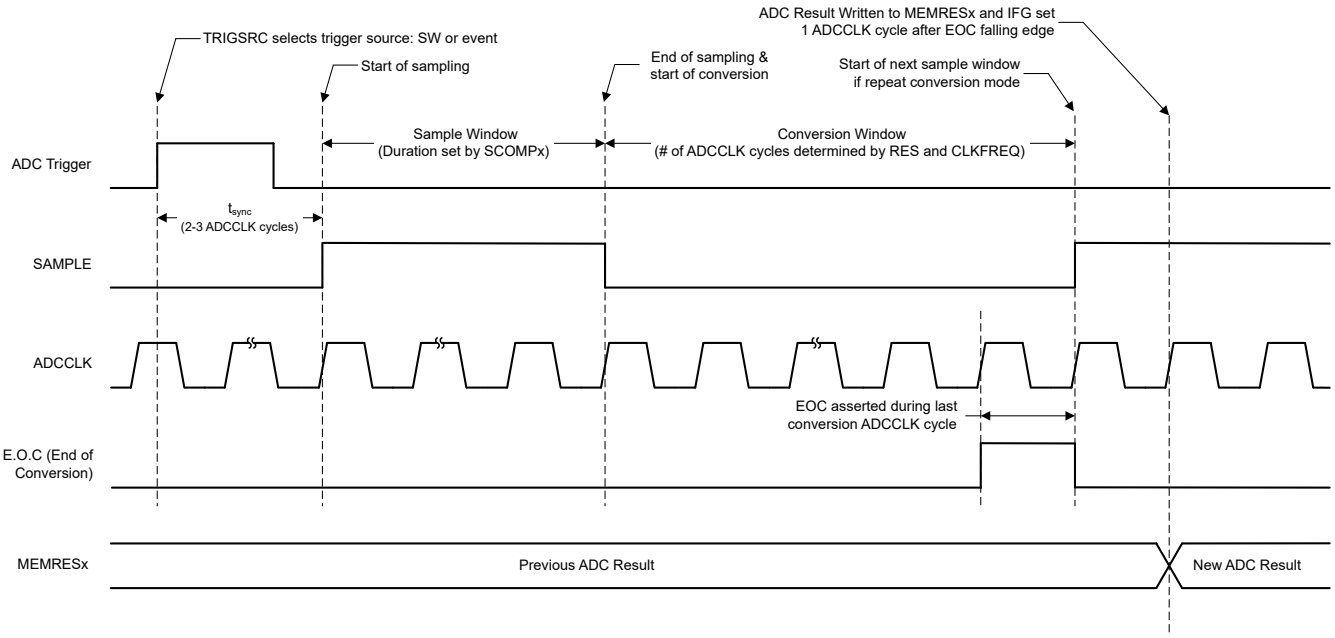


Figure 17-2. AUTO Sampling Mode—ADC Sample and Conversion Timing Diagram

Note

When the reset value of PWRDN is set as '0,' which has the default behavior of automatic power down, the ADC wake-up time must be considered in each sample window. Refer to the device-specific data sheet for specifications on the ADC wake-up time. For example, if the maximum ADC wake-up time is 5µs, then the duration set by SCOMPx is > (5µs + duration for the sample window).

17.3.6.2 MANUAL Sampling Mode

In MANUAL mode, the sample signal is generated when the SC bit is set, which can be asynchronous to the sampling clock. The duration of the sampling window is controlled by software by holding the SC bit high.

Because an event is always edge-triggered, manual mode with event trigger is not supported for any of the conversion modes. Software trigger with manual sampling mode is supported only for single channel single conversion mode and is not supported for any of the other three conversion modes.

There is a 2-3 cycle synchronization latency from when the sample window ends to when the conversion window begins.

Figure 17-3 shows the ADC sample and conversion timing diagram when the ADC is configured in MANUAL sampling mode:

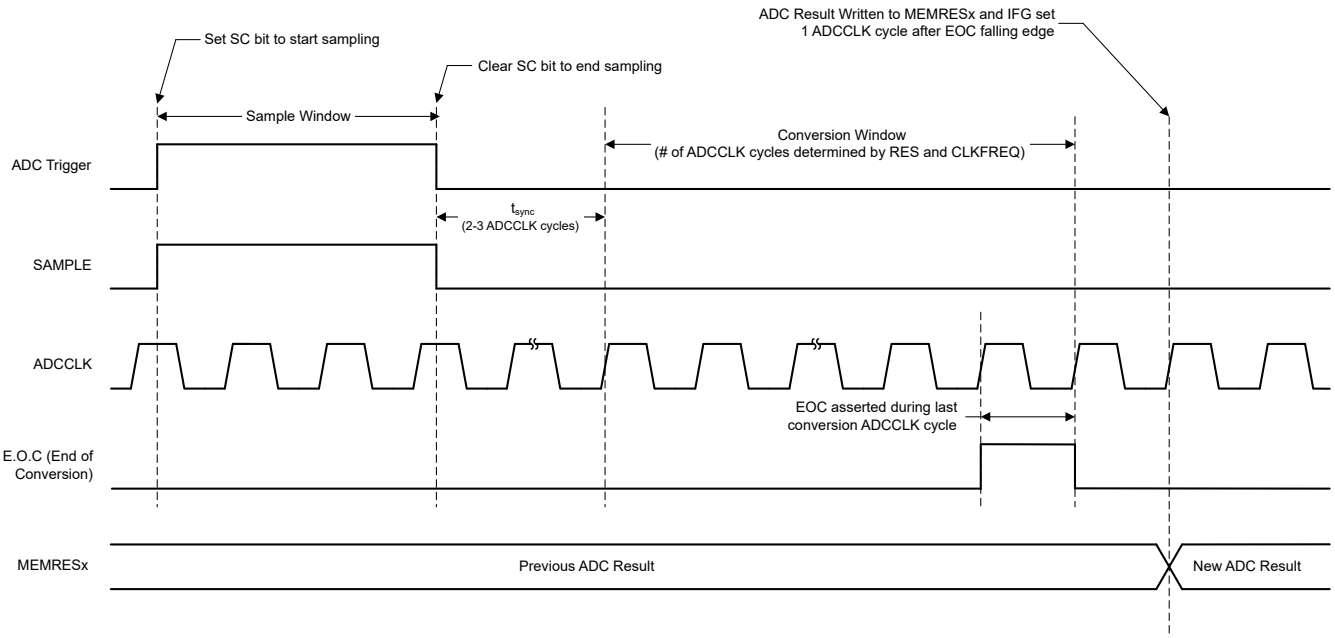


Figure 17-3. MANUAL Sampling Mode—ADC Sample and Conversion Timing Diagram

Note

When the reset value of PWRDN is set as '0,' which has the default behavior of automatic power down, ADC wake-up time must be considered before the sample window. Refer to the device-specific data sheet for specifications on the ADC wake-up time.

17.3.7 Sampling Period

The sampling clock source is selected in the CKMD module using the SRC bits in the ADCCLKSEL register.

The desired sampling period for ADC operation can be generated using the internal clock divider and (or) the sample timer, which applies to AUTO sampling mode. The internal clock divider is configured using the SCLKDIV bits in the CTL0 register and has divide options of 1, 2, 4, 8, 16, 24, 32, and 48.

The duration of the sampling period can be programmed to one of two user-defined values set by the SCOMP0 and SCOMP1 sample timer registers. The value in SCOMPx configures the sampling period by defining the number of sample time clocks to set the sample window to. The default SCOMPx sample timer value translates to 1 cycle-wide sample pulse, which allows the sampling period to be solely based on the sample clock and SCLKDIV. In general, there are three parameters that can be used to control the sample period: SCOMPx, SCLKDIV, and the source of the sample clock.

When AUTO power down mode is selected using PWRDN=0, the module enable signal to the ADC peripheral is generated one sampling clock cycle after the sample signal is asserted. This should be considered by the user in the sample window calculation in addition to the ADC power time or settling time needs of other analog modules such as the temperature sensor, VREF, and so on.

17.3.8 Conversion Modes

There are four conversion modes available in the ADC:

1. Single channel single conversion
   - The channel can be selected using MEMCTL.
   - The selected channel is sampled and converted only once.
2. Repeat single-channel conversion
   - The channel can be selected using MEMCTL.
   - The selected channel is repeatedly sampled and converted until ENC is cleared by software.
3. Sequence of channel conversion
   - Groups of channels can be formed using STARTADD, ENDADD, and MEMCTL registers.
   - Each of the channels in the group is sampled and converted only once.
   - The sequence completes even if ENC is cleared in the middle of the sequence.
4. Repeat the sequence of channels conversion
   - Groups of channels can be formed using STARTADD, ENDADD, and MEMCTL registers.
   - The group of channels is sampled and converted repeatedly until ENC is cleared by software.
   - When ENC is cleared the operation stops at the end of the ongoing sequence.

The following steps outline the recommended process for configuring the ADC for a desired conversion mode:

1. Use the CONSEQ bits in the CTL1 register to select the desired ADC conversion mode.
2. Use the STARTADD bits in the CTL2 register to select which MEMCTLx is used for single conversion or as the first MEMCTL for a sequence mode.
3. If using a sequence mode, use the ENDADD bits in the CTL2 register to select which MEMCTLx is used for the last conversion of the sequence.
4. Assign an ADC input channel to the appropriate MEMCTLx register using the CHANSEL bits.
   - For sequence modes, the ADC input channel must be assigned for each MEMCTLx that is part of the configured sequence.
5. Select EVENT or SOFTWARE trigger using the TRIGSRC bit in the CTL1 register.
6. Select AUTO or MANUAL sampling mode using the SAMPMODE bit in the CTL1 register.
   - If using AUTO mode, program the desired sample timer value in the SCOMPx register and use the STIME bits in the MEMCTLx register to select the appropriate sample timer source (SCOMP0 or SCOMP1).
7. If using repeat single channel or sequence conversion modes, program the TRIG bit in each MEMCTLx register to indicate if a trigger is needed to step to the next MEMCTL in the sequence.
8. Set the ENC bit in the CTL1 register to enable ADC conversions.
9. Table 17-1 depicts the next step of ADC configuration and usage based on the selected trigger and sampling modes:

Table 17-1. Trigger and Sample Mode ADC Usage Matrix

	Software Trigger	Event Trigger
AUTO Sampling Mode	• Set the SC bit to start the sample phase (duration determined by sample timer). • Conversion starts once the sample phase is over. • In single channel single conversion, ENC is cleared when conversion is over. • SC bit is automatically cleared once the trigger is captured. For repeat and sequence modes, if TRIG is set in MEMCTL, the SC bit needs to be set for the next conversion to proceed.	• EVENT trigger starts the sample phase (duration determined by sample timer). • Conversion starts once the sample phase is over. • In single channel single conversion, ENC is cleared when conversion is over. For repeat and sequence modes, the ADC waits for the EVENT trigger or automatically starts the next conversion based on the TRIG setting.
MANUAL Sampling Mode	• Set the SC bit to start the sample phase (the SC bit is not automatically reset). • Clear the SC bit to end the sample phase and start the conversion. • In single channel single conversion, the ENC bit is cleared when the conversion is over. Repeated/sequential conversion modes are NOT supported in this configuration.	ADC operation is NOT supported in this configuration.

10. The ADC results are stored in the MEMRES register of the associated MEMCTL (for example, the MEMCTL0 result is stored in MEMRES0).
    - For repeat conversion modes, the result in MEMRES is updated after every associated MEMCTL conversion.
11. For repeated conversion modes, clear the ENC bit to stop the ADC operation.

Note

In case a hardware event is being used as the sample trigger source, software must ensure that the event trigger is disabled first before clearing the ENC bit to stop ADC operations.

17.3.9 ADC Data Format

The ADC supports two data formats—unsigned binary and 2's complement signed binary. Unsigned binary results are stored right-justified in the MEMRES register or FIFO. Signed binary results are stored left-justified in the MEMRES register or FIFO.

Table 17-2. ADC Data Formats

Data Format	Resolution	Result Range (decimal)	Result Range (hex)
Unsigned	8-bit	0 to 255	0000h to 00FFh
	10-bit	0 to 1023	0000h to 03FFh
	12-bit	0 to 4095	0000h to 0FFFh
Signed	8-bit	-128 to 127	8000h to 7F00h
	10-bit	-512 to 511	8000h to 7FC0h
	12-bit	-2048 to 2047	8000h to 7FF0h

17.3.10 Status Register

The ADC status register, STA, contains two bits—ASCACT and BUSY.

- BUSY equaling '1' indicates that the ADC is busy performing a sample or conversion operation.
  - For **single channel single conversion**, BUSY signals that a trigger has been received and sample or conversion is ongoing. BUSY is cleared when the conversion completes.

- For **repeat single conversion**, BUSY signals that repeat single operation has begun and has not ended. BUSY is cleared when ENC is written '0' and the last conversion completes.
- For **sequence of channels conversion**, BUSY signals that the sequence of channels conversion has started. BUSY is cleared at the end of the sequence
- For **repeat sequence of channels conversion**, BUSY signals the repeat sequence is ongoing. BUSY is cleared when ENC is written '0' and the last conversion in the sequence completes.

Note

In case an ADC start of conversion is issued by the software through the SC bit, the software must wait for at least nine CLKSVT clock cycles if polling for the BUSY status bit in the program code. This is to account for internal clock synchronization latencies before the ADC status bit is updated.

17.3.11 ADC Events

The ADC peripheral contains three event publishers and one event subscriber.

One event publisher (INT_EVENT0) manages ADC interrupt requests (IRQs) to the CPU subsystem through a static event route. The second event publisher (INT_EVENT1) can be used to publish ADC events to a subscriber through a generic event route channel. The third event publisher (INT_EVENT2) can be used as an ADC to µDMA trigger to send ADC events directly to the µDMA.

The event subscriber can be used to subscribe to events that are published to the event fabric through a generic event route channel.

The ADC events are summarized in Table 17-3.

Table 17-3. ADC Events

Event	Type	Source	Destination	Configuration	Functionality
CPU interrupt event	Publisher	ADC	CPU Subsystem	INT_EVENT0 registers	Fixed interrupt route from ADC to CPU
Generic publisher event	Publisher	ADC	Generic event channel	INT_EVENT1 registers	Trigger generic event channel from ADC
µDMA trigger event	Publisher	ADC	µDMA	INT_EVENT2 registers	Fixed trigger route from ADC to µDMA
Generic subscriber event	Subscriber	Other peripherals	ADC	EVTSVT.ADCTRGSEL	ADC subscription to the generic event within EVTSVT

17.3.11.1 CPU Interrupt Event Publisher (INT_EVENT0)

The ADC peripheral provides many interrupt sources that can be configured to source a CPU interrupt event. The CPU interrupt events from the ADC are given in Table 17-4.

Table 17-4. ADC CPU Interrupt Event Conditions (INT_EVENT0)

RIS (Bit Index)	Name	Description
0x0	OVIFG	The conversion overflow interrupt flag is set when the ADC updates MEMRESx before the previous sample is read by the CPU or DMA.
0x1	TOVIFG	The sequence conversion time overflow interrupt flag is set when the ADC receives a new sampling trigger while the previous sample+conversion is still in progress.
0x2	HIGHIFG	A high threshold compare interrupt flag is set when the MEMRESx result register is higher than the WCHIGH threshold of the window comparator.

Table 17-4. ADC CPU Interrupt Event Conditions (INT_EVENT0) (continued)

RIS (Bit Index)	Name	Description
0x3	LOWIFG	A low threshold compare interrupt flag is set when the MEMRESx result register is lower than the WCLOW threshold of the window comparator.
0x4	INIFG	The in-range comparator interrupt flag is set when the MEMRESx result register is within the range of WCLOW and WCHIGH of the window comparator.
0x5	DMADONE	µDMA done interrupt flag is set when the DMA data transfer of programmed block size is completed.
0x6	UVIFG	Conversion underflow interrupt flag, the UVIFG flag is set when the CPU or DMA reads the MEMRESx register before the next conversion result is available.
0x7	ASC	Ad-hoc single conversion done
0x8 to 0xB	MEMRESIFG[0 to 3]	The Memory register interrupt flag is set when MEMRESx is loaded with a new conversion result.

The CPU interrupt event configuration is managed with the INT_EVENT0 event management registers. Interrupt (RIS) flags are cleared upon software writing to the respective ICLR register bits.

17.3.11.2 Generic Event Publisher (INT_EVENT1)

The ADC peripheral provides four interrupt sources, one of which can be configured to publish an event as a generic ADC event. Table 17-5 lists these interrupt sources.

Table 17-5. ADC Generic Event Publisher Conditions (INT_EVENT1)

Index	Name	Description
0x2	HIGHIFG	A high threshold compare interrupt flag is set when the MEMRESx result register is higher than the WCHIGH threshold of the window comparator.
0x3	LOWIFG	A low threshold compare interrupt flag is set when the MEMRESx result register is lower than the WCLOW threshold of the window comparator.
0x4	INIFG	An in-range comparator interrupt flag is set when the MEMRESx result register is within the range of WCLOW and WCHIGH of the window comparator.
0x8	MEMRESIFG0	The memory register interrupt flag is set when MEMRES0 is loaded with a new conversion result.

The generic event publisher configuration is managed with the INT_EVENT1 event management registers.

17.3.11.3 DMA Trigger Event Publisher (INT_EVENT2)

The ADC module provides many interrupt sources which can be configured to source the DMA trigger. In order of decreasing interrupt priority, the DMA trigger events from the ADC are given in Table 17-6. When the DMA channel is needed by the ADC, the DMA trigger is unmasked in the IMASK register of INT_EVENT2, and the DMA is configured as needed to support the ADC operation.

Table 17-6. ADC DMA Trigger Event Conditions (INT_EVENT2)

RIS Index	Name	Description
0x8 to 0xB	MEMRESIFG[0 to 3]	The memory register interrupt flag is set when MEMRESx is loaded with a new conversion result.

The DMA trigger event configuration is managed with the INT_EVENT2 event management registers. The interrupt (RIS) flags are cleared based on ACK from DMA.

17.3.11.4 Generic Event Subscriber

The ADC peripheral supports receiving events routed through a generic channel from other peripherals through the EVTSVT.ADCTRGSEL register.

17.4 Advanced Features

The following sections describe the additional features and benefits provided with the ADC peripheral and how to leverage them in an application.

17.4.1 Window Comparator

There is one window comparator unit available in the ADC which can be used to check if the input signal is within predefined threshold values set by software. The ADC result that goes into MEMRES or FIFO is checked against the threshold values of the window comparator.

Based on the comparison the window comparator can generate 3 interrupt conditions:

1. LOWIFG—Conversion result is below the low threshold (WCLOW).
2. HIGHIFG—Conversion result is above the high threshold (WCHIGH).
3. INIFG—Conversion result is in between or equal to the Low and High thresholds.

The window comparator low and high threshold values are global for all channels and the window comparison feature can be enabled for each channel as needed using the WINCOMP bit in the MEMCTL register.

When the ADC result data format (CTL2.DF) or resolution (CTL2.RES) configuration is changed, the window comparator threshold values are not reset by hardware and are retained as-is. The software application is expected to reconfigure the threshold values as appropriate after changing the data format and (or) resolution configuration.

17.4.2 DMA and FIFO Operation

The ADC has a dedicated interface for communicating with the µDMA. This interface is useful to offload work from the CPU by using the µDMA to store ADC results to memory automatically.

The DMAEN bit in the CTL2 register is used to enable the µDMA for ADC data transfer. The DMAEN bit is cleared by ADC hardware when the µDMA “DONE" status signal is asserted. Software is expected to reenable the µDMA using DMAEN to arm the ADC to generate the next µDMA trigger.

The ADC also incorporates an optional first-in-first-out buffer to provide a way for ADC results to be stored for future use, such as transferring to memory by the µDMA. Either the CPU or the µDMA can be used to move data from the ADC regardless of whether the FIFO is enabled or disabled. The memory result flags in the RIS register of the third event publisher serve as the FIFO threshold and can be unmasked to generate the µDMA trigger.

The following sections explain the details of using the ADC with µDMA or CPU in various conversion modes and with the FIFO enabled or disabled.

17.4.2.1 DMA/CPU Operation in Non-FIFO Mode (FIFOEN=0)

- Single Conversion and Repeat Single Conversion
  - Configure STARTADD bits to select the desired MEMCTLx register.
    - MEMCTLx **is correlated to** MEMRESx.
    - MEMRESx **is correlated to** MEMRESIFGx.
  - Configure MEMCTL CHANSEL bits to select the desired ADC channel.

  - Conversion data is available in MEMRESx.
  - MEMRESIFGx can be set to generate a CPU interrupt or the DMA trigger.
  - The conversion overflow flag is set when the ADC updates MEMRESx before the previous sample is read by the CPU or DMA.
  - The conversion underflow flag is set when the CPU or DMA reads the MEMRESx register before the next conversion result is available.
- Sequence Conversion and Repeat Sequence Conversion
  - Configure STARTADD bits to select the first MEMCTL in the sequence.
  - Configure ENDADD bits to select the last MEMCTL in the sequence.
    - MEMCTLx **is correlated to** MEMRESIFGx.
  - Configure each MEMCTLx CHANSEL bits to select the desired ADC channels.
  - Conversion data is available in MEMRESx.
  - MEMRESIFGx can be set to generate a CPU interrupt or the DMA trigger.
  - The conversion overflow flag is set when the ADC updates MEMRESx before the previous sample is read by the CPU or DMA.
  - The conversion underflow flag is set when the CPU or DMA reads the MEMRESx register before the next conversion result is available.

Note

For DMA based operation, the MEMCTL start address should be smaller than the end address for single sequence conversion as DMA source does not roll back. Repeat sequence conversion mode does not support DMA based data transfer because the DMA does not support circular addressing mode.

17.4.2.2 DMA/CPU Operation in FIFO Mode (FIFOEN=1)

- Single Conversion and Repeat Single Conversion
  - Configure STARTADD bits to select the desired MEMCTLx register.
    - MEMCTLx **is NOT correlated to** MEMRESx.
    - MEMRESx **is correlated to** MEMRESIFGx.
  - Configure MEMCTL CHANSEL bits to select the desired ADC channel.
  - Conversion data is loaded sequentially into MEMRES0,1,2,….N (organized as a FIFO).
  - The CPU or DMA must read ADC samples from the dedicated FIFODATA register and not from MEMRES registers directly.
    - Data in the FIFO is always compacted with two samples and provided as 32-bit data upon a FIFODATA read by CPU or DMA.
  - MEMRESIFGx can be used as a threshold condition to generate a CPU interrupt or DMA trigger.
    - For full use of the FIFO, the last MEMRESIFG can be used.
  - The conversion overflow flag is set when the ADC updates MEMRESx before the previous sample is read by the CPU or DMA.
  - The conversion underflow flag is set when the CPU or DMA reads the FIFODATA register before the conversion result is available in the MEMRESx registers.

Note

Single conversion mode with FIFO enabled is not recommended for CPU or DMA-based operation. This leads to underflow conditions and unwanted 16-bit data has to be discarded by software.

- Sequence Conversion and Repeat Sequence Conversion
  - Configure STARTADD bits to select the first MEMCTL in the sequence.
  - Configure ENDADD bits to select the last MEMCTL in the sequence.
    - MEMCTLx **is NOT correlated to** MEMRESx.
    - MEMRESx **is correlated to** MEMRESIFGx.
  - Configure each MEMCTLx CHANSEL bit to select the desired ADC channels
  - Conversion data is loaded sequentially into MEMRES0,1,2,….N (organized as a FIFO).

- The CPU or DMA must read ADC samples from the dedicated FIFODATA register and not from MEMRES registers directly.
  - Data in the FIFO is always compacted with two samples and provided as 32-bit data upon a FIFODATA read by CPU or DMA.
- MEMRESIFGx can be used as a threshold condition to generate a CPU interrupt or DMA trigger.
  - For full use of the FIFO, the last MEMRESIFG can be used.
- The conversion overflow flag is set when the ADC updates MEMRESx before the previous sample is read by the CPU or DMA.

17.4.2.3 DMA/CPU Operation Summary Matrix

Table 17-7. DMA/CPU Operation Summary Matrix

Conversion Mode	FIFO Disabled (FIFOEN=0) Samples not compacted. Read from MEMRESx registers directly		FIFO Enabled (FIFOEN=1) Samples always compacted Read from FIFODAT register only	
	CPU Read/Write	DMA Read/Write	CPU Read/Write	DMA Read/Write
Single	Supported	Supported	Not recommended Underflow flag is set Ignore unwanted 16 bits	Not recommended Underflow flag is set Ignore unwanted 16 bits
Repeat Single	Supported	Supported	Supported MEMRESIFG=CPU interrupt FIFODATA read in 32 bits	Supported MEMRESIFG=DMA trigger FIFODATA read in 32 bits
Sequence	Supported	Supported	Supported MEMRESIFG=CPU interrupt FIFODATA read in 32 bits	Supported MEMRESIFG=DMA trigger FIFODATA read in 32 bits
Repeat Sequence	Supported	Not Supported	Supported MEMRESIFG=CPU interrupt FIFODATA read in 32 bits	Supported MEMRESIFG=DMA trigger FIFODATA read in 32 bits

17.4.3 Ad-Hoc Single Conversion

A mechanism to allow ADC to perform ad-hoc single conversions (ASC) without affecting the scheduled conversions is provided. The ADC sequencer slots the ASC request at a time when it finds an idle window in the middle of scheduled conversions without affecting the timing integrity of the scheduled conversions.

This is requested through the CTL3 register, which has fields for specifying the ADC channel number, voltage reference option, and sample period for conversion. Any write to this register is treated as ad-hoc single conversion request by the sequencer. There is a separate result register available to store the data for ad-hoc single conversion (ASCRES). This is a dedicated register for ad-hoc single conversion operation which is different than result registers/FIFO available to store results from conversion on sensor channels.

Once software writes into the ASC configuration register for ad-hoc single conversion there is a status bit that indicates the ASC is active (ASCACT) and goes low once the ASC operation is completed.

When the ASC operation is completed, an interrupt flag ASC done (ASCDONE) is set that can be unmasked by the software to read the ASC result in the interrupt service routine.

The software can write into the ASC configuration register at any time in an ad-hoc manner and that request is registered by the sequencer and serviced at a suitable time.

Figure 17-4 shows the ADC sequencer state-machine for ASC operation.

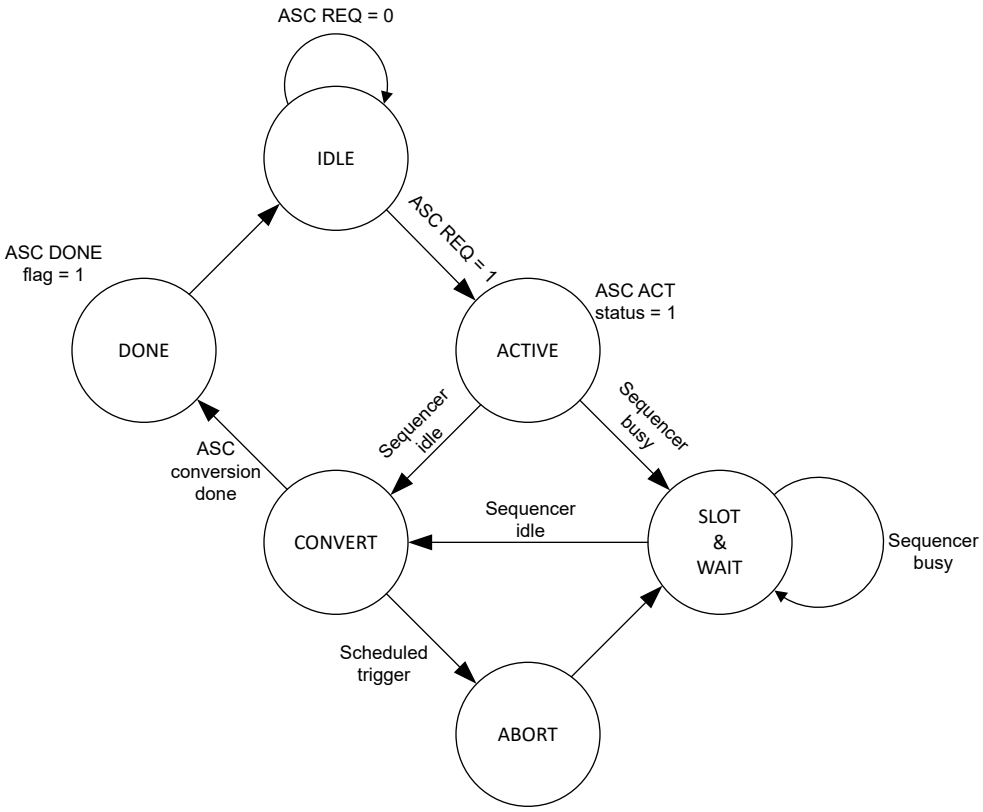


Figure 17-4. ADC Sequencer State-Machine for ASC Operation

Repeat Single Channel Mode and ASC Request

- When the sequencer operates in a repeat single channel with sample trigger policy as auto-next, then the selected sensor channel is converted back to back continuously and the ASC request is pended by the sequencer and taken up and serviced only when the software stops repeat single channel conversion. When the sample trigger policy is trigger-next, then upon ASC request, the sequencer tries to schedule the ASC operation at the end of ongoing conversion (end of conversion [EOC]).
- It starts ASC operation and will complete it successfully if the scheduled trigger on the sensor channel does not arrive in between.
- If the scheduled trigger is received in the middle of the ASC operation, then ASC conversion is aborted immediately and scheduled conversion is performed.
- If the sequencer is not successful in completing ASC operation in the middle of scheduled conversions, then it will be serviced only when the software stops repeat single-channel conversions.

Sequence of Channels mode and ASC Request

- In the case of a sequence of channels operation with a sample trigger policy as auto-next for all channels in the sequence, the ASC request will be slotted at the end of the sequence and completed.
- If the sample trigger policy is trigger-next for one or more channels in the sequence, then the sequencer tries to schedule the ASC operation at the EOC of the channel with trigger next policy set.
- If it cannot complete ASC conversion successfully due to the arrival of the scheduled trigger then ASC operation is taken up and completed at the end after the conversion of all channels in the sequence are completed.

Repeat Sequence of Channels Mode and ASC Request

- In the case of a repeat sequence of channels operation with the sample trigger policy as auto-next for all channels in the sequence, the ASC request will be slotted and completed when the repeat sequence operation is stopped by the software.

- If the sample trigger policy is trigger-next for one or more channels in the sequence, then the sequencer tries to schedule the ASC operation at the EOC of the channel with the trigger-next policy set.
- If it can't complete ASC conversion successfully due to the arrival of the scheduled trigger then ASC operation is taken up and completed when the repeat sequence operation is stopped by software.

ASC Operation Abort Due to Scheduled Trigger

- When the scheduled trigger arrives during the sample phase of ASC operation then the sequencer pulls the sample signal low immediately, and applies reset to ADC SAR logic, and then generates a sample trigger for the scheduled conversion.
- When the scheduled trigger arrives during the conversion phase of ASC operation then sequencer applies reset to ADC SAR logic and then generates a sample trigger for the scheduled conversion.
- When the ASC operation gets aborted due to the arrival of the scheduled trigger, the sequencer attempts to perform the ASC operation automatically at the next earliest idle slot without software requiring to reissue ASC request.
- ASC request is not pipelined which means software has to issue ASC request only when ASC active status is low.
- If an ASC request is raised while the previous ASC operation is not completed then that ASC request is ignored and the software has to reissue the ASC request when the ASC active is low.

17.5 ADC Registers

Table 17-8 lists the memory-mapped registers for the ADC registers. All register offset addresses not listed in Table 17-8 should be considered as reserved locations and the register contents should not be modified.

Table 17-8. ADC Registers

Offset	Acronym	Register Name	Section
28h	IMASK0	Interrupt mask	Go
30h	RIS0	Raw interrupt status	Go
38h	MIS0	Masked interrupt status	Go
40h	ISET0	Interrupt set	Go
48h	ICLR0	Interrupt clear	Go
58h	IMASK1	Interrupt mask	Go
60h	RIS1	Raw interrupt status	Go
68h	MIS1	Masked interrupt status	Go
70h	ISET1	Interrupt set	Go
78h	ICLR1	Interrupt clear	Go
88h	IMASK2	Interrupt mask	Go
90h	RIS2	Raw interrupt status	Go
98h	MIS2	Masked interrupt status	Go
A0h	ISET2	Interrupt set	Go
A8h	ICLR2	Interrupt clear	Go
100h	CTL0	Control Register 0	Go
104h	CTL1	Control Register 1	Go
108h	CTL2	Control Register 2	Go
10Ch	CTL3	Control Register 3	Go
114h	SCOMP0	Sample Time Compare 0 Register	Go
118h	SCOMP1	Sample Time Compare 1 Register	Go
11Ch	REFCFG	Reference Buffer Configuration Register	Go
148h	WCLOW	Window Comparator Low Threshold Register	Go
150h	WCHIGH	Window Comparator High Threshold Register	Go
160h	FIFODATA	FIFO Data Register	Go
170h	ASCRES	ASC Result Register	Go
180h	MEMCTL0	Conversion Memory Control Register 0	Go
184h	MEMCTL1	Conversion Memory Control Register 1	Go
188h	MEMCTL2	Conversion Memory Control Register 2	Go
18Ch	MEMCTL3	Conversion Memory Control Register 3	Go
280h	MEMRES0	Memory Result Register 0	Go
284h	MEMRES1	Memory Result Register 1	Go
288h	MEMRES2	Memory Result Register 2	Go
28Ch	MEMRES3	Memory Result Register 3	Go
340h	STA	Status Register	Go
E00h	TEST0	Internal. Only to be used through TI provided API.	Go
E08h	TEST2	Internal. Only to be used through TI provided API.	Go
E0Ch	TEST3	Internal. Only to be used through TI provided API.	Go
E10h	TEST4	Internal. Only to be used through TI provided API.	Go
E14h	TEST5	Internal. Only to be used through TI provided API.	Go
E18h	TEST6	Internal. Only to be used through TI provided API.	Go

Table 17-8. ADC Registers (continued)

Offset	Acronym	Register Name	Section
E20h	DEBUG1	Internal. Only to be used through TI provided API.	Go
E24h	DEBUG2	Internal. Only to be used through TI provided API.	Go
E28h	DEBUG3	Internal. Only to be used through TI provided API.	Go
E2Ch	DEBUG4	Internal. Only to be used through TI provided API.	Go

Complex bit access types are encoded to fit into small table cells. Table 17-9 shows the codes that are used for access types in this section.

Table 17-9. ADC Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

17.5.1 IMASK0 Register (Offset = 28h) [Reset = 00000000h]

IMASK0 is shown in Table 17-10.

Return to the Summary Table.

Interrupt mask. This register selects interrupt sources which are allowed to pass from RIS0 to MIS0 when the corresponding bit-fields are set to 1.

Table 17-10. IMASK0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-12	RESERVED	R	0h	Reserved
11	MEMRESIFG3	R/W	0h	MEMRES3 conversion result interrupt mask. 0h = Disable interrupt mask 1h = Enable interrupt mask
10	MEMRESIFG2	R/W	0h	MEMRES2 conversion result interrupt mask. 0h = Disable interrupt mask 1h = Enable interrupt mask
9	MEMRESIFG1	R/W	0h	MEMRES1 conversion result interrupt mask. 0h = Disable interrupt mask 1h = Enable interrupt mask
8	MEMRESIFG0	R/W	0h	MEMRES0 conversion result interrupt mask. 0h = Disable interrupt mask 1h = Enable interrupt mask
7	ASCDONE	R/W	0h	Mask for ASC done raw interrupt flag. 0h = Disable interrupt mask 1h = Enable interrupt mask
6	UVIFG	R/W	0h	Conversion underflow interrupt mask. 0h = Disable interrupt mask 1h = Enable interrupt mask
5	DMADONE	R/W	0h	DMA done interrupt mask. 0h = Disable interrupt mask 1h = Enable interrupt mask
4	INIFG	R/W	0h	In-range comparator interrupt mask. 0h = Disable interrupt mask 1h = Enable interrupt mask
3	LOWIFG	R/W	0h	Low threshold compare interrupt mask. 0h = Disable interrupt mask 1h = Enable interrupt mask
2	HIGHIFG	R/W	0h	High threshold compare interrupt mask. 0h = Disable interrupt mask 1h = Enable interrupt mask
1	TOVIFG	R/W	0h	Sequence conversion time overflow interrupt mask. 0h = Disable interrupt mask 1h = Enable interrupt mask
0	OVIFG	R/W	0h	Conversion overflow interrupt mask. 0h = Disable interrupt mask 1h = Enable interrupt mask

17.5.2 RIS0 Register (Offset = 30h) [Reset = 00000000h]

RIS0 is shown in Table 17-11.

Return to the Summary Table.

Raw interrupt status. This register reflects the state of all pending interrupts, regardless of masking. This register allows the user to implement a poll scheme. A flag set in this register can be cleared by writing 1 to the corresponding ICLR0 register bit.

Table 17-11. RIS0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-12	RESERVED	R	0h	Reserved
11	MEMRESIFG3	R/W	0h	Raw interrupt status for MEMRES3. This bit is set to 1 when MEMRES3 is loaded with a new conversion result. Reading MEMRES3 register will clear this bit, or when the corresponding bit in ICLR0 is set to 1 0h = No new data ready. 1h = A new data is ready to be read.
10	MEMRESIFG2	R/W	0h	Raw interrupt status for MEMRES2. This bit is set to 1 when MEMRES2 is loaded with a new conversion result. Reading MEMRES2 register will clear this bit, or when the corresponding bit in ICLR0 is set to 1 0h = No new data ready. 1h = A new data is ready to be read.
9	MEMRESIFG1	R/W	0h	Raw interrupt status for MEMRES1. This bit is set to 1 when MEMRES1 is loaded with a new conversion result. Reading MEMRES1 register will clear this bit, or when the corresponding bit in ICLR0 is set to 1 0h = No new data ready. 1h = A new data is ready to be read.
8	MEMRESIFG0	R/W	0h	Raw interrupt status for MEMRES0. This bit is set to 1 when MEMRES0 is loaded with a new conversion result. Reading MEMRES0 register will clear this bit, or when the corresponding bit in ICLR0 is set to 1 0h = No new data ready. 1h = A new data is ready to be read.
7	ASCDONE	R/W	0h	Raw interrupt flag for ASC done. 0h = Interrupt is not pending. 1h = Interrupt is pending.
6	UVIFG	R/W	0h	Raw interrupt flag for MEMRESx underflow. 0h = Interrupt is not pending. 1h = Interrupt is pending.
5	DMADONE	R/W	0h	Raw interrupt flag for DMADONE. 0h = Interrupt is not pending. 1h = Interrupt is pending.
4	INIFG	R/W	0h	Raw interrupt status for In-range comparator. 0h = Interrupt is not pending. 1h = Interrupt is pending.
3	LOWIFG	R/W	0h	Raw interrupt flag for the MEMRESx result register being below than the WCLOWx threshold of the window comparator. 0h = Interrupt is not pending. 1h = Interrupt is pending.
2	HIGHIFG	R/W	0h	Raw interrupt flag for the MEMRESx result register being higher than the WCHIGHx threshold of the window comparator. 0h = Interrupt is not pending. 1h = Interrupt is pending.

Table 17-11. RIS0 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
1	TOVIFG	R/W	0h	Raw interrupt flag for sequence conversion trigger overflow. 0h = Interrupt is not pending. 1h = Interrupt is pending.
0	OVIFG	R/W	0h	Raw interrupt flag for MEMRESx overflow. 0h = Interrupt is not pending. 1h = Interrupt is pending.

17.5.3 MIS0 Register (Offset = 38h) [Reset = 00000000h]

MIS0 is shown in Table 17-12.

Return to the Summary Table.

Masked interrupt status. This register is simply a bitwise AND of the contents of IMASK and RIS registers. A flag set in this register can be cleared by writing 1 to the corresponding ICLR register bit.

Table 17-12. MIS0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-12	RESERVED	R	0h	Reserved
11	MEMRESIFG3	R/W	0h	Masked interrupt status for MEMRES3. 0h = No new data ready. 1h = A new data is ready to be read.
10	MEMRESIFG2	R/W	0h	Masked interrupt status for MEMRES2. 0h = No new data ready. 1h = A new data is ready to be read.
9	MEMRESIFG1	R/W	0h	Masked interrupt status for MEMRES1. 0h = No new data ready. 1h = A new data is ready to be read.
8	MEMRESIFG0	R/W	0h	Masked interrupt status for MEMRES0. 0h = No new data ready. 1h = A new data is ready to be read.
7	ASCDONE	R/W	0h	Masked interrupt status for ASC done. 0h = Interrupt is not pending. 1h = Interrupt is pending.
6	UVIFG	R/W	0h	Masked interrupt flag for MEMRESx underflow. 0h = Interrupt is not pending. 1h = Interrupt is pending.
5	DMADONE	R/W	0h	Masked interrupt flag for DMADONE. 0h = Interrupt is not pending. 1h = Interrupt is pending.
4	INIFG	R/W	0h	Mask INIFG in MIS0 register. 0h = Interrupt is not pending. 1h = Interrupt is pending.
3	LOWIFG	R/W	0h	Masked interrupt flag for the MEMRESx result register being below than the WCLOWx threshold of the window comparator. 0h = Interrupt is not pending. 1h = Interrupt is pending.
2	HIGHIFG	R/W	0h	Masked interrupt flag for the MEMRESx result register being higher than the WCHIGHx threshold of the window comparator. 0h = Interrupt is not pending. 1h = Interrupt is pending.
1	TOVIFG	R/W	0h	Masked interrupt flag for sequence conversion timeout overflow. 0h = Interrupt is not pending. 1h = Interrupt is pending.
0	OVIFG	R/W	0h	Masked interrupt flag for MEMRESx overflow. 0h = Interrupt is not pending. 1h = Interrupt is pending.

17.5.4 ISET0 Register (Offset = 40h) [Reset = 00000000h]

ISET0 is shown in Table 17-13.

Return to the Summary Table.

Interrupt set register. This register can used by software for diagnostics and safety checking purposes. Writing a 1 to a bit in this register will set the event and the corresponding RIS bit also gets set. If the corresponding IMASK bit is set, then the corresponding MIS register bit also gets set.

Table 17-13. ISET0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-12	RESERVED	R	0h	Reserved
11	MEMRESIFG3	R/W	0h	Set interrupt status for MEMRES3. 0h = No new data ready. 1h = A new data is ready to be read.
10	MEMRESIFG2	R/W	0h	Set interrupt status for MEMRES2. 0h = No new data ready. 1h = A new data is ready to be read.
9	MEMRESIFG1	R/W	0h	Set interrupt status for MEMRES1. 0h = No new data ready. 1h = A new data is ready to be read.
8	MEMRESIFG0	R/W	0h	Set Interrupt status for MEMRES0. 0h = No new data ready. 1h = A new data is ready to be read.
7	ASCDONE	R/W	0h	Set interrupt for ASC done. 0h = Interrupt is not pending. 1h = Interrupt is pending.
6	UVIFG	R/W	0h	Set interrupt for MEMRESx underflow. 0h = Interrupt is not pending. 1h = Interrupt is pending.
5	DMADONE	R/W	0h	Set interrupt for DMADONE. 0h = Interrupt is not pending. 1h = Interrupt is pending.
4	INIFG	R/W	0h	Set INIFG interrupt register. 0h = Interrupt is not pending. 1h = Interrupt is pending.
3	LOWIFG	R/W	0h	Set interrupt for MEMRESx result register being below than the WCLOWx threshold of the window comparator. 0h = Interrupt is not pending. 1h = Interrupt is pending.
2	HIGHIFG	R/W	0h	Set Interrupt for the MEMRESx result register being higher than the WCHIGHx threshold of the window comparator. 0h = Interrupt is not pending. 1h = Interrupt is pending.
1	TOVIFG	R/W	0h	Set interrupt for sequence conversion timeout overflow. 0h = Interrupt is not pending. 1h = Interrupt is pending.
0	OVIFG	R/W	0h	Set Interrupt for MEMRESx overflow. 0h = Interrupt is not pending. 1h = Interrupt is pending.

17.5.5 ICLR0 Register (Offset = 48h) [Reset = 00000000h]

ICLR0 is shown in Table 17-14.

Return to the Summary Table.

Interrupt clear register. This register allows software to clear interrupts. Writing a 1 to a bit in this register will clear the event and the corresponding RIS bit also gets cleared. If the corresponding IMASK bit is set, then the corresponding MIS register bit also gets cleared.

Table 17-14. ICLR0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-12	RESERVED	R	0h	Reserved
11	MEMRESIFG3	R/W	0h	Clear interrupt status for MEMRES3. 0h = No new data ready. 1h = A new data is ready to be read.
10	MEMRESIFG2	R/W	0h	Clear interrupt status for MEMRES2. 0h = No new data ready. 1h = A new data is ready to be read.
9	MEMRESIFG1	R/W	0h	Clear interrupt status for MEMRES1. 0h = No new data ready. 1h = A new data is ready to be read.
8	MEMRESIFG0	R/W	0h	Clear interrupt status for MEMRES0. 0h = No new data ready. 1h = A new data is ready to be read.
7	ASCDONE	R/W	0h	Clear ASC done flag in RIS. 0h = Interrupt is not pending. 1h = Interrupt is pending.
6	UVIFG	R/W	0h	Clear interrupt flag for MEMRESx underflow. 0h = Interrupt is not pending. 1h = Interrupt is pending.
5	DMADONE	R/W	0h	Clear interrupt flag for DMADONE. 0h = Interrupt is not pending. 1h = Interrupt is pending.
4	INIFG	R/W	0h	Clear INIFG in MIS0 register. 0h = Interrupt is not pending. 1h = Interrupt is pending.
3	LOWIFG	R/W	0h	Clear interrupt flag for the MEMRESx result register being below than the WCLOWx threshold of the window comparator. 0h = Interrupt is not pending. 1h = Interrupt is pending.
2	HIGHIFG	R/W	0h	Clear interrupt flag for the MEMRESx result register being higher than the WCHIGHx threshold of the window comparator. 0h = Interrupt is not pending. 1h = Interrupt is pending.
1	TOVIFG	R/W	0h	Clear interrupt flag for sequence conversion timeout overflow. 0h = Interrupt is not pending. 1h = Interrupt is pending.
0	OVIFG	R/W	0h	Clear interrupt flag for MEMRESx overflow. 0h = Interrupt is not pending. 1h = Interrupt is pending.

17.5.6 IMASK1 Register (Offset = 58h) [Reset = 00000000h]

IMASK1 is shown in Table 17-15.

Return to the Summary Table.

Interrupt mask. This register selects interrupt sources which are allowed to pass from RIS0 to MIS0 when the corresponding bit-fields are set to 1.

Table 17-15. IMASK1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-9	RESERVED	R	0h	Reserved
8	MEMRESIFG0	R/W	0h	MEMRES0 conversion result interrupt mask. 0h = No new data ready. 1h = A new data is ready to be read.
7-5	RESERVED	R	0h	Reserved
4	INIFG	R/W	0h	In-range comparator interrupt mask. 0h = Interrupt is not pending. 1h = Interrupt is pending.
3	LOWIFG	R/W	0h	Low threshold compare interrupt mask. 0h = Interrupt is not pending. 1h = Interrupt is pending.
2	HIGHIFG	R/W	0h	High threshold compare interrupt mask. 0h = Interrupt is not pending. 1h = Interrupt is pending.
1-0	RESERVED	R	0h	Reserved

17.5.7 RIS1 Register (Offset = 60h) [Reset = 00000000h]

RIS1 is shown in Table 17-16.

Return to the Summary Table.

Raw interrupt status. This register reflects the state of all pending interrupts, regardless of masking. This register allows the user to implement a poll scheme. A flag set in this register can be cleared by writing 1 to the corresponding ICLR0 register bit.

Table 17-16. RIS1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-9	RESERVED	R	0h	Reserved
8	MEMRESIFG0	R/W	0h	Raw interrupt status for MEMRES0. This bit is set to 1 when MEMRES0 is loaded with a new conversion result. Reading MEMRES0 register will clear this bit, or when the corresponding bit in ICLR1 is set to 1 0h = No new data ready. 1h = A new data is ready to be read.
7-5	RESERVED	R	0h	Reserved
4	INIFG	R/W	0h	Raw interrupt status for In-range comparator. 0h = Interrupt is not pending. 1h = Interrupt is pending.
3	LOWIFG	R/W	0h	Raw interrupt flag for the MEMRESx result register being below than the WCLOWx threshold of the window comparator. 0h = Interrupt is not pending. 1h = Interrupt is pending.
2	HIGHIFG	R/W	0h	Raw interrupt flag for the MEMRESx result register being higher than the WCHIGHx threshold of the window comparator. 0h = Interrupt is not pending. 1h = Interrupt is pending.
1-0	RESERVED	R	0h	Reserved

17.5.8 MIS1 Register (Offset = 68h) [Reset = 00000000h]

MIS1 is shown in Table 17-17.

Return to the Summary Table.

Masked interrupt status. This register is simply a bitwise AND of the contents of IMASK and RIS registers. A flag set in this register can be cleared by writing 1 to the corresponding ICLR register bit.

Table 17-17. MIS1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-9	RESERVED	R	0h	Reserved
8	MEMRESIFG0	R/W	0h	Masked interrupt status for MEMRES0. 0h = No new data ready. 1h = A new data is ready to be read.
7-5	RESERVED	R	0h	Reserved
4	INIFG	R/W	0h	Mask INIFG in MIS1 register. 0h = Interrupt is not pending. 1h = Interrupt is pending.
3	LOWIFG	R/W	0h	Masked interrupt flag for the MEMRESx result register being below than the WCLOWx threshold of the window comparator. 0h = Interrupt is not pending. 1h = Interrupt is pending.
2	HIGHIFG	R/W	0h	Masked interrupt flag for the MEMRESx result register being higher than the WCHIGHx threshold of the window comparator. 0h = Interrupt is not pending. 1h = Interrupt is pending.
1-0	RESERVED	R	0h	Reserved

17.5.9 ISET1 Register (Offset = 70h) [Reset = 00000000h]

ISET1 is shown in Table 17-18.

Return to the Summary Table.

Interrupt set register. This register can used by software for diagnostics and safety checking purposes. Writing a 1 to a bit in this register will set the event and the corresponding RIS bit also gets set. If the corresponding IMASK bit is set, then the corresponding MIS register bit also gets set.

Table 17-18. ISET1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-9	RESERVED	R	0h	Reserved
8	MEMRESIFG0	R/W	0h	Set Interrupt status for MEMRES0. 0h = No new data ready. 1h = A new data is ready to be read.
7-5	RESERVED	R	0h	Reserved
4	INIFG	R/W	0h	Set INIFG interrupt register. 0h = Interrupt is not pending. 1h = Interrupt is pending.
3	LOWIFG	R/W	0h	Set interrupt for MEMRESx result register being below than the WCLOWx threshold of the window comparator. 0h = Interrupt is not pending. 1h = Interrupt is pending.
2	HIGHIFG	R/W	0h	Set Interrupt for the MEMRESx result register being higher than the WCHIGHx threshold of the window comparator. 0h = Interrupt is not pending. 1h = Interrupt is pending.
1-0	RESERVED	R	0h	Reserved

17.5.10 ICLR1 Register (Offset = 78h) [Reset = 00000000h]

ICLR1 is shown in Table 17-19.

Return to the Summary Table.

Interrupt clear register. This register allows software to clear interrupts. Writing a 1 to a bit in this register will clear the event and the corresponding RIS bit also gets cleared. If the corresponding IMASK bit is set, then the corresponding MIS register bit also gets cleared.

Table 17-19. ICLR1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-9	RESERVED	R	0h	Reserved
8	MEMRESIFG0	R/W	0h	Clear interrupt status for MEMRES0. 0h = No new data ready. 1h = A new data is ready to be read.
7-5	RESERVED	R	0h	Reserved
4	INIFG	R/W	0h	Clear INIFG in MIS1 register. 0h = Interrupt is not pending. 1h = Interrupt is pending.
3	LOWIFG	R/W	0h	Clear interrupt flag for the MEMRESx result register being below than the WCLOWx threshold of the window comparator. 0h = Interrupt is not pending. 1h = Interrupt is pending.
2	HIGHIFG	R/W	0h	Clear interrupt flag for the MEMRESx result register being higher than the WCHIGHx threshold of the window comparator. 0h = Interrupt is not pending. 1h = Interrupt is pending.
1-0	RESERVED	R	0h	Reserved

17.5.11 IMASK2 Register (Offset = 88h) [Reset = 00000000h]

IMASK2 is shown in Table 17-20.

Return to the Summary Table.

Interrupt mask. This register selects interrupt sources which are allowed to pass from RIS0 to MIS0 when the corresponding bit-fields are set to 1.

Table 17-20. IMASK2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-12	RESERVED	R	0h	Reserved
11	MEMRESIFG3	R/W	0h	MEMRES3 conversion result interrupt mask. 0h = No new data ready. 1h = A new data is ready to be read.
10	MEMRESIFG2	R/W	0h	MEMRES2 conversion result interrupt mask. 0h = No new data ready. 1h = A new data is ready to be read.
9	MEMRESIFG1	R/W	0h	MEMRES1 conversion result interrupt mask. 0h = No new data ready. 1h = A new data is ready to be read.
8	MEMRESIFG0	R/W	0h	MEMRES0 conversion result interrupt mask. 0h = No new data ready. 1h = A new data is ready to be read.
7-0	RESERVED	R	0h	Reserved

17.5.12 RIS2 Register (Offset = 90h) [Reset = 00000000h]

RIS2 is shown in Table 17-21.

Return to the Summary Table.

Raw interrupt status. This register reflects the state of all pending interrupts, regardless of masking. This register allows the user to implement a poll scheme. A flag set in this register can be cleared by writing 1 to the corresponding ICLR0 register bit.

Table 17-21. RIS2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-12	RESERVED	R	0h	Reserved
11	MEMRESIFG3	R/W	0h	Raw interrupt status for MEMRES3. This bit is set to 1 when MEMRES3 is loaded with a new conversion result. Reading MEMRES3 register will clear this bit, or when the corresponding bit in ICLR2 is set to 1 0h = No new data ready. 1h = A new data is ready to be read.
10	MEMRESIFG2	R/W	0h	Raw interrupt status for MEMRES2. This bit is set to 1 when MEMRES2 is loaded with a new conversion result. Reading MEMRES2 register will clear this bit, or when the corresponding bit in ICLR2 is set to 1 0h = No new data ready. 1h = A new data is ready to be read.
9	MEMRESIFG1	R/W	0h	Raw interrupt status for MEMRES1. This bit is set to 1 when MEMRES1 is loaded with a new conversion result. Reading MEMRES1 register will clear this bit, or when the corresponding bit in ICLR2 is set to 1 0h = No new data ready. 1h = A new data is ready to be read.
8	MEMRESIFG0	R/W	0h	Raw interrupt status for MEMRES0. This bit is set to 1 when MEMRES0 is loaded with a new conversion result. Reading MEMRES0 register will clear this bit, or when the corresponding bit in ICLR2 is set to 1 0h = No new data ready. 1h = A new data is ready to be read.
7-0	RESERVED	R	0h	Reserved

17.5.13 MIS2 Register (Offset = 98h) [Reset = 00000000h]

MIS2 is shown in Table 17-22.

Return to the Summary Table.

Extension of Masked interrupt status. This register is simply a bitwise AND of the contents of IMASK and RIS registers. A flag set in this register can be cleared by writing 1 to the corresponding ICLR register bit.

Table 17-22. MIS2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-12	RESERVED	R	0h	Reserved
11	MEMRESIFG3	R/W	0h	Masked interrupt status for MEMRES3. 0h = No new data ready. 1h = A new data is ready to be read.
10	MEMRESIFG2	R/W	0h	Masked interrupt status for MEMRES2. 0h = No new data ready. 1h = A new data is ready to be read.
9	MEMRESIFG1	R/W	0h	Masked interrupt status for MEMRES1. 0h = No new data ready. 1h = A new data is ready to be read.
8	MEMRESIFG0	R/W	0h	Masked interrupt status for MEMRES0. 0h = No new data ready. 1h = A new data is ready to be read.
7-0	RESERVED	R	0h	Reserved

17.5.14 ISET2 Register (Offset = A0h) [Reset = 00000000h]

ISET2 is shown in Table 17-23.

Return to the Summary Table.

Interrupt set register. This register can used by software for diagnostics and safety checking purposes. Writing a 1 to a bit in this register will set the event and the corresponding RIS bit also gets set. If the corresponding IMASK bit is set, then the corresponding MIS register bit also gets set.

Table 17-23. ISET2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-12	RESERVED	R	0h	Reserved
11	MEMRESIFG3	R/W	0h	Set interrupt status for MEMRES3. 0h = No new data ready. 1h = A new data is ready to be read.
10	MEMRESIFG2	R/W	0h	Set interrupt status for MEMRES2. 0h = No new data ready. 1h = A new data is ready to be read.
9	MEMRESIFG1	R/W	0h	Set interrupt status for MEMRES1. 0h = No new data ready. 1h = A new data is ready to be read.
8	MEMRESIFG0	R/W	0h	Set Interrupt status for MEMRES0. 0h = No new data ready. 1h = A new data is ready to be read.
7-0	RESERVED	R	0h	Reserved

17.5.15 ICLR2 Register (Offset = A8h) [Reset = 00000000h]

ICLR2 is shown in Table 17-24.

Return to the Summary Table.

Interrupt clear register. This register allows software to clear interrupts. Writing a 1 to a bit in this register will clear the event and the corresponding RIS bit also gets cleared. If the corresponding IMASK bit is set, then the corresponding MIS register bit also gets cleared.

Table 17-24. ICLR2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-12	RESERVED	R	0h	Reserved
11	MEMRESIFG3	R/W	0h	Clear interrupt status for MEMRES3. 0h = No new data ready. 1h = A new data is ready to be read.
10	MEMRESIFG2	R/W	0h	Clear interrupt status for MEMRES2. 0h = No new data ready. 1h = A new data is ready to be read.
9	MEMRESIFG1	R/W	0h	Clear interrupt status for MEMRES1. 0h = No new data ready. 1h = A new data is ready to be read.
8	MEMRESIFG0	R/W	0h	Clear interrupt status for MEMRES0. 0h = No new data ready. 1h = A new data is ready to be read.
7-0	RESERVED	R	0h	Reserved

17.5.16 CTL0 Register (Offset = 100h) [Reset = 00000000h]

CTL0 is shown in Table 17-25.

Return to the Summary Table.

Control Register 0

Table 17-25. CTL0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-27	RESERVED	R	0h	Reserved
26-24	SCLKDIV	R/W	0h	Sample clock divider 0h = Do not divide clock source 1h = Divide clock source by 2 2h = Divide clock source by 4 3h = Divide clock source by 8 4h = Divide clock source by 16 5h = Divide clock source by 24 6h = Divide clock source by 32 7h = Divide clock source by 48
23-17	RESERVED	R	0h	Reserved
16	PWRDN	R/W	0h	Power down policy 0h = ADC is powered down on completion of a conversion if there is no pending trigger 1h = ADC remains powered on as long as it is enabled through software.
15-1	RESERVED	R	0h	Reserved
0	ENC	R/W	0h	Enable conversion 0h = Conversion disabled. ENC change from ON to OFF will abort single or repeat sequence on a MEMCTLx boundary. The current conversion will finish and result stored in corresponding MEMRESx. 1h = Conversion enabled. ADC sequencer waits for the programmed trigger (software or hardware).

17.5.17 CTL1 Register (Offset = 104h) [Reset = 00000000h]

CTL1 is shown in Table 17-26.

Return to the Summary Table.

Control Register 1

Table 17-26. CTL1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-21	RESERVED	R	0h	Reserved
20	SAMPMODE	R/W	0h	Sample mode. This bit selects the source of the sampling signal. MANUAL option is not applicable when TRIGSRC is selected as hardware event trigger. 0h = Sample timer high phase is used as sample signal 1h = Software trigger is used as sample signal
19-18	RESERVED	R	0h	Reserved
17-16	CONSEQ	R/W	0h	Conversion sequence mode 0h = ADC channel in MEMCTLx pointed by STARTADD will be converted once 1h = ADC channel sequence pointed by STARTADD and ENDADD will be converted once 2h = ADC channel in MEMCTLx pointed by STARTADD will be converted repeatedly 3h = ADC channel sequence pointed by STARTADD and ENDADD will be converted repeatedly
15-9	RESERVED	R	0h	Reserved
8	SC	R/W	0h	Start of conversion 0h = When SAMPMODE is set to MANUAL, clearing this bit will end the sample phase and the conversion phase will start. When SAMPMODE is set to AUTO, writing 0 has no effect. 1h = When SAMPMODE is set to MANUAL, setting this bit will start the sample phase. Sample phase will last as long as this bit is set. When SAMPMODE is set to AUTO, setting this bit will trigger the timer based sample time.
7-1	RESERVED	R	0h	Reserved
0	TRIGSRC	R/W	0h	Sample trigger source 0h = Software trigger 1h = Hardware event trigger

17.5.18 CTL2 Register (Offset = 108h) [Reset = 00000000h]

CTL2 is shown in Table 17-27.

Return to the Summary Table.

Control Register 2

Table 17-27. CTL2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-29	RESERVED	R	0h	Reserved
28-24	ENDADD	R/W	0h	Sequence end address. These bits select which MEMCTLx is the last one for the sequence mode. The value of ENDADD is 0x00 to 0x03 corresponding to MEMRES0 to MEMRES3. 0h = MEMCTL0 is selected as end address of sequence. 1h = MEMCTL1 is selected as end address of sequence. 2h = MEMCTL2 is selected as end address of sequence. 3h = MEMCTL3 is selected as end address of sequence.
23-21	RESERVED	R	0h	Reserved
20-16	STARTADD	R/W	0h	Sequencer start address. These bits select which MEMCTLx is used for single conversion or as first MEMCTL for sequence mode. The value of STARTADD is 0x00 to 0x17, corresponding to MEMRES0 to MEMRES23. 0h = MEMCTL0 is selected as start address of a sequence or for a single conversion. 1h = MEMCTL1 is selected as start address of a sequence or for a single conversion. 2h = MEMCTL2 is selected as start address of a sequence or for a single conversion. 3h = MEMCTL3 is selected as start address of a sequence or for a single conversion.
15-11	RESERVED	R	0h	Reserved
10	FIFOEN	R/W	0h	Enable FIFO based operation 0h = Disable 1h = Enable
9	RESERVED	R	0h	Reserved
8	DMAEN	R/W	0h	Enable DMA trigger for data transfer. Note: DMAEN bit is cleared by hardware based on DMA done signal at the end of data transfer. Software has to re-enable DMAEN bit for ADC to generate DMA triggers. 0h = DMA trigger not enabled 1h = DMA trigger enabled
7-3	RESERVED	R	0h	Reserved
2-1	RES	R/W	0h	Resolution. These bits define the resolutoin of ADC conversion result. Note : A value of 3 defaults to 12-bits resolution. 0h = 12-bits resolution 1h = 10-bits resolution 2h = 8-bits resolution
0	DF	R/W	0h	Data read-back format. Data is always stored in binary unsigned format. 0h = Digital result reads as Binary Unsigned. 1h = Digital result reads Signed Binary. (2s complement), left aligned.

17.5.19 CTL3 Register (Offset = 10Ch) [Reset = 00000000h]

CTL3 is shown in Table 17-28.

Return to the Summary Table.

Control Register 3. This register is used to configure ADC for ad-hoc single conversion.

Table 17-28. CTL3 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-14	RESERVED	R	0h	Reserved
13-12	ASCVRSEL	R/W	0h	Selects voltage reference for ASC operation. AREF- must be connected to on-board ground when external reference option is selected. Note: Writing value 0x3 defaults to INTREF. 0h = VDDS reference 1h = External reference from AREF+/AREF- pins 2h = Internal reference
11-9	RESERVED	R	0h	Reserved
8	ASCSTIME	R/W	0h	ASC sample time compare value select. This is used to select between SCOMP0 and SCOMP1 registers for ASC operation. 0h = Select SCOMP0 1h = Select SCOMP1
7-5	RESERVED	R	0h	Reserved
4-0	ASCCHSEL	R/W	0h	ASC channel select 0h = Selects channel 0 1h = Selects channel 1 2h = Selects channel 2 3h = Selects channel 3 4h = Selects channel 4 5h = Selects channel 5 6h = Selects channel 6 7h = Selects channel 7 8h = Selects channel 8 9h = Selects channel 9 Ah = Selects channel 10 Bh = Selects channel 11 Ch = Selects channel 12 Dh = Selects channel 13 Eh = Selects channel 14 Fh = Selects channel 15

17.5.20 SCOMP0 Register (Offset = 114h) [Reset = 00000000h]

SCOMP0 is shown in Table 17-29.

Return to the Summary Table.

Sample time compare 0 register. Specifies the sample time, in number of ADC sample clock cycles. CTL0.ENC must be set to 0 to write to this register.

Table 17-29. SCOMP0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-10	RESERVED	R	0h	Reserved
9-0	VAL	R/W	0h	Specifies the number of sample clocks. When VAL = 0 or 1, number of sample clocks = Sample clock divide value. When VAL > 1, number of sample clocks = VAL x Sample clock divide value. Note: Sample clock divide value is not the value written to SCLKDIV but the actual divide value (SCLKDIV = 2 implies divide value is 4). Example: VAL = 4, SCLKDIV = 3 implies 32 sample clock cycles.

17.5.21 SCOMP1 Register (Offset = 118h) [Reset = 00000000h]

SCOMP1 is shown in Table 17-30.

Return to the Summary Table.

Sample time compare 1 register. Specifies the sample time, in number of ADC sample clock cycles. CTL0.ENC must be set to 0 to write to this register.

Table 17-30. SCOMP1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-10	RESERVED	R	0h	Reserved
9-0	VAL	R/W	0h	Specifies the number of sample clocks. When VAL = 0 or 1, number of sample clocks = Sample clock divide value. When VAL > 1, number of sample clocks = VAL x Sample clock divide value. Note: Sample clock divide value is not the value written to SCLKDIV but the actual divide value (SCLKDIV = 2 implies divide value is 4). Example: VAL = 4, SCLKDIV = 3 implies 32 sample clock cycles.

17.5.22 REFCFG Register (Offset = 11Ch) [Reset = 00000000h]

REFCFG is shown in Table 17-31.

Return to the Summary Table.

Reference buffer configuration register

Table 17-31. REFCFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-5	RESERVED	R	0h	Reserved
4-3	IBPROG	R/W	0h	Configures reference buffer bias current output value 0h = 1uA 1h = 0.5uA 2h = 2uA 3h = 0.67uA
2	SPARE	R/W	0h	Spare bit
1	REFVSEL	R/W	0h	Configures reference buffer output voltage 0h = REFBUF generates 2.5V output 1h = REFBUF generates 1.4V output
0	REFEN	R/W	0h	Reference buffer enable 0h = Disable 1h = Enable

17.5.23 WCLOW Register (Offset = 148h) [Reset = 00000000h]

WCLOW is shown in Table 17-32.

Return to the Summary Table.

Window Comparator Low Threshold Register.
The data format that is used to write and read WCLOW depends on the value of DF bit in CTL2 register.
CTL0.ENC must be 0 to write to this register.
Note: Change in ADC data format or resolution does not reset WCLOW.

Table 17-32. WCLOW Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	DATA	R/W	0h	If DF = 0, unsigned binary format has to be used. The value based on the resolution has to be right aligned with the MSB on the left. For 10-bits and 8-bits resolution, unused bits have to be 0s. If DF = 1, 2s-complement format has to be used. The value based on the resolution has to be left aligned with the LSB on the right. For 10-bits and 8-bits resolution, unused bits have to be 0s.

17.5.24 WCHIGH Register (Offset = 150h) [Reset = 00000000h]

WCHIGH is shown in Table 17-33.

Return to the Summary Table.

Window Comparator High Threshold Register.
The data format that is used to write and read WCHIGH depends on the value of DF bit in CTL2 register.
CTL0.ENC must be 0 to write to this register.
Note: Change in ADC data format or resolution does not reset WCHIGH.

Table 17-33. WCHIGH Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	DATA	R/W	0h	If DF = 0, unsigned binary format has to be used. The threshold value has to be right aligned, with the MSB on the left. For 10-bits and 8-bits resolution, unused bit have to be 0s. If DF = 1, 2s-complement format has to be used. The value based on the resolution has to be left aligned with the LSB on the right. For 10-bits and 8-bits resolution, unused bit have to be 0s.

17.5.25 FIFODATA Register (Offset = 160h) [Reset = 00000000h]

FIFODATA is shown in Table 17-34.

Return to the Summary Table.

FIFO data register. This is a virtual register used to read from FIFO.

Table 17-34. FIFODATA Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	DATA	R/W	0h	Read from this data field returns the ADC sample from FIFO.

17.5.26 ASCRES Register (Offset = 170h) [Reset = 00000000h]

ASCRES is shown in Table 17-35.

Return to the Summary Table.

ASC result register

Table 17-35. ASCRES Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	DATA	R	0h	Result of ADC ad-hoc single conversion. If DF = 0, unsigned binary: The conversion result is right aligned. In 10 and 8 bit modes, the unused MSB bits are forced to 0. If DF = 1, 2s-complement format: The conversion result is left aligned. In 10 and 8 bit modes, the unused LSB bits are forced to 0. The data is stored in the right-justified format and is converted to the left-justified 2s-complement format during read back.

17.5.27 MEMCTL0 Register (Offset = 180h) [Reset = 00000000h]

MEMCTL0 is shown in Table 17-36.

Return to the Summary Table.

Conversion Memory Control Register 0.
CTL0.ENC must be set to 0 to write to this register.

Table 17-36. MEMCTL0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-29	RESERVED	R	0h	Reserved
28	WINCOMP	R/W	0h	Enable window comparator. 0h = Disable 1h = Enable
27-25	RESERVED	R	0h	Reserved
24	TRG	R/W	0h	Trigger policy. Indicates if a trigger will be needed to step to the next MEMCTL in the sequence or to perform next conversion in the case of repeat single channel conversions. 0h = Next conversion is automatic 1h = Next conversion requires a trigger
23-13	RESERVED	R	0h	Reserved
12	STIME	R/W	0h	Selects the source of sample timer period between SCOMP0 and SCOMP1. 0h = Select SCOMP0 1h = Select SCOMP1
11-10	RESERVED	R	0h	Reserved
9-8	VRSEL	R/W	0h	Voltage reference selection. AREF- must be connected to on-board ground when external reference option is selected. Note: Writing value 0x3 defaults to INTREF. 0h = VDDS reference 1h = External reference from AREF+/AREF- pins 2h = Internal reference
7-5	RESERVED	R	0h	Reserved
4-0	CHANSEL	R/W	0h	Input channel select. 0h = Selects channel 0 1h = Selects channel 1 2h = Selects channel 2 3h = Selects channel 3 4h = Selects channel 4 5h = Selects channel 5 6h = Selects channel 6 7h = Selects channel 7 8h = Selects channel 8 9h = Selects channel 9 Ah = Selects channel 10 Bh = Selects channel 11 Ch = Selects channel 12 Dh = Selects channel 13 Eh = Selects channel 14 Fh = Selects channel 15

17.5.28 MEMCTL1 Register (Offset = 184h) [Reset = 00000000h]

MEMCTL1 is shown in Table 17-37.

Return to the Summary Table.

Conversion Memory Control Register 1.
CTL0.ENC must be set to 0 to write to this register.

Table 17-37. MEMCTL1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-29	RESERVED	R	0h	Reserved
28	WINCOMP	R/W	0h	Enable window comparator. 0h = Disable 1h = Enable
27-25	RESERVED	R	0h	Reserved
24	TRG	R/W	0h	Trigger policy. Indicates if a trigger will be needed to step to the next MEMCTL in the sequence or to perform next conversion in the case of repeat single channel conversions. 0h = Next conversion is automatic 1h = Next conversion requires a trigger
23-13	RESERVED	R	0h	Reserved
12	STIME	R/W	0h	Selects the source of sample timer period between SCOMP0 and SCOMP1. 0h = Select SCOMP0 1h = Select SCOMP1
11-10	RESERVED	R	0h	Reserved
9-8	VRSEL	R/W	0h	Voltage reference selection. AREF- must be connected to on-board ground when external reference option is selected. Note: Writing value 0x3 defaults to INTREF. 0h = VDDS reference 1h = External reference from AREF+/AREF- pins 2h = Internal reference
7-5	RESERVED	R	0h	Reserved
4-0	CHANSEL	R/W	0h	Input channel select. 0h = Selects channel 0 1h = Selects channel 1 2h = Selects channel 2 3h = Selects channel 3 4h = Selects channel 4 5h = Selects channel 5 6h = Selects channel 6 7h = Selects channel 7 8h = Selects channel 8 9h = Selects channel 9 Ah = Selects channel 10 Bh = Selects channel 11 Ch = Selects channel 12 Dh = Selects channel 13 Eh = Selects channel 14 Fh = Selects channel 15

17.5.29 MEMCTL2 Register (Offset = 188h) [Reset = 00000000h]

MEMCTL2 is shown in Table 17-38.

Return to the Summary Table.

Conversion Memory Control Register 2.
CTL0.ENC must be set to 0 to write to this register.

Table 17-38. MEMCTL2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-29	RESERVED	R	0h	Reserved
28	WINCOMP	R/W	0h	Enable window comparator. 0h = Disable 1h = Enable
27-25	RESERVED	R	0h	Reserved
24	TRG	R/W	0h	Trigger policy. Indicates if a trigger will be needed to step to the next MEMCTL in the sequence or to perform next conversion in the case of repeat single channel conversions. 0h = Next conversion is automatic 1h = Next conversion requires a trigger
23-13	RESERVED	R	0h	Reserved
12	STIME	R/W	0h	Selects the source of sample timer period between SCOMP0 and SCOMP1. 0h = Select SCOMP0 1h = Select SCOMP1
11-10	RESERVED	R	0h	Reserved
9-8	VRSEL	R/W	0h	Voltage reference selection. AREF- must be connected to on-board ground when external reference option is selected. Note: Writing value 0x3 defaults to INTREF. 0h = VDDS reference 1h = External reference from AREF+/AREF- pins 2h = Internal reference
7-5	RESERVED	R	0h	Reserved
4-0	CHANSEL	R/W	0h	Input channel select. 0h = Selects channel 0 1h = Selects channel 1 2h = Selects channel 2 3h = Selects channel 3 4h = Selects channel 4 5h = Selects channel 5 6h = Selects channel 6 7h = Selects channel 7 8h = Selects channel 8 9h = Selects channel 9 Ah = Selects channel 10 Bh = Selects channel 11 Ch = Selects channel 12 Dh = Selects channel 13 Eh = Selects channel 14 Fh = Selects channel 15

17.5.30 MEMCTL3 Register (Offset = 18Ch) [Reset = 00000000h]

MEMCTL3 is shown in Table 17-39.

Return to the Summary Table.

Conversion Memory Control Register 3.
CTL0.ENC must be set to 0 to write to this register.

Table 17-39. MEMCTL3 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-29	RESERVED	R	0h	Reserved
28	WINCOMP	R/W	0h	Enable window comparator. 0h = Disable 1h = Enable
27-25	RESERVED	R	0h	Reserved
24	TRG	R/W	0h	Trigger policy. Indicates if a trigger will be needed to step to the next MEMCTL in the sequence or to perform next conversion in the case of repeat single channel conversions. 0h = Next conversion is automatic 1h = Next conversion requires a trigger
23-13	RESERVED	R	0h	Reserved
12	STIME	R/W	0h	Selects the source of sample timer period between SCOMP0 and SCOMP1. 0h = Select SCOMP0 1h = Select SCOMP1
11-10	RESERVED	R	0h	Reserved
9-8	VRSEL	R/W	0h	Voltage reference selection. AREF- must be connected to on-board ground when external reference option is selected. Note: Writing value 0x3 defaults to INTREF. 0h = VDDS reference 1h = External reference from AREF+/AREF- pins 2h = Internal reference
7-5	RESERVED	R	0h	Reserved
4-0	CHANSEL	R/W	0h	Input channel select. 0h = Selects channel 0 1h = Selects channel 1 2h = Selects channel 2 3h = Selects channel 3 4h = Selects channel 4 5h = Selects channel 5 6h = Selects channel 6 7h = Selects channel 7 8h = Selects channel 8 9h = Selects channel 9 Ah = Selects channel 10 Bh = Selects channel 11 Ch = Selects channel 12 Dh = Selects channel 13 Eh = Selects channel 14 Fh = Selects channel 15

17.5.31 MEMRES0 Register (Offset = 280h) [Reset = 00000000h]

MEMRES0 is shown in Table 17-40.

Return to the Summary Table.

Memory Result Register 0

Table 17-40. MEMRES0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	DATA	R/W	0h	If DF = 0, unsigned binary: The conversion results are right aligned. In 10 and 8 bit modes, the unused MSB bits are forced to 0. If DF = 1, 2s-complement format: The conversion results are left aligned. In 10 and 8 bit modes, the unused LSB bits are forced to 0. The data is stored in the right-justified format and is converted to the left-justified 2s-complement format during read back.

17.5.32 MEMRES1 Register (Offset = 284h) [Reset = 00000000h]

MEMRES1 is shown in Table 17-41.

Return to the Summary Table.

Memory Result Register 1

Table 17-41. MEMRES1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	DATA	R/W	0h	If DF = 0, unsigned binary: The conversion results are right aligned. In 10 and 8 bit modes, the unused MSB bits are forced to 0. If DF = 1, 2s-complement format: The conversion results are left aligned. In 10 and 8 bit modes, the unused LSB bits are forced to 0. The data is stored in the right-justified format and is converted to the left-justified 2s-complement format during read back.

17.5.33 MEMRES2 Register (Offset = 288h) [Reset = 00000000h]

MEMRES2 is shown in Table 17-42.

Return to the Summary Table.

Memory Result Register 2

Table 17-42. MEMRES2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	DATA	R/W	0h	If DF = 0, unsigned binary: The conversion results are right aligned. In 10 and 8 bit modes, the unused MSB bits are forced to 0. If DF = 1, 2s-complement format: The conversion results are left aligned. In 10 and 8 bit modes, the unused LSB bits are forced to 0. The data is stored in the right-justified format and is converted to the left-justified 2s-complement format during read back.

17.5.34 MEMRES3 Register (Offset = 28Ch) [Reset = 00000000h]

MEMRES3 is shown in Table 17-43.

Return to the Summary Table.

Memory Result Register 3

Table 17-43. MEMRES3 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	DATA	R/W	0h	If DF = 0, unsigned binary: The conversion results are right aligned. In 10 and 8 bit modes, the unused MSB bits are forced to 0. If DF = 1, 2s-complement format: The conversion results are left aligned. In 10 and 8 bit modes, the unused LSB bits are forced to 0. The data is stored in the right-justified format and is converted to the left-justified 2s-complement format during read back.

17.5.35 STA Register (Offset = 340h) [Reset = 00000000h]

STA is shown in Table 17-44.

Return to the Summary Table.

Status Register

Table 17-44. STA Register Field Descriptions

Bit	Field	Type	Reset	Description
31-3	RESERVED	R	0h	Reserved
2	ASCACT	R	0h	ASC active 0h = Idle or done 1h = ASC active
1	RESERVED	R	0h	Reserved
0	BUSY	R	0h	Busy. This bit indicates that an active ADC sample or conversion operation is in progress. 0h = No ADC sampling or conversion in progress. 1h = ADC sampling or conversion is in progress.

17.5.36 TEST0 Register (Offset = E00h) [Reset = 00000000h]

TEST0 is shown in Table 17-45.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 17-45. TEST0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	ATEST0_EN	R/W	0h	Internal. Only to be used through TI provided API.
29	ATEST1_EN	R/W	0h	Internal. Only to be used through TI provided API.
28-13	RESERVED	R	0h	Reserved
12-8	ATEST1_MUXSEL	R/W	0h	Internal. Only to be used through TI provided API.
7-5	RESERVED	R	0h	Reserved
4-0	ATEST0_MUXSEL	R/W	0h	Internal. Only to be used through TI provided API.

17.5.37 TEST2 Register (Offset = E08h) [Reset = 00000000h]

TEST2 is shown in Table 17-46.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 17-46. TEST2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	CDAC_OVST_EN	R/W	0h	Internal. Only to be used through TI provided API.
30-25	RESERVED	R	0h	Reserved
24	LATCH_TRIM_EN	R/W	0h	Internal. Only to be used through TI provided API.
23-21	RESERVED	R	0h	Reserved
20	COMP_GAIN_TRIM	R/W	0h	Internal. Only to be used through TI provided API.
19-9	RESERVED	R	0h	Reserved
8	MUX_TEST_SEL	R/W	0h	Internal. Only to be used through TI provided API.
7-0	RESERVED	R	0h	Reserved

17.5.38 TEST3 Register (Offset = E0Ch) [Reset = 00000000h]

TEST3 is shown in Table 17-47.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 17-47. TEST3 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	CAL_ACUML	R/W	0h	Internal. Only to be used through TI provided API.

17.5.39 TEST4 Register (Offset = E10h) [Reset = 00000000h]

TEST4 is shown in Table 17-48.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 17-48. TEST4 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	HW_STEP_SEL_DIS	R/W	0h	Internal. Only to be used through TI provided API.
30-25	RESERVED	R	0h	Reserved
24	CAL_MODE_EN	R/W	0h	Internal. Only to be used through TI provided API.
23-22	RESERVED	R	0h	Reserved
21-16	CAL_STEP_SEL	R/W	0h	Internal. Only to be used through TI provided API.
15-0	RESERVED	R	0h	Reserved

17.5.40 TEST5 Register (Offset = E14h) [Reset = 00000000h]

TEST5 is shown in Table 17-49.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 17-49. TEST5 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-10	RESERVED	R	0h	Reserved
9-0	CAL_CAP_CTL	R/W	0h	Internal. Only to be used through TI provided API.

17.5.41 TEST6 Register (Offset = E18h) [Reset = 00000000h]

TEST6 is shown in Table 17-50.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 17-50. TEST6 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3-0	ATESTSEL	R/W	0h	Internal. Only to be used through TI provided API.

17.5.42 DEBUG1 Register (Offset = E20h) [Reset = 00801000h]

DEBUG1 is shown in Table 17-51.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 17-51. DEBUG1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	CTRL	R/W	00801000h	Internal. Only to be used through TI provided API.

17.5.43 DEBUG2 Register (Offset = E24h) [Reset = 00000000h]

DEBUG2 is shown in Table 17-52.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 17-52. DEBUG2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-30	RESERVED	R	0h	Reserved
29-28	VTOI_CTRL	R/W	0h	Internal. Only to be used through TI provided API.
27-25	RESERVED	R	0h	Reserved
24	VTOI_TESTMODE_EN	R/W	0h	Internal. Only to be used through TI provided API.
23-0	RESERVED	R	0h	Reserved

17.5.44 DEBUG3 Register (Offset = E28h) [Reset = 00000000h]

DEBUG3 is shown in Table 17-53.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 17-53. DEBUG3 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Reserved
5	DEC1_DIS	R/W	0h	Internal. Only to be used through TI provided API.
4	DEC0_DIS	R/W	0h	Internal. Only to be used through TI provided API.
3-1	RESERVED	R	0h	Reserved
0	BOOST_ENZ	R/W	0h	Internal. Only to be used through TI provided API.

17.5.45 DEBUG4 Register (Offset = E2Ch) [Reset = 00000000h]

DEBUG4 is shown in Table 17-54.

Return to the Summary Table.

Internal. Only to be used through TI provided API.

Table 17-54. DEBUG4 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	ADC_CTRL0	R/W	0h	Internal. Only to be used through TI provided API.

Chapter 18

I/O Controller (IOC)

This chapter describes the input/output controller (IOC) and the general-purpose inputs and outputs (GPIOs).

18.1 Introduction

The I/O controller configures I/O pins and maps peripheral signals to physical pins (DIOx). This chapter explains the I/O controller functions and gives examples on how to map peripheral functions to the pins chosen by the user.

- Each pin can be mapped to a specific set of peripherals.
- GPIO is the base function where the IOC.IOC*n*[2:0] PORTCFG bit field is set to 0x0.
- DIOn (DIO0 to DIO25) are the logical names of the different I/O pins on the specific package, see the device-specific data sheet for more information on package pin designation.
- 12 of these DIOs also have analog capabilities.
- Pins can also be mapped to the digital test bus (DTB) to bring out clocks or physical signals like interrupts.
- The device-specific data sheet provides:
  - Mapping between DIOn and pins for the different packages
  - Peripheral pin mapping

18.2 Block Diagram

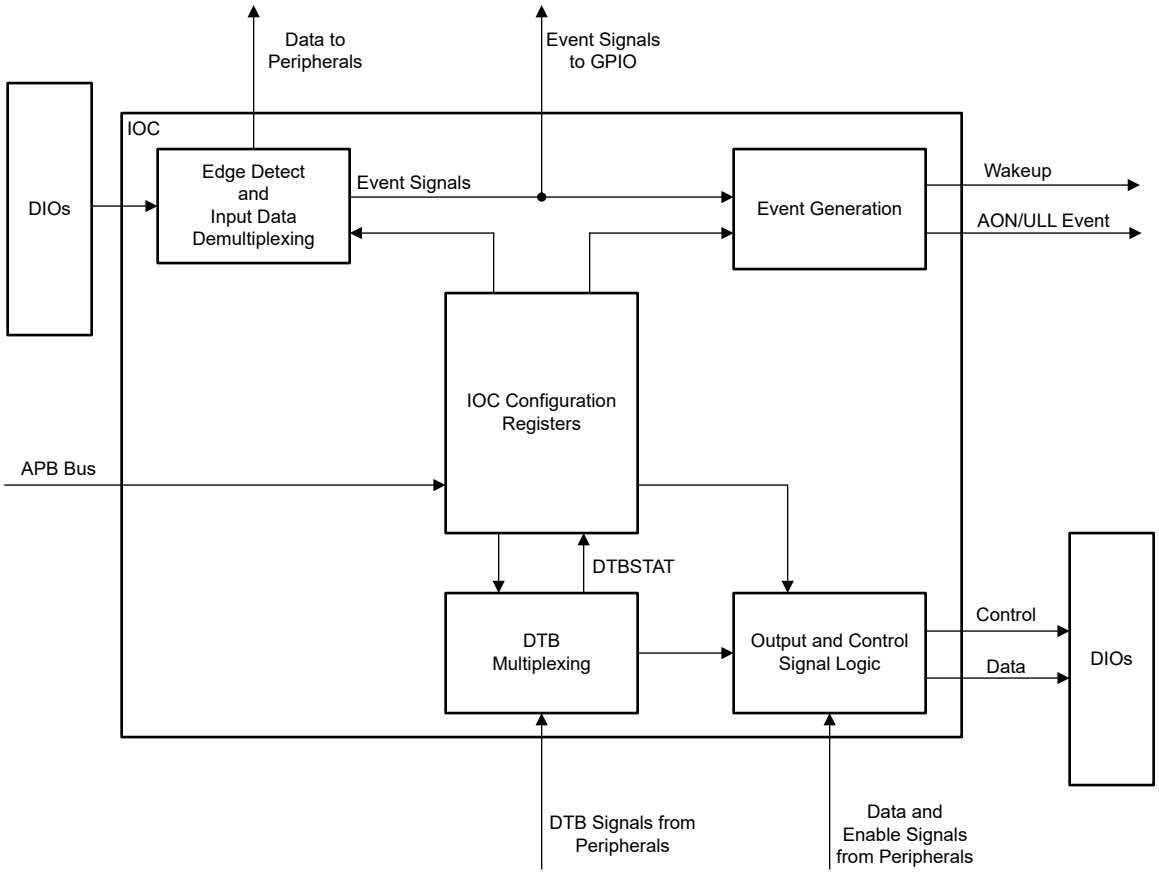


Figure 18-1. IOC Simplified Block Diagram

18.3 I/O Mapping and Configuration

Each peripheral with pin functions can be mapped to a specific set of pins. Refer to the device specific data sheet for the possible mappings.

18.3.1 Basic I/O Mapping

To map a peripheral function to a DIOn, where n can range from 0 to 25, set IOC:IOCn.PORTCFG to the value that represents the target function. For example, to set DIO0 to the base function (GPIO) set IOC:IOC0.PORTCFG = 0x0.

18.3.2 Radio GPO

There are eight data output signals from the device radio named LRFDn where n is from 0 to 7. After selecting the IOC function for these signals, configure the LRFDDBELL registers GPOSEL0 or GPOSEL1 to select the source for those signals. See Chapter 22 for more information.

For example:

- Configure DIO3 for LRFD0 by setting IOC.IOC3[2:0] PORTCFG = 0x3 (See Pin Mapping .)
- Select source for LRFD0 by setting LRFDDBELL.GPOSEL0[0:4] SRC0 bit field.

18.3.3 Pin Mapping

See the device specific data sheet for peripheral and I/O mapping and capability.

18.3.4 DTB Muxing

Internal clocks and interrupts can be brought out to I/O pins through the digital test bus (DTB). For more information on configuring the DTB for clock or interrupt signals see Chapter 6 and Chapter 4.

Note

Due to the way data is latched into the DTBSTAT register, a single read is not always accurate. The value of DTBSTAT can only be relied on if two subsequent register reads indicate the same value.

Note

When observing clocks using the DTB the first clock cycle can have an incorrect duty cycle. This can also occur if the clocks that are being observed are switched, or if the internal clock divider values are updated.

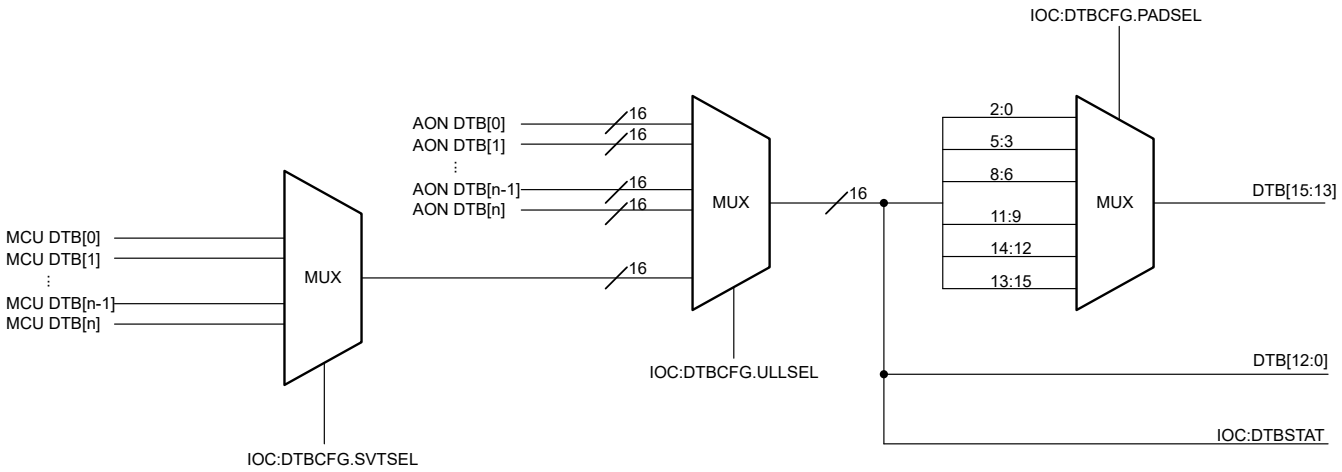


Figure 18-2. DTB Multiplexing

Table 18-1. MCU/SVT Signals Select

SVTSEL	Module
0	Disabled
1	Internal Use Only
2	Internal Use Only
3	Internal Use Only

Table 18-1. MCU/SVT Signals Select (continued)

SVTSEL	Module
4	Internal Use Only
5	Internal Use Only
6	Internal Use Only
7	Internal Use Only
8	Internal Use Only
9	Internal Use Only
10	Internal Use Only
11	Internal Use Only
12	Internal Use Only
13	Internal Use Only
14	EVTSVT
15	Internal Use Only
16	Internal Use Only
17	Internal Use Only
18	Internal Use Only

Table 18-2. AON/ULL Signals Select

ULLSEL	Module
0	SVT/MCU DTB
1	CKMD
2	EVTULL
3	Internal Use Only
4	Internal Use Only
5	Internal Use Only
6	Internal Use Only
7	Internal Use Only
8	Internal Use Only
9	Internal Use Only

18.4 Edge Detection

The IOC supports detecting rising, falling, or both rising and falling edges.

When an edge is detected on a DIO, the IOC publishes an event to the AON event fabric if IOC.EVTCFG is configured to publish the event. Only one DIOn can be selected to generate an event on the AON event fabric. The event flag is cleared by the user by clearing IOC.EVTCFG[8] EVTIFG.

The IOC can also generate a wake-up from standby signal to PMCTL on edge detection by setting IOC.IOC*n*[18] WUENSB. Any or all DIOn can be selected to generate a wake-up from standby on edge detection.

Note

Care must be taken to ensure that spurious edges are not generated while configuring the edge detection feature. The general recommendation is to first disable the event/interrupt enable setting, then configure the IP as needed, clear the event/interrupt flag, and set the event/interrupt enable.

18.5 GPIO

The MCU GPIO is a general-purpose input/output module that allows software to write to and read from the DIOs. GPIO supports up to 26 programmable I/O pins. These pins are configured by the IOC module. To modify

a single GPIO output value, use the GPIO.DOUT ***n*** registers. The following describes the necessary steps to set up DIO1 as a GPIO output and toggle the bit.

TI recommends using the GPIO driver in the SimpleLink™ CC23xx Software Development Kit (SDK) when managing general purpose I/Os.

1. Map DIO1 as a GPIO output by setting the IOC.IOC1[2:0] PORTCFG bit field to 0 (Base function - GPIO).
2. Set DIO1 as output by clearing the IOC.IOC1[29] INPEN bit. More port configurations can also be set in the IOC.IOC1 register (for more details, see Section 18.6).
3. Set the data output enable bit for DIO1 in the GPIO.DOE31_0[1] DIO1 bit field by issuing a read-modify-write operation.
4. Toggle the DIO1 output by writing a 1 to the GPIO.DOUTTGL31_0[1] DIO1 bit.

18.6 I/O Pins

The IOC allows software to configure the pins based on the requirements of the application. The software can configure different characteristic settings for any or all of the I/O pins. All of the following features are controlled in the IOC:IOCn registers:

- **Drive Strength**(IOC:IOCFGn.IOSTR)
  - Configures the output drive strength of an I/O pin. By setting the IOC:IOCn IOSTR bit to 0x0, the drive strength is automatically updated based on inputs from the battery monitor, BATMON, to maintain current output across the VDDS range.
- **Drive Current**(IOC.IOCFGn[11:10] IOCURR)
  - Configures the maximum current of an I/O pin. See the device-specific data sheet for individual I/O current capability.
- **Pull Control**(IOC:IOCn.PULLCTL)
  - Configures a weak pull on an I/O pin. The following can be set: pullup, pulldown, or no pull. See the data sheet for specific pullup and pulldown current.
- **Slew Control** (IOC:IOCn.SLEWRED)
  - Sets normal or reduced slew rate on an I/O pin
- **Hysteresis** (IOC:IOCFGn.HYSTEN)
  - Enables or disables input hysteresis on an I/O pin
- **Open-Source or Open-Drain Configuration** (IOC:IOCn.IOMODE)
  - Configures the pin as normal, open source, or open drain. All of these configurations can be set to either inverted or normal (non-inverted).
- **Edge Detection** (IOC:IOCn.EDGEDET)
  - Enables edge detection on I/O pin. The following modes are supported:
    - Rising edge
    - Falling edge
    - Trigger on both rising and falling
    - No edge detection
  - Edge detection can be used for event generation on the AON event fabric and a wake from standby signal to PMCTL.
- **Wake from Shutdown** (IOC:IOCn.WUCFGSD)
  - Enables wake-up from shutdown. The following modes are supported:
    - Wake on low, wake-up triggered when the pin level is low.
    - Wake on high, wake-up triggered when the pin level is high.
- **Input Driver** (IOC:IOCn.INPEN)
  - Enables or disables the I/O input driver
- **Pin Configuration** (IOC.IOCn[2:0] PORTCFG)
  - Selects the function of the pin. See the device specific data sheet for available functionality per I/O pin.

18.7 Unused Pins

By default, the I/O driver (output) and input buffer (input) are disabled (tri-state mode) at power on or reset, and thus the I/O pin can safely be left unconnected (floating). If the I/O pin is in a tri-state condition and connected to a node with a different voltage potential, a small leakage current can go through the pin. The same applies to an I/O pin configured as input, where the pin is connected to a voltage source (for example VDD/2). The input is then an undefined value of either 0 or 1.

18.8 Debug Configuration

When the DBGSS.DBGCTL[5] SWDCEN bit is set, DIOs corresponding to the SWDIO and SWCLK pads are connected to the IceMelter wakeup circuit. SWDCEN is set to enable a debug connection. See Chapter 5 for more information on the Debug Subsystem.

When using SWD debug, configure the IOC. IOC*n*[2:0] PORTCFG bit field of the DIOs corresponding to SWDIO and SWCLK to a value 0x0 (GPIO) before setting the DBGSS.DBGCTL[5] SWDCEN bit and do not write to GPIO.DOUT*n* and GPIO.DOE*n* registers. This prevents data from IOC interfering with IceMelter operation and SWDIO and SWCLK data from interfering with peripherals.

When DBGSS.DBGCTL[5] SWDCEN is cleared, DIOs corresponding to pads SWDIO and SWCLK are connected to IOC (instead of Icemelter). Always clear SWDCEN before using the DIOs corresponding to SWDIO and SWCLK for non-debug purposes, to avoid possible timing violations within IceMelter.

18.9 IOC Registers

Table 18-3 lists the memory-mapped registers for the IOC registers. All register offset addresses not listed in Table 18-3 should be considered as reserved locations and the register contents should not be modified.

Table 18-3. IOC Registers

Offset	Acronym	Register Name	Section
0h	DESC	Module Description	Go
4h	DESCEX	Extended Module Description	Go
100h	IOC0	Configuration	Go
104h	IOC1	Configuration	Go
108h	IOC2	Configuration	Go
10Ch	IOC3	Configuration	Go
110h	IOC4	Configuration	Go
114h	IOC5	Configuration	Go
118h	IOC6	Configuration	Go
11Ch	IOC7	Configuration	Go
120h	IOC8	Configuration	Go
124h	IOC9	Configuration	Go
128h	IOC10	Configuration	Go
12Ch	IOC11	Configuration	Go
130h	IOC12	Configuration	Go
134h	IOC13	Configuration	Go
138h	IOC14	Configuration	Go
13Ch	IOC15	Configuration	Go
140h	IOC16	Configuration	Go
144h	IOC17	Configuration	Go
148h	IOC18	Configuration	Go
14Ch	IOC19	Configuration	Go
150h	IOC20	Configuration	Go
154h	IOC21	Configuration	Go
158h	IOC22	Configuration	Go
15Ch	IOC23	Configuration	Go
160h	IOC24	Configuration	Go
164h	IOC25	Configuration	Go
C00h	DTBCFG	DTB configuration	Go
C04h	DTBOE	DTB output enable	Go
C08h	EVTCFG	Event configuration	Go
C0Ch	TEST	Test	Go
C10h	DTBSTAT	DTB status	Go

Complex bit access types are encoded to fit into small table cells. Table 18-4 shows the codes that are used for access types in this section.

Table 18-4. IOC Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		

Table 18-4. IOC Access Type Codes (continued)

Access Type	Code	Description
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

18.9.1 DESC Register (Offset = 0h) [Reset = D4401010h]

DESC is shown in Table 18-5.

Return to the Summary Table.

Description Register. This register provides IP module ID, revision information, instance index and standard MMR registers offset.

Table 18-5. DESC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	MODID	R/W	D440h	Module identifier used to uniquely identify this IP.
15-12	STDIPOFF	R/W	1h	Standard IP MMR block offset. Standard IP MMRs are the set of from aggregated IRQ registers till DTB. 0: Standard IP MMRs do not exist 0x1-0xF: Standard IP MMRs begin at offset of (64*STDIPOFF from the base IP address)
11-8	INSTIDX	R/W	0h	IP Instance ID number. If multiple instances of IP exist in the device, this field can identify the instance number (0-15).
7-4	MAJREV	R/W	1h	Major revision of IP (0-15).
3-0	MINREV	R/W	0h	Minor revision of IP (0-15).

18.9.2 DESCEX Register (Offset = 4h) [Reset = 0000F2D9h]

DESCEX is shown in Table 18-6.

Return to the Summary Table.

Extended Description Register. This register provides configuration details of the IP to software drivers and end users.

Table 18-6. DESCEX Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-12	NUMDTBIO	R/W	Fh	Number of DTB IOs supported. Total DTB IOs supported is NUMDTBIO value +1. 0h = Smallest value Fh = Highest possible value
11-7	NUMHDIO	R/W	5h	Number of high drive IOs supported. Total high drive IOs supported is NUMHDIO value +1. 0h = Smallest value 1Fh = Highest possible value
6	HDIO	R/W	1h	High drive IO supported by IOC. 0h = HD IO not supported by IOC 1h = HD IO supported by IOC
5-0	NUMDIO	R/W	19h	Number of DIOs supported. Total DIOs supported is NUMDIO value +1. 0h = Smallest value 3Fh = Highest possible value

18.9.3 IOC0 Register (Offset = 100h) [Reset = 00000000h]

IOC0 is shown in Table 18-7.

Return to the Summary Table.

Configuration of DIO0

Table 18-7. IOC0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	HYSTEN	R/W	0h	This field controls input hysteresis 0h = Input hysteresis disabled 1h = Input hysteresis enabled
29	INPEN	R/W	0h	This field controls the input capability of DIO 0h = Input disabled 1h = Input enabled
28-27	RESERVED	R	0h	Reserved
26-24	IOMODE	R/W	0h	IO Mode. Setting this to value 0x6 or 0x7 will default to normal IO behavior. 0h = Normal IO 1h = Inverted IO 2h = Open Drain, normal IO 3h = Open Drain, inverted IO 4h = Open Source, normal IO 5h = Open Source, inverted IO
23-22	RESERVED	R	0h	Reserved
21-20	WUCFGSD	R/W	0h	Wakeup configuration from shutdown 0h = Wakeup disabled 1h = Wakeup disabled 2h = Wakeup triggered when pad level is low 3h = Wakeup triggered when pad level is high
19	RESERVED	R	0h	Reserved
18	WUENSB	R/W	0h	Wakeup enable from standby 0h = Wakeup disabled 1h = Wakeup enabled (effective only if EDGEDET is enabled)
17-16	EDGEDET	R/W	0h	Edge detect configuration 0h = No edge detection 1h = Negative edge detection 2h = Positive edge detection 3h = Positive and negative edge detection
15	RESERVED	R	0h	Reserved
14-13	PULLCTL	R/W	0h	Pull control. Setting this to value 0x3 disables pull. 0h = No pull 1h = Pull down enabled 2h = Pull up enabled
12-3	RESERVED	R	0h	Reserved
2-0	PORTCFG	R/W	0h	Selects usage of DIO0 0h = Base function 1h = Digital peripheral function-1 2h = Digital peripheral function-2 3h = Digital peripheral function-3 4h = Digital peripheral function-4 5h = Digital peripheral function-5 6h = Analog function 7h = Digital Test Bus function

18.9.4 IOC1 Register (Offset = 104h) [Reset = 00000000h]

IOC1 is shown in Table 18-8.

Return to the Summary Table.

Configuration of DIO1

Table 18-8. IOC1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	HYSTEN	R/W	0h	This field controls input hysteresis 0h = Input hysteresis disabled 1h = Input hysteresis enabled
29	INPEN	R/W	0h	This field controls the input capability of DIO 0h = Input disabled 1h = Input enabled
28-27	RESERVED	R	0h	Reserved
26-24	IOMODE	R/W	0h	IO Mode. Setting this to value 0x6 or 0x7 will default to normal IO behavior. 0h = Normal IO 1h = Inverted IO 2h = Open Drain, normal IO 3h = Open Drain, inverted IO 4h = Open Source, normal IO 5h = Open Source, inverted IO
23-22	RESERVED	R	0h	Reserved
21-20	WUCFGSD	R/W	0h	Wakeup configuration from shutdown 0h = Wakeup disabled 1h = Wakeup disabled 2h = Wakeup triggered when pad level is low 3h = Wakeup triggered when pad level is high
19	RESERVED	R	0h	Reserved
18	WUENSB	R/W	0h	Wakeup enable from standby 0h = Wakeup disabled 1h = Wakeup enabled (effective only if EDGEDET is enabled)
17-16	EDGEDET	R/W	0h	Edge detect configuration 0h = No edge detection 1h = Negative edge detection 2h = Positive edge detection 3h = Positive and negative edge detection
15	RESERVED	R	0h	Reserved
14-13	PULLCTL	R/W	0h	Pull control. Setting this to value 0x3 disables pull. 0h = No pull 1h = Pull down enabled 2h = Pull up enabled
12-3	RESERVED	R	0h	Reserved
2-0	PORTCFG	R/W	0h	Selects usage of DIO1 0h = Base function 1h = Digital peripheral function-1 2h = Digital peripheral function-2 3h = Digital peripheral function-3 4h = Digital peripheral function-4 5h = Digital peripheral function-5 6h = Analog function 7h = Digital Test Bus function

18.9.5 IOC2 Register (Offset = 108h) [Reset = 00000000h]

IOC2 is shown in Table 18-9.

Return to the Summary Table.

Selects usage of DIO2

Table 18-9. IOC2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	HYSTEN	R/W	0h	This field controls input hysteresis 0h = Input hysteresis disabled 1h = Input hysteresis enabled
29	INPEN	R/W	0h	This field controls the input capability of DIO 0h = Input disabled 1h = Input enabled
28-27	RESERVED	R	0h	Reserved
26-24	IOMODE	R/W	0h	IO Mode. Setting this to value 0x6 or 0x7 will default to normal IO behavior. 0h = Normal IO 1h = Inverted IO 2h = Open Drain, normal IO 3h = Open Drain, inverted IO 4h = Open Source, normal IO 5h = Open Source, inverted IO
23-22	RESERVED	R	0h	Reserved
21-20	WUCFGSD	R/W	0h	Wakeup configuration from shutdown 0h = Wakeup disabled 1h = Wakeup disabled 2h = Wakeup triggered when pad level is low 3h = Wakeup triggered when pad level is high
19	RESERVED	R	0h	Reserved
18	WUENSB	R/W	0h	Wakeup enable from standby 0h = Wakeup disabled 1h = Wakeup enabled (effective only if EDGEDET is enabled)
17-16	EDGEDET	R/W	0h	Edge detect configuration 0h = No edge detection 1h = Negative edge detection 2h = Positive edge detection 3h = Positive and negative edge detection
15	RESERVED	R	0h	Reserved
14-13	PULLCTL	R/W	0h	Pull control. Setting this to value 0x3 disables pull. 0h = No pull 1h = Pull down enabled 2h = Pull up enabled
12-3	RESERVED	R	0h	Reserved
2-0	PORTCFG	R/W	0h	Selects usage of DIO2 0h = Base function 1h = Digital peripheral function-1 2h = Digital peripheral function-2 3h = Digital peripheral function-3 4h = Digital peripheral function-4 5h = Digital peripheral function-5 6h = Analog function 7h = Digital Test Bus function

18.9.6 IOC3 Register (Offset = 10Ch) [Reset = 00000000h]

IOC3 is shown in Table 18-10.

Return to the Summary Table.

Configuration of DIO3

Table 18-10. IOC3 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	HYSTEN	R/W	0h	This field controls input hysteresis 0h = Input hysteresis disabled 1h = Input hysteresis enabled
29	INPEN	R/W	0h	This field controls the input capability of DIO 0h = Input disabled 1h = Input enabled
28-27	RESERVED	R	0h	Reserved
26-24	IOMODE	R/W	0h	IO Mode. Setting this to value 0x6 or 0x7 will default to normal IO behavior. 0h = Normal IO 1h = Inverted IO 2h = Open Drain, normal IO 3h = Open Drain, inverted IO 4h = Open Source, normal IO 5h = Open Source, inverted IO
23-22	RESERVED	R	0h	Reserved
21-20	WUCFGSD	R/W	0h	Wakeup configuration from shutdown 0h = Wakeup disabled 1h = Wakeup disabled 2h = Wakeup triggered when pad level is low 3h = Wakeup triggered when pad level is high
19	RESERVED	R	0h	Reserved
18	WUENSB	R/W	0h	Wakeup enable from standby 0h = Wakeup disabled 1h = Wakeup enabled (effective only if EDGEDET is enabled)
17-16	EDGEDET	R/W	0h	Edge detect configuration 0h = No edge detection 1h = Negative edge detection 2h = Positive edge detection 3h = Positive and negative edge detection
15	RESERVED	R	0h	Reserved
14-13	PULLCTL	R/W	0h	Pull control. Setting this to value 0x3 disables pull. 0h = No pull 1h = Pull down enabled 2h = Pull up enabled
12-3	RESERVED	R	0h	Reserved
2-0	PORTCFG	R/W	0h	Selects usage of DIO3 0h = Base function 1h = Digital peripheral function-1 2h = Digital peripheral function-2 3h = Digital peripheral function-3 4h = Digital peripheral function-4 5h = Digital peripheral function-5 6h = Analog function 7h = Digital Test Bus function

18.9.7 IOC4 Register (Offset = 110h) [Reset = 00000000h]

IOC4 is shown in Table 18-11.

Return to the Summary Table.

Configuration of DIO4

Table 18-11. IOC4 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	HYSTEN	R/W	0h	This field controls input hysteresis 0h = Input hysteresis disabled 1h = Input hysteresis enabled
29	INPEN	R/W	0h	This field controls the input capability of DIO 0h = Input disabled 1h = Input enabled
28-27	RESERVED	R	0h	Reserved
26-24	IOMODE	R/W	0h	IO Mode. Setting this to value 0x6 or 0x7 will default to normal IO behavior. 0h = Normal IO 1h = Inverted IO 2h = Open Drain, normal IO 3h = Open Drain, inverted IO 4h = Open Source, normal IO 5h = Open Source, inverted IO
23-22	RESERVED	R	0h	Reserved
21-20	WUCFGSD	R/W	0h	Wakeup configuration from shutdown 0h = Wakeup disabled 1h = Wakeup disabled 2h = Wakeup triggered when pad level is low 3h = Wakeup triggered when pad level is high
19	RESERVED	R	0h	Reserved
18	WUENSB	R/W	0h	Wakeup enable from standby 0h = Wakeup disabled 1h = Wakeup enabled (effective only if EDGEDET is enabled)
17-16	EDGEDET	R/W	0h	Edge detect configuration 0h = No edge detection 1h = Negative edge detection 2h = Positive edge detection 3h = Positive and negative edge detection
15	RESERVED	R	0h	Reserved
14-13	PULLCTL	R/W	0h	Pull control. Setting this to value 0x3 disables pull. 0h = No pull 1h = Pull down enabled 2h = Pull up enabled
12-3	RESERVED	R	0h	Reserved
2-0	PORTCFG	R/W	0h	Selects usage of DIO4 0h = Base function 1h = Digital peripheral function-1 2h = Digital peripheral function-2 3h = Digital peripheral function-3 4h = Digital peripheral function-4 5h = Digital peripheral function-5 6h = Analog function 7h = Digital Test Bus function

18.9.8 IOC5 Register (Offset = 114h) [Reset = 00000000h]

IOC5 is shown in Table 18-12.

Return to the Summary Table.

Configuration of DIO5

Table 18-12. IOC5 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	HYSTEN	R/W	0h	This field controls input hysteresis 0h = Input hysteresis disabled 1h = Input hysteresis enabled
29	INPEN	R/W	0h	This field controls the input capability of DIO 0h = Input disabled 1h = Input enabled
28-27	RESERVED	R	0h	Reserved
26-24	IOMODE	R/W	0h	IO Mode. Setting this to value 0x6 or 0x7 will default to normal IO behavior. 0h = Normal IO 1h = Inverted IO 2h = Open Drain, normal IO 3h = Open Drain, inverted IO 4h = Open Source, normal IO 5h = Open Source, inverted IO
23-22	RESERVED	R	0h	Reserved
21-20	WUCFGSD	R/W	0h	Wakeup configuration from shutdown 0h = Wakeup disabled 1h = Wakeup disabled 2h = Wakeup triggered when pad level is low 3h = Wakeup triggered when pad level is high
19	RESERVED	R	0h	Reserved
18	WUENSB	R/W	0h	Wakeup enable from standby 0h = Wakeup disabled 1h = Wakeup enabled (effective only if EDGEDET is enabled)
17-16	EDGEDET	R/W	0h	Edge detect configuration 0h = No edge detection 1h = Negative edge detection 2h = Positive edge detection 3h = Positive and negative edge detection
15	RESERVED	R	0h	Reserved
14-13	PULLCTL	R/W	0h	Pull control. Setting this to value 0x3 disables pull. 0h = No pull 1h = Pull down enabled 2h = Pull up enabled
12-3	RESERVED	R	0h	Reserved
2-0	PORTCFG	R/W	0h	Selects usage of DIO5 0h = Base function 1h = Digital peripheral function-1 2h = Digital peripheral function-2 3h = Digital peripheral function-3 4h = Digital peripheral function-4 5h = Digital peripheral function-5 6h = Analog function 7h = Digital Test Bus function

18.9.9 IOC6 Register (Offset = 118h) [Reset = 00000000h]

IOC6 is shown in Table 18-13.

Return to the Summary Table.

Configuration of DIO6

Table 18-13. IOC6 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	HYSTEN	R/W	0h	This field controls input hysteresis 0h = Input hysteresis disabled 1h = Input hysteresis enabled
29	INPEN	R/W	0h	This field controls the input capability of DIO 0h = Input disabled 1h = Input enabled
28-27	RESERVED	R	0h	Reserved
26-24	IOMODE	R/W	0h	IO Mode. Setting this to value 0x6 or 0x7 will default to normal IO behavior. 0h = Normal IO 1h = Inverted IO 2h = Open Drain, normal IO 3h = Open Drain, inverted IO 4h = Open Source, normal IO 5h = Open Source, inverted IO
23-22	RESERVED	R	0h	Reserved
21-20	WUCFGSD	R/W	0h	Wakeup configuration from shutdown 0h = Wakeup disabled 1h = Wakeup disabled 2h = Wakeup triggered when pad level is low 3h = Wakeup triggered when pad level is high
19	RESERVED	R	0h	Reserved
18	WUENSB	R/W	0h	Wakeup enable from standby 0h = Wakeup disabled 1h = Wakeup enabled (effective only if EDGEDET is enabled)
17-16	EDGEDET	R/W	0h	Edge detect configuration 0h = No edge detection 1h = Negative edge detection 2h = Positive edge detection 3h = Positive and negative edge detection
15	RESERVED	R	0h	Reserved
14-13	PULLCTL	R/W	0h	Pull control. Setting this to value 0x3 disables pull. 0h = No pull 1h = Pull down enabled 2h = Pull up enabled
12-3	RESERVED	R	0h	Reserved
2-0	PORTCFG	R/W	0h	Selects usage of DIO6 0h = Base function 1h = Digital peripheral function-1 2h = Digital peripheral function-2 3h = Digital peripheral function-3 4h = Digital peripheral function-4 5h = Digital peripheral function-5 6h = Analog function 7h = Digital Test Bus function

18.9.10 IOC7 Register (Offset = 11Ch) [Reset = 00000000h]

IOC7 is shown in Table 18-14.

Return to the Summary Table.

Configuration of DIO7

Table 18-14. IOC7 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	HYSTEN	R/W	0h	This field controls input hysteresis 0h = Input hysteresis disabled 1h = Input hysteresis enabled
29	INPEN	R/W	0h	This field controls the input capability of DIO 0h = Input disabled 1h = Input enabled
28-27	RESERVED	R	0h	Reserved
26-24	IOMODE	R/W	0h	IO Mode. Setting this to value 0x6 or 0x7 will default to normal IO behavior. 0h = Normal IO 1h = Inverted IO 2h = Open Drain, normal IO 3h = Open Drain, inverted IO 4h = Open Source, normal IO 5h = Open Source, inverted IO
23-22	RESERVED	R	0h	Reserved
21-20	WUCFGSD	R/W	0h	Wakeup configuration from shutdown 0h = Wakeup disabled 1h = Wakeup disabled 2h = Wakeup triggered when pad level is low 3h = Wakeup triggered when pad level is high
19	RESERVED	R	0h	Reserved
18	WUENSB	R/W	0h	Wakeup enable from standby 0h = Wakeup disabled 1h = Wakeup enabled (effective only if EDGEDET is enabled)
17-16	EDGEDET	R/W	0h	Edge detect configuration 0h = No edge detection 1h = Negative edge detection 2h = Positive edge detection 3h = Positive and negative edge detection
15	RESERVED	R	0h	Reserved
14-13	PULLCTL	R/W	0h	Pull control. Setting this to value 0x3 disables pull. 0h = No pull 1h = Pull down enabled 2h = Pull up enabled
12-3	RESERVED	R	0h	Reserved
2-0	PORTCFG	R/W	0h	Selects usage of DIO7 0h = Base function 1h = Digital peripheral function-1 2h = Digital peripheral function-2 3h = Digital peripheral function-3 4h = Digital peripheral function-4 5h = Digital peripheral function-5 6h = Analog function 7h = Digital Test Bus function

18.9.11 IOC8 Register (Offset = 120h) [Reset = 00000000h]

IOC8 is shown in Table 18-15.

Return to the Summary Table.

Configuration of DIO8

Table 18-15. IOC8 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	HYSTEN	R/W	0h	This field controls input hysteresis 0h = Input hysteresis disabled 1h = Input hysteresis enabled
29	INPEN	R/W	0h	This field controls the input capability of DIO 0h = Input disabled 1h = Input enabled
28-27	RESERVED	R	0h	Reserved
26-24	IOMODE	R/W	0h	IO Mode. Setting this to value 0x6 or 0x7 will default to normal IO behavior. 0h = Normal IO 1h = Inverted IO 2h = Open Drain, normal IO 3h = Open Drain, inverted IO 4h = Open Source, normal IO 5h = Open Source, inverted IO
23-22	RESERVED	R	0h	Reserved
21-20	WUCFGSD	R/W	0h	Wakeup configuration from shutdown 0h = Wakeup disabled 1h = Wakeup disabled 2h = Wakeup triggered when pad level is low 3h = Wakeup triggered when pad level is high
19	RESERVED	R	0h	Reserved
18	WUENSB	R/W	0h	Wakeup enable from standby 0h = Wakeup disabled 1h = Wakeup enabled (effective only if EDGEDET is enabled)
17-16	EDGEDET	R/W	0h	Edge detect configuration 0h = No edge detection 1h = Negative edge detection 2h = Positive edge detection 3h = Positive and negative edge detection
15	RESERVED	R	0h	Reserved
14-13	PULLCTL	R/W	0h	Pull control. Setting this to value 0x3 disables pull. 0h = No pull 1h = Pull down enabled 2h = Pull up enabled
12-3	RESERVED	R	0h	Reserved
2-0	PORTCFG	R/W	0h	Selects usage of DIO8 0h = Base function 1h = Digital peripheral function-1 2h = Digital peripheral function-2 3h = Digital peripheral function-3 4h = Digital peripheral function-4 5h = Digital peripheral function-5 6h = Analog function 7h = Digital Test Bus function

18.9.12 IOC9 Register (Offset = 124h) [Reset = 00000000h]

IOC9 is shown in Table 18-16.

Return to the Summary Table.

Configuration of DIO9

Table 18-16. IOC9 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	HYSTEN	R/W	0h	This field controls input hysteresis 0h = Input hysteresis disabled 1h = Input hysteresis enabled
29	INPEN	R/W	0h	This field controls the input capability of DIO 0h = Input disabled 1h = Input enabled
28-27	RESERVED	R	0h	Reserved
26-24	IOMODE	R/W	0h	IO Mode. Setting this to value 0x6 or 0x7 will default to normal IO behavior. 0h = Normal IO 1h = Inverted IO 2h = Open Drain, normal IO 3h = Open Drain, inverted IO 4h = Open Source, normal IO 5h = Open Source, inverted IO
23-22	RESERVED	R	0h	Reserved
21-20	WUCFGSD	R/W	0h	Wakeup configuration from shutdown 0h = Wakeup disabled 1h = Wakeup disabled 2h = Wakeup triggered when pad level is low 3h = Wakeup triggered when pad level is high
19	RESERVED	R	0h	Reserved
18	WUENSB	R/W	0h	Wakeup enable from standby 0h = Wakeup disabled 1h = Wakeup enabled (effective only if EDGEDET is enabled)
17-16	EDGEDET	R/W	0h	Edge detect configuration 0h = No edge detection 1h = Negative edge detection 2h = Positive edge detection 3h = Positive and negative edge detection
15	RESERVED	R	0h	Reserved
14-13	PULLCTL	R/W	0h	Pull control. Setting this to value 0x3 disables pull. 0h = No pull 1h = Pull down enabled 2h = Pull up enabled
12-3	RESERVED	R	0h	Reserved
2-0	PORTCFG	R/W	0h	Selects usage of DIO9 0h = Base function 1h = Digital peripheral function-1 2h = Digital peripheral function-2 3h = Digital peripheral function-3 4h = Digital peripheral function-4 5h = Digital peripheral function-5 6h = Analog function 7h = Digital Test Bus function

18.9.13 IOC10 Register (Offset = 128h) [Reset = 00000000h]

IOC10 is shown in Table 18-17.

Return to the Summary Table.

Configuration of DIO10

Table 18-17. IOC10 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	HYSTEN	R/W	0h	This field controls input hysteresis 0h = Input hysteresis disabled 1h = Input hysteresis enabled
29	INPEN	R/W	0h	This field controls the input capability of DIO 0h = Input disabled 1h = Input enabled
28-27	RESERVED	R	0h	Reserved
26-24	IOMODE	R/W	0h	IO Mode. Setting this to value 0x6 or 0x7 will default to normal IO behavior. 0h = Normal IO 1h = Inverted IO 2h = Open Drain, normal IO 3h = Open Drain, inverted IO 4h = Open Source, normal IO 5h = Open Source, inverted IO
23-22	RESERVED	R	0h	Reserved
21-20	WUCFGSD	R/W	0h	Wakeup configuration from shutdown 0h = Wakeup disabled 1h = Wakeup disabled 2h = Wakeup triggered when pad level is low 3h = Wakeup triggered when pad level is high
19	RESERVED	R	0h	Reserved
18	WUENSB	R/W	0h	Wakeup enable from standby 0h = Wakeup disabled 1h = Wakeup enabled (effective only if EDGEDET is enabled)
17-16	EDGEDET	R/W	0h	Edge detect configuration 0h = No edge detection 1h = Negative edge detection 2h = Positive edge detection 3h = Positive and negative edge detection
15	RESERVED	R	0h	Reserved
14-13	PULLCTL	R/W	0h	Pull control. Setting this to value 0x3 disables pull. 0h = No pull 1h = Pull down enabled 2h = Pull up enabled
12-3	RESERVED	R	0h	Reserved
2-0	PORTCFG	R/W	0h	Selects usage of DIO10 0h = Base function 1h = Digital peripheral function-1 2h = Digital peripheral function-2 3h = Digital peripheral function-3 4h = Digital peripheral function-4 5h = Digital peripheral function-5 6h = Analog function 7h = Digital Test Bus function

18.9.14 IOC11 Register (Offset = 12Ch) [Reset = 00000000h]

IOC11 is shown in Table 18-18.

Return to the Summary Table.

Configuration of DIO11

Table 18-18. IOC11 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	HYSTEN	R/W	0h	This field controls input hysteresis 0h = Input hysteresis disabled 1h = Input hysteresis enabled
29	INPEN	R/W	0h	This field controls the input capability of DIO 0h = Input disabled 1h = Input enabled
28-27	RESERVED	R	0h	Reserved
26-24	IOMODE	R/W	0h	IO Mode. Setting this to value 0x6 or 0x7 will default to normal IO behavior. 0h = Normal IO 1h = Inverted IO 2h = Open Drain, normal IO 3h = Open Drain, inverted IO 4h = Open Source, normal IO 5h = Open Source, inverted IO
23-22	RESERVED	R	0h	Reserved
21-20	WUCFGSD	R/W	0h	Wakeup configuration from shutdown 0h = Wakeup disabled 1h = Wakeup disabled 2h = Wakeup triggered when pad level is low 3h = Wakeup triggered when pad level is high
19	RESERVED	R	0h	Reserved
18	WUENSB	R/W	0h	Wakeup enable from standby 0h = Wakeup disabled 1h = Wakeup enabled (effective only if EDGEDET is enabled)
17-16	EDGEDET	R/W	0h	Edge detect configuration 0h = No edge detection 1h = Negative edge detection 2h = Positive edge detection 3h = Positive and negative edge detection
15	RESERVED	R	0h	Reserved
14-13	PULLCTL	R/W	0h	Pull control. Setting this to value 0x3 disables pull. 0h = No pull 1h = Pull down enabled 2h = Pull up enabled
12-3	RESERVED	R	0h	Reserved
2-0	PORTCFG	R/W	0h	Selects usage of DIO11 0h = Base function 1h = Digital peripheral function-1 2h = Digital peripheral function-2 3h = Digital peripheral function-3 4h = Digital peripheral function-4 5h = Digital peripheral function-5 6h = Analog function 7h = Digital Test Bus function

18.9.15 IOC12 Register (Offset = 130h) [Reset = 00000000h]

IOC12 is shown in Table 18-19.

Return to the Summary Table.

Configuration of DIO12

Table 18-19. IOC12 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	HYSTEN	R/W	0h	This field controls input hysteresis 0h = Input hysteresis disabled 1h = Input hysteresis enabled
29	INPEN	R/W	0h	This field controls the input capability of DIO 0h = Input disabled 1h = Input enabled
28-27	RESERVED	R	0h	Reserved
26-24	IOMODE	R/W	0h	IO Mode. Setting this to value 0x6 or 0x7 will default to normal IO behavior. 0h = Normal IO 1h = Inverted IO 2h = Open Drain, normal IO 3h = Open Drain, inverted IO 4h = Open Source, normal IO 5h = Open Source, inverted IO
23-22	RESERVED	R	0h	Reserved
21-20	WUCFGSD	R/W	0h	Wakeup configuration from shutdown 0h = Wakeup disabled 1h = Wakeup disabled 2h = Wakeup triggered when pad level is low 3h = Wakeup triggered when pad level is high
19	RESERVED	R	0h	Reserved
18	WUENSB	R/W	0h	Wakeup enable from standby 0h = Wakeup disabled 1h = Wakeup enabled (effective only if EDGEDET is enabled)
17-16	EDGEDET	R/W	0h	Edge detect configuration 0h = No edge detection 1h = Negative edge detection 2h = Positive edge detection 3h = Positive and negative edge detection
15	RESERVED	R	0h	Reserved
14-13	PULLCTL	R/W	0h	Pull control. Setting this to value 0x3 disables pull. 0h = No pull 1h = Pull down enabled 2h = Pull up enabled
12	SLEWRED	R/W	0h	Slew rate configuration 0h = Normal slew rate 1h = Reduced slew rate
11-10	IOCURR	R/W	0h	Output current configuration. Writing value 0x3 defaults to 2mA current setting. 0h = 2mA 1h = 4mA 2h = 8mA
9-8	IOSTR	R/W	0h	Drive strength configuration 0h = Automatic drive strength adjustment 1h = Minimum drive stregnth 2h = Medium drive strength 3h = Maximum drive strength
7-3	RESERVED	R	0h	Reserved

Table 18-19. IOC12 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
2-0	PORTCFG	R/W	0h	Selects usage of DIO12 0h = Base function 1h = Digital peripheral function-1 2h = Digital peripheral function-2 3h = Digital peripheral function-3 4h = Digital peripheral function-4 5h = Digital peripheral function-5 6h = Analog function 7h = Digital Test Bus function

18.9.16 IOC13 Register (Offset = 134h) [Reset = 00000000h]

IOC13 is shown in Table 18-20.

Return to the Summary Table.

Configuration of DIO13

Table 18-20. IOC13 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	HYSTEN	R/W	0h	This field controls input hysteresis 0h = Input hysteresis disabled 1h = Input hysteresis enabled
29	INPEN	R/W	0h	This field controls the input capability of DIO 0h = Input disabled 1h = Input enabled
28-27	RESERVED	R	0h	Reserved
26-24	IOMODE	R/W	0h	IO Mode. Setting this to value 0x6 or 0x7 will default to normal IO behavior. 0h = Normal IO 1h = Inverted IO 2h = Open Drain, normal IO 3h = Open Drain, inverted IO 4h = Open Source, normal IO 5h = Open Source, inverted IO
23-22	RESERVED	R	0h	Reserved
21-20	WUCFGSD	R/W	0h	Wakeup configuration from shutdown 0h = Wakeup disabled 1h = Wakeup disabled 2h = Wakeup triggered when pad level is low 3h = Wakeup triggered when pad level is high
19	RESERVED	R	0h	Reserved
18	WUENSB	R/W	0h	Wakeup enable from standby 0h = Wakeup disabled 1h = Wakeup enabled (effective only if EDGEDET is enabled)
17-16	EDGEDET	R/W	0h	Edge detect configuration 0h = No edge detection 1h = Negative edge detection 2h = Positive edge detection 3h = Positive and negative edge detection
15	RESERVED	R	0h	Reserved
14-13	PULLCTL	R/W	0h	Pull control. Setting this to value 0x3 disables pull. 0h = No pull 1h = Pull down enabled 2h = Pull up enabled
12-3	RESERVED	R	0h	Reserved
2-0	PORTCFG	R/W	0h	Selects usage of DIO13 0h = Base function 1h = Digital peripheral function-1 2h = Digital peripheral function-2 3h = Digital peripheral function-3 4h = Digital peripheral function-4 5h = Digital peripheral function-5 6h = Analog function 7h = Digital Test Bus function

18.9.17 IOC14 Register (Offset = 138h) [Reset = 00000000h]

IOC14 is shown in Table 18-21.

Return to the Summary Table.

Configuration of DIO14

Table 18-21. IOC14 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	HYSTEN	R/W	0h	This field controls input hysteresis 0h = Input hysteresis disabled 1h = Input hysteresis enabled
29	INPEN	R/W	0h	This field controls the input capability of DIO 0h = Input disabled 1h = Input enabled
28-27	RESERVED	R	0h	Reserved
26-24	IOMODE	R/W	0h	IO Mode. Setting this to value 0x6 or 0x7 will default to normal IO behavior. 0h = Normal IO 1h = Inverted IO 2h = Open Drain, normal IO 3h = Open Drain, inverted IO 4h = Open Source, normal IO 5h = Open Source, inverted IO
23-22	RESERVED	R	0h	Reserved
21-20	WUCFGSD	R/W	0h	Wakeup configuration from shutdown 0h = Wakeup disabled 1h = Wakeup disabled 2h = Wakeup triggered when pad level is low 3h = Wakeup triggered when pad level is high
19	RESERVED	R	0h	Reserved
18	WUENSB	R/W	0h	Wakeup enable from standby 0h = Wakeup disabled 1h = Wakeup enabled (effective only if EDGEDET is enabled)
17-16	EDGEDET	R/W	0h	Edge detect configuration 0h = No edge detection 1h = Negative edge detection 2h = Positive edge detection 3h = Positive and negative edge detection
15	RESERVED	R	0h	Reserved
14-13	PULLCTL	R/W	0h	Pull control. Setting this to value 0x3 disables pull. 0h = No pull 1h = Pull down enabled 2h = Pull up enabled
12-3	RESERVED	R	0h	Reserved
2-0	PORTCFG	R/W	0h	Selects usage of DIO14 0h = Base function 1h = Digital peripheral function-1 2h = Digital peripheral function-2 3h = Digital peripheral function-3 4h = Digital peripheral function-4 5h = Digital peripheral function-5 6h = Analog function 7h = Digital Test Bus function

18.9.18 IOC15 Register (Offset = 13Ch) [Reset = 00000000h]

IOC15 is shown in Table 18-22.

Return to the Summary Table.

Configuration of DIO15

Table 18-22. IOC15 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	HYSTEN	R/W	0h	This field controls input hysteresis 0h = Input hysteresis disabled 1h = Input hysteresis enabled
29	INPEN	R/W	0h	This field controls the input capability of DIO 0h = Input disabled 1h = Input enabled
28-27	RESERVED	R	0h	Reserved
26-24	IOMODE	R/W	0h	IO Mode. Setting this to value 0x6 or 0x7 will default to normal IO behavior. 0h = Normal IO 1h = Inverted IO 2h = Open Drain, normal IO 3h = Open Drain, inverted IO 4h = Open Source, normal IO 5h = Open Source, inverted IO
23-22	RESERVED	R	0h	Reserved
21-20	WUCFGSD	R/W	0h	Wakeup configuration from shutdown 0h = Wakeup disabled 1h = Wakeup disabled 2h = Wakeup triggered when pad level is low 3h = Wakeup triggered when pad level is high
19	RESERVED	R	0h	Reserved
18	WUENSB	R/W	0h	Wakeup enable from standby 0h = Wakeup disabled 1h = Wakeup enabled (effective only if EDGEDET is enabled)
17-16	EDGEDET	R/W	0h	Edge detect configuration 0h = No edge detection 1h = Negative edge detection 2h = Positive edge detection 3h = Positive and negative edge detection
15	RESERVED	R	0h	Reserved
14-13	PULLCTL	R/W	0h	Pull control. Setting this to value 0x3 disables pull. 0h = No pull 1h = Pull down enabled 2h = Pull up enabled
12-3	RESERVED	R	0h	Reserved
2-0	PORTCFG	R/W	0h	Selects usage of DIO15 0h = Base function 1h = Digital peripheral function-1 2h = Digital peripheral function-2 3h = Digital peripheral function-3 4h = Digital peripheral function-4 5h = Digital peripheral function-5 6h = Analog function 7h = Digital Test Bus function

18.9.19 IOC16 Register (Offset = 140h) [Reset = 00004000h]

IOC16 is shown in Table 18-23.

Return to the Summary Table.

Configuration of DIO16

Table 18-23. IOC16 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	HYSTEN	R/W	0h	This field controls input hysteresis 0h = Input hysteresis disabled 1h = Input hysteresis enabled
29	INPEN	R/W	0h	This field controls the input capability of DIO 0h = Input disabled 1h = Input enabled
28-27	RESERVED	R	0h	Reserved
26-24	IOMODE	R/W	0h	IO Mode. Setting this to value 0x6 or 0x7 will default to normal IO behavior. 0h = Normal IO 1h = Inverted IO 2h = Open Drain, normal IO 3h = Open Drain, inverted IO 4h = Open Source, normal IO 5h = Open Source, inverted IO
23-22	RESERVED	R	0h	Reserved
21-20	WUCFGSD	R/W	0h	Wakeup configuration from shutdown 0h = Wakeup disabled 1h = Wakeup disabled 2h = Wakeup triggered when pad level is low 3h = Wakeup triggered when pad level is high
19	RESERVED	R	0h	Reserved
18	WUENSB	R/W	0h	Wakeup enable from standby 0h = Wakeup disabled 1h = Wakeup enabled (effective only if EDGEDET is enabled)
17-16	EDGEDET	R/W	0h	Edge detect configuration 0h = No edge detection 1h = Negative edge detection 2h = Positive edge detection 3h = Positive and negative edge detection
15	RESERVED	R	0h	Reserved
14-13	PULLCTL	R/W	2h	Pull control. Setting this to value 0x3 disables pull. 0h = No pull 1h = Pull down enabled 2h = Pull up enabled
12	SLEWRED	R/W	0h	Slew rate configuration 0h = Normal slew rate 1h = Reduced slew rate
11-10	IOCURR	R/W	0h	Output current configuration. Writing value 0x3 defaults to 2mA current setting. 0h = 2mA 1h = 4mA 2h = 8mA
9-8	IOSTR	R/W	0h	Drive strength configuration 0h = Automatic drive strength adjustment 1h = Minimum drive stregnth 2h = Medium drive strength 3h = Maximum drive strength
7-3	RESERVED	R	0h	Reserved

Table 18-23. IOC16 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
2-0	PORTCFG	R/W	0h	Selects usage of DIO16 0h = Base function 1h = Digital peripheral function-1 2h = Digital peripheral function-2 3h = Digital peripheral function-3 4h = Digital peripheral function-4 5h = Digital peripheral function-5 6h = Analog function 7h = Digital Test Bus function

18.9.20 IOC17 Register (Offset = 144h) [Reset = 00002000h]

IOC17 is shown in Table 18-24.

Return to the Summary Table.

Configuration of DIO17

Table 18-24. IOC17 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	HYSTEN	R/W	0h	This field controls input hysteresis 0h = Input hysteresis disabled 1h = Input hysteresis enabled
29	INPEN	R/W	0h	This field controls the input capability of DIO 0h = Input disabled 1h = Input enabled
28-27	RESERVED	R	0h	Reserved
26-24	IOMODE	R/W	0h	IO Mode. Setting this to value 0x6 or 0x7 will default to normal IO behavior. 0h = Normal IO 1h = Inverted IO 2h = Open Drain, normal IO 3h = Open Drain, inverted IO 4h = Open Source, normal IO 5h = Open Source, inverted IO
23-22	RESERVED	R	0h	Reserved
21-20	WUCFGSD	R/W	0h	Wakeup configuration from shutdown 0h = Wakeup disabled 1h = Wakeup disabled 2h = Wakeup triggered when pad level is low 3h = Wakeup triggered when pad level is high
19	RESERVED	R	0h	Reserved
18	WUENSB	R/W	0h	Wakeup enable from standby 0h = Wakeup disabled 1h = Wakeup enabled (effective only if EDGEDET is enabled)
17-16	EDGEDET	R/W	0h	Edge detect configuration 0h = No edge detection 1h = Negative edge detection 2h = Positive edge detection 3h = Positive and negative edge detection
15	RESERVED	R	0h	Reserved
14-13	PULLCTL	R/W	1h	Pull control. Setting this to value 0x3 disables pull. 0h = No pull 1h = Pull down enabled 2h = Pull up enabled
12	SLEWRED	R/W	0h	Slew rate configuration 0h = Normal slew rate 1h = Reduced slew rate
11-10	IOCURR	R/W	0h	Output current configuration. Writing value 0x3 defaults to 2mA current setting. 0h = 2mA 1h = 4mA 2h = 8mA
9-8	IOSTR	R/W	0h	Drive strength configuration 0h = Automatic drive strength adjustment 1h = Minimum drive stregnth 2h = Medium drive strength 3h = Maximum drive strength
7-3	RESERVED	R	0h	Reserved

Table 18-24. IOC17 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
2-0	PORTCFG	R/W	0h	Selects usage of DIO17 0h = Base function 1h = Digital peripheral function-1 2h = Digital peripheral function-2 3h = Digital peripheral function-3 4h = Digital peripheral function-4 5h = Digital peripheral function-5 6h = Analog function 7h = Digital Test Bus function

18.9.21 IOC18 Register (Offset = 148h) [Reset = 00000000h]

IOC18 is shown in Table 18-25.

Return to the Summary Table.

Configuration of DIO18

Table 18-25. IOC18 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	HYSTEN	R/W	0h	This field controls input hysteresis 0h = Input hysteresis disabled 1h = Input hysteresis enabled
29	INPEN	R/W	0h	This field controls the input capability of DIO 0h = Input disabled 1h = Input enabled
28-27	RESERVED	R	0h	Reserved
26-24	IOMODE	R/W	0h	IO Mode. Setting this to value 0x6 or 0x7 will default to normal IO behavior. 0h = Normal IO 1h = Inverted IO 2h = Open Drain, normal IO 3h = Open Drain, inverted IO 4h = Open Source, normal IO 5h = Open Source, inverted IO
23-22	RESERVED	R	0h	Reserved
21-20	WUCFGSD	R/W	0h	Wakeup configuration from shutdown 0h = Wakeup disabled 1h = Wakeup disabled 2h = Wakeup triggered when pad level is low 3h = Wakeup triggered when pad level is high
19	RESERVED	R	0h	Reserved
18	WUENSB	R/W	0h	Wakeup enable from standby 0h = Wakeup disabled 1h = Wakeup enabled (effective only if EDGEDET is enabled)
17-16	EDGEDET	R/W	0h	Edge detect configuration 0h = No edge detection 1h = Negative edge detection 2h = Positive edge detection 3h = Positive and negative edge detection
15	RESERVED	R	0h	Reserved
14-13	PULLCTL	R/W	0h	Pull control. Setting this to value 0x3 disables pull. 0h = No pull 1h = Pull down enabled 2h = Pull up enabled
12	SLEWRED	R/W	0h	Slew rate configuration 0h = Normal slew rate 1h = Reduced slew rate
11-10	IOCURR	R/W	0h	Output current configuration. Writing value 0x3 defaults to 2mA current setting. 0h = 2mA 1h = 4mA 2h = 8mA
9-8	IOSTR	R/W	0h	Drive strength configuration 0h = Automatic drive strength adjustment 1h = Minimum drive stregnth 2h = Medium drive strength 3h = Maximum drive strength
7-3	RESERVED	R	0h	Reserved

Table 18-25. IOC18 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
2-0	PORTCFG	R/W	0h	Selects usage of DIO18 0h = Base function 1h = Digital peripheral function-1 2h = Digital peripheral function-2 3h = Digital peripheral function-3 4h = Digital peripheral function-4 5h = Digital peripheral function-5 6h = Analog function 7h = Digital Test Bus function

18.9.22 IOC19 Register (Offset = 14Ch) [Reset = 00000000h]

IOC19 is shown in Table 18-26.

Return to the Summary Table.

Configuration of DIO19

Table 18-26. IOC19 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	HYSTEN	R/W	0h	This field controls input hysteresis 0h = Input hysteresis disabled 1h = Input hysteresis enabled
29	INPEN	R/W	0h	This field controls the input capability of DIO 0h = Input disabled 1h = Input enabled
28-27	RESERVED	R	0h	Reserved
26-24	IOMODE	R/W	0h	IO Mode. Setting this to value 0x6 or 0x7 will default to normal IO behavior. 0h = Normal IO 1h = Inverted IO 2h = Open Drain, normal IO 3h = Open Drain, inverted IO 4h = Open Source, normal IO 5h = Open Source, inverted IO
23-22	RESERVED	R	0h	Reserved
21-20	WUCFGSD	R/W	0h	Wakeup configuration from shutdown 0h = Wakeup disabled 1h = Wakeup disabled 2h = Wakeup triggered when pad level is low 3h = Wakeup triggered when pad level is high
19	RESERVED	R	0h	Reserved
18	WUENSB	R/W	0h	Wakeup enable from standby 0h = Wakeup disabled 1h = Wakeup enabled (effective only if EDGEDET is enabled)
17-16	EDGEDET	R/W	0h	Edge detect configuration 0h = No edge detection 1h = Negative edge detection 2h = Positive edge detection 3h = Positive and negative edge detection
15	RESERVED	R	0h	Reserved
14-13	PULLCTL	R/W	0h	Pull control. Setting this to value 0x3 disables pull. 0h = No pull 1h = Pull down enabled 2h = Pull up enabled
12	SLEWRED	R/W	0h	Slew rate configuration 0h = Normal slew rate 1h = Reduced slew rate
11-10	IOCURR	R/W	0h	Output current configuration. Writing value 0x3 defaults to 2mA current setting. 0h = 2mA 1h = 4mA 2h = 8mA
9-8	IOSTR	R/W	0h	Drive strength configuration 0h = Automatic drive strength adjustment 1h = Minimum drive stregnth 2h = Medium drive strength 3h = Maximum drive strength
7-3	RESERVED	R	0h	Reserved

Table 18-26. IOC19 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
2-0	PORTCFG	R/W	0h	Selects usage of DIO19 0h = Base function 1h = Digital peripheral function-1 2h = Digital peripheral function-2 3h = Digital peripheral function-3 4h = Digital peripheral function-4 5h = Digital peripheral function-5 6h = Analog function 7h = Digital Test Bus function

18.9.23 IOC20 Register (Offset = 150h) [Reset = 00000000h]

IOC20 is shown in Table 18-27.

Return to the Summary Table.

Configuration of DIO20

Table 18-27. IOC20 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	HYSTEN	R/W	0h	This field controls input hysteresis 0h = Input hysteresis disabled 1h = Input hysteresis enabled
29	INPEN	R/W	0h	This field controls the input capability of DIO 0h = Input disabled 1h = Input enabled
28-27	RESERVED	R	0h	Reserved
26-24	IOMODE	R/W	0h	IO Mode. Setting this to value 0x6 or 0x7 will default to normal IO behavior. 0h = Normal IO 1h = Inverted IO 2h = Open Drain, normal IO 3h = Open Drain, inverted IO 4h = Open Source, normal IO 5h = Open Source, inverted IO
23-22	RESERVED	R	0h	Reserved
21-20	WUCFGSD	R/W	0h	Wakeup configuration from shutdown 0h = Wakeup disabled 1h = Wakeup disabled 2h = Wakeup triggered when pad level is low 3h = Wakeup triggered when pad level is high
19	RESERVED	R	0h	Reserved
18	WUENSB	R/W	0h	Wakeup enable from standby 0h = Wakeup disabled 1h = Wakeup enabled (effective only if EDGEDET is enabled)
17-16	EDGEDET	R/W	0h	Edge detect configuration 0h = No edge detection 1h = Negative edge detection 2h = Positive edge detection 3h = Positive and negative edge detection
15	RESERVED	R	0h	Reserved
14-13	PULLCTL	R/W	0h	Pull control. Setting this to value 0x3 disables pull. 0h = No pull 1h = Pull down enabled 2h = Pull up enabled
12-3	RESERVED	R	0h	Reserved
2-0	PORTCFG	R/W	0h	Selects usage of DIO20 0h = Base function 1h = Digital peripheral function-1 2h = Digital peripheral function-2 3h = Digital peripheral function-3 4h = Digital peripheral function-4 5h = Digital peripheral function-5 6h = Analog function 7h = Digital Test Bus function

18.9.24 IOC21 Register (Offset = 154h) [Reset = 00000000h]

IOC21 is shown in Table 18-28.

Return to the Summary Table.

Configuration of DIO21

Table 18-28. IOC21 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	HYSTEN	R/W	0h	This field controls input hysteresis 0h = Input hysteresis disabled 1h = Input hysteresis enabled
29	INPEN	R/W	0h	This field controls the input capability of DIO 0h = Input disabled 1h = Input enabled
28-27	RESERVED	R	0h	Reserved
26-24	IOMODE	R/W	0h	IO Mode. Setting this to value 0x6 or 0x7 will default to normal IO behavior. 0h = Normal IO 1h = Inverted IO 2h = Open Drain, normal IO 3h = Open Drain, inverted IO 4h = Open Source, normal IO 5h = Open Source, inverted IO
23-22	RESERVED	R	0h	Reserved
21-20	WUCFGSD	R/W	0h	Wakeup configuration from shutdown 0h = Wakeup disabled 1h = Wakeup disabled 2h = Wakeup triggered when pad level is low 3h = Wakeup triggered when pad level is high
19	RESERVED	R	0h	Reserved
18	WUENSB	R/W	0h	Wakeup enable from standby 0h = Wakeup disabled 1h = Wakeup enabled (effective only if EDGEDET is enabled)
17-16	EDGEDET	R/W	0h	Edge detect configuration 0h = No edge detection 1h = Negative edge detection 2h = Positive edge detection 3h = Positive and negative edge detection
15	RESERVED	R	0h	Reserved
14-13	PULLCTL	R/W	0h	Pull control. Setting this to value 0x3 disables pull. 0h = No pull 1h = Pull down enabled 2h = Pull up enabled
12-3	RESERVED	R	0h	Reserved
2-0	PORTCFG	R/W	0h	Selects usage of DIO21 0h = Base function 1h = Digital peripheral function-1 2h = Digital peripheral function-2 3h = Digital peripheral function-3 4h = Digital peripheral function-4 5h = Digital peripheral function-5 6h = Analog function 7h = Digital Test Bus function

18.9.25 IOC22 Register (Offset = 158h) [Reset = 00000000h]

IOC22 is shown in Table 18-29.

Return to the Summary Table.

Configuration of DIO22

Table 18-29. IOC22 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	HYSTEN	R/W	0h	This field controls input hysteresis 0h = Input hysteresis disabled 1h = Input hysteresis enabled
29	INPEN	R/W	0h	This field controls the input capability of DIO 0h = Input disabled 1h = Input enabled
28-27	RESERVED	R	0h	Reserved
26-24	IOMODE	R/W	0h	IO Mode. Setting this to value 0x6 or 0x7 will default to normal IO behavior. 0h = Normal IO 1h = Inverted IO 2h = Open Drain, normal IO 3h = Open Drain, inverted IO 4h = Open Source, normal IO 5h = Open Source, inverted IO
23-22	RESERVED	R	0h	Reserved
21-20	WUCFGSD	R/W	0h	Wakeup configuration from shutdown 0h = Wakeup disabled 1h = Wakeup disabled 2h = Wakeup triggered when pad level is low 3h = Wakeup triggered when pad level is high
19	RESERVED	R	0h	Reserved
18	WUENSB	R/W	0h	Wakeup enable from standby 0h = Wakeup disabled 1h = Wakeup enabled (effective only if EDGEDET is enabled)
17-16	EDGEDET	R/W	0h	Edge detect configuration 0h = No edge detection 1h = Negative edge detection 2h = Positive edge detection 3h = Positive and negative edge detection
15	RESERVED	R	0h	Reserved
14-13	PULLCTL	R/W	0h	Pull control. Setting this to value 0x3 disables pull. 0h = No pull 1h = Pull down enabled 2h = Pull up enabled
12-3	RESERVED	R	0h	Reserved
2-0	PORTCFG	R/W	0h	Selects usage of DIO22 0h = Base function 1h = Digital peripheral function-1 2h = Digital peripheral function-2 3h = Digital peripheral function-3 4h = Digital peripheral function-4 5h = Digital peripheral function-5 6h = Analog function 7h = Digital Test Bus function

18.9.26 IOC23 Register (Offset = 15Ch) [Reset = 00000000h]

IOC23 is shown in Table 18-30.

Return to the Summary Table.

Configuration of DIO23

Table 18-30. IOC23 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	HYSTEN	R/W	0h	This field controls input hysteresis 0h = Input hysteresis disabled 1h = Input hysteresis enabled
29	INPEN	R/W	0h	This field controls the input capability of DIO 0h = Input disabled 1h = Input enabled
28-27	RESERVED	R	0h	Reserved
26-24	IOMODE	R/W	0h	IO Mode. Setting this to value 0x6 or 0x7 will default to normal IO behavior. 0h = Normal IO 1h = Inverted IO 2h = Open Drain, normal IO 3h = Open Drain, inverted IO 4h = Open Source, normal IO 5h = Open Source, inverted IO
23-22	RESERVED	R	0h	Reserved
21-20	WUCFGSD	R/W	0h	Wakeup configuration from shutdown 0h = Wakeup disabled 1h = Wakeup disabled 2h = Wakeup triggered when pad level is low 3h = Wakeup triggered when pad level is high
19	RESERVED	R	0h	Reserved
18	WUENSB	R/W	0h	Wakeup enable from standby 0h = Wakeup disabled 1h = Wakeup enabled (effective only if EDGEDET is enabled)
17-16	EDGEDET	R/W	0h	Edge detect configuration 0h = No edge detection 1h = Negative edge detection 2h = Positive edge detection 3h = Positive and negative edge detection
15	RESERVED	R	0h	Reserved
14-13	PULLCTL	R/W	0h	Pull control. Setting this to value 0x3 disables pull. 0h = No pull 1h = Pull down enabled 2h = Pull up enabled
12-3	RESERVED	R	0h	Reserved
2-0	PORTCFG	R/W	0h	Selects usage of DIO23 0h = Base function 1h = Digital peripheral function-1 2h = Digital peripheral function-2 3h = Digital peripheral function-3 4h = Digital peripheral function-4 5h = Digital peripheral function-5 6h = Analog function 7h = Digital Test Bus function

18.9.27 IOC24 Register (Offset = 160h) [Reset = 00000000h]

IOC24 is shown in Table 18-31.

Return to the Summary Table.

Configuration of DIO24

Table 18-31. IOC24 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	HYSTEN	R/W	0h	This field controls input hysteresis 0h = Input hysteresis disabled 1h = Input hysteresis enabled
29	INPEN	R/W	0h	This field controls the input capability of DIO 0h = Input disabled 1h = Input enabled
28-27	RESERVED	R	0h	Reserved
26-24	IOMODE	R/W	0h	IO Mode. Setting this to value 0x6 or 0x7 will default to normal IO behavior. 0h = Normal IO 1h = Inverted IO 2h = Open Drain, normal IO 3h = Open Drain, inverted IO 4h = Open Source, normal IO 5h = Open Source, inverted IO
23-22	RESERVED	R	0h	Reserved
21-20	WUCFGSD	R/W	0h	Wakeup configuration from shutdown 0h = Wakeup disabled 1h = Wakeup disabled 2h = Wakeup triggered when pad level is low 3h = Wakeup triggered when pad level is high
19	RESERVED	R	0h	Reserved
18	WUENSB	R/W	0h	Wakeup enable from standby 0h = Wakeup disabled 1h = Wakeup enabled (effective only if EDGEDET is enabled)
17-16	EDGEDET	R/W	0h	Edge detect configuration 0h = No edge detection 1h = Negative edge detection 2h = Positive edge detection 3h = Positive and negative edge detection
15	RESERVED	R	0h	Reserved
14-13	PULLCTL	R/W	0h	Pull control. Setting this to value 0x3 disables pull. 0h = No pull 1h = Pull down enabled 2h = Pull up enabled
12	SLEWRED	R/W	0h	Slew rate configuration 0h = Normal slew rate 1h = Reduced slew rate
11-10	IOCURR	R/W	0h	Output current configuration. Writing value 0x3 defaults to 2mA current setting. 0h = 2mA 1h = 4mA 2h = 8mA
9-8	IOSTR	R/W	0h	Drive strength configuration 0h = Automatic drive strength adjustment 1h = Minimum drive stregnth 2h = Medium drive strength 3h = Maximum drive strength
7-3	RESERVED	R	0h	Reserved

Table 18-31. IOC24 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
2-0	PORTCFG	R/W	0h	Selects usage of DIO24 0h = Base function 1h = Digital peripheral function-1 2h = Digital peripheral function-2 3h = Digital peripheral function-3 4h = Digital peripheral function-4 5h = Digital peripheral function-5 6h = Analog function 7h = Digital Test Bus function

18.9.28 IOC25 Register (Offset = 164h) [Reset = 00000000h]

IOC25 is shown in Table 18-32.

Return to the Summary Table.

Selects usage of DIO25

Table 18-32. IOC25 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	HYSTEN	R/W	0h	This field controls input hysteresis 0h = Input hysteresis disabled 1h = Input hysteresis enabled
29	INPEN	R/W	0h	This field controls the input capability of DIO 0h = Input disabled 1h = Input enabled
28-27	RESERVED	R	0h	Reserved
26-24	IOMODE	R/W	0h	IO Mode. Setting this to value 0x6 or 0x7 will default to normal IO behavior. 0h = Normal IO 1h = Inverted IO 2h = Open Drain, normal IO 3h = Open Drain, inverted IO 4h = Open Source, normal IO 5h = Open Source, inverted IO
23-22	RESERVED	R	0h	Reserved
21-20	WUCFGSD	R/W	0h	Wakeup configuration from shutdown 0h = Wakeup disabled 1h = Wakeup disabled 2h = Wakeup triggered when pad level is low 3h = Wakeup triggered when pad level is high
19	RESERVED	R	0h	Reserved
18	WUENSB	R/W	0h	Wakeup enable from standby 0h = Wakeup disabled 1h = Wakeup enabled (effective only if EDGEDET is enabled)
17-16	EDGEDET	R/W	0h	Edge detect configuration 0h = No edge detection 1h = Negative edge detection 2h = Positive edge detection 3h = Positive and negative edge detection
15	RESERVED	R	0h	Reserved
14-13	PULLCTL	R/W	0h	Pull control. Setting this to value 0x3 disables pull. 0h = No pull 1h = Pull down enabled 2h = Pull up enabled
12-3	RESERVED	R	0h	Reserved
2-0	PORTCFG	R/W	0h	Selects usage of DIO25 0h = Base function 1h = Digital peripheral function-1 2h = Digital peripheral function-2 3h = Digital peripheral function-3 4h = Digital peripheral function-4 5h = Digital peripheral function-5 6h = Analog function 7h = Digital Test Bus function

18.9.29 DTBCFG Register (Offset = C00h) [Reset = 00000000h]

DTBCFG is shown in Table 18-33.

Return to the Summary Table.

DTB configuration

Table 18-33. DTBCFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-24	RESERVED	R	0h	Reserved
23	DTB0DIV	R/W	0h	This bit is used to divide DTB[0] output by 8. 0h = No divide 1h = Divide DTB[0] output by 8
22-19	RESERVED	R	0h	Reserved
18-16	PADSEL	R/W	0h	Selects which 3 DTB lines out of total 16 are routed to DTB pins 15 to 13. 0h = DTB[15:13] selected 1h = DTB[14:12] selected 2h = DTB[11:9] selected 3h = DTB[8:6] selected 4h = DTB[5:3] selected 5h = DTB[2:0] selected
15-13	RESERVED	R	0h	Reserved
12-8	ULLSEL	R/W	0h	ULL DTB Mux selection
7-5	RESERVED	R	0h	Reserved
4-0	SVTSEL	R/W	0h	SVT DTB Mux selection

18.9.30 DTBOE Register (Offset = C04h) [Reset = 00000000h]

DTBOE is shown in Table 18-34.

Return to the Summary Table.

DTB output enable

Table 18-34. DTBOE Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15	EN15	R/W	0h	Enables DTB output 15 0h = DTB output disabled 1h = DTB output enabled
14	EN14	R/W	0h	Enables DTB output 14 0h = DTB output disabled 1h = DTB output enabled
13	EN13	R/W	0h	Enables DTB output 13 0h = DTB output disabled 1h = DTB output enabled
12	EN12	R/W	0h	Enables DTB output 12 0h = DTB output disabled 1h = DTB output enabled
11	EN11	R/W	0h	Enables DTB output 11 0h = DTB output disabled 1h = DTB output enabled
10	EN10	R/W	0h	Enables DTB output 10 0h = DTB output disabled 1h = DTB output enabled
9	EN9	R/W	0h	Enables DTB output 9 0h = DTB output disabled 1h = DTB output enabled
8	EN8	R/W	0h	Enables DTB output 8 0h = DTB output disabled 1h = DTB output enabled
7	EN7	R/W	0h	Enables DTB output 7 0h = DTB output disabled 1h = DTB output enabled
6	EN6	R/W	0h	Enables DTB output 6 0h = DTB output disabled 1h = DTB output enabled
5	EN5	R/W	0h	Enables DTB output 5 0h = DTB output disabled 1h = DTB output enabled
4	EN4	R/W	0h	Enables DTB output 4 0h = DTB output disabled 1h = DTB output enabled
3	EN3	R/W	0h	Enables DTB output 3 0h = DTB output disabled 1h = DTB output enabled
2	EN2	R/W	0h	Enables DTB output 2 0h = DTB output disabled 1h = DTB output enabled
1	EN1	R/W	0h	Enables DTB output 1 0h = DTB output disabled 1h = DTB output enabled
0	EN0	R/W	0h	Enables DTB output 0 0h = DTB output disabled 1h = DTB output enabled

18.9.31 EVTCFG Register (Offset = C08h) [Reset = 00000000h]

EVTCFG is shown in Table 18-35.

Return to the Summary Table.

Event configuration. This register is used to select DIO for IOC to publish event on ULL event fabric. It also contains enable bit that is used to mask the event and event flag bit.

Table 18-35. EVTCFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-9	RESERVED	R	0h	Reserved
8	EVTIFG	R/W	0h	Event flag. It is set when edge is detected on selected DIO. Note: The edge detector flop is cleared for the selected DIO when EVTIFG is cleared by software. 0h = Clear ULL event 1h = Set ULL event
7	EVTEN	R/W	0h	Enables IOC to publish event on AON event fabric when EVTIFG is set. 0h = Disable 1h = Enable
6	RESERVED	R	0h	Reserved
5-0	DIOSEL	R/W	0h	This is used to select DIO for event generation. For example, DIOSEL = 0x0 selects DIO0 and DIOSEL = 0x8 selects DIO8.

18.9.32 TEST Register (Offset = C0Ch) [Reset = 00000000h]

TEST is shown in Table 18-36.

Return to the Summary Table.

Test register.

Table 18-36. TEST Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Reserved
0	SEL	R/W	0h	This is used to drive SWDIO (Serial Wire DIO) output data and output enable from debug sub-system onto DIO12 (Test Data Output) pad. 0h = Output data and output enable managed by IOC 1h = Output data and output enable driven based on debug sub-system inputs

18.9.33 DTBSTAT Register (Offset = C10h) [Reset = 00000000h]

DTBSTAT is shown in Table 18-37.

Return to the Summary Table.

DTB status register.

Table 18-37. DTBSTAT Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	VAL	R	0h	This bit field captures the final 16-bit value of DTB signals provided from IOC to device pins.

18.10 GPIO Registers

Table 18-38 lists the memory-mapped registers for the GPIO registers. All register offset addresses not listed in Table 18-38 should be considered as reserved locations and the register contents should not be modified.

Table 18-38. GPIO Registers

Offset	Acronym	Register Name	Section
0h	DESC	Module Description	Go
4h	DESCEX	Module Description Extended	Go
44h	IMASK	Interrupt Mask	Go
4Ch	RIS	Raw interrupt status	Go
54h	MIS	Masked interrupt status	Go
5Ch	ISET	Interrupt set	Go
64h	ICLR	Interrupt clear	Go
6Ch	IMSET	Interrupt mask set	Go
74h	IMCLR	Interrupt mask clear	Go
100h	DOUT3_0	Alias for Data out 3 to 0	Go
104h	DOUT7_4	Alias for Data out 7 to 4	Go
108h	DOUT11_8	Alias for Data out 11 to 8	Go
10Ch	DOUT15_12	Alias for Data out 15 to 12	Go
110h	DOUT19_16	Alias for Data out 19 to 16	Go
114h	DOUT23_20	Alias for Data out 23 to 20	Go
118h	DOUT27_24	Alias for Data out 27 to 24	Go
200h	DOUT31_0	Data out 31 to 0	Go
210h	DOUTSET31_0	Data out set 31 to 0	Go
220h	DOUTCLR31_0	Data out clear 31 to 0	Go
230h	DOUTTGL31_0	Data out toggle 31 to 0	Go
300h	DOUTTGL3_0	Alias for Data out toggle 3 to 0	Go
304h	DOUTTGL7_4	Alias for Data out toggle 7 to 4	Go
308h	DOUTTGL11_8	Alias for Data out toggle 11 to 8	Go
30Ch	DOUTTGL15_12	Alias for Data out toggle 15 to 12	Go
310h	DOUTTGL19_16	Alias for Data out toggle 19 to 16	Go
314h	DOUTTGL23_20	Alias for Data out toggle 23 to 20	Go
318h	DOUTTGL27_24	Alias for Data out toggle 27 to 24	Go
400h	DOE3_0	Alias for Data out enable 3 to 0	Go
404h	DOE7_4	Alias for Data out enable 7 to 4	Go
408h	DOE11_8	Alias for Data out enable 11 to 8	Go
40Ch	DOE15_12	Alias for Data out enable 15 to 12	Go
410h	DOE19_16	Alias for Data out enable 19 to 16	Go
414h	DOE23_20	Alias for Data out enable 23 to 20	Go
418h	DOE27_24	Alias for Data out enable 27 to 24	Go
500h	DOE31_0	Data out enable 31 to 0	Go
510h	DOESET31_0	Data out enable set 31 to 0	Go
520h	DOECLR31_0	Data out enable clear 31 to 0	Go
530h	DOETGL31_0	Data out enable toggle 31 to 0	Go
600h	DIN3_0	Alias for Data input 3 to 0	Go
604h	DIN7_4	Alias for Data input 7 to 4	Go
608h	DIN11_8	Alias for Data input 11 to 8	Go

Table 18-38. GPIO Registers (continued)

Offset	Acronym	Register Name	Section
60Ch	DIN15_12	Alias for Data input 15 to 12	Go
610h	DIN19_16	Alias for Data input 19 to 16	Go
614h	DIN23_20	Alias for Data input 23 to 20	Go
618h	DIN27_24	Alias for Data input 27 to 24	Go
700h	DIN31_0	Data input 31 to 0	Go
800h	EVTCFG	Event configuration	Go

Complex bit access types are encoded to fit into small table cells. Table 18-39 shows the codes that are used for access types in this section.

Table 18-39. GPIO Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

18.10.1 DESC Register (Offset = 0h) [Reset = 7C491010h]

DESC is shown in Table 18-40.

Return to the Summary Table.

Description Register. This register provides IP module ID, revision information, instance index and standard MMR registers offset.

Table 18-40. DESC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	MODID	R	7C49h	Module identifier used to uniquely identify this IP.
15-12	STDIPOFF	R	1h	Standard IP MMR block offset. Standard IP MMRs are the set from aggregated IRQ registers till DTB. 0: Standard IP MMRs do not exist 0x1-0xF: Standard IP MMRs begin at offset of (64*STDIPOFF from the base IP address)
11-8	INSTIDX	R	0h	IP Instance ID number. If multiple instances of IP exist in the device, this field can identify the instance number (0-15).
7-4	MAJREV	R	1h	Major revision of IP (0-15).
3-0	MINREV	R	0h	Minor revision of IP (0-15).

18.10.2 DESCEX Register (Offset = 4h) [Reset = 00000019h]

DESCEX is shown in Table 18-41.

Return to the Summary Table.

Extended Description Register. This register provides configuration details of the IP to software drivers and end users.

Table 18-41. DESCEX Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Reserved
5-0	NUMDIO	R	19h	This provides the total number of DIOs supported by GPIO. The number of DIOs supprted is NUMDIO + 1

18.10.3 IMASK Register (Offset = 44h) [Reset = 00000000h]

IMASK is shown in Table 18-42.

Return to the Summary Table.

Interrupt mask for DIO pins

Table 18-42. IMASK Register Field Descriptions

Bit	Field	Type	Reset	Description
31-26	RESERVED	R	0h	Reserved
25	DIO25	R/W	0h	Interrupt mask for DIO25 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
24	DIO24	R/W	0h	Interrupt mask for DIO24 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
23	DIO23	R/W	0h	Interrupt mask for DIO23 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
22	DIO22	R/W	0h	Interrupt mask for DIO22 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
21	DIO21	R/W	0h	Interrupt mask for DIO21 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
20	DIO20	R/W	0h	Interrupt mask for DIO20 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
19	DIO19	R/W	0h	Interrupt mask for DIO19 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
18	DIO18	R/W	0h	Interrupt mask for DIO18 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
17	DIO17	R/W	0h	Interrupt mask for DIO17 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
16	DIO16	R/W	0h	Interrupt mask for DIO16 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
15	DIO15	R/W	0h	Interrupt mask for DIO15 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
14	DIO14	R/W	0h	Interrupt mask for DIO14 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
13	DIO13	R/W	0h	Interrupt mask for DIO13 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
12	DIO12	R/W	0h	Interrupt mask for DIO12 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
11	DIO11	R/W	0h	Interrupt mask for DIO11 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
10	DIO10	R/W	0h	Interrupt mask for DIO10 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask

Table 18-42. IMASK Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
9	DIO9	R/W	0h	Interrupt mask for DIO9 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
8	DIO8	R/W	0h	Interrupt mask for DIO8 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
7	DIO7	R/W	0h	Interrupt mask for DIO7 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
6	DIO6	R/W	0h	Interrupt mask for DIO6 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
5	DIO5	R/W	0h	Interrupt mask for DIO5 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
4	DIO4	R/W	0h	Interrupt mask for DIO4 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
3	DIO3	R/W	0h	Interrupt mask for DIO3 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
2	DIO2	R/W	0h	Interrupt mask for DIO2 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
1	DIO1	R/W	0h	Interrupt mask for DIO1 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
0	DIO0	R/W	0h	Interrupt mask for DIO0 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask

18.10.4 RIS Register (Offset = 4Ch) [Reset = 00000000h]

RIS is shown in Table 18-43.

Return to the Summary Table.

Raw interrupt flag for DIO pins

Table 18-43. RIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-26	RESERVED	R	0h	Reserved
25	DIO25	R	0h	Raw interrupt flag for DIO25 0h = Interrupt did not occur 1h = Interrupt occured
24	DIO24	R	0h	Raw interrupt flag for DIO24 0h = Interrupt did not occur 1h = Interrupt occured
23	DIO23	R	0h	Raw interrupt flag for DIO23 0h = Interrupt did not occur 1h = Interrupt occured
22	DIO22	R	0h	Raw interrupt flag for DIO22 0h = Interrupt did not occur 1h = Interrupt occured
21	DIO21	R	0h	Raw interrupt flag for DIO21 0h = Interrupt did not occur 1h = Interrupt occured
20	DIO20	R	0h	Raw interrupt flag for DIO20 0h = Interrupt did not occur 1h = Interrupt occured
19	DIO19	R	0h	Raw interrupt flag for DIO19 0h = Interrupt did not occur 1h = Interrupt occured
18	DIO18	R	0h	Raw interrupt flag for DIO18 0h = Interrupt did not occur 1h = Interrupt occured
17	DIO17	R	0h	Raw interrupt flag for DIO17 0h = Interrupt did not occur 1h = Interrupt occured
16	DIO16	R	0h	Raw interrupt flag for DIO16 0h = Interrupt did not occur 1h = Interrupt occured
15	DIO15	R	0h	Raw interrupt flag for DIO15 0h = Interrupt did not occur 1h = Interrupt occured
14	DIO14	R	0h	Raw interrupt flag for DIO14 0h = Interrupt did not occur 1h = Interrupt occured
13	DIO13	R	0h	Raw interrupt flag for DIO13 0h = Interrupt did not occur 1h = Interrupt occured
12	DIO12	R	0h	Raw interrupt flag for DIO12 0h = Interrupt did not occur 1h = Interrupt occured
11	DIO11	R	0h	Raw interrupt flag for DIO11 0h = Interrupt did not occur 1h = Interrupt occured
10	DIO10	R	0h	Raw interrupt flag for DIO10 0h = Interrupt did not occur 1h = Interrupt occured

Table 18-43. RIS Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
9	DIO9	R	0h	Raw interrupt flag for DIO9 0h = Interrupt did not occur 1h = Interrupt occured
8	DIO8	R	0h	Raw interrupt flag for DIO8 0h = Interrupt did not occur 1h = Interrupt occured
7	DIO7	R	0h	Raw interrupt flag for DIO7 0h = Interrupt did not occur 1h = Interrupt occured
6	DIO6	R	0h	Raw interrupt flag for DIO6 0h = Interrupt did not occur 1h = Interrupt occured
5	DIO5	R	0h	Raw interrupt flag for DIO5 0h = Interrupt did not occur 1h = Interrupt occured
4	DIO4	R	0h	Raw interrupt flag for DIO4 0h = Interrupt did not occur 1h = Interrupt occured
3	DIO3	R	0h	Raw interrupt flag for DIO3 0h = Interrupt did not occur 1h = Interrupt occured
2	DIO2	R	0h	Raw interrupt flag for DIO2 0h = Interrupt did not occur 1h = Interrupt occured
1	DIO1	R	0h	Raw interrupt flag for DIO1 0h = Interrupt did not occur 1h = Interrupt occured
0	DIO0	R	0h	Raw interrupt flag for DIO0 0h = Interrupt did not occur 1h = Interrupt occured

18.10.5 MIS Register (Offset = 54h) [Reset = 00000000h]

MIS is shown in Table 18-44.

Return to the Summary Table.

Masked interrupt flag for DIO pins

Table 18-44. MIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-26	RESERVED	R	0h	Reserved
25	DIO25	R	0h	Masked interrupt flag for DIO25 0h = Interrupt did not occur 1h = Interrupt occured
24	DIO24	R	0h	Masked interrupt flag for DIO24 0h = Interrupt did not occur 1h = Interrupt occured
23	DIO23	R	0h	Masked interrupt flag for DIO23 0h = Interrupt did not occur 1h = Interrupt occured
22	DIO22	R	0h	Masked interrupt flag for DIO22 0h = Interrupt did not occur 1h = Interrupt occured
21	DIO21	R	0h	Masked interrupt flag for DIO21 0h = Interrupt did not occur 1h = Interrupt occured
20	DIO20	R	0h	Masked interrupt flag for DIO20 0h = Interrupt did not occur 1h = Interrupt occured
19	DIO19	R	0h	Masked interrupt flag for DIO19 0h = Interrupt did not occur 1h = Interrupt occured
18	DIO18	R	0h	Masked interrupt flag for DIO18 0h = Interrupt did not occur 1h = Interrupt occured
17	DIO17	R	0h	Masked interrupt flag for DIO17 0h = Interrupt did not occur 1h = Interrupt occured
16	DIO16	R	0h	Masked interrupt flag for DIO16 0h = Interrupt did not occur 1h = Interrupt occured
15	DIO15	R	0h	Masked interrupt flag for DIO15 0h = Interrupt did not occur 1h = Interrupt occured
14	DIO14	R	0h	Masked interrupt flag for DIO14 0h = Interrupt did not occur 1h = Interrupt occured
13	DIO13	R	0h	Masked interrupt flag for DIO13 0h = Interrupt did not occur 1h = Interrupt occured
12	DIO12	R	0h	Masked interrupt flag for DIO12 0h = Interrupt did not occur 1h = Interrupt occured
11	DIO11	R	0h	Masked interrupt flag for DIO11 0h = Interrupt did not occur 1h = Interrupt occured
10	DIO10	R	0h	Masked interrupt flag for DIO10 0h = Interrupt did not occur 1h = Interrupt occured

Table 18-44. MIS Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
9	DIO9	R	0h	Masked interrupt flag for DIO9 0h = Interrupt did not occur 1h = Interrupt occured
8	DIO8	R	0h	Masked interrupt flag for DIO8 0h = Interrupt did not occur 1h = Interrupt occured
7	DIO7	R	0h	Masked interrupt flag for DIO7 0h = Interrupt did not occur 1h = Interrupt occured
6	DIO6	R	0h	Masked interrupt flag for DIO6 0h = Interrupt did not occur 1h = Interrupt occured
5	DIO5	R	0h	Masked interrupt flag for DIO5 0h = Interrupt did not occur 1h = Interrupt occured
4	DIO4	R	0h	Masked interrupt flag for DIO4 0h = Interrupt did not occur 1h = Interrupt occured
3	DIO3	R	0h	Masked interrupt flag for DIO3 0h = Interrupt did not occur 1h = Interrupt occured
2	DIO2	R	0h	Masked interrupt flag for DIO2 0h = Interrupt did not occur 1h = Interrupt occured
1	DIO1	R	0h	Masked interrupt flag for DIO1 0h = Interrupt did not occur 1h = Interrupt occured
0	DIO0	R	0h	Masked interrupt flag for DIO0 0h = Interrupt did not occur 1h = Interrupt occured

18.10.6 ISET Register (Offset = 5Ch) [Reset = 00000000h]

ISET is shown in Table 18-45.

Return to the Summary Table.

Set interrupt flag in RIS by writing a one

Table 18-45. ISET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-26	RESERVED	R	0h	Reserved
25	DIO25	W	0h	Set DIO25 in RIS 0h = Writing 0 has no effect 1h = Set Interrupt
24	DIO24	W	0h	Set DIO24 in RIS 0h = Writing 0 has no effect 1h = Set Interrupt
23	DIO23	W	0h	Set DIO23 in RIS 0h = Writing 0 has no effect 1h = Set Interrupt
22	DIO22	W	0h	Set DIO22 in RIS 0h = Writing 0 has no effect 1h = Set Interrupt
21	DIO21	W	0h	Set DIO21 in RIS 0h = Writing 0 has no effect 1h = Set Interrupt
20	DIO20	W	0h	Set DIO20 in RIS 0h = Writing 0 has no effect 1h = Set Interrupt
19	DIO19	W	0h	Set DIO19 in RIS 0h = Writing 0 has no effect 1h = Set Interrupt
18	DIO18	W	0h	Set DIO18 in RIS 0h = Writing 0 has no effect 1h = Set Interrupt
17	DIO17	W	0h	Set DIO17 in RIS 0h = Writing 0 has no effect 1h = Set Interrupt
16	DIO16	W	0h	Set DIO16 in RIS 0h = Writing 0 has no effect 1h = Set Interrupt
15	DIO15	W	0h	Set DIO15 in RIS 0h = Writing 0 has no effect 1h = Set Interrupt
14	DIO14	W	0h	Set DIO14 in RIS 0h = Writing 0 has no effect 1h = Set Interrupt
13	DIO13	W	0h	Set DIO13 in RIS 0h = Writing 0 has no effect 1h = Set Interrupt
12	DIO12	W	0h	Set DIO12 in RIS 0h = Writing 0 has no effect 1h = Set Interrupt
11	DIO11	W	0h	Set DIO11 in RIS 0h = Writing 0 has no effect 1h = Set Interrupt
10	DIO10	W	0h	Set DIO10 in RIS 0h = Writing 0 has no effect 1h = Set Interrupt

Table 18-45. ISET Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
9	DIO9	W	0h	Set DIO9 in RIS 0h = Writing 0 has no effect 1h = Set Interrupt
8	DIO8	W	0h	Set DIO8 in RIS 0h = Writing 0 has no effect 1h = Set Interrupt
7	DIO7	W	0h	Set DIO7 in RIS 0h = Writing 0 has no effect 1h = Set Interrupt
6	DIO6	W	0h	Set DIO6 in RIS 0h = Writing 0 has no effect 1h = Set Interrupt
5	DIO5	W	0h	Set DIO5 in RIS 0h = Writing 0 has no effect 1h = Set Interrupt
4	DIO4	W	0h	Set DIO4 in RIS 0h = Writing 0 has no effect 1h = Set Interrupt
3	DIO3	W	0h	Set DIO3 in RIS 0h = Writing 0 has no effect 1h = Set Interrupt
2	DIO2	W	0h	Set DIO2 in RIS 0h = Writing 0 has no effect 1h = Set Interrupt
1	DIO1	W	0h	Set DIO1 in RIS 0h = Writing 0 has no effect 1h = Set Interrupt
0	DIO0	W	0h	Set DIO0 in RIS 0h = Writing 0 has no effect 1h = Set Interrupt

18.10.7 ICLR Register (Offset = 64h) [Reset = 00000000h]

ICLR is shown in Table 18-46.

Return to the Summary Table.

Clear interrupt flag in RIS by writing a one

Table 18-46. ICLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-26	RESERVED	R	0h	Reserved
25	DIO25	W	0h	Clears DIO25 in RIS 0h = Writing 0 has no effect 1h = Clear Interrupt
24	DIO24	W	0h	Clears DIO24 in RIS 0h = Writing 0 has no effect 1h = Clear Interrupt
23	DIO23	W	0h	Clears DIO23 in RIS 0h = Writing 0 has no effect 1h = Clear Interrupt
22	DIO22	W	0h	Clears DIO22 in RIS 0h = Writing 0 has no effect 1h = Clear Interrupt
21	DIO21	W	0h	Clears DIO21 in RIS 0h = Writing 0 has no effect 1h = Clear Interrupt
20	DIO20	W	0h	Clears DIO20 in RIS 0h = Writing 0 has no effect 1h = Clear Interrupt
19	DIO19	W	0h	Clears DIO19 in RIS 0h = Writing 0 has no effect 1h = Clear Interrupt
18	DIO18	W	0h	Clears DIO18 in RIS 0h = Writing 0 has no effect 1h = Clear Interrupt
17	DIO17	W	0h	Clears DIO17 in RIS 0h = Writing 0 has no effect 1h = Clear Interrupt
16	DIO16	W	0h	Clears DIO16 in RIS 0h = Writing 0 has no effect 1h = Clear Interrupt
15	DIO15	W	0h	Clears DIO15 in RIS 0h = Writing 0 has no effect 1h = Clear Interrupt
14	DIO14	W	0h	Clears DIO14 in RIS 0h = Writing 0 has no effect 1h = Clear Interrupt
13	DIO13	W	0h	Clears DIO13 in RIS 0h = Writing 0 has no effect 1h = Clear Interrupt
12	DIO12	W	0h	Clears DIO12 in RIS 0h = Writing 0 has no effect 1h = Clear Interrupt
11	DIO11	W	0h	Clears DIO11 in RIS 0h = Writing 0 has no effect 1h = Clear Interrupt
10	DIO10	W	0h	Clears DIO10 in RIS 0h = Writing 0 has no effect 1h = Clear Interrupt

Table 18-46. ICLR Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
9	DIO9	W	0h	Clears DIO9 in RIS 0h = Writing 0 has no effect 1h = Clear Interrupt
8	DIO8	W	0h	Clears DIO8 in RIS 0h = Writing 0 has no effect 1h = Clear Interrupt
7	DIO7	W	0h	Clears DIO7 in RIS 0h = Writing 0 has no effect 1h = Clear Interrupt
6	DIO6	W	0h	Clears DIO6 in RIS 0h = Writing 0 has no effect 1h = Clear Interrupt
5	DIO5	W	0h	Clears DIO5 in RIS 0h = Writing 0 has no effect 1h = Clear Interrupt
4	DIO4	W	0h	Clears DIO4 in RIS 0h = Writing 0 has no effect 1h = Clear Interrupt
3	DIO3	W	0h	Clears DIO3 in RIS 0h = Writing 0 has no effect 1h = Clear Interrupt
2	DIO2	W	0h	Clears DIO2 in RIS 0h = Writing 0 has no effect 1h = Clear Interrupt
1	DIO1	W	0h	Clears DIO1 in RIS 0h = Writing 0 has no effect 1h = Clear Interrupt
0	DIO0	W	0h	Clears DIO0 in RIS 0h = Writing 0 has no effect 1h = Clear Interrupt

18.10.8 IMSET Register (Offset = 6Ch) [Reset = 00000000h]

IMSET is shown in Table 18-47.

Return to the Summary Table.

Set interrupt mask in IMASK by writing a one

Table 18-47. IMSET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-26	RESERVED	R	0h	Reserved
25	DIO25	W	0h	Sets DIO25 in IMASK 0h = Writing 0 has no effect 1h = Set interrupt mask
24	DIO24	W	0h	Sets DIO24 in IMASK 0h = Writing 0 has no effect 1h = Set interrupt mask
23	DIO23	W	0h	Sets DIO23 in IMASK 0h = Writing 0 has no effect 1h = Set interrupt mask
22	DIO22	W	0h	Sets DIO22 in IMASK 0h = Writing 0 has no effect 1h = Set interrupt mask
21	DIO21	W	0h	Sets DIO21 in IMASK 0h = Writing 0 has no effect 1h = Set interrupt mask
20	DIO20	W	0h	Sets DIO20 in IMASK 0h = Writing 0 has no effect 1h = Set interrupt mask
19	DIO19	W	0h	Sets DIO19 in IMASK 0h = Writing 0 has no effect 1h = Set interrupt mask
18	DIO18	W	0h	Sets DIO18 in IMASK 0h = Writing 0 has no effect 1h = Set interrupt mask
17	DIO17	W	0h	Sets DIO17 in IMASK 0h = Writing 0 has no effect 1h = Set interrupt mask
16	DIO16	W	0h	Sets DIO16 in IMASK 0h = Writing 0 has no effect 1h = Set interrupt mask
15	DIO15	W	0h	Sets DIO15 in IMASK 0h = Writing 0 has no effect 1h = Set interrupt mask
14	DIO14	W	0h	Sets DIO14 in IMASK 0h = Writing 0 has no effect 1h = Set interrupt mask
13	DIO13	W	0h	Sets DIO13 in IMASK 0h = Writing 0 has no effect 1h = Set interrupt mask
12	DIO12	W	0h	Sets DIO12 in IMASK 0h = Writing 0 has no effect 1h = Set interrupt mask
11	DIO11	W	0h	Sets DIO11 in IMASK 0h = Writing 0 has no effect 1h = Set interrupt mask
10	DIO10	W	0h	Sets DIO10 in IMASK 0h = Writing 0 has no effect 1h = Set interrupt mask

Table 18-47. IMSET Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
9	DIO9	W	0h	Sets DIO9 in IMASK 0h = Writing 0 has no effect 1h = Set interrupt mask
8	DIO8	W	0h	Sets DIO8 in IMASK 0h = Writing 0 has no effect 1h = Set interrupt mask
7	DIO7	W	0h	Sets DIO7 in IMASK 0h = Writing 0 has no effect 1h = Set interrupt mask
6	DIO6	W	0h	Sets DIO6 in IMASK 0h = Writing 0 has no effect 1h = Set interrupt mask
5	DIO5	W	0h	Sets DIO5 in IMASK 0h = Writing 0 has no effect 1h = Set interrupt mask
4	DIO4	W	0h	Sets DIO4 in IMASK 0h = Writing 0 has no effect 1h = Set interrupt mask
3	DIO3	W	0h	Sets DIO3 in IMASK 0h = Writing 0 has no effect 1h = Set interrupt mask
2	DIO2	W	0h	Sets DIO2 in IMASK 0h = Writing 0 has no effect 1h = Set interrupt mask
1	DIO1	W	0h	Sets DIO1 in IMASK 0h = Writing 0 has no effect 1h = Set interrupt mask
0	DIO0	W	0h	Sets DIO0 in IMASK 0h = Writing 0 has no effect 1h = Set interrupt mask

18.10.9 IMCLR Register (Offset = 74h) [Reset = 00000000h]

IMCLR is shown in Table 18-48.

Return to the Summary Table.

Clear interrupt mask in IMASK by writing a one

Table 18-48. IMCLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-26	RESERVED	R	0h	Reserved
25	DIO25	W	0h	Clears DIO25 in IMASK 0h = Writing 0 has no effect 1h = Clear interrupt mask
24	DIO24	W	0h	Clears DIO24 in IMASK 0h = Writing 0 has no effect 1h = Clear interrupt mask
23	DIO23	W	0h	Clears DIO23 in IMASK 0h = Writing 0 has no effect 1h = Clear interrupt mask
22	DIO22	W	0h	Clears DIO22 in IMASK 0h = Writing 0 has no effect 1h = Clear interrupt mask
21	DIO21	W	0h	Clears DIO21 in IMASK 0h = Writing 0 has no effect 1h = Clear interrupt mask
20	DIO20	W	0h	Clears DIO20 in IMASK 0h = Writing 0 has no effect 1h = Clear interrupt mask
19	DIO19	W	0h	Clears DIO19 in IMASK 0h = Writing 0 has no effect 1h = Clear interrupt mask
18	DIO18	W	0h	Clears DIO18 in IMASK 0h = Writing 0 has no effect 1h = Clear interrupt mask
17	DIO17	W	0h	Clears DIO17 in IMASK 0h = Writing 0 has no effect 1h = Clear interrupt mask
16	DIO16	W	0h	Clears DIO16 in IMASK 0h = Writing 0 has no effect 1h = Clear interrupt mask
15	DIO15	W	0h	Clears DIO15 in IMASK 0h = Writing 0 has no effect 1h = Clear interrupt mask
14	DIO14	W	0h	Clears DIO14 in IMASK 0h = Writing 0 has no effect 1h = Clear interrupt mask
13	DIO13	W	0h	Clears DIO13 in IMASK 0h = Writing 0 has no effect 1h = Clear interrupt mask
12	DIO12	W	0h	Clears DIO12 in IMASK 0h = Writing 0 has no effect 1h = Clear interrupt mask
11	DIO11	W	0h	Clears DIO11 in IMASK 0h = Writing 0 has no effect 1h = Clear interrupt mask
10	DIO10	W	0h	Clears DIO10 in IMASK 0h = Writing 0 has no effect 1h = Clear interrupt mask

Table 18-48. IMCLR Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
9	DIO9	W	0h	Clears DIO9 in IMASK 0h = Writing 0 has no effect 1h = Clear interrupt mask
8	DIO8	W	0h	Clears DIO8 in IMASK 0h = Writing 0 has no effect 1h = Clear interrupt mask
7	DIO7	W	0h	Clears DIO7 in IMASK 0h = Writing 0 has no effect 1h = Clear interrupt mask
6	DIO6	W	0h	Clears DIO6 in IMASK 0h = Writing 0 has no effect 1h = Clear interrupt mask
5	DIO5	W	0h	Clears DIO5 in IMASK 0h = Writing 0 has no effect 1h = Clear interrupt mask
4	DIO4	W	0h	Clears DIO4 in IMASK 0h = Writing 0 has no effect 1h = Clear interrupt mask
3	DIO3	W	0h	Clears DIO3 in IMASK 0h = Writing 0 has no effect 1h = Clear interrupt mask
2	DIO2	W	0h	Clears DIO2 in IMASK 0h = Writing 0 has no effect 1h = Clear interrupt mask
1	DIO1	W	0h	Clears DIO1 in IMASK 0h = Writing 0 has no effect 1h = Clear interrupt mask
0	DIO0	W	0h	Clears DIO0 in IMASK 0h = Writing 0 has no effect 1h = Clear interrupt mask

18.10.10 DOUT3_0 Register (Offset = 100h) [Reset = 00000000h]

DOUT3_0 is shown in Table 18-49.

Return to the Summary Table.

Alias register for byte access to DOUT31_0[3:0] bits.

Table 18-49. DOUT3_0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-25	RESERVED	R	0h	Reserved
24	DIO3	R/W	0h	Data output for DIO3 0h = Output is set to 0 1h = Output is set to 1
23-17	RESERVED	R	0h	Reserved
16	DIO2	R/W	0h	Data output for DIO2 0h = Output is set to 0 1h = Output is set to 1
15-9	RESERVED	R	0h	Reserved
8	DIO1	R/W	0h	Data output for DIO1 0h = Output is set to 0 1h = Output is set to 1
7-1	RESERVED	R	0h	Reserved
0	DIO0	R/W	0h	Data output for DIO0 0h = Output is set to 0 1h = Output is set to 1

18.10.11 DOUT7_4 Register (Offset = 104h) [Reset = 00000000h]

DOUT7_4 is shown in Table 18-50.

Return to the Summary Table.

Alias register for byte access to DOUT31_0[7:4] bits

Table 18-50. DOUT7_4 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-25	RESERVED	R	0h	Reserved
24	DIO7	R/W	0h	Data output for DIO7 0h = Output is set to 0 1h = Output is set to 1
23-17	RESERVED	R	0h	Reserved
16	DIO6	R/W	0h	Data output for DIO6 0h = Output is set to 0 1h = Output is set to 1
15-9	RESERVED	R	0h	Reserved
8	DIO5	R/W	0h	Data output for DIO5 0h = Output is set to 0 1h = Output is set to 1
7-1	RESERVED	R	0h	Reserved
0	DIO4	R/W	0h	Data output for DIO4 0h = Output is set to 0 1h = Output is set to 1

18.10.12 DOUT11_8 Register (Offset = 108h) [Reset = 00000000h]

DOUT11_8 is shown in Table 18-51.

Return to the Summary Table.

Alias register for byte access to DOUT31_0[11:8] bits

Table 18-51. DOUT11_8 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-25	RESERVED	R	0h	Reserved
24	DIO11	R/W	0h	Data output for DIO11 0h = Output is set to 0 1h = Output is set to 1
23-17	RESERVED	R	0h	Reserved
16	DIO10	R/W	0h	Data output for DIO10 0h = Output is set to 0 1h = Output is set to 1
15-9	RESERVED	R	0h	Reserved
8	DIO9	R/W	0h	Data output for DIO9 0h = Output is set to 0 1h = Output is set to 1
7-1	RESERVED	R	0h	Reserved
0	DIO8	R/W	0h	Data output for DIO8 0h = Output is set to 0 1h = Output is set to 1

18.10.13 DOUT15_12 Register (Offset = 10Ch) [Reset = 00000000h]

DOUT15_12 is shown in Table 18-52.

Return to the Summary Table.

Alias register for byte access to DOUT31_0[15:12] bits

Table 18-52. DOUT15_12 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-25	RESERVED	R	0h	Reserved
24	DIO15	R/W	0h	Data output for DIO15 0h = Output is set to 0 1h = Output is set to 1
23-17	RESERVED	R	0h	Reserved
16	DIO14	R/W	0h	Data output for DIO14 0h = Output is set to 0 1h = Output is set to 1
15-9	RESERVED	R	0h	Reserved
8	DIO13	R/W	0h	Data output for DIO13 0h = Output is set to 0 1h = Output is set to 1
7-1	RESERVED	R	0h	Reserved
0	DIO12	R/W	0h	Data output for DIO12 0h = Output is set to 0 1h = Output is set to 1

18.10.14 DOUT19_16 Register (Offset = 110h) [Reset = 00000000h]

DOUT19_16 is shown in Table 18-53.

Return to the Summary Table.

Alias register for byte access to DOUT31_0[19:16] bits

Table 18-53. DOUT19_16 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-25	RESERVED	R	0h	Reserved
24	DIO19	R/W	0h	Data output for DIO19 0h = Output is set to 0 1h = Output is set to 1
23-17	RESERVED	R	0h	Reserved
16	DIO18	R/W	0h	Data output for DIO18 0h = Output is set to 0 1h = Output is set to 1
15-9	RESERVED	R	0h	Reserved
8	DIO17	R/W	0h	Data output for DIO17 0h = Output is set to 0 1h = Output is set to 1
7-1	RESERVED	R	0h	Reserved
0	DIO16	R/W	0h	Data output for DIO16 0h = Output is set to 0 1h = Output is set to 1

18.10.15 DOUT23_20 Register (Offset = 114h) [Reset = 00000000h]

DOUT23_20 is shown in Table 18-54.

Return to the Summary Table.

Alias register for byte access to DOUT31_0[23:20] bits

Table 18-54. DOUT23_20 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-25	RESERVED	R	0h	Reserved
24	DIO23	R/W	0h	Data output for DIO23 0h = Output is set to 0 1h = Output is set to 1
23-17	RESERVED	R	0h	Reserved
16	DIO22	R/W	0h	Data output for DIO22 0h = Output is set to 0 1h = Output is set to 1
15-9	RESERVED	R	0h	Reserved
8	DIO21	R/W	0h	Data output for DIO21 0h = Output is set to 0 1h = Output is set to 1
7-1	RESERVED	R	0h	Reserved
0	DIO20	R/W	0h	Data output for DIO20 0h = Output is set to 0 1h = Output is set to 1

18.10.16 DOUT27_24 Register (Offset = 118h) [Reset = 00000000h]

DOUT27_24 is shown in Table 18-55.

Return to the Summary Table.

Alias register for byte access to DOUT31_0[27:24] bits

Table 18-55. DOUT27_24 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-9	RESERVED	R	0h	Reserved
8	DIO25	R/W	0h	Data output for DIO25 0h = Output is set to 0 1h = Output is set to 1
7-1	RESERVED	R	0h	Reserved
0	DIO24	R/W	0h	Data output for DIO24 0h = Output is set to 0 1h = Output is set to 1

18.10.17 DOUT31_0 Register (Offset = 200h) [Reset = 00000000h]

DOUT31_0 is shown in Table 18-56.

Return to the Summary Table.

Data Output for DIO 31 to 0 pins.

Table 18-56. DOUT31_0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-26	RESERVED	R	0h	Reserved
25	DIO25	R/W	0h	Data output for DIO25 0h = Output is set to 0 1h = Output is set to 1
24	DIO24	R/W	0h	Data output for DIO24 0h = Output is set to 0 1h = Output is set to 1
23	DIO23	R/W	0h	Data output for DIO23 0h = Output is set to 0 1h = Output is set to 1
22	DIO22	R/W	0h	Data output for DIO22 0h = Output is set to 0 1h = Output is set to 1
21	DIO21	R/W	0h	Data output for DIO21 0h = Output is set to 0 1h = Output is set to 1
20	DIO20	R/W	0h	Data output for DIO20 0h = Output is set to 0 1h = Output is set to 1
19	DIO19	R/W	0h	Data output for DIO19 0h = Output is set to 0 1h = Output is set to 1
18	DIO18	R/W	0h	Data output for DIO18 0h = Output is set to 0 1h = Output is set to 1
17	DIO17	R/W	0h	Data output for DIO17 0h = Output is set to 0 1h = Output is set to 1
16	DIO16	R/W	0h	Data output for DIO16 0h = Output is set to 0 1h = Output is set to 1
15	DIO15	R/W	0h	Data output for DIO15 0h = Output is set to 0 1h = Output is set to 1
14	DIO14	R/W	0h	Data output for DIO14 0h = Output is set to 0 1h = Output is set to 1
13	DIO13	R/W	0h	Data output for DIO13 0h = Output is set to 0 1h = Output is set to 1
12	DIO12	R/W	0h	Data output for DIO12 0h = Output is set to 0 1h = Output is set to 1
11	DIO11	R/W	0h	Data output for DIO11 0h = Output is set to 0 1h = Output is set to 1
10	DIO10	R/W	0h	Data output for DIO10 0h = Output is set to 0 1h = Output is set to 1

Table 18-56. DOUT31_0 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
9	DIO9	R/W	0h	Data output for DIO9 0h = Output is set to 0 1h = Output is set to 1
8	DIO8	R/W	0h	Data output for DIO8 0h = Output is set to 0 1h = Output is set to 1
7	DIO7	R/W	0h	Data output for DIO7 0h = Output is set to 0 1h = Output is set to 1
6	DIO6	R/W	0h	Data output for DIO6 0h = Output is set to 0 1h = Output is set to 1
5	DIO5	R/W	0h	Data output for DIO5 0h = Output is set to 0 1h = Output is set to 1
4	DIO4	R/W	0h	Data output for DIO4 0h = Output is set to 0 1h = Output is set to 1
3	DIO3	R/W	0h	Data output for DIO3 0h = Output is set to 0 1h = Output is set to 1
2	DIO2	R/W	0h	Data output for DIO2 0h = Output is set to 0 1h = Output is set to 1
1	DIO1	R/W	0h	Data output for DIO1 0h = Output is set to 0 1h = Output is set to 1
0	DIO0	R/W	0h	Data output for DIO0 0h = Output is set to 0 1h = Output is set to 1

18.10.18 DOUTSET31_0 Register (Offset = 210h) [Reset = 00000000h]

DOUTSET31_0 is shown in Table 18-57.

Return to the Summary Table.

Alias regiser to set the corresponding bits of DOUT31_0 register.

Table 18-57. DOUTSET31_0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-26	RESERVED	R	0h	Reserved
25	DIO25	W	0h	Set bit DOUT31_0.DIO25 0h = No effect 1h = Set
24	DIO24	W	0h	Set bit DOUT31_0.DIO24 0h = No effect 1h = Set
23	DIO23	W	0h	Set bit DOUT31_0.DIO23 0h = No effect 1h = Set
22	DIO22	W	0h	Set bit DOUT31_0.DIO22 0h = No effect 1h = Set
21	DIO21	W	0h	Set bit DOUT31_0.DIO21 0h = No effect 1h = Set
20	DIO20	W	0h	Set bit DOUT31_0.DIO20 0h = No effect 1h = Set
19	DIO19	W	0h	Set bit DOUT31_0.DIO19 0h = No effect 1h = Set
18	DIO18	W	0h	Set bit DOUT31_0.DIO18 0h = No effect 1h = Set
17	DIO17	W	0h	Set bit DOUT31_0.DIO17 0h = No effect 1h = Set
16	DIO16	W	0h	Set bit DOUT31_0.DIO16 0h = No effect 1h = Set
15	DIO15	W	0h	Set bit DOUT31_0.DIO15 0h = No effect 1h = Set
14	DIO14	W	0h	Set bit DOUT31_0.DIO14 0h = No effect 1h = Set
13	DIO13	W	0h	Set bit DOUT31_0.DIO13 0h = No effect 1h = Set
12	DIO12	W	0h	Set bit DOUT31_0.DIO12 0h = No effect 1h = Set
11	DIO11	W	0h	Set bit DOUT31_0.DIO11 0h = No effect 1h = Set
10	DIO10	W	0h	Set bit DOUT31_0.DIO10 0h = No effect 1h = Set

Table 18-57. DOUTSET31_0 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
9	DIO9	W	0h	Set bit DOUT31_0.DIO9 0h = No effect 1h = Set
8	DIO8	W	0h	Set bit DOUT31_0.DIO8 0h = No effect 1h = Set
7	DIO7	W	0h	Set bit DOUT31_0.DIO7 0h = No effect 1h = Set
6	DIO6	W	0h	Set bit DOUT31_0.DIO6 0h = No effect 1h = Set
5	DIO5	W	0h	Set bit DOUT31_0.DIO5 0h = No effect 1h = Set
4	DIO4	W	0h	Set bit DOUT31_0.DIO4 0h = No effect 1h = Set
3	DIO3	W	0h	Set bit DOUT31_0.DIO3 0h = No effect 1h = Set
2	DIO2	W	0h	Set bit DOUT31_0.DIO2 0h = No effect 1h = Set
1	DIO1	W	0h	Set bit DOUT31_0.DIO1 0h = No effect 1h = Set
0	DIO0	W	0h	Set bit DOUT31_0.DIO0 0h = No effect 1h = Set

18.10.19 DOUTCLR31_0 Register (Offset = 220h) [Reset = 00000000h]

DOUTCLR31_0 is shown in Table 18-58.

Return to the Summary Table.

Alias regiser to clear the corresponding bits of DOUT31_0 register.

Table 18-58. DOUTCLR31_0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-26	RESERVED	R	0h	Reserved
25	DIO25	W	0h	Clear bit DOUT31_0.DIO25 0h = No effect 1h = Clear
24	DIO24	W	0h	Clear bit DOUT31_0.DIO24 0h = No effect 1h = Clear
23	DIO23	W	0h	Clear bit DOUT31_0.DIO23 0h = No effect 1h = Clear
22	DIO22	W	0h	Clear bit DOUT31_0.DIO22 0h = No effect 1h = Clear
21	DIO21	W	0h	Clear bit DOUT31_0.DIO21 0h = No effect 1h = Clear
20	DIO20	W	0h	Clear bit DOUT31_0.DIO20 0h = No effect 1h = Clear
19	DIO19	W	0h	Clear bit DOUT31_0.DIO19 0h = No effect 1h = Clear
18	DIO18	W	0h	Clear bit DOUT31_0.DIO18 0h = No effect 1h = Clear
17	DIO17	W	0h	Clear bit DOUT31_0.DIO17 0h = No effect 1h = Clear
16	DIO16	W	0h	Clear bit DOUT31_0.DIO16 0h = No effect 1h = Clear
15	DIO15	W	0h	Clear bit DOUT31_0.DIO15 0h = No effect 1h = Clear
14	DIO14	W	0h	Clear bit DOUT31_0.DIO14 0h = No effect 1h = Clear
13	DIO13	W	0h	Clear bit DOUT31_0.DIO13 0h = No effect 1h = Clear
12	DIO12	W	0h	Clear bit DOUT31_0.DIO12 0h = No effect 1h = Clear
11	DIO11	W	0h	Clear bit DOUT31_0.DIO11 0h = No effect 1h = Clear
10	DIO10	W	0h	Clear bit DOUT31_0.DIO10 0h = No effect 1h = Clear

Table 18-58. DOUTCLR31_0 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
9	DIO9	W	0h	Clear bit DOUT31_0.DIO9 0h = No effect 1h = Clear
8	DIO8	W	0h	Clear bit DOUT31_0.DIO8 0h = No effect 1h = Clear
7	DIO7	W	0h	Clear bit DOUT31_0.DIO7 0h = No effect 1h = Clear
6	DIO6	W	0h	Clear bit DOUT31_0.DIO6 0h = No effect 1h = Clear
5	DIO5	W	0h	Clear bit DOUT31_0.DIO5 0h = No effect 1h = Clear
4	DIO4	W	0h	Clear bit DOUT31_0.DIO4 0h = No effect 1h = Clear
3	DIO3	W	0h	Clear bit DOUT31_0.DIO3 0h = No effect 1h = Clear
2	DIO2	W	0h	Clear bit DOUT31_0.DIO2 0h = No effect 1h = Clear
1	DIO1	W	0h	Clear bit DOUT31_0.DIO1 0h = No effect 1h = Clear
0	DIO0	W	0h	Clear bit DOUT31_0.DIO0 0h = No effect 1h = Clear

18.10.20 DOUTTGL31_0 Register (Offset = 230h) [Reset = 00000000h]

DOUTTGL31_0 is shown in Table 18-59.

Return to the Summary Table.

Alias regiser to toggle the corresponding bits of DOUT31_0 register.

Table 18-59. DOUTTGL31_0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-26	RESERVED	R	0h	Reserved
25	DIO25	W	0h	Toggles bit DOUT31_0.DIO25 0h = No effect 1h = Toggle
24	DIO24	W	0h	Toggles bit DOUT31_0.DIO24 0h = No effect 1h = Toggle
23	DIO23	W	0h	Toggles bit DOUT31_0.DIO23 0h = No effect 1h = Toggle
22	DIO22	W	0h	Toggles bit DOUT31_0.DIO22 0h = No effect 1h = Toggle
21	DIO21	W	0h	Toggles bit DOUT31_0.DIO21 0h = No effect 1h = Toggle
20	DIO20	W	0h	Toggles bit DOUT31_0.DIO20 0h = No effect 1h = Toggle
19	DIO19	W	0h	Toggles bit DOUT31_0.DIO19 0h = No effect 1h = Toggle
18	DIO18	W	0h	Toggles bit DOUT31_0.DIO18 0h = No effect 1h = Toggle
17	DIO17	W	0h	Toggles bit DOUT31_0.DIO17 0h = No effect 1h = Toggle
16	DIO16	W	0h	Toggles bit DOUT31_0.DIO16 0h = No effect 1h = Toggle
15	DIO15	W	0h	Toggles bit DOUT31_0.DIO15 0h = No effect 1h = Toggle
14	DIO14	W	0h	Toggles bit DOUT31_0.DIO14 0h = No effect 1h = Toggle
13	DIO13	W	0h	Toggles bit DOUT31_0.DIO13 0h = No effect 1h = Toggle
12	DIO12	W	0h	Toggles bit DOUT31_0.DIO12 0h = No effect 1h = Toggle
11	DIO11	W	0h	Toggles bit DOUT31_0.DIO11 0h = No effect 1h = Toggle
10	DIO10	W	0h	Toggles bit DOUT31_0.DIO10 0h = No effect 1h = Toggle

Table 18-59. DOUTTGL31_0 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
9	DIO9	W	0h	Toggles bit DOUT31_0.DIO9 0h = No effect 1h = Toggle
8	DIO8	W	0h	Toggles bit DOUT31_0.DIO8 0h = No effect 1h = Toggle
7	DIO7	W	0h	Toggles bit DOUT31_0.DIO7 0h = No effect 1h = Toggle
6	DIO6	W	0h	Toggles bit DOUT31_0.DIO6 0h = No effect 1h = Toggle
5	DIO5	W	0h	Toggles bit DOUT31_0.DIO5 0h = No effect 1h = Toggle
4	DIO4	W	0h	Toggles bit DOUT31_0.DIO4 0h = No effect 1h = Toggle
3	DIO3	W	0h	Toggles bit DOUT31_0.DIO3 0h = No effect 1h = Toggle
2	DIO2	W	0h	Toggles bit DOUT31_0.DIO2 0h = No effect 1h = Toggle
1	DIO1	W	0h	Toggles bit DOUT31_0.DIO1 0h = No effect 1h = Toggle
0	DIO0	W	0h	Toggles bit DOUT31_0.DIO0 0h = No effect 1h = Toggle

18.10.21 DOUTTGL3_0 Register (Offset = 300h) [Reset = 00000000h]

DOUTTGL3_0 is shown in Table 18-60.

Return to the Summary Table.

Alias regiser to toggle the corresponding bits of DOUT31_0[3:0] register.

Table 18-60. DOUTTGL3_0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-25	RESERVED	R	0h	Reserved
24	DIO3	W	0h	Toggles bit DOUT31_0.DIO3 0h = No effect 1h = Toggle
23-17	RESERVED	R	0h	Reserved
16	DIO2	W	0h	Toggles bit DOUT31_0.DIO2 0h = No effect 1h = Toggle
15-9	RESERVED	R	0h	Reserved
8	DIO1	W	0h	Toggles bit DOUT31_0.DIO1 0h = No effect 1h = Toggle
7-1	RESERVED	R	0h	Reserved
0	DIO0	W	0h	Toggles bit DOUT31_0.DIO0 0h = No effect 1h = Toggle

18.10.22 DOUTTGL7_4 Register (Offset = 304h) [Reset = 00000000h]

DOUTTGL7_4 is shown in Table 18-61.

Return to the Summary Table.

Alias regiser to toggle the corresponding bits of DOUT31_0[7:4] register.

Table 18-61. DOUTTGL7_4 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-25	RESERVED	R	0h	Reserved
24	DIO7	W	0h	Toggles bit DOUT31_0.DIO7 0h = No effect 1h = Toggle
23-17	RESERVED	R	0h	Reserved
16	DIO6	W	0h	Toggles bit DOUT31_0.DIO6 0h = No effect 1h = Toggle
15-9	RESERVED	R	0h	Reserved
8	DIO5	W	0h	Toggles bit DOUT31_0.DIO5 0h = No effect 1h = Toggle
7-1	RESERVED	R	0h	Reserved
0	DIO4	W	0h	Toggles bit DOUT31_0.DIO4 0h = No effect 1h = Toggle

18.10.23 DOUTTGL11_8 Register (Offset = 308h) [Reset = 00000000h]

DOUTTGL11_8 is shown in Table 18-62.

Return to the Summary Table.

Alias regiser to toggle the corresponding bits of DOUT31_0[11:8] register.

Table 18-62. DOUTTGL11_8 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-25	RESERVED	R	0h	Reserved
24	DIO11	W	0h	Toggles bit DOUT31_0.DIO11 0h = No effect 1h = Toggle
23-17	RESERVED	R	0h	Reserved
16	DIO10	W	0h	Toggles bit DOUT31_0.DIO10 0h = No effect 1h = Toggle
15-9	RESERVED	R	0h	Reserved
8	DIO9	W	0h	Toggles bit DOUT31_0.DIO9 0h = No effect 1h = Toggle
7-1	RESERVED	R	0h	Reserved
0	DIO8	W	0h	Toggles bit DOUT31_0.DIO8 0h = No effect 1h = Toggle

18.10.24 DOUTTGL15_12 Register (Offset = 30Ch) [Reset = 00000000h]

DOUTTGL15_12 is shown in Table 18-63.

Return to the Summary Table.

Alias regiser to toggle the corresponding bits of DOUT31_0[15:12] register.

Table 18-63. DOUTTGL15_12 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-25	RESERVED	R	0h	Reserved
24	DIO15	W	0h	Toggles bit DOUT31_0.DIO15 0h = No effect 1h = Toggle
23-17	RESERVED	R	0h	Reserved
16	DIO14	W	0h	Toggles bit DOUT31_0.DIO14 0h = No effect 1h = Toggle
15-9	RESERVED	R	0h	Reserved
8	DIO13	W	0h	Toggles bit DOUT31_0.DIO13 0h = No effect 1h = Toggle
7-1	RESERVED	R	0h	Reserved
0	DIO12	W	0h	Toggles bit DOUT31_0.DIO12 0h = No effect 1h = Toggle

18.10.25 DOUTTGL19_16 Register (Offset = 310h) [Reset = 00000000h]

DOUTTGL19_16 is shown in Table 18-64.

Return to the Summary Table.

Alias regiser to toggle the corresponding bits of DOUT31_0[19:16] register.

Table 18-64. DOUTTGL19_16 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-25	RESERVED	R	0h	Reserved
24	DIO19	W	0h	Toggles bit DOUT31_0.DIO19 0h = No effect 1h = Toggle
23-17	RESERVED	R	0h	Reserved
16	DIO18	W	0h	Toggles bit DOUT31_0.DIO18 0h = No effect 1h = Toggle
15-9	RESERVED	R	0h	Reserved
8	DIO17	W	0h	Toggles bit DOUT31_0.DIO17 0h = No effect 1h = Toggle
7-1	RESERVED	R	0h	Reserved
0	DIO16	W	0h	Toggles bit DOUT31_0.DIO16 0h = No effect 1h = Toggle

18.10.26 DOUTTGL23_20 Register (Offset = 314h) [Reset = 00000000h]

DOUTTGL23_20 is shown in Table 18-65.

Return to the Summary Table.

Alias regiser to toggle the corresponding bits of DOUT31_0[23:20] register.

Table 18-65. DOUTTGL23_20 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-25	RESERVED	R	0h	Reserved
24	DIO23	W	0h	Toggles bit DOUT31_0.DIO23 0h = No effect 1h = Toggle
23-17	RESERVED	R	0h	Reserved
16	DIO22	W	0h	Toggles bit DOUT31_0.DIO22 0h = No effect 1h = Toggle
15-9	RESERVED	R	0h	Reserved
8	DIO21	W	0h	Toggles bit DOUT31_0.DIO21 0h = No effect 1h = Toggle
7-1	RESERVED	R	0h	Reserved
0	DIO20	W	0h	Toggles bit DOUT31_0.DIO20 0h = No effect 1h = Toggle

18.10.27 DOUTTGL27_24 Register (Offset = 318h) [Reset = 00000000h]

DOUTTGL27_24 is shown in Table 18-66.

Return to the Summary Table.

Alias regiser to toggle the corresponding bits of DOUT31_0[27:24] register.

Table 18-66. DOUTTGL27_24 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-9	RESERVED	R	0h	Reserved
8	DIO25	W	0h	Toggles bit DOUT31_0.DIO25 0h = No effect 1h = Toggle
7-1	RESERVED	R	0h	Reserved
0	DIO24	W	0h	Toggles bit DOUT31_0.DIO24 0h = No effect 1h = Toggle

18.10.28 DOE3_0 Register (Offset = 400h) [Reset = 00000000h]

DOE3_0 is shown in Table 18-67.

Return to the Summary Table.

Alias register for byte access to DOE31_0[3:0] bits.

Table 18-67. DOE3_0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-25	RESERVED	R	0h	Reserved
24	DIO3	R/W	0h	Data output enable for DIO3 0h = Output disabled 1h = Output enabled
23-17	RESERVED	R	0h	Reserved
16	DIO2	R/W	0h	Data output enable for DIO2 0h = Output disabled 1h = Output enabled
15-9	RESERVED	R	0h	Reserved
8	DIO1	R/W	0h	Data output enable for DIO1 0h = Output disabled 1h = Output enabled
7-1	RESERVED	R	0h	Reserved
0	DIO0	R/W	0h	Data output enable for DIO0 0h = Output disabled 1h = Output enabled

18.10.29 DOE7_4 Register (Offset = 404h) [Reset = 00000000h]

DOE7_4 is shown in Table 18-68.

Return to the Summary Table.

Alias register for byte access to DOUT31_0[7:4] bits.

Table 18-68. DOE7_4 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-25	RESERVED	R	0h	Reserved
24	DIO7	R/W	0h	Data output enable for DIO7 0h = Output disabled 1h = Output enabled
23-17	RESERVED	R	0h	Reserved
16	DIO6	R/W	0h	Data output enable for DIO6 0h = Output disabled 1h = Output enabled
15-9	RESERVED	R	0h	Reserved
8	DIO5	R/W	0h	Data output enable for DIO5 0h = Output disabled 1h = Output enabled
7-1	RESERVED	R	0h	Reserved
0	DIO4	R/W	0h	Data output enable for DIO4 0h = Output disabled 1h = Output enabled

18.10.30 DOE11_8 Register (Offset = 408h) [Reset = 00000000h]

DOE11_8 is shown in Table 18-69.

Return to the Summary Table.

Alias register for byte access to DOUT31_0[11:8] bits.

Table 18-69. DOE11_8 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-25	RESERVED	R	0h	Reserved
24	DIO11	R/W	0h	Data output enable for DIO11 0h = Output disabled 1h = Output enabled
23-17	RESERVED	R	0h	Reserved
16	DIO10	R/W	0h	Data output enable for DIO10 0h = Output disabled 1h = Output enabled
15-9	RESERVED	R	0h	Reserved
8	DIO9	R/W	0h	Data output enable for DIO9 0h = Output disabled 1h = Output enabled
7-1	RESERVED	R	0h	Reserved
0	DIO8	R/W	0h	Data output enable for DIO8 0h = Output disabled 1h = Output enabled

18.10.31 DOE15_12 Register (Offset = 40Ch) [Reset = 00000000h]

DOE15_12 is shown in Table 18-70.

Return to the Summary Table.

Alias register for byte access to DOUT31_0[15:12] bits.

Table 18-70. DOE15_12 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-25	RESERVED	R	0h	Reserved
24	DIO15	R/W	0h	Data output enable for DIO15 0h = Output disabled 1h = Output enabled
23-17	RESERVED	R	0h	Reserved
16	DIO14	R/W	0h	Data output enable for DIO14 0h = Output disabled 1h = Output enabled
15-9	RESERVED	R	0h	Reserved
8	DIO13	R/W	0h	Data output enable for DIO13 0h = Output disabled 1h = Output enabled
7-1	RESERVED	R	0h	Reserved
0	DIO12	R/W	0h	Data output enable for DIO12 0h = Output disabled 1h = Output enabled

18.10.32 DOE19_16 Register (Offset = 410h) [Reset = 00000000h]

DOE19_16 is shown in Table 18-71.

Return to the Summary Table.

Alias register for byte access to DOUT31_0[19:16] bits.

Table 18-71. DOE19_16 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-25	RESERVED	R	0h	Reserved
24	DIO19	R/W	0h	Data output enable for DIO19 0h = Output disabled 1h = Output enabled
23-17	RESERVED	R	0h	Reserved
16	DIO18	R/W	0h	Data output enable for DIO18 0h = Output disabled 1h = Output enabled
15-9	RESERVED	R	0h	Reserved
8	DIO17	R/W	0h	Data output enable for DIO17 0h = Output disabled 1h = Output enabled
7-1	RESERVED	R	0h	Reserved
0	DIO16	R/W	0h	Data output enable for DIO16 0h = Output disabled 1h = Output enabled

18.10.33 DOE23_20 Register (Offset = 414h) [Reset = 00000000h]

DOE23_20 is shown in Table 18-72.

Return to the Summary Table.

Alias register for byte access to DOUT31_0[23:20] bits.

Table 18-72. DOE23_20 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-25	RESERVED	R	0h	Reserved
24	DIO23	R/W	0h	Data output enable for DIO23 0h = Output disabled 1h = Output enabled
23-17	RESERVED	R	0h	Reserved
16	DIO22	R/W	0h	Data output enable for DIO22 0h = Output disabled 1h = Output enabled
15-9	RESERVED	R	0h	Reserved
8	DIO21	R/W	0h	Data output enable for DIO21 0h = Output disabled 1h = Output enabled
7-1	RESERVED	R	0h	Reserved
0	DIO20	R/W	0h	Data output enable for DIO20 0h = Output disabled 1h = Output enabled

18.10.34 DOE27_24 Register (Offset = 418h) [Reset = 00000000h]

DOE27_24 is shown in Table 18-73.

Return to the Summary Table.

Alias register for byte access to DOUT31_0[27:24] bits.

Table 18-73. DOE27_24 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-9	RESERVED	R	0h	Reserved
8	DIO25	R/W	0h	Data output enable for DIO25 0h = Output disabled 1h = Output enabled
7-1	RESERVED	R	0h	Reserved
0	DIO24	R/W	0h	Data output enable for DIO24 0h = Output disabled 1h = Output enabled

18.10.35 DOE31_0 Register (Offset = 500h) [Reset = 00000000h]

DOE31_0 is shown in Table 18-74.

Return to the Summary Table.

Data output control for DIO 31 to 0 pins.

Table 18-74. DOE31_0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-26	RESERVED	R	0h	Reserved
25	DIO25	R/W	0h	Data output enable for DIO25 0h = Output disabled 1h = Output enabled
24	DIO24	R/W	0h	Data output enable for DIO24 0h = Output disabled 1h = Output enabled
23	DIO23	R/W	0h	Data output enable for DIO23 0h = Output disabled 1h = Output enabled
22	DIO22	R/W	0h	Data output enable for DIO22 0h = Output disabled 1h = Output enabled
21	DIO21	R/W	0h	Data output enable for DIO21 0h = Output disabled 1h = Output enabled
20	DIO20	R/W	0h	Data output enable for DIO20 0h = Output disabled 1h = Output enabled
19	DIO19	R/W	0h	Data output enable for DIO19 0h = Output disabled 1h = Output enabled
18	DIO18	R/W	0h	Data output enable for DIO18 0h = Output disabled 1h = Output enabled
17	DIO17	R/W	0h	Data output enable for DIO17 0h = Output disabled 1h = Output enabled
16	DIO16	R/W	0h	Data output enable for DIO16 0h = Output disabled 1h = Output enabled
15	DIO15	R/W	0h	Data output enable for DIO15 0h = Output disabled 1h = Output enabled
14	DIO14	R/W	0h	Data output enable for DIO14 0h = Output disabled 1h = Output enabled
13	DIO13	R/W	0h	Data output enable for DIO13 0h = Output disabled 1h = Output enabled
12	DIO12	R/W	0h	Data output enable for DIO12 0h = Output disabled 1h = Output enabled
11	DIO11	R/W	0h	Data output enable for DIO11 0h = Output disabled 1h = Output enabled
10	DIO10	R/W	0h	Data output enable for DIO10 0h = Output disabled 1h = Output enabled

Table 18-74. DOE31_0 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
9	DIO9	R/W	0h	Data output enable for DIO9 0h = Output disabled 1h = Output enabled
8	DIO8	R/W	0h	Data output enable for DIO8 0h = Output disabled 1h = Output enabled
7	DIO7	R/W	0h	Data output enable for DIO7 0h = Output disabled 1h = Output enabled
6	DIO6	R/W	0h	Data output enable for DIO6 0h = Output disabled 1h = Output enabled
5	DIO5	R/W	0h	Data output enable for DIO5 0h = Output disabled 1h = Output enabled
4	DIO4	R/W	0h	Data output enable for DIO4 0h = Output disabled 1h = Output enabled
3	DIO3	R/W	0h	Data output enable for DIO3 0h = Output disabled 1h = Output enabled
2	DIO2	R/W	0h	Data output enable for DIO2 0h = Output disabled 1h = Output enabled
1	DIO1	R/W	0h	Data output enable for DIO1 0h = Output disabled 1h = Output enabled
0	DIO0	R/W	0h	Data output enable for DIO0 0h = Output disabled 1h = Output enabled

18.10.36 DOESET31_0 Register (Offset = 510h) [Reset = 00000000h]

DOESET31_0 is shown in Table 18-75.

Return to the Summary Table.

Alias regiser to set the corresponding bits of DOE31_0 register.

Table 18-75. DOESET31_0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-26	RESERVED	R	0h	Reserved
25	DIO25	W	0h	Sets bit DOE31_0.DIO25 0h = No effect 1h = Set
24	DIO24	W	0h	Sets bit DOE31_0.DIO24 0h = No effect 1h = Set
23	DIO23	W	0h	Sets bit DOE31_0.DIO23 0h = No effect 1h = Set
22	DIO22	W	0h	Sets bit DOE31_0.DIO22 0h = No effect 1h = Set
21	DIO21	W	0h	Sets bit DOE31_0.DIO21 0h = No effect 1h = Set
20	DIO20	W	0h	Sets bit DOE31_0.DIO20 0h = No effect 1h = Set
19	DIO19	W	0h	Sets bit DOE31_0.DIO19 0h = No effect 1h = Set
18	DIO18	W	0h	Sets bit DOE31_0.DIO18 0h = No effect 1h = Set
17	DIO17	W	0h	Sets bit DOE31_0.DIO17 0h = No effect 1h = Set
16	DIO16	W	0h	Sets bit DOE31_0.DIO16 0h = No effect 1h = Set
15	DIO15	W	0h	Sets bit DOE31_0.DIO15 0h = No effect 1h = Set
14	DIO14	W	0h	Sets bit DOE31_0.DIO14 0h = No effect 1h = Set
13	DIO13	W	0h	Sets bit DOE31_0.DIO13 0h = No effect 1h = Set
12	DIO12	W	0h	Sets bit DOE31_0.DIO12 0h = No effect 1h = Set
11	DIO11	W	0h	Sets bit DOE31_0.DIO11 0h = No effect 1h = Set
10	DIO10	W	0h	Sets bit DOE31_0.DIO10 0h = No effect 1h = Set

Table 18-75. DOESET31_0 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
9	DIO9	W	0h	Sets bit DOE31_0.DIO9 0h = No effect 1h = Set
8	DIO8	W	0h	Sets bit DOE31_0.DIO8 0h = No effect 1h = Set
7	DIO7	W	0h	Sets bit DOE31_0.DIO7 0h = No effect 1h = Set
6	DIO6	W	0h	Sets bit DOE31_0.DIO6 0h = No effect 1h = Set
5	DIO5	W	0h	Sets bit DOE31_0.DIO5 0h = No effect 1h = Set
4	DIO4	W	0h	Sets bit DOE31_0.DIO4 0h = No effect 1h = Set
3	DIO3	W	0h	Sets bit DOE31_0.DIO3 0h = No effect 1h = Set
2	DIO2	W	0h	Sets bit DOE31_0.DIO2 0h = No effect 1h = Set
1	DIO1	W	0h	Sets bit DOE31_0.DIO1 0h = No effect 1h = Set
0	DIO0	W	0h	Sets bit DOE31_0.DIO0 0h = No effect 1h = Set

18.10.37 DOECLR31_0 Register (Offset = 520h) [Reset = 00000000h]

DOECLR31_0 is shown in Table 18-76.

Return to the Summary Table.

Alias regiser to clear the corresponding bits of DOE31_0 register.

Table 18-76. DOECLR31_0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-26	RESERVED	R	0h	Reserved
25	DIO25	W	0h	Clears bit DOE31_0.DIO25 0h = No effect 1h = Clear
24	DIO24	W	0h	Clears bit DOE31_0.DIO24 0h = No effect 1h = Clear
23	DIO23	W	0h	Clears bit DOE31_0.DIO23 0h = No effect 1h = Clear
22	DIO22	W	0h	Clears bit DOE31_0.DIO22 0h = No effect 1h = Clear
21	DIO21	W	0h	Clears bit DOE31_0.DIO21 0h = No effect 1h = Clear
20	DIO20	W	0h	Clears bit DOE31_0.DIO20 0h = No effect 1h = Clear
19	DIO19	W	0h	Clears bit DOE31_0.DIO19 0h = No effect 1h = Clear
18	DIO18	W	0h	Clears bit DOE31_0.DIO18 0h = No effect 1h = Clear
17	DIO17	W	0h	Clears bit DOE31_0.DIO17 0h = No effect 1h = Clear
16	DIO16	W	0h	Clears bit DOE31_0.DIO16 0h = No effect 1h = Clear
15	DIO15	W	0h	Clears bit DOE31_0.DIO15 0h = No effect 1h = Clear
14	DIO14	W	0h	Clears bit DOE31_0.DIO14 0h = No effect 1h = Clear
13	DIO13	W	0h	Clears bit DOE31_0.DIO13 0h = No effect 1h = Clear
12	DIO12	W	0h	Clears bit DOE31_0.DIO12 0h = No effect 1h = Clear
11	DIO11	W	0h	Clears bit DOE31_0.DIO11 0h = No effect 1h = Clear
10	DIO10	W	0h	Clears bit DOE31_0.DIO10 0h = No effect 1h = Clear

Table 18-76. DOECLR31_0 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
9	DIO9	W	0h	Clears bit DOE31_0.DIO9 0h = No effect 1h = Clear
8	DIO8	W	0h	Clears bit DOE31_0.DIO8 0h = No effect 1h = Clear
7	DIO7	W	0h	Clears bit DOE31_0.DIO7 0h = No effect 1h = Clear
6	DIO6	W	0h	Clears bit DOE31_0.DIO6 0h = No effect 1h = Clear
5	DIO5	W	0h	Clears bit DOE31_0.DIO5 0h = No effect 1h = Clear
4	DIO4	W	0h	Clears bit DOE31_0.DIO4 0h = No effect 1h = Clear
3	DIO3	W	0h	Clears bit DOE31_0.DIO3 0h = No effect 1h = Clear
2	DIO2	W	0h	Clears bit DOE31_0.DIO2 0h = No effect 1h = Clear
1	DIO1	W	0h	Clears bit DOE31_0.DIO1 0h = No effect 1h = Clear
0	DIO0	W	0h	Clears bit DOE31_0.DIO0 0h = No effect 1h = Clear

18.10.38 DOETGL31_0 Register (Offset = 530h) [Reset = 00000000h]

DOETGL31_0 is shown in Table 18-77.

Return to the Summary Table.

Alias regiser to toggle the corresponding bits of DOE31_0 register.

Table 18-77. DOETGL31_0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-26	RESERVED	R	0h	Reserved
25	DIO25	W	0h	Toggles bit DOE31_0.DIO25 0h = No effect 1h = Toggle
24	DIO24	W	0h	Toggles bit DOE31_0.DIO24 0h = No effect 1h = Toggle
23	DIO23	W	0h	Toggles bit DOE31_0.DIO23 0h = No effect 1h = Toggle
22	DIO22	W	0h	Toggles bit DOE31_0.DIO22 0h = No effect 1h = Toggle
21	DIO21	W	0h	Toggles bit DOE31_0.DIO21 0h = No effect 1h = Toggle
20	DIO20	W	0h	Toggles bit DOE31_0.DIO20 0h = No effect 1h = Toggle
19	DIO19	W	0h	Toggles bit DOE31_0.DIO19 0h = No effect 1h = Toggle
18	DIO18	W	0h	Toggles bit DOE31_0.DIO18 0h = No effect 1h = Toggle
17	DIO17	W	0h	Toggles bit DOE31_0.DIO17 0h = No effect 1h = Toggle
16	DIO16	W	0h	Toggles bit DOE31_0.DIO16 0h = No effect 1h = Toggle
15	DIO15	W	0h	Toggles bit DOE31_0.DIO15 0h = No effect 1h = Toggle
14	DIO14	W	0h	Toggles bit DOE31_0.DIO14 0h = No effect 1h = Toggle
13	DIO13	W	0h	Toggles bit DOE31_0.DIO13 0h = No effect 1h = Toggle
12	DIO12	W	0h	Toggles bit DOE31_0.DIO12 0h = No effect 1h = Toggle
11	DIO11	W	0h	Toggles bit DOE31_0.DIO11 0h = No effect 1h = Toggle
10	DIO10	W	0h	Toggles bit DOE31_0.DIO10 0h = No effect 1h = Toggle

Table 18-77. DOETGL31_0 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
9	DIO9	W	0h	Toggles bit DOE31_0.DIO9 0h = No effect 1h = Toggle
8	DIO8	W	0h	Toggles bit DOE31_0.DIO8 0h = No effect 1h = Toggle
7	DIO7	W	0h	Toggles bit DOE31_0.DIO7 0h = No effect 1h = Toggle
6	DIO6	W	0h	Toggles bit DOE31_0.DIO6 0h = No effect 1h = Toggle
5	DIO5	W	0h	Toggles bit DOE31_0.DIO5 0h = No effect 1h = Toggle
4	DIO4	W	0h	Toggles bit DOE31_0.DIO4 0h = No effect 1h = Toggle
3	DIO3	W	0h	Toggles bit DOE31_0.DIO3 0h = No effect 1h = Toggle
2	DIO2	W	0h	Toggles bit DOE31_0.DIO2 0h = No effect 1h = Toggle
1	DIO1	W	0h	Toggles bit DOE31_0.DIO1 0h = No effect 1h = Toggle
0	DIO0	W	0h	Toggles bit DOE31_0.DIO0 0h = No effect 1h = Toggle

18.10.39 DIN3_0 Register (Offset = 600h) [Reset = 00000000h]

DIN3_0 is shown in Table 18-78.

Return to the Summary Table.

Alias register for byte access to DIN31_0[3:0] bits.

Table 18-78. DIN3_0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-25	RESERVED	R	0h	Reserved
24	DIO3	R	0h	Data input from DIO3 0h = Input value is 0 1h = Input value is 1
23-17	RESERVED	R	0h	Reserved
16	DIO2	R	0h	Data input from DIO2 0h = Input value is 0 1h = Input value is 1
15-9	RESERVED	R	0h	Reserved
8	DIO1	R	0h	Data input from DIO1 0h = Input value is 0 1h = Input value is 1
7-1	RESERVED	R	0h	Reserved
0	DIO0	R	0h	Data input from DIO0 0h = Input value is 0 1h = Input value is 1

18.10.40 DIN7_4 Register (Offset = 604h) [Reset = 00000000h]

DIN7_4 is shown in Table 18-79.

Return to the Summary Table.

Alias register for byte access to DIN31_0[7:4] bits.

Table 18-79. DIN7_4 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-25	RESERVED	R	0h	Reserved
24	DIO7	R	0h	Data input from DIO7 0h = Input value is 0 1h = Input value is 1
23-17	RESERVED	R	0h	Reserved
16	DIO6	R	0h	Data input from DIO6 0h = Input value is 0 1h = Input value is 1
15-9	RESERVED	R	0h	Reserved
8	DIO5	R	0h	Data input from DIO5 0h = Input value is 0 1h = Input value is 1
7-1	RESERVED	R	0h	Reserved
0	DIO4	R	0h	Data input from DIO4 0h = Input value is 0 1h = Input value is 1

18.10.41 DIN11_8 Register (Offset = 608h) [Reset = 00000000h]

DIN11_8 is shown in Table 18-80.

Return to the Summary Table.

Alias register for byte access to DIN31_0[11:8] bits.

Table 18-80. DIN11_8 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-25	RESERVED	R	0h	Reserved
24	DIO11	R	0h	Data input from DIO11 0h = Input value is 0 1h = Input value is 1
23-17	RESERVED	R	0h	Reserved
16	DIO10	R	0h	Data input from DIO10 0h = Input value is 0 1h = Input value is 1
15-9	RESERVED	R	0h	Reserved
8	DIO9	R	0h	Data input from DIO9 0h = Input value is 0 1h = Input value is 1
7-1	RESERVED	R	0h	Reserved
0	DIO8	R	0h	Data input from DIO8 0h = Input value is 0 1h = Input value is 1

18.10.42 DIN15_12 Register (Offset = 60Ch) [Reset = 00000000h]

DIN15_12 is shown in Table 18-81.

Return to the Summary Table.

Alias register for byte access to DIN31_0[15:12] bits.

Table 18-81. DIN15_12 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-25	RESERVED	R	0h	Reserved
24	DIO15	R	0h	Data input from DIO15 0h = Input value is 0 1h = Input value is 1
23-17	RESERVED	R	0h	Reserved
16	DIO14	R	0h	Data input from DIO14 0h = Input value is 0 1h = Input value is 1
15-9	RESERVED	R	0h	Reserved
8	DIO13	R	0h	Data input from DIO13 0h = Input value is 0 1h = Input value is 1
7-1	RESERVED	R	0h	Reserved
0	DIO12	R	0h	Data input from DIO12 0h = Input value is 0 1h = Input value is 1

18.10.43 DIN19_16 Register (Offset = 610h) [Reset = 00000000h]

DIN19_16 is shown in Table 18-82.

Return to the Summary Table.

Alias register for byte access to DIN31_0[19:16] bits.

Table 18-82. DIN19_16 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-25	RESERVED	R	0h	Reserved
24	DIO19	R	0h	Data input from DIO19 0h = Input value is 0 1h = Input value is 1
23-17	RESERVED	R	0h	Reserved
16	DIO18	R	0h	Data input from DIO18 0h = Input value is 0 1h = Input value is 1
15-9	RESERVED	R	0h	Reserved
8	DIO17	R	0h	Data input from DIO17 0h = Input value is 0 1h = Input value is 1
7-1	RESERVED	R	0h	Reserved
0	DIO16	R	0h	Data input from DIO16 0h = Input value is 0 1h = Input value is 1

18.10.44 DIN23_20 Register (Offset = 614h) [Reset = 00000000h]

DIN23_20 is shown in Table 18-83.

Return to the Summary Table.

Alias register for byte access to DIN31_0[23:20] bits.

Table 18-83. DIN23_20 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-25	RESERVED	R	0h	Reserved
24	DIO23	R	0h	Data input from DIO23 0h = Input value is 0 1h = Input value is 1
23-17	RESERVED	R	0h	Reserved
16	DIO22	R	0h	Data input from DIO22 0h = Input value is 0 1h = Input value is 1
15-9	RESERVED	R	0h	Reserved
8	DIO21	R	0h	Data input from DIO21 0h = Input value is 0 1h = Input value is 1
7-1	RESERVED	R	0h	Reserved
0	DIO20	R	0h	Data input from DIO20 0h = Input value is 0 1h = Input value is 1

18.10.45 DIN27_24 Register (Offset = 618h) [Reset = 00000000h]

DIN27_24 is shown in Table 18-84.

Return to the Summary Table.

Alias register for byte access to DIN31_0[27:24] bits.

Table 18-84. DIN27_24 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-9	RESERVED	R	0h	Reserved
8	DIO25	R	0h	Data input from DIO25 0h = Input value is 0 1h = Input value is 1
7-1	RESERVED	R	0h	Reserved
0	DIO24	R	0h	Data input from DIO24 0h = Input value is 0 1h = Input value is 1

18.10.46 DIN31_0 Register (Offset = 700h) [Reset = 00000000h]

DIN31_0 is shown in Table 18-85.

Return to the Summary Table.

Data input from DIO 31 to 0 pins.

Table 18-85. DIN31_0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-26	RESERVED	R	0h	Reserved
25	DIO25	R	0h	Data input from DIO25 0h = Input value is 0 1h = Input value is 1
24	DIO24	R	0h	Data input from DIO24 0h = Input value is 0 1h = Input value is 1
23	DIO23	R	0h	Data input from DIO23 0h = Input value is 0 1h = Input value is 1
22	DIO22	R	0h	Data input from DIO22 0h = Input value is 0 1h = Input value is 1
21	DIO21	R	0h	Data input from DIO21 0h = Input value is 0 1h = Input value is 1
20	DIO20	R	0h	Data input from DIO20 0h = Input value is 0 1h = Input value is 1
19	DIO19	R	0h	Data input from DIO19 0h = Input value is 0 1h = Input value is 1
18	DIO18	R	0h	Data input from DIO18 0h = Input value is 0 1h = Input value is 1
17	DIO17	R	0h	Data input from DIO17 0h = Input value is 0 1h = Input value is 1
16	DIO16	R	0h	Data input from DIO16 0h = Input value is 0 1h = Input value is 1
15	DIO15	R	0h	Data input from DIO15 0h = Input value is 0 1h = Input value is 1
14	DIO14	R	0h	Data input from DIO14 0h = Input value is 0 1h = Input value is 1
13	DIO13	R	0h	Data input from DIO13 0h = Input value is 0 1h = Input value is 1
12	DIO12	R	0h	Data input from DIO12 0h = Input value is 0 1h = Input value is 1
11	DIO11	R	0h	Data input from DIO11 0h = Input value is 0 1h = Input value is 1
10	DIO10	R	0h	Data input from DIO10 0h = Input value is 0 1h = Input value is 1

Table 18-85. DIN31_0 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
9	DIO9	R	0h	Data input from DIO9 0h = Input value is 0 1h = Input value is 1
8	DIO8	R	0h	Data input from DIO8 0h = Input value is 0 1h = Input value is 1
7	DIO7	R	0h	Data input from DIO7 0h = Input value is 0 1h = Input value is 1
6	DIO6	R	0h	Data input from DIO6 0h = Input value is 0 1h = Input value is 1
5	DIO5	R	0h	Data input from DIO5 0h = Input value is 0 1h = Input value is 1
4	DIO4	R	0h	Data input from DIO4 0h = Input value is 0 1h = Input value is 1
3	DIO3	R	0h	Data input from DIO3 0h = Input value is 0 1h = Input value is 1
2	DIO2	R	0h	Data input from DIO2 0h = Input value is 0 1h = Input value is 1
1	DIO1	R	0h	Data input from DIO1 0h = Input value is 0 1h = Input value is 1
0	DIO0	R	0h	Data input from DIO0 0h = Input value is 0 1h = Input value is 1

18.10.47 EVTCFG Register (Offset = 800h) [Reset = 00000000h]

EVTCFG is shown in Table 18-86.

Return to the Summary Table.

Event configuration. This register is used to select DIO for GPIO to publish event on SVT event fabric (EVTSVT). It also contains enable bit that is used to mask the event.

Table 18-86. EVTCFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-9	RESERVED	R	0h	Reserved
8	EVTEN	R/W	0h	Enables GPIO to publish edge qualified selected DIO event on SVT event fabric. Design note: The edge detector flop is cleared automatically for the selected DIO once the event is published. 0h = Disable 1h = Enable
7-6	RESERVED	R	0h	Reserved
5-0	DIOSEL	R/W	0h	This is used to select DIO for event generation. For example, DIOSEL = 0x0 selects DIO0 and DIOSEL = 0x8 selects DIO8. 0h = Minimum value 3Fh = Maximum value

Chapter 19

Universal Asynchronous Receiver/Transmitter (UART)

This chapter describes the features and functions of the Universal Asynchronous Receiver/Transmitter (UART).

19.1 Introduction

The UART supports the following features:

- Programmable baud rate generator allowing speeds up to 3Mbps
- Separate 8 × 8 transmit (TX) and 8 × 12 receive (RX) first-in first-out (FIFO) buffers to reduce CPU interrupt service loading
- Programmable FIFO length, including 1-byte deep operation providing conventional double-buffered interface
- FIFO trigger levels of ¼, ½, ¾.
- Standard asynchronous communication bits for start, stop, and parity
- Line-break generation and detection
- Fully programmable serial interface characteristics:
  - 5, 6, 7, or 8 data bits
  - Even, odd, stick, or no parity bit generation and detection
  - 1 or 2 stop-bit generation
- FIFO, RX FIFO RX time-out, modem status, and error conditions
- Standard FIFO-level and end-of-transmission interrupts
- Efficient transfers using micro direct memory access controller (µDMA):
  - Separate channels for transmit and receive.
  - Receive single request asserted when data is in the FIFO; burst request asserted at programmed FIFO level.
  - Transmit single request is asserted when there is space in the FIFO; burst request is asserted at programmed FIFO level.
- Programmable hardware flow control
- Support for standard IrDA and low power IrDA modes.
- Provision to combine both TX and RX FIFOs in transmit mode

19.2 Block Diagram

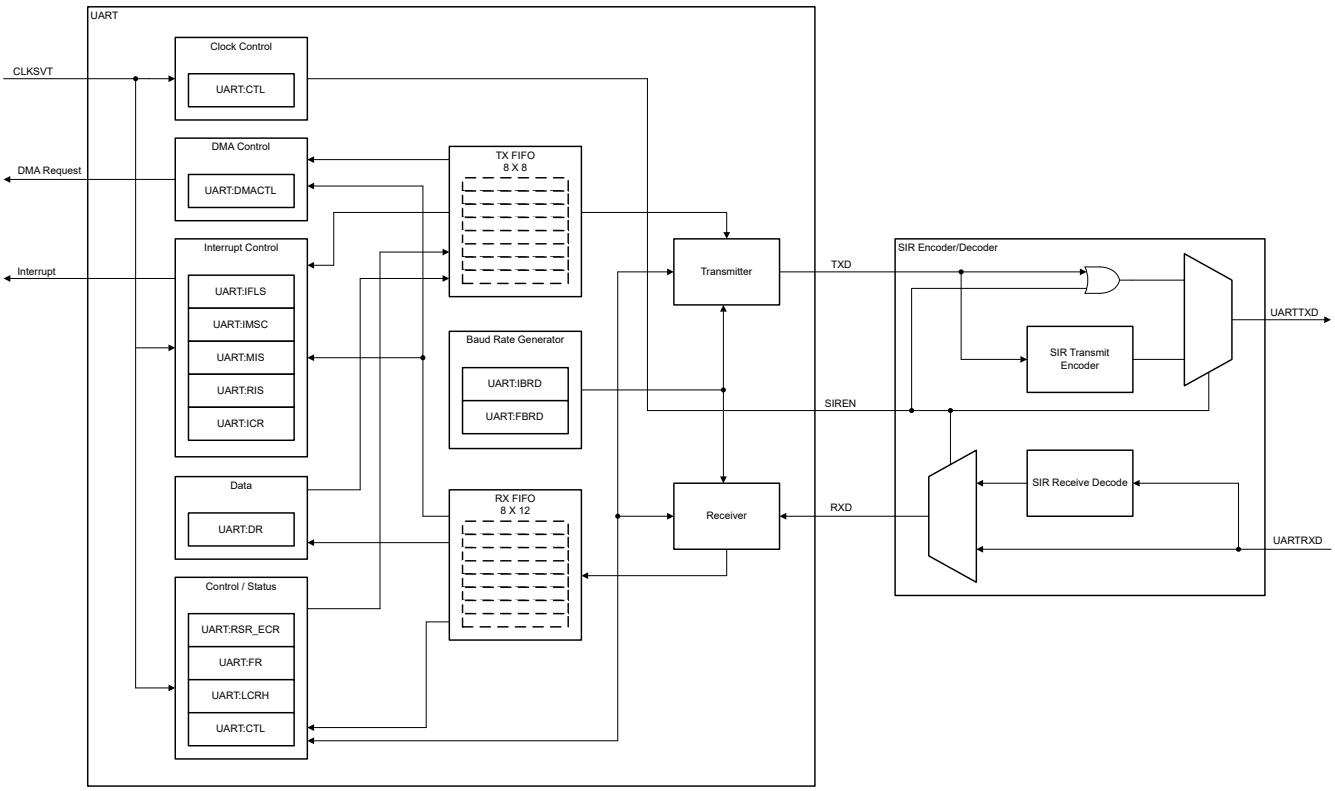


Figure 19-1. UART Block Diagram

19.3 Functional Description

The CC23xx UART performs the functions of parallel-to-serial and serial-to-parallel conversions. The CC23xx UART is similar in functionality to a 16C550 UART, but is not register compatible. The UART is configured for transmit and receive through the UART Control Register (UART:CTL) TXE and RXE bits. Transmit and receive are both enabled out of reset. Before any control registers are programmed, the UART must be disabled by clearing the UART:CTL UARTEN register bit. If the UART is disabled during a transmit or receive operation, the current transaction completes before the UART stops.

19.3.1 Transmit and Receive Logic

The transmit logic performs parallel-to-serial conversion on the data read from the TX FIFO. The control logic outputs the serial bit stream beginning with a start bit and followed by the data bits (LSB first), parity bit, and the stop bits, according to the programmed configuration in the control registers. For details, see Figure 19-2

The receive logic performs serial-to-parallel conversion on the received bit stream after a valid start pulse is detected. Overrun, parity, frame error checking, and line-break detection are also performed, and their status accompanies the data written to the RX FIFO.

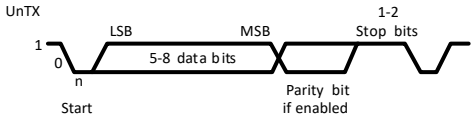


Figure 19-2. UART Character Frame

19.3.2 Baud Rate Generation

The baud rate divisor (BRD) is a 22-bit number consisting of a 16-bit integer and a 6-bit fractional part. The number formed by these two values is used by the baud rate generator to determine the bit period. Having a fractional baud rate divider allows the UART to generate all standard baud rates.

The 16-bit integer is loaded through the UART Integer Baud Rate Divisor Register (UART.IBRD), and the 6-bit fractional part is loaded with the UART Fractional Baud Rate Divisor Register (UART.FBRD).

Equation 4 shows the relationship of the BRD and the system clock.

$$BRD = BRDI + BRDF = CLKSVT / (ClkDiv \times Baud\ Rate) \quad (4)$$

where:

- BRDI is the integer part of the BRD.
- BRDF is the fractional part, separated by a decimal place.
- CLKSVT is the system clock connected to the UART.
- ClkDiv is 16.

The 6-bit fractional number that is loaded into the UART:FBRD.DIVFRAC bit field can be calculated by taking the fractional part of the baud rate divisor, multiplying by 64, and adding 0.5 to account for rounding errors, as shown by Equation 5

$$UART.FBRD[5:0]\ DIVFRAC = integer\ (BRDF \times 64 + 0.5) \quad (5)$$

Along with the UART Line Control High Byte Register (UART.LCRH), the UART.IBRD and the UART.FBRD registers form an internal 30-bit register. This internal register is updated only when a write operation to the UART.LCRH register is performed, so a write to the UART.LCRH register must follow any changes to the BRD for the changes to take effect.

The four possible sequences to update the baud-rate registers are as follows:

- UART.IBRD write, UART.FBRD write, and UART.LCRH write
- UART.FBRD write, UART.IBRD write, and UART.LCRH write

- UART.IBRD write and UART.LCRH write
- UART.FBRD write and UART.LCRH write

For an example calculation see Section 19.5.

19.3.3 FIFO Operation

The UART has two 8-entry FIFOs. One FIFO for transmit and one FIFO for receive. Both FIFOs are accessed through the UART Data Register, UART.DR. Read operations of the UART.DR register returns a 12-bit value consisting of 8 data bits and 4 error flags, while write operations place 8-bit data in the TX FIFO.

Out of reset, both FIFOs are disabled and act as 1-byte-deep holding registers. The FIFOs are enabled by setting the UART.LCRH[4] FEN bit.

FIFO status can be monitored through the UART Flag Register, UART.FR and the UART Receive Status Register, UART.RSR_ECR. Hardware monitors empty, full, and overrun conditions. The UART.FR register contains empty and full flags (TXFE, TXFF, RXFE, and RXFF bits), and the UART.RSR_ECR register shows overrun status through the OE bit. If the FIFOs are disabled, the empty and full flags are set according to the status of the 1-byte deep holding registers.

The trigger points at which the FIFOs generate interrupts are controlled through the UART Interrupt FIFO Level Select Register (UART:IFLS). Both FIFOs can be individually configured to trigger interrupts at different levels. Available configurations include ¼, ½, and ¾. For example, if the ¼ option is selected for the receive FIFO, the UART generates a receive interrupt after 4 data bytes are received. Out of reset, both FIFOs are configured to trigger an interrupt at the ½ mark.

19.3.3.1 FIFO Remapping

The UART supports the concatenation of TX and RX FIFOs in TX only mode—leading to 16 TX entries. Remapping and concatenation for RX only mode is not supported. This mode is enabled by setting the UART.CTL[6] FCEN bit to 1.

19.3.4 Data Transmission

Data received or transmitted is stored in two FIFOs, though the RX FIFO has an extra 4 bits per character for status information. For transmission, data is written into the TX FIFO. If the UART is enabled, a data frame starts transmitting with the parameters indicated in the UART.LCRH register. Data transmission continues until no data is left in the TX FIFO. The UART Flag Register (UART.FR) BUSY bit is asserted as soon as data is written to the TX FIFO (that is, if the FIFO is not empty), and remains asserted while data is transmitting. The BUSY bit is negated only when the TX FIFO is empty, and the last character has been transmitted from the shift register, including the stop bits. The UART can indicate that it is busy even if the UART is no longer enabled.

When the receiver is idle (the UARTRXD signal is continuously 1), and the data input goes low (a start bit was received), the receive counter begins running and data is sampled.

The start bit is valid and recognized if the UARTRXD signal is still low on the eighth cycle of the baud rate clock otherwise the start bit is ignored. After a valid start bit is detected, successive data bits are sampled every sixteenth cycle of the baud rate clock. The parity bit is then checked if parity mode is enabled. Data length and parity are defined in the UART:LCRH register.

Lastly, a valid stop bit is confirmed if the UARTRXD signal is high; otherwise, a framing error has occurred. When a full word is received, the data is stored in the receive FIFO with any error bits associated with that word.

19.3.5 Flow Control

Flow control can be accomplished by hardware and the following sections describe the implementation method. Hardware flow control between two devices is accomplished by connecting the RTS (request-to-send) output to the CTS (clear-to-send) input on the receiving device and connecting the RTS output on the receiving device to the CTS input. The RTS output signal is low active, the CTS input expects a low signal on a send request, as shown in Figure 19-3.

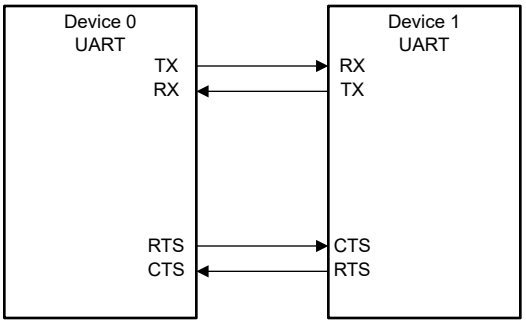


Figure 19-3. UART Flow Control

The CTS input controls the transmitter, the Device 0 and Device 1 transmitter can only transmit data when their CTS input is asserted low. When RTS flow control is enabled, the RTS output signal indicates the state of the receive FIFO. For example, the CTS of the Device 1 remains asserted low until the preprogrammed RX FIFO level of Device 0 is reached, indicating that the receive FIFO of Device 0 has no space to store additional characters.

The UART:CTL register bits CTSEN and RTSEN specify the flow control mode as shown in the following table.

CTSEN	RTSEN	Description
1	1	RTS and CTS flow control enabled
1	0	Only CTS flow control enabled
0	1	Only RTS flow control enabled
0	0	RTS and CTS flow control disabled

When RTSEN is set to 1, the value of the UART:CTL.RTS bit is ignored and the RTS output signal is generated by the hardware trigger levels as described below. When RTSEN bit is cleared, the RTS signal output is controlled by the UART:CTL.RTS bit for SW control.

RTS Flow Control

The RTS flow control logic is linked to the programmable receive FIFO trigger levels. The trigger level can be configured using the UART.IFLS register. When RTS flow control is enabled, the RTS is asserted (low) until the receive FIFO is filled up to the trigger level. When the receive FIFO trigger level is reached, the RTS signal is deasserted (high), indicating that there is no more room to receive any more data. The transmission of data is expected to cease after the current character has been transmitted. The RTS signal is reasserted (low) when data has been read out of the receive FIFO so that the FIFO is filled to less than the trigger level. If RTS flow control is disabled and the UART is still enabled, then data is received until the receive FIFO is full, or no more data is transmitted.

The RTS signal is deasserted when the FIFO trigger level is reached by putting the last received character into the FIFO. This means that on a back-to-back transmit, another character transfer can already be started by the sender prior to the RTS signal being deasserted. In such cases, the trigger level needs to be set to one level lower so that all data can be received and added into the FIFO.

CTS Flow Control

If CTS flow control is enabled, then the transmitter checks the CTS signal before transmitting the next byte. If the CTS signal is asserted (low), it transmits the byte otherwise transmission does not occur. The data continues to be transmitted while CTS is asserted (low), and the transmit FIFO is not empty. If the transmit FIFO is empty and the CTS signal is asserted (low) no data is transmitted. If the CTS signal is deasserted (high) and CTS flow control is enabled, then the current character transmission is completed before stopping. If CTS flow control is disabled and the UART is enabled, then the data continues to be transmitted until the transmit FIFO is empty.

Software Flow Control

Software flow control between two devices is accomplished by using interrupts to indicate the status of the UART. Interrupts can be generated for the CTS signal by setting the UART.IMSC[1] CTSMIM bit. The raw and masked interrupt status can be checked using the UART.RIS and UART.MIS registers. These interrupts can be cleared using the UART.ICR register.

19.3.6 IrDA Encoding and Decoding

When the UART.CTL[1] SIREN bit is set, the IrDA (SIR) encoder and decoder are enabled and provide hardware bit shaping for IrDA communication. In this protocol, from the transmitter perspective, a zero logic level is transmitted as a high pulse of 3/16th duration of the selected baud rate bit period, while logic one levels are transmitted as a static LOW signal.

The SIR decoder converts the IrDA-compliant receive signal into a bit stream for the UART core. The SIR receive logic interprets a high state as a logic one and low pulses as logic zeros.

Setting the UART.CTL SIRLP[2] bit enables low power mode. In the low-power mode, the width of the pulse is set to three times the time period of the IrLPBaud16 signal. The IrLPBaud16 signal is generated by dividing down CLKSVT (48MHz) according to the low-power divisor value written to the UART.UARTILPR register. The low-power divisor value is calculated as follows:

low-power divisor (ILPDVSR) = (CLKSVT / (FIrLPBaud16)) (6)

where FIrLPBaud16 is nominally 1.8432MHz.

The divisor must be selected such that 1.42MHz < FIrLPBaud16 < 2.12MHz, which results in a low-power pulse duration of 1.41-2.11µs (three times the period of IrLPBaud16).

Regardless of whether low-power mode is used or not UART.UARTILPR must be configured to generate IrLPBaud16 at the correct frequency to allow a normal-mode UART to receive data from a low-power mode UART, which can transmit pulses as small as 1.41µs. If reception from a low-power mode UART is not required then UART.UARTILPR can be left unconfigured.

Note

In low-power IrDA mode, the UART rejects random noise on the received serial data input by ignoring SIRIN pulses that are less than three periods of IrLPBaud16.

The maximum supported bit-rate for IrDA is 115.2Kbps.

19.3.7 Interrupts

The UART can generate interrupts when the following conditions are observed:

- Overrun error
- Break error
- Parity error
- Framing error
- Receive time-out
- Transmit (when the condition defined in the UART:IFLS TXSEL register bit is met)
- Receive (when the condition defined in the UART:IFLS RXSEL register bit is met)
- End of transmission (when no data on the TX line and TX FIFO underflow
- RX DMA Done
- TX DMA Done

All of the interrupt events are ORed together before being sent to the MCU event fabric, so the UART can only generate a single interrupt request at any given time. The software can service multiple interrupt events in a single interrupt service routine (ISR) by reading the UART Masked Interrupt Status Register (UART.MIS).

The interrupt events that can trigger a controller-level interrupt are defined in the UART Interrupt Mask Register (UART.IMSC) by setting the corresponding bits. If interrupts are not used, the raw interrupt status is always visible through the UART Raw Interrupt Status Register (UART.RIS).

Interrupts can be cleared (for the UART.MIS and UART.RIS registers) by setting the corresponding bit in the UART Interrupt Clear Register (UART.ICR).

The receive time-out interrupt is asserted when the RX FIFO is not empty, and no further data is received over a 32-bit period. The receive time-out interrupt is cleared either when the FIFO becomes empty through reading all the data (or by reading the holding register), or when the corresponding bit in the UART.ICR register is set.

The UART module provides the possibility of setting and clearing masks for every individual interrupt source using the UART Interrupt Mask Set/Clear Register (UART.IMSC). The five events that can cause combined interrupts to CPU are:

- RX: The receive interrupt changes state when one of the following events occurs:
  - If the FIFOs are enabled and the receive FIFO reaches the programmed trigger level. When this happens, the receive interrupt is asserted high. The receive interrupt is cleared by reading data from the receive FIFO until it becomes less than the trigger level, or by clearing the interrupt.
  - If the FIFOs are disabled (have a depth of one location) and data is received, thereby filling the location, the receive interrupt is asserted high. The receive interrupt is cleared by performing a single read of the receive FIFO, or by clearing the interrupt.
- TX: The transmit interrupt changes state when one of the following events occurs:
  - If the FIFOs are enabled and the transmit FIFO is equal to or lower than the programmed trigger level, then the transmit interrupt is asserted high. The transmit interrupt is cleared by writing data to the transmit FIFO until it becomes greater than the trigger level, or by clearing the interrupt.
  - If the FIFOs are disabled (have a depth of one location) and there is no data present in the transmitter's single location, the transmit interrupt is asserted high. The interrupt is cleared by performing a single write to the transmit FIFO, or by clearing the interrupt.
- RX time-out: The receive time-out interrupt is asserted when the receive FIFO is not empty, and no more data is received during a 32-bit period. The receive time-out interrupt is cleared either when the FIFO becomes empty through reading all the data (or by reading the holding register), or when 1 is written to the corresponding bit of the Interrupt Clear Register (UART:ICR).
- Modem status: The modem status interrupt is asserted if the modem status signal CTS toggles. It can be cleared using the corresponding clear bit in the UART.ICR register.
- Error: The error interrupt is asserted when an error occurs in the reception of data by the UART. The interrupt can be caused by a number of different error conditions:
  - framing
  - parity
  - break
  - overrun

The cause of the interrupt can be determined by reading the UART.RIS register or the UART.MIS register. The interrupt can be cleared by writing to the relevant bits of the UART.ICR register.

In addition to the five events produced by the UART module, two additional events are ORed to the interrupt line:

- RX DMA done: Indicates that the receiver µDMA task is completed. This is a level interrupt provided by the µDMA module.
- TX DMA done: Indicates that the transmit µDMA task is completed. This is a level interrupt provided by the µDMA module.

19.3.8 Loopback Operation

The UART can be placed into an internal loopback mode for diagnostic or debug work by setting the UART:CTL.LBE register bit. In loopback mode, data transmitted on the UARTTXD output is received on the UARTRXD input. The LBE bit must be set before the UART is enabled.

19.4 Interface to µDMA

This UART provide an interface to connect to the µDMA controller. Figure 19-4 shows the interface between the µDMA and UART.

- This interface contains four µDMA requests as outputs (UART0_RX_DMASREQ, UART0_RX_DMABREQ, UART0_TX_DMASREQ, and UART0_TX_DMABREQ). The µDMA interface also has two µDMA request clears as inputs (for clearing TX and RX µDMA requests).
- Each µDMA request signal remains asserted until the relevant µDMA clear signal is asserted. After the µDMA clear signal is deasserted, a request signal can become active again, if conditions are setup correctly. The µDMA clear signal must be connected to the µDMA active signal from the µDMA module. This signal is asserted when µDMA is granted access and is active. The µDMA active signal is deasserted when the µDMA transfer completes. Connecting the µDMA active signal from µDMA to the µDMA request clear input of the UART module ensures that no requests are generated by the UART module while the µDMA is active.
- The burst transfer and single transfer request signals are not mutually exclusive, and both can be asserted at the same time. For example, when there is more data than the watermark level in the receive FIFO, the burst transfer request and the single transfer request are asserted.
- The single and burst requests cannot be masked separately by the UART module and if corresponding µDMA (RX or TX) is enabled, both of these requests are sent to the µDMA.
- All request signals are deasserted if the UART is disabled or if the relevant µDMA enable bit (TXDMAE or RXDMAE) in the µDMA Control Register (UART:DMACTL) is cleared.

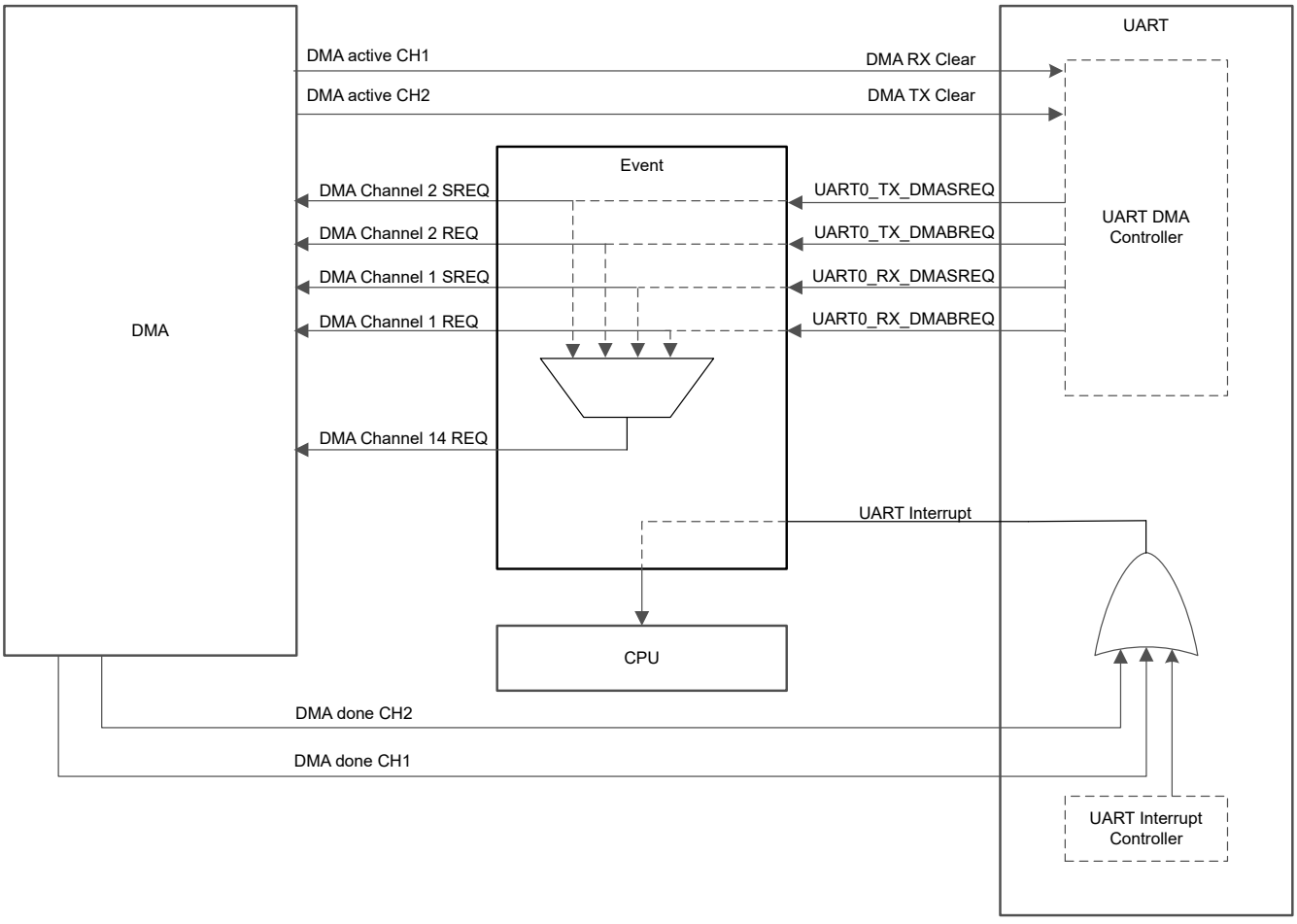


Figure 19-4. µDMA Example

When the UART is in the FIFO enabled mode, data transfers can be made by either single or burst transfers depending on the programmed watermark level and the amount of data in the FIFO. Table 19-1 lists the trigger points for the transmit and receive FIFOs. In addition, if the UART.DMACTL[2] DMAONERR bit is set, the µDMA receive request outputs (for single and burst requests) are disabled when the UART error interrupt is asserted (more specifically if any of the error interrupts in the RIS register, PERIS, BERIS, FERIS or OERIS are asserted). The µDMA receive request outputs remain inactive until the error bit is cleared. The µDMA transmit request outputs are unaffected.

Table 19-1. µDMA Trigger Points for the Transmit and Receive FIFOs

Watermark Level	Transmit Burst Length (number of empty locations)	Receive Burst Length (number of filled locations)
1/8	28	4
1/4	24	8
1/2	16	16
3/4	8	24
7/8	4	28

Sequence for Using µDMA Triggers

- Configure the EVTSVT.DMACH2SEL[2:0] IPID bit field for selecting UARTTXD as trigger source for µDMA.
- Configure the EVTSVT.DMACH3SEL[2:0] IPID bit field for selecting UARTRXD as trigger source for µDMA.
- Enable Transmit and Receive µDMA Enable by setting the UART.DMACTL TXDMAE[1] bit and UART.DMACTL[0] RXDMA to 1.
- Enable the µDMA clock by setting CLKCTL.CLKENSET0[17] µDMA bit to 1.
- Configure the dedicated µDMA channels for UARTTXD and UARTRXD through µDMA registers for setting the Source and Destination addresses, Arbitration size and Data size (see Chapter 15 for more details).
- Set the µDMA.CFG[0] MASTERENABLE bit.
- Configure the UART.IMSC[12] TXDMADONEIM bit and UART:IMSC[13] RXDMADONEIM bit to trigger the interrupts for µDMA Done.

Enable the UART by setting the UART.CTL[0] UARTEN bit.

19.5 Initialization and Configuration

The UART module provides four I/O signals to be routed to the DIOs. The following signals are selected through the IOCFGn registers in the IOC module.

- Inputs: RXD, CTS
- Outputs: TXD, RTS

CTS and RTS lines are active low.

Note

IOC must be configured before enabling the UART to avoid unwanted transitions on the input being processed as UART signals. When IOC is configured as UART-specific I/Os (RXD, CTS, TXD, or RTS), IOC sets static output driver enable to the DIO (output driver enable = 1 for output TXD and RTS and output driver enable = 0 for inputs RXD and CTS).

To enable and initialize the UART, use the following steps:

1. Enable the UART module in the CLKCTRL module by writing to the CLKCTRL.CLKENSET0[2] UART0 bit to 1. This enables the clock to UART.
2. Configure the IOC module to map UART signals to the correct GPIO pins. For more information on pin connections, see Chapter 18.

This section discusses the steps required to use a UART module. For this example, the UART clock is assumed to be 48 MHz, and the desired UART configuration is the following:

- Baud rate: 115 200
- Data length of 8 bits
- One stop bit
- No parity
- FIFOs disabled
- No interrupts

The first thing to consider when programming the UART is the BRD because the UART.IBRD and UART.FBRD registers must be written before the UART.LCRH register. The BRD can be calculated using the equation described in Section 19.3.2

$$BRD = 48\ 000\ 000 / (16 \times 115\ 200) = 26.0416 \quad (7)$$

The result of the previous equation indicates that the UART.IBRD[15:0] DIVINT bit field must be set to 26 decimal or 0x1A.

$$UART.FBRD[5:0]\ DIVFRAC = integer\ (0.0416 \times 64 + 0.5) = 3 \quad (8)$$

the previous equation calculates the value to be loaded into the UART.FBRD register.

With the BRD values available, the UART configuration is written to the module in the following order:

1. Disable the UART by clearing the UART.CTL[0] UARTEN bit.
2. Write the integer portion of the BRD to the UART.IBRD register.
3. Write the fractional portion of the BRD to the UART.FBRD register.
4. Write the desired serial parameters to the UART.LCRH register (in this case, a value of 0x0000 0060).
5. Enable the UART by setting the UART.CTL[0] UARTEN bit.

19.6 UART Registers

Table 19-2 lists the memory-mapped registers for the UART registers. All register offset addresses not listed in Table 19-2 should be considered as reserved locations and the register contents should not be modified.

Table 19-2. UART Registers

Offset	Acronym	Register Name	Section
0h	DR	Data	Go
4h	RSR_ECR	Status	Go
18h	FR	Flag	Go
20h	UARTILPR	IrDA Low-Power Counter Register	Go
24h	IBRD	Integer Baud-Rate Divisor	Go
28h	FBRD	Fractional Baud-Rate Divisor	Go
2Ch	LCRH	Line Control	Go
30h	CTL	Control	Go
34h	IFLS	Interrupt FIFO Level Select	Go
38h	IMSC	Interrupt Mask Set/Clear	Go
3Ch	RIS	Raw Interrupt Status	Go
40h	MIS	Masked Interrupt Status	Go
44h	ICR	Interrupt Clear	Go
48h	DMACTL	DMA Control	Go

Complex bit access types are encoded to fit into small table cells. Table 19-3 shows the codes that are used for access types in this section.

Table 19-3. UART Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

19.6.1 DR Register (Offset = 0h) [Reset = 00000000h]

DR is shown in Table 19-4.

Return to the Summary Table.

Data
For words to be transmitted:
- if the FIFOs are enabled (LCRH.FEN = 1), data written to this location is pushed onto the transmit FIFO
- if the FIFOs are not enabled (LCRH.FEN = 0), data is stored in the transmitter holding register (the bottom word of the transmit FIFO).
The write operation initiates transmission from the UART. The data is prefixed with a start bit, appended with the appropriate parity bit (if parity is enabled), and a stop bit.
The resultant word is then transmitted.
For received words:
- if the FIFOs are enabled (LCRH.FEN = 1), the data byte and the 4-bit status (break, frame, parity, and overrun) is pushed onto the 12-bit wide receive FIFO
- if the FIFOs are not enabled (LCRH.FEN = 0), the data byte and status are stored in the receiving holding register (the bottom word of the receive FIFO).
The received data byte is read by performing reads from this register along with the corresponding status information. The status information can also be read by a read of the RSR_ECR register.

Table 19-4. DR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-12	RESERVED	R	0h	Reserved
11	OE	R	0h	UART Overrun Error: This bit is set to 1 if data is received and the receive FIFO is already full. The FIFO contents remain valid because no more data is written when the FIFO is full, only the contents of the shift register are overwritten. This is cleared to 0 once there is an empty space in the FIFO and a new character can be written to it.
10	BE	R	0h	UART Break Error: This bit is set to 1 if a break condition was detected, indicating that the received data input (UARTRXD input pin) was held LOW for longer than a full-word transmission time (defined as start, data, parity and stop bits). In FIFO mode, this error is associated with the character at the top of the FIFO (i.e., the oldest received data character since last read). When a break occurs, a 0 character is loaded into the FIFO. The next character is enabled after the receive data input (UARTRXD input pin) goes to a 1 (marking state), and the next valid start bit is received.
9	PE	R	0h	UART Parity Error: When set to 1, it indicates that the parity of the received data character does not match the parity that the LCRH.EPS and LCRH.SPS select. In FIFO mode, this error is associated with the character at the top of the FIFO (i.e., the oldest received data character since last read).
8	FE	R	0h	UART Framing Error: When set to 1, it indicates that the received character did not have a valid stop bit (a valid stop bit is 1). In FIFO mode, this error is associated with the character at the top of the FIFO (i.e., the oldest received data character since last read).
7-0	DATA	R/W	0h	Data transmitted or received: On writes, the transmit data character is pushed into the FIFO. On reads, the oldest received data character since the last read is returned.

19.6.2 RSR_ECR Register (Offset = 4h) [Reset = 00000000h]

RSR_ECR is shown in Table 19-5.

Return to the Summary Table.

Status
This register is mapped to the same address as ECR register. Reads from this address are associated with RSR_ECR register and return the receive status. Writes to this address are associated with ECR register and clear the receive status flags (framing, parity, break, and overrun errors).
If the status is read from this register, then the status information for break, framing and parity corresponds to the data character read from the Data Register DR, prior to reading the RSR_ECR. The status information for overrun is set immediately when an overrun condition occurs.

Table 19-5. RSR_ECR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3	OE	R/W	0h	UART Overrun Error: This bit is set to 1 if data is received and the receive FIFO is already full. The FIFO contents remain valid because no more data is written when the FIFO is full, only the contents of the shift register are overwritten. This is cleared to 0 once there is an empty space in the FIFO and a new character can be written to it. 0h = Error flag is not set 1h = Clears error flag if error is set. Write value is not important.
2	BE	R/W	0h	UART Break Error: This bit is set to 1 if a break condition was detected, indicating that the received data input (UARTRXD input pin) was held LOW for longer than a full-word transmission time (defined as start, data, parity and stop bits). When a break occurs, a 0 character is loaded into the FIFO. The next character is enabled after the receive data input (UARTRXD input pin) goes to a 1 (marking state), and the next valid start bit is received. 0h = Error flag is not set 1h = Clears error flag if error is set. Write value is not important.
1	PE	R/W	0h	UART Parity Error: When set to 1, it indicates that the parity of the received data character does not match the parity that the LCRH.EPS and LCRH.SPS select. 0h = Error flag is not set 1h = Clears error flag if error is set. Write value is not important.
0	FE	R/W	0h	UART Framing Error: When set to 1, it indicates that the received character did not have a valid stop bit (a valid stop bit is 1). 0h = Error flag is not set 1h = Clears error flag if error is set. Write value is not important.

19.6.3 FR Register (Offset = 18h) [Reset = 00000091h]

FR is shown in Table 19-6.

Return to the Summary Table.

Flag
Reads from this register return the UART flags.

Table 19-6. FR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Reserved
7	TXFE	R	1h	UART Transmit FIFO Empty: The meaning of this bit depends on the state of LCRH.FEN . - If the FIFO is disabled, this bit is set when the transmit holding register is empty. - If the FIFO is enabled, this bit is set when the transmit FIFO is empty. This bit does not indicate if there is data in the transmit shift register.
6	RXFF	R	0h	UART Receive FIFO Full: The meaning of this bit depends on the state of LCRH.FEN. - If the FIFO is disabled, this bit is set when the receive holding register is full. - If the FIFO is enabled, this bit is set when the receive FIFO is full.
5	TXFF	R	0h	UART Transmit FIFO Full: Transmit FIFO full. The meaning of this bit depends on the state of LCRH.FEN. - If the FIFO is disabled, this bit is set when the transmit holding register is full. - If the FIFO is enabled, this bit is set when the transmit FIFO is full.
4	RXFE	R	1h	UART Receive FIFO Empty: Receive FIFO empty. The meaning of this bit depends on the state of LCRH.FEN. - If the FIFO is disabled, this bit is set when the receive holding register is empty. - If the FIFO is enabled, this bit is set when the receive FIFO is empty.
3	BUSY	R	0h	UART Busy: If this bit is set to 1, the UART is busy transmitting data. This bit remains set until the complete byte, including all the stop bits, has been sent from the shift register. This bit is set as soon as the transmit FIFO becomes non-empty, regardless of whether the UART is enabled or not.
2-1	RESERVED	R	0h	Reserved
0	CTS	R	1h	Clear To Send: This bit is the complement of the active-low UART CTS input pin. That is, the bit is 1 when CTS input pin is LOW.

19.6.4 UARTILPR Register (Offset = 20h) [Reset = 00000000h]

UARTILPR is shown in Table 19-7.

Return to the Summary Table.

IrDA Low-Power Counter Register.
This is an 8-bit register that stores the low-power counter divisor value used to generate the IrLPBaud16 signal by dividing down of UARTCLK.

Table 19-7. UARTILPR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R/W	0h	Reserved
7-0	ILPDVSR	R/W	0h	8 bit low-power divisor value. In low-power IrDA mode the UART rejects random noise on the received serial data input by ignoring SIRIN pulses that are less than 3 periods of IrLPBaud16.

19.6.5 IBRD Register (Offset = 24h) [Reset = 00000000h]

IBRD is shown in Table 19-8.

Return to the Summary Table.

Integer Baud-Rate Divisor
If this register is modified while trasmission or reception is on-going, the baudrate will not be updated until transmission or reception of the current character is complete.

Table 19-8. IBRD Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R/W	0h	Reserved
15-0	DIVINT	R/W	0h	The integer baud rate divisor: The baud rate divisor is calculated using the formula below: Baud rate divisor = (UART reference clock frequency) / (16 * Baud rate) Baud rate divisor must be minimum 1 and maximum 65535. That is, DIVINT=0 does not give a valid baud rate. Similarly, if DIVINT=0xFFFF, any non-zero values in FBRD.DIVFRAC will be illegal. Refer to Section 19.1.5 for an example calculation. A valid value must be written to this field before the UART can be used for RX or TX operations.

19.6.6 FBRD Register (Offset = 28h) [Reset = 00000000h]

FBRD is shown in Table 19-9.

Return to the Summary Table.

Fractional Baud-Rate Divisor
If this register is modified while trasmission or reception is on-going, the baudrate will not be updated until transmission or reception of the current character is complete.

Table 19-9. FBRD Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R/W	0h	Reserved
5-0	DIVFRAC	R/W	0h	Fractional Baud-Rate Divisor: The baud rate divisor is calculated using the formula below: Baud rate divisor = (UART reference clock frequency) / (16 * Baud rate) Baud rate divisor must be minimum 1 and maximum 65535. That is, IBRD.DIVINT=0 does not give a valid baud rate. Similarly, if IBRD.DIVINT=0xFFFF, any non-zero values in DIVFRAC will be illegal. Refer to Section 19.1.5 for an example calculation. A valid value must be written to this field before the UART can be used for RX or TX operations.

19.6.7 LCRH Register (Offset = 2Ch) [Reset = 00000000h]

LCRH is shown in Table 19-10.

Return to the Summary Table.

Line Control

Table 19-10. LCRH Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R/W	0h	Reserved
7	SPS	R/W	0h	UART Stick Parity Select: 0: Stick parity is disabled 1: The parity bit is transmitted and checked as invert of EPS field (i.e. the parity bit is transmitted and checked as 1 when EPS = 0). This bit has no effect when PEN disables parity checking and generation.
6-5	WLEN	R/W	0h	UART Word Length: These bits indicate the number of data bits transmitted or received in a frame. 0h = Word Length 5 bits 1h = Word Length 6 bits 2h = Word Length 7 bits 3h = Word Length 8 bits
4	FEN	R/W	0h	UART Enable FIFOs 0h = FIFOs are disabled (character mode) that is, the FIFOs become 1-byte-deep holding registers. 1h = Transmit and receive FIFO buffers are enabled (FIFO mode)
3	STP2	R/W	0h	UART Two Stop Bits Select: If this bit is set to 1, two stop bits are transmitted at the end of the frame. The receive logic does not check for two stop bits being received.
2	EPS	R/W	0h	UART Even Parity Select 0h = Odd parity: The UART generates or checks for an odd number of 1s in the data and parity bits. 1h = Even parity: The UART generates or checks for an even number of 1s in the data and parity bits.
1	PEN	R/W	0h	UART Parity Enable This bit controls generation and checking of parity bit. 0h = Parity is disabled and no parity bit is added to the data frame 1h = Parity checking and generation is enabled.
0	BRK	R/W	0h	UART Send Break If this bit is set to 1, a low-level is continually output on the UARTTXD output pin, after completing transmission of the current character. For the proper execution of the break command, the software must set this bit for at least two complete frames. For normal use, this bit must be cleared to 0.

19.6.8 CTL Register (Offset = 30h) [Reset = 00000300h]

CTL is shown in Table 19-11.

Return to the Summary Table.

Control

Table 19-11. CTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15	CTSEN	R/W	0h	CTS hardware flow control enable 0h = CTS hardware flow control disabled 1h = CTS hardware flow control enabled
14	RTSEN	R/W	0h	RTS hardware flow control enable 0h = RTS hardware flow control disabled 1h = RTS hardware flow control enabled
13-12	RESERVED	R/W	0h	Reserved
11	RTS	R/W	0h	Request to Send This bit is the complement of the active-low UART RTS output. That is, when the bit is programmed to a 1 then RTS output on the pins is LOW.
10	RESERVED	R/W	0h	Reserved
9	RXE	R/W	1h	UART Receive Enable If the UART is disabled in the middle of reception, it completes the current character before stopping. 0h = UART Receive disabled 1h = UART Receive enabled
8	TXE	R/W	1h	UART Transmit Enable If the UART is disabled in the middle of transmission, it completes the current character before stopping. 0h = UART Transmit disabled 1h = UART Transmit enabled
7	LBE	R/W	0h	UART Loop Back Enable Enabling the loop-back mode connects the UARTTXD output from the UART to UARTRXD input of the UART. 0h = Loop Back disabled 1h = Loop Back enabled
6	FCEN	R/W	0h	UART FIFO Concatenation Enable Enabling the FIFO concatenation in TX mode resulting in 16 TX buffers. 0h = UART FIFO Concatenation disabled 1h = UART FIFO Concatenation enabled
5-3	RESERVED	R/W	0h	Reserved
2	SIRLP	R/W	0h	SIR low power IrDA mode This bit selects the IrDA encoding mode 0h = Low-level bits are transmitted as active high with a 3/16th period width, 1h = Low-level bits are transmitted with a pulse width of 3 times the period of IrLPBaud16(which has a frequency of (UARTCLK frequency)/UARTILP.ILPDVSR), regardless of the selected bit rate. In low-power IrDA mode the UART rejects random noise on the received serial data input by ignoring SIRIN pulses that are less than 3 periods of IrLPBaud16.
1	SIREN	R/W	0h	SIR Enable This bit has no effect if UARTEN bit disables the UART. 0h = IrDA SIR ENDEC is disabled 1h = IrDA SIR ENDEC is enabled. Data is transmitted and received via nSIROUT and SIRIN.

Table 19-11. CTL Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
0	UARTEN	R/W	0h	UART Enable 0h = UART disabled 1h = UART enabled

19.6.9 IFLS Register (Offset = 34h) [Reset = 00000012h]

IFLS is shown in Table 19-12.

Return to the Summary Table.

Interrupt FIFO Level Select

Table 19-12. IFLS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R/W	0h	Reserved
5-3	RXSEL	R/W	2h	Receive interrupt FIFO level select: This field sets the trigger points for the receive interrupt. Values 0b101-0b111 are reserved. 1h = Receive FIFO becomes >= 1/4 full 2h = Receive FIFO becomes >= 1/2 full 3h = Receive FIFO becomes >= 3/4 full
2-0	TXSEL	R/W	2h	Transmit interrupt FIFO level select: This field sets the trigger points for the transmit interrupt. Values 0b101-0b111 are reserved. 1h = Transmit FIFO becomes <= 1/4 full 2h = Transmit FIFO becomes <= 1/2 full 3h = Transmit FIFO becomes <= 3/4 full

19.6.10 IMSC Register (Offset = 38h) [Reset = 00000000h]

IMSC is shown in Table 19-13.

Return to the Summary Table.

Interrupt Mask Set/Clear

Table 19-13. IMSC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-14	RESERVED	R/W	0h	Reserved
13	RXDMADONE	R/W	0h	RX DMA done interrupt mask. A read returns the current mask for UART's RXDMADONE interrupt. On a write of 1, the mask of the RXDMADONE interrupt is set which means the interrupt state will be reflected in MIS.RXDMADONE. A write of 0 clears the mask which means MIS.RXDMADONE will not reflect the interrupt.
12	TXDMADONE	R/W	0h	TX DMA done interrupt mask. A read returns the current mask for UART's TXDMADONE interrupt. On a write of 1, the mask of the TXDMADONE interrupt is set which means the interrupt state will be reflected in MIS.TXDMADONE. A write of 0 clears the mask which means MIS.TXDMADONE will not reflect the interrupt.
11	EOT	R/W	0h	End of Transmission interrupt mask. A read returns the current mask for UART's EoT interrupt. On a write of 1, the mask of the EoT interrupt is set which means the interrupt state will be reflected in MIS.EOT. A write of 0 clears the mask which means MIS.EOT will not reflect the interrupt.
10	OE	R/W	0h	Overrun error interrupt mask. A read returns the current mask for UART's overrun error interrupt. On a write of 1, the mask of the overrun error interrupt is set which means the interrupt state will be reflected in MIS.OE. A write of 0 clears the mask which means MIS.OE will not reflect the interrupt.
9	BE	R/W	0h	Break error interrupt mask. A read returns the current mask for UART's break error interrupt. On a write of 1, the mask of the overrun error interrupt is set which means the interrupt state will be reflected in MIS.BE. A write of 0 clears the mask which means MIS.BE will not reflect the interrupt.
8	PE	R/W	0h	Parity error interrupt mask. A read returns the current mask for UART's parity error interrupt. On a write of 1, the mask of the overrun error interrupt is set which means the interrupt state will be reflected in MIS.PE. A write of 0 clears the mask which means MIS.PE will not reflect the interrupt.
7	FE	R/W	0h	Framing error interrupt mask. A read returns the current mask for UART's framing error interrupt. On a write of 1, the mask of the overrun error interrupt is set which means the interrupt state will be reflected in MIS.FE. A write of 0 clears the mask which means MIS.FE will not reflect the interrupt.
6	RT	R/W	0h	Receive timeout interrupt mask. A read returns the current mask for UART's receive timeout interrupt. On a write of 1, the mask of the overrun error interrupt is set which means the interrupt state will be reflected in MIS.RT. A write of 0 clears the mask which means this bitfield will not reflect the interrupt. The raw interrupt for receive timeout RIS.RT cannot be set unless the mask is set (RT = 1). This is because the mask acts as an enable for power saving. That is, the same status can be read from MIS.RT and RIS.RT.
5	TX	R/W	0h	Transmit interrupt mask. A read returns the current mask for UART's transmit interrupt. On a write of 1, the mask of the overrun error interrupt is set which means the interrupt state will be reflected in MIS.TX. A write of 0 clears the mask which means MIS.TX will not reflect the interrupt.

Table 19-13. IMSC Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
4	RX	R/W	0h	Receive interrupt mask. A read returns the current mask for UART's receive interrupt. On a write of 1, the mask of the overrun error interrupt is set which means the interrupt state will be reflected in MIS.RX. A write of 0 clears the mask which means MIS.RX will not reflect the interrupt.
3-2	RESERVED	R/W	0h	Reserved
1	CTSM	R/W	0h	Clear to Send (CTS) modem interrupt mask. A read returns the current mask for UART's clear to send interrupt. On a write of 1, the mask of the overrun error interrupt is set which means the interrupt state will be reflected in MIS.CTSM. A write of 0 clears the mask which means MIS.CTSM will not reflect the interrupt.
0	RESERVED	R/W	0h	Reserved

19.6.11 RIS Register (Offset = 3Ch) [Reset = 0000000Dh]

RIS is shown in Table 19-14.

Return to the Summary Table.

Raw Interrupt Status

Table 19-14. RIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-14	RESERVED	R	0h	Reserved
13	RXDMADONE	R	0h	RX DMA done interrupt status: This field returns the raw interrupt state of UART's RX DMA done interrupt. RX DMA done flag is set when you recieve RX DMA done status from dma module.
12	TXDMADONE	R	0h	TX DMA done interrupt status: This field returns the raw interrupt state of UART's TX DMA done interrupt. TX DMA done flag is set when you recieve TX DMA done status from dma module.
11	EOT	R	0h	End of Transmission interrupt status: This field returns the raw interrupt state of UART's end of transmission interrupt. End of transmission flag is set when all the Transmit data in the FIFO and on the TX Line is tranmitted.
10	OE	R	0h	Overrun error interrupt status: This field returns the raw interrupt state of UART's overrun error interrupt. Overrun error occurs if data is received and the receive FIFO is full.
9	BE	R	0h	Break error interrupt status: This field returns the raw interrupt state of UART's break error interrupt. Break error is set when a break condition is detected, indicating that the received data input (UARTRXD input pin) was held LOW for longer than a full-word transmission time (defined as start, data, parity and stop bits).
8	PE	R	0h	Parity error interrupt status: This field returns the raw interrupt state of UART's parity error interrupt. Parity error is set if the parity of the received data character does not match the parity that the LCRH.EPS and LCRH.SPS select.
7	FE	R	0h	Framing error interrupt status: This field returns the raw interrupt state of UART's framing error interrupt. Framing error is set if the received character does not have a valid stop bit (a valid stop bit is 1).
6	RT	R	0h	Receive timeout interrupt status: This field returns the raw interrupt state of UART's receive timeout interrupt. The receive timeout interrupt is asserted when the receive FIFO is not empty, and no more data is received during a 32-bit period. The receive timeout interrupt is cleared either when the FIFO becomes empty through reading all the data, or when a 1 is written to ICR.RT. The raw interrupt for receive timeout cannot be set unless the mask is set (IMSC.RT = 1). This is because the mask acts as an enable for power saving. That is, the same status can be read from MIS.RT and RT.

Table 19-14. RIS Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
5	TX	R	0h	Transmit interrupt status: This field returns the raw interrupt state of UART's transmit interrupt. When FIFOs are enabled (LCRH.FEN = 1), the transmit interrupt is asserted if the number of bytes in transmit FIFO is equal to or lower than the programmed trigger level (IFLS.TXSEL). The transmit interrupt is cleared by writing data to the transmit FIFO until it becomes greater than the trigger level, or by clearing the interrupt through ICR.TX. When FIFOs are disabled (LCRH.FEN = 0), that is they have a depth of one location, the transmit interrupt is asserted if there is no data present in the transmitters single location. It is cleared by performing a single write to the transmit FIFO, or by clearing the interrupt through ICR.TX.
4	RX	R	0h	Receive interrupt status: This field returns the raw interrupt state of UART's receive interrupt. When FIFOs are enabled (LCRH.FEN = 1), the receive interrupt is asserted if the receive FIFO reaches the programmed trigger level (IFLS.RXSEL). The receive interrupt is cleared by reading data from the receive FIFO until it becomes less than the trigger level, or by clearing the interrupt through ICR.RX. When FIFOs are disabled (LCRH.FEN = 0), that is they have a depth of one location, the receive interrupt is asserted if data is received thereby filling the location. The receive interrupt is cleared by performing a single read of the receive FIFO, or by clearing the interrupt through ICR.RX.
3-2	RESERVED	R	3h	Reserved
1	CTSM	R	0h	Clear to Send (CTS) modem interrupt status: This field returns the raw interrupt state of UART's clear to send interrupt.
0	RESERVED	R	1h	Reserved

19.6.12 MIS Register (Offset = 40h) [Reset = 00000000h]

MIS is shown in Table 19-15.

Return to the Summary Table.

Masked Interrupt Status

Table 19-15. MIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-14	RESERVED	R	0h	Reserved
13	RXDMADONE	R	0h	RX DMA done interrupt status: This field returns the masked interrupt state of the RX DMA done interrupt which is the bitiwse AND product of raw interrupt state RIS.RXDMADONE and the mask setting IMSC.RXDMADONE.
12	TXDMADONE	R	0h	TX DMA done interrupt status: This field returns the masked interrupt state of the TX DMA done interrupt which is the bitwise AND product of raw interrupt state RIS.TXDMADONE and the mask setting IMSC.TXDMADONE.
11	EOT	R	0h	End of Transmission interrupt status: This field returns the masked interrupt state of the End of transmission interrupt which is the bitwise AND product of raw interrupt state RIS.EOT and the mask setting IMSC.EOT.
10	OE	R	0h	Overrun error masked interrupt status: This field returns the masked interrupt state of the overrun interrupt which is the bitwise AND product of raw interrupt state RIS.OE and the mask setting IMSC.OE.
9	BE	R	0h	Break error masked interrupt status: This field returns the masked interrupt state of the break error interrupt which is the bitiwse AND product of raw interrupt state RIS.BE and the mask setting IMSC.BE.
8	PE	R	0h	Parity error masked interrupt status: This field returns the masked interrupt state of the parity error interrupt which is the bitiwise AND product of raw interrupt state RIS.PE and the mask setting IMSC.PE.
7	FE	R	0h	Framing error masked interrupt status: Returns the masked interrupt state of the framing error interrupt which is the bitiwse AND product of raw interrupt state RIS.FE and the mask setting IMSC.FE.
6	RT	R	0h	Receive timeout masked interrupt status: Returns the masked interrupt state of the receive timeout interrupt. The raw interrupt for receive timeout cannot be set unless the mask is set (IMSC.RT = 1). This is because the mask acts as an enable for power saving. That is, the same status can be read from MIS.RT and RIS.RT.
5	TX	R	0h	Transmit masked interrupt status: This field returns the masked interrupt state of the transmit interrupt which is the bitiwse AND product of raw interrupt state RIS.TX and the mask setting IMSC.TX.
4	RX	R	0h	Receive masked interrupt status: This field returns the masked interrupt state of the receive interrupt which is the bitwise AND product of raw interrupt state RIS.RX and the mask setting IMSC.RX.
3-2	RESERVED	R	0h	Reserved
1	CTSM	R	0h	Clear to Send (CTS) modem masked interrupt status: This field returns the masked interrupt state of the clear to send interrupt which is the bitwise AND product of raw interrupt state RIS.CTS and the mask setting IMSC.CTS.
0	RESERVED	R	0h	Reserved

19.6.13 ICR Register (Offset = 44h) [Reset = 00000000h]

ICR is shown in Table 19-16.

Return to the Summary Table.

Interrupt Clear
On a write of 1, the corresponding interrupt is cleared. A write of 0 has no effect.

Table 19-16. ICR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-14	RESERVED	W	0h	Reserved
13	RXDMADONE	W	0h	RX DMA Done interrupt clear: Writing 1 to this field clears the RX DMA done interrupt (RIS.RXDMADONE). Writing 0 has no effect.
12	TXDMADONE	W	0h	TX DMA Done interrupt clear: Writing 1 to this field clears the TX DMA done interrupt (RIS.TXDMADONE). Writing 0 has no effect.
11	EOT	W	0h	End of Transmission interrupt clear: Writing 1 to this field clears the End of Transmission interrupt (RIS.EOT). Writing 0 has no effect.
10	OE	W	0h	Overrun error interrupt clear: Writing 1 to this field clears the overrun error interrupt (RIS.OE). Writing 0 has no effect.
9	BE	W	0h	Break error interrupt clear: Writing 1 to this field clears the break error interrupt (RIS.BE). Writing 0 has no effect.
8	PE	W	0h	Parity error interrupt clear: Writing 1 to this field clears the parity error interrupt (RIS.PE). Writing 0 has no effect.
7	FE	W	0h	Framing error interrupt clear: Writing 1 to this field clears the framing error interrupt (RIS.FE). Writing 0 has no effect.
6	RT	W	0h	Receive timeout interrupt clear: Writing 1 to this field clears the receive timeout interrupt (RIS.RT). Writing 0 has no effect.
5	TX	W	0h	Transmit interrupt clear: Writing 1 to this field clears the transmit interrupt (RIS.TX). Writing 0 has no effect.
4	RX	W	0h	Receive interrupt clear: Writing 1 to this field clears the receive interrupt (RIS.RX). Writing 0 has no effect.
3-2	RESERVED	W	0h	Reads to this field return zero, writes to this field are ignored. Write 0
1	CTSM	W	0h	Clear to Send (CTS) modem interrupt clear: Writing 1 to this field clears the clear to send interrupt (RIS.CTS). Writing 0 has no effect.
0	RESERVED	W	0h	Reads to this field return zero, writes to this field are ignored. Write 0.

19.6.14 DMACTL Register (Offset = 48h) [Reset = 00000000h]

DMACTL is shown in Table 19-17.

Return to the Summary Table.

DMA Control

Table 19-17. DMACTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-3	RESERVED	R/W	0h	Reserved
2	DMAONERR	R/W	0h	DMA on error. If this bit is set to 1, the DMA receive request outputs (for single and burst requests) are disabled when the UART error interrupt is asserted (more specifically if any of the error interrupts RIS.PERIS, RIS.BERIS, RIS.FERIS or RIS.OERIS are asserted).
1	TXDMAE	R/W	0h	Transmit DMA enable. If this bit is set to 1, DMA for the transmit FIFO is enabled.
0	RXDMAE	R/W	0h	Receive DMA enable. If this bit is set to 1, DMA for the receive FIFO is enabled.

Chapter 20

Serial Peripheral Interface (SPI)

This chapter describes the Serial Peripheral Interface (SPI) module.

20.1 Overview

The Serial Peripheral Interface (SPI) module provides a standardized serial interface to transfer data to and from external devices using SPI protocol (such as sensors, memories, ADCs, or DACs).

20.1.1 Features

The SPI module has the following features:

- Programmable interface operation for Motorola SPI (3-wire and 4-wire), MICROWIRE, or TI Synchronous Serial format
- Configurable as a controller or a peripheral on the interface
- Programmable clock bit rate and prescaler
- CRC8-CCITT or CRC16-CCITT CRC capability
- Separate transmit (TX) and receive (RX) first-in first-out buffers (FIFOs)
  - If Data Size Select (DSS) is 4 to 8 bits, FIFOs are 16 locations, 8 bits wide
  - If Data Size Select (DSS) is 9 to 16 bits, FIFOs are 8 locations, 16 bits wide
- Programmable data frame size from 4 bits to 16 bits (controller mode) or 7 to 16 bits (peripheral mode)
- Internal loop-back test mode for diagnostic and debug testing
- Interrupts for transmit and receive FIFOs, overrun and time-out interrupts, and DMA-done interrupts
- Efficient transfers using micro direct memory access controller (µDMA):
  - Separate channels for transmit and receive
  - Receive single request asserted when data is in the FIFO; burst request asserted when FIFO contains a configurable number of entries.
  - Transmit single request asserted when there is space in the FIFO; burst request asserted when FIFO contains a configurable number of entries.

20.1.2 Block Diagram

Figure 20-1 shows the SPI block diagram.

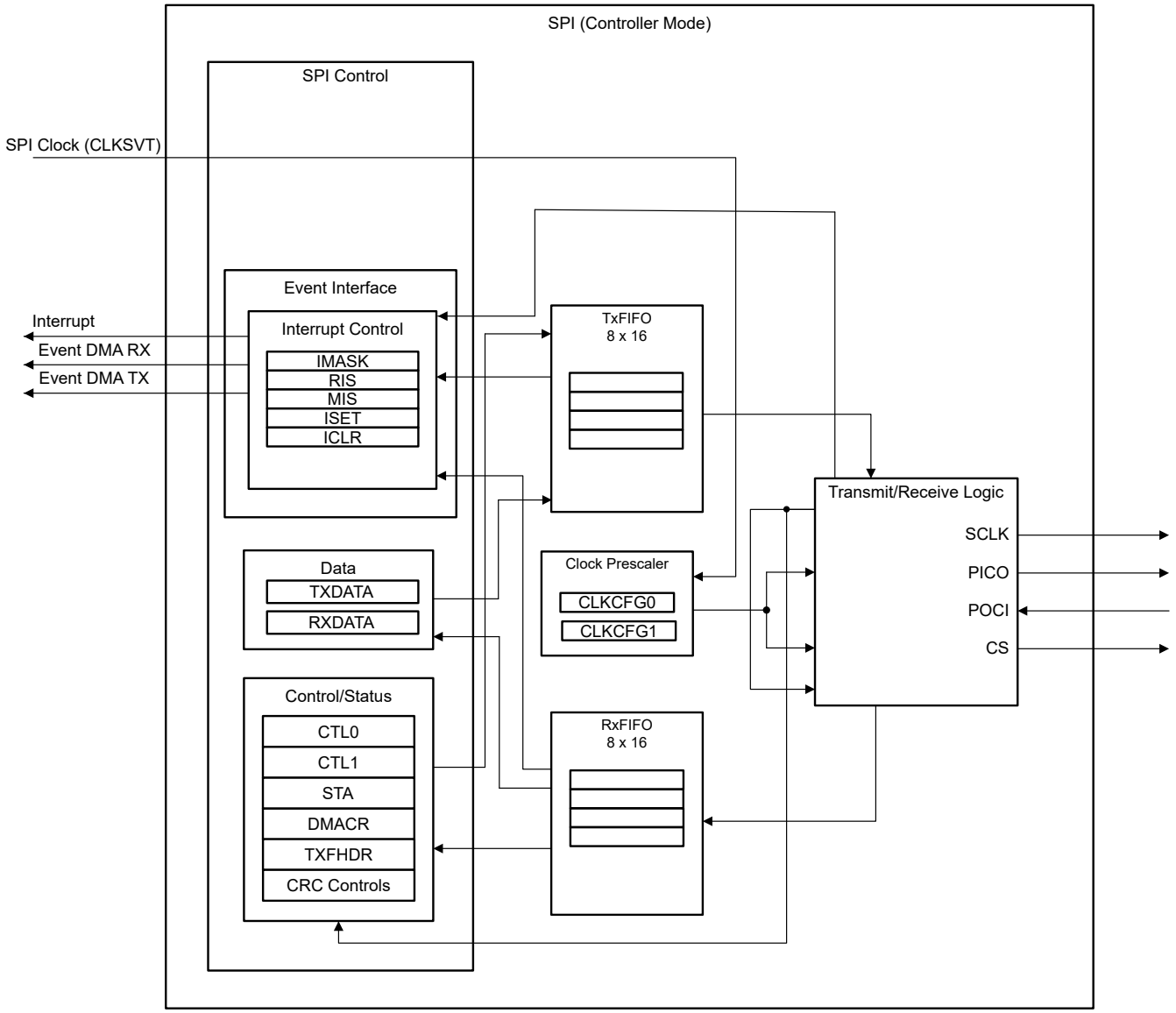


Figure 20-1. SPI Block Diagram

20.2 Signal Description

Table 20-1 lists the external signals of SPI and describes the function of each signal. The SPI signals are selected in the IOC module through the IOCFGn registers. For more information on the configuration of DIOs, see Chapter 18

Table 20-1. SPI Signal Description

Signal Name	Pin Number	Description
SCLK	Assigned in I/O Controller	**SPI Clock** Controller Mode: SCLK is an output. Peripheral Mode: SCLK is an input.
CS		**SPI Chip Select** Controller Mode: CS is an output. Peripheral Mode: CS is an input.
PICO		**Peripheral In, Controller Out** Controller mode: PICO is the data output line. Peripheral mode: PICO is the data input line.
POCI		**Peripheral Out, Controller In** Controller mode: POCI is the data input line. Peripheral mode: POCI is the data output line.

20.3 Functional Description

SPI performs serial-to-parallel conversion on data received from a peripheral device. The CPU accesses data, control, and status information. Internal FIFO memories buffer the transmit and receive paths, allowing independent storage of up to eight 16-bit values in both transmit and receive modes. The SPI also supports the µDMA interface. The TX and RX FIFOs can be programmed as destination or source addresses in the µDMA module. The µDMA operation is enabled by setting the appropriate bits in the SPI:DMACR register. The SPI module also includes a CRC engine that can be used for data checking during SPI transmission. If SPI is not being used, this can act as a general-purpose CRC engine. Additional capability is added to enable the SPI module to be used efficiently with an external controller in a transceiver setup by allowing atomic operations to update header information in the FIFO, including atomic FIFO pointers reset capability.

20.3.1 Clock Control

The SPI includes a programmable bit rate clock divider and prescaler to generate the serial output clock.

The serial bit rate is derived by dividing down the input clock, CLKSVT (48MHz).

First, the clock is divided by a prescaler with a value from 1 to 8, which is programmed in the SPI.CLKCFG0[2:0] PRESC field (a value of 0x1 means that the clock is divided by 2). The clock is further divided by a value from 2 to 2048, which is 2×(1 + SCR), where SCR is the value programmed in the SPI.CLKCFG1[9:0] SCR field.

Equation 9 defines the frequency of the output clock SCLK.

$$SCLK = \frac{CLKSVT}{(1 + PRESC) \times (2 \times (1 + SCR))} \quad (9)$$

Note

For both peripheral and controller modes, the core clock (CLKSVT) must be at least two times faster than the SCLK.

The maximum SPI frequency supported with controller and peripheral modes depends on the device clock option and IO option. Please refer to specific device data sheet specification for more information.

20.3.2 FIFO Operation

20.3.2.1 Transmit FIFO

The common TX FIFO is a 16-bit wide, eight locations deep, first-in first-out memory buffer if the selected SPI data frame size is greater than 8 bits. The organization of this FIFO is modified dynamically if the selected data size is less than or equal to 8 bits, and for better FIFO utilization behaves as an 8-bit wide, 16 locations deep

FIFO. The CPU writes data to the FIFO through the SPI.TXDATA register and data is stored in the FIFO until the data is read out by the transmission logic.

When configured as a controller (or a peripheral), parallel data is written into the TX FIFO before serial conversion and transmission to the attached peripheral (or controller) through the PICO (or POCI) pin.

In peripheral mode, the SPI transmits data each time the controller initiates a transaction. If the TX FIFO is empty and the controller initiates a transaction, the peripheral transmits garbage data. The user or software engineer is responsible for making valid data available in the FIFO as needed. The SPI can be configured to generate an interrupt when a configurable level within the FIFO is selected by SPI:IFLS, or a µDMA request when the FIFO is not FULL.

20.3.2.2 Repeated Transmit Operation

Using the SPI.CTL1[23:16] REPTX bit field, the last data frame transmitted can be repeated as many times as configured within the field. The SPI transfer can be started by writing a data once into the TX FIFO. This feature then transmits the same data repeatedly as if the data were written into the TXFIFO [*SPI.CTL1[23:16] REPTX bit field*] number of times. The repeated transfer operation can be used to clean a transfer or to pull a certain amount of data from a peripheral. A value of 0 in the SPI.CTL1[23:16] REPTX bit field disables this mode. This function is only available in controller mode.

When the repeated transmit operation is used, the function needs to be aligned with the data in the FIFO. The following sequence is used when setting up the repeated transmit operation:

- Check and wait till FIFO is empty.
- Setup REPTX.
- Write to TXDATA / TXFIFO.
- Wait till the requested data is received.

20.3.2.3 Receive FIFO

The common RX FIFO is a 16-bit wide, eight locations deep, first-in-first-out memory buffer if the selected SPI data frame size is greater than 8 bits. The organization of this FIFO is modified dynamically if the selected data size is less than or equal to 8 bits, and for better FIFO utilization behaves as an 8-bit wide, 16 locations deep FIFO. Received data from the serial interface is stored in the buffer until read out by the CPU, which accesses the read FIFO by reading the SPI.RXDATA register.

When configured as a controller (or peripheral), serial data received through the POCI (or PICO) pin is registered before parallel loading into the RX FIFO.

20.3.2.4 FIFO Flush

SPI includes a feature to reset the TX and RX FIFO pointers to flush FIFOs. This must be triggered when no SPI transactions are in progress. If a FIFO flush is triggered when a transaction is in progress, then a second FIFO flush is needed when no operations are ongoing, before restarting new SPI transfers.

The FIFO flush operation is atomic. When the CPU writes into SPI.CTL0[11] FIFORST register bit, the SPI hardware internally ensures that TX and RX FIFO pointers are set to zero, and auto-clears the FIFORST bit after 4 CLKSVT clock cycles. CPU can poll the FIFORST bit to identify when the FIFO pointer reset operation is completed.

Note

FIFO pointers also get reset when SPI is disabled after the SPI.CTL1[0] EN bit toggles from 0x1 to 0x0.

20.3.3 Interrupts

The SPI can generate interrupts when the following conditions are observed:

- TX FIFO service (with the TX FIFO level configured through the SPI.IFLS[2:0] TXSEL bit field)
- RX FIFO service (with the RX FIFO level configured through the SPI.IFLS[10:8] RXSEL bit field)

- RX FIFO timeout
- RX FIFO overrun
- TX FIFO empty
- TX DMA done/RX DMA done
- Idle
- Parity error

All interrupt events are ORed together before being sent to the SVT event fabric, so the SPI generates a single interrupt request regardless of the number of active interrupts. The interrupt conditions listed above can be masked by setting the appropriate bit in the SPI.IMASK register. Setting the appropriate mask bit in the SPI.IMASK register enables the interrupt. SPI.IMSET and SPI.IMCLR are alias registers, which can be used to set and clear individual bits of SPI.IMASK register.

The status of the individual interrupt sources can be read from the SPI Raw Interrupt Status register (SPI.RIS) and the SPI Masked Interrupt Status register (SPI.MIS). SPI.ICLR can be used to clear interrupt flags within RIS and MIS. SPI.ISET can be used to set these interrupt flags for debug or test purposes.

The transmit FIFO service interrupt request SPI.RIS[4] TX bit is not gated with the SPI enable signal, which allows data to be written to the transmit FIFO before enabling the SPI by an interrupt service routine (ISR).

Note

TX and RX FIFO interrupts are best serviced by µDMA rather than CPU. In case CPU services TX and RX FIFO interrupts, TXEMPTY and RXOVF can be configured as well, so that if the FIFO interrupt is missed by the CPU in corner cases, these act as a failsafe.

The receive FIFO overflow interrupt SPI.RIS[0] RXOVF is asserted when the FIFO is already full and an additional data frame is received, causing an overrun of the FIFO. Data is overwritten in the receive shift register, but not in the FIFO.

The parity error interrupt SPI.RIS[1] PER bit is set when a parity error is detected. SPI.CTL1[5] PEN bit can be written to enable the parity check, where the last bit received is used as parity to test the integrity of the previous bits. SPI.CTL1[7] PBS bit selects the parity mode as even or odd. When a parity fault is detected, the interrupt flag SPI.RIS[1] PER bit is set (to mark the data as invalid).

The idle interrupt SPI.RIS[6] IDLE is set when the SPI transmission has concluded and the SPI module moves back to idle mode. This is set when SPI.STA[4] BUSY goes low.

The SPI Receive Timeout interrupt is set when SPI is in peripheral mode and has not been receiving data for the number of functional clock cycles (CLKSVT) configured within SPI.CTL1[29:24] RTOUT bit field. A value of 0 disables this function. The countdown is started when SPI is in the peripheral mode and the first SCLK positive edge is detected and the countdown is restarted on each subsequent SCLK positive edge. A timeout error is asserted if the count reaches zero before the next SCLK toggles.

20.3.4 Data Format

Each data frame is between 4 and 16 bits long, depending on the size of the data programmed. The control bit SPI.CTL1[4] MSB field can be programmed to define the direction of the data input and output as most-significant-bit (MSB) or least-significant-bit (LSB) first. If parity is enabled, the parity bit is always received as the last bit.

With SPI.CTL0[3:0] DSS bit field, the bit length per transfer is defined between 4–16 bits for controller mode and 7–16 bits for peripheral mode.

20.3.5 Delayed Data Sampling

In cases when the input data arrives at the POCI pin with additional delay due to run-time conditions or path delays, on the following input data sampling stage, the previous data can be sampled at the sampling clock edge. To compensate for this, sampling of input data in controller mode can be delayed using the SPI.CLKCFG1[19:16] DSAMPLE bit field. The delayed sampling is only available in controller mode. The delay can be adjusted in steps of undivided SPI input clocks (CLKSVT) programmed within the SPI.CLKCFG1[19:16]

DSAMPLE bit field. The range of values of DSAMPLE is 0 to SCR+1. Typically, values of 1 and 2 are sufficient even for the highest supported SPI frequencies.

20.3.6 Chip Select Control

SPI can be configured to controller mode by setting the SPI.CTL1[2] MS bit to 1, and to the peripheral mode by clearing the SPI.CTL1[2] MS bit.

The chip select signal needs to be provided by the controller in Motorola 4-wire mode.

SPI includes a feature to keep the CS active low until all data has been transferred from TXFIFO, regardless of the configuration of PHA or POL, in controller mode and Motorola 4-wire frame format. This feature is enabled by the SPI.CTL0[10] HWCSN bit. If SPI.CTL0[14] AUTOCRC is set, and then CS is kept low until the CRC has been transferred as well.

In peripheral mode, the clock is provided by the controller and used by the SPI to capture the data. The peripheral has the option to operate in 3-wire or 4-wire mode. The 4-wire mode only accepts data transfers if the CS is activated.

When SPI is in peripheral mode and the SPI.CTL0[12] CSCLR bit is set, the receive shift register is cleared automatically when CS goes to the inactive state.

20.3.7 Command Data Control

When using the Motorola 3-wire frame format with the SPI.CTL0[3:0] DSS bit field programmed for 8 bits, the SPI.CTL1[11] CDEN bit can be set to use the CS line as a signal to distinguish between Command and Data information. This is often used for LCD or data storage devices.

- CS level low: command function
- CS level high: data function

The SPI.CTL1[15:12] CDMODE bit field can be written with a value of 1-14 to specify the number of bytes the CS line is set low for, starting with the next value to be transmitted. After the number of bytes are transmitted, CS is set high automatically. If a value of 0xF is set, CS stays low permanently. A value of 0 sets the CS line to high immediately after the current data byte has been transmitted.

This option is only available in controller mode. The SPI.CTL1[11] CDEN bit can only be updated when the SPI module is disabled. SPI.CTL1[15:12] CDMODE can be updated between the different data packages. The counter is reset with CDEN or SPI ENABLE set to disabled. Before setting a new value in CTL1.CDMODE, check that the FIFO is empty and that SPI is in idle mode (SPI.STA[4] BUSY bit is cleared to 0).

When writing a new value into the SPI.CTL1[15:12] CDMODE bit field, the internal counter is reset and the new value is used for counting. If the counter counts down to 0 and another command package is sent, the CDMODE needs to be set again. Otherwise, the next frames are sent as data with the CS pin signaling data mode.

20.3.8 Protocol Descriptions

The protocol format mode can be selected by using the SPI.CTL0[6:5] FRF bit field. The supported options include Motorola 3-wire, Motorola 4-wire, Texas Instruments Synchronous and MICROWIRE.

20.3.8.1 Motorola SPI Frame Format

The Motorola SPI is a 4-wire interface where the CS signal behaves as a peripheral select. In the 3-wire mode, the CS signal is not required and the module behaves as if always selected. The main feature of the Motorola SPI format is that the inactive state and phase of the SCLK signal can be programmed through the SPO and SPH bits in the SPI.CTL0 control register.

SPO Clock Polarity Bit

If the SPI.CTL0[8] SPO clock polarity control bit is cleared, the SCLK pin outputs a steady-state low value when data is not being transferred. If the SPI.CTL0[8] SPO bit is set, the SCLK pin outputs a steady-state high value when data is not being transferred.

SPH Phase-Control Bit

The SPI.CTL0[9] SPH phase-control bit selects the clock edge that captures data. The state of this bit has the most impact on the first bit transmitted, by either allowing or not allowing a clock transition before the first data capture edge. If the SPI.CTL0[9] SPH phase-control bit is cleared, data is captured on the first clock edge transition. If the SPH bit is set, data is captured on the second clock edge transition.

Note

For all combinations of SPO and SPH, the minimum CS inactive period (where CS is held high) must be at least one SCLK period wide.

Motorola SPI Frame Format with SPO = 0 and SPH = 0

Figure 20-2 shows signal sequences for Motorola SPI format with SPO = 0 and SPH = 0.

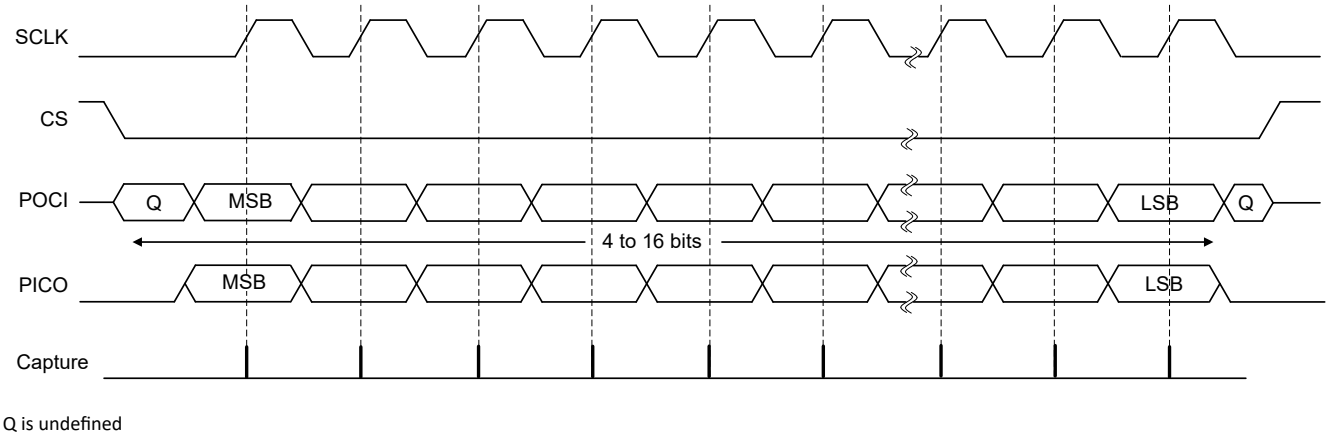


Figure 20-2. Motorola SPI Frame Format with SPO=0 and SPH=0

In this configuration, the following occurs during idle periods:

- SCLK is forced low.
- CS is forced high.
- The transmit data line PICO is forced low.
- When the SPI is configured as a controller, the SCLK pin is enabled.
- When the SPI is configured as a peripheral, the SCLK pin is disabled.

If the SPI is enabled and valid data is in the TX FIFO, the CS controller signal is driven low at the start of transmission which causes enabling of peripheral data onto the POCI input line of the controller. The controller PICO output pin is enabled.

One-half SCLK period later, valid controller data is transferred to the PICO pin. Once both the controller and peripheral data is set, the SCLK controller clock pin goes high after an additional one-half SCLK period. The data is now captured on the rising edges and propagated on the falling edges of the SCLK signal.

For a single-word transmission after all bits of the data frame are transferred, the CS line is returned to the IDLE high state one SCLK period after the last bit is captured. For continuous back-to-back transmissions, the CS signal must pulse high between each data word transfer because the peripheral-select pin freezes the data in the serial peripheral register and does not allow altering of the data if the SPH bit is clear. The controller device must raise the CS pin of the peripheral device between each data transfer to enable the serial peripheral data write. SPI.CTL0[10] HWCSN described in Section 20.3.6 can be used to override this behavior. When the continuous transfer completes, the CS pin is returned to the IDLE state one SCLK period after the last bit is captured.

Motorola SPI Frame Format with SPO = 0 and SPH = 1

Figure 20-3 shows the signal sequence for Motorola SPI format with SPO = 0 and SPH = 1.

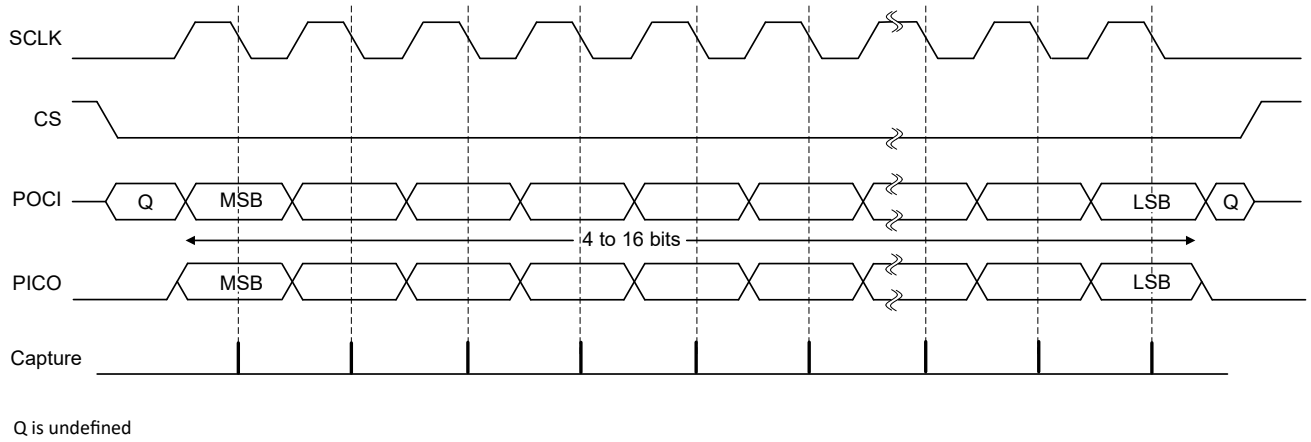


Figure 20-3. Motorola SPI Frame Format with SPO=0 and SPH=1

If the SPI is enabled and valid data is in the TX FIFO, the CS controller signal goes low at the start of transmission. The controller PICO output is enabled. After an additional one-half SCLK period, both controller and peripheral valid data are enabled onto their respective transmission lines. At the same time, SCLK is enabled with a rising-edge transition. Data is then captured on the falling edges and propagated on the rising edges of the SCLK signal.

For a single-word transfer, after all bits are transferred, the CS line is returned to the IDLE high state one SCLK period after the last bit is captured. For continuous back-to-back transfers, the CS pin is held low between successive data words and terminates like a single-word transfer.

Motorola SPI Frame Format with SPO = 1 and SPH = 0

Figure 20-4 shows signal sequences for Motorola SPI format with SPO = 1 and SPH = 0.

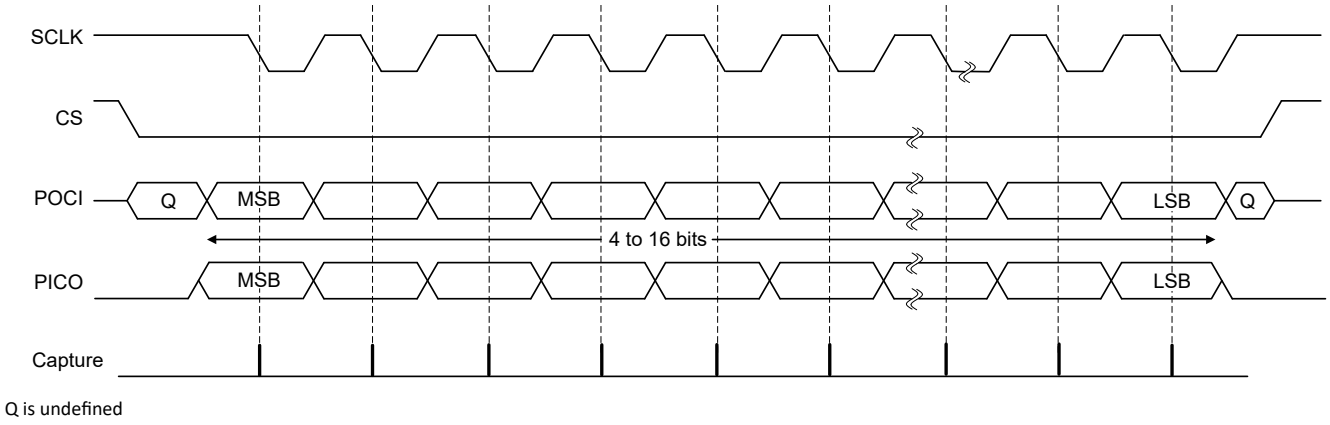


Figure 20-4. Motorola SPI Frame Format with SPO=1 and SPH=0

In this configuration, the following occurs during idle periods:

- SCLK is forced high.
- CS is forced high.
- The transmit data line PICO is forced low.
- When the SPI is configured as a controller, the SCLK pin is enabled.
- When the SPI is configured as a peripheral, the SCLK pin is disabled.

If the SPI is enabled and valid data is in the TX FIFO, the SPI CS controller signal goes low at the start of transmission and transfers peripheral data onto the POCI line of the controller immediately. The controller PICO output pin is enabled.

One-half SCLK period later, valid controller data is transferred to the PICO line. When both the controller and peripheral data have been set, the SCLK controller clock pin becomes low after one additional half SCLK period. Data is captured on the falling edges and propagated on the rising edges of the SCLK signal.

For a single-word transmission after all bits of the data word are transferred, the CS line is returned to the IDLE high state one SCLK period after the last bit is captured. For continuous back-to-back transmissions, the CS signal must pulse high between each data word transfer as the peripheral-select pin freezes the data in the serial peripheral register and keeps it from being altered if the SPH bit is clear. The controller device must raise the CS pin of the peripheral device between each data transfer to enable the serial peripheral data write. SPI.CTL0[10] HWCSN described in Section 20.3.6 can be used to override this behavior. When the continuous transfer completes, the CS pin returns to its IDLE state one SCLK period after the last bit is captured.

Motorola SPI Frame Format with SPO = 1 and SPH = 1

Figure 20-5 shows the signal sequence for Motorola SPI format with SPO = 1 and SPH = 1.

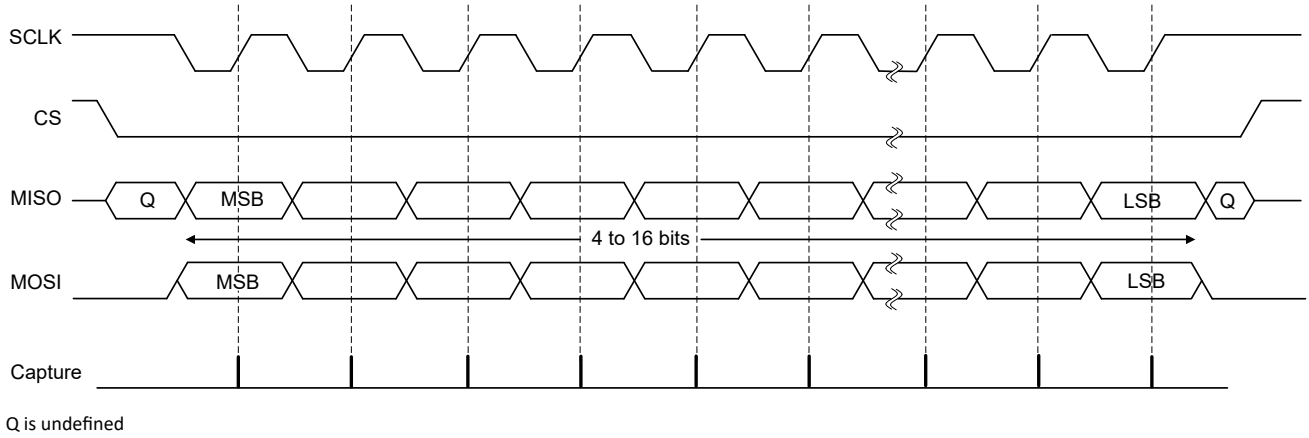


Figure 20-5. Motorola SPI Frame Format with SPO=1 and SPH=1

In this configuration, the following occurs during idle periods:

- SCLK is forced high.
- CS is forced high.
- The transmit data line PICO is forced low.
- When the SPI is configured as a controller, the SCLK pin is enabled.
- When the SPI is configured as a peripheral, the SCLK pin is disabled.

If the SPI is enabled and valid data is in the TX FIFO, the start of transmission is signified by the CS controller signal going low. The controller PICO output pin is enabled. After an additional one-half SCLK period, both controller and peripheral data are enabled onto their respective transmission lines. At the same time, SCLK is enabled with a falling-edge transition. Data is then captured on the rising edges and propagated on the falling edges of the SCLK signal.

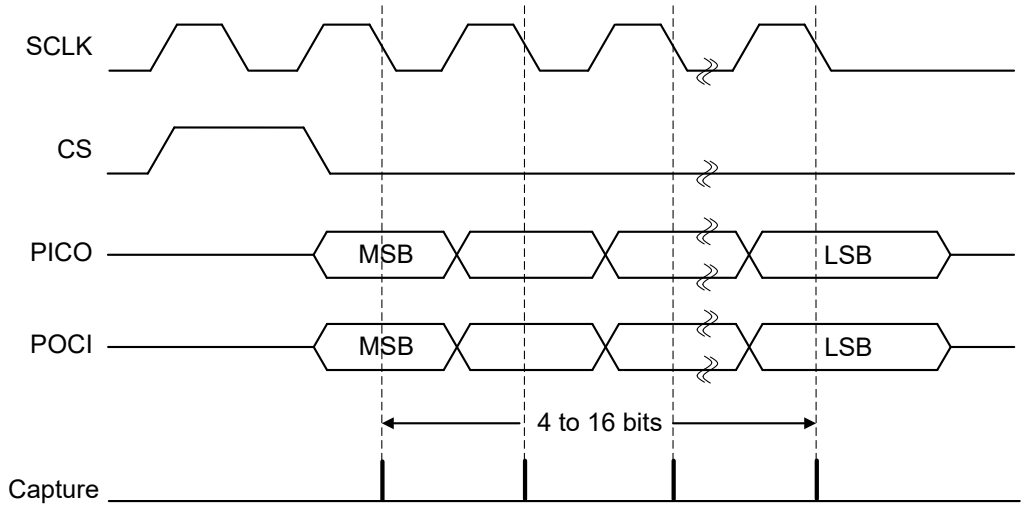
For a single word transmission, after all bits are transferred, the CS line returns to its IDLE high state one SCLK period after the last bit is captured. For continuous back-to-back transmissions, the CS pin remains in its active low state until the final bit of the last word is captured and then returns to its IDLE state. For continuous back-to-back transfers, the CS pin is held low between successive data words and terminates like a single-word transfer.

The serial clock (SCLK) is held inactive while the SPI is idle and SCLK transitions at the programmed frequency only during active transmission or reception of data. The IDLE state of SCLK provides a receive timeout indication that occurs when the RX FIFO still contains data after a timeout period.

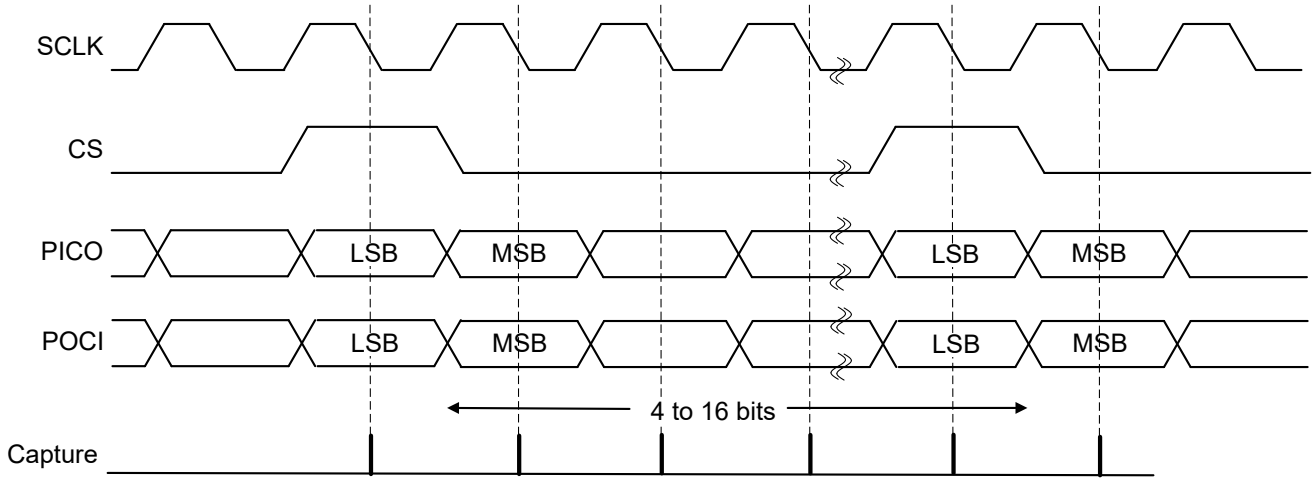
20.3.8.2 Texas Instruments Synchronous Serial Frame Format

The SPI module is compatible with the Texas Instruments Synchronous Serial frame format.

Figure 20-6 shows the TI synchronous serial frame format for a single and continuous transmitted frame.



Single Transmission Signal Sequences



Continuous Transmission Signal Sequences

Figure 20-6. TI Synchronous Serial Frame Format

SCLK and CS are forced low and the transmit data line PICO is put in tristate whenever the SPI is idle. When the TX FIFO contains data, CS is pulsed high for one SCLK period. The transmitted value is also transferred from the TX FIFO to the serial shift register of the transmit logic. On the next rising edge of SCLK, the MSB of the 4- to 16-bit data frame is shifted out on the PICO pin. Likewise, the MSB of the received data is shifted onto the POCI pin by the off-chip serial peripheral device. Both the SPI and the off-chip serial peripheral device then

clock each data bit into their serial shifter on each falling edge of SCLK. The received data is transferred from the serial shifter to the RX FIFO on the first rising edge of SCLK after the least significant bit (LSB) is latched.

The serial clock (SCLK) is held inactive while the SPI is idle and SCLK transitions at the programmed frequency only during active transmission or reception of data. The IDLE state of SCLK provides a receive time-out indication that occurs when the RX FIFO still contains data after a time-out period.

Note

When configured as a peripheral in TI Synchronous frame format, the off-chip controller device has to ensure that spurious pulses are not driven on the SCLK when CS is high.

This can result in the first data frame getting transmitted incorrectly by the peripheral.

20.3.8.3 MICROWIRE Frame Format

Figure 20-7 shows the MICROWIRE frame format for a single frame. Figure 20-8 shows the same format when back-to-back frames are transmitted.

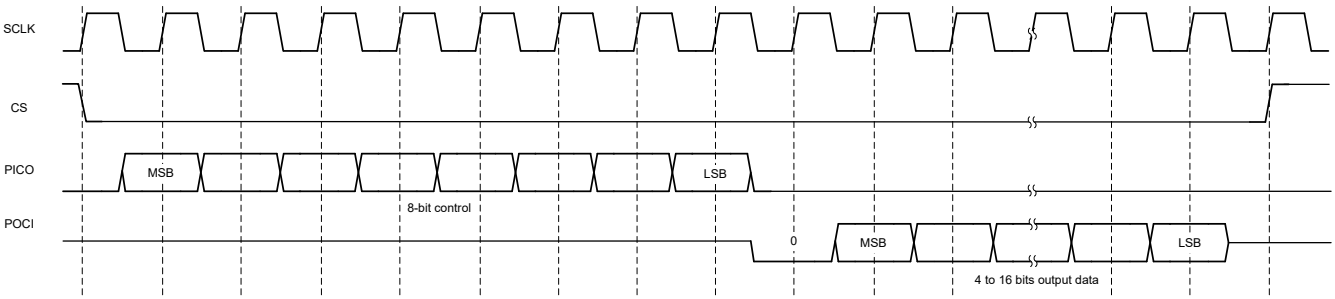


Figure 20-7. MICROWIARE Frame Format (Single Frame)

MICROWIRE format is similar to SPI format, except that transmission is half-duplex and uses a controller-peripheral message-passing technique. Each serial transmission begins with an 8-bit control word that is transmitted from the SPI to the off-chip peripheral device. During this transmission, the SPI does not receive incoming data. After the message is sent, the off-chip peripheral decodes the message and waits one serial clock after the last bit of the 8-bit control message is sent. The off-chip peripheral then responds with the required data. The returned data is 4 to 16 bits long, making the total frame length anywhere from 13 to 25 bits.

In this configuration, the following occurs during idle periods:

- SCLK is forced low.
- CS is forced high.
- The transmit data line, PICO, is typically forced low.

Writing a control byte to the TX FIFO triggers a transmission. The falling edge of CS transfers the value of the TX FIFO to the serial shift register of the transmit logic and shifts the MSB of the 8-bit control frame out onto the PICO pin. CS remains low for the duration of the frame transmission. The POCI pin remains in the tri-state condition during this transmission.

The off-chip serial peripheral device latches each control bit into the serial shifter on each rising edge of SCLK. After the last bit is latched by the peripheral device, the control byte is decoded during a one clock wait state and the peripheral responds by transmitting data back to the SPI. Each bit is driven onto the POCI line on the falling edge of SCLK. The SPI latches each bit on the rising edge of SCLK. At the end of the frame for single transfers, the CS signal is pulled high one clock period after the last bit is latched in the receive serial shifter transferring the data to the RX FIFO.

Note

The off-chip peripheral device can place the receive line in a tri-state condition either on the falling edge of SCLK (after the LSB has been latched by the receive shifter), or when the CS pin goes high.

For continuous transfers, data transmission begins and ends like a single transfer, but the CS line is held low and data transmits back-to-back. The control byte of the next frame follows the LSB of the received data from the current frame. After the LSB of the frame is latched into the SPI, each received value is transferred from the receive shifter on the falling edge of SCLK.

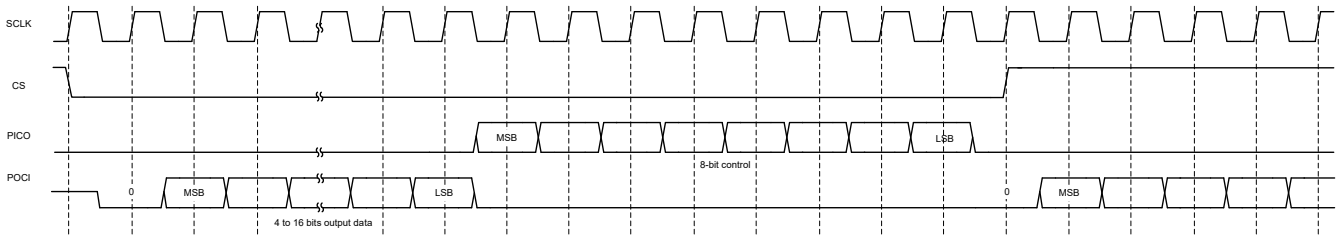


Figure 20-8. MICROWIRE Frame Format (Continuous Transfer)

In the MICROWIRE mode, the SPI peripheral samples the first bit of receive data on the rising edge of SCLK after CS has gone low. Controllers driving a free-running SCLK must ensure that the CS signal has sufficient setup and hold margins with respect to the rising edge of SCLK.

20.3.9 CRC Configuration

- CRC8-CCITT and CRC16-CCITT schemes are implemented in the SPI module independently for transmit and receive operations.
- CRC functionality is enabled by the application when the SPI frame size is only 8 bits or 16 bits.
- This is a software guideline and no specific check is done in hardware based on frame size configuration.
- For 8-bit transfers, the CRC8 or CRC16 schemes can be selected by application.
- For 16-bit transfers, the CRC16 scheme has to be selected.
- The CRC on TX and RX paths are always active and there is no need to enable CRC functionality explicitly in software.

Transmitter side operation (CRC8/CR16):

- The TX CRC block is logically located between the SPI.TXDATA register and TXFIFO in the design.
- Select CRC polynomial (8 or 16) based on SPI data frame size (selection applies to both TX and RX CRC units).
- TX side CRC state register resets to the seed value of 0xFF or 0xFFFF.
- When the bus controller (CPU or µDMA) writes data into SPI.TXDATA register, the data is written into the TXFIFO and at the same time used to update the SPI.TXCRC register by hardware logic.
- Data from TXFIFO gets loaded into the shift register and transmitted out.
- After the required amount of data has been written into SPI.TXDATA register, the software has to read the SPI.TXCRC state register.
- Software must write the obtained CRC checksum into the SPI.TXDATA register for transmission.
- Initialize the SPI.TXCRC state register and repeat this sequence for every block of SPI data transmission.

Receiver side operation (CRC8/CRC16):

- The RX CRC block is logically located between RXDATA and RXFIFO in the design.
- Select a CRC polynomial (8 or 16) based on the SPI data frame size (selection applies to both TX and RX CRC units).
- The RX side CRC state register resets to the seed value of 0xFF or 0xFFFF.
- Data is received into the shift register and is loaded into RXFIFO during the receive operation.
- When the bus controller (CPU or µDMA) reads data from the SPI.RXDATA register, the data is returned from RXFIFO and at the same time used to update the SPI.RXCRC state register by hardware logic.

- After the required amount of data is read from SPI.RXDATA, software can either:
  - Read out the received checksum and then check the SPI.RXCRC state, which is zero if there are no errors.
  - Read out the SPI.RXCRC state and then read out the checksum through SPI.RXDATA and compare the two values.
- Initialize SPI.RXCRC state register and repeat this sequence for every block of SPI data reception.

In case SPI functionality is not being used, the CRC engine can be used as a general-purpose CRC generator.

CTL0.GPCRCEN register bit can be set to enable this functionality. The transmit side CRC can then be used by application software when SPI enable is zero.

20.3.10 Auto CRC Functionality

The SPI includes a feature to automatically insert the CRC when the TX FIFO underflows. This feature can be enabled by the SPI.CTL0[14] AUTOCRC bit.

When this bit is set, the SPI module loads the calculated CRC checksum into the TX FIFO after all bytes are transmitted when TXFIFO underflow is signaled. This causes the CRC to be transmitted out automatically at the end of the data block.

There is no need for software to read and load this CRC value into the FIFO through a TXDATA register write.

Software must read the SPI.TXCRC[31] AUTOINS bit to reinitialize the TX CRC engine to a seed of all ones after the transfer of data and CRC is done.

Similarly, after reading all the received data through the SPI.RXDATA register, software must read the SPI.RXCRC register to obtain the CRC value and auto-initialize the RX CRC engine to a seed value of all ones.

Note

Care must be taken to ensure the TXFIFO does not empty and signal an underflow if FIFO filling and SPI transmission occur in parallel. This can lead to a CRC being automatically inserted at an unwanted instant. When operating at SPI rates 8MHz or higher, it is advisable to ensure that the TXFIFO is loaded with all the required data (or at least two data frames) before enabling SPI transfers with Auto CRC enabled.

Note

In MICROWIRE frame format, the AUTOCRC feature only supports CRC8 configuration in controller mode.

20.3.11 Auto Header Functionality

When SPI is to be used within a transceiver setting with strict timing requirements in peripheral mode response, the software overhead associated with loading/transmitting data can, in some cases, be limiting.

To ensure that the latest data (for example FIFO level, RSSI value, event flags, and so on) is contained in the peripheral header, a mechanism to atomically update the FIFO with header data several times before the external controller starts an SPI transmission is included.

SPI.TXFHDR32, SPI.TXFHDR24, SPI.TXFHDR16, and SPI.TXFHDR8 registers are included within the SPI module. When any of these registers are written:

- The write is ignored if the external controller has already started transmission by pulling the CS low.
- If no active SPI transmission is ongoing, the RXFIFO and TXFIFO pointers are reset.
- The TX CRC engine is reset to a value of all ones.
- Depending on whether TXFHDR8, TXFHDR16, TXFHDR24, TXFHDR32 registers are written – 1, 2, 3, or 4 bytes of data are written into the TXFIFO in consecutive CLKSVT cycles.

- µDMA and CPU interrupts are blocked until the written header is committed on the POCI, after the controller starts SPI transmissions so that no active CPU or µDMA jobs fill up the TXFIFO at the same time as the header gets updated by this feature.

The SPI.TXFHDRC register is included to control this feature. The SPI.TXFHDRC register contains these fields:

- HDREN – This bit has to be set to enable the atomic header feature when CSGATE is set to BLK. Otherwise, any write to TXFHDR registers sets this bit automatically.
- HDRCMT – This bit is set when the external controller pulls the CS low and the peripheral commits the header written into the FIFO on the POCI. This bit continues to be set until the software clears it. The hold on CPU or µDMA interrupt lines is released when this bit is set.
- HDRIGN – This bit is set when a write to header registers occurs after the HDRCMT is set. This bit can be polled by software to figure out whether the last payload was written to the TXFIFO or not.
- CSGATE – This is set to UNBLK by default. The first header write can occur even when CS is low (active). When CSGATE is set to BLK, the header writes must occur only when CS is inactive.

SPI.CTL0[17] IDLEPOCI register bit is included to drive a configurable high/low value of the POCI pin to ease signaling to the controller about a change in the peripheral readiness. If the MSB of header data written into the FIFO is opposite to the IDLEPOCI, the external controller can detect the same by polling the POCI pin.

20.3.12 SPI Status

The external controller is supposed to always send the number of clocks equal to the DSS value written, before deasserting CS and ending transmission. In case CS is deactivated before the entire data frame has been sent out by the peripheral, then the CSD error bit is set and can be read within SPI.STA[5] CSD. This bit, once set, must be cleared by software.

The TX FIFO full level indicating the number of entries written into the TXFIFO can be read out through the SPI.STA[13:8] TXFIFOLVL bit field.

Additional status bits related to peripheral mode transfer complete, SPI busy indication, and TX and RX FIFO flags can be read out from the SPI.STA register. The peripheral mode transfer-complete indication bit, once set, must be cleared by the software.

20.3.13 Debug Halt

Debug halt is available in the SPI module and is controlled by the SPI.EMU register. When the SPI.EMU[0] HALT bit is set to 1, and the SPI module freezes operations as described below.

- If SPI is configured in Controller mode, then debug halt freezes SPI operations at the next DSS boundary.
- If SPI is configured in Peripheral mode, then debug halt freezes SPI operations immediately.
- FIFO pointers are not incremented if RXDATA read is attempted during a debug halt.

20.4 µDMA Operation

The SPI module provides an interface to the µDMA controller with separate channels for transmit and receive. The SPI DMA Control register (SPI.DMACR) allows the µDMA to operate with the SPI. When µDMA operation is enabled, the SPI asserts a µDMA request on the receive or transmit channel when the associated FIFO can transfer data. For the receive channel, a single transfer request is asserted whenever any data is in the RX FIFO. Whenever data in the RX FIFO reaches the configured level set in the SPI.IFLS[10:8] RXSEL bit field, a burst transfer request is asserted. The supported settings for RX FIFO are: ¼, ½ (default), ¾, full, and at least one location is available in the FIFO. For the transmit channel, a single transfer request is asserted whenever at least one empty location is in the TX FIFO. Whenever the TX FIFO reaches the configured level set through SPI.IFLS[2:0] TXSEL, the burst request is asserted. The supported settings for TX FIFO are ¼, ½ (default), ¾, and empty, and at least one location is available in the FIFO. The µDMA controller handles the single and burst µDMA transfer requests automatically depending on how the µDMA channel is configured.

Note

Software must avoid ¾ threshold selection for µDMA operation.

To enable µDMA operation for the receive channel, set the SPI.DMACR[0] RXEN bit. To enable µDMA operation for the transmit channel, set the SPI.DMACR[8] TXEN register bit. If the µDMA is enabled and appropriate bits are cleared in the DMA Done Mask register (DMA.DONEMASK) the µDMA controller triggers an interrupt when a transfer completes. This interrupt can be chosen as one of the sources of the combined SPI interrupt. If interrupts are used for SPI operation and the µDMA is enabled, the SPI interrupt handler must be designed to handle the µDMA completion interrupt. The status of TX and RX DMA done interrupts can be read from the Channel Request Done register (DMA.REQDONE). They can also be read from SPI.RIS[8] DMATX bit and the SPI.RIS[7] DMARX bit. For clearing the TX and RX DMA done interrupts, the corresponding bits in the DMA.REQDONE register must be set to 1.

For more details about programming the µDMA controller, see Chapter 15

20.5 Initialization and Configuration

The following describes the necessary steps to enable and initialize the SPI.

TI recommends using the SPI driver in the SimpleLink™ CC23xx Software Development Kit (SDK) when using the SPI.

1. Ensure the SVT power domain is powered up properly.

2. Enable the SPI module clock in CLKCTL by setting the CLKCTL.CLKCFG0[10] SPI0 bit.

3. Configure the IOC module to route the PICO, POCI, CS, and SCLK functionality from I/Os to the SPI module. IOCFGn.PORTCFG must be written to the correct IDs.

For each of the frame formats, the SPI is configured using the following steps:

1. Ensure that the SPI.CTL1[0] ENABLE bit is cleared before making any configuration changes.
2. Configure the clock prescaler divisor by writing to the SPI.CLKCFG0[2:0] PRESC and SPI.CLKCFG1[9:0] SCR bit fields.
3. Write the SPI.CTL0 register with the following configuration:
   a. Desired clock phase and polarity, if using Motorola™ SPI mode (SPH and SPO)
   b. The protocol mode: Motorola SPI (4-wire or 3-wire), TI SSF, MICROWIRE (FRF)
   c. The data size (DSS)
4. Select whether the SPI is a controller or peripheral:
   a. For controller operations, SPI.CTL1[2] MS is 1.
   b. For peripheral mode (output enabled), SPI.CTL1[2] MS bit is 0.
   c. For peripheral mode (output disabled), clear the SPI.CTL1[2] MS bit to 0 and set the SPI.CTL1[3] POD bit to 1.
5. Optionally, configure the µDMA channel (see Chapter 15) and enable the µDMA options in the SPI.DMACR register.
6. Enable the SPI by setting the EN bit in the SPI.CTL1 register.

As an example, assume that the SPI configuration is required to operate with the following parameters:

- Controller operation
- Texas Instruments Synchronous SPI mode
- 1-Mbps bit rate
- 8 data bits

Assuming the system clock is 48MHz, the bit-rate calculation is shown in Equation 10

$$\text{SCLK} = \text{CLKSVT} / [\text{PRESC} \times (1 + \text{SCR})]1000000 \text{ bps} = 48000000 \text{ Hz} / [2 \times (1 + 23)] \quad (10)$$

In this case, if PRESC = 0x2, SCR must be 0x18.

The configuration sequence is:

- Verify that the EN bit in the SPI.CTL1 register is cleared.
- Write the SPI.CLKCFG0 register with a value of 0x00000002.

- Write the SPI.CLKCFG1 register with a value of 0x00000018.
- Write the SPI.CTL0 register with a value of 0x000047.
- Write the SPI.CTL1 register with a value of 0x00000004.
- The SPI is then enabled by setting the EN bit in the SPI.CTL1 register.

20.6 SPI Registers

Table 20-2 lists the memory-mapped registers for the SPI registers. All register offset addresses not listed in Table 20-2 should be considered as reserved locations and the register contents should not be modified.

Table 20-2. SPI Registers

Offset	Acronym	Register Name	Section
0h	DESC	Module Description	Go
44h	IMASK	Interrupt mask	Go
48h	RIS	Raw interrupt status	Go
4Ch	MIS	Masked interrupt status	Go
50h	ISET	Interrupt set	Go
54h	ICLR	Interrupt clear	Go
58h	IMSET	Interrupt mask set	Go
5Ch	IMCLR	Interrupt mask clear	Go
60h	EMU	Emulation	Go
100h	CTL0	Control 0	Go
104h	CTL1	Control 1	Go
108h	CLKCFG0	Clock configuration 0	Go
10Ch	CLKCFG1	Clock configuration 1	Go
110h	IFLS	Interrupt FIFO Level Select	Go
114h	DMACR	DMA control	Go
118h	RXCRC	Receive CRC	Go
11Ch	TXCRC	Transmit CRC	Go
120h	TXFHDR32	Header write for 32bits	Go
124h	TXFHDR24	Header write for 24bits	Go
128h	TXFHDR16	Header write for 16bits	Go
12Ch	TXFHDR8	Header write for 8bits	Go
130h	TXFHDRC	Atomic header control	Go
140h	RXDATA	Receive data	Go
150h	TXDATA	Transmit data	Go
160h	STA	Status	Go

Complex bit access types are encoded to fit into small table cells. Table 20-3 shows the codes that are used for access types in this section.

Table 20-3. SPI Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

20.6.1 DESC Register (Offset = 0h) [Reset = 604D1010h]

DESC is shown in Table 20-4.

Return to the Summary Table.

Description Register. This register provides IP module ID, revision information, instance index and standard MMR registers offset.

Table 20-4. DESC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	MODID	R/W	604Dh	Module identifier used to uniquely identify this IP.
15-12	STDIPOFF	R/W	1h	Standard IP MMR block offset. Standard IP MMRs are the set of from aggregated IRQ registers till DTB. 0: Standard IP MMRs do not exist 0x1-0xF: Standard IP MMRs begin at offset of (64*STDIPOFF from the base IP address)
11-8	INSTIDX	R/W	0h	IP Instance ID number. If multiple instances of IP exist in the device, this field can identify the instance number (0-15).
7-4	MAJREV	R/W	1h	Major revision of IP (0-15).
3-0	MINREV	R/W	0h	Minor revision of IP (0-15).

20.6.2 IMASK Register (Offset = 44h) [Reset = 00000000h]

IMASK is shown in Table 20-5.

Return to the Summary Table.

Interrupt mask. This register selects interrupt sources which are allowed to pass from RIS to MIS when the corresponding bit-fields are set to 1.

Table 20-5. IMASK Register Field Descriptions

Bit	Field	Type	Reset	Description
31-9	RESERVED	R	0h	Reserved
8	DMATX	R/W	0h	DMA Done TX event mask. 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
7	DMARX	R/W	0h	DMA Done RX event mask. 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
6	IDLE	R/W	0h	SPI Idle event mask. 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
5	TXEMPTY	R/W	0h	Transmit FIFO Empty event mask. 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
4	TX	R/W	0h	Transmit FIFO event mask. 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
3	RX	R/W	0h	Receive FIFO event. 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
2	RTOUT	R/W	0h	SPI Receive Time-Out event mask. 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
1	PER	R/W	0h	Parity error event mask. 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask
0	RXOVF	R/W	0h	RXFIFO overflow event mask. 0h = Clear Interrupt Mask 1h = Set Interrrupt Mask

20.6.3 RIS Register (Offset = 48h) [Reset = 00000000h]

RIS is shown in Table 20-6.

Return to the Summary Table.

Raw interrupt status. This register reflects the state of all pending interrupts, regardless of masking. This register allows the user to implement a poll scheme. A flag set in this register can be cleared by writing 1 to the corresponding ICLR register bit.

Table 20-6. RIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-9	RESERVED	R	0h	Reserved
8	DMATX	R/W	0h	DMA Done event for TX. This interrupt is set if the TX DMA channel sends the DONE signal. This allows the handling of the TX DMA event inside SPI. 0h = Interrupt did not occur 1h = Interrupt occurred
7	DMARX	R/W	0h	DMA Done event for RX. This interrupt is set if the RX DMA channel sends the DONE signal. This allows handling of the DMA RX event inside SPI. 0h = Interrupt did not occur 1h = Interrupt occurred
6	IDLE	R/W	0h	SPI has completed transfers and moved to IDLE mode. This bit is set when STA.BUSY goes low. 0h = Interrupt did not occur 1h = Interrupt occurred
5	TXEMPTY	R/W	0h	Transmit FIFO Empty interrupt mask. This interrupt is set when all data in the Transmit FIFO has been moved to the shift register. 0h = Interrupt did not occur 1h = Interrupt occurred
4	TX	R/W	0h	Transmit FIFO event. This interrupt is set if the selected Transmit FIFO level has been reached. 0h = Interrupt did not occur 1h = Interrupt occurred
3	RX	R/W	0h	Receive FIFO event. This interrupt is set if the selected Receive FIFO level has been reached 0h = Interrupt did not occur 1h = Interrupt occurred
2	RTOUT	R/W	0h	SPI Receive Time-Out event. This interrupt is set if no activity is detected on the input clock line within the time period dictated by CTL1.RTOUT value. This is applicable only in peripheral mode. 0h = Interrupt did not occur 1h = Interrupt occurred
1	PER	R/W	0h	Parity error event. This bit is set if a Parity error has been detected 0h = Interrupt did not occur 1h = Interrupt occurred
0	RXOVF	R/W	0h	RXFIFO overflow event. This interrupt is set if an RX FIFO overflow has been detected. 0h = Interrupt did not occur 1h = Interrupt occurred

20.6.4 MIS Register (Offset = 4Ch) [Reset = 00000000h]

MIS is shown in Table 20-7.

Return to the Summary Table.

Masked interrupt status. This register is simply a bitwise AND of the contents of IMASK and RIS registers. A flag set in this register can be cleared by writing 1 to the corresponding ICLR register bit.

Table 20-7. MIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-9	RESERVED	R	0h	Reserved
8	DMATX	R/W	0h	Masked DMA Done event for TX. 0h = Interrupt did not occur 1h = Interrupt occurred
7	DMARX	R/W	0h	Masked DMA Done event for RX. 0h = Interrupt did not occur 1h = Interrupt occurred
6	IDLE	R/W	0h	Masked SPI IDLE event. 0h = Interrupt did not occur 1h = Interrupt occurred
5	TXEMPTY	R/W	0h	Masked Transmit FIFO Empty event. 0h = Interrupt did not occur 1h = Interrupt occurred
4	TX	R/W	0h	Masked Transmit FIFO event. 0h = Interrupt did not occur 1h = Interrupt occurred
3	RX	R/W	0h	Masked receive FIFO event. 0h = Interrupt did not occur 1h = Interrupt occurred
2	RTOUT	R/W	0h	Masked SPI Receive Time-Out event. 0h = Interrupt did not occur 1h = Interrupt occurred
1	PER	R/W	0h	Masked Parity error event. 0h = Interrupt did not occur 1h = Interrupt occurred
0	RXOVF	R/W	0h	Masked RXFIFO overflow event. 0h = Interrupt did not occur 1h = Interrupt occurred

20.6.5 ISET Register (Offset = 50h) [Reset = 00000000h]

ISET is shown in Table 20-8.

Return to the Summary Table.

Interrupt set register. This register can used by software for diagnostics and safety checking purposes. Writing a 1 to a bit in this register will set the event and the corresponding RIS bit also gets set. If the corresponding IMASK bit is set, then the corresponding MIS register bit also gets set.

Table 20-8. ISET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-9	RESERVED	R	0h	Reserved
8	DMATX	R/W	0h	Set DMA Done event for TX. 0h = Writing 0 has no effect 1h = Set Interrupt
7	DMARX	R/W	0h	Set DMA Done event for RX. 0h = Writing 0 has no effect 1h = Set Interrupt
6	IDLE	R/W	0h	Set SPI IDLE event. 0h = Writing 0 has no effect 1h = Set Interrupt
5	TXEMPTY	R/W	0h	Set Transmit FIFO Empty event. 0h = Writing 0 has no effect 1h = Set Interrupt
4	TX	R/W	0h	Set Transmit FIFO event. 0h = Writing 0 has no effect 1h = Set Interrupt
3	RX	R/W	0h	Set Receive FIFO event. 0h = Writing 0 has no effect 1h = Set Interrupt
2	RTOUT	R/W	0h	Set SPI Receive Time-Out Event. 0h = Writing 0 has no effect 1h = Set Interrrupt Mask
1	PER	R/W	0h	Set Parity error event. 0h = Writing 0 has no effect 1h = Set Interrupt
0	RXOVF	R/W	0h	Set RXFIFO overflow event. 0h = Writing 0 has no effect 1h = Set Interrupt

20.6.6 ICLR Register (Offset = 54h) [Reset = 00000000h]

ICLR is shown in Table 20-9.

Return to the Summary Table.

Interrupt clear register. This register allows software to clear interrupts. Writing a 1 to a bit in this register will clear the event and the corresponding RIS bit also gets cleared. If the corresponding IMASK bit is set, then the corresponding MIS register bit also gets cleared.

Table 20-9. ICLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-9	RESERVED	R	0h	Reserved
8	DMATX	R/W	0h	Clear DMA Done event for TX. 0h = Writing 0 has no effect 1h = Clear Interrupt
7	DMARX	R/W	0h	Clear DMA Done event for RX. 0h = Writing 0 has no effect 1h = Clear Interrupt
6	IDLE	R/W	0h	Clear SPI IDLE event. 0h = Writing 0 has no effect 1h = Clear Interrupt
5	TXEMPTY	R/W	0h	Clear Transmit FIFO Empty event. 0h = Writing 0 has no effect 1h = Clear Interrupt
4	TX	R/W	0h	Clear Transmit FIFO event. 0h = Writing 0 has no effect 1h = Clear Interrupt
3	RX	R/W	0h	Clear Receive FIFO event. 0h = Writing 0 has no effect 1h = Clear Interrupt
2	RTOUT	R/W	0h	Clear SPI Receive Time-Out Event. 0h = Writing 0 has no effect 1h = Set Interrrupt Mask
1	PER	R/W	0h	Clear Parity error event. 0h = Writing 0 has no effect 1h = Clear Interrupt
0	RXOVF	R/W	0h	Clear RXFIFO overflow event. 0h = Writing 0 has no effect 1h = Clear Interrupt

20.6.7 IMSET Register (Offset = 58h) [Reset = 00000000h]

IMSET is shown in Table 20-10.

Return to the Summary Table.

Interrupt mask set register. Writing a 1 to a bit in this register will set the corresponding IMASK bit.

Table 20-10. IMSET Register Field Descriptions

Bit	Field	Type	Reset	Description
31-9	RESERVED	R	0h	Reserved
8	DMATX	R/W	0h	Set DMA Done for TX event mask 0h = Writing 0 has no effect 1h = Set interrupt mask
7	DMARX	R/W	0h	Set DMA Done for RX event mask 0h = Writing 0 has no effect 1h = Set interrupt mask
6	IDLE	R/W	0h	Set SPI IDLE event mask 0h = Writing 0 has no effect 1h = Set interrupt mask
5	TXEMPTY	R/W	0h	Set Transmit FIFO Empty event mask 0h = Writing 0 has no effect 1h = Set interrupt mask
4	TX	R/W	0h	Set Transmit FIFO event mask 0h = Writing 0 has no effect 1h = Set interrupt mask
3	RX	R/W	0h	Set Receive FIFO event mask 0h = Writing 0 has no effect 1h = Set interrupt mask
2	RTOUT	R/W	0h	Set SPI Receive Time-Out event mask 0h = Writing 0 has no effect 1h = Set interrupt mask
1	PER	R/W	0h	Set Parity error event mask 0h = Writing 0 has no effect 1h = Set interrupt mask
0	RXOVF	R/W	0h	Set RXFIFO overflow event mask 0h = Writing 0 has no effect 1h = Set interrupt mask

20.6.8 IMCLR Register (Offset = 5Ch) [Reset = 00000000h]

IMCLR is shown in Table 20-11.

Return to the Summary Table.

Interrupt mask clear register. Writing a 1 to a bit in this register will clear the corresponding IMASK bit.

Table 20-11. IMCLR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-9	RESERVED	R	0h	Reserved
8	DMATX	R/W	0h	Clear DMA Done for TX event mask 0h = Writing 0 has no effect 1h = Clear interrupt mask
7	DMARX	R/W	0h	Clear DMA Done for RX event mask 0h = Writing 0 has no effect 1h = Clear interrupt mask
6	IDLE	R/W	0h	Clear SPI IDLE event mask 0h = Writing 0 has no effect 1h = Clear interrupt mask
5	TXEMPTY	R/W	0h	Clear Transmit FIFO Empty event mask 0h = Writing 0 has no effect 1h = Clear interrupt mask
4	TX	R/W	0h	Clear Transmit FIFO event mask 0h = Writing 0 has no effect 1h = Clear interrupt mask
3	RX	R/W	0h	Clear Receive FIFO event mask 0h = Writing 0 has no effect 1h = Clear interrupt mask
2	RTOUT	R/W	0h	Clear SPI Receive Time-Out event mask 0h = Writing 0 has no effect 1h = Clear interrupt mask
1	PER	R/W	0h	Clear Parity error event mask 0h = Writing 0 has no effect 1h = Clear interrupt mask
0	RXOVF	R/W	0h	Clear RXFIFO overflow event mask 0h = Writing 0 has no effect 1h = Clear interrupt mask

20.6.9 EMU Register (Offset = 60h) [Reset = 00000000h]

EMU is shown in Table 20-12.

Return to the Summary Table.

Emulation control register. This register controls the behavior of the IP related to core halted input.

Table 20-12. EMU Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Reserved
0	HALT	R/W	0h	Halt control 0h = Free run option. The IP ignores the state of the core halted input. 1h = Freeze option. The IP freezes functionality when the core halted input is asserted, and resumes when it is deasserted. The freeze can either be immediate or after the IP has reached a boundary (end of word boundary, based on DSS configuration) from where it can resume without corruption.

20.6.10 CTL0 Register (Offset = 100h) [Reset = 00000000h]

CTL0 is shown in Table 20-13.

Return to the Summary Table.

SPI control register 0

Table 20-13. CTL0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-18	RESERVED	R	0h	Reserved
17	IDLEPOCI	R/W	0h	The Idle value of POCI - when TXFIFO is empty and before data is written into TXFIFO - can be controlled by this field. 0h = POCI output idle value of '0' 1h = POCI outputs idle value of '1'
16	GPCRCEN	R/W	0h	General purpose CRC enable. This bit enables transmit side CRC unit for general purpose use by software when SPI is disabled (CTL1.EN = 0). This bit must be 0 when SPI is enabled. 0h = Transmit side CRC unit is not available for general purpose software use 1h = Transmit side CRC unit is available for general purpose software use
15	CRCPOLY	R/W	0h	CRC polynomial selection. 0h = Selects 8-bit CCITT CRC polynomial 1h = Selects 16-bit CCITT CRC polynomial
14	AUTOCRC	R/W	0h	Auto insert CRC 0h = Do not insert CRC into TXFIFO upon TXFIFO underflow 1h = Insert CRC into TXFIFO upon TXFIFO underflow
13	CRCEND	R/W	0h	CRC16 Endianness 0h = Auto-insertion of CRC16 is most-significant byte first 1h = Auto-insertion of CRC16 is least-significant byte first
12	CSCLR	R/W	0h	Clear shift register counter on CS inactive. This bit is relevant only in the peripheral mode, when CTL1.MS=0. 0h = Disable automatic clear of shift register when CS goes inactive. 1h = Enable automatic clear of shift register when CS goes inactive.
11	FIFORST	R/W	0h	This bit is used to reset transmit and receive FIFO pointers. This bit is auto cleared once the FIFO pointer reset operation is completed. 0h = FIFO pointers reset completed when 0 is read 1h = Trigger FIFO pointers reset when written to 1.
10	HWCSN	R/W	0h	Hardware controlled chip select (CS) value. When set CS is zero till TX FIFO is empty, as in - a. CS is de-asserted b. All data bytes are transmitted c. CS is asserted 0h = HWCSN Disable 1h = HWCSN Enable
9	SPH	R/W	0h	SCLK phase (Motorola SPI frame format only). This bit selects the clock edge that captures data and enables it to change state. It has the most impact on the first bit transmitted by either permitting or not permitting a clock transition before the first data capture clock edge. 0h = Data is captured on the first clock edge transition. 1h = Data is captured on the second clock edge transition.
8	SPO	R/W	0h	SCLK polarity (Motorola SPI frame format only). 0h = SPI produces a steady state LO value on the SCLK 1h = SPI produces a steady state HI value on the SCLK
7	RESERVED	R	0h	Reserved

Table 20-13. CTL0 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
6-5	FRF	R/W	0h	Frame format select 0h = Motorola SPI frame format (3-wire mode) 1h = Motorola SPI frame format (4-wire mode) 2h = TI synchronous serial frame format 3h = MICROWIRE frame format
4	RESERVED	R	0h	Reserved
3-0	DSS	R/W	0h	Data size select. The applicable DSS values for controller mode operation are 0x3 to 0xF and for peripheral mode operation are 0x6 to 0xF. DSS values 0x0 to 0x2 are reserved and must not be used. 3h = 4-bits data size 4h = 5-bits data size 5h = 6-bits data size 6h = 7-bits data size 7h = 8-bits data size 8h = 9-bits data size 9h = 10-bits data size Ah = 11-bits data size Bh = 12-bits data size Ch = 13-bits data size Dh = 14-bits data size Eh = 15-bits data size Fh = 16-bits data size

20.6.11 CTL1 Register (Offset = 104h) [Reset = 00000004h]

CTL1 is shown in Table 20-14.

Return to the Summary Table.

SPI control register 1

Table 20-14. CTL1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-30	RESERVED	R	0h	Reserved
29-24	RTOUT	R/W	0h	Receive Timeout (only for Peripheral mode) Defines the number of CLKSVT clock cycles after which the Receive Timeout flag RIS.RTOUT is set. A value of 0 disables this function.
23-16	REPTX	R/W	0h	Counter to repeat last transfer (only in controller mode) 0: repeat last transfer is disabled. x: repeat the last transfer with the provided value. The transfer will be started with writing a data into the TX FIFO. Sending the data will be repeated REPTX number of times, so the data will be transferred x+1 times in total. It can be used to clean a transfer or to pull a certain amount of data by a peripheral.
15-12	CDMODE	R/W	0h	Commnd Data Mode. This bit field value determines the behavior of C/D or CS signal when CDEN = 1. CS pin held low indicates command phase and CS pin held high indicates data phase. When CDMODE = 0x0, the CS pin is always held high during transfer indicating data phase only operation (manual mode). When CDMODE = 0xF, the CS pin is always held low during transfer indicating command phase only operation (manual mode). When CDMODE = 0x1 to 0xE, the CS pin is held low for the number of bytes indicated by CDMODE value for the command phase and held high for the remaining transfers in the data phase (automatic mode). When CDMODE is set to value 0x1 to 0xE, reading CDMODE during operation indicates the remaining bytes to be transferred in the command phase. 0h = Manual mode: Data Fh = Manual mode: Command
11	CDEN	R/W	0h	Command/Data mode enable. This feature is applicable only in controller mode and for 8-bit transfers (CTL0.DSS = 7). The chip select pin is used for command/data signaling in Motorola SPI frame format (3-wire) operation. 0h = C/D Mode Disable 1h = C/D Mode Enable
10-8	RESERVED	R	0h	Reserved
7	PBS	R/W	0h	Parity bit select 0h = Bit 0 is used for Parity 1h = Bit 1 is used for Parity, Bit 0 is ignored
6	PES	R/W	0h	Even parity select. 0h = Odd Parity mode 1h = Even Parity mode
5	PEN	R/W	0h	Parity enable. If enabled the last bit will be used as parity to evaluate the correct reception of the previous bits. In case of parity mismatch the parity error flag RIS.PER will be set. This feature is available only in SPI controller mode. 0h = Disable Parity function 1h = Enable Parity function
4	MSB	R/W	0h	MSB first select. Controls the direction of receive and transmit shift register. MSB first configuration (MSB = 1) must be selected when CRC feature is used for SPI communication. 0h = LSB first 1h = MSB first

Table 20-14. CTL1 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
3	POD	R/W	0h	Peripheral data output disable. This bit is relevant only in the peripheral mode, MS=1. In multiple-peripheral systems, it is possible for a SPI controller to broadcast a message to all peripherals in the system while ensuring that only one peripheral drives data onto its serial output line. In such systems the POCI lines from multiple peripherals could be tied together. To operate in such systems, this bit field can be set if the SPI peripheral is not supposed to drive the POCI output. 0h = SPI can drive the POCI output in peripheral mode. 1h = SPI cannot drive the POCI output in peripheral mode.
2	MS	R/W	1h	Controller or peripheral mode select. This bit can be modified only when SPI is disabled, CTL1.EN=0. 0h = Select Peripheral mode 1h = Select Controller mode
1	LBM	R/W	0h	Loop back mode control 0h = Disable loopback mode. Normal serial port operation enabled. 1h = Enable loopback mode. Output of transmit serial shifter is connected to input of receive serial shifter internally.
0	EN	R/W	0h	SPI enable. NOTE: This bit field must be set to 1 using a separate write access, after the other bit fields have been configured. 0h = SPI is disabled 1h = SPI Enabled and released for operation.

20.6.12 CLKCFG0 Register (Offset = 108h) [Reset = 00000000h]

CLKCFG0 is shown in Table 20-15.

Return to the Summary Table.

Clock configuration register 0. This register is used to configure the clock prescaler.

Table 20-15. CLKCFG0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-3	RESERVED	R	0h	Reserved
2-0	PRESC	R/W	0h	Prescaler configuration 0h = Do not divide clock source 1h = Divide clock source by 2 2h = Divide clock source by 3 3h = Divide clock source by 4 4h = Divide clock source by 5 5h = Divide clock source by 6 6h = Divide clock source by 7 7h = Divide clock source by 8

20.6.13 CLKCFG1 Register (Offset = 10Ch) [Reset = 00000000h]

CLKCFG1 is shown in Table 20-16.

Return to the Summary Table.

Clock configuration register 1. This register is used to configure serial clock rate and clock count for delayed sampling in controller mode.

Table 20-16. CLKCFG1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-20	RESERVED	R	0h	Reserved
19-16	DSAMPLE	R/W	0h	Delayed sampling. In controller mode the data on the POCI pin will be delayed sampled by the defined CLKSVT clock cycles. DSAMPLE values can range from 0 to SCR+1. Typically, values of 1 or 2 would suffice.
15-10	RESERVED	R	0h	Reserved
9-0	SCR	R/W	0h	Serial clock divider. This is used to generate the transmit and receive bit rate of the SPI. The SPI bit rate: (SPI functional clock frequency)/((SCR+1)*2). SCR value can be from 0 to 1023.

20.6.14 IFLS Register (Offset = 110h) [Reset = 00000202h]

IFLS is shown in Table 20-17.

Return to the Summary Table.

Interrupt FIFO level select register. This register can be used to define the levels at which the RIS.TX, RIS.RX flags are triggered. The interrupts are generated based on FIFO level. Out of reset, the IFLS.TXSEL and IFLS.RXSEL bits are configured so that the FIFOs trigger an interrupt at the half-way mark.

Table 20-17. IFLS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-11	RESERVED	R	0h	Reserved
10-8	RXSEL	R/W	2h	Receive FIFO Level Select. The trigger points for the receive interrupt are as follows: 0h = Reserved 1h = RX FIFO >= 1/4 full 2h = RX FIFO >= 1/2 full (default) 3h = RX FIFO >= 3/4 full 4h = Reserved 5h = RX FIFO is full 6h = Reserved 7h = Trigger when RX FIFO contains >= 1 byte
7-3	RESERVED	R	0h	Reserved
2-0	TXSEL	R/W	2h	Transmit FIFO Level Select. The trigger points for the transmit interrupt are as follows: 0h = Reserved 1h = TX FIFO <= 3/4 empty 2h = TX FIFO <= 1/2 empty (default) 3h = TX FIFO <= 1/4 empty 4h = Reserved 5h = TX FIFO is empty 6h = Reserved 7h = Trigger when TX FIFO has >= 1 byte free

20.6.15 DMACR Register (Offset = 114h) [Reset = 00000000h]

DMACR is shown in Table 20-18.

Return to the Summary Table.

uDMA Control Register

Table 20-18. DMACR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-9	RESERVED	R	0h	Reserved
8	TXEN	R/W	0h	Transmit DMA enable. If this bit is set to 1, DMA for the trasmit FIFO is enabled. 0h = Disable TX DMA 1h = Enable TX DMA
7-1	RESERVED	R	0h	Reserved
0	RXEN	R/W	0h	Receive DMA enable. If this bit is set to 1, DMA for the receive FIFO is enabled. 0h = Disable RX DMA 1h = Enable RX DMA

20.6.16 RXCRC Register (Offset = 118h) [Reset = 00000000h]

RXCRC is shown in Table 20-19.

Return to the Summary Table.

Receive CRC register. Reading this register provides the computed CRC value from the receive side CRC unit. Reading this register or writing to this register with any value auto initializes the seed. The seed value is 0xFF when CTL0.CRCPOLY = 0 and 0xFFFF when CTL0.CRCPOLY = 1 for CCITT CRC polynomials. Bits[15:8] are don't care when CTL0.CRCPOLY = 0.

Table 20-19. RXCRC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	DATA	R/W	0h	CRC value SW should read RXCRC register at the end of data transmission to reinitiaze the seed value to all ones

20.6.17 TXCRC Register (Offset = 11Ch) [Reset = 00000000h]

TXCRC is shown in Table 20-20.

Return to the Summary Table.

Transmit CRC register. Reading this register provides the computed CRC value from the transmit side CRC unit. Reading this register or writing to this register with any value auto initializes the seed. The seed value is 0xFF when CTL0.CRCPOLY = 0 and 0xFFFF when CTL0.CRCPOLY = 1 for CCITT CRC polynomials. Bits[15:8] are don't care when CTL0.CRCPOLY = 0.

Table 20-20. TXCRC Register Field Descriptions

Bit	Field	Type	Reset	Description
31	AUTOCRCINS	R	0h	Status to indicate if Auto CRC has been inserted into TXFIFO. This is applicable only if CTL0.AUTOCRC enable bit is set. SW should read TXCRC register to clear auto inserted CRC at the end of the transfer. 0h = Auto CRC not yet inserted 1h = Auto CRC inserted
30-16	RESERVED	R	0h	Reserved
15-0	DATA	R/W	0h	CRC value

20.6.18 TXFHDR32 Register (Offset = 120h) [Reset = 00000000h]

TXFHDR32 is shown in Table 20-21.

Return to the Summary Table.

Header update reigster for 32 bits of header data into the TXFIFO.

Table 20-21. TXFHDR32 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	DATA	W	0h	This field can be used to write four bytes of header data into the TXFIFO

20.6.19 TXFHDR24 Register (Offset = 124h) [Reset = 00000000h]

TXFHDR24 is shown in Table 20-22.

Return to the Summary Table.

Header update reigster for 24 bits of header data into the TXFIFO.

Table 20-22. TXFHDR24 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	DATA	W	0h	This field can be used to write three bytes of header data into the TXFIFO.

20.6.20 TXFHDR16 Register (Offset = 128h) [Reset = 00000000h]

TXFHDR16 is shown in Table 20-23.

Return to the Summary Table.

Header update reigster for 16 bits of data into the TXFIFO.

Table 20-23. TXFHDR16 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	DATA	W	0h	This field can be used to write two bytes of header data into the TXFIFO.

20.6.21 TXFHDR8 Register (Offset = 12Ch) [Reset = 00000000h]

TXFHDR8 is shown in Table 20-24.

Return to the Summary Table.

Header update reigster for 8 bits of header data into the TXFIFO.

Table 20-24. TXFHDR8 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	DATA	W	0h	This field can be used to write one byte of header data into the TXFIFO.

20.6.22 TXFHDRC Register (Offset = 130h) [Reset = 00000000h]

TXFHDRC is shown in Table 20-25.

Return to the Summary Table.

Atomic Header Control register

Table 20-25. TXFHDRC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3	CSGATE	R/W	0h	Chip Select Gating control register. If this bit is set, header update register writes are blocked when chip select (CS) is active low, and HDRIGN bit is set. This bit resets to 0. 0h = The first header update register write is not blocked based on CS active status (low). If no header update occurred when CS was high (inactive), the first header update is allowed when CS is low (active), and the HDRCMT bit is set. The use case is for the external controller to ensure that the SCLK is not driven during this header update. If the header is already updated when CS is high and inactive, HDRCMT is set immediately when CS drops to active low state, and header writes when CS is low are ignored even if this UNBLK bit is set. 1h = Header update register writes are blocked when CS is active (low)
2	HDRCMT	R/W	0h	Header Committed field. This bit is set when the HDREN bit is set and CS is sampled low. This bit remains 0 otherwise. When set, this bit can be written to a value of 0 to clear. 0h = Header update is not committed 1h = Header update is committed
1	HDRIGN	R/W	0h	Header Ignored field. When CSGATE is set to BLK, this bit is set when the last Header update register TXFHDRn is written when CS is low or HDRCMT is already set. When CSGATE is set to UNBLK, this bit is set only when the header update register is written when HDRCMT is already set. This bit remains 0 otherwise. When set, this bit can be written to a value of 0 to clear. 0h = Header update is not ignored 1h = Header update is ignored
0	HDREN	R/W	0h	Header enable field. When CSGATE is set to BLK, this bit has to be set by software to enable atomic header feature. When CSGATE is set to UNBLK, this field is set automatically whenever a write to header update registers TXFHDRn occurs. 0h = Atomic header update feature disable 1h = Atomic header update feature enable

20.6.23 RXDATA Register (Offset = 140h) [Reset = 00000000h]

RXDATA is shown in Table 20-26.

Return to the Summary Table.

RXDATA Register. Reading this register returns first value in the RX FIFO. If the FIFO is empty the last read value is returned. Writing has no effect and is ignored.

Table 20-26. RXDATA Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	DATA	R	0h	Received Data. When read, the entry in the receive FIFO, pointed to by the current FIFO read pointer is accessed. As data values are read by the receive logic from the incoming data frame, they are placed into the entry in the receive FIFO, pointed to by the current RX FIFO write pointer. Received data less than 16 bits is automatically right-justified in the receive buffer.

20.6.24 TXDATA Register (Offset = 150h) [Reset = 00000000h]

TXDATA is shown in Table 20-27.

Return to the Summary Table.

TXDATA Register. Writing a value in this register puts the data into the TX FIFO. Reading this register returns the last writen value.

Table 20-27. TXDATA Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	RESERVED	R	0h	Reserved
15-0	DATA	R/W	0h	Transmit Data. When read, the last entry in the transmit FIFO, pointed to by the current FIFO write pointer is accessed. When written, the entry in the TX FIFO pointed to by the write pointer, is written to. Data values are read from the transmit FIFO by the transmit logic. It is loaded into the transmit serial shifter, then serially shifted out onto the output pin at the programmed bit rate. When a data size of less than 16 bits is selected, the user must right-justify data written to the transmit FIFO. The transmit logic ignores the unused bits.

20.6.25 STA Register (Offset = 160h) [Reset = 0000000Fh]

STA is shown in Table 20-28.

Return to the Summary Table.

Status Register

Table 20-28. STA Register Field Descriptions

Bit	Field	Type	Reset	Description
31-14	RESERVED	R	0h	Reserved
13-8	TXFIFOLVL	R	0h	Indicates how many locations of TXFIFO are currently filled with data
7	RESERVED	R	0h	Reserved
6	TXDONE	R/W	0h	Transmit done. Indicates whether the last bit has left the Shift register after a transmission 0h = Last bit has not yet left the Shift register, and the transmission is ongoing. 1h = Last bit has been shifted out, and the transmission is done
5	CSD	R/W	0h	Detection of CS deassertion in the middle of a data frame transmission results in this error being set. This feature is only available in the peripheral mode. 0h = No CS posedge is detected before the entire data frame has been transmitted. 1h = An error is generated when CS posedge (deassertion) is detected before the entire data frame is transmitted.
4	BUSY	R	0h	SPI Busy status 0h = SPI is in idle mode. 1h = SPI is currently transmitting and/or recieving data, or transmit FIFO is not empty.
3	RNF	R	1h	Receive FIFO not full status. 0h = Receive FIFO is full. 1h = Receive FIFO is not full.
2	RFE	R	1h	Receive FIFO empty status. 0h = Receive FIFO is not empty. 1h = Receive FIFO is empty.
1	TNF	R	1h	Transmit FIFO not full status. 0h = Transmit FIFO is full. 1h = Transmit FIFO is not full.
0	TFE	R	1h	Transmit FIFO empty status. 0h = Transmit FIFO is not empty. 1h = Transmit FIFO is empty.

Chapter 21

Inter-Integrated Circuit (I^2C)

This chapter describes the inter-integrated circuit (I^2C) interface.

21.1 Introduction

The I2C bus provides bidirectional data transfer through a 2-wire design, a serial data line (SDA) and a serial clock line (SCL), and interfaces to external I2C devices such as serial memory (RAM and ROM), networking devices, LCDs, tone generators, and so on. The I2C bus can also be used for system testing and diagnostic purposes in product development and manufacture.

The I2C module has following features:

- Devices on the I2C bus can be designated as either a controller or a target:
  - Supports both transmitting and receiving data as either a controller or a target
  - Supports simultaneous controller and target operation
- Four I2C modes:
  - Controller transmit
  - Controller receive
  - Target transmit
  - Target receive
- Two transmission speeds: standard (100kbps) and fast (400kbps)
- Controller and target interrupt generation:
  - Controller generates interrupts when a transmit or receive operation completes (or aborts due to an error).
  - Target generates interrupts when data has been transferred or requested by a controller or when a Start or Stop condition is detected.
- Controller with arbitration and clock synchronization, multicontroller support, and 7-bit addressing mode
- Glitch filter to suppress glitches lesser than 50ns on both SDA and SCL

21.2 Block Diagram

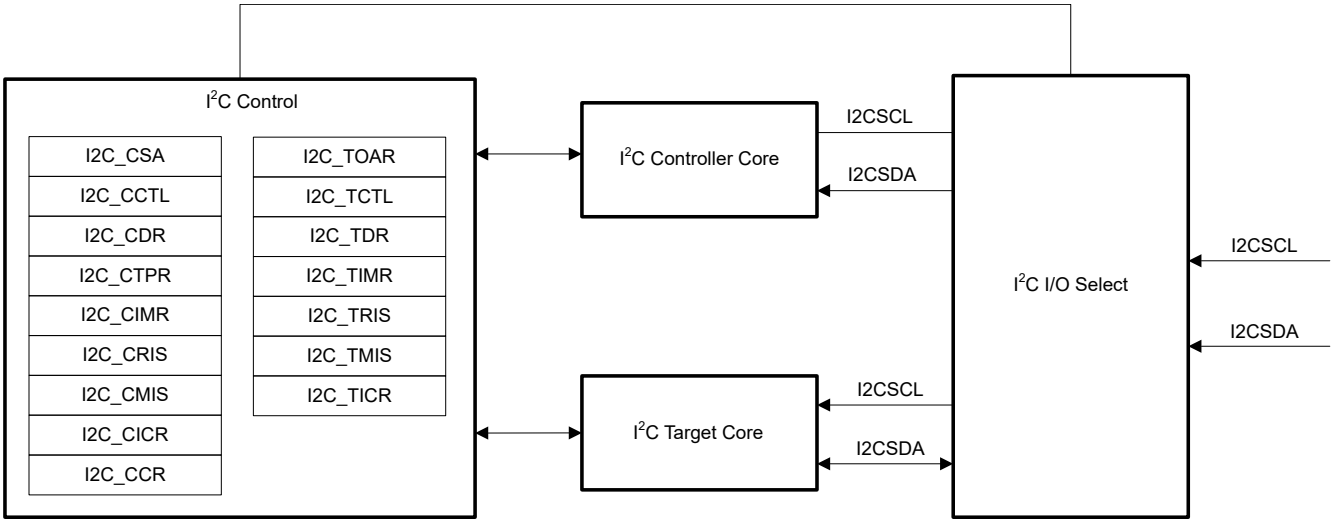


Figure 21-1. I2C Block Diagram

21.3 Functional Description

The I2C module is comprised of both controller and peripheral functions. For proper operation, the SDA pin must be configured as an open-drain signal. See Chapter 18 for more information on configuring pin functions. Figure 21-2 shows a typical I2C bus configuration.

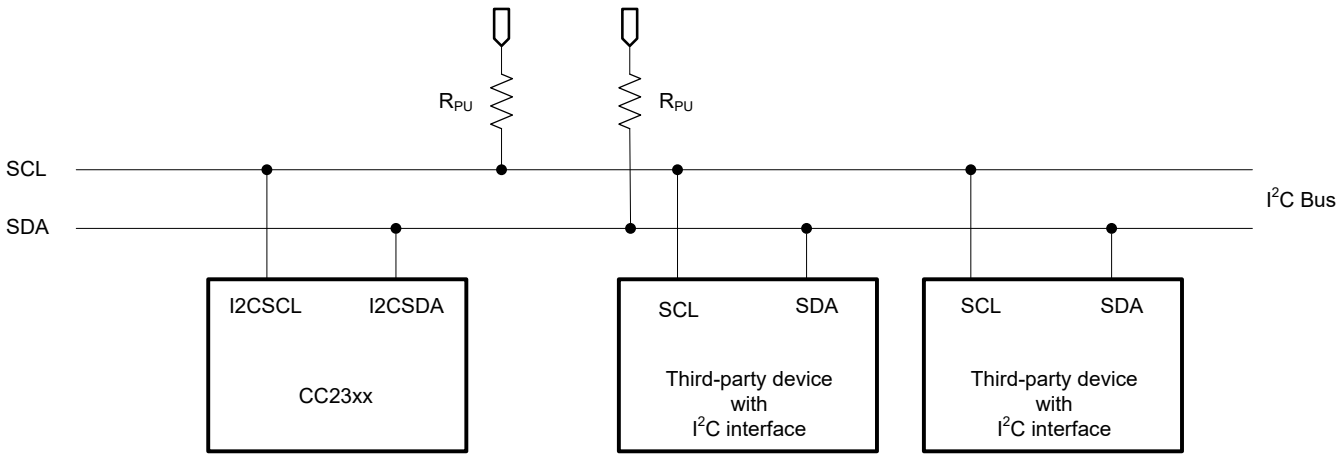


Figure 21-2. I2C Bus Configuration

21.3.1 Functional Overview

The I2C bus uses only two signals: SDA and SCL. SDA is the bidirectional serial data line and SCL line is the bidirectional serial clock line. The bus is considered idle when both lines are high.

Every transaction on the I2C bus is 9 bits long, consisting of 8 data bits and 1 acknowledge bit. A transfer is defined as the time between a valid start and stop condition, see Section 21.3.1.1. The number of bytes per transfer is unrestricted, an acknowledge bit must follow each byte, and data must be transferred by the MSB first. When a receiver cannot receive another complete byte, the receiver can hold the clock line SCL low and force the transmitter into a wait state. The data transfer continues when the receiver releases the clock SCL.

21.3.1.1 Start and Stop Conditions

The protocol of the I2C bus defines two states to begin and end a transaction: Start and Stop. A high-to-low transition on the SDA line while the SCL is high is defined as a Start condition, and a low-to-high transition on the SDA line while the SCL line is high is defined as a Stop condition. The bus is considered busy after a Start condition and free after a Stop condition (see Figure 21-3).

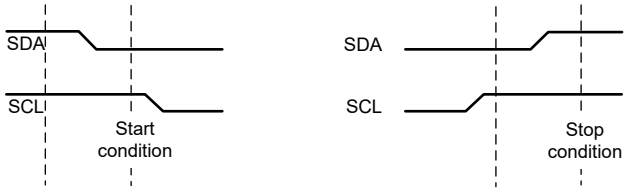


Figure 21-3. Start and Stop Conditions

The STOP bit determines if the cycle stops at the end of the data cycle or continues on to a Repeated Start condition. To generate a single transmit cycle, the I2C Controller Target Address I2C.CSA register is written with the desired address, the R/S bit is cleared, and the control register, I2C.CCTL, is written with ACK = X (0 or 1), STOP = 1, START = 1, and RUN = 1 to perform the operation and stop. When the operation is completed (or aborted due an error), the interrupt becomes active and the data is readable from the I2C Controller Data I2C.CDR register. When the I2C module operates in controller receiver mode, the ACK bit is normally set, causing the I2C bus controller to transmit an acknowledge automatically after each byte. When the I2C bus controller requires no further data transmission from the target transmitter, the ACK bit must be cleared.

When operating in target mode, two bits in the I2C Target Raw Interrupt Status I2C:TRIS register indicate detection of Start and Stop conditions on the bus, while two bits in the I2C Target Masked Interrupt Status I2C:TMIS register allow promotion of Start and Stop conditions to controller interrupts (when interrupts are enabled).

21.3.1.2 Data Format with 7-Bit Address

Data transfers follow the format shown in Figure 21-4. After the Start condition, a target address is transmitted. This address is 7 bits long followed by an eighth bit, which is a data direction bit (the RS bit in the I2C:CSA register). If the RS bit is clear, the operation is a transmit (send), and if the RS bit is set, the operation is a request for data (receive). A data transfer is always terminated by a Stop condition generated by the controller; however, a controller can initiate communications with another device on the bus, by generating a Repeated Start condition and addressing another target without first generating a Stop condition. Various combinations of receive and transmit formats are then possible within a single transfer.

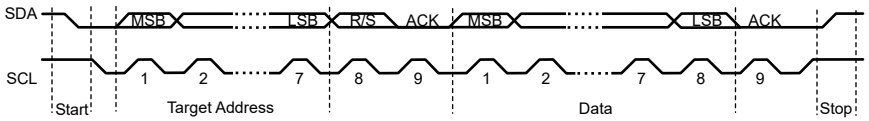


Figure 21-4. Complete Data Transfer with a 7-Bit Address

The first 7 bits of the first byte comprise the target address (see Figure 21-5). The eighth bit determines the direction of the message. A 0 in the R/S position of the first byte means that the controller transmits (sends) data to the selected target, and a 1 in this position means that the controller receives data from the target.

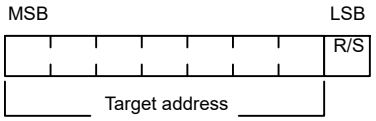


Figure 21-5. R/S Bit in First Byte

21.3.1.3 Data Validity

The SDA line must contain stable data during the high period of the clock, and the data line can change only when SCL is low (see Figure 21-6).

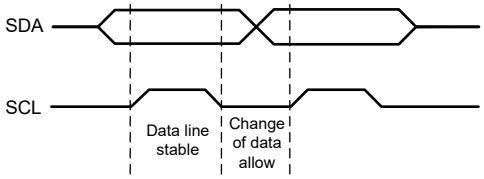


Figure 21-6. Data Validity During Bit Transfer on the I2C Bus

21.3.1.4 Acknowledge

All bus transactions have a required acknowledge clock cycle generated by the controller. During the acknowledge cycle, the transmitter (controller or target) releases the SDA line. To acknowledge the transaction, the receiver must pull down SDA during the acknowledge clock cycle. The data transmitted by the receiver during the acknowledge cycle must comply with the data validity requirements described in Section 21.3.1.3.

When a target receiver does not acknowledge the target address, the target must leave SDA high so that the controller can generate a Stop condition and abort the current transfer. If the controller device is acting as a receiver during a transfer, the controller is responsible for acknowledging each transfer made by the target. Because the controller controls the number of bytes in the transfer, the controller signals the end of data to the target transmitter by not generating an acknowledge on the last data byte. The target transmitter must then release SDA to let the controller generate a Stop or a Repeated Start condition.

21.3.1.5 Arbitration

A controller can start a transfer only if the bus is idle. However in some cases, two or more controllers can generate a Start condition at a similar time. During the transfer, all controllers constantly monitor the SCL and SDA lines. All controllers that detect SDA as low when the controllers expect SDA to be high (as driven by them)

acknowledge that another controller has won the arbitration and immediately stop transfers, and retire until the bus is idle again.

Arbitration can occur over several bits. The first stage of arbitration is a comparison of address bits; if both controllers are trying to address the same device, arbitration continues to the comparison of data bits.

21.3.2 Available Speed Modes

The I2C bus can run in either standard mode (100 kbps) or fast mode (400 kbps). The selected mode must match the speed of the other I2C devices on the bus.

Standard and Fast Modes

Standard and fast modes are selected using a value in the I2C Controller Timer Period I2C.CTPR register that results in an SCL frequency of 100 kbps for standard mode, or 400 kbps for fast mode.

The I2C clock rate is determined by the parameters CLK_PRD, TPR, SCL_LP, and SCL_HP where:

- CLK_PRD is the CLKSVT period.
- TPR is the programmed value in the I2C.CTPR register.
- SCL_LP is the low phase of SCL (fixed at 6).
- SCL_HP is the high phase of SCL (fixed at 4). The I2C clock period is calculated as follows:

$$SCL\_PERIOD = 2 \times (1 + TPR) \times (SCL\_LP + SCL\_HP) \times CLK\_PRD \quad (11)$$

21.3.3 Interrupts

The I2C can generate interrupts when the following conditions are observed:

- Controller transaction completed
- Controller arbitration lost
- Controller transaction error
- Controller bus time-out
- Target transaction received
- Target transaction requested
- Stop condition on bus detected
- Start condition on bus detected

The I2C controller and I2C target modules have separate interrupt signals. While both modules can generate interrupts for multiple conditions, the signals are bitwise ORed together so only a single interrupt signal is sent to the MCU event fabric.

21.3.3.1 I2C Controller Interrupts

The I2C controller module generates an interrupt when a transaction completes (either transmit or receive), when arbitration is lost, or when an error occurs during a transaction. To enable the I2C controller interrupt, software must set the IM bit in the I2C Controller Interrupt Mask register, I2C.CIMR. When an interrupt condition is met, software must check the I2C.CSTAT[4] ARBLST and I2C.CSTAT[1] ERR bits to verify that an error did not occur during the last transaction, and to check that arbitration has not been lost. An error condition is asserted if the last transaction was not acknowledged by the target. If an error is not detected and the controller has not lost arbitration, the application can proceed with the transfer. The interrupt is cleared by setting the IC bit in the I2C Controller Interrupt Clear register (I2C:CICR) to 1.

If the application does not require the use of interrupts, the raw interrupt status is always visible through the I2C Controller Raw Interrupt Status register (I2C:CRIS).

21.3.3.2 I2C Target Interrupts

The target module can generate an interrupt when data is received or requested. This interrupt is enabled by setting the I2C Target Interrupt Mask register (I2C.TIMR). Software determines whether the module must write (transmit) or read (receive) data from the I2C Target Data register, I2C.TDR[7:0] DATA bit field and by checking

the I2C.TSTAT[0] RREQ and I2C.TSTAT[1] TREQ bits. If the target module is in receive mode and the first byte of a transfer is received, the I2C.TSTAT[2] FBR and I2C.TSTAT[0] RREQ bits are set. The interrupt is cleared by setting the I2C Target Interrupt Clear register I2C.TICR[0] DATAIC bit.

In addition, the target module generates an interrupt when a Start and a Stop condition is detected. These interrupts are enabled by setting the I2C.TIMR[1] STARTIM and I2C.TIMR[2] STOPIM bits; these interrupts are cleared by setting the I2C.TICR[1] STARTIC and I2C.TICR[2] STOPIC bits to 1.

If the application does not require the use of interrupts, the raw interrupt status is always visible through the I2C Target Raw Interrupt Status register (I2C.TRIS).

21.3.4 Loopback Operation

The I2C modules can be placed into an internal-loopback mode for diagnostic or debug purposes by setting the I2C Controller Configuration register I2C.CCR[0] LPBK bit. In loopback mode, the SDA and SCL signals from the controller and target modules are tied together.

21.3.5 Command Sequence Flow Charts

This section details the steps required to perform the various I^2C transfer types in both the controller and target modes. To do this, the SDA and SCL signal configuration must be done in the IOC.IOC ***n*** registers.

21.3.5.1 I^2C Controller Command Sequences

Figure 21-7 through Figure 21-12 show the command sequences available for the I^2C controller.

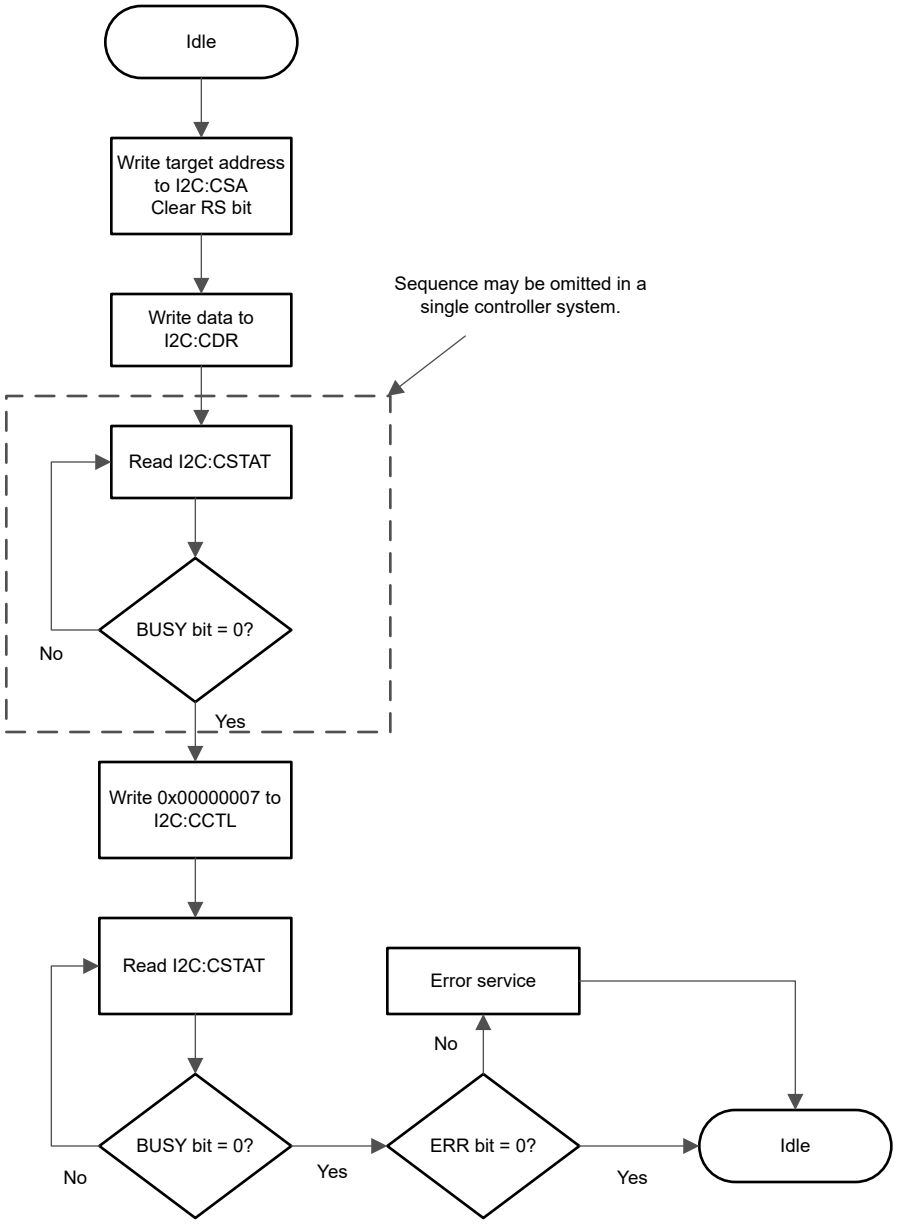


Figure 21-7. Controller Single Transmit

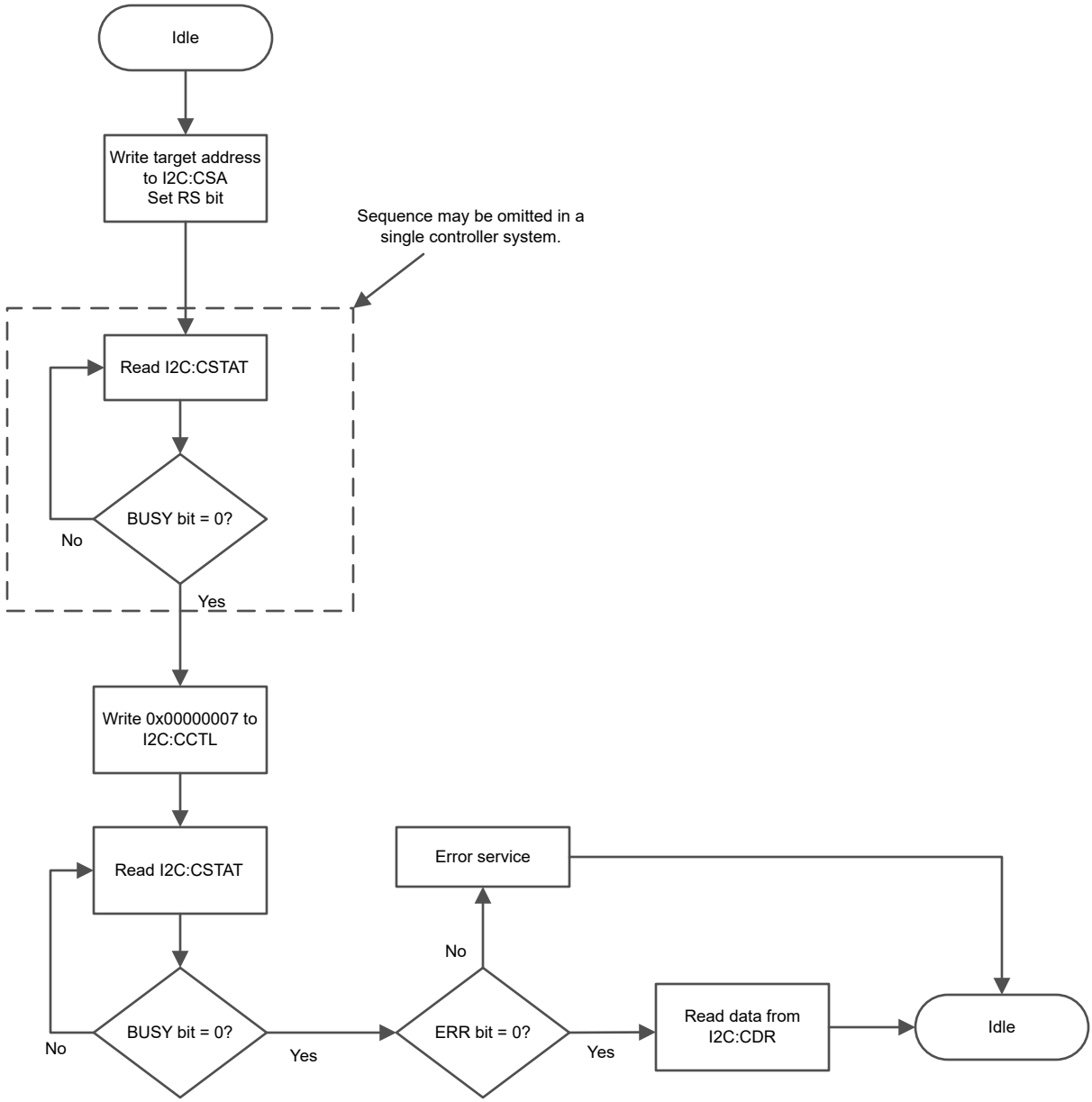


Figure 21-8. Controller Single Receive

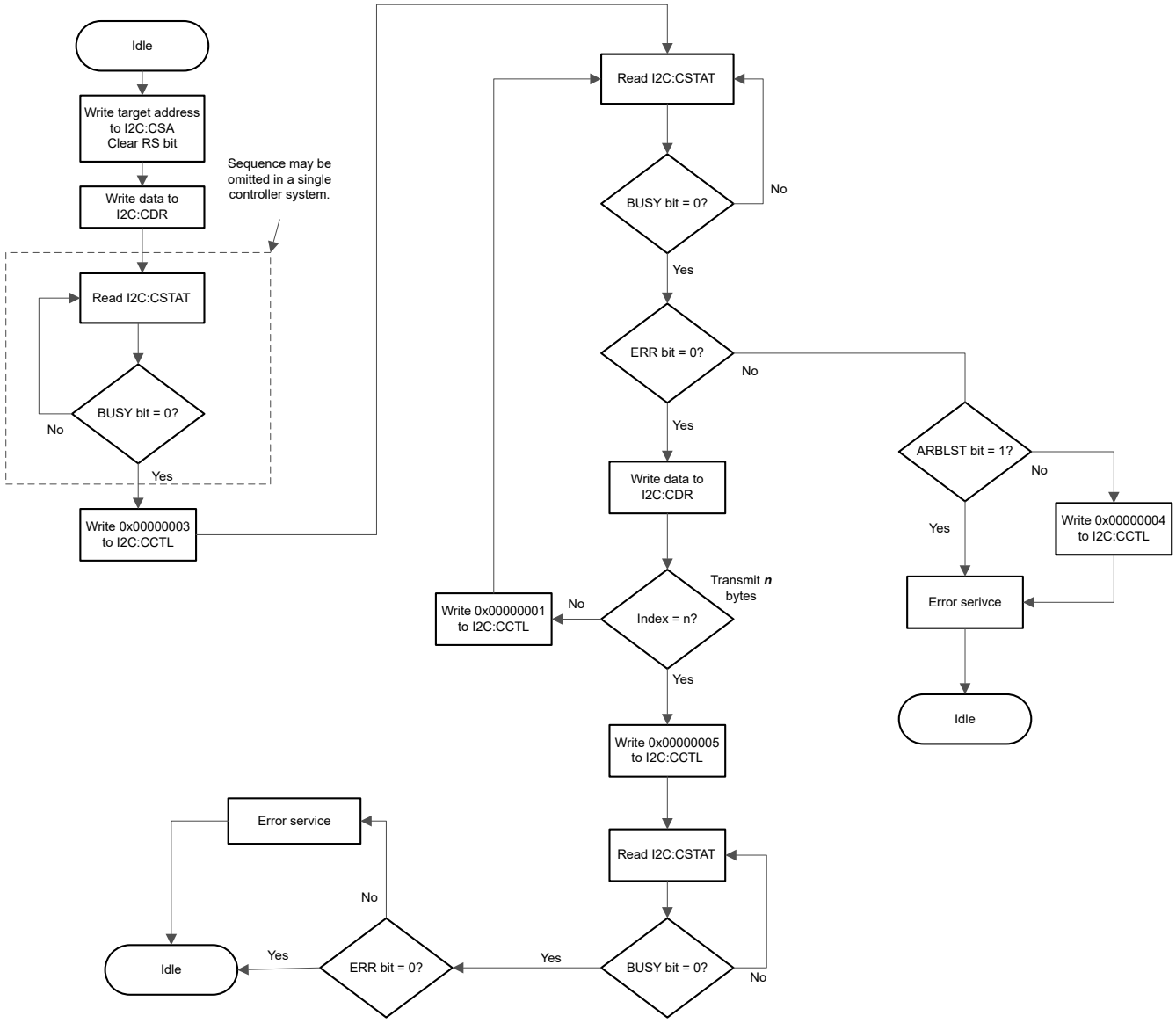


Figure 21-9. Controller Transmit with Repeated Start Condition

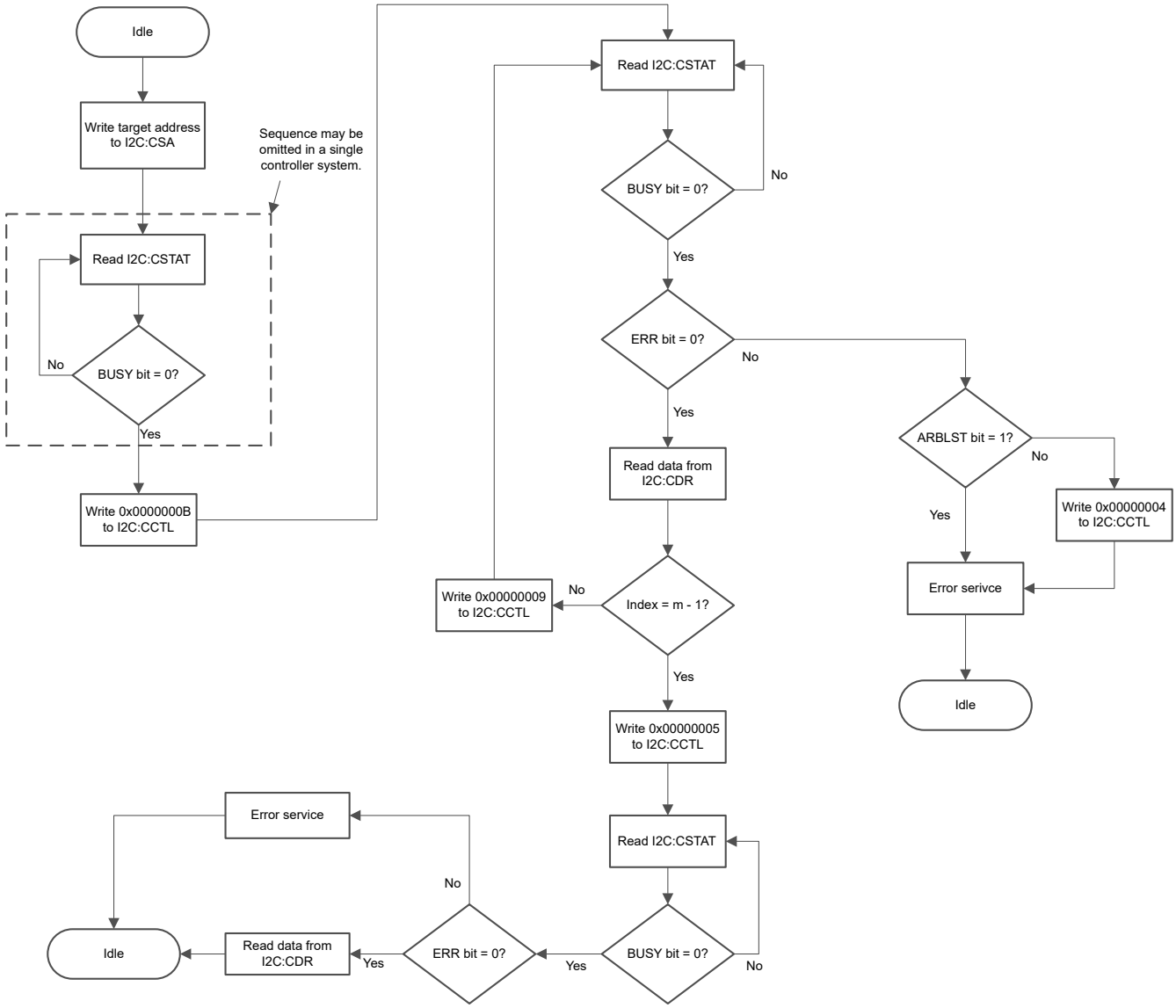


Figure 21-10. Controller Receive with Repeated Start Condition

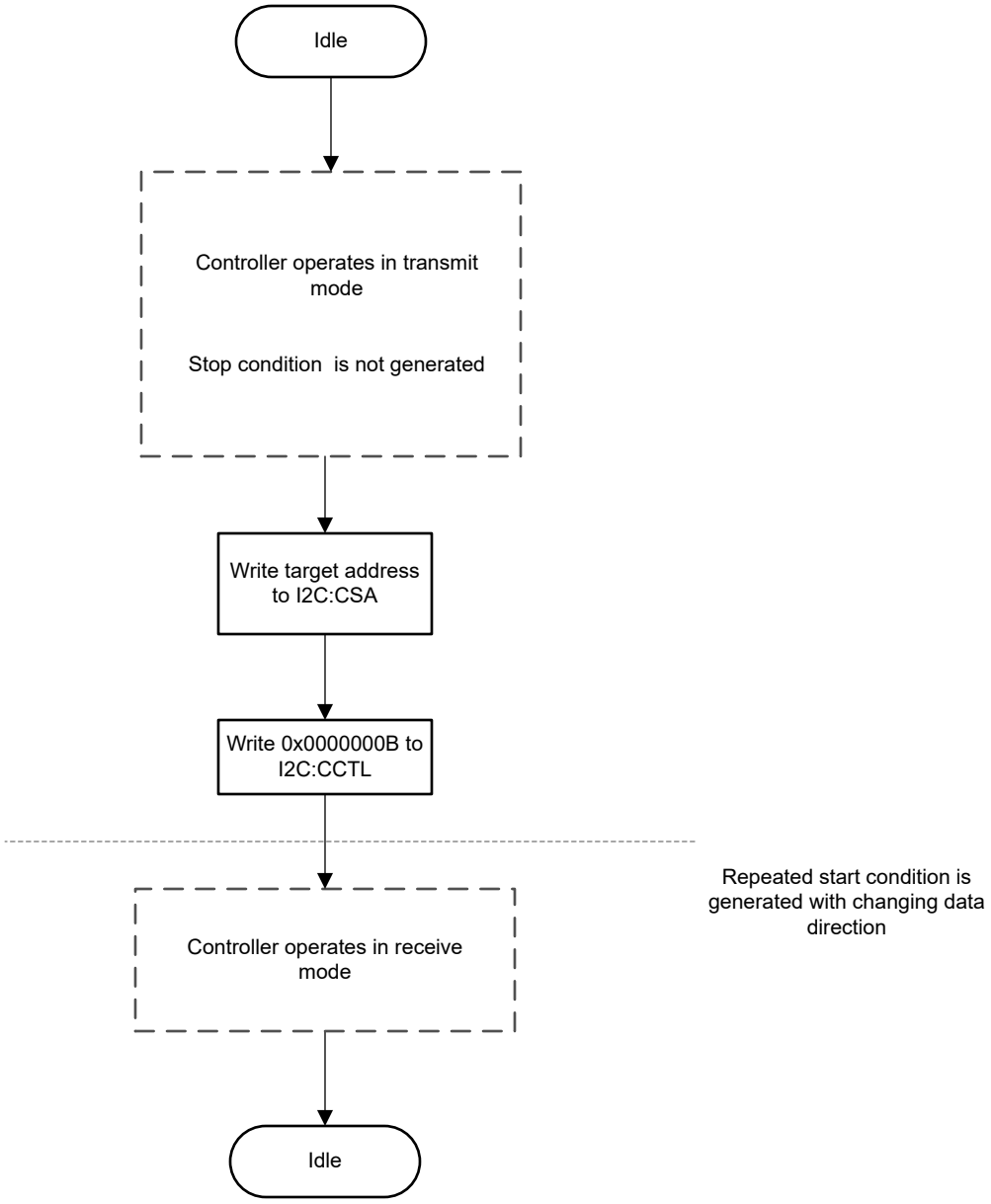


Figure 21-11. Controller Receive with Repeated Start after Transmit with Repeated Start Condition

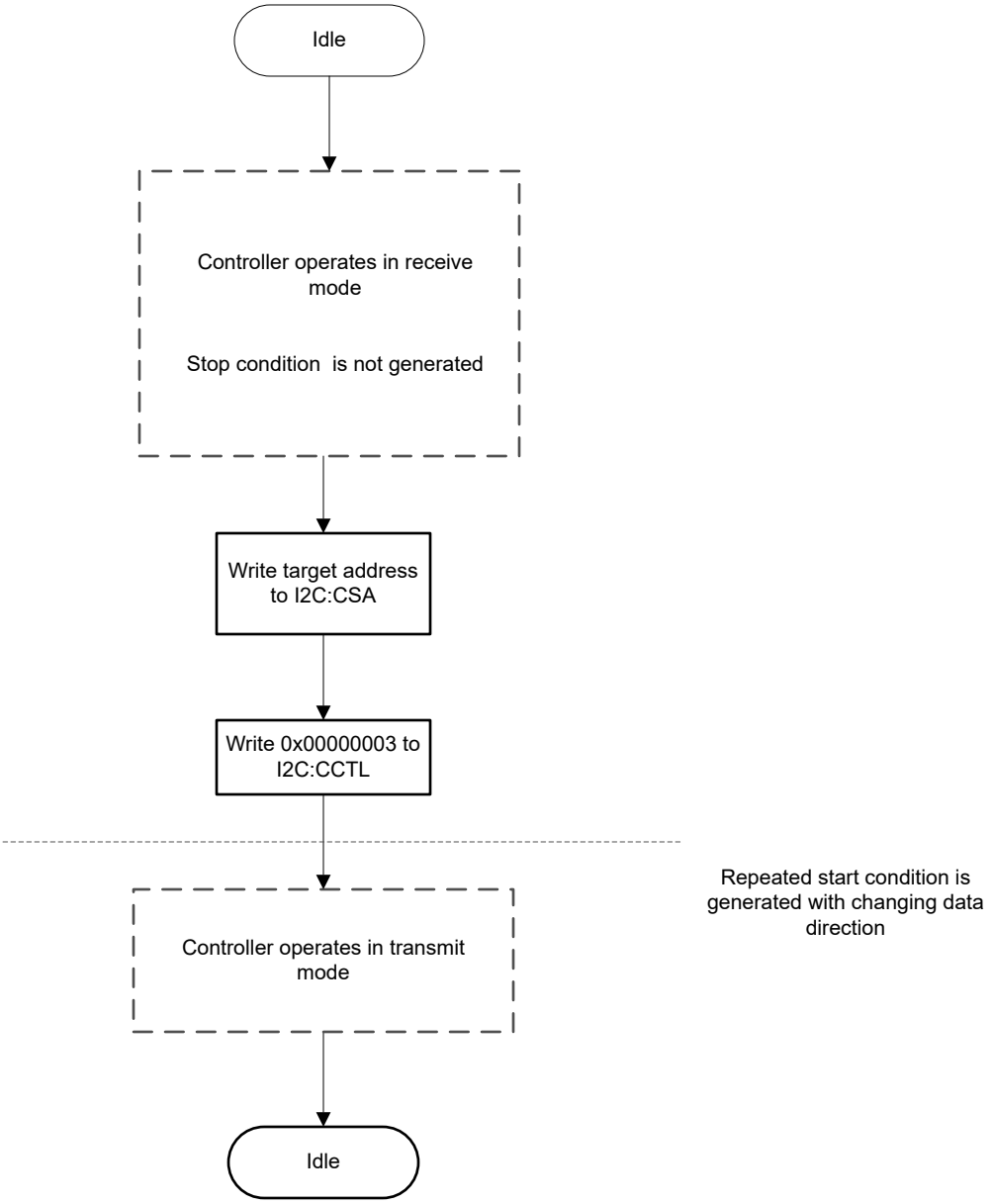


Figure 21-12. Controller Transmit with Repeated Start after Receive with Repeated Start Condition

21.3.5.2 I2C Target Command Sequences

Figure 21-13 shows the command sequence available for the I2C target.

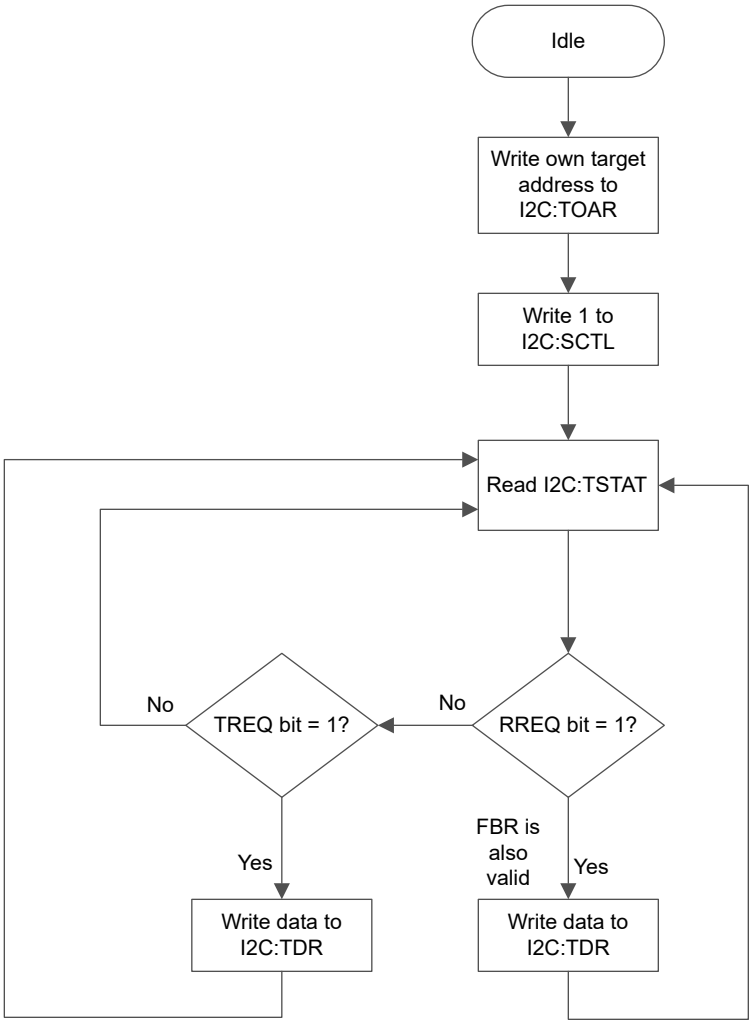


Figure 21-13. Target Command Sequence

21.4 Initialization and Configuration

The following example shows how to configure the I2C module to transmit a single byte as a controller, assuming that the system clock is 48MHz.

- Enable the serial power domain and enable the I2C module in CLKCTL by setting the CLKCTL.CLKENSET0[6] I2C0 bit.
- Configure the IOC module to route the SDA and SCL signals from I/Os to the I2C and configure the pins for open-drain. See Chapter 18 for more information.
- Initialize the I2C controller by writing the I2C:CCR register with a value of 0x0010
- Set the desired SCL clock speed of 100kbps by writing the I2C.CTPR register with the correct value. The value written to the I2C.CTPR register represents the number of system clock periods in one SCL clock period. The TPR value is determined by Equation 12,Equation 13, and Equation 14.

$$TPR = [CLKSVT / (2 \times (SCL\_LP + SCL\_HP) \times SCL\_FREQ)] - 1 \quad (12)$$

$$TPR = [48MHz / (2 \times (6 + 4) \times 100000)] - 1 \quad (13)$$

$$TPR = 23 \quad (14)$$

Write the I2C:CTPR register with the value of 0x0000 0017.

- Specify the target address of the controller and that the next operation is a transmit by writing the I2C.CSA register with a value of 0x0000 0076, which sets the target address to 0x3B.
- Place data (byte) to be transmitted in the data register by writing the I2C:CDR register with the desired data.
- Initiate a single-byte transmit of the data from controller to target by writing the I2C.CCTL register with a value of 0x0000 0007 (Stop, Start, Run).
- Wait until the transmission completes by polling the I2C.CSTAT[6] BUSBSY bit until the bit is cleared.
- Check the I2C.CSTAT[1] ERR bit to confirm the transmit was acknowledged.

21.5 I2C Registers

Table 21-1 lists the memory-mapped registers for the I2C registers. All register offset addresses not listed in Table 21-1 should be considered as reserved locations and the register contents should not be modified.

Table 21-1. I2C Registers

Offset	Acronym	Register Name	Section
0h	TOAR	Target Own Address	Go
4h	TSTA	Target Control and Status	Go
4h	TCTL	Target control	Go
8h	TDR	Target Data	Go
Ch	TIMR	Target Interrupt Mask	Go
10h	TRIS	Target Raw Interrupt Status	Go
14h	TMIS	Target Masked Interrupt Status	Go
18h	TICR	Target Interrupt Clear	Go
800h	CTA	Controller Target Address	Go
804h	CSTA	Controller Control and Status	Go
804h	CCTL	Controller control	Go
808h	CDR	Controller Data	Go
80Ch	CTPR	Controller Timer Period	Go
810h	CIMR	Controller Interrupt Mask	Go
814h	CRIS	Controller Raw Interrupt Status	Go
818h	CMIS	Controller Masked Interrupt Status	Go
81Ch	CICR	Controller Interrupt Clear	Go
820h	CCR	Controller Configuration	Go

Complex bit access types are encoded to fit into small table cells. Table 21-2 shows the codes that are used for access types in this section.

Table 21-2. I2C Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

21.5.1 TOAR Register (Offset = 0h) [Reset = 00000000h]

TOAR is shown in Table 21-3.

Return to the Summary Table.

Target Own Address
This register consists of seven address bits that identify this I2C device on the I2C bus.

Table 21-3. TOAR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-7	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior.
6-0	OAR	R/W	0h	Target own address. This field specifies bits a6 through a0 of the target address.

21.5.2 TSTA Register (Offset = 4h) [Reset = 00000000h]

TSTA is shown in Table 21-4.

Return to the Summary Table.

Target status
This register functions as a status register of the target.

Table 21-4. TSTA Register Field Descriptions

Bit	Field	Type	Reset	Description
31-3	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior.
2	FBR	R	0h	First byte received. This bit is only applicable when the TSTA.RREQ bit is set and is automatically cleared when data has been read from the TDR register. Note: This bit is not used for target transmit operations. 0h = The first byte following the target's own address has not been received 1h = The first byte following the target's own address has been received.
1	TREQ	R	0h	This field reflects the transmit request status 0h = No outstanding transmit request 1h = The I2C has been addressed as a target transmitter and is using clock stretching to delay the controller until data has been written to the TDR register.
0	RREQ	R	0h	This field reflects the receive request status. 0h = No outstanding receive data 1h = The target has outstanding receive data from the external controller and is using clock stretching to delay the controller until data has been read from the TDR register.

21.5.3 TCTL Register (Offset = 4h) [Reset = 00000000h]

TCTL is shown in Table 21-5.

Return to the Summary Table.

Target control
This registers functions as a target control register.

Table 21-5. TCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior.
0	DA	W	0h	This field sets the device active control 0h = Disable the target operation 1h = Enable the target operation

21.5.4 TDR Register (Offset = 8h) [Reset = 00000000h]

TDR is shown in Table 21-6.

Return to the Summary Table.

Target data register
This register contains the data to be transmitted when in the target transmit state, and the data received when in the target receive state.

Table 21-6. TDR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior.
7-0	DATA	R/W	0h	Data for transfer. This field contains the data for transfer during a target receive or a transmit operation. When written, the register data is used as transmit data. When read, this register returns the last data received. Data is stored until next update, either by a system write to the controller for transmit or by an external controller to the target for receive.

21.5.5 TIMR Register (Offset = Ch) [Reset = 00000000h]

TIMR is shown in Table 21-7.

Return to the Summary Table.

Target interrupt mask
This register controls whether a raw interrupt is promoted to a controller interrupt.

Table 21-7. TIMR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-3	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior.
2	STOPIM	R/W	0h	Stop condition interrupt mask 0h = Disable interrupt mask 1h = Enable interrupt mask
1	STARTIM	R/W	0h	Start condition interrupt mask 0h = Disable interrupt mask 1h = Enable interrupt mask
0	DATAIM	R/W	0h	Data interrupt mask 0h = Disable interrupt mask 1h = Enable interrupt mask

21.5.6 TRIS Register (Offset = 10h) [Reset = 00000000h]

TRIS is shown in Table 21-8.

Return to the Summary Table.

Target raw interrupt status
This register shows the unmasked interrupt status.

Table 21-8. TRIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-3	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior.
2	STOPRIS	R	0h	Stop condition raw interrupt status This bit is cleared by writing a 1 to TICR.STOPIC. 0h = Interrupt did not occur 1h = Interrupt occured
1	STARTRIS	R	0h	Start condition raw interrupt status This bit is cleared by writing a 1 to TICR.STARTIC. 0h = Interrupt did not occur 1h = Interrupt occured
0	DATARIS	R	0h	Data raw interrupt status This bit is cleared by writing a 1 to TICR.DATAIC. 0h = Interrupt did not occur 1h = Interrupt occured

21.5.7 TMIS Register (Offset = 14h) [Reset = 00000000h]

TMIS is shown in Table 21-9.

Return to the Summary Table.

Target Masked Interrupt Status
This register shows which interrupt is active (based on result from TRIS and TIMR registers).

Table 21-9. TMIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-3	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior.
2	STOPMIS	R	0h	Stop condition masked interrupt status This bit is cleared by writing a 1 to TICR.STOPIC. 0h = Masked interrupt did not occur 1h = Masked interrupt occured
1	STARTMIS	R	0h	Start condition masked interrupt status This bit is cleared by writing a 1 to TICR.STARTIC. 0h = Masked interrupt did not occur 1h = Masked interrput occured
0	DATAMIS	R	0h	Start condition masked interrupt status This bit is cleared by writing a 1 to TICR.DATAIC. 0h = Masked interrupt did not occur 1h = Masked interrupt occured

21.5.8 TICR Register (Offset = 18h) [Reset = 00000000h]

TICR is shown in Table 21-10.

Return to the Summary Table.

Target Interrupt Clear
This register clears the raw interrupt TRIS.

Table 21-10. TICR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-3	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior.
2	STOPIC	W	0h	Stop condition interrupt clear 0h = No effect 1h = Clear interrupt Writing 1 to this bit clears TRIS.STOPRIS and TMIS.STOPMIS
1	STARTIC	W	0h	Start condition interrupt clear 0h = No effect 1h = Clear interrupt Writing 1 to this bit clears TRIS.STARTRIS and TMIS.STARTMIS
0	DATAIC	W	0h	Data interrupt clear 0h = No effect 1h = Clear interrupt Writing 1 to this bit clears TRIS.DATARIS and TMIS.DATAMIS

21.5.9 CTA Register (Offset = 800h) [Reset = 00000000h]

CTA is shown in Table 21-11.

Return to the Summary Table.

Controller target address
This register contains seven address bits of the target to be accessed by the controller (a6-a0), and an CTA.RS bit determining if the next operation is a receive or transmit.

Table 21-11. CTA Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior.
7-1	SA	R/W	0h	Controller target address Defines which target is addressed for the transaction in controller mode
0	RS	R/W	0h	Receive or Send This bit-field specifies the next operation with addressed target CTA.SA. 0h = Transmit/send data to target 1h = Receive data from target

21.5.10 CSTA Register (Offset = 804h) [Reset = 00000020h]

CSTA is shown in Table 21-12.

Return to the Summary Table.

Controller status
This register functions as a controller status register.

Table 21-12. CSTA Register Field Descriptions

Bit	Field	Type	Reset	Description
31-7	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior.
6	BUSBSY	R	0h	Bus busy Note:The bit changes based on the CCTRL.START and CCTRL.STOP conditions. 0h = The bus is idle. 1h = The bus is busy.
5	IDLE	R	1h	This field specifies whether I2C is idle or not 0h = The controller is not idle. 1h = The controller is idle.
4	ARBLST	R	0h	The filed specifies the arbitration status 0h = The controller won arbitration. 1h = The controller lost arbitration.
3	DATACKN	R	0h	This field contains Data acknowledge status 0h = The transmitted data was acknowledged. 1h = The transmitted data was not acknowledged.
2	ADRACKN	R	0h	This field reflects the address acknowledge status 0h = The transmitted address was acknowledged. 1h = The transmitted address was not acknowledged.
1	ERR	R	0h	This field reflects the error status 0h = No error was detected on the last operation 1h = An error occurred with the last operation
0	BUSY	R	0h	This field reflects the I2C busy status Note: The I2C controller requires four CLKSVT clock cycles to assert the BUSY status after I2C controller operation has been initiated through a write into CCTL register. Hence after programming CCTL register, application is requested to wait for four CLKSVT clock cycles before issuing a controller status inquiry through a read from CSTA register. Any prior inquiry would result in wrong status being reported. 0h = The controller is idle. 1h = The controller is busy.

21.5.11 CCTL Register (Offset = 804h) [Reset = 00000000h]

CCTL is shown in Table 21-13.

Return to the Summary Table.

Controller control
This register functions as a controller control register.

Table 21-13. CCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior.
3	ACK	W	0h	This field is to enable the data acknowledge. Note:This bit-field must be cleared when the I2C bus controller requires no further data to be transmitted from the target transmitter. 0h = The received data byte is not acknowledged automatically by the controller. 1h = The received data byte is acknowledged automatically by the controller.
2	STOP	W	0h	This field is to set stop condition . Note: This bit-field determines if the cycle stops at the end of the data cycle or continues on to a repeated start condition. 0h = The controller does not generate the stop condition. 1h = The controller generates the stop condition.
1	START	W	0h	This field is to set start or repeated start condition. 0h = The controller does not generate the start condition 1h = The controller generates the start condition.
0	RUN	W	0h	This field is to set the controller enable. 0h = The controller is disabled. 1h = The controller is enabled to transmit or receive data.

21.5.12 CDR Register (Offset = 808h) [Reset = 00000000h]

CDR is shown in Table 21-14.

Return to the Summary Table.

Controller data
This register contains the data to be transmitted when in the controller transmit state and the data received when in the controller receive state.

Table 21-14. CDR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior.
7-0	DATA	R/W	0h	When Read: Last RX Data is returned When Written: Data is transferred during TX transaction.

21.5.13 CTPR Register (Offset = 80Ch) [Reset = 00000001h]

CTPR is shown in Table 21-15.

Return to the Summary Table.

Controller timer period
This register specifies the period of the SCL clock.

Table 21-15. CTPR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior.
7	TPR_7	R/W	0h	Must be set to 0 to set CTPR.TPR. If set to 1, a write to CTPR.TPR will be ignored.
6-0	TPR	R/W	1h	SCL clock period This field specifies the period of the SCL clock. SCL_PRD = 2*(1+TPR)*(SCL_LP + SCL_HP)*CLK_PRD, where: SCL_PRD is the SCL line period (I2C clock). TPR is the timer period register value (range of 1 to 127) SCL_LP is the SCL low period (fixed at 6). SCL_HP is the SCL high period (fixed at 4). CLK_PRD is the CLKSVT period in ns.

21.5.14 CIMR Register (Offset = 810h) [Reset = 00000000h]

CIMR is shown in Table 21-16.

Return to the Summary Table.

Controller interrupt mask
This register controls whether a raw interrupt is promoted to a controller interrupt.

Table 21-16. CIMR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior.
0	IM	R/W	0h	Interrupt mask 0h = Disable interrupt mask 1h = Enable interrupt mask

21.5.15 CRIS Register (Offset = 814h) [Reset = 00000000h]

CRIS is shown in Table 21-17.

Return to the Summary Table.

Controller raw interrupt status
This register shows the unmasked interrupt status.

Table 21-17. CRIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior.
0	RIS	R	0h	Raw interrupt status This bit is cleared by writing 1 to CICR.IC bit. 0h = Interrupt did not occur 1h = Interrupt occured

21.5.16 CMIS Register (Offset = 818h) [Reset = 00000000h]

CMIS is shown in Table 21-18.

Return to the Summary Table.

Controller masked interrupt status
This register shows which interrupt is active (based on result from CRIS and CIMR registers).

Table 21-18. CMIS Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior.
0	MIS	R	0h	Masked interrupt status This bit is cleared by writing 1 to CICR.IC bit. 0h = Masked interrupt did not occur 1h = Masked interrupt occured

21.5.17 CICR Register (Offset = 81Ch) [Reset = 00000000h]

CICR is shown in Table 21-19.

Return to the Summary Table.

Controller interrupt clear
This register clears the raw and masked interrupt.

Table 21-19. CICR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-1	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior.
0	IC	W	0h	Interrupt clear 0h = No effect 1h = Clear Interrupt Writing 1 to this bit clears CRIS.RIS and CMIS.MIS.

21.5.18 CCR Register (Offset = 820h) [Reset = 00000000h]

CCR is shown in Table 21-20.

Return to the Summary Table.

Controller Configuration
This register configures the mode (Controller or Target) and sets the interface for test mode loopback.

Table 21-20. CCR Register Field Descriptions

Bit	Field	Type	Reset	Description
31-6	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior.
5	TFE	R/W	0h	I2C target function enable 0h = Target mode disabled 1h = Target mode enabled
4	CFE	R/W	0h	I2C controller function enable 0h = Controller mode disabled 1h = Controller mode enabled
3-1	RESERVED	R	0h	Software should not rely on the value of a reserved. Writing any other value than the reset value may result in undefined behavior.
0	LPBK	R/W	0h	I2C loopback 0h = Test mode (Loopback operation) disabled 1h = Test mode (Loopback operation) enabled

Chapter 22
Radio

This section describes the operation of the 2.4GHz radio.

22.1 Introduction

The 2.4GHz radio supports several formats. The radio provides significant baseband automation such as analog radio control, modulation, demodulation, address checking, CRC-calculation and checking. The customer application interfaces with the radio through RCL (Radio Control Layer), which is a software layer provided in the CC23xx Software Development Kit (SDK), and documented in the SDK. Alternatively, the customer application interfaces with the TI provided stack (Bluetooth® Low Energy or IEEE), and the stack then uses the RCL to interface with the radio. The documentation provided here is to give an overview of the radio structure and usage model.

22.2 Block Diagram

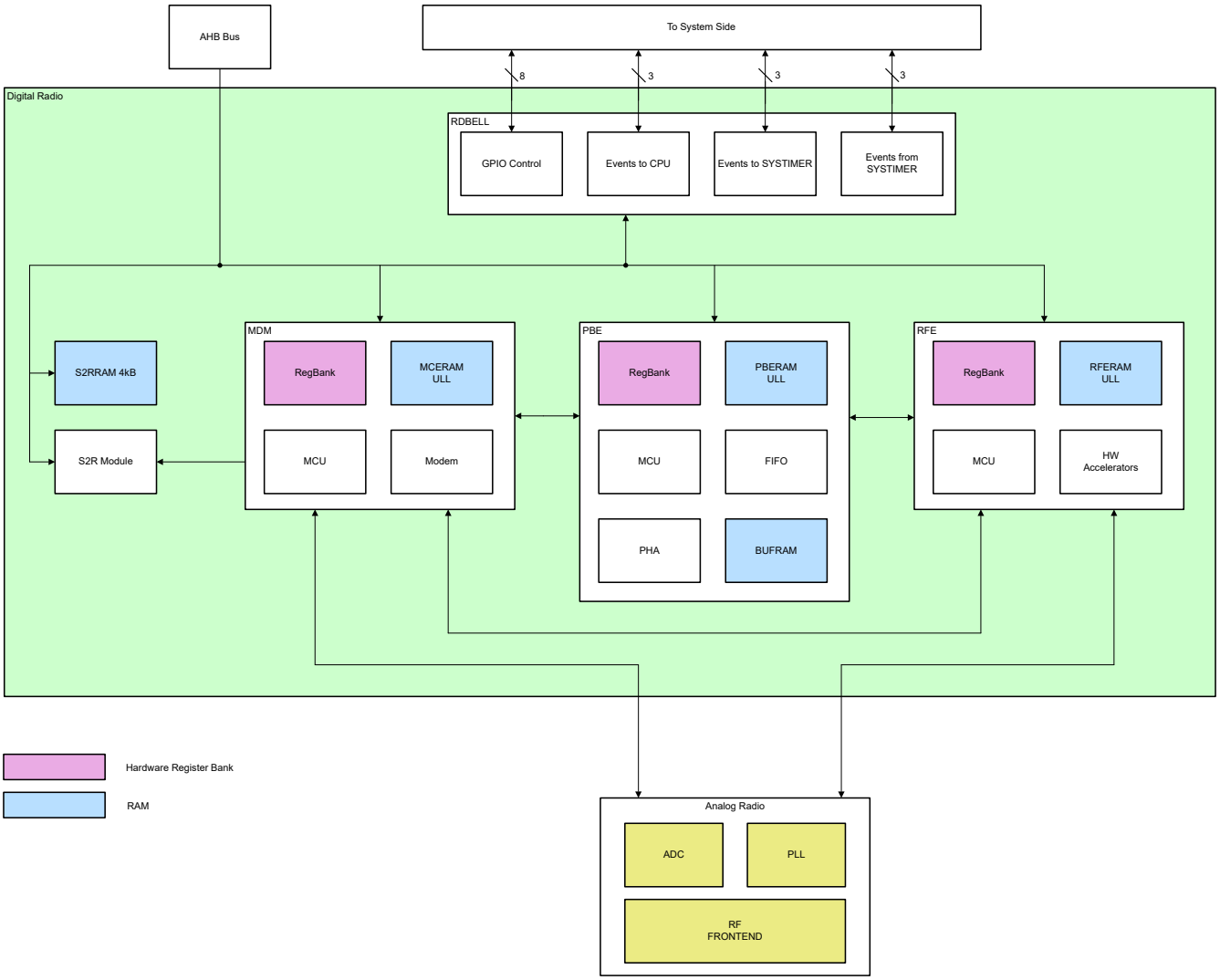


Figure 22-1. Radio Block Diagram

22.3 Overview

22.3.1 Radio Sub-Domains

The digital part of the radio consists of three main sub-domains: Modem (MDM), Packet Build Engine (PBE), and RF Engine (RFE). Each of these sub-domains have their own internal processor and program memories for storing FW images (MCERAM, PBERAM, RFERAM). They also have separate register blocks for configuration of HW and various HW accelerators. The different sub domains have different responsibilities:

Modem (MDM)

The MDM domain is responsible for:

- Conversion from information/data bits to TX-symbols including:
  - Preamble and sync word insertion
  - Direct Sequence Spread Spectrum (DSSS)
  - Forward Error Correction (FEC) encoding
- Demodulation including:
  - Timing recovery
  - Frequency offset estimation/compensation
  - Rate offset estimation/compensation
  - FEC decoding

Packet build engine (PBE)

The PBE domain is responsible for:

- Packet timing
- Composition and decomposition of radio packets
- FIFO handling
- Packet Handling Accelerator (PHA) for CRC and whitening
- Address checking

RF Engine (RFE)

The RFE domain is responsible for:

- Sequencing of analog modules
- PLL calibration algorithm
- Modulation/Shaping of the transmitted data
- Received Signal Strength Indication (RSSI) estimation
- Automatic Gain Control (AGC) algorithms

22.3.2 Radio RAMs

The RAMs marked with ULL in Figure 22-1 , which are used for storing firmware images, have ultra-low leakage (ULL). ULL supports low power consumption while retaining the contents of these memories when entering standby. This avoids having to reprogram the firmware images after each standby. The other memories (S2RRAM, BUFRAM) have higher leakage and the power consumption of having retention on these in standby is higher. The BUFRAM is used for data that usually have a short lifespan, so retaining the contents of these is not necessary. The S2RRAM is generally left unused.

Note

If not used by the drivers and stacks provided by TI, the S2RRAM can be leveraged by the application. Consult the User's Guide of the stack used to verify whether the S2RRAM is available. The User's Guide also contains guidance on the way to use the S2RRAM, configure the build environment and on the precautions to take when using the S2RRAM.

22.3.3 Doorbell (DBELL)

The DBELL module has clock control registers, GPIO signals to and from the radio, and also controls events to and from the system side. This chapter gives an overview, see the LRFDDBELL register descriptions for details.

22.3.3.1 Interrupts

The radio has three interrupts on the system side. Each of the interrupts has the same 32 possible sources internally in the radio. Settings in the interrupt mask registers LRFDDBELL.IMASK ***n*** are used to configure the

radio internal events to be the trigger sources for the three individual interrupts. The raw state of the internal interrupt sources can be read from LRFDDBELL.RIS ***n*** , and the masked interrupt source status can be read from LRFDDBELL.MIS ***n*** . If any of the bits in the LRFDDBELL.MIS ***n*** registers are set then the corresponding interrupt line to the system side is triggered. To clear the interrupt source, set the corresponding bit in the LRFDDBELL.ICLR ***n*** register. An internal interrupt source flag can be set by setting the corresponding bit in the LRFDDBELL:ISET ***n*** register.

22.3.3.2 GPIO Control

The radio has eight general purpose outputs (GPO) and 8 general purpose inputs (GPI). Each of the eight individual output lines can be independently configured to output various internal signals. The selection of output source is done in the LRFDDBELL:GPOSEL0 and LRFDDBELL:GPOSEL1 registers. The GPO signals originating in MDM, PBE, and RFE are all fully firmware-driven. Different use cases have different features mapped to the GPO. Uses for the GPO include control of external switches, LNAs, PAs, and handshaking with other devices. Any control of GPO is documented in the SDK.

The GPI lines are all routed to the MDM, PBE, and RFE for use by the processors in these sub-domains.

22.3.3.3 SYSTIM Interface

There are three SYSTIM channels assigned to the radio. The radio receives compare events from the SYSTIM that are routed to the MDM, PBE, and RFE. These are used to trigger time start or stop commands. The radio can also be configured to output capture events to the SYSTIM, the selection of what event triggers the SYSTIM capture is done in the LRFDDBELL.SYSTIMOEV register. The normal use case is that RCL sets up and uses the SYSTIM interface.

If the radio is not in use (or not using SYSTIM interface), SYSTIM capture events can be captured from software by writing to the LRFDBELL.SYSTDMATRIG register.

22.4 Radio Usage Model

The RCL automates the low-level operation of the radio for the user, but a description of the usage model is included here for reference. The behavior of the radio is to a large degree firmware-defined. The description in this section describes the typical usage model, but other usage models are possible. Before any radio operation can be attempted the radio needs to be configured, this includes:

- Enable the necessary clocks
- Write the firmware images to the radio's internal RAMs
- Configure hardware registers
- Apply device-specific trims

The configuration is considered static and does not need to change when changing RF frequency, switching between TX and RX, and so on. The configurations for different PHYs can be exported from SmartRF™ Studio, a Windows application that helps in generation of the RF configuration register values and commands, as well as practical testing and debugging of the RF system. A set of predefined configurations for PHYs that are documented in the data sheet are available, and these configurations can be modified in SmartRF Studio to fit the use case. The PHY cannot be changed at run time.

After the radio is configured, the RCL sends an API command to the PBE. Any parameters that change depending on frequency, role, and packet contents are passed as command parameters. These parameters are set up in the BUFRAM and various HW registers, the FIFO is populated if needed, and then the API command ID is passed to the PBE. The BUFRAM parameters are documented as part of the SDK, as the parameters are fully firmware-defined.

The PBE then uses the command ID and the command parameters and does further sequencing with the MDM and RFE. As the command executes, the radio generates data, events and interrupts that can be used by the rest of the system. Once the command finishes execution, the MCE and RFE report that they are done to the PBE, and then the PBE signals to the system side that the command is completed.

22.4.1 CRC and Whitening

Many proprietary radio systems require a specific CRC or whitening algorithm. This is done through the static configuration which can be exported from SmartRF™ Studio. The registers and fields in the following description can only be manipulated with SmartRF Studio. However to be able to configure the relevant registers correctly a detailed explanation of the implementation is included here for reference. The CRC and whitening are done using the Packet Handling Accelerator (PHA). The PHA is implemented as linear-feedback shift registers (LFSRs) with programmable polynomials.

LFSR sub-engine

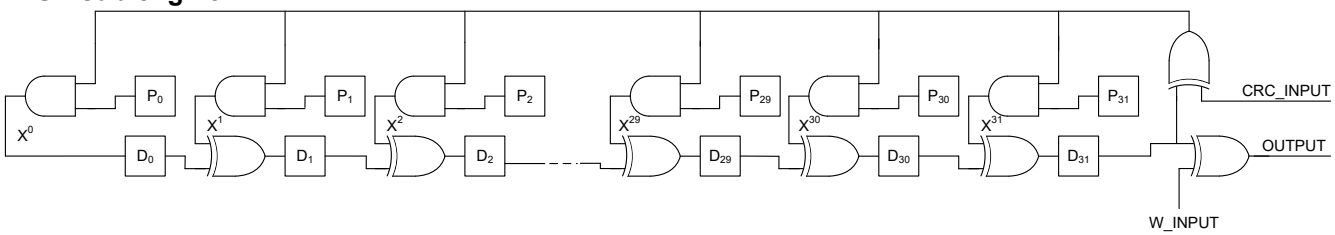


Figure 22-2. LFSR Logic

There are 2 LSFR cores, numbered 0 and 1. In the description below, the value *k* is used to identify the core. Each LSFR core consists of two 32-bit registers: One is the 32-bit polynomial POLY*k*[31:0], identified as p0–p31 in Figure 22-2; POLY*k*[*n*] corresponds to p*n*. The other is the 32-bit value register LFSR*k*VAL[31:0], identified as d0–d31 in Figure 22-2; LFSR*k*VAL[*n*] corresponds to d*n*.

The LFSR0 core can be in CRC mode or whitener mode. LFSR1 can only be in CRC mode. The mode of LFSR0 is selected through the PHACFG MODE0 bit.

An LFSR core is assumed to be operated in a bit-serial way. For each new bit, the shift register d0–d31 is clocked once as shown in the block diagram above. In CRC mode, the *crc_input* line corresponds to the bit that is entered, while output is ignored so that *w_input* is don't-care. In whitener mode, *w_input* corresponds to the bit that is entered, while *crc_input* is always 0. The polynomial register p0–p31 defines where there are taps in the shift register. A C equivalent of the operation is given below:

LFSR pseudocode

```
int j, fb;

fb = crc_input ^ LFSRVAL[31];                          /* feedback */

for (j = 31; j >= 1; j--)
{
    LFSRVAL[j] = LFSRVAL[j-1] ^ (fb & POLY[j]);    /* LFSRVAL[j-1] XOR (fb AND POLY[j]) */
}

LFSRVAL[0] = fb & POLY[0];                          /* fb AND POLY[0] */

output = w_input ^ fb;                                 /* w_input XOR fb */
```

Combination of two LFSRs

Two LFSRs can be run in parallel or in a chain. This is configured through the bits PHACFG[0:1] MODE1 bit field. If MODE1[1:0] is 00, the operation of LFSR 0 and LFSR 1 shall be independent.

If MODE*1*[1:0] is 01, the output of LFSR0 shall be input to LFSR1 in addition to the output register, and LFSR1 shall be clocked whenever a bit is output from LFSR0. This mode is only allowed when LFSR0 is in whitener mode.

If MODE*1*[1:0] is 10, the input to LFSR0 shall also be input to LFSR1, and the both these LFSR cores shall be clocked whenever a bit is input to LFSR0. This mode is allowed regardless of the mode of LFSR0.

CRC and whitening usage scenarios

This section describes how the PHA can be configured to implement certain common CRC and whitening schemes.

CRC

The PHA can support any CRC up to 32 bits. A 32-bit CRC polynomial can be described by the polynomial $x^{32} + a_{31} x^{31} + \ldots + a_1 x^1 + 1$, where all *an* are 0 or 1. To represent this, each POLY*k*[*n*] bit in the register POLY*k* are set to *an*, and POLY*k* [0] is set to 1. For a polynomial of order *m*, described by $x^m + a_{m-1} x^{m-1} + \ldots + a_1 x^1 + 1$, POLY*k*[32−*m*+*n*] are set to *an*, for all *n*=1..*m*−1, POLY*k*[32−*m*] are set to 1, and POLY*k*[31−*m*:0] are set to all zeros.

Together with the polynomial the start state of the LFSR also needs to be defined. This is done through RCL, see SDK documentation for details. Bit ordering through the CRC calculation is also handled by the radio firmware, and is controlled through the RCL.

Some CRCs used in different systems

System	Bit ordering	CRC	POLY *k* [31:0]	Initialization of LFSR *k* [31:0]
	msb first	CRC-8-ATM $x^8 + x^2 + x + 1$	0x0700 0000	0xFF00 0000
CC2500, CC2510	msb first	CRC-16 $x^{16} + x^{15} + x^2 + 1$	0x8005 0000	0xFFFF 0000
IEEE 802.15.4, OQPSK 2.4GHz	lsb first	CRC-16-CCITT $x^{16} + x^{12} + x^5 + 1$	0x1021 0000	0x0000 0000
BLE	lsb first	CRC-24-BLE $x^{24} + x^{10} + x^9 + x^6 + x^4 + x^3 + x + 1$	0x0006 5B00	0x5555 5500 (on advertising channels)
		CRC-32-IEEE 802.3 $x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$	0x04C1 1DB7	0xFFFF FFFF (for example)

Whitening

Each LFSR can be used to generate pseudo-random bit sequences, LFSR0 which supports whitener mode can collect the output in a vector. The LFSR supports generator polynomials up to order 32. A 32-bit generator polynomial can be described by the equation $x^{32} + a_{31} x^{31} + \ldots + a_1 x^1 + 1$, where all a_n are 0 or 1.
To represent this, each POLY*k*[*n*] bit in the register POLY*k* is set to *an*, and POLY*k*[0] is be set to 1. For a polynomial of order *m*, described by $x^m + a_{m-1} x^{m-1} + \ldots + a_1 x^1 + 1$, POLY*k* [32−*m*+*n*] are set to *an*, for all *n*=1..*m*−1, POLY*k*[32−*m*] are set to 1, and POLY*k*[32−*m*:0] are set to all zeros.

In whitener mode, the output of the shift register is XORed with the input bit sequence.

The LFSRs use a Galois structure. Many whitening specifications assume a Fibonacci structure. These are equivalent, but the initialization value for the shift registers must be different to get the same start point of the sequence. The whitener initialization state is handled by the radio firmware, see the SDK documentation for details.

Whitener examples

In Bluetooth Low Energy, a whitener with polynomial $x^7 + x^4 + 1$ is specified. The whitener is specified with a Galois structure and with an initialization as follows:

Position 0 is set to one.

Positions 1 to 6 are set to the channel index of the channel used when transmitting or receiving, from the most significant bit in position 1 to the least significant bit in position 6.

This whitening scheme can be obtained by setting POLY*k* to 0x22000000 and initializing the register by writing the channel index OR'ed by 0x40 into LFSR*k*BR.

In TI's 2.4 GHz products CC2500, CC2510, CC2511, the whitener has a polynomial of $x^9 + x^4 + 1$. The whitener is specified with a Fibonacci structure that is initialized with all ones. To obtain the legacy TI whitening, the PHAPOLY*k* register is set to 0x08800000. Initialization of the register is done by setting LFSR*k* to 0xF8000000, as this is equivalent to the all ones initialization with the structure used in CC25xx.

22.5 LRFDDBELL Registers

Table 22-1 lists the memory-mapped registers for the LRFDDBELL registers. All register offset addresses not listed in Table 22-1 should be considered as reserved locations and the register contents should not be modified.

Table 22-1. LRFDDBELL Registers

Offset	Acronym	Register Name	Section
0h	DESC	Description	Go
4h	CLKCTL	Clock control	Go
8h	DMACFG	DMA Configuration	Go
Ch	SYSTIMOEV	Systimer Output Event Control Register	Go
10h	SYSTDMATRIG	System DMA Trigger	Go
14h	GPOSEL0	GPO control	Go
18h	GPOSEL1	GPO control	Go
44h	IMASK0	Interrupt mask	Go
48h	RIS0	Raw interrupt status	Go
4Ch	MIS0	Masked interrupt status	Go
50h	ISET0	Interrupt set	Go
54h	ICLR0	Interrupt clear	Go
84h	IMASK1	Interrupt mask	Go
88h	RIS1	Raw interrupt status	Go
8Ch	MIS1	Masked interrupt status	Go
90h	ISET1	Interrupt set	Go
94h	ICLR1	Interrupt clear	Go
C4h	IMASK2	Interrupt mask	Go
C8h	RIS2	Raw interrupt status	Go
CCh	MIS2	Masked interrupt status	Go
D0h	ISET2	Interrupt set	Go
D4h	ICLR2	Interrupt clear	Go

Complex bit access types are encoded to fit into small table cells. Table 22-2 shows the codes that are used for access types in this section.

Table 22-2. LRFDDBELL Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

22.5.1 DESC Register (Offset = 0h) [Reset = 01411010h]

DESC is shown in Table 22-3.

Return to the Summary Table.

Description.
This register identifies the peripheral and its exact version.

Table 22-3. DESC Register Field Descriptions

Bit	Field	Type	Reset	Description
31-16	MODULEID	R/W	141h	Module identifier used to uniquely identify this IP.
15-12	STDIPOFF	R/W	1h	Standard IP MMR block offset. Standard IP MMRs are the set of from aggregated IRQ registers till DTB. 0h = STDIP MMRs do not exist 1h = These MMRs begin at offset 64*STDIPOFF from IP base address
11-8	INSTNUM	R/W	0h	IP Instance Number. If multiple instances of IP exist in the device, this field can identify the instance number
7-4	MAJREV	R/W	1h	Major rev of the IP
3-0	MINREV	R/W	0h	Minor rev of the IP

22.5.2 CLKCTL Register (Offset = 4h) [Reset = 00000001h]

CLKCTL is shown in Table 22-4.

Return to the Summary Table.

Controls the functional clock gates for the individual sub-modules.
Writing a bit to zero does not necessarily switch off the corresponding clock. It can also be requested internally.
A clock will only be switched off if internal and external requests are removed

Table 22-4. CLKCTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-14	RESERVED	R	0h	Reserved
13	DEM	R/W	0h	Enable the clock to the demodulator. The modem will request this clock automatically. This bit is to force the clock to be free running 0h = Clock not requested 1h = Clock is requested
12	MOD	R/W	0h	Enable the clock to the modulator. Modem will request this clock automatically, this bit is to force the modulator clock to be free running. 0h = Clock not requested 1h = Clock is requested
11	S2RRAM	R/W	0h	Enable the clock to the S2R RAM 0h = Clock not requested 1h = Clock is requested
10	BUFRAM	R/W	0h	Enable the clock to the BUFRAM 0h = Clock not requested 1h = Clock is requested
9	DSBRAM	R/W	0h	Enable the clock to the DSB RAM 0h = Clock not requested 1h = Clock is requested
8	RFERAM	R/W	0h	Enable the clock to the RFE RAM 0h = Clock not requested 1h = Clock is requested
7	MCERAM	R/W	0h	Enable the clock to the MCE RAM 0h = Clock not requested 1h = Clock is requested
6	PBERAM	R/W	0h	Enable the clock to the PBE RAM 0h = Clock not requested 1h = Clock is requested
5	TRC	R/W	0h	Enable the clock to the Tracer 0h = Clock not requested 1h = Clock is requested
4	S2R	R/W	0h	Enable the clock to Samples2RAM 0h = Clock not requested 1h = Clock is requested
3	RFE	R/W	0h	Enable the clock to the RFE 0h = Clock not requested 1h = Clock is requested
2	MDM	R/W	0h	Enable the clock to the Modem 0h = Clock not requested 1h = Clock is requested
1	PBE	R/W	0h	Enable the clock to the PBE 0h = Clock not requested 1h = Clock is requested
0	BRIDGE	R/W	1h	Clock enable to AHB bridge. The bridge will request it's own clock, this bit it to override that feature to have a free running clock. 0h = Clock not requested 1h = Clock is requested

22.5.3 DMACFG Register (Offset = 8h) [Reset = 00000000h]

DMACFG is shown in Table 22-5.

Return to the Summary Table.

DMA Configuration

Table 22-5. DMACFG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-3	RESERVED	R	0h	Reserved
2-1	TRIGSRC	R/W	0h	Select DMA trigger source 0h = The DMA is triggered by the PBE FW trigger 1h = The DMA is triggered by the MCE FW trigger 2h = The DMA is triggered by the MCE FW trigger 3h = The DMA is triggered from the FIFO. See the FIFO configration register for what FIFO event will generate the trigger
0	EN	R/W	0h	Enables the DMA interface 0h = Disable DMA interface, no activity on interface 1h = Enable DMA interface. The triggers are able to give activity on the interface

22.5.4 SYSTIMOEV Register (Offset = Ch) [Reset = 00000000h]

SYSTIMOEV is shown in Table 22-6.

Return to the Summary Table.

Systimer Output Event Control Register.
Controls routing of internal events to the three systimer output events

Table 22-6. SYSTIMOEV Register Field Descriptions

Bit	Field	Type	Reset	Description
31-12	RESERVED	R	0h	Reserved
11-8	SRC2	R/W	0h	Select source of systimer output event 2 (capture source) 0h = Output not enabled, always 0. 1h = RFE FW systimer capture event 0 2h = RFE FW systimer capture event 1 3h = RFE FW systimer capture event 2 4h = MCE FW systimer capture event 0 5h = MCE FW systimer capture event 1 6h = MCE FW systimer capture event 2 7h = MDM HW event 0 8h = MDM HW event 1 9h = MDM HW event 2 Ah = PBE FW systimer capture event 0 Bh = PBE FW systimer capture event 1 Ch = PBE FW systimer capture event 2
7-4	SRC1	R/W	0h	Select source of systimer output event 1 (capture source) 0h = Output not enabled, always 0. 1h = RFE FW systimer capture event 0 2h = RFE FW systimer capture event 1 3h = RFE FW systimer capture event 2 4h = MCE FW systimer capture event 0 5h = MCE FW systimer capture event 1 6h = MCE FW systimer capture event 2 7h = MDM HW event 0 8h = MDM HW event 1 9h = MDM HW event 2 Ah = PBE FW systimer capture event 0 Bh = PBE FW systimer capture event 1 Ch = PBE FW systimer capture event 2
3-0	SRC0	R/W	0h	Select source of systimer output event 0 (capture source) 0h = Output not enabled, always 0. 1h = RFE FW systimer capture event 0 2h = RFE FW systimer capture event 1 3h = RFE FW systimer capture event 2 4h = MCE FW systimer capture event 0 5h = MCE FW systimer capture event 1 6h = MCE FW systimer capture event 2 7h = MDM HW event 0 8h = MDM HW event 1 9h = MDM HW event 2 Ah = PBE FW systimer capture event 0 Bh = PBE FW systimer capture event 1 Ch = PBE FW systimer capture event 2

22.5.5 SYSTDMATRIG Register (Offset = 10h) [Reset = 00000000h]

SYSTDMATRIG is shown in Table 22-7.

Return to the Summary Table.

System DMA Trigger
Manual triggering of systimer capture event or DMA trigger
This comes on top of any HW driven sources configured in SYSTIMOEV

Table 22-7. SYSTDMATRIG Register Field Descriptions

Bit	Field	Type	Reset	Description
31-4	RESERVED	R	0h	Reserved
3	DMA	W	0h	Trigger a DMA request from the Radio 0h = DMA not manually triggered 1h = DMA request manually triggered
2	SYST2	W	0h	Trigger a capture event on systimer event 0 from the radio 0h = Not capture event triggered 1h = Capture event triggered
1	SYST1	W	0h	Trigger a capture event on systimer event 0 from the radio 0h = Not capture event triggered 1h = Capture event triggered
0	SYST0	W	0h	Trigger a capture event on systimer event 0 from the radio 0h = Not capture event triggered 1h = Capture event triggered

22.5.6 GPOSEL0 Register (Offset = 14h) [Reset = 00000000h]

GPOSEL0 is shown in Table 22-8.

Return to the Summary Table.

Controls routing of GPO signals from MDM, RFE and PBE to the radio GPO lines

Table 22-8. GPOSEL0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-29	RESERVED	R	0h	Reserved
28-24	SRC3	R/W	0h	Select source of radio GPO line 3 0h = Output not enabled 1h = Select PBE GPO line 0 2h = Select PBE GPO line 1 3h = Select PBE GPO line 2 4h = Select PBE GPO line 3 5h = Select PBE GPO line 4 6h = Select PBE GPO line 5 7h = Select PBE GPO line 6 8h = Select PBE GPO line 7 9h = Select MCE GPO line 0 Ah = Select MCE GPO line 1 Bh = Select MCE GPO line 2 Ch = Select MCE GPO line 3 Dh = Select MCE GPO line 4 Eh = Select MCE GPO line 5 Fh = Select MCE GPO line 6 10h = Select MCE GPO line 7 11h = Select RFE GPO line 0 12h = Select RFE GPO line 1 13h = Select RFE GPO line 2 14h = Select RFE GPO line 3 15h = Select RFE GPO line 4 16h = Select RFE GPO line 5 17h = Select RFE GPO line 6 18h = Select RFE GPO line 7 19h = Select RFCTRC GPO line 3
23-21	RESERVED	R	0h	Reserved
20-16	SRC2	R/W	0h	Select source of radio GPO line 2 0h = Output not enabled 1h = Select PBE GPO line 0 2h = Select PBE GPO line 1 3h = Select PBE GPO line 2 4h = Select PBE GPO line 3 5h = Select PBE GPO line 4 6h = Select PBE GPO line 5 7h = Select PBE GPO line 6 8h = Select PBE GPO line 7 9h = Select MCE GPO line 0 Ah = Select MCE GPO line 1 Bh = Select MCE GPO line 2 Ch = Select MCE GPO line 3 Dh = Select MCE GPO line 4 Eh = Select MCE GPO line 5 Fh = Select MCE GPO line 6 10h = Select MCE GPO line 7 11h = Select RFE GPO line 0 12h = Select RFE GPO line 1 13h = Select RFE GPO line 2 14h = Select RFE GPO line 3 15h = Select RFE GPO line 4 16h = Select RFE GPO line 5 17h = Select RFE GPO line 6 18h = Select RFE GPO line 7 19h = Select RFCTRC GPO line 2

Table 22-8. GPOSEL0 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
15-13	RESERVED	R	0h	Reserved
12-8	SRC1	R/W	0h	Select source of radio GPO line 1 0h = Output not enabled 1h = Select PBE GPO line 0 2h = Select PBE GPO line 1 3h = Select PBE GPO line 2 4h = Select PBE GPO line 3 5h = Select PBE GPO line 4 6h = Select PBE GPO line 5 7h = Select PBE GPO line 6 8h = Select PBE GPO line 7 9h = Select MCE GPO line 0 Ah = Select MCE GPO line 1 Bh = Select MCE GPO line 2 Ch = Select MCE GPO line 3 Dh = Select MCE GPO line 4 Eh = Select MCE GPO line 5 Fh = Select MCE GPO line 6 10h = Select MCE GPO line 7 11h = Select RFE GPO line 0 12h = Select RFE GPO line 1 13h = Select RFE GPO line 2 14h = Select RFE GPO line 3 15h = Select RFE GPO line 4 16h = Select RFE GPO line 5 17h = Select RFE GPO line 6 18h = Select RFE GPO line 7 19h = Select RFCTRC GPO line 1
7-5	RESERVED	R	0h	Reserved
4-0	SRC0	R/W	0h	Select source of radio GPO line 0 0h = Output not enabled 1h = Select PBE GPO line 0 2h = Select PBE GPO line 1 3h = Select PBE GPO line 2 4h = Select PBE GPO line 3 5h = Select PBE GPO line 4 6h = Select PBE GPO line 5 7h = Select PBE GPO line 6 8h = Select PBE GPO line 7 9h = Select MCE GPO line 0 Ah = Select MCE GPO line 1 Bh = Select MCE GPO line 2 Ch = Select MCE GPO line 3 Dh = Select MCE GPO line 4 Eh = Select MCE GPO line 5 Fh = Select MCE GPO line 6 10h = Select MCE GPO line 7 11h = Select RFE GPO line 0 12h = Select RFE GPO line 1 13h = Select RFE GPO line 2 14h = Select RFE GPO line 3 15h = Select RFE GPO line 4 16h = Select RFE GPO line 5 17h = Select RFE GPO line 6 18h = Select RFE GPO line 7 19h = Select RFCTRC GPO line 0

22.5.7 GPOSEL1 Register (Offset = 18h) [Reset = 00000000h]

GPOSEL1 is shown in Table 22-9.

Return to the Summary Table.

Controls routing of GPO signals from MDM, RFE and PBE to the radio GPO lines

Table 22-9. GPOSEL1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31-29	RESERVED	R	0h	Reserved
28-24	SRC7	R/W	0h	Select source of radio GPO line 7 0h = No output not enabled 1h = Select PBE GPO line 0 2h = Select PBE GPO line 1 3h = Select PBE GPO line 2 4h = Select PBE GPO line 3 5h = Select PBE GPO line 4 6h = Select PBE GPO line 5 7h = Select PBE GPO line 6 8h = Select PBE GPO line 7 9h = Select MCE GPO line 0 Ah = Select MCE GPO line 1 Bh = Select MCE GPO line 2 Ch = Select MCE GPO line 3 Dh = Select MCE GPO line 4 Eh = Select MCE GPO line 5 Fh = Select MCE GPO line 6 10h = Select MCE GPO line 7 11h = Select RFE GPO line 0 12h = Select RFE GPO line 1 13h = Select RFE GPO line 2 14h = Select RFE GPO line 3 15h = Select RFE GPO line 4 16h = Select RFE GPO line 5 17h = Select RFE GPO line 6 18h = Select RFE GPO line 7 19h = Select RFCTRC GPO line 7
23-21	RESERVED	R	0h	Reserved
20-16	SRC6	R/W	0h	Select source of radio GPO line 6 0h = No output not enabled 1h = Select PBE GPO line 0 2h = Select PBE GPO line 1 3h = Select PBE GPO line 2 4h = Select PBE GPO line 3 5h = Select PBE GPO line 4 6h = Select PBE GPO line 5 7h = Select PBE GPO line 6 8h = Select PBE GPO line 7 9h = Select MCE GPO line 0 Ah = Select MCE GPO line 1 Bh = Select MCE GPO line 2 Ch = Select MCE GPO line 3 Dh = Select MCE GPO line 4 Eh = Select MCE GPO line 5 Fh = Select MCE GPO line 6 10h = Select MCE GPO line 7 11h = Select RFE GPO line 0 12h = Select RFE GPO line 1 13h = Selevt RFE GPO line 2 14h = Select RFE GPO line 3 15h = Select RFE GPO line 4 16h = Select RFE GPO line 5 17h = Select RFE GPO line 6 18h = Select RFE GPO line 7 19h = Select RFCTRC GPO line 6

Table 22-9. GPOSEL1 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
15-13	RESERVED	R	0h	Reserved
12-8	SRC5	R/W	0h	Select source of radio GPO line 5 0h = No output not enabled 1h = Select PBE GPO line 0 2h = Select PBE GPO line 1 3h = Select PBE GPO line 2 4h = Select PBE GPO line 3 5h = Select PBE GPO line 4 6h = Select PBE GPO line 5 7h = Select PBE GPO line 6 8h = Select PBE GPO line 7 9h = Select MCE GPO line 0 Ah = Select MCE GPO line 1 Bh = Select MCE GPO line 2 Ch = Select MCE GPO line 3 Dh = Select MCE GPO line 4 Eh = Select MCE GPO line 5 Fh = Select MCE GPO line 6 10h = Select MCE GPO line 7 11h = Select RFE GPO line 0 12h = Select RFE GPO line 1 13h = Select RFE GPO line 2 14h = Select RFE GPO line 3 15h = Select RFE GPO line 4 16h = Select RFE GPO line 5 17h = Select RFE GPO line 6 18h = Select RFE GPO line 7 19h = Select RFCTRC GPO line 5
7-5	RESERVED	R	0h	Reserved
4-0	SRC4	R/W	0h	Select source of radio GPO line 4 0h = No output not enabled 1h = Select PBE GPO line 0 2h = Select PBE GPO line 1 3h = Select PBE GPO line 2 4h = Select PBE GPO line 3 5h = Select PBE GPO line 4 6h = Select PBE GPO line 5 7h = Select PBE GPO line 6 8h = Select PBE GPO line 7 9h = Select MCE GPO line 0 Ah = Select MCE GPO line 1 Bh = Select MCE GPO line 2 Ch = Select MCE GPO line 3 Dh = Select MCE GPO line 4 Eh = Select MCE GPO line 5 Fh = Select MCE GPO line 6 10h = Select MCE GPO line 7 11h = Select RFE GPO line 0 12h = Select RFE GPO line 1 13h = Select RFE GPO line 2 14h = Select RFE GPO line 3 15h = Select RFE GPO line 4 16h = Select RFE GPO line 5 17h = Select RFE GPO line 6 18h = Select RFE GPO line 7 19h = Select RFCTRC GPO line 4

22.5.8 IMASK0 Register (Offset = 44h) [Reset = 00000000h]

IMASK0 is shown in Table 22-10.

Return to the Summary Table.

Interrupt mask.
This register selects interrupt sources which are allowed to pass from RIS to MIS when the corresponding bit-fields are set to 1.

Table 22-10. IMASK0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	SYSTIM2	R/W	0h	SYSTIM2 event 0h = Disable interrupt mask 1h = Enable interrupt mask
30	SYSTIM1	R/W	0h	SYSTIM1 event 0h = Disable interrupt mask 1h = Enable interrupt mask
29	SYSTIM0	R/W	0h	SYSTIM0 event 0h = Disable interrupt mask 1h = Enable interrupt mask
28	MDMDONE	R/W	0h	MDMDONE event 0h = Disable interrupt mask 1h = Enable interrupt mask
27	MDMIN	R/W	0h	MDMIN event 0h = Disable interrupt mask 1h = Enable interrupt mask
26	MDMOUT	R/W	0h	MDMOUT event 0h = Disable interrupt mask 1h = Enable interrupt mask
25	MDMSOFT2	R/W	0h	MDMSOFT2 event 0h = Disable interrupt mask 1h = Enable interrupt mask
24	MDMSOFT1	R/W	0h	MDMSOFT2 event 0h = Disable interrupt mask 1h = Enable interrupt mask
23	MDMSOFT0	R/W	0h	MDMSOFT2 event 0h = Disable interrupt mask 1h = Enable interrupt mask
22	RFEDONE	R/W	0h	RFEDONE event 0h = Disable interrupt mask 1h = Enable interrupt mask
21	RFESOFT1	R/W	0h	RFESOFT1 event 0h = Disable interrupt mask 1h = Enable interrupt mask
20	RFESOFT0	R/W	0h	RFESOFT0 event 0h = Disable interrupt mask 1h = Enable interrupt mask
19	LOCK	R/W	0h	LOCK event 0h = Disable interrupt mask 1h = Enable interrupt mask
18	LOL	R/W	0h	LOSS_OF_LOCK event 0h = Disable interrupt mask 1h = Enable interrupt mask
17	TXFIFO	R/W	0h	TXFIFO event 0h = Disable interrupt mask 1h = Enable interrupt mask
16	RXFIFO	R/W	0h	RXFIFO event 0h = Disable interrupt mask 1h = Enable interrupt mask

Table 22-10. IMASK0 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
15	PBE15	R/W	0h	PBE15 event 0h = Disable interrupt mask 1h = Enable interrupt mask
14	PBE14	R/W	0h	PBE14 event 0h = Disable interrupt mask 1h = Enable interrupt mask
13	PBE13	R/W	0h	PBE13 event 0h = Disable interrupt mask 1h = Enable interrupt mask
12	PBE12	R/W	0h	PBE12 event 0h = Disable interrupt mask 1h = Enable interrupt mask
11	PBE11	R/W	0h	PBE11 event 0h = Disable interrupt mask 1h = Enable interrupt mask
10	PBE10	R/W	0h	PBE10 event 0h = Disable interrupt mask 1h = Enable interrupt mask
9	RESERVED	R	0h	Reserved
8	PBE8	R/W	0h	PBE8 event 0h = Disable interrupt mask 1h = Enable interrupt mask
7	PBE7	R/W	0h	PBE7 event 0h = Disable interrupt mask 1h = Enable interrupt mask
6	PBE6	R/W	0h	PBE6 event 0h = Disable interrupt mask 1h = Enable interrupt mask
5	PBE5	R/W	0h	PBE5 event 0h = Disable interrupt mask 1h = Enable interrupt mask
4	PBE4	R/W	0h	PBE4 event 0h = Disable interrupt mask 1h = Enable interrupt mask
3	PBE3	R/W	0h	PBE3 event 0h = Disable interrupt mask 1h = Enable interrupt mask
2	PBE2	R/W	0h	PBE2 event 0h = Disable interrupt mask 1h = Enable interrupt mask
1	PBE1	R/W	0h	PBE1 event 0h = Disable interrupt mask 1h = Enable interrupt mask
0	PBE0	R/W	0h	PBE0 event 0h = Disable interrupt mask 1h = Enable interrupt mask

22.5.9 RIS0 Register (Offset = 48h) [Reset = 00000000h]

RIS0 is shown in Table 22-11.

Return to the Summary Table.

Raw interrupt status. This register reflects the state of all pending interrupts, regardless of masking. This register allows the user to implement a poll scheme. A flag set in this register can be cleared by writing 1 to the corresponding ICLR register bit.

Table 22-11. RIS0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	SYSTIM2	R	0h	SYSTIM2 event 0h = Interrupt did not occur 1h = Interrupt occurred
30	SYSTIM1	R	0h	SYSTIM1 event 0h = Interrupt did not occur 1h = Interrupt occurred
29	SYSTIM0	R	0h	SYSTIM0 event 0h = Interrupt did not occur 1h = Interrupt occurred
28	MDMDONE	R	0h	MDMDONE event 0h = Interrupt did not occur 1h = Interrupt occurred
27	MDMIN	R	0h	MDMIN event 0h = Interrupt did not occur 1h = Interrupt occurred
26	MDMOUT	R	0h	MDMOUT event 0h = Interrupt did not occur 1h = Interrupt occurred
25	MDMSOFT2	R	0h	MDMSOFT event 0h = Interrupt did not occur 1h = Interrupt occurred
24	MDMSOFT1	R	0h	MDMSOFT1 event 0h = Interrupt did not occur 1h = Interrupt occurred
23	MDMSOFT0	R	0h	MDMSOFT event 0h = Interrupt did not occur 1h = Interrupt occurred
22	RFEDONE	R	0h	RFEDONE event 0h = Interrupt did not occur 1h = Interrupt occurred
21	RFESOFT1	R	0h	RFESOFT1 event 0h = Interrupt did not occur 1h = Interrupt occurred
20	RFESOFT0	R	0h	RFESOFT0 event 0h = Interrupt did not occur 1h = Interrupt occurred
19	LOCK	R	0h	LOCK event 0h = Interrupt did not occur 1h = Interrupt occurred
18	LOL	R	0h	LOSS_OF_LOCK event 0h = Interrupt did not occur 1h = Interrupt occurred
17	TXFIFO	R	0h	TXFIFO event 0h = Interrupt did not occur 1h = Interrupt occurred
16	RXFIFO	R	0h	RXFIFO event 0h = Interrupt did not occur 1h = Interrupt occurred

Table 22-11. RIS0 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
15	PBE15	R	0h	PBE15 event 0h = Interrupt did not occur 1h = Interrupt occurred
14	PBE14	R	0h	PBE14 event 0h = Interrupt did not occur 1h = Interrupt occurred
13	PBE13	R	0h	PBE13 event 0h = Interrupt did not occur 1h = Interrupt occurred
12	PBE12	R	0h	PBE12 event 0h = Interrupt did not occur 1h = Interrupt occurred
11	PBE11	R	0h	PBE11 event 0h = Interrupt did not occur 1h = Interrupt occurred
10	PBE10	R	0h	PBE10 event 0h = Interrupt did not occur 1h = Interrupt occurred
9	PBE9	R	0h	PBE9 event 0h = Interrupt did not occur 1h = Interrupt occurred
8	PBE8	R	0h	PBE8 event 0h = Interrupt did not occur 1h = Interrupt occurred
7	PBE7	R	0h	PBE7 event 0h = Interrupt did not occur 1h = Interrupt occurred
6	PBE6	R	0h	PBE6 event 0h = Interrupt did not occur 1h = Interrupt occurred
5	PBE5	R	0h	PBE5 event 0h = Interrupt did not occur 1h = Interrupt occurred
4	PBE4	R	0h	PBE4 event 0h = Interrupt did not occur 1h = Interrupt occurred
3	PBE3	R	0h	PBE3 event 0h = Interrupt did not occur 1h = Interrupt occurred
2	PBE2	R	0h	PBE2 event 0h = Interrupt did not occur 1h = Interrupt occurred
1	PBE1	R	0h	PBE1 event 0h = Interrupt did not occur 1h = Interrupt occurred
0	PBE0	R	0h	PBE0 event 0h = Interrupt did not occur 1h = Interrupt occurred

22.5.10 MIS0 Register (Offset = 4Ch) [Reset = 00000000h]

MIS0 is shown in Table 22-12.

Return to the Summary Table.

Masked interrupt status. This register is simply a bitwise AND of the contents of IMASK and RIS.*] registers. A flag set in this register can be cleared by writing 1 to the corresponding ICLR register bit.

Table 22-12. MIS0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	SYSTIM2	R	0h	SYSTIM2 event 0h = Interrupt did not occur 1h = Interrupt occurred
30	SYSTIM1	R	0h	SYSTIM1 event 0h = Interrupt did not occur 1h = Interrupt occurred
29	SYSTIM0	R	0h	SYSTIM0 event 0h = Interrupt did not occur 1h = Interrupt occurred
28	MDMDONE	R	0h	MDMDONE event 0h = Interrupt did not occur 1h = Interrupt occurred
27	MDMIN	R	0h	MDMIN event 0h = Interrupt did not occur 1h = Interrupt occurred
26	MDMOUT	R	0h	MDMOUT event 0h = Interrupt did not occur 1h = Interrupt occurred
25	MDMSOFT2	R	0h	MDMSOFT event 0h = Interrupt did not occur 1h = Interrupt occurred
24	MDMSOFT1	R	0h	MDMSOFT1 event 0h = Interrupt did not occur 1h = Interrupt occurred
23	MDMSOFT0	R	0h	MDMSOFT event 0h = Interrupt did not occur 1h = Interrupt occurred
22	RFEDONE	R	0h	RFEDONE event 0h = Interrupt did not occur 1h = Interrupt occurred
21	RFESOFT1	R	0h	RFESOFT1 event 0h = Interrupt did not occur 1h = Interrupt occurred
20	RFESOFT0	R	0h	RFESOFT0 event 0h = Interrupt did not occur 1h = Interrupt occurred
19	LOCK	R	0h	LOCK event 0h = Interrupt did not occur 1h = Interrupt occurred
18	LOL	R	0h	LOSS_OF_LOCK event 0h = Interrupt did not occur 1h = Interrupt occurred
17	TXFIFO	R	0h	TXFIFO event 0h = Interrupt did not occur 1h = Interrupt occurred
16	RXFIFO	R	0h	RXFIFO event 0h = Interrupt did not occur 1h = Interrupt occurred

Table 22-12. MIS0 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
15	PBE15	R	0h	PBE15 event 0h = Interrupt did not occur 1h = Interrupt occurred
14	PBE14	R	0h	PBE14 event 0h = Interrupt did not occur 1h = Interrupt occurred
13	PBE13	R	0h	PBE13 event 0h = Interrupt did not occur 1h = Interrupt occurred
12	PBE12	R	0h	PBE12 event 0h = Interrupt did not occur 1h = Interrupt occurred
11	PBE11	R	0h	PBE11 event 0h = Interrupt did not occur 1h = Interrupt occurred
10	PBE10	R	0h	PBE10 event 0h = Interrupt did not occur 1h = Interrupt occurred
9	PBE9	R	0h	PBE9 event 0h = Interrupt did not occur 1h = Interrupt occurred
8	PBE8	R	0h	PBE8 event 0h = Interrupt did not occur 1h = Interrupt occurred
7	PBE7	R	0h	PBE7 event 0h = Interrupt did not occur 1h = Interrupt occurred
6	PBE6	R	0h	PBE6 event 0h = Interrupt did not occur 1h = Interrupt occurred
5	PBE5	R	0h	PBE5 event 0h = Interrupt did not occur 1h = Interrupt occurred
4	PBE4	R	0h	PBE4 event 0h = Interrupt did not occur 1h = Interrupt occurred
3	PBE3	R	0h	PBE3 event 0h = Interrupt did not occur 1h = Interrupt occurred
2	PBE2	R	0h	PBE2 event 0h = Interrupt did not occur 1h = Interrupt occurred
1	PBE1	R	0h	PBE1 event 0h = Interrupt did not occur 1h = Interrupt occurred
0	PBE0	R	0h	PBE0 event 0h = Interrupt did not occur 1h = Interrupt occurred

22.5.11 ISET0 Register (Offset = 50h) [Reset = 00000000h]

ISET0 is shown in Table 22-13.

Return to the Summary Table.

Interrupt set register. This register can used by software for diagnostics and safety checking purposes. Writing a 1 to a bit in this register will set the event and the corresponding RIS bit also gets set. If the corresponding IMASK bit is set, then the corresponding MIS register bit also gets set.

Table 22-13. ISET0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	SYSTIM2	R/W	0h	SYSTIM2 event 0h = Writing 0 has no effect 1h = Set Interrupt
30	SYSTIM1	R/W	0h	SYSTIM1 event 0h = Writing 0 has no effect 1h = Set Interrupt
29	SYSTIM0	R/W	0h	SYSTIM0 event 0h = Writing 0 has no effect 1h = Set Interrupt
28	MDMDONE	R/W	0h	MDMDONE event 0h = Writing 0 has no effect 1h = Set Interrupt
27	MDMIN	R/W	0h	MDMIN event 0h = Writing 0 has no effect 1h = Set Interrupt
26	MDMOUT	R/W	0h	MDMOUT event 0h = Writing 0 has no effect 1h = Set Interrupt
25	MDMSOFT2	R/W	0h	MDMSOFT event 0h = Writing 0 has no effect 1h = Set Interrupt
24	MDMSOFT1	R/W	0h	MDMSOFT1 event 0h = Writing 0 has no effect 1h = Set Interrupt
23	MDMSOFT0	R/W	0h	MDMSOFT event 0h = Writing 0 has no effect 1h = Set Interrupt
22	RFEDONE	R/W	0h	RFEDONE event 0h = Writing 0 has no effect 1h = Set Interrupt
21	RFESOFT1	R/W	0h	RFESOFT1 event 0h = Writing 0 has no effect 1h = Set Interrupt
20	RFESOFT0	R/W	0h	RFESOFT0 event 0h = Writing 0 has no effect 1h = Set Interrupt
19	LOCK	R/W	0h	LOCK event 0h = Writing 0 has no effect 1h = Set Interrupt
18	LOL	R/W	0h	LOSS_OF_LOCK event 0h = Writing 0 has no effect 1h = Set Interrupt
17	TXFIFO	R/W	0h	TXFIFO event 0h = Writing 0 has no effect 1h = Set Interrupt
16	RXFIFO	R/W	0h	RXFIFO event 0h = Writing 0 has no effect 1h = Set Interrupt

Table 22-13. ISET0 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
15	PBE15	R/W	0h	PBE15 event 0h = Writing 0 has no effect 1h = Set Interrupt
14	PBE14	R/W	0h	PBE14 event 0h = Writing 0 has no effect 1h = Set Interrupt
13	PBE13	R/W	0h	PBE13 event 0h = Writing 0 has no effect 1h = Set Interrupt
12	PBE12	R/W	0h	PBE12 event 0h = Writing 0 has no effect 1h = Set Interrupt
11	PBE11	R/W	0h	PBE11 event 0h = Writing 0 has no effect 1h = Set Interrupt
10	PBE10	R/W	0h	PBE10 event 0h = Writing 0 has no effect 1h = Set Interrupt
9	PBE9	R/W	0h	PBE9 event 0h = Writing 0 has no effect 1h = Set Interrupt
8	PBE8	R/W	0h	PBE8 event 0h = Writing 0 has no effect 1h = Set Interrupt
7	PBE7	R/W	0h	PBE7 event 0h = Writing 0 has no effect 1h = Set Interrupt
6	PBE6	R/W	0h	PBE6 event 0h = Writing 0 has no effect 1h = Set Interrupt
5	PBE5	R/W	0h	PBE5 event 0h = Writing 0 has no effect 1h = Set Interrupt
4	PBE4	R/W	0h	PBE4 event 0h = Writing 0 has no effect 1h = Set Interrupt
3	PBE3	R/W	0h	PBE3 event 0h = Writing 0 has no effect 1h = Set Interrupt
2	PBE2	R/W	0h	PBE2 event 0h = Writing 0 has no effect 1h = Set Interrupt
1	PBE1	R/W	0h	PBE1 event 0h = Writing 0 has no effect 1h = Set Interrupt
0	PBE0	R/W	0h	PBE0 event 0h = Writing 0 has no effect 1h = Set Interrupt

22.5.12 ICLR0 Register (Offset = 54h) [Reset = 00000000h]

ICLR0 is shown in Table 22-14.

Return to the Summary Table.

Interrupt clear. Write a 1 to clear corresponding Interrupt.

Table 22-14. ICLR0 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	SYSTIM2	R/W	0h	SYSTIM2 event 0h = Writing 0 has no effect 1h = Clear Interrupt
30	SYSTIM1	R/W	0h	SYSTIM1 event 0h = Writing 0 has no effect 1h = Clear Interrupt
29	SYSTIM0	R/W	0h	SYSTIM0 event 0h = Writing 0 has no effect 1h = Clear Interrupt
28	MDMDONE	R/W	0h	MDMDONE event 0h = Writing 0 has no effect 1h = Clear Interrupt
27	MDMIN	R/W	0h	MDMIN event 0h = Writing 0 has no effect 1h = Clear Interrupt
26	MDMOUT	R/W	0h	MDMOUT event 0h = Writing 0 has no effect 1h = Clear Interrupt
25	MDMSOFT2	R/W	0h	MDMSOFT event 0h = Writing 0 has no effect 1h = Clear Interrupt
24	MDMSOFT1	R/W	0h	MDMSOFT1 event 0h = Writing 0 has no effect 1h = Clear Interrupt
23	MDMSOFT0	R/W	0h	MDMSOFT event 0h = Writing 0 has no effect 1h = Clear Interrupt
22	RFEDONE	R/W	0h	RFEDONE event 0h = Writing 0 has no effect 1h = Clear Interrupt
21	RFESOFT1	R/W	0h	RFESOFT1 event 0h = Writing 0 has no effect 1h = Clear Interrupt
20	RFESOFT0	R/W	0h	RFESOFT0 event 0h = Writing 0 has no effect 1h = Clear Interrupt
19	LOCK	R/W	0h	LOCK event 0h = Writing 0 has no effect 1h = Clear Interrupt
18	LOL	R/W	0h	LOSS_OF_LOCK event 0h = Writing 0 has no effect 1h = Clear Interrupt
17	TXFIFO	R/W	0h	TXFIFO event 0h = Writing 0 has no effect 1h = Clear Interrupt
16	RXFIFO	R/W	0h	RXFIFO event 0h = Writing 0 has no effect 1h = Clear Interrupt

Table 22-14. ICLR0 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
15	PBE15	R/W	0h	PBE15 event 0h = Writing 0 has no effect 1h = Clear Interrupt
14	PBE14	R/W	0h	PBE14 event 0h = Writing 0 has no effect 1h = Clear Interrupt
13	PBE13	R/W	0h	PBE13 event 0h = Writing 0 has no effect 1h = Clear Interrupt
12	PBE12	R/W	0h	PBE12 event 0h = Writing 0 has no effect 1h = Clear Interrupt
11	PBE11	R/W	0h	PBE11 event 0h = Writing 0 has no effect 1h = Clear Interrupt
10	PBE10	R/W	0h	PBE10 event 0h = Writing 0 has no effect 1h = Clear Interrupt
9	PBE9	R/W	0h	PBE9 event 0h = Writing 0 has no effect 1h = Clear Interrupt
8	PBE8	R/W	0h	PBE8 event 0h = Writing 0 has no effect 1h = Clear Interrupt
7	PBE7	R/W	0h	PBE7 event 0h = Writing 0 has no effect 1h = Clear Interrupt
6	PBE6	R/W	0h	PBE6 event 0h = Writing 0 has no effect 1h = Clear Interrupt
5	PBE5	R/W	0h	PBE5 event 0h = Writing 0 has no effect 1h = Clear Interrupt
4	PBE4	R/W	0h	PBE4 event 0h = Writing 0 has no effect 1h = Clear Interrupt
3	PBE3	R/W	0h	PBE3 event 0h = Writing 0 has no effect 1h = Clear Interrupt
2	PBE2	R/W	0h	PBE2 event 0h = Writing 0 has no effect 1h = Clear Interrupt
1	PBE1	R/W	0h	PBE1 event 0h = Writing 0 has no effect 1h = Clear Interrupt
0	PBE0	R/W	0h	PBE0 event 0h = Writing 0 has no effect 1h = Clear Interrupt

22.5.13 IMASK1 Register (Offset = 84h) [Reset = 00000000h]

IMASK1 is shown in Table 22-15.

Return to the Summary Table.

Interrupt mask. This register selects interrupt sources which are allowed to pass from RIS to MIS when the corresponding bit-fields are set to 1.

Table 22-15. IMASK1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	SYSTIM2	R/W	0h	SYSTIM2 event 0h = Disable interrupt mask 1h = Enable interrupt mask
29	SYSTIM1	R/W	0h	SYSTIM1 event 0h = Disable interrupt mask 1h = Enable interrupt mask
28	SYSTIM0	R/W	0h	SYSTIM0 event 0h = Disable interrupt mask 1h = Enable interrupt mask
27	MDMDONE	R/W	0h	MDMDONE event 0h = Disable interrupt mask 1h = Enable interrupt mask
26	MDMIN	R/W	0h	MDMIN event 0h = Disable interrupt mask 1h = Enable interrupt mask
25	MDMOUT	R/W	0h	MDMOUT event 0h = Disable interrupt mask 1h = Enable interrupt mask
24	MDMSOFT2	R/W	0h	MDMSOFT2 event 0h = Disable interrupt mask 1h = Enable interrupt mask
23	MDMSOFT1	R/W	0h	MDMSOFT2 event 0h = Disable interrupt mask 1h = Enable interrupt mask
22	MDMSOFT0	R/W	0h	MDMSOFT2 event 0h = Disable interrupt mask 1h = Enable interrupt mask
21	RFEDONE	R/W	0h	RFEDONE event 0h = Disable interrupt mask 1h = Enable interrupt mask
20	RFESOFT1	R/W	0h	RFESOFT1 event 0h = Disable interrupt mask 1h = Enable interrupt mask
19	RFESOFT0	R/W	0h	RFESOFT0 event 0h = Disable interrupt mask 1h = Enable interrupt mask
18	LOCK	R/W	0h	LOCK event 0h = Disable interrupt mask 1h = Enable interrupt mask
17	LOL	R/W	0h	LOSS_OF_LOCK event 0h = Disable interrupt mask 1h = Enable interrupt mask
16	TXFIFO	R/W	0h	TXFIFO event 0h = Disable interrupt mask 1h = Enable interrupt mask
15	RXFIFO	R/W	0h	RXFIFO event 0h = Disable interrupt mask 1h = Enable interrupt mask

Table 22-15. IMASK1 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
14	PBE15	R/W	0h	PBE15 event 0h = Disable interrupt mask 1h = Enable interrupt mask
13	PBE14	R/W	0h	PBE14 event 0h = Disable interrupt mask 1h = Enable interrupt mask
12	PBE13	R/W	0h	PBE13 event 0h = Disable interrupt mask 1h = Enable interrupt mask
11	PBE12	R/W	0h	PBE12 event 0h = Disable interrupt mask 1h = Enable interrupt mask
10	PBE11	R/W	0h	PBE11 event 0h = Disable interrupt mask 1h = Enable interrupt mask
9	PBE10	R/W	0h	PBE10 event 0h = Disable interrupt mask 1h = Enable interrupt mask
8	PBE8	R/W	0h	PBE8 event 0h = Disable interrupt mask 1h = Enable interrupt mask
7	PBE7	R/W	0h	PBE7 event 0h = Disable interrupt mask 1h = Enable interrupt mask
6	PBE6	R/W	0h	PBE6 event 0h = Disable interrupt mask 1h = Enable interrupt mask
5	PBE5	R/W	0h	PBE5 event 0h = Disable interrupt mask 1h = Enable interrupt mask
4	PBE4	R/W	0h	PBE4 event 0h = Disable interrupt mask 1h = Enable interrupt mask
3	PBE3	R/W	0h	PBE3 event 0h = Disable interrupt mask 1h = Enable interrupt mask
2	PBE2	R/W	0h	PBE2 event 0h = Disable interrupt mask 1h = Enable interrupt mask
1	PBE1	R/W	0h	PBE1 event 0h = Disable interrupt mask 1h = Enable interrupt mask
0	PBE0	R/W	0h	PBE0 event 0h = Disable interrupt mask 1h = Enable interrupt mask

22.5.14 RIS1 Register (Offset = 88h) [Reset = 00000000h]

RIS1 is shown in Table 22-16.

Return to the Summary Table.

Raw interrupt status. Reflects all pending interrupts, regardless of masking. The RIS0 register allows the user to implement a poll scheme. A flag set in this register can be cleared by writing 1 to the ICLR register bit even if the corresponding IMASK bit is not enabled.

Table 22-16. RIS1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	SYSTIM2	R	0h	SYSTIM2 event 0h = Interrupt did not occur 1h = Interrupt occurred
30	SYSTIM1	R	0h	SYSTIM1 event 0h = Interrupt did not occur 1h = Interrupt occurred
29	SYSTIM0	R	0h	SYSTIM0 event 0h = Interrupt did not occur 1h = Interrupt occurred
28	MDMDONE	R	0h	MDMDONE event 0h = Interrupt did not occur 1h = Interrupt occurred
27	MDMIN	R	0h	MDMIN event 0h = Interrupt did not occur 1h = Interrupt occurred
26	MDMOUT	R	0h	MDMOUT event 0h = Interrupt did not occur 1h = Interrupt occurred
25	MDMSOFT2	R	0h	MDMSOFT event 0h = Interrupt did not occur 1h = Interrupt occurred
24	MDMSOFT1	R	0h	MDMSOFT1 event 0h = Interrupt did not occur 1h = Interrupt occurred
23	MDMSOFT0	R	0h	MDMSOFT event 0h = Interrupt did not occur 1h = Interrupt occurred
22	RFEDONE	R	0h	RFEDONE event 0h = Interrupt did not occur 1h = Interrupt occurred
21	RFESOFT1	R	0h	RFESOFT1 event 0h = Interrupt did not occur 1h = Interrupt occurred
20	RFESOFT0	R	0h	RFESOFT0 event 0h = Interrupt did not occur 1h = Interrupt occurred
19	LOCK	R	0h	LOCK event 0h = Interrupt did not occur 1h = Interrupt occurred
18	LOL	R	0h	LOSS_OF_LOCK event 0h = Interrupt did not occur 1h = Interrupt occurred
17	TXFIFO	R	0h	TXFIFO event 0h = Interrupt did not occur 1h = Interrupt occurred
16	RXFIFO	R	0h	RXFIFO event 0h = Interrupt did not occur 1h = Interrupt occurred

Table 22-16. RIS1 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
15	PBE15	R	0h	PBE15 event 0h = Interrupt did not occur 1h = Interrupt occurred
14	PBE14	R	0h	PBE14 event 0h = Interrupt did not occur 1h = Interrupt occurred
13	PBE13	R	0h	PBE13 event 0h = Interrupt did not occur 1h = Interrupt occurred
12	PBE12	R	0h	PBE12 event 0h = Interrupt did not occur 1h = Interrupt occurred
11	PBE11	R	0h	PBE11 event 0h = Interrupt did not occur 1h = Interrupt occurred
10	PBE10	R	0h	PBE10 event 0h = Interrupt did not occur 1h = Interrupt occurred
9	PBE9	R	0h	PBE9 event 0h = Interrupt did not occur 1h = Interrupt occurred
8	PBE8	R	0h	PBE8 event 0h = Interrupt did not occur 1h = Interrupt occurred
7	PBE7	R	0h	PBE7 event 0h = Interrupt did not occur 1h = Interrupt occurred
6	PBE6	R	0h	PBE6 event 0h = Interrupt did not occur 1h = Interrupt occurred
5	PBE5	R	0h	PBE5 event 0h = Interrupt did not occur 1h = Interrupt occurred
4	PBE4	R	0h	PBE4 event 0h = Interrupt did not occur 1h = Interrupt occurred
3	PBE3	R	0h	PBE3 event 0h = Interrupt did not occur 1h = Interrupt occurred
2	PBE2	R	0h	PBE2 event 0h = Interrupt did not occur 1h = Interrupt occurred
1	PBE1	R	0h	PBE1 event 0h = Interrupt did not occur 1h = Interrupt occurred
0	PBE0	R	0h	PBE0 event 0h = Interrupt did not occur 1h = Interrupt occurred

22.5.15 MIS1 Register (Offset = 8Ch) [Reset = 00000000h]

MIS1 is shown in Table 22-17.

Return to the Summary Table.

Masked interrupt status. This is an AND of the IMASK and RIS registers.

Table 22-17. MIS1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	SYSTIM2	R	0h	SYSTIM2 event 0h = Interrupt did not occur 1h = Interrupt occurred
30	SYSTIM1	R	0h	SYSTIM1 event 0h = Interrupt did not occur 1h = Interrupt occurred
29	SYSTIM0	R	0h	SYSTIM0 event 0h = Interrupt did not occur 1h = Interrupt occurred
28	MDMDONE	R	0h	MDMDONE event 0h = Interrupt did not occur 1h = Interrupt occurred
27	MDMIN	R	0h	MDMIN event 0h = Interrupt did not occur 1h = Interrupt occurred
26	MDMOUT	R	0h	MDMOUT event 0h = Interrupt did not occur 1h = Interrupt occurred
25	MDMSOFT2	R	0h	MDMSOFT event 0h = Interrupt did not occur 1h = Interrupt occurred
24	MDMSOFT1	R	0h	MDMSOFT1 event 0h = Interrupt did not occur 1h = Interrupt occurred
23	MDMSOFT0	R	0h	MDMSOFT event 0h = Interrupt did not occur 1h = Interrupt occurred
22	RFEDONE	R	0h	RFEDONE event 0h = Interrupt did not occur 1h = Interrupt occurred
21	RFESOFT1	R	0h	RFESOFT1 event 0h = Interrupt did not occur 1h = Interrupt occurred
20	RFESOFT0	R	0h	RFESOFT0 event 0h = Interrupt did not occur 1h = Interrupt occurred
19	LOCK	R	0h	LOCK event 0h = Interrupt did not occur 1h = Interrupt occurred
18	LOL	R	0h	LOSS_OF_LOCK event 0h = Interrupt did not occur 1h = Interrupt occurred
17	TXFIFO	R	0h	TXFIFO event 0h = Interrupt did not occur 1h = Interrupt occurred
16	RXFIFO	R	0h	RXFIFO event 0h = Interrupt did not occur 1h = Interrupt occurred

Table 22-17. MIS1 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
15	PBE15	R	0h	PBE15 event 0h = Interrupt did not occur 1h = Interrupt occurred
14	PBE14	R	0h	PBE14 event 0h = Interrupt did not occur 1h = Interrupt occurred
13	PBE13	R	0h	PBE13 event 0h = Interrupt did not occur 1h = Interrupt occurred
12	PBE12	R	0h	PBE12 event 0h = Interrupt did not occur 1h = Interrupt occurred
11	PBE11	R	0h	PBE11 event 0h = Interrupt did not occur 1h = Interrupt occurred
10	PBE10	R	0h	PBE10 event 0h = Interrupt did not occur 1h = Interrupt occurred
9	PBE9	R	0h	PBE9 event 0h = Interrupt did not occur 1h = Interrupt occurred
8	PBE8	R	0h	PBE8 event 0h = Interrupt did not occur 1h = Interrupt occurred
7	PBE7	R	0h	PBE7 event 0h = Interrupt did not occur 1h = Interrupt occurred
6	PBE6	R	0h	PBE6 event 0h = Interrupt did not occur 1h = Interrupt occurred
5	PBE5	R	0h	PBE5 event 0h = Interrupt did not occur 1h = Interrupt occurred
4	PBE4	R	0h	PBE4 event 0h = Interrupt did not occur 1h = Interrupt occurred
3	PBE3	R	0h	PBE3 event 0h = Interrupt did not occur 1h = Interrupt occurred
2	PBE2	R	0h	PBE2 event 0h = Interrupt did not occur 1h = Interrupt occurred
1	PBE1	R	0h	PBE1 event 0h = Interrupt did not occur 1h = Interrupt occurred
0	PBE0	R	0h	PBE0 event 0h = Interrupt did not occur 1h = Interrupt occurred

22.5.16 ISET1 Register (Offset = 90h) [Reset = 00000000h]

ISET1 is shown in Table 22-18.

Return to the Summary Table.

Interrupt set register. This register can used by software for diagnostics and safety checking purposes. Writing a 1 to a bit in this register will set the event and the corresponding RIS bit also gets set. If the corresponding IMASK bit is set, then the corresponding MIS register bit also gets set.

Table 22-18. ISET1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	SYSTIM2	R/W	0h	SYSTIM2 event 0h = Writing 0 has no effect 1h = Set Interrupt
30	SYSTIM1	R/W	0h	SYSTIM1 event 0h = Writing 0 has no effect 1h = Set Interrupt
29	SYSTIM0	R/W	0h	SYSTIM0 event 0h = Writing 0 has no effect 1h = Set Interrupt
28	MDMDONE	R/W	0h	MDMDONE event 0h = Writing 0 has no effect 1h = Set Interrupt
27	MDMIN	R/W	0h	MDMIN event 0h = Writing 0 has no effect 1h = Set Interrupt
26	MDMOUT	R/W	0h	MDMOUT event 0h = Writing 0 has no effect 1h = Set Interrupt
25	MDMSOFT2	R/W	0h	MDMSOFT event 0h = Writing 0 has no effect 1h = Set Interrupt
24	MDMSOFT1	R/W	0h	MDMSOFT1 event 0h = Writing 0 has no effect 1h = Set Interrupt
23	MDMSOFT0	R/W	0h	MDMSOFT event 0h = Writing 0 has no effect 1h = Set Interrupt
22	RFEDONE	R/W	0h	RFEDONE event 0h = Writing 0 has no effect 1h = Set Interrupt
21	RFESOFT1	R/W	0h	RFESOFT1 event 0h = Writing 0 has no effect 1h = Set Interrupt
20	RFESOFT0	R/W	0h	RFESOFT0 event 0h = Writing 0 has no effect 1h = Set Interrupt
19	LOCK	R/W	0h	LOCK event 0h = Writing 0 has no effect 1h = Set Interrupt
18	LOL	R/W	0h	LOSS_OF_LOCK event 0h = Writing 0 has no effect 1h = Set Interrupt
17	TXFIFO	R/W	0h	TXFIFO event 0h = Writing 0 has no effect 1h = Set Interrupt
16	RXFIFO	R/W	0h	RXFIFO event 0h = Writing 0 has no effect 1h = Set Interrupt

Table 22-18. ISET1 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
15	PBE15	R/W	0h	PBE15 event 0h = Writing 0 has no effect 1h = Set Interrupt
14	PBE14	R/W	0h	PBE14 event 0h = Writing 0 has no effect 1h = Set Interrupt
13	PBE13	R/W	0h	PBE13 event 0h = Writing 0 has no effect 1h = Set Interrupt
12	PBE12	R/W	0h	PBE12 event 0h = Writing 0 has no effect 1h = Set Interrupt
11	PBE11	R/W	0h	PBE11 event 0h = Writing 0 has no effect 1h = Set Interrupt
10	PBE10	R/W	0h	PBE10 event 0h = Writing 0 has no effect 1h = Set Interrupt
9	PBE9	R/W	0h	PBE9 event 0h = Writing 0 has no effect 1h = Set Interrupt
8	PBE8	R/W	0h	PBE8 event 0h = Writing 0 has no effect 1h = Set Interrupt
7	PBE7	R/W	0h	PBE7 event 0h = Writing 0 has no effect 1h = Set Interrupt
6	PBE6	R/W	0h	PBE6 event 0h = Writing 0 has no effect 1h = Set Interrupt
5	PBE5	R/W	0h	PBE5 event 0h = Writing 0 has no effect 1h = Set Interrupt
4	PBE4	R/W	0h	PBE4 event 0h = Writing 0 has no effect 1h = Set Interrupt
3	PBE3	R/W	0h	PBE3 event 0h = Writing 0 has no effect 1h = Set Interrupt
2	PBE2	R/W	0h	PBE2 event 0h = Writing 0 has no effect 1h = Set Interrupt
1	PBE1	R/W	0h	PBE1 event 0h = Writing 0 has no effect 1h = Set Interrupt
0	PBE0	R/W	0h	PBE0 event 0h = Writing 0 has no effect 1h = Set Interrupt

22.5.17 ICLR1 Register (Offset = 94h) [Reset = 00000000h]

ICLR1 is shown in Table 22-19.

Return to the Summary Table.

Interrupt clear. Write a 1 to clear corresponding Interrupt.

Table 22-19. ICLR1 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	SYSTIM2	R/W	0h	SYSTIM2 event 0h = Writing 0 has no effect 1h = Clear Interrupt
30	SYSTIM1	R/W	0h	SYSTIM1 event 0h = Writing 0 has no effect 1h = Clear Interrupt
29	SYSTIM0	R/W	0h	SYSTIM0 event 0h = Writing 0 has no effect 1h = Clear Interrupt
28	MDMDONE	R/W	0h	MDMDONE event 0h = Writing 0 has no effect 1h = Clear Interrupt
27	MDMIN	R/W	0h	MDMIN event 0h = Writing 0 has no effect 1h = Clear Interrupt
26	MDMOUT	R/W	0h	MDMOUT event 0h = Writing 0 has no effect 1h = Clear Interrupt
25	MDMSOFT2	R/W	0h	MDMSOFT event 0h = Writing 0 has no effect 1h = Clear Interrupt
24	MDMSOFT1	R/W	0h	MDMSOFT1 event 0h = Writing 0 has no effect 1h = Clear Interrupt
23	MDMSOFT0	R/W	0h	MDMSOFT event 0h = Writing 0 has no effect 1h = Clear Interrupt
22	RFEDONE	R/W	0h	RFEDONE event 0h = Writing 0 has no effect 1h = Clear Interrupt
21	RFESOFT1	R/W	0h	RFESOFT1 event 0h = Writing 0 has no effect 1h = Clear Interrupt
20	RFESOFT0	R/W	0h	RFESOFT0 event 0h = Writing 0 has no effect 1h = Clear Interrupt
19	LOCK	R/W	0h	LOCK event 0h = Writing 0 has no effect 1h = Clear Interrupt
18	LOL	R/W	0h	LOSS_OF_LOCK event 0h = Writing 0 has no effect 1h = Clear Interrupt
17	TXFIFO	R/W	0h	TXFIFO event 0h = Writing 0 has no effect 1h = Clear Interrupt
16	RXFIFO	R/W	0h	RXFIFO event 0h = Writing 0 has no effect 1h = Clear Interrupt

Table 22-19. ICLR1 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
15	PBE15	R/W	0h	PBE15 event 0h = Writing 0 has no effect 1h = Clear Interrupt
14	PBE14	R/W	0h	PBE14 event 0h = Writing 0 has no effect 1h = Clear Interrupt
13	PBE13	R/W	0h	PBE13 event 0h = Writing 0 has no effect 1h = Clear Interrupt
12	PBE12	R/W	0h	PBE12 event 0h = Writing 0 has no effect 1h = Clear Interrupt
11	PBE11	R/W	0h	PBE11 event 0h = Writing 0 has no effect 1h = Clear Interrupt
10	PBE10	R/W	0h	PBE10 event 0h = Writing 0 has no effect 1h = Clear Interrupt
9	PBE9	R/W	0h	PBE9 event 0h = Writing 0 has no effect 1h = Clear Interrupt
8	PBE8	R/W	0h	PBE8 event 0h = Writing 0 has no effect 1h = Clear Interrupt
7	PBE7	R/W	0h	PBE7 event 0h = Writing 0 has no effect 1h = Clear Interrupt
6	PBE6	R/W	0h	PBE6 event 0h = Writing 0 has no effect 1h = Clear Interrupt
5	PBE5	R/W	0h	PBE5 event 0h = Writing 0 has no effect 1h = Clear Interrupt
4	PBE4	R/W	0h	PBE4 event 0h = Writing 0 has no effect 1h = Clear Interrupt
3	PBE3	R/W	0h	PBE3 event 0h = Writing 0 has no effect 1h = Clear Interrupt
2	PBE2	R/W	0h	PBE2 event 0h = Writing 0 has no effect 1h = Clear Interrupt
1	PBE1	R/W	0h	PBE1 event 0h = Writing 0 has no effect 1h = Clear Interrupt
0	PBE0	R/W	0h	PBE0 event 0h = Writing 0 has no effect 1h = Clear Interrupt

22.5.18 IMASK2 Register (Offset = C4h) [Reset = 00000000h]

IMASK2 is shown in Table 22-20.

Return to the Summary Table.

Interrupt mask. This register selects interrupt sources which are allowed to pass from RIS to MIS when the corresponding bit-fields are set to 1.

Table 22-20. IMASK2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0h	Reserved
30	SYSTIM2	R/W	0h	SYSTIM2 event 0h = Disable interrupt mask 1h = Enable interrupt mask
29	SYSTIM1	R/W	0h	SYSTIM1 event 0h = Disable interrupt mask 1h = Enable interrupt mask
28	SYSTIM0	R/W	0h	SYSTIM0 event 0h = Disable interrupt mask 1h = Enable interrupt mask
27	MDMDONE	R/W	0h	MDMDONE event 0h = Disable interrupt mask 1h = Enable interrupt mask
26	MDMIN	R/W	0h	MDMIN event 0h = Disable interrupt mask 1h = Enable interrupt mask
25	MDMOUT	R/W	0h	MDMOUT event 0h = Disable interrupt mask 1h = Enable interrupt mask
24	MDMSOFT2	R/W	0h	MDMSOFT2 event 0h = Disable interrupt mask 1h = Enable interrupt mask
23	MDMSOFT1	R/W	0h	MDMSOFT2 event 0h = Disable interrupt mask 1h = Enable interrupt mask
22	MDMSOFT0	R/W	0h	MDMSOFT2 event 0h = Disable interrupt mask 1h = Enable interrupt mask
21	RFEDONE	R/W	0h	RFEDONE event 0h = Disable interrupt mask 1h = Enable interrupt mask
20	RFESOFT1	R/W	0h	RFESOFT1 event 0h = Disable interrupt mask 1h = Enable interrupt mask
19	RFESOFT0	R/W	0h	RFESOFT0 event 0h = Disable interrupt mask 1h = Enable interrupt mask
18	LOCK	R/W	0h	LOCK event 0h = Disable interrupt mask 1h = Enable interrupt mask
17	LOL	R/W	0h	LOSS_OF_LOCK event 0h = Disable interrupt mask 1h = Enable interrupt mask
16	TXFIFO	R/W	0h	TXFIFO event 0h = Disable interrupt mask 1h = Enable interrupt mask
15	RXFIFO	R/W	0h	RXFIFO event 0h = Disable interrupt mask 1h = Enable interrupt mask

Table 22-20. IMASK2 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
14	PBE15	R/W	0h	PBE15 event 0h = Disable interrupt mask 1h = Enable interrupt mask
13	PBE14	R/W	0h	PBE14 event 0h = Disable interrupt mask 1h = Enable interrupt mask
12	PBE13	R/W	0h	PBE13 event 0h = Disable interrupt mask 1h = Enable interrupt mask
11	PBE12	R/W	0h	PBE12 event 0h = Disable interrupt mask 1h = Enable interrupt mask
10	PBE11	R/W	0h	PBE11 event 0h = Disable interrupt mask 1h = Enable interrupt mask
9	PBE10	R/W	0h	PBE10 event 0h = Disable interrupt mask 1h = Enable interrupt mask
8	PBE8	R/W	0h	PBE8 event 0h = Disable interrupt mask 1h = Enable interrupt mask
7	PBE7	R/W	0h	PBE7 event 0h = Disable interrupt mask 1h = Enable interrupt mask
6	PBE6	R/W	0h	PBE6 event 0h = Disable interrupt mask 1h = Enable interrupt mask
5	PBE5	R/W	0h	PBE5 event 0h = Disable interrupt mask 1h = Enable interrupt mask
4	PBE4	R/W	0h	PBE4 event 0h = Disable interrupt mask 1h = Enable interrupt mask
3	PBE3	R/W	0h	PBE3 event 0h = Disable interrupt mask 1h = Enable interrupt mask
2	PBE2	R/W	0h	PBE2 event 0h = Disable interrupt mask 1h = Enable interrupt mask
1	PBE1	R/W	0h	PBE1 event 0h = Disable interrupt mask 1h = Enable interrupt mask
0	PBE0	R/W	0h	PBE0 event 0h = Disable interrupt mask 1h = Enable interrupt mask

22.5.19 RIS2 Register (Offset = C8h) [Reset = 00000000h]

RIS2 is shown in Table 22-21.

Return to the Summary Table.

Raw interrupt status. Reflects all pending interrupts, regardless of masking. The RIS0 register allows the user to implement a poll scheme. A flag set in this register can be cleared by writing 1 to the ICLR register bit even if the corresponding IMASK bit is not enabled.

Table 22-21. RIS2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	SYSTIM2	R	0h	SYSTIM2 event 0h = Interrupt did not occur 1h = Interrupt occurred
30	SYSTIM1	R	0h	SYSTIM1 event 0h = Interrupt did not occur 1h = Interrupt occurred
29	SYSTIM0	R	0h	SYSTIM0 event 0h = Interrupt did not occur 1h = Interrupt occurred
28	MDMDONE	R	0h	MDMDONE event 0h = Interrupt did not occur 1h = Interrupt occurred
27	MDMIN	R	0h	MDMIN event 0h = Interrupt did not occur 1h = Interrupt occurred
26	MDMOUT	R	0h	MDMOUT event 0h = Interrupt did not occur 1h = Interrupt occurred
25	MDMSOFT2	R	0h	MDMSOFT event 0h = Interrupt did not occur 1h = Interrupt occurred
24	MDMSOFT1	R	0h	MDMSOFT1 event 0h = Interrupt did not occur 1h = Interrupt occurred
23	MDMSOFT0	R	0h	MDMSOFT event 0h = Interrupt did not occur 1h = Interrupt occurred
22	RFEDONE	R	0h	RFEDONE event 0h = Interrupt did not occur 1h = Interrupt occurred
21	RFESOFT1	R	0h	RFESOFT1 event 0h = Interrupt did not occur 1h = Interrupt occurred
20	RFESOFT0	R	0h	RFESOFT0 event 0h = Interrupt did not occur 1h = Interrupt occurred
19	LOCK	R	0h	LOCK event 0h = Interrupt did not occur 1h = Interrupt occurred
18	LOL	R	0h	LOSS_OF_LOCK event 0h = Interrupt did not occur 1h = Interrupt occurred
17	TXFIFO	R	0h	TXFIFO event 0h = Interrupt did not occur 1h = Interrupt occurred
16	RXFIFO	R	0h	RXFIFO event 0h = Interrupt did not occur 1h = Interrupt occurred

Table 22-21. RIS2 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
15	PBE15	R	0h	PBE15 event 0h = Interrupt did not occur 1h = Interrupt occurred
14	PBE14	R	0h	PBE14 event 0h = Interrupt did not occur 1h = Interrupt occurred
13	PBE13	R	0h	PBE13 event 0h = Interrupt did not occur 1h = Interrupt occurred
12	PBE12	R	0h	PBE12 event 0h = Interrupt did not occur 1h = Interrupt occurred
11	PBE11	R	0h	PBE11 event 0h = Interrupt did not occur 1h = Interrupt occurred
10	PBE10	R	0h	PBE10 event 0h = Interrupt did not occur 1h = Interrupt occurred
9	PBE9	R	0h	PBE9 event 0h = Interrupt did not occur 1h = Interrupt occurred
8	PBE8	R	0h	PBE8 event 0h = Interrupt did not occur 1h = Interrupt occurred
7	PBE7	R	0h	PBE7 event 0h = Interrupt did not occur 1h = Interrupt occurred
6	PBE6	R	0h	PBE6 event 0h = Interrupt did not occur 1h = Interrupt occurred
5	PBE5	R	0h	PBE5 event 0h = Interrupt did not occur 1h = Interrupt occurred
4	PBE4	R	0h	PBE4 event 0h = Interrupt did not occur 1h = Interrupt occurred
3	PBE3	R	0h	PBE3 event 0h = Interrupt did not occur 1h = Interrupt occurred
2	PBE2	R	0h	PBE2 event 0h = Interrupt did not occur 1h = Interrupt occurred
1	PBE1	R	0h	PBE1 event 0h = Interrupt did not occur 1h = Interrupt occurred
0	PBE0	R	0h	PBE0 event 0h = Interrupt did not occur 1h = Interrupt occurred

22.5.20 MIS2 Register (Offset = CCh) [Reset = 00000000h]

MIS2 is shown in Table 22-22.

Return to the Summary Table.

Masked interrupt status. This is an AND of the IMASK and RIS registers.

Table 22-22. MIS2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	SYSTIM2	R	0h	SYSTIM2 event 0h = Interrupt did not occur 1h = Interrupt occurred
30	SYSTIM1	R	0h	SYSTIM1 event 0h = Interrupt did not occur 1h = Interrupt occurred
29	SYSTIM0	R	0h	SYSTIM0 event 0h = Interrupt did not occur 1h = Interrupt occurred
28	MDMDONE	R	0h	MDMDONE event 0h = Interrupt did not occur 1h = Interrupt occurred
27	MDMIN	R	0h	MDMIN event 0h = Interrupt did not occur 1h = Interrupt occurred
26	MDMOUT	R	0h	MDMOUT event 0h = Interrupt did not occur 1h = Interrupt occurred
25	MDMSOFT2	R	0h	MDMSOFT event 0h = Interrupt did not occur 1h = Interrupt occurred
24	MDMSOFT1	R	0h	MDMSOFT1 event 0h = Interrupt did not occur 1h = Interrupt occurred
23	MDMSOFT0	R	0h	MDMSOFT event 0h = Interrupt did not occur 1h = Interrupt occurred
22	RFEDONE	R	0h	RFEDONE event 0h = Interrupt did not occur 1h = Interrupt occurred
21	RFESOFT1	R	0h	RFESOFT1 event 0h = Interrupt did not occur 1h = Interrupt occurred
20	RFESOFT0	R	0h	RFESOFT0 event 0h = Interrupt did not occur 1h = Interrupt occurred
19	LOCK	R	0h	LOCK event 0h = Interrupt did not occur 1h = Interrupt occurred
18	LOL	R	0h	LOSS_OF_LOCK event 0h = Interrupt did not occur 1h = Interrupt occurred
17	TXFIFO	R	0h	TXFIFO event 0h = Interrupt did not occur 1h = Interrupt occurred
16	RXFIFO	R	0h	RXFIFO event 0h = Interrupt did not occur 1h = Interrupt occurred

Table 22-22. MIS2 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
15	PBE15	R	0h	PBE15 event 0h = Interrupt did not occur 1h = Interrupt occurred
14	PBE14	R	0h	PBE14 event 0h = Interrupt did not occur 1h = Interrupt occurred
13	PBE13	R	0h	PBE13 event 0h = Interrupt did not occur 1h = Interrupt occurred
12	PBE12	R	0h	PBE12 event 0h = Interrupt did not occur 1h = Interrupt occurred
11	PBE11	R	0h	PBE11 event 0h = Interrupt did not occur 1h = Interrupt occurred
10	PBE10	R	0h	PBE10 event 0h = Interrupt did not occur 1h = Interrupt occurred
9	PBE9	R	0h	PBE9 event 0h = Interrupt did not occur 1h = Interrupt occurred
8	PBE8	R	0h	PBE8 event 0h = Interrupt did not occur 1h = Interrupt occurred
7	PBE7	R	0h	PBE7 event 0h = Interrupt did not occur 1h = Interrupt occurred
6	PBE6	R	0h	PBE6 event 0h = Interrupt did not occur 1h = Interrupt occurred
5	PBE5	R	0h	PBE5 event 0h = Interrupt did not occur 1h = Interrupt occurred
4	PBE4	R	0h	PBE4 event 0h = Interrupt did not occur 1h = Interrupt occurred
3	PBE3	R	0h	PBE3 event 0h = Interrupt did not occur 1h = Interrupt occurred
2	PBE2	R	0h	PBE2 event 0h = Interrupt did not occur 1h = Interrupt occurred
1	PBE1	R	0h	PBE1 event 0h = Interrupt did not occur 1h = Interrupt occurred
0	PBE0	R	0h	PBE0 event 0h = Interrupt did not occur 1h = Interrupt occurred

22.5.21 ISET2 Register (Offset = D0h) [Reset = 00000000h]

ISET2 is shown in Table 22-23.

Return to the Summary Table.

Interrupt set register. This register can used by software for diagnostics and safety checking purposes. Writing a 1 to a bit in this register will set the event and the corresponding RIS bit also gets set. If the corresponding IMASK bit is set, then the corresponding MIS register bit also gets set.

Table 22-23. ISET2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	SYSTIM2	R/W	0h	SYSTIM2 event 0h = Writing 0 has no effect 1h = Set Interrupt
30	SYSTIM1	R/W	0h	SYSTIM1 event 0h = Writing 0 has no effect 1h = Set Interrupt
29	SYSTIM0	R/W	0h	SYSTIM0 event 0h = Writing 0 has no effect 1h = Set Interrupt
28	MDMDONE	R/W	0h	MDMDONE event 0h = Writing 0 has no effect 1h = Set Interrupt
27	MDMIN	R/W	0h	MDMIN event 0h = Writing 0 has no effect 1h = Set Interrupt
26	MDMOUT	R/W	0h	MDMOUT event 0h = Writing 0 has no effect 1h = Set Interrupt
25	MDMSOFT2	R/W	0h	MDMSOFT event 0h = Writing 0 has no effect 1h = Set Interrupt
24	MDMSOFT1	R/W	0h	MDMSOFT1 event 0h = Writing 0 has no effect 1h = Set Interrupt
23	MDMSOFT0	R/W	0h	MDMSOFT event 0h = Writing 0 has no effect 1h = Set Interrupt
22	RFEDONE	R/W	0h	RFEDONE event 0h = Writing 0 has no effect 1h = Set Interrupt
21	RFESOFT1	R/W	0h	RFESOFT1 event 0h = Writing 0 has no effect 1h = Set Interrupt
20	RFESOFT0	R/W	0h	RFESOFT0 event 0h = Writing 0 has no effect 1h = Set Interrupt
19	LOCK	R/W	0h	LOCK event 0h = Writing 0 has no effect 1h = Set Interrupt
18	LOL	R/W	0h	LOSS_OF_LOCK event 0h = Writing 0 has no effect 1h = Set Interrupt
17	TXFIFO	R/W	0h	TXFIFO event 0h = Writing 0 has no effect 1h = Set Interrupt
16	RXFIFO	R/W	0h	RXFIFO event 0h = Writing 0 has no effect 1h = Set Interrupt

Table 22-23. ISET2 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
15	PBE15	R/W	0h	PBE15 event 0h = Writing 0 has no effect 1h = Set Interrupt
14	PBE14	R/W	0h	PBE14 event 0h = Writing 0 has no effect 1h = Set Interrupt
13	PBE13	R/W	0h	PBE13 event 0h = Writing 0 has no effect 1h = Set Interrupt
12	PBE12	R/W	0h	PBE12 event 0h = Writing 0 has no effect 1h = Set Interrupt
11	PBE11	R/W	0h	PBE11 event 0h = Writing 0 has no effect 1h = Set Interrupt
10	PBE10	R/W	0h	PBE10 event 0h = Writing 0 has no effect 1h = Set Interrupt
9	PBE9	R/W	0h	PBE9 event 0h = Writing 0 has no effect 1h = Set Interrupt
8	PBE8	R/W	0h	PBE8 event 0h = Writing 0 has no effect 1h = Set Interrupt
7	PBE7	R/W	0h	PBE7 event 0h = Writing 0 has no effect 1h = Set Interrupt
6	PBE6	R/W	0h	PBE6 event 0h = Writing 0 has no effect 1h = Set Interrupt
5	PBE5	R/W	0h	PBE5 event 0h = Writing 0 has no effect 1h = Set Interrupt
4	PBE4	R/W	0h	PBE4 event 0h = Writing 0 has no effect 1h = Set Interrupt
3	PBE3	R/W	0h	PBE3 event 0h = Writing 0 has no effect 1h = Set Interrupt
2	PBE2	R/W	0h	PBE2 event 0h = Writing 0 has no effect 1h = Set Interrupt
1	PBE1	R/W	0h	PBE1 event 0h = Writing 0 has no effect 1h = Set Interrupt
0	PBE0	R/W	0h	PBE0 event 0h = Writing 0 has no effect 1h = Set Interrupt

22.5.22 ICLR2 Register (Offset = D4h) [Reset = 00000000h]

ICLR2 is shown in Table 22-24.

Return to the Summary Table.

Interrupt clear. Write a 1 to clear corresponding Interrupt.

Table 22-24. ICLR2 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	SYSTIM2	R/W	0h	SYSTIM2 event 0h = Writing 0 has no effect 1h = Clear Interrupt
30	SYSTIM1	R/W	0h	SYSTIM1 event 0h = Writing 0 has no effect 1h = Clear Interrupt
29	SYSTIM0	R/W	0h	SYSTIM0 event 0h = Writing 0 has no effect 1h = Clear Interrupt
28	MDMDONE	R/W	0h	MDMDONE event 0h = Writing 0 has no effect 1h = Clear Interrupt
27	MDMIN	R/W	0h	MDMIN event 0h = Writing 0 has no effect 1h = Clear Interrupt
26	MDMOUT	R/W	0h	MDMOUT event 0h = Writing 0 has no effect 1h = Clear Interrupt
25	MDMSOFT2	R/W	0h	MDMSOFT event 0h = Writing 0 has no effect 1h = Clear Interrupt
24	MDMSOFT1	R/W	0h	MDMSOFT1 event 0h = Writing 0 has no effect 1h = Clear Interrupt
23	MDMSOFT0	R/W	0h	MDMSOFT event 0h = Writing 0 has no effect 1h = Clear Interrupt
22	RFEDONE	R/W	0h	RFEDONE event 0h = Writing 0 has no effect 1h = Clear Interrupt
21	RFESOFT1	R/W	0h	RFESOFT1 event 0h = Writing 0 has no effect 1h = Clear Interrupt
20	RFESOFT0	R/W	0h	RFESOFT0 event 0h = Writing 0 has no effect 1h = Clear Interrupt
19	LOCK	R/W	0h	LOCK event 0h = Writing 0 has no effect 1h = Clear Interrupt
18	LOL	R/W	0h	LOSS_OF_LOCK event 0h = Writing 0 has no effect 1h = Clear Interrupt
17	TXFIFO	R/W	0h	TXFIFO event 0h = Writing 0 has no effect 1h = Clear Interrupt
16	RXFIFO	R/W	0h	RXFIFO event 0h = Writing 0 has no effect 1h = Clear Interrupt

Table 22-24. ICLR2 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
15	PBE15	R/W	0h	PBE15 event 0h = Writing 0 has no effect 1h = Clear Interrupt
14	PBE14	R/W	0h	PBE14 event 0h = Writing 0 has no effect 1h = Clear Interrupt
13	PBE13	R/W	0h	PBE13 event 0h = Writing 0 has no effect 1h = Clear Interrupt
12	PBE12	R/W	0h	PBE12 event 0h = Writing 0 has no effect 1h = Clear Interrupt
11	PBE11	R/W	0h	PBE11 event 0h = Writing 0 has no effect 1h = Clear Interrupt
10	PBE10	R/W	0h	PBE10 event 0h = Writing 0 has no effect 1h = Clear Interrupt
9	PBE9	R/W	0h	PBE9 event 0h = Writing 0 has no effect 1h = Clear Interrupt
8	PBE8	R/W	0h	PBE8 event 0h = Writing 0 has no effect 1h = Clear Interrupt
7	PBE7	R/W	0h	PBE7 event 0h = Writing 0 has no effect 1h = Clear Interrupt
6	PBE6	R/W	0h	PBE6 event 0h = Writing 0 has no effect 1h = Clear Interrupt
5	PBE5	R/W	0h	PBE5 event 0h = Writing 0 has no effect 1h = Clear Interrupt
4	PBE4	R/W	0h	PBE4 event 0h = Writing 0 has no effect 1h = Clear Interrupt
3	PBE3	R/W	0h	PBE3 event 0h = Writing 0 has no effect 1h = Clear Interrupt
2	PBE2	R/W	0h	PBE2 event 0h = Writing 0 has no effect 1h = Clear Interrupt
1	PBE1	R/W	0h	PBE1 event 0h = Writing 0 has no effect 1h = Clear Interrupt
0	PBE0	R/W	0h	PBE0 event 0h = Writing 0 has no effect 1h = Clear Interrupt

22.6 LRFDRXF Registers

Table 22-25 lists the memory-mapped registers for the LRFDRXF registers. All register offset addresses not listed in Table 22-25 should be considered as reserved locations and the register contents should not be modified.

Table 22-25. LRFDRXF Registers

Offset	Acronym	Register Name	Section
0h	RXD	Data to from RXFIFO	Go

Complex bit access types are encoded to fit into small table cells. Table 22-26 shows the codes that are used for access types in this section.

Table 22-26. LRFDRXF Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

22.6.1 RXD Register (Offset = 0h) [Reset = 00000000h]

RXD is shown in Table 22-27.

Return to the Summary Table.

RX FIFO data. When written the register data is pushed to the RX FIFO. When read, data is popped from the RX FIFO

Table 22-27. RXD Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	DATA	R/W	0h	RX FIFO data. When written the register data is pushed to the RX FIFO. When read, data is popped from the RX FIFO. When writing or reading this register the access size will determine how many bytes are pushed to or popped from the FIFO. It is possible to push or pop 1,2 or 4 bytes depending on the access being done.

22.7 LRFDTXF Registers

Table 22-28 lists the memory-mapped registers for the LRFDTXF registers. All register offset addresses not listed in Table 22-28 should be considered as reserved locations and the register contents should not be modified.

Table 22-28. LRFDTXF Registers

Offset	Acronym	Register Name	Section
0h	TXD	Data to from TXFIFO	Go

Complex bit access types are encoded to fit into small table cells. Table 22-29 shows the codes that are used for access types in this section.

Table 22-29. LRFDTXF Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value

22.7.1 TXD Register (Offset = 0h) [Reset = 00000000h]

TXD is shown in Table 22-30.

Return to the Summary Table.

TX FIFO data. When written the register data is pushed to the TX FIFO. When read, data is popped from the TX FIFO

Table 22-30. TXD Register Field Descriptions

Bit	Field	Type	Reset	Description
31-0	DATA	R/W	0h	TX FIFO data. When written the register data is pushed to the TX FIFO. When read, data is popped from the TX FIFO. When writing or reading this register the access size will determine how many bytes are pushed to or popped from the FIFO. It is possible to push or pop 1,2 or 4 bytes depending on the access being done.

Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

Changes from May 1, 2023 to August 1, 2024	Page
• Clarified description for idle mode in Table 6-2	260
• Clarified what permissions need to be set to completely block debug access	425
• Added example debug flow charts	429
• Updated flash protection description	467
• Added CCFG CRC information	473
• Added information on preparing for production	473
• Clarified that the CCFG allowChipErase field also affects the bootloader	473
• Corrected values in the example	489
• Updated channel availability	701
• Removed duplicate information	813
• Updated feature list	835
• Clarified langauge about the supported VREF options	887
• Updated configuration note	950
• Updated description of Drive Strength	951
• Clarified IrDA encoding and decoding	1064
• Corrected clock scaling description example	1090
• Added note regarding clock signal integrity	1097
• Updated description of Radio RAMs	1167

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

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