ADuCM330/ADuCM331 Hardware Reference Manual

UG-716

One Technology Way • P.O. Box 9106 • Norwood, MA 02062-9106, U.S.A. • Tel: 781.329.4700 • Fax: 781.461.3113 • www.analog.com

Integrated, Precision Battery Sensor (ADuCM330/ADuCM331)

SCOPE

This reference manual provides a detailed description of the functionality and features of the ADuCM330/ADuCM331.

The information is relevant for Silicon Revision L6x, where x represents a number between 0 and 9.

Full specifications on the ADuCM330/ADuCM331 are available in the product data sheet, which should be consulted in conjunction with this reference manual when working with the devices.

FUNCTIONAL BLOCK DIAGRAM

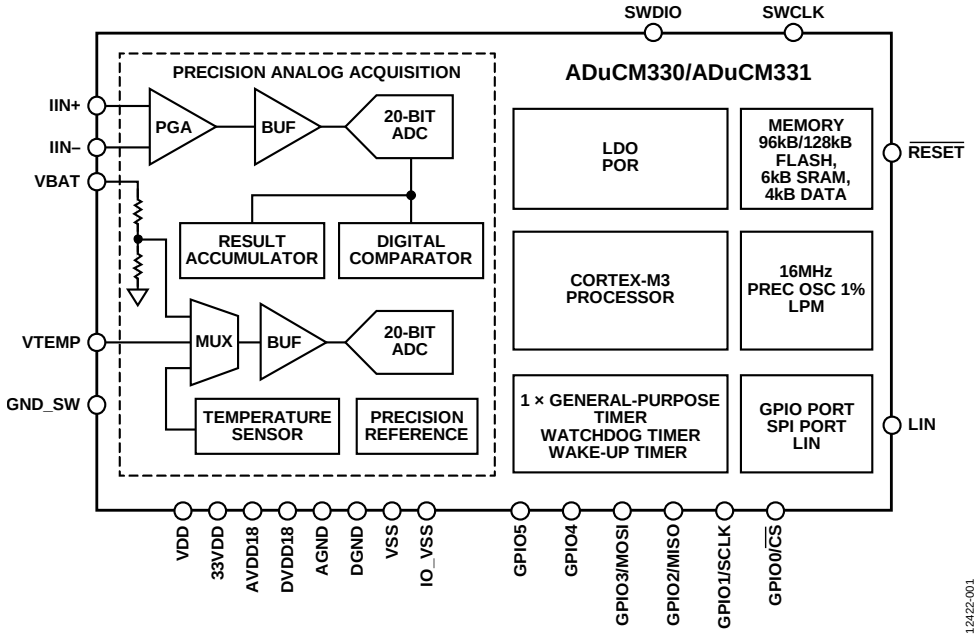

Figure 1. ADuCM330/ADuCM331 Block Diagram

TABLE OF CONTENTS

REVISION HISTORY

11/2018—Rev. C to Rev. D

Changes to Power-Down Mode Section and Modifying ADC Settings Section.....51
Added ADC Power-Up Sequence Section.....51

11/2017—Rev. B to Rev. C

Added Modifying ADC Settings Section.....51

7/2015—Revision B: Initial Version

USING THE ADuCM330/ADuCM331 HARDWARE REFERENCE MANUAL

NUMBER NOTATIONS

Table 1.

Notation	Description
Bit N	Bits are numbered in little endian format, that is, the least significant bit of a number is referred to as Bit 0.
V[x:y]	Bit field representation for Bit x to Bit y of a value or a field (V).
0xNN	Hexadecimal (Base 16) numbers are preceded by the prefix 0x.
0bNN	Binary (Base 2) numbers are preceded by the prefix 0b.
NN	Decimal (Base 10) numbers are represented using no additional prefixes or suffixes.

REGISTER ACCESS CONVENTIONS

Table 2.

Mode	Description
RW	Memory location has read and write access.
R	Memory location is read access only. A read always returns 0, unless otherwise specified.
W	Memory location is write access only.

ACRONYMS AND ABBREVIATIONS

Table 3.

Acronym/Abbreviation	Description
Σ-Δ	Sigma-delta
ADC	Analog-to-digital converter
ARM	Advanced RISC machine
CD	Clock divider
CRC	Cyclic redundancy check
HDR	High data retention
IBS	Intelligent battery sensor
JTAG	Joint test action group
LDO	Low dropout
LIN	Local interconnect network
LSB	Least significant byte/bit
MCU	Microcontroller
MMR	Memory mapped register
MSB	Most significant byte/bit
NAD	Node address for diagnostic
NVIC	Nested vectored interrupt controller
PGA	Programmable gain amplifier
PID	Protected identifier
PMU	Power management unit
POR	Power-on reset
RISC	Reduced instruction set computer
Rx	Receive
SPI	Serial peripheral interface
SWD	Serial wire debug
Tx	Transmit

ADuCM330/ADuCM331 OVERVIEW

The ADuCM330/ADuCM331 are fully integrated, 8 kSPS, data acquisition systems incorporating dual, high performance, multichannel, Σ-Δ analog-to-digital converters (ADCs), 32-bit ARM Cortex™-M3 core, and flash memory. The ADuCM330 has 96 kB program flash. The ADuCM331 has 128 kB program flash. Both devices have 4 kB data flash.

The ADuCM330/ADuCM331 can be used in many systems requiring voltage, current, and temperature sensing with processing power, but are designed as optimal and complete system solutions for shunt-based battery monitoring in automotive and nonautomotive applications. The ADuCM330/ADuCM331 integrate all of the required features to precisely and intelligently monitor, process, and diagnose 12 V battery parameters including battery current, voltage, and temperature over a wide range of operating conditions.

Minimizing external system components, the devices are powered directly from a 12 V battery. On-chip, low dropout (LDO) regulators generate the supply voltage for two integrated Σ-Δ ADCs. The ADCs precisely measure battery current, voltage, and temperature to characterize the state of health and the charge of the car battery.

The devices operate from an on-chip, 16 MHz, high frequency oscillator that supplies the system clock. This clock is routed through a programmable clock divider from which the core clock operating frequency is generated. The devices also contain a 32 kHz oscillator for low power operation.

The analog subsystem consists of an ADC with a PGA to allow various current and voltage ranges to be monitored. It also includes a precision reference on chip.

The ADuCM330/ADuCM331 also integrate a range of on-chip peripherals that can be configured under core software control as required in the application. These peripherals include an SPI serial input/output communication controller, six GPIO pins, one general-purpose timer, a wake-up timer, and a watchdog timer.

The ADuCM330/ADuCM331 are specifically designed to operate in battery powered applications where low power operation is critical. The microcontroller core can be configured in a normal operating mode, resulting in an overall system current consumption of <18.5 mA when all peripherals are active. The devices can also be configured in a number of low power operating modes under direct program control, consuming <100 μA. The ADuCM330/ADuCM331 also include a LIN physical interface for single wire, high voltage communications in automotive environments.

The devices operate from an external 3.6 V to 18 V (on VDD, Pin 26) voltage supply and are specified over a temperature range of −40°C to +115°C, with additional typical specifications at +115°C to +125°C.

MAIN FEATURES OF THE ADuCM330/ADuCM331

- High precision ADCs
- Dual channel, simultaneous sampling, 20-bit, Σ-Δ ADCs (thus minimizing gain switching)
- Programmable ADC conversion rate from 1 Hz to 8 kHz
- On-chip, ±5 ppm/°C voltage reference
- Current channel
  - Fully differential, buffered input
  - Programmable gain
  - ADC input range: −200 mV to +300 mV
  - Digital comparator with current accumulator feature
- Voltage channel
  - Buffered, on-chip attenuator for 12 V battery input
- Temperature channel
  - External and on-chip temperature sensor options

Microcontroller

- ARM Cortex-M3 32-bit processor
- Serial wire download and debug
- 16 MHz oscillator (16.384 MHz)
- 32 kHz (32.768 kHz) oscillator for low power operation

Memory

- 96 kB (ADuCM330) or128 kB (ADuCM331) program Flash/EE memory options
- 4 kB data Flash/EE memory
- 6 kB SRAM
- 10,000 cycle Flash/EE endurance
- 20 year Flash/EE retention
- In-circuit download via serial wire and LIN
- Error correction code (ECC) available on all Flash and SRAM memories

Power

- Operates directly from a 12 V battery supply
- Power consumption
  - See the ADuCM330/ADuCM331 data sheet for exact power consumption
  - Low power monitor mode

On-Chip Peripherals

- On-chip power-on reset
- One general-purpose timer
- Wake-up timer
- Watchdog timer
- GPIO port
- SPI serial input/output
- High voltage interface
- LIN 2.2/SAE J2602-2

Packages and Temperature Range

- 32-lead, 6 mm × 6 mm LFCSP
- Fully specified for −40°C to +115°C operation; additional specifications for 115°C to 125°C

CORTEX-M3 CORE

CORTEX-M3 CORE FEATURES

High Performance

- 1.25 DMIPS/MHz.
- Many instructions, including multiply, are single cycle.
- Separate data and instruction busses allow simultaneous data and instruction accesses to be performed.
- Optimized for single-cycle flash usage.

Low Power

- Low standby current.
- Core implemented using advanced clock gating; therefore, only the actively used logic consumes dynamic power.
- Two sleep modes (sleep mode and deep sleep mode) provide different levels of power saving. The SLEEPDEEP bit of the system control register selects which sleep mode is used.

Advanced Interrupt Handling

- The nested vectored interrupt controller (NVIC) supports up to 240 interrupts. The vectored interrupt feature greatly reduces interrupt latency because additional software is not needed to determine which interrupt handler to serve. In addition, additional software is not needed to set up nested interrupt support.
- The Cortex-M3 processor automatically pushes registers onto the stack at entry interrupt and retrieves them at the exit interrupt. This process reduces interrupt handling latency and allows interrupt handlers to be normal C functions.
- Dynamic priority controls for each interrupt.
- Latency reduction using late arrival interrupt acceptance and tail chain interrupt entry.
- Immediate execution of a nonmaskable interrupt request for safety critical applications.

System Features

- Support for bit band operation and unaligned data access.
- Advanced fault handling features include various exception types and fault status registers.

Debug Support

- Serial wire debug interfaces (SW-DP).
- Flash patch and breakpoint (FPB) unit for implementing breakpoints. Limited to two active breakpoints.
- Data watchpoint and trigger (DWT) unit for implementing watchpoints, trigger resources, and system profiling. Limited to one active watchpoint.

CORTEX-M3 CORE OVERVIEW

The ADuCM330/ADuCM331 contain an embedded ARM Cortex-M3 processor, Revision r2p0 (AT420). The ARM Cortex-M3 provides a high performance, low cost platform that meets the system requirements of minimal memory implementation, reduced pin count, and low power consumption while delivering outstanding computational performance and exceptional system response to interrupts.

CORTEX-M3 CORE OPERATION

Several Cortex-M3 components are flexible in their implementation. The following sections detail the actual implementation of these components in the ADuCM330/ADuCM331.

Serial Wire Debug (SW/JTAG-DP)

The ADuCM330/ADuCM331 only support the serial wire interface via the SWCLK and SWDIO pins. These devices do not support the 5-wire JTAG interface.

ROM Table

The ADuCM330/ADuCM331 implement the default ROM table.

Nested Vectored Interrupt Controller Interrupts (NVIC)

The Cortex-M3 processor includes an interrupt controller, the NVIC. The NVIC is closely coupled with the processor core and provides a number of features:

- Nested interrupt support
- Vectored interrupt support
- Dynamic priority changes support
- Interrupt masking

In addition, the NVIC also has a nonmaskable interrupt (NMI) input. The NVIC is implemented on the ADuCM330/ADuCM331 and is described in more detail in the Nested Vectored Interrupt Controller section.

Wake-Up Interrupt Controller (WIC)

Analog Devices, Inc., has implemented a modified WIC, which provides the lowest possible power-down current.

More information is available in the Power Management Unit section.

Note that if the device enters a power saving mode when servicing an interrupt, the device can only be woken up by a higher priority interrupt source.

RELATED DOCUMENTS

- Cortex-M3 Devices, Generic User Guide, ARM DUI 0552A (ID121610)
- Cortex-M3 Revision r2p0 Technical Reference Manual (DDI 0337)
- ARM Errata Notice – Cortex-M3/Cortex-M3 with ETM (AT420/AT425)
- ARMv7-M Architecture Reference Manual (DDI 0403)
- ARM Processor Cortex-M3 (AT420) and Cortex-M3 with (ETM AT425): Errata Notice
- ARM Debug Interface v5 (IHI 0031)

EXCEPTIONS AND INTERRUPTS

CORTEX-M3 AND FAULT MANAGEMENT

The Cortex-M3 supports a number of system exceptions and peripherals/external interrupts, as shown in Table 4 and Table 5.

Table 4. System Exceptions

Number	Type	Priority	Description
1	Reset	−3 (highest)	Any reset
2	Reserved	N/A[1]	
3	Hard fault	−1	All fault conditions, if the corresponding fault handler is not enabled
4	Memory management fault	Programmable	Memory management fault; access to illegal locations
5	Bus fault	Programmable	Pre-fetch fault, memory access fault, and other address/memory related faults
6	Usage fault	Programmable	Faults such as undefined instruction executed or illegal state transition attempt
[7:10]	Reserved	N/A[1]	
11	SVCALL	Programmable	System service call with SVC instruction
12	Debug monitor	Programmable	Debug monitor (breakpoint, watchpoint, or external debug requests)
13	Reserved	N/A[1]	
14	PENDSV	Programmable	Pendable request for system service
15	SYSTICK	Programmable	System tick timer

[1] N/A means not applicable.

NESTED VECTORED INTERRUPT CONTROLLER

The ADuCM330/ADuCM331 interrupts are controlled by the NVIC, and four levels of priority are available. NVIC interrupts can be enabled and disabled by writing to their corresponding interrupt set-enable or interrupt clear-enable register bit field. Only a limited number of interrupts can wake up the core from low power hibernate mode. These interrupts are described in Table 5. When the ADuCM330/ADuCM331 are woken up from any mode, they return to active mode.

Table 5. Interrupt Vectors

Position Number	Vector	Wake Up Core from Hibernate Mode
0	Wake-up timer	Yes
1	External Interrupt 0 (P0.3)	Yes
2	External Interrupt 1 (P0.4)	Yes
3	Watchdog timer	Yes
4	General-Purpose Timer 0	No
5	ADC	No
6	Flash	No
7	SPI	No
8	LIN0: LIN header or frame interrupt	No
9	LIN1: LIN error detected interrupt	No
10	LIN2: LIN sleep or wake-up event interrupt	Yes
11	High voltage interface interrupt	No
12	CALOSC: oscillator calibration interrupt	No
13	SRAM ECC interrupt	No

For the ADuCM330/ADuCM331, each interrupt can have eight levels of priority. The priority levels are 0 to 7, where 0 is the highest priority and 7 is the lowest priority. Internally, the highest user programmable priority (0) is treated as fourth priority, after a reset, an NMI, and a hardware fault. Note that 0 is the default priority for all the programmable priorities.

If the same priority level is assigned to two or more interrupts, their hardware priority (the lower the position number, the higher the priority, as shown in Table 5) determines the order in which the processor activates them. For example, if both the ADC and SPI are Priority Level 1, the ADC has higher priority.

SETTING INTERRUPT PRIORITIES

The Cortex-M3 IPR0 to IPR3 registers control the interrupt priority settings for the ADuCM330/ADuCM331. These registers can be adjusted by the user to change the default interrupt priority and to create a user specific interrupt vector table to suit the user's application.

Every interrupt has 8 possible priority levels, with 0 being the highest priority and 7 being the lowest priority setting. Each interrupt priority register supports four interrupt sources.

Table 6 uses IPR0 as an example to explain the relevant bits.

Table 6. Bit Descriptions for the Interrupt Priority Registers, IPR0

Bits	Name	Description
[31:29]	WATCHDOG TIMER	000: highest interrupt priority level for watchdog timer 111: lowest interrupt priority level
[28:24]	RESERVED	Reserved
[23:21]	EXTINT1	000: highest interrupt priority level for external IRQ1 111: lowest interrupt priority level
[20:16]	RESERVED	Reserved
[15:13]	EXTINT0	000: highest interrupt priority level for external IRQ0 111: lowest interrupt priority level
[12:8]	RESERVED	Reserved
[7:5]	WAKE-UP TIMER	000: highest interrupt priority level for wake-up timer 111: lowest interrupt priority level
[4:0]	RESERVED	Reserved

Note that the IPR1 to IPR3 interrupt priority registers are configured similarly, with default priority according to Table 5.

CORTEX-M3 AND NVIC REGISTERS

The set-enable register and the clear-enable register enable and disable the interrupts. The set-pending register and the clear-pending register pend the interrupts. These registers use a write 1 to enable and a write 1 to clear policy. Each bit in the interrupt set enable register corresponds to 1 of 32 interrupts. Setting a bit in the interrupt set-enable register enables the interrupt.

When the enable bit of a pending interrupt is set in the set-pending register, the processor activates the interrupt based on its priority; however, if the corresponding bit is clear in the set-enable register, asserting the interrupt signal pends the interrupt, but it is not possible to activate the interrupt, regardless of its priority.

Pend the interrupt means that, if the interrupt occurs, it is possible for the user code to investigate the set-pending register and verify whether the interrupt occurs. Therefore, a disabled interrupt can serve as a latched general-purpose bit.

Reading and clearing interrupts occur without invoking an interrupt.

Clear the enable state by writing a 1 to the corresponding bit in the interrupt clear-enable register. This write also clears the corresponding bit in the interrupt set-enable register. Writing to the interrupt clear-pending register has no effect on an interrupt that is active, unless the interrupt is also pending. If an interrupt is active when it is disabled, it remains in its active state until cleared by reset or an exception return.

The NVIC is an integral part of the Cortex-M3 microprocessor. An interrupt generated from peripherals is notified to the NVIC within one clock cycle of the clock used by this peripheral, unless otherwise noted.

Table 7. Cortex-M3 and NVIC Registers

Address	Analog Devices Header File Name	Description	Access
0xE000E004	ICTR	Shows the number of interrupt lines that the NVIC supports	Read
0xE000E010	STCSR	SYSTICK control and status register	Read/write
0xE000E014	STRVR	SYSTICK reload value register	Read/write
0xE000E018	STCVR	SYSTICK current value register	Read/write
0xE000E01C	STCR	SYSTICK calibration value register	Read
0xE000E100	ISER0	Set IRQ0 to IRQ13 enable	Read/write
0xE000E180	ICER0	Clear IRQ0 to IRQ13 enable	Read/write
0xE000E200	ISPR0	Set IRQ0 to IRQ13 pending	Read/write
0xE000E280	ICPR0	Clear IRQ0 to IRQ13 pending	Read/write

Address	Analog Devices Header File Name	Description	Access
0xE000E300	IABR0	IRQ0 to IRQ13 active bits	Read/write
0xE000E400	IPR0	IRQ0 to IRQ3 priority	Read/write
0xE000E404	IPR1	IRQ4 to IRQ7 priority	Read/write
0xE000E408	IPR2	IRQ8 to IRQ11 priority	Read/write
0xE000E40C	IPR3	IRQ12 to IRQ13 priority	Read/write
0xE000ED00	CPUID	CPUID base register	Read
0xE000ED04	ICSR	Interrupt control and status register	Read/write
0xE000ED08	VTOR	Vector table offset register	Read/write
0xE000ED0C	AIRCR	Application interrupt/reset control register	Read/write
0xE000ED10	SCR	System control register	Read/write
0xE000ED14	CCR	Configuration control register	Read/write
0xE000ED18	SHPR1	System Handlers Register 1	Read/write
0xE000ED1C	SHPR2	System Handlers Register 2	Read/write
0xE000ED20	SHPR3	System Handlers Register 3	Read/write
0xE000ED24	SHCSR	System handler control and state	Read/write
0xE000ED28	CFSR	Configurable fault status	Read/write
0xE000ED2C	HFSR	Hard fault status	Read/write
0xE000ED34	MMAR	Memory manage address	Read/write
0xE000ED38	BFAR	Bus fault address	Read/write
0xE000EF00	STIR	Software trigger interrupt register	Write

EXTERNAL INTERRUPT CONFIGURATION

Two external interrupts are implemented. The interrupts can be separately configured to detect any combination of the following type of events:

- Rising edge. The logic detects a transition from low to high and generates a pulse. Only one pulse is sent to the Cortex-M3 per rising edge.
- Falling edge. The logic detects a transition from high to low and generates a pulse. Only one pulse is sent to the Cortex-M3 per falling edge.
- Rising or falling edge. The logic detects a transition from low to high or from high to low and generates a pulse. Only one pulse is sent to the Cortex-M3 per edge.
- High level. The logic detects a high level. The appropriate interrupt is asserted and sent to the Cortex-M3. The interrupt line is held asserted until the external source deasserts. The high level must be maintained for one core clock cycle minimum to be detected.
- Low level. The logic detects a low level. The appropriate interrupt is asserted and sent to the Cortex-M3. The interrupt line is held asserted until the external source deasserts. The low level must be maintained for one core clock cycle minimum to be detected.

The external interrupt detection unit block is in the always on section and allows external interrupt to wake up the device when in hibernate mode.

EXTERNAL INTERRUPT CONFIGURATION MEMORY MAPPED REGISTERS

The interrupt detection unit consists of memory mapped registers (MMRs) contained in the always on section; the MMRs are based at Address 0x40002400.

Table 8. Interrupt Detection Unit Memory Mapped Registers (Base Address 0x40002400)

Offset	Name	Description	Access	Default
0x0020	EI0CFG	External Interrupt Configuration Register 0	RW	0x0000
0x0030	EICLR	External interrupt clear register	RW	0x0000

External Interrupt Configuration Register 0

Address: 0x40002420, Reset: 0x0000, Name: EI0CFG

Table 9. EI0CFG Register Bit Descriptions

Bits	Name	Description	
[15:8]	RESERVED	Reserved	
7	IRQ1EN	External Interrupt 1 enable bit 0: External Interrupt 1 disabled 1: External Interrupt 1 enabled	
[6:4]	IRQ1MDE	External Interrupt 1 mode registers	
		EI0CFG[6:4]	**Description**
		000	Rising edge
		001	Falling edge
		010	Rising or falling edge
		011	High level
		100	Low level
		101	Falling edge (same as 001)
		110	Rising or falling edge (same as 010)
		111	High level (same as 011)
3	IRQ0EN	External Interrupt 0 enable bit 0: External Interrupt 0 disabled 1: External Interrupt 0 enabled	
[2:0]	IRQ0MDE	External Interrupt 0 mode registers	
		EI0CFG[2:0]	**Description**
		000	Rising edge
		001	Falling edge
		010	Rising or falling edge
		011	High level
		100	Low level
		101	Falling edge (same as 001)
		110	Rising or falling edge (same as 010)
		111	High level (same as 011)

External Interrupt Clear Register

Address: 0x40002430, Reset: 0x0000, Name: EICLR

Table 10. EICLR Register Bit Descriptions

Bits	Name	Description
[15:2]	RESERVED	Reserved
1	IRQ1	External Interrupt 1 clear bit 0: cleared by software 1: clear External Interrupt 1 flag
0	IRQ0	External Interrupt 0 clear bit 0: cleared by software 1: clear External Interrupt 0 flag

Ensure that the register write has fully completed before returning from the interrupt handler. Use the data synchronization barrier (DSB) instruction if necessary.

POWER MANAGEMENT UNIT

POWER MANAGEMENT UNIT FEATURES

Three power modes are available:

- Active
- SYSHALT
- Hibernate

Additionally, the Cortex-M3 has two power saving modes:

- Sleep mode. This mode stops the system clock (FCLK).
- Deep sleep mode. This mode stops the system clock (FCLK) and, in conjunction with the PMU, switches off some circuitry such as the UCLK, the HFOSC, and the HP LDO.

POWER MANAGEMENT UNIT OVERVIEW

The power management unit (PMU) controls the power modes of the ADuCM330/ADuCM331. The Cortex-M3 sleep modes are linked to the PMU modes and are described in this section.

The PMU is in the always on section. Three power modes are available. Each mode provides a power reduction benefit with a corresponding reduction in functionality. Hibernate mode provides a power-down figure of <100 μA, and SYSHALT mode provides a power-down figure of 1 mA. Table 11 lists all the power modes available. For active mode and hibernate mode, estimated current values and wake-up times are available in the ADuCM330/ADuCM331 data sheet.

Table 11. System Power Mode Summary

Mode	Description
Active	Default
SYSHALT	Gate both HCLK and PCLK when Cortex-M3 is in sleep modes
Hibernate	Gate power to flash block

The WFI instruction places the Cortex-M3 in sleep mode. If deep sleep mode is enabled in the system control register of the Cortex-M3 when issuing the WFI instruction, the Cortex-M3 enters deep sleep mode; otherwise, it enters sleep mode.

Note that enabling deep sleep mode in SYSHALT mode has no effect. Not enabling deep sleep mode in hibernate mode leaves the ADuCM330/ADuCM331 in active mode. Before entering SYSHALT mode with ADCs active, deep sleep (not sleep) mode must be enabled in the Cortex-M3 system control register.

The PMU and Cortex-M3 power modes are summarized in Table 12.

The SPI peripheral must be disabled before entering low power modes where PCLK is disabled. Disabling the peripheral resets the state machine of these peripherals while keeping their configuration.

The debugger must be disconnected to achieve lower power performance.

The following is a typical code example for achieving low power mode (hibernate mode):

```
SCR = 0x04;                          // Enable deep sleep mode in the core
PWRKEY = 0x4859;
PWRKEY = 0xF27B;                     // PWRMOD keys
PWRMOD = 0x5;                        // Hibernate Mode
__dsb();                             // Wait until all memory accesses complete
__wfi();                             // Wait for interrupt
```

POWER MANAGEMENT UNIT LDO SWITCHING

Three LDO regulators are integrated on the ADuCM330/ADuCM331.

- The high voltage LDO (HV LDO) regulates voltages from 12 V to 3.3 V.
- The high power LDO (HP LDO) regulates voltages from 3.3 V to 1.8 V and supplies the device in active mode.
- The low power LDO (LP LDO) regulates voltages from 3.3 V to 1.8 V and achieves an extremely low quiescent current. However, it can only supply very low load currents (<50 μA). It is used in hibernate mode.

Switching between the LDOs is done automatically.

POWER MANAGEMENT UNIT POWER MODES OPERATION

Power Mode: Active Mode

The system is fully active. None of the clocks described in Figure 3 are gated. Memories and all user enabled peripherals are clocked, and the Cortex-M3 executes instructions. The Cortex-M3 manages its internal clocks and can be in a partial clock gated state. The clock gating only affects the internal Cortex-M3 processing core. Automatic clock gating is used on all blocks and is transparent to the user. User code can use a WFI command to place the Cortex-M3 in sleep mode; the WFI command is independent of the power mode settings of the PMU. When the ADuCM330/ADuCM331 wake up from any of the low power modes, the devices return to active mode.

Power Mode: SYSHALT Mode

The system gates HCLK and PCLK at an early stage after the Cortex-M3 enters sleep mode. The Cortex-M3 FCLK is active, and the NVIC wakes up the device.

Power Mode: Hibernate Mode

The system gates power to the digital flash memory. All states are retained during this power gating. Hibernate mode appears to the user as a clock gating of FCLK, HCLK, and PCLK at an early stage after the Cortex-M3 has entered deep sleep mode, but with a lower leakage current. There is a response time difference; the device is slower to come up. The Cortex-M3 FCLK is thus stopped, and only the peripherals listed in Table 5 are able to wake up the Cortex-M3.

Table 12. Power Modes Summary

Clock and Power	Active	SYSHALT	Hibernate
HP LDO	On	On	Off
LP LDO	Off	Off	On
HFOSC	On	On	Off
LFOSC	On	On	On
Power gate	On	On	Off
UCLK	On	On	Off
FCLK	On	On	Off
HCLK	On	Off	Off
PCLK	On	Off	Off
ACLK	On	On	Off
SRAM	On	On	On
Cortex-M3	Active	Sleep	Deep sleep

Note that the debugger must be disconnected to achieve low power operation.

POWER SUPPLY SUPPORT CIRCUITS

The ADuCM330/ADuCM331 each incorporates three on-chip, low dropout (LDO) regulators. One regulator is driven directly from the battery voltage to generate a 3.3 V internal supply. This 3.3 V supply is used as the supply for two internal 1.8 V LDOs. These two 1.8 V LDOs operate independently to allow normal mode (high power LDO) and low power mode (low power LDO).

The high power LDO (HP LDO) functions with two output capacitors (0.47 µF) on DVDD and AVDD.

The effective series resistance (ESR) of the output capacitor affects the stability of the LDO control loop. A capacitor with a low ESR is recommended to ensure the stability of the regulators.

In addition, the power-on reset (POR) function is integrated to ensure safe operation of the processor, as well as continuous monitoring of the battery power supply.

The POR circuit operates with the external circuit shown in the Recommended Schematic section of the ADuCM330/ADuCM331 data sheet.

As shown in Figure 2, when the supply voltage on VDD reaches a typical operating voltage of 3.1 V, a POR signal keeps the Cortex-M3 processor in a reset state for 18 ms. This delay ensures that the regulated power supply voltage (DVDD33) applied to the Cortex-M3 processor and its associated peripherals is greater than the minimum operational voltage, thereby guaranteeing full functionality. A POR flag is set in the RSTSTA MMR to indicate that a POR reset event has occurred.

At voltages below the POR level voltage, the SRAM can be corrupted. When the SRAM implements ECC, this corruption can be detected by the kernel. If the SRAM is corrupted, the kernel initializes the entire SRAM to 0x00 and corrects all ECC bits. This initialization process doubles the length of time spent in reset. If the SRAM is not corrupted, the device exits reset in 1.25 ms typically.

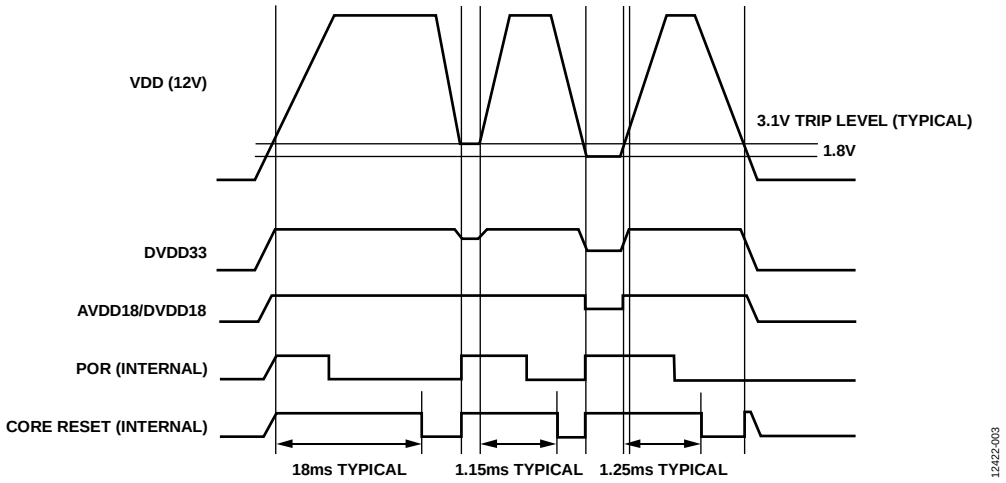

Figure 2. Power Supply Diagram

POWER MODES MEMORY MAPPED REGISTERS

The power modes are controlled by a single register based in the always on section at Address 0x40002400.

Table 13. System Clocks Memory Mapped Register (Base Address 0x40002400)

Offset	Name	Description	Access	Default
0x0000	PWRMOD	Power modes register	RW	0x00
0x0004	PWRKEY	Power modes key register	RW	Not applicable

Power Modes Control Register

Address: 0x40002400, Reset: 0x00, Name: PWRMOD

Table 14. PWRMOD Register Bit Descriptions

Bits	Name	Description	
7	RESERVED	Reserved. These bits must be written 0 by user code.	
6	HFOSC_LPM	Enables high or low precision mode in the 16 MHz high frequency oscillator (HFOSC). 0: enable high precision 1% (HPOSC) 1: enable low precision 3% (LPOSC)	
[5:3]	RESERVED	Reserved. These bits must be written 0 by user code.	
[2:0]	MOD	Power modes control bits. These bits select the power mode to enter. When read, these bits contain the last power mode value entered by user code.	
		MOD	**Description**
		000	Active (normal mode)
		011	SYSHALT
		101	Hibernate
		Other	Reserved

To place the Cortex-M3 in deep sleep mode for hibernate mode, Bit 2 (SLEEPDEEP) in the Cortex-M3 system control register (Address 0xE000ED10) must be 1.

Power Modes Key Register

Address: 0x40002404, Reset: Not applicable, Name: PWRKEY

Table 15. PWRKEY Register Bit Descriptions

Bits	Name	Description
[15:0]	VALUE	Power control key register. The PWRMOD register is key protected. Two writes to the key are necessary to change the value in the PWRMOD register: 0x4859, followed by 0xF27B. Following these writes, the PWRMOD register can be written.

SYSTEM CLOCKS

SYSTEM CLOCK FEATURES

The ADuCM330/ADuCM331 integrate two internal clock sources:

- HFOSC is a 16.384 MHz oscillator that can operate in two modes:
  - High precision (HPOSC 1%) internal oscillator.
  - Low precision (LPOSC 3%) internal oscillator.
- LFOSC is a 32.768 kHz, 5% low power, internal oscillator.

SYSTEM CLOCK OVERVIEW

The CLKCON MMR controls the clocking source to the ADuCM330/ADuCM331. It does not control which clocks are enabled.

One of the outputs of the clock generation circuit is directed through the clock divider circuit (CD bits), where this clock can be divided down to a minimum of 125 kHz (for HFOSC) by the user. This clock is referred to as the core clock within the ADuCM330/ADuCM331, and is the clock that drives the Cortex-M3. This clock is directed even further to all the digital peripherals in the system (PCLK), the NVIC (FCLK), and the ADC (ACLK).

Internally, the system clock is divided into five clocks:

- UCLK system clock
- FCLK for the core
- HCLK for the flash, SRAM
- PCLK for the LIN, SPI, HV interface and timers
- ACLK for the ADC interface

Figure 3 shows all the clocks available and includes clock gates for power management. More information on the clock gates is available in the Power Management Unit section.

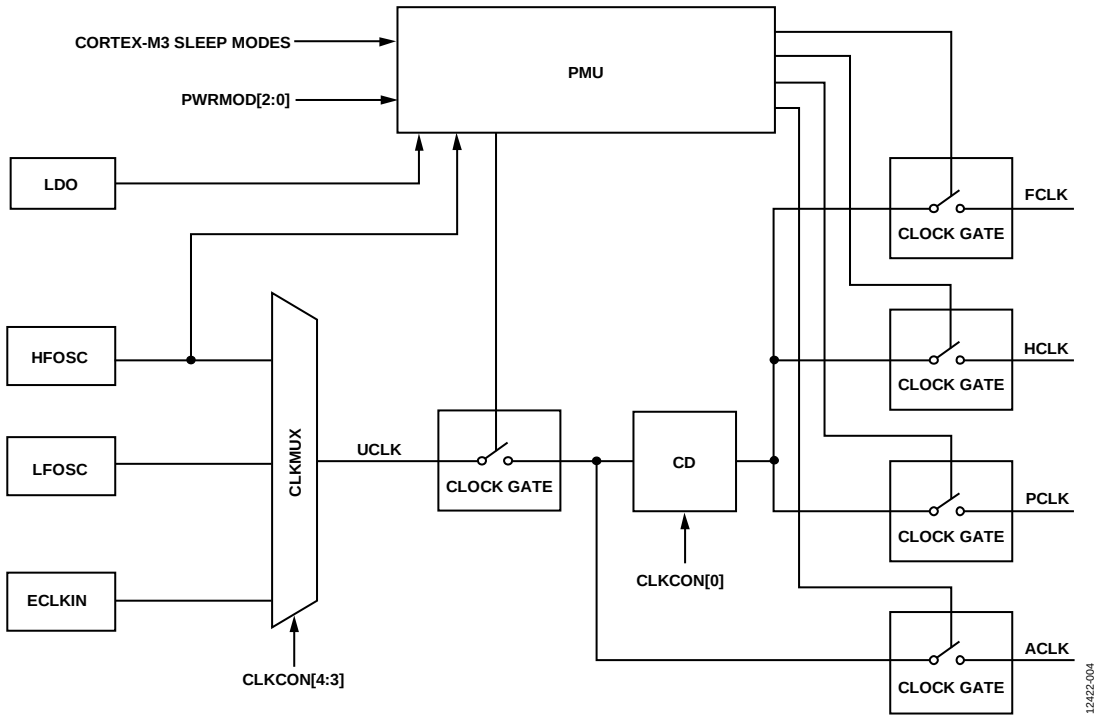

Figure 3. System Clock Architecture Block Diagram

SYSTEM CLOCKS OPERATION

At power-up, the core executes from the HFOSC internal oscillator in high precision mode. User code can select the clock source for the system clock and can divide the clock by a factor of 2. Dividing the clock allows slower code execution and reduced power consumption. UCLK is also passed to some of the serial peripherals so that the timings are not affected by CD changes.

SYSTEM CLOCKS MEMORY MAPPED REGISTERS

Table 16. Clock Control Memory Mapped Register

Address	Name	Description	Access	Default
0x40002000	CLKCON	System clocks control register	RW	0x00E0

System Clocks Control Register

Address: 0x40002000, Reset: 0x00E0, Name: CLKCON

Table 17. CLKCON Register Bit Descriptions

Bits	Name	Description
[15:8]	RESERVED	Write 0x00 to these bits.
[7:5]	RESERVED	Write 0x7 to these bits.
[4:3]	CLKMUX	Clock in multiplexer selection bits. 00: HFOSC (default) 01: HFOSC 10: LFOSC 11: ECLKIN (P0.4)
[2:1]	RESERVED	Write 0 to these bits.
0	CD	Clock divide bit. 0: UCLK 1: UCLK ÷ 2

HIGH FREQUENCY OSCILLATOR (HFOSC) CALIBRATION

The ADuCM330/ADuCM331 on-chip, 16 MHz oscillator (HFOSC) can be calibrated using the LIN interface

A number of MMR registers control the trimming in normal mode and power-down mode (see Table 18). These registers are protected and are only writable following a write to the key register, LINCALOCK. The LINCALSTA register is a read only register that shows which mode is currently active. This information is held in LINCALSTA[0]. LINCALSTA[2:1] can also be used to determine if the TRIM values have been altered by the system since the LINCALSTA register was last read. HFOSC can be trimmed in either low or high precision mode (set by PWRMOD[6]).

There are two modes of trimming: user trim mode and system trim mode.

User Trim Mode

User trim mode allows any trim register values to be written and downloaded by the user. This should always be the first trim mode. Factory calculated trim values are automatically contained within trim registers as the default.

The required value is written to the LINCALVAL0 and LINCALVAL1 registers. In user trim mode, these values automatically match the values in the LINCAL2 and LINCAL3 registers. The values in the LINCAL2 and LINCAL3 read only registers are sent to the oscillator.

LINCALVA0 and LINCALVAL1 are key protected and require two sequential writes: write the unlock key to the LINCALOCK register followed by the desired trim value to LINCALVAL0 or LINCALVAL1.

An example of how to program a user trim value follows.

```
LINCALOCK = LIN_CAL_KEY;                    // Unlock key protection
LINCALVAL0 = 0x7B;                          // Write trim values 0x7B to trim HFOSC LP Mode
LINCALOCK = LIN_CAL_KEY;                    // Unlock key protection
LINCALVAL1 = 0x200;                         // Write trim values 0x200 to trim HFOSC HP Mode
```

Note that it is not advisable to write values to these register unless it is fully understood what effect such a write has on the oscillator frequency. Invalid values can result in corruption of flash data when written.

System Trim Mode

System trim mode allows a calibration of the clock relative to the LIN baud rate of the system. This mode can be used in situations where the user wants greater accuracy from either the normal mode or low power mode. System trim mode cannot operate if there are no LIN communications. The ADuCM330/ADuCM331 automatically adapt to download internally calculated trim values to operate within >1% accuracy over the full operating range, as long as there is at least 1 LIN transaction for every 10°C of change.

After the user trim mode is set at startup, setting the device into system trim mode overrides the user set trim values with iteratively calculated values derived from LIN communications.

Using the LIN baud rate, the device determines how accurate the trim value is and automatically increments or decrements a step each time a valid LIN communication occurs within a set window of calibration. The calibration window is defined from values set in the LINCALMINL/LINCALMINH and LINCALMAXL/LINCALMAXH registers, and from the number of steps defined in the LINCALCON register (key protected). The user sets these values. In system trim mode, LINCAL2 and LINCAL3 may not match LINCAL0 and LINCAL1.

The following is a typical sequence for starting system trim mode:

```
LINCALMIN = EXPECTED_LINBR_VALUE-0x20;              //Define tolerance
LINCALMAX = EXPECTED_LINBR_VALUE+0x20;              //Define tolerance
LIN_CAL_LOCK = LIN_CAL_KEY;                         // unlock key protection
LINCALCON = 0x1;                                    // Enable LINCAL mode with step size = 1
```

HFOSC CALIBRATION MEMORY MAPPED REGISTERS

Table 18. HFOSC Calibration Memory Mapped Registers (Base Address 0x40005C00)

Offset	Name	Description	Access	Default[1]
0x00	LINCALCON	LIN calibration control register	RW	0x0000
0x1C	LINCALSTA	System calibration status	R	0x0000
0x04	LINCALVAL0	User calibration value (low precision mode)	RW	0x0000
0x08	LINCALVAL1	User calibration value (high precision mode)	RW	0x0000
0x20	LINCALVAL2	System calibration value (low precision mode)	R	0xXXX
0x24	LINCALVAL3	System calibration value (high precision mode)	R	0xXXX
0x14	LINCALMINL	Minimum control window [15:0]	RW	0x0000
0x18	LINCALMINH	Minimum control window [18:16]	RW	0x0000
0x0C	LINCALMAXL	Maximum control window [15:0]	RW	0x0000
0x10	LINCALMAXH	Maximum control window [18:16]	RW	0x0000
0x28	LINCALOCK	Calibration lock register	RW	0x0000

[1] X means don't care.

LIN Calibration Control Register

Address: 0x40005C00, Reset: 0x0000, Name: LINCALCON

Table 19. LINCALCON Register Bit Descriptions (Key Protected)

Bits	Name	Description	
[15:3]	RESERVED	Reserved.	
[2:1]	STEP	System mode oscillator trim step.	
		LINCALCON[2:1]	**Description**
		00	Step Size 1
		01	Step Size 2
		10	Step Size 3
		11	Step Size 4
0	CALMODE	Calibration mode. 0: user mode. This setting uses factory calibrated trim values by default. 1: system mode. For HFOSC calibration from a LIN sync field.	

System Calibration Status Register

Address: 0x40005C1C Reset: 0x0000, Name: LINCALSTA

Table 20. LINCALSTA Register Bit Descriptions

Bits	Name	Description
[15:3]	RESERVED	Reserved.
2	LPACC	This bit allows the user to monitor accuracy in low precision mode. 0: reset during a read operation 1: set when LINCALVAL2 is altered
1	HPACC	This bit allows the user to monitor accuracy in high precision mode. 0: reset during a read operation 1: set when LINCALVAL3 is altered
0	PWRMODE	Power mode selected for calibration. 0: HPOSC (high precision mode) 1: LPOSC (low precision mode)

User Calibration Value Register (Low Precision Mode)

Address: 0x40005C04, Reset: 0x0000, Name: LINCALVAL0

Table 21. LINCALVAL0 Register Bit Descriptions (Key Protected)

Bits	Name	Description
[15:9]	RESERVED	Reserved
[8:0]	LPTRIM	The 9-bit, user trim value used in low precision mode

User Calibration Value Register (High Precision Mode)

Address: 0x40005C08, Reset: 0x0000, Name: LINCALVAL1

Table 22. LINCALVAL1 Register Bit Descriptions (Key Protected)

Bits	Name	Description
[15:9]	RESERVED	Reserved.
[8:0]	HPTRIM	The 9-bit, user trim value used in high precision mode.

System Calibration Value Register (Low Precision Mode)

Address: 0x40005C20, Reset: 0xXXXX, Name: LINCALVAL2

Table 23. LINCALVAL2 Register Bit Descriptions

Bits	Name	Description
[15:9]	RESERVED	Reserved.
[8:0]	LPTRIM	The 9-bit, LINCAL trim value used by the HFOSC in low precision 3% mode.

System Calibration Value Register (High Precision Mode)

Address: 0x40005C24, Reset: 0xXXXX, Name: LINCALVAL3

Table 24. LINCALVAL3 Register Bit Descriptions

Bits	Name	Description
[15:9]	RESERVED	Reserved
[8:0]	HPTRIM	The 9-bit, LINCAL trim value used by the HFOSC in high precision 1% mode

Minimum Control Window Register [15:0]

Address: 0x40005C14, Reset: 0x0000, Name: LINCALMINL

Table 25. LINCALMINL Register Bit Descriptions

Bits	Name	Description
[15:0]	MINSYNC[15:0]	Minimum tolerance value for the HFOSC system trim

Minimum Control Window Register [18:16]

Address: 0x40005C18, Reset: 0x0000, Name: LINCALMINH

Table 26. LINCALMINH Register Bit Descriptions

Bits	Name	Description
[15:3]	RESERVED	Reserved
[2:0]	MINSYNC[18:16]	Minimum tolerance value for the HFOSC system trim

Maximum Control Window Register [15:0]

Address: 0x40005C0C, Reset: 0x0000, Name: LINCALMAXL

Table 27. LINCALMAXL Register Bit Descriptions

Bits	Name	Description
[15:0]	MAXSYNC[15:0]	Maximum tolerance value for the HFOSC system trim

Maximum Control Window Register [18:16]

Address: 0x40005C10, Reset: 0x0000, Name: LINCALMAXH

Table 28. LINCALMAXH Register Bit Descriptions

Bits	Name	Description
[15:3]	RESERVED	Reserved
[2:0]	MAXSYNC[18:16]	Maximum tolerance value for the HFOSC system trim

Calibration Lock Register

Address: 0x40005C28, Reset: 0x0000, Name: LINCALOCK

Table 29. LINCALOCK Register Bit Descriptions

Bits	Name	Description
[15:0]	LOCK	This lock register must written with the unlock code, 0x1324, immediately before the desired value is written to any key protected register.

LOW FREQUENCY OSCILLATOR (LFOSC) CALIBRATION

The accuracy of the 32 kHz oscillator (LFOSC) can be further improved using the high precision, 16 MHz oscillator (HFOSC).

The hardware counts the number of HFOSC clocks in a specified number of LFOSC clock periods, and the results are compared. If the count is longer or shorter, the oscillator trim value is incremented or decremented accordingly. This procedure must be manually iterated by the user until no further increments or decrements are possible, or until the results are outside the maximum and minimum calibration register values.

There are a number of dedicated calibration MMRs, as shown in Table 30.

At the end of the time base period, the value of EXPUCLK (TRMUCTGT[12:0]) and UCLKCNT (TRMUCCNT[12:0]) are compared within the tolerance specified by TOLSEL (TRMCON[5]). The tolerance can be set at a wider tolerance (32) or a tighter tolerance (16). If the EXPUCLK and UCLKCNT values are within this set tolerance, no further increment or decrement occurs.

Otherwise,

- If UCLKCNT > EXPUCLK + 16(1 + TOLSEL), the trim value is incremented by 1 as LFOSC runs slow.
- If UCLKCNT < EXPUCLK − 16(1 + TOLSEL), the trim value is decremented by 1 as LFOSC runs fast.

Note that TOLSEL is either 0 or 1.

The value for EXPUCLK can be calculated using

$$EXPUCLK = MAXLFOSC \times (f_{UCLK}/f_{LFOSC})$$

Note that the maximum trim value is factory set; there is no option to modify this. If the user exceeds the maximum trim value, the value reverts to factory settings. The minimum trim value is user programmable. If the 16 MHz oscillator is disabled, the current trim cycle is immediately aborted, with the trim value unaffected.

LFOSC CALIBRATION MEMORY MAPPED REGISTERS

Table 30. LFOSC Trim Memory Mapped Registers (Base Address 0x40009C00)

Offset	Name	Description	Access	Default[1]
0x0000	TRMSTA	Status register	R	0x00
0x0004	TRMCON	Control register	RW	0x00
0x0008	TRMMXC	Maximum calibration value register	R	0x3F
0x000C	TRMMNC	Minimum calibration value register	RW	0x00
0x0010	TRMVAL	Oscillator trim value	RW	0xXX
0x0014	TRM32TGT	LFOSC target count	RW	0x0
0x0018	TRM32CNT	LFOSC current count	R	0x0
0x001C	TRMUCTGT	UCLK target count	RW	0x0000
0x0020	TRMUCTCNT	UCLK current count	R	0x0000

[1] X means don't care.

LFOSC Calibration Status Register

Address: 0x40009C00, Reset: 0x00, Name: TRMSTA

These bits are automatically cleared after they are read.

Table 31. TRMSTA Register Bit Descriptions

Bits	Name	Description
[7:6]	RESERVED	Reserved.
5	DECCAL	Last calibration cycle was a decrement (even when minimum is reached or NOINCDEC is set).
4	INCCAL	Last calibration cycle was an increment (even when maximum is reached or NOINCDEC is set).
3	MINCAL	Minimum trim value has been reached (interrupt source).
2	MAXCAL	Maximum trim value has been reached (interrupt source).
1	UCNTOF	UCLK count overflow (interrupt source).
0	CYCEND	Calibration cycle ended (interrupt source). This bit does not indicate that the LFSOSC is fully trimmed, only that a single calibration cycle has completed.

LFOSC Calibration Control Register

Address: 0x40009C04, Reset: 0x00, Name: TRMCON

Table 32. TRMCON Register Bit Descriptions

Bits	Name	Description
7	CLREN	0: the enable bit (TRMCON[4]) is not automatically cleared at the end of the next calibration cycle. 1: the enable bit (TRMCON[4]) is automatically cleared at the end of the next calibration cycle.
6	NOINCDEC	0: the trim register is incremented or decremented at the end of a calibration cycle. 1: the trim register is not incremented or decremented at the end of a calibration cycle.
5	TOLSEL	Tolerance select. 0: tolerance of 16. 1: tolerance of 32.
4	ENABLE	0: disable the calibration block. 1: enable the calibration block.
3	MNIRQEN	0: disable the minimum trim value interrupt. 1: enable the minimum trim value interrupt.
2	MXIRQEN	0: disable the maximum trim value interrupt. 1: enable the maximum trim value interrupt.
1	UIRQEN	0: disable the UCLK counter overflow interrupt. 1: enable the UCLK counter overflow interrupt.
0	CIRQEN	0: disable the cycle end interrupt. 1: enable the cycle end interrupt.

Maximum Calibration Value Register

Address: 0x40009C08, Reset: 0x3F, Name: TRMMXC

Table 33. TRMMXC Register Bit Descriptions

Bits	Name	Description
[7:6]	RESERVED	Reserved.
[5:0]	MAXCAL	Maximum trim value (the value in this register is factory set and not user programmable).

Minimum Calibration Value Register

Address: 0x40009C0C, Reset: 0x00, Name: TRMMNC

Table 34. TRMMNC Register Bit Descriptions

Bits	Name	Description
[7:6]	RESERVED	Reserved.
[5:0]	MINCAL	Minimum trim value (the oscillator trim logic does not decrement below the minimum trim value specified in this register).

Oscillator Trim Value Register

Address: 0x40009C10, Reset: Calibration Value, Name: TRMVAL

Table 35. TRMVAL Register Bit Descriptions

Bits	Name	Description
[7:6]	RESERVED	Reserved.
[5:0]	VCOTRIM	Calibration trim value. This defaults to MAXCAL if the value exceeds MAXCAL. During each calibration cycle, this value is increased or decreased by one as necessary.

LFOSC Target Count Register

Address: 0x40009C14, Reset: 0x0000, Name: TRM32TGT

Table 36. TRM32TGT Register Bit Descriptions

Bits	Name	Description
[7:3]	RESERVED	Reserved.
[2:0]	MAXLFOSC	The number of LFOSC clocks to count. 0 means disable.

LFOSC Current Count Register

Address: 0x40009C18, Reset: 0x0, Name: TRM32CNT

Table 37. TRM32CNT Register Bit Descriptions

Bits	Name	Description
[7:3]	RESERVED	Reserved
[2:0]	LFOSCCNT	Stores the current number of LFOSC clocks

UCLK Target Count Register

Address: 0x40009C1C, Reset: 0x0000, Name: TRMUCTGT

Table 38. TRMUCTGT Register Bit Descriptions

Bits	Name	Description
[15:13]	RESERVED	Reserved
[12:0]	EXPUCLK	Stores the expected number of HFOSC clocks during the time base

UCLK Current Count Register

Address: 0x40009C20, Reset: 0x0000, Name: TRMUCCNT

Table 39. TRMUCCNT Register Bit Descriptions

Bits	Name	Description
[15:13]	RESERVED	Reserved
[12:0]	UCLKCNT	Stores the current number of HFOSC clocks during the time base after the calibration cycle is complete

KERNEL

The ADuCM330/ADuCM331 feature a protected, on-chip, kernel resident in the top 2 kB of the Flash/EE code space. After a reset event, the hardware calculates its own kernel checksum and compares it to the checksum programmed during production test to ensure that the kernel does not contain any errors. If an error occurs, the kernel cannot continue and stops execution.

If the checksum is correct, the kernel copies the factory calibrated data from the manufacturing data space into the various on-chip peripherals. The following peripherals are calibrated by the kernel:

- Precision oscillator
- Low power oscillator
- 33VDD, DVDD18, AVDD18
- Voltage reference
- Current ADC (offset and gain)
- Voltage/temperature ADC (offset and gain)

The following processor registers and user registers are modified by the kernel and differ from their POR default values:

- R0 to R15
- GP0CON
- FEEADR/FEEDATL/FEEDATH/FEECON0/FEECON1/FEESIGN
- HVDAT/HVCON
- HVDCFG0

The ADuCM330/ADuCM331 also feature an on-chip LIN downloader. Note that kernel entry and downloader exit can only occur via a reset. SWD-JTAG access is disabled during kernel execution. Before exiting to user code, the kernel checks if the SWD lock location of the uppermost page in flash contains the key value, 0x160320 (see Figure 9). If this key is present, SWD-JTAG access is not granted after kernel exit. If any other value is present, SWD-JTAG access is enabled. This process provides additional security to ensure that SWD access is not possible between kernel exit and user code SWD disabling.

KERNEL IMPLEMENTATION

The main sequences that the kernel follows are described in the following sections and shown in Figure 4.

Debug Mode

After a reset, the kernel initializes the device and tests the state of the GPIO5 pin. If this pin is low, the kernel branches to user space using Address 0x00000000. This mode is intended for code development only.

Normal Application Mode

If after a reset event, the kernel finds the GPIO5 pin high, the kernel checks for a valid first page of user flash. A valid page is one that has a valid CRC at Address 0x7FC. If the kernel determines that the first page is valid, it branches to the beginning of that page. For robustness, it is recommended that the first page contain code that validates the rest of the application code before exiting the first page.

Instead of a CRC at Address 0x7FC, a key value of 0x16400000 can also be used. However, this does not ensure that the first page is not corrupted.

Note that if the 0x7FC location has already been programmed and user code is required to modify it, either Page 0 must be erased or the 0x7FC location be rewritten with all zeros. Overwriting the 0x7FC location with any other value is not recommended, because an invalid ECC can result.

Boot Loading Mode

If after a reset, the kernel finds the GPIO5 pin to be high and Page 0 not to have a valid CRC or key, the kernel checks for a valid user boot loader. The boot loader structure is as follows. The boot loader can be of any size up to 30 kB but must be located at the top of the user flash.

- Address (Flash Size – 0x4) must contain the CRC of the boot loader.
- Address (Flash Size – 0x1C) must contain the lowest address of the boot loader block.
- Address (Flash Size – 0x20) must contain the entry point of the boot loader code.
- All three addresses must contain valid information to enter the boot loader correctly.

An extra consideration for the entry point is that, due to the Cortex-M3 addressing architecture, any address branched to must be a half word boundary + 1.

For example:

- If the lowest address of the boot loader on a 96 kb device (ADuCM330) is 0x10800, the entry point must be 0x10801.
- If the lowest address of the boot loader on a 128 kb device (ADuCM331) is 0x18800, the entry point must be 0x18801.

The flash size is used to designate the size of user flash. For the ADuCM330 (96 kB), the size of user flash is 0x18000. For the ADuCM331 (128 kB), the size of user flash is 0x20000.

The kernel uses the values at these addresses, shown in the previous bullet list, to determine if the boot loader is valid.

If the boot loader feature is not used, a value of 0xFFFF FFFF must be placed at (Flash Size – 0x20).

The normal application can receive commands via the LIN to change to the boot loader that is located at the top of the user flash. The boot loader can use an appropriate protocol to update the application code. As the first step of reflashing, the boot loader must ensure that the value at Address 0x007FC is not the Page 0 CRC or the 0x16400000 key. The last step of reflashing must rewrite Address 0x007FC to ensure that recovery after partial reflashing is possible.

Note that if the boot loader locations have already been programmed and user code is required to modify them, the user must either erase that page or overwrite the location with all zeros. Overwriting the boot loader locations with any other value is not recommended because an invalid ECC can result.

Interrupted Boot Loading

If reflashing the application is, for any reason, interrupted before it is complete, reflashing can be restarted as follows.

Upon a reset after a partial reflash, the kernel detects that the application is corrupted (value at 0x007FC is incorrect). The kernel checks the boot loader, and if the boot loader is found to be valid, the kernel passes control to the boot loader, and the reflashing can be repeated by the boot loader until complete.

Downloader

If the kernel finds that the boot loader is also not valid, the kernel enters the downloader of the kernel, and it is possible to reflash the user code as described in the Application Note AN-946, *Flash/EE Memory Programming via LIN—Protocol 6*. When in this state, if no valid LIN frames are received for nominally one hour, the device goes into power down.

Additionally, download mode offers a fast LIN download option so that the user can program the device more quickly, at speeds of up to 100 kbaud, if capable programming hardware is available. Note that download mode cannot be entered from any of the other modes except via a reset. Also note that this mode cannot be exited except via a reset.

Interrupted Downloader

If downloading to the user space is, for any reason, interrupted before it is complete, downloading can be restarted as follows. As long as 0x007FC does not contain the Page 0 CRC or key and the boot loader is not valid, the downloader can be restarted at any time by resetting the device (for example, POR). For this reason, the downloader must only update 0x007FC after the full application has been downloaded and verified. In addition, if the boot loader feature is used, the values at (Flash Size – 0x4) and (Flash Size – 0x1C) must only be updated after the entire boot loader has been downloaded and verified.

Boot Updater

The boot loader can be updated as follows. Using the method described previously, code can be downloaded as application code with the following capabilities:

- The code must look like a valid application to the system.
- The code must consist mainly of a copy of the new boot loader.
- The code must contain code that autonomously erases the old boot loader and replaces it with the new boot loader.

When this process is complete, the system can use the new boot loader to reflash the application space with a battery monitoring application.

Interrupted Boot Updater

If reflashing the boot loader is, for any reason, interrupted before it is complete, reflashing can be repeated by resetting the device (for example, POR) because the boot updater application is still valid.

Direct LIN Interface (Ext Mode)

Single battery monitoring can be achieved using the LIN pin for communications. If the kernel detects that a download is required, it drives the GPIO3 pin high, which can be used to enable an external LIN interface. The kernel then monitors both the LIN pin and the Rx pin. If a frame start is detected on the LIN pin, the kernel assumes that the application is in single battery monitoring mode. If a frame start is first detected on the LIN_RX pin (GPIO4), the kernel switches over to use the LIN_RX/LIN_TX pins (GPIO4/GPIO1) instead.

This allows use of an external LIN transceiver. For the kernel, the only difference between these modes is this switching from the LIN pin to the Rx/Tx pins. The kernel only performs this switching when it reaches download mode. If the kernel exits to user or boot loader mode, the user code must switch to the Rx/Tx pins and drive the GPIO3 pin to control the transceiver.

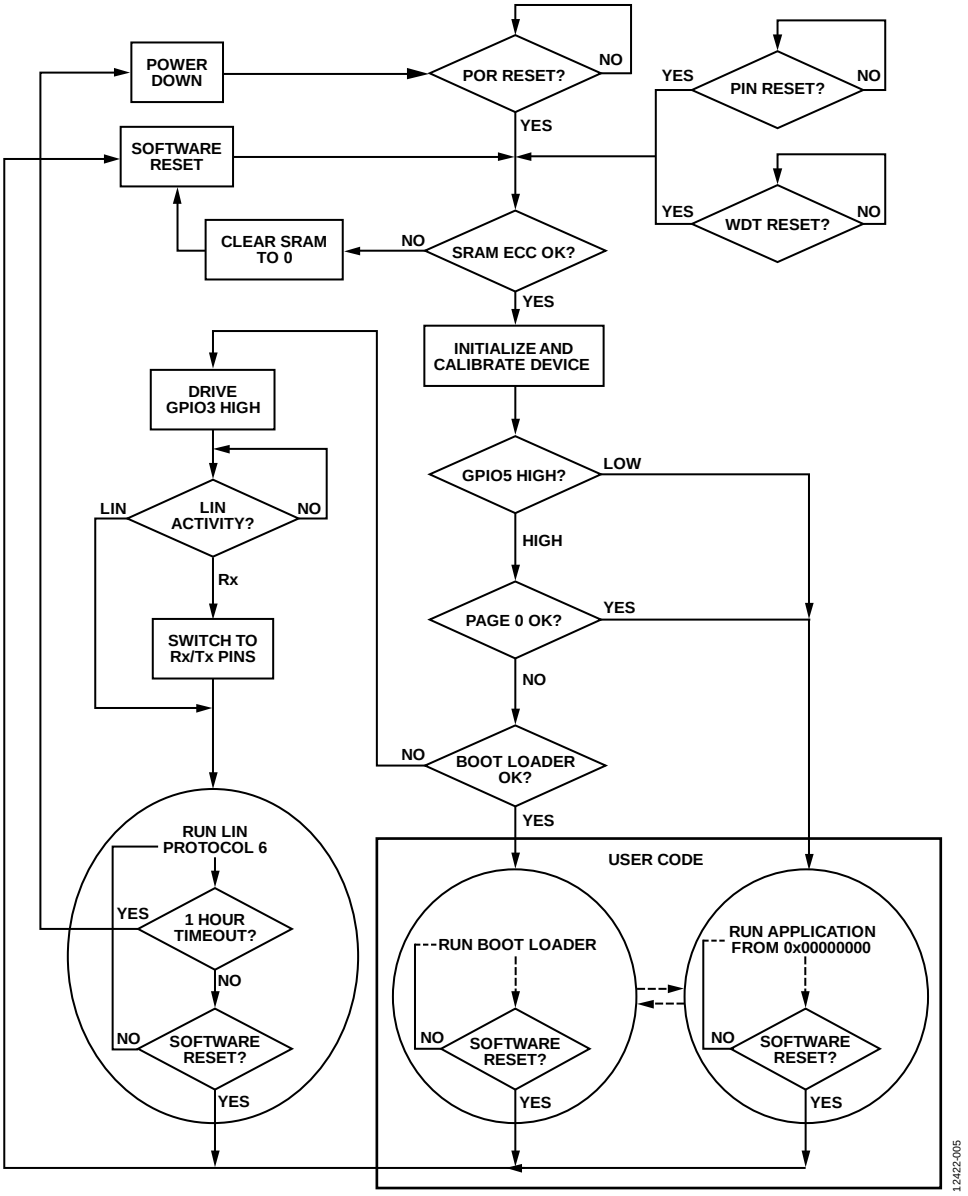

Figure 4. ADuCM330/ADuCM331 Kernel Flowchart

RESET

RESET FEATURES

There are four kinds of resets:

- External reset (EXTRST)
- Power-on reset (POR)
- Watchdog timeout (WDRST)
- Software system reset (SWRST)

RESET OPERATION

The software system reset is provided as part of the Cortex-M3 core. To generate a software system reset, the value of 0x05FA0004 must be written to the application interrupt/reset control register (AIRCR register). This register is part of the NVIC register and is located at Address 0xE000ED0C. The RSTSTA register stores the cause for the reset until it is cleared by writing to the RSTCLR register. RSTSTA and RSTCLR can be used during a reset exception service routine to identify the source of the reset.

The external reset pin does not reset debug logic.

Table 40. Device Reset Implications

	Impact						
Reset	**Reset External Pins to Default State**	**Execute Kernel**	**Reset All MMRs Except RSTSTA**	**Reset All Top Die Registers**	**Reset All Peripherals**	**Valid SRAM**	**RSTSTA After Reset Event**
SWRST	Yes	Yes	Yes	No	Yes	Yes/No[1]	RSTSTA[3] = 1
WDRST	Yes	Yes	Yes	No	Yes	Yes/No[1]	RSTSTA[2] = 1
EXTRST	Yes	Yes	Yes	No	Yes	Yes/No[1]	RSTSTA[1] = 1
POR	Yes	Yes	Yes	Yes	Yes	Yes/No[2]	RSTSTA[0] = 1

[1] RAM is not valid when an ECC error is detected during kernel initialization (SRAM initialized to zero).
[2] RAM is not valid where the low voltage flag is set.

RESET MEMORY MAPPED REGISTERS

Table 41. Reset Memory Mapped Register (Base Address 0x40002400)

Offset	Name	Description	Access	Default
0x0040	RSTSTA	Reset status register	R	Depends on the type of reset
0x0040	RSTCLR	Reset clear register	W	Not applicable

Reset Status/Reset Clear Registers: RSTSTA/RSTCLR (Address 0x40002440)

Table 42. RSTSTA/RSTCLR Register Bit Descriptions

Bits	Name	Description
[7:4]	RESERVED	Reserved
3	SWRST	Software reset 0: cleared by setting the corresponding bit in the RSTCLR register 1: set automatically when the Cortex-M3 system reset is generated
2	WDRST	Watchdog timeout 0: cleared by setting the corresponding bit in the RSTCLR register 1: set automatically when a watchdog timeout occurs
1	EXTRST	External reset 0: cleared by setting the corresponding bit in the RSTCLR register 1: set automatically when an external reset occurs
0	POR	Power-on reset 0: cleared by setting the corresponding bit in the RSTCLR register 1: set automatically when a power-on reset occurs

MEMORY ORGANIZATION

Four separate blocks of memory are accessible to the user:

- 6 kB of SRAM from 0x2000 0000 to 0x2000 17FF
- 4 kB of data flash memory 0x0040 0000 to 0x0040 0FFF
- 96 kB of on-chip Flash/EE memory available to the user from 0 to 0x0001 7FFF (ADuCM330)
- 128 kB of on-chip Flash/EE memory available to the user from 0 to 0x0001 FFFF (ADuCM331)
- An additional 2 kB reserved for the kernel space from 0x0002 0000 to 0x0002 07FF.

These blocks are mapped according to the Cortex-M3 memory map, as shown in Figure 5. All on-chip peripherals are accessed via the memory mapped registers (MMRs), situated in the bit band region. Any access to MMRs takes three clock cycles of the clock used in the related functional block, unless otherwise noted.

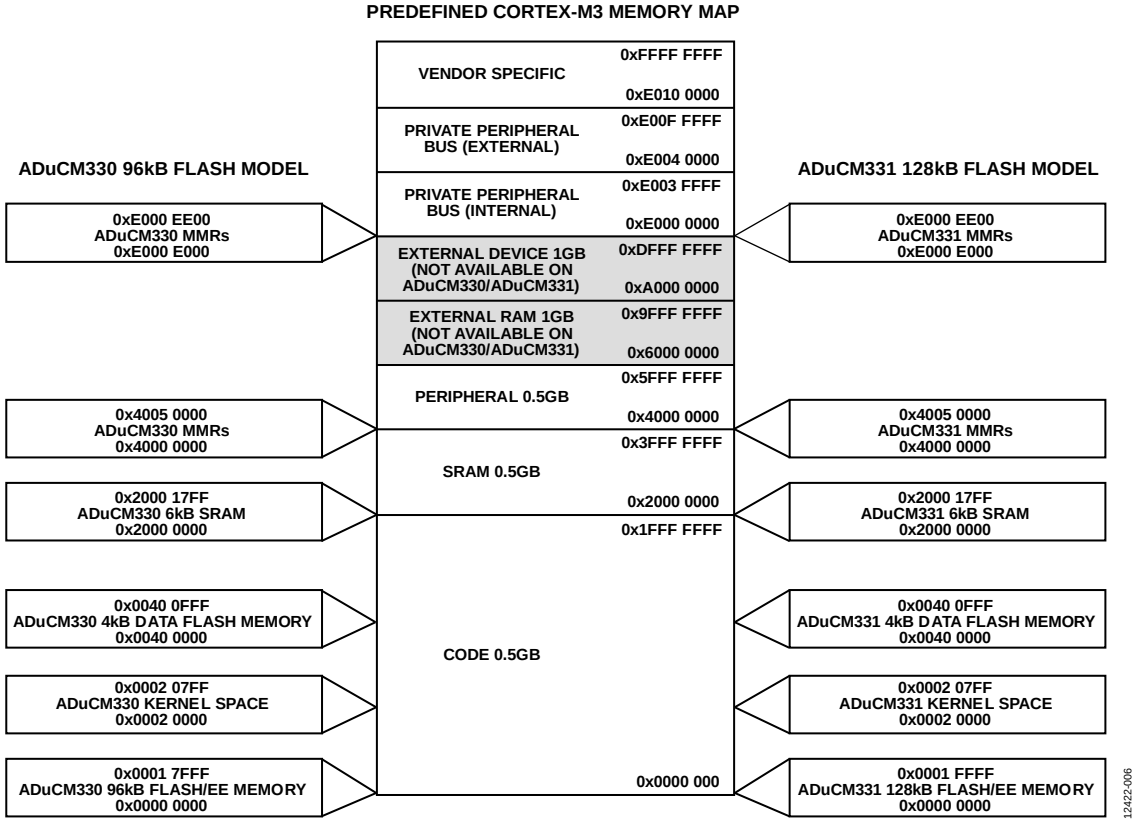

Figure 5. ADuCM330/ADuCM331 Memory Map Diagram

FLASH CONTROLLER

FLASH CONTROLLER FEATURES

- 96 kB (ADuCM330) or 128 kB (ADuCM331) program flash
- 4 kB data flash

Commands Supported

- Write
- Mass erase and page erase
- Generation of signatures for single or multiple pages
- Command abort

Note that accesses from the core on program flash are not stalled if the command in progress is in data flash. Accesses in data flash are stalled if the command in progress is in program flash.

Flash Protection

- Write protection for user space and data flash
- Ability to lock serial wire interface

Flash Integrity

- Automatic signature check of kernel space on reset
- User signature for application code
- 8-bit ECC
- 1-bit ECC errors can generate a flash ECC interrupt
- 2-bit or greater ECC errors generate a hard fault exception

FLASH CONTROLLER OVERVIEW

The flash controller supports two embedded high data retention (HDR) flash memories: 96 kB (ADuCM330) or 128 kB (ADuCM331) program flash and 4 kB data flash. Program flash memory is for storing user code and has an additional 2 kB of information space to store the kernel. Data flash memory can store additional data by the user. A write to the flash is executed via keyhole access.

FLASH MEMORY OPERATION AND ORGANIZATION

On the ADuCM330, the controller supports 96 kB of program flash finishing at Address 0x17FFF with 2 kB of information space containing the kernel, as shown in Figure 6.

On the ADuCM331, the controller supports 128 kB of program flash finishing at Address 0x1FFFF with 2 kB of information space containing the kernel, as shown in Figure 7.

The ADuCM330/ADuCM331 additionally contain a separate block with 4 kB of data flash memory, as shown in Figure 8. Page sizes are 2 kB for program flash and 512 bytes for data flash.

Program Flash Information Space

The program flash information space is mapped above the program flash user space. The information space contains the kernel.

Program Flash User Space

The top 24 bytes of program flash user space are reserved for a signature, the user write protection, and the user failure analysis key (USERFAKEY), as shown in Figure 9.

If the user tries to read from or write to a portion of memory that is not available, a bus error is returned. If the user tries to write via the keyhole to a portion of memory that is not available, the appropriate error flag is set.

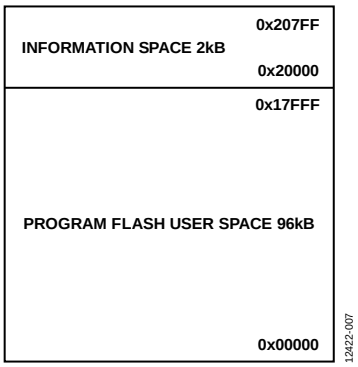

Figure 6. ADuCM330 Program Flash Memory Map

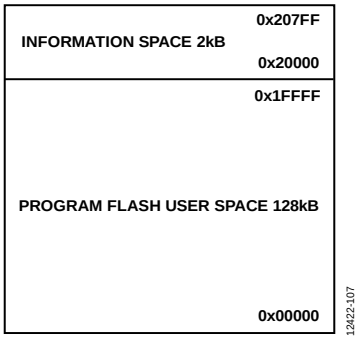

Figure 7. ADuCM331 Program Flash Memory Map

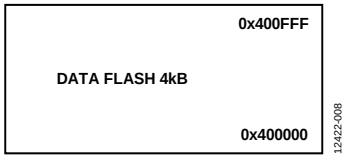

Figure 8. Data Flash Memory Map

63 40	39	31 0	96kB	128kB
SIGNATURE		RESERVED	0x17FF8	0x1FFF8
SWD LOCK KEY	DATA PROTECTION	PROGRAM FLASH WRITE PROTECTION	0x17FF0	0x1FFF0
USER FA KEY			0x17FE8	0x1FFE8
BOOT LOADER LOWEST ADDRESS		BOOT LOADER ENTRY POINT	0x17FE0	0x1FFE0
REST OF THE UPPERMOST PAGE IN USER SPACE			0x17800	0x1F800

12422-009

Figure 9. ADuCM330/ADuCM331 Uppermost Page of User Memory

WRITING TO FLASH/EE MEMORY

Writing to program and data flash is achieved through keyhole access. Each write programs 64 bits of data.

Keyhole Access

Keyhole access consists of the following:

- Flash address
- Data registers
- Command register

To write to a flash location, the following sequence is required.

- Write to the FEEADDR register with the 23-bit memory mapped address of the flash location.
- Write to the FEEDATL and FEEDATH registers with 64 bits of data.
- Write to the FEECMD register with write command.

After the write command is given, the controller writes to the program or the data flash, based on the address provided. Note that a single 64-bit location can only be written to once without an erase.

Example Code: How to Write to Flash Memory

```
unsigned int uiSTA;
FEEADR = 0x00001800;                          // A 64bit flash location
FEEDATL = 0x01234567;                         // Data
FEEDATH = 0x89ABCDEF;
FEECMD = 0x4;                                 // Flash write command
do{
 do{
 uiSTA = FEESTA;
 }while(uiSTA & 0x1);                         // Wait until not busy
}while( ! (uiSTA & 0x4));                     // Ensure command completed
```

ERASING FLASH/EE MEMORY

User code can call two flash erase commands.

- Mass erase: this command erases the entire user flash memory. After entering the user protection key into the FEEKEY register, write the mass erase command to the FEECMD register.
- Page erase: this command erases 2 kB in user space of program flash or 512 bytes of data flash. The page is selected by the FEEADR1L register. After entering the user protection keys into the FEEKEY register, load the FEEADR1L register with the page address to be erased. Finally, write the page erase command to the FEECMD register. CMDDONE (FEESTA[2]) indicates that the page erase command is complete.

During a page or mass erase sequence, the flash controller and flash block consume extra current for the duration of the flash erase sequence. See the ADuCM330/ADuCM331 data sheet for exact specifications.

Example Code: How to Mass Erase Data Flash Memory

```
unsigned int uiSTA;
FEEKEY = 0xF456;                              // Enter Keys
FEEKEY = 0xF123;
FEECMD = 0x7;                                 // Mass Erase Data Flash
do{
 do{
  uiSTA = FEESTA;
 }while(uiSTA & 0x1);                         // Wait until not busy
}while( ! (uiSTA & 0x4));                     // Ensure command completed
```

Example Code: How to Erase Flash Page 32

```
unsigned int uiSTA;
FEEKEY = 0xF456;                                    // Enter Keys
FEEKEY = 0xF123;
FEEADR1L = 0x00010000;                              // A location in the page
FEECMD = 0x1;                                       // Flash erase page command
do{
 do{
  uiSTA = FEESTA;
 }while(uiSTA & 0x1);                               // Wait until not busy
}while( ! (uiSTA & 0x4));                           // Ensure command completed
```

CONTROLLER OPERATION

The ADuCM330/ADuCM331 flash controller supports simultaneous access to both data and program flash during certain operations. Table 43 shows these operations.

Table 43. Flash Controller Access Matrix

Data Flash	Program Flash	Available
Standby	Standby	Yes
Read	Standby	Yes
Program	Standby	Yes
Erase	Standby	Yes
Standby	Read	Yes
Standby	Program	Yes
Standby	Erase	Yes
Read	Read	Yes
Program	Read	Yes
Erase	Read	Yes
Read	Program	No
Read	Erase	No
Program/Erase	Program	No
Erase/Program	Erase	No

FLASH PROTECTION

Three types of protection are implemented:

- Key protection
- Read protection
- Write protection

Flash Protection: Key Protection

Some of the flash controller registers are key protected to avoid accidental writes to these registers.

The user key is 0xF123F456. This key must be entered to run certain user commands, to write to certain locations in flash, or to enable write access to the setup (FEECON1) register. When entered, the key remains asserted unless a command is written to the FEECMD register. When the command starts, the key clears automatically. If this key is entered to enable write access to the setup register or to enable writes to certain locations in flash, it must be cleared by user code afterwards. To clear the key, write any value to the key register.

Flash Protection: User Read Protection

User space read protection is provided by disabling serial wire access. The user can disable serial wire access by writing 0 to the DBG bit in the flash FEECON1 register. Serial wire access is disabled while the kernel is running. Otherwise, serial wire access can prevent the kernel from running to completion. When the kernel has completed, it reenables serial wire access for the user (unless the SWD restriction key is valid; see the Kernel section for more information).

Flash Protection: User Write Protection

User write protection is provided to prevent accidental writes to pages in user space and to protect blocks of user code when downloading extra code to flash. If a write or erase of a protected location is detected, the ADuCM330/ADuCM331 flash controller generates an interrupt when the command error or complete interrupt are enabled.

The write protection is stored at the top of the user space. The top four bytes are for a signature; the next four are reserved. The next 64-bit flash location contains the following.

96 kB Program Flash

Table 44. Program Flash Write Protection [31:0]

Bit 31	Bit 30	Bit 29	Bit 28	Bit 27	Bit 26	Bit 25	Bit 24	Bit 23	Bit 22	Bit 21	Bit 20	Bit 19	Bit 18	Bit 17	Bit 16
Reserved								Page 46 and Page 47	Page 44 and Page 45	Page 42 and Page 43	Page 40 Page and 41	Page 38 and Page 39	Page 36 and Page 37	Page 34 and Page 35	Page 32 and Page 33
Bit 15	**Bit 14**	**Bit 13**	**Bit 12**	**Bit 11**	**Bit 10**	**Bit 9**	**Bit 8**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Page 30 and Page 31	Page 28 and Page 29	Page 26 and Page 27	Page 24 and Page 25	Page 22 and Page 23	Page 20 and Page 21	Page 18 and Page 19	Page 16 and Page 17	Page 14 and Page 15	Page 12 and Page 13	Page 10 and Page 11	Page 8 and Page 9	Page 6 and Page 7	Page 4 and Page 5	Page 2 and Page 3	Page 0 and Page 1

128 kB Program Flash

Table 45. Program Flash Write Protection [31:0]

Bit 31	Bit 30	Bit 29	Bit 28	Bit 27	Bit 26	Bit 25	Bit 24	Bit 23	Bit 22	Bit 21	Bit 20	Bit 19	Bit 18	Bit 17	Bit 16
Page 62 and Page 63	Page 60 and Page 61	Page 58 and Page 59	Page 56 and Page 57	Page 54 and Page 55	Page 52 and Page 53	Page 50 and Page 51	Page 48 and Page 49	Page 46 and Page 47	Page 44 and Page 45	Page 42 and Page 43	Page 40 Page and 41	Page 38 and Page 39	Page 36 and Page 37	Page 34 and Page 35	Page 32 and Page 33
Bit 15	**Bit 14**	**Bit 13**	**Bit 12**	**Bit 11**	**Bit 10**	**Bit 9**	**Bit 8**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Page 30 and Page 31	Page 28 and Page 29	Page 26 and Page 27	Page 24 and Page 25	Page 22 and Page 23	Page 20 and Page 21	Page 18 and Page 19	Page 16 and Page 17	Page 14 and Page 15	Page 12 and Page 13	Page 10 and Page 11	Page 8 and Page 9	Page 6 and Page 7	Page 4 and Page 5	Page 2 and Page 3	Page 0 and Page 1

4 kB Data Flash

Table 46. SWD Lock Key [63:40], Data Flash Write Protection [39:32]

Bits[63:40]	Bit 39	Bit 38	Bit 37	Bit 36	Bit 35	Bit 34	Bit 33	Bit 32
SWD Lock Key	Page 7	Page 6	Page 5	Page 4	Page 3	Page 2	Page 1	Page 0

The write protection is uploaded by the flash controller into local registers after a reset. To write to the write protection bits, the user must first write 0xF456 followed by 0xF123 to the key register. After the write protection has been written, it cannot be rewritten without a mass erase of user space or a page erase of the last page (if the last page is not protected). After a mass erase, the device must be reset to deassert the uploaded copy of the write protection bits.

The following is the sequence to program the Write Protection Bits[31:0]:

- Ensure that the last page of user space is erased.
- Write 0xF456 followed by 0xF123 to the key register, FEEKEY.
- Write the required write protection directly to flash. Write 0 to enable protection. The write protection address is 0x17FF0 for 96 kB of flash (ADuCM330). The write protection address is 0x1FFF0 for 128 kB of flash (ADuCM331).
- Verify that the write completed successfully by polling the status register FEESTA[3] or by enabling a write complete interrupt.
- Reset the device, and the write protection is uploaded from the user space and activated by the flash controller.

If the write protection in flash has not been programmed (that is, 0xFFFF FFFF is uploaded from flash on power-up), the FEEPROP/FEEPROD register can be written to directly from user code. This allows the user to verify the write protection before committing it to flash. If the write protection in flash has been programmed, the MSB of the write protection must be programmed to 0 to prevent erasing of the write protection block. For the ADuCM330 write protection, the memory is split into 24 blocks, that is, for 96 kB of flash, memory is split into 24 × 4 kB blocks. For the ADuCM331 write protection, the memory is split into 32 blocks, that is, for 128 kB of flash, memory is split into 32 × 4 kB blocks.

If an attempt is made to write to the write protection word in flash without setting the FEEKEY first, a flag is set.

FLASH CONTROLLER FAILURE ANALYSIS KEY

It may be necessary to perform failure analysis on devices that are returned by the user even though read protection is enabled. A method has been provided to allow failure analysis of protected memory by a user failure analysis (FA) key.

The FA key is a 64-bit key that is stored at the top of the user space in flash, as shown in Figure 9. This key is used to gain access to user code if the serial wire interface has been locked. It is the responsibility of the user to program this key to a value. A value of 0xFFFF FFFF is the default, and any other value programmed in this location is treated as the user FA key. This same key must be programmed into the USERFAKEY registers to unlock read protection via SWD. The key must be given to Analog Devices to enable access to user code.

FLASH INTEGRITY SIGNATURE FEATURE

The signature is used to check the integrity of the flash device. The signature is calculated on 64-bit data by splitting it into two 32-bit data-words. The CRC calculation sequence is first the lower 32 bits, then the higher 32 bits of the 64-bit flash data.

The ECC is checked on each 72-bit flash read. If errors are corrected by the ECC, the ERRCORRECTED flag in the status register is set after the signature check is completed. If errors are detected and cannot be corrected by ECC, the ERRDETECTED flag in the status register is set in the status register. A signature check is treated as a failure when the computed signature is not equal to the stored signature.

The software can call a signature check command occasionally or whenever a new block of code is about to be executed. The signature is a 24-bit CRC with the polynomial $x^{24} + x^{23} + x^6 + x^5 + x + 1$. Contact Analog Devices for more information, if required.

The sign command can be used to generate or check the signature of a block of code, where a block can be a single or multiple pages. A 24-bit linear feedback shift register (LFSR) is used to generate the signature. The hardware assumes that the signature for a block is stored in the upper four bytes of the most significant page of a block; therefore, these four bytes cannot be included when generating the signature.

The following procedure must be followed to generate a signature.

1. Write the start address of the block to the FEEADR1L register.
2. Write the end address of the block to the FEEADR1H register.
3. Write the sign command to the command register.
4. When the command is complete, the signature is available in the sign register, FEESIGN. The signature is compared with the data stored in the upper four bytes of the uppermost page of the block. If the data does not match the signature, a fail status of VERIFYERR is returned in the status register (FEESTA[5:4] = 10).

While the signature is being computed, all other accesses to flash are stalled.

FEEADR1L and FEEADR1H addresses are byte addresses; however, only pages need to be identified, that is, the lower nine bits are ignored by the hardware.

Note that the user must run the CRC polynomial in user code first to generate the CRC value and must write this to the upper four bytes of the uppermost page of a block. When this operation is complete, any call of the signature feature compares this 4-byte value to the result of the signature check function.

ECC Error Handling

During a read of the flash, if there is a 1-bit error, the error is corrected, and the appropriate flags are set in the status register. A 1-bit ECC interrupt in the command control register (FEECON0) can also be enabled, if required.

If there is a 2-bit ECC error, an error is issued by the controller. ECC errors that are 2-bit or greater generate an exception, unless they are encountered by the sign command.

An ECC error is signaled by the ECC error detection/correction module when a flash location is read. Depending on when the read happens (during command execution, during a read) and from which flash (program/data) the read happens, the appropriate flags are set in the status register (ECCERRCMD, ECCERRREAD, and so on).

Note that 1-bit errors corrected meet full data sheet specification.

ECC Error During Read

Because a program and data flash read can happen simultaneously, two separate ECCERRREAD flags are present in the status register: FEESTA[10:9] and FEESTA[12:11]. If the interrupt is configured to be generated when an ECC error occurs, the address at which the error is detected is available as readback for the user.

ECC Error During Execution of Sign Command

If there is an ECC error during signature check, ECC error data and address registers are not updated. After the command is complete, ECCERRCMD flags in FEESTA[8:7] are updated. If ECCCMDINTEN in the FEECON0 register is set, an interrupt is generated if any of the ECCERRCMD bits are set.

FLASH CONTROLLER PERFORMANCE AND COMMAND DURATION

- Direct single write access (72-bit location): 72.187 µs
- Mass erase: 17.126 ms
- Page erase: 17.012 ms
- Page write: program flash user space (256, 72-bit locations) = 8 × 72.187 µs + 248 × 49.7 4 µs = 12.913 ms

For a 4 kB data flash, each row has 32 × 72 bit locations; however, there are only 2 rows per page. Assuming that the page is written in sequence, and that the writes are done back to back,

- Page write: 4 kB data flash = 2 × ((1 × 72.187 µs) + (31 × 49.74 µs)) = 3.228 ms

FLASH CONTROLLER MEMORY MAPPED REGISTERS

Table 47. Flash Controller Memory Mapped Registers (Base Address 0x40018000)

Offset	Name	Description	Access	Default
0x0000	FEESTA	Status register	R	0x0000 0000
0x0004	FEECON0	Command control register	RW	0x0010
0x0008	FEECMD	Command register	RW	0x0000
0x000C	FEEADR	Flash address keyhole register	RW	0x0000 0000
0x0010	FEEDATL	Flash data register (lower)	RW	0x0000 0000
0x0014	FEEDATH	Flash data register (upper)	RW	0x0000 0000
0x0018	FEEADR1L	Lower page address register	RW	0x0000 0000
0x001C	FEEADR1H	Upper page address register	RW	0x0000 0000
0x0020	FEEKEY	Key register	W	0x0000
0x0028	FEEPROP	Program flash write protection register	RW	0xFFFF FFFF
0x002C	FEEPROD	Data flash write protection register	RW	0xFF
0x0030	FEEECC	Data flash ECC disable register	RW	0x0000 0000
0x0034	FEESIGN	Signature register	R	0x0000 0000
0x0038	FEECON1	Serial wire control register	RW	0x0001
0X0040	FEEABORT	Write abort address register	R	0x0000 0000
0x0048	FEEAEN0	Abort enable register	RW	0x0000
0x0068	USERFAKEY0	USERFAKEY low register [31:0]	RW	0x0000 0000
0x006C	USERFAKEY1	USERFAKEY high register [63:32]	RW	0x0000 0000
0x0074	FEEPECC	Program flash address for ECC error	R	0x0000 0000
0x0078	FEEDECC	Data flash address for ECC error	R	0x0000 0000

Flash Memory Status Register

Address: 0x40018000, Reset: 0x0000 0000, Name: FEESTA

Table 48. FEESTA Register Bit Descriptions

Bits	Name	Description		
[31:25]	RESERVED	Reserved. These bits return 0 when read.		
[24:22]	ECCCOUNTDATA	This is a 3-bit counter that reflects the number of 1-bit ECC read errors in data flash after FEESTA[12:11] = 0x2 and before FEESTA is read. This counter does not count on ECC 2-bit errors. The counter is cleared when FEESTA is read by the user.		
[21:20]	RESERVED	Reserved.		
[19:17]	ECCCOUNTPROG	This is a 3-bit counter that reflects the number of 1-bit ECC read errors in program flash after FEESTA[10:9] = 0x2 and before FEESTA is read. This counter does not count on ECC 2 bit errors. The counter is cleared when FEESTA is read by the user.		
[16:15]	ECCERRSIGN	ECC error during initial signature check.		
		Bits	**Name**	**Description**
		00	NOERR	No error. Successful flash read operation during initial signature check or page signature check.
		01	ERRDETECTED	During initial signature check, 2-bit errors are detected, and not corrected for at least one flash location
		10	ERRCORRECTED	1-bit error is corrected for one flash location during a signature command.
		11	ERR1BIT_2Bit	During the initial signature command, 1-bit errors and 2-bit errors are detected on one or more flash locations.
14	INIT	Initialization upload in progress. After a reset, the flash controller uploads FLASHCONFIG, checks the information space signature, and uploads the user write protection. 0: cleared to 0 when the upload has completed. User code cannot run until this bit deasserts. 1: set to 1 while upload is in progress.		
13	SIGNERR	Information space signature check on reset error. After a reset, the flash controller automatically checks the information space signature. User code does not execute if this bit is set. 0: cleared to 0 if the signature check returns no errors. 1: set to 1 if the signature check fails.		

Bits	Name	Description		
[12:11]	ECCERRREADDT	ECC errors during a read of data flash		
		Bits	**Name**	**Description**
		00	NOERR	No error. Successful read from data flash.
		01	ERRDETECTED	2-bit error detected in one or more flash locations during a read from data flash. The errors are not corrected.
		10	ERRCORRECTED	1-bit error detected for one flash location during a read from data flash. The error is corrected.
		11	ERR1BIT_2Bit	During the initial signature command, 1-bit errors and 2-bit errors are detected on one or more flash locations.
[10:9]	ECCERRREADPG	ECC errors during read of program flash.		
		Bits	**Name**	**Description**
		00	NOERR	No error. Successful read from program flash.
		01	ERRDETECTED	2-bit error detected in one or more flash locations during a read from program flash. The errors are not corrected.
		10	ERRCORRECTED	1-bit error detected for one flash location while during read from program flash. The error is corrected.
		11	ERR1BIT_2Bit	During the read, 1-bit error and 2-bit errors are detected in program flash.
[8:7]	ECCERRCMD	ECC errors.		
		Bits	**Name**	**Description**
		00	NOERR	No error. Successful flash read operation during the signature check.
		01	ERRDETECTED	2-bit error detected in one or more flash locations during the signature command. The errors are not corrected.
		10	ERRCORRECTED	1-bit error detected for one flash location while doing a signature check. The error is corrected.
		11	ERR1BIT_2Bit	During the signature command, 1-bit error and 2-bit errors are detected on one or more flash locations.
6	RESERVED	Reserved.		
[5:4]	CMDFAIL	Status of a command on completion.		
		Bits	**Name**	**Description**
		00	SUCCESSCOMP	Successful completion of a command or a write.
		01	LOCATIONPROT	Attempted write or erase of a protected location. The command is ignored.
		10	VERIFYERR	Read verify error. After an erase, the controller reads the corresponding word(s) to verify that the transaction completed successfully. If data read is not all Fs, this is the resulting status. If the sign command is executed, and the resulting signature does not match the data in the upper four bytes of the upper page in a block, this is the resulting status.
		11	Aborted	Indicates that a command or a write was aborted by an abort command, or that a system interrupt caused an abort.
3	WRALCOMP	Write almost complete. 0: cleared when read. 1: set to 1 after the second 24-bit write of the three 24-bit writes is complete.		
2	CMDDONE	Command complete. 0: cleared when read. 1: set to 1 when a command completes. If there are multiple commands, this status bit asserts after the first command completes and stays asserted until read. It is recommended to wait until this bit is set before continuing.		
1	WRCLOSE	Write close. 0: cleared after the WRALCOMP bit is set to 1. 1: set to 1 when the user writes all keyhole register for a flash write, and the controller starts writing. If this bit is set to 1, all keyhole registers except the command register are closed for writing.		
0	CMDBUSY	Command busy. 0: cleared to 0 when the flash block is not executing any commands entered via the command register. 1: set to 1 when the flash block is executing a command entered via the command register.		

Flash Memory Command Control Register

Address: 0x40018004, Reset: 0x0010, Name: FEECON0

Table 49. FEECON0 Register Bit Descriptions

Bits	Name	Description
[15:5]	RESERVED	Reserved. These bits return 0 when read.
4	ECCCMDINTEN	Interrupt enable when a 1-bit ECC error happens during a read from program or data flash. This is enabled by default.
3	RESERVED	Reserved.
2	CMDERRINTEN	Command fail interrupt enable.
1	WRALCOMP	Write almost complete interrupt enable.
0	CMDCOMPINTEN	Command complete interrupt enable.

Flash Memory Command Register

Address: 0x40018008, Reset: 0x0000, Name: FEECMD

Table 50. FEECMD Register Bit Descriptions

Bits	Name	Description
[15:4]	RESERVED	Reserved. These bits return 0 when read.
[3:0]	CMD	Flash controller commands (see Table 51).

The commands shown in Table 51 are supported by the flash block. For repeated page erase commands, the key must be entered before each command. If a command is entered without entering the key first, no action is taken, and CMDDONE does not assert. Accesses from the core on program flash are not stalled if the command in progress is in data flash. Accesses in data flash are stalled if the command in progress is in program flash. CMDDONE is asserted if the key is not entered or if an incorrect key is entered.

Table 51. Flash Controller Commands (FEECMD[3:0])

CMD	Name	Description
0000	IDLE	No command executed.
0001	ERASEPAGE	Write the address of the page to be erased to the FEEADR1L register, then write this code to the FEECMD register, and the flash erases the page. When the erase is complete, the flash reads every location in the page to verify that all the words in the page are erased. If there is a read verify error, it is indicated in the FEESTA register. ECC is disabled for this command. To erase multiple pages, wait until a previous page erase has completed, check the status, and then issue a command to start the next page erase. Before entering this command, 0xF456 followed by 0xF123 must be written to the key register.
0010	SIGN	Use this command to generate a signature for a block of data. The signature is generated on a page by page basis. To generate a signature, enter the address of the first page of the block in FEEADR1L, write the address of the last page to FEEADR1H, then write this code to the FEECMD register. When the command is complete, the signature is available for reading in the FEESIGN register. The last four bytes of the last page in a block is reserved for storing the signature. Before entering this command, 0xF456 followed by 0xF123 must be written to the key register. ECC is checked with this command (even if it is specifically disabled for a page). ECC errors during the sign command update FEESTA[8:7] bits. ECC errors during the sign command do not update FEEPECC or FEEDECC registers. ECC errors during the sign command only generate an interrupt if FEECON0[2] is set to 1.
0100	WRITE	Write to flash locations. This command takes the address from the FEEADR register and data FEEDAT keyhole registers. Before entering this command, 0xF456 followed by 0xF123 must be written to the key register for writing into the write protection location and the user FA key location. No key is required for other flash locations.
0101	MASSERASEPROG	Erase all of user space in program flash. When the mass erase is complete, the controller reads every location to verify that all locations are 0xFFFF FFFF. If there is a read verify error, it is indicated in the FEESTA register. ECC is disabled for this command. To enable this operation, 0xF456 followed by 0xF123 must be written to the FEEKEY register (this is to prevent accidental erases).
0110	MASSERASEDATA	Erase the data flash (only first 2048 bytes). When the mass erase is complete, the controller reads every location of data flash to verify that all locations are 0xFF FFFF FFFF FFFF FFFF. If there is a read verify error, it is indicated in the status register. ECC is disabled for this command. Before entering this command, 0xF456 followed by 0xF123 must be written to the key register.

CMD	Name	Description
0111	MASSERASEALLDATA	Erase all user space and all information space in data flash (4096 bytes). When the mass erase is complete, the controller reads every location of data flash to verify that all locations are 0xFF FFFF FFFF FFFF FFFF. If there is a read verify error this is indicated in the status register. ECC is disabled for this command. Before entering this command, 0xF456 followed by 0xF123 must be written to the key register.
1000	ABORT	If this command is issued, any command currently in progress is stopped. The status indicates command completed with an error status in FEESTA[6:4]. Note that this is the only command that can be issued while another command is already in progress. This command can also be used to stop a write that is in progress. If a write or erase is aborted, the flash timing is violated, and it is not possible to determine if the write or erase completed successfully. To enable this operation, 0xF456 followed by 0xF123 must first be written to the FEEKEY register (this is to prevent accidental aborts).

Flash Address Keyhole Register

Address: 0x4001800C, Reset: 0x0000 0000, Name: FEEADR

Table 52. FEEADR Register Bit Descriptions

Bits	Name	Description
[31:23]	RESERVED	Reserved
[22:3]	VALUE	Memory mapped address for the flash location
[2:0]	RESERVED	Reserved

Flash Data Register (Lower)

Address: 0x40018010, Reset: 0x0000 0000, Name: FEEDATL

Table 53. FEEDATL Register Bit Descriptions

Bits	Name	Description
[31:0]	VALUE	DATA[31:0] of the data to be written to flash. Note that this register holds traceability information on power-up. The upper nibble reflects the device version.

Flash Data Register (Upper)

Address: 0x40018014, Reset: 0x0000 0000, Name: FEEDATH

Table 54. FEEDATH Register Bit Descriptions

Bits	Name	Description
[31:0]	VALUE	DATA[63:32] of the data to be written to flash

Flash Controller Page Address Registers

Lower Page Address Register

Address: 0x40018018, Reset: 0x0000 0000, Name: FEEADR1L

Table 55. FEEADR1L Register Bit Descriptions

Bits	Name	Description
[31:23]	RESERVED	Reserved.
[22:6]	VALUE	Used for locating the start address of a page in flash. Used by the erase and sign commands for specific page addresses.
[5:0]	RESERVED	The six reserved bits for byte addresses. The lower six bits of a byte address are ignored here as the sign command uses page address. Returns 0x0 if read.

Upper Page Address Register

Address: 0x4001801C, Reset: 0x0000 0000, Name: FEEADR1H

Table 56. FEEADR1H Register Bit Descriptions

Bits	Name	Description
[31:23]	RESERVED	Reserved.
[22:6]	VALUE	Used for locating the end address of a page in flash. Used only by the sign command.
[5:0]	RESERVED	The six reserved bits for byte addresses. The lower six bits of a byte address are ignored here as the sign command uses page address. Returns 0x0 if read.

Flash Controller Key Register

Address: 0x40018020, Reset: 0x0000, Name: FEEKEY

Table 57. FEEKEY Register Bit Descriptions

Bits	Name	Description
[15:0]	VALUE	Enter 0xF456 followed by 0xF123. Returns 0x0 if read.

Program Flash Write Protection Register

Address: 0x40018028, Reset: 0xFFFF FFFF, Name: FEEPROP

Table 58. FEEPROP Register Bit Descriptions

Bits	Name	Description
[31:0]	VALUE	Write protection for program flash. This register is read only if the write protection in flash has been programmed. 0: protect a section of flash. 1: leave a flash block unprotected.

Data Flash Write Protection Register

Address: 0x4001802C, Reset: 0xFF, Name: FEEPROD

Table 59. FEEPROD Register Bit Descriptions

Bits	Name	Description
[7:0]	VALUE	Write protection for data flash. This register is read only if the write protection in flash has been programmed. 0: protect a section of flash. 1: leave a flash block unprotected.

Data Flash ECC Disable Register

Address: 0x40018030, Reset: 0x0000 0000, Name: FEEECC

This register is key protected. To write to the register, the user must first write 0xF456 followed by 0xF123 to the key register.

Table 60. FEEECC Register Bit Descriptions

Bits	Name	Description
31	ECCDISDT_EN	Set to 1 for Bit 2 to Bit 0 to be valid.
[30:3]	RESERVED	Reserved.
[2:0]	ECCDISDT	Page number for which ECC is to be disabled in data flash. Only one page can be disabled at a time. Also with ECC disabled on a data flash page, a write does not update the ECC bits.

Table 61. FEEECC Register Write Examples

FEEECC[31:0]	Result on Data Flash Page
0x80000000	Page 0 ECC disabled
0x80000001	Page 1 ECC disabled
0x80000002	Page 2 ECC disabled
0x80000003	Page 3 ECC disabled
0x80000004	Page 4 ECC disabled
0x80000005	Page 5 ECC disabled
0x80000006	Page 6 ECC disabled
0x80000007	Page 7 ECC disabled
0x0000000X	ECC enabled on all pages

Flash Controller Signature Register

Address: 0x40018034, Reset: 0x0000 0000, Name: FEESIGN

Table 62. FEESIGN Register Bit Descriptions

Bits	Name	Description
[31:24]	RESERVED	Reserved
[23:0]	VALUE	Signature[23:0]

Serial Wire Control Register

Address: 0x40018038, Reset: 0x0001, Name: FEECON1

The FEECON1 register is key protected. To write to the FEECON1 register, the user keys must be entered in the FEEKEY register. After writing to FEECON1, a 16-bit value must be written again to the FEEKEY register to lock in the key protection. Note that when serial wire debug mode is disabled, the only way to access the device is via the USERFAKEY registers; contact Analog Devices to access these registers.

Table 63. FEECON1 Register Bit Descriptions

Bits	Name	Description
[15:1]	RESERVED	Reserved. Returns 0 when read.
0	DBG	Serial wire debug enable. The kernel sets this bit to 1 when it has finished executing, thus enabling debug access for the user. 0: disable access via the serial wire debug interface. 1: enable access via the serial wire debug interface.

Flash Controller Write Abort Address Register

Address: 0x40018040, Reset: 0x0000 0000, Name: FEEABORT

Table 64. FEEABORT Register Bit Descriptions

Bits	Name	Description
[31:0]	VALUE	If a write is aborted, these bits contain the address of the location being written when the write was aborted. This register has an appropriate value if a command abort occurred. This register is read after the command is aborted and must be read before any other command is given. After a reset, the value is 0x0; however, after the initial signature check is completed, the value can be random.

Flash Controller Abort Enable Register

Address: 0x40018048, Reset: 0x0000, Name: FEEAEN0

Table 65. FEEAEN0 Register Bit Descriptions

Bits	Name	Description
[15:0]	VALUE	To allow a system interrupt to abort a write or a command (erase, sign, or mass verify), write a 1 to the appropriate bit in this register. The appropriate bit is determined by the interrupt required to abort the flash command. For example, if the external IRQ1 is required to abort a flash command, set FEEAEN0 = 0x4. FEEAEN0[14:0] enable interrupts 14 to 0 to abort the flash operation. See Table 5.

Flash Controller User Failure Analysis Key Registers

USERFAKEY Low Register [31:0]

Address: 0x40018068, Reset: 0x0000 0000, Name: USERFAKEY0

Table 66. USERFAKEY0 Register Bit Descriptions

Bits	Name	Description
[31:0]	VALUE	User failure analysis key, USERFAKEY[31:0]. The user FA key is a 64-bit key that is used to disable user read protection. It is the responsibility of the user to program this key to a value. To enable this operation, 0xF456 followed by 0xF123 must first be written to the FEEKEY register (this is to prevent accidental setting) before the user FA key can be written. This key must be shared with Analog Devices, should failure analysis be required to enable access to user code.

USERFAKEY High Register [63:32]

Address: 0x4001806C, Reset: 0x0000 0000, Name: USERFAKEY1

Table 67. USERFAKEY1 Register Bit Descriptions

Bits	Name	Description
[31:0]	VALUE	USERFAKEY[63:32]

Flash Controller ECC Registers

Program Flash Address for ECC Error Register

Address: 0x40018074, Reset: 0x0000 0000, Name: FEEPECC

Table 68. FEEPECC Register Bit Descriptions

Bits	Name	Description
[31:23]	RESERVED	Reserved.
[22:3]	VALUE	Address in program flash where the ECC error is detected. The contents are not cleared on a read.
[2:0]	RESERVED	Reserved.

Data Flash Address for ECC Error Register

Address: 0x40018078, Reset: 0x0000 0000, Name: FEEDECC

Table 69. FEEDECC Register Bit Descriptions

Bits	Name	Description
[31:23]	RESERVED	Reserved.
[22:3]	VALUE	Address in data flash where the ECC error is detected. The contents are not cleared on a read.
[2:0]	RESERVED	Reserved.

SRAM

SRAM INTERFACE FEATURES

The ADuCM330/ADuCM331 feature 6 kB of SRAM organized as 6144 data bytes, that is, 1536 words, which are located at 0x2000 0000. The SRAM space can be used as data memory and as a volatile program space.

SRAM Integrity

The ADuCM330/ADuCM331 implement ECC on the SRAM by adding 7 bits to 32-bit words. This allows single-bit correction and 2-bit detection. One-bit ECC errors can generate an SRAM ECC interrupt by enabling SRAMCTRL[1].

SRAM Initialization

After a power-on reset or any reset event, the kernel checks all SRAM locations for ECC errors. If the kernel finds any location with an ECC error (2-bit or more), the entire SRAM is reinitialized with valid data (0s) with ECC, and a software reset is generated.

SRAM Control Register

Address: 0x4000202C, Reset: 0x0001, Name: SRAMCTRL

Table 70. SRAMCTRL Register Bit Descriptions

Bits	Name	Description
15	SRAM INITIALIZATION COMPLETE	This is a read only bit. This bit is valid only if Bit 2 is set. 0: cleared to 0 on power up. 1: set to 1 after SRAM initialization is complete to indicate that all SRAM locations are written to 0 with valid ECC codes.
[14:3]	RESERVED	Reserved. Reads back 0x00.
2	SRAM INITIALIZATION ENABLE	0: cleared to 0 on power up. This bit is self cleared and returns to 0 by itself. The user does not need to clear the bit after setting. 1: set by the user to enable SRAM initialization.
1	SINGLE BIT ERROR INTERRUPT ENABLE	0: cleared by the user to disable interrupt. This bit is cleared by the kernel after power-up. 1: set by the user to enable the interrupt in the event that a single bit SRAM error is detected and corrected by ECC.
0	RESERVED	Reserved. The user must write 1 to this bit.

SRAM Error Location Register

Address: 0x40002030, Reset: 0x0000, Name: SRAMERR

Table 71. SRAMERR Descriptions

Bits	Name	Description
[15:0]	SRAM ERROR LOCATION	This is a read only register. It contains the address of the last SRAM access that caused an ECC error (1 bit or 2 bits). The address is an offset to the start address of 0x2000 0000.

ADC

ADC FEATURES

- 20-bit, current ADC (ADC0)
- 20-bit, voltage/temperature ADC (ADC1)

ADC OVERVIEW

The ADuCM330/ADuCM331 incorporate two Σ-Δ, analog-to-digital converters (ADCs). ADC0 is a 20-bit (19 data bits, 1 sign bit) current ADC (IADC). ADC1 is a 20-bit voltage/temperature ADC (VADC/TADC).

These precision measurement channels integrate attenuator, on-chip buffering; a programmable gain amplifier; Σ-Δ modulators; and digital filtering for precise measurement of current, voltage, and temperature variables in 12 V automotive battery systems.

The simplified ADC transfer function can be described as follows.

- Current ADC:

$$V_{IN} = \frac{ADC0DAT \times V_{REF}}{2^{28} - 1}$$

- Voltage ADC:

$$V_{IN} = \frac{24(ADC1DAT \times V_{REF})}{2^{28} - 1}$$

  Note that the multiply by 24 in the voltage channel formula is not required for the VINx_AUX inputs.
- Temperature ADC:

$$V_{IN} = \frac{ADC1DAT \times V_{REF}}{2^{28} - 1}$$

$$Temp_{KELVIN} = \frac{V_{IN}}{274\ \mu V}$$

$$Temp_{CELSIUS} = Temp_{KELVIN} - 273.15$$

To use the factory calibrated gain coefficient for the internal temperature sensor, copy the value from memory location 0x000207EA into the ADC2GN register. The factory default offset coefficient is held at location 0x000207E8.

Four options are available for the ADC reference voltage:

- External reference to the VREF pin (default configuration)
- Internal reference (1.2 V) to AGND
- AVDD18 to AGND
- AVDD18 to GND_SW for temperature channel

The supply voltage is 1.8 V from the LDO. The two ADCs can independently select a different reference or the same reference.

Note that if using the internal 1.2 V reference, ensure that the ADC reference and reference buffer are enabled via the HRFCTRL and IRFPD registers.

Current ADC (IADC)

The IADC converts battery current sensed through, for example, an external 100 μΩ shunt resistor. On-chip programmable gain means that the IADC can be configured to accommodate battery current levels of up to ±1500 A.

As shown in Figure 10, the IADC employs a Σ-Δ conversion technique to realize 19 bits plus a sign bit of no missing codes performance.

There are four pairs of input differential signals, which are selected inside the input mux:

- IIN+/IIN−
- IIN+/IIN− (internal short configuration)
- (AVDD18/136)/GND (voltage input to diagnostic)
- IIN+_AUX/IIN−_AUX (auxiliary channel)

The Σ-Δ modulator converts the sampled input signal into a digital pulse train, whose duty cycle contains the digital information. A modified sinc3 or sinc4, programmable, low-pass filter is employed to decimate the modulator output data stream to give a valid 20-bit data conversion result at programmable conversion rates from 10 Hz to 8 kHz in normal mode and 1 Hz to 656 Hz in low power mode.

The IADC also incorporates a counter, comparator, and accumulator logic. This allows the IADC result to generate an interrupt after a predefined number of conversions have elapsed, or if the IADC result exceeds a programmable threshold value. When enabled, a 32-bit accumulator automatically sums the 20-bit IADC results.

The time to a first valid (fully settled) result on the current channel is three ADC conversion cycles, with chop mode turned off, and two ADC conversion cycles, with chop mode turned on. An interrupt can be generated even on unsettled ADC samples by enabling the ADC continuous interrupt option.

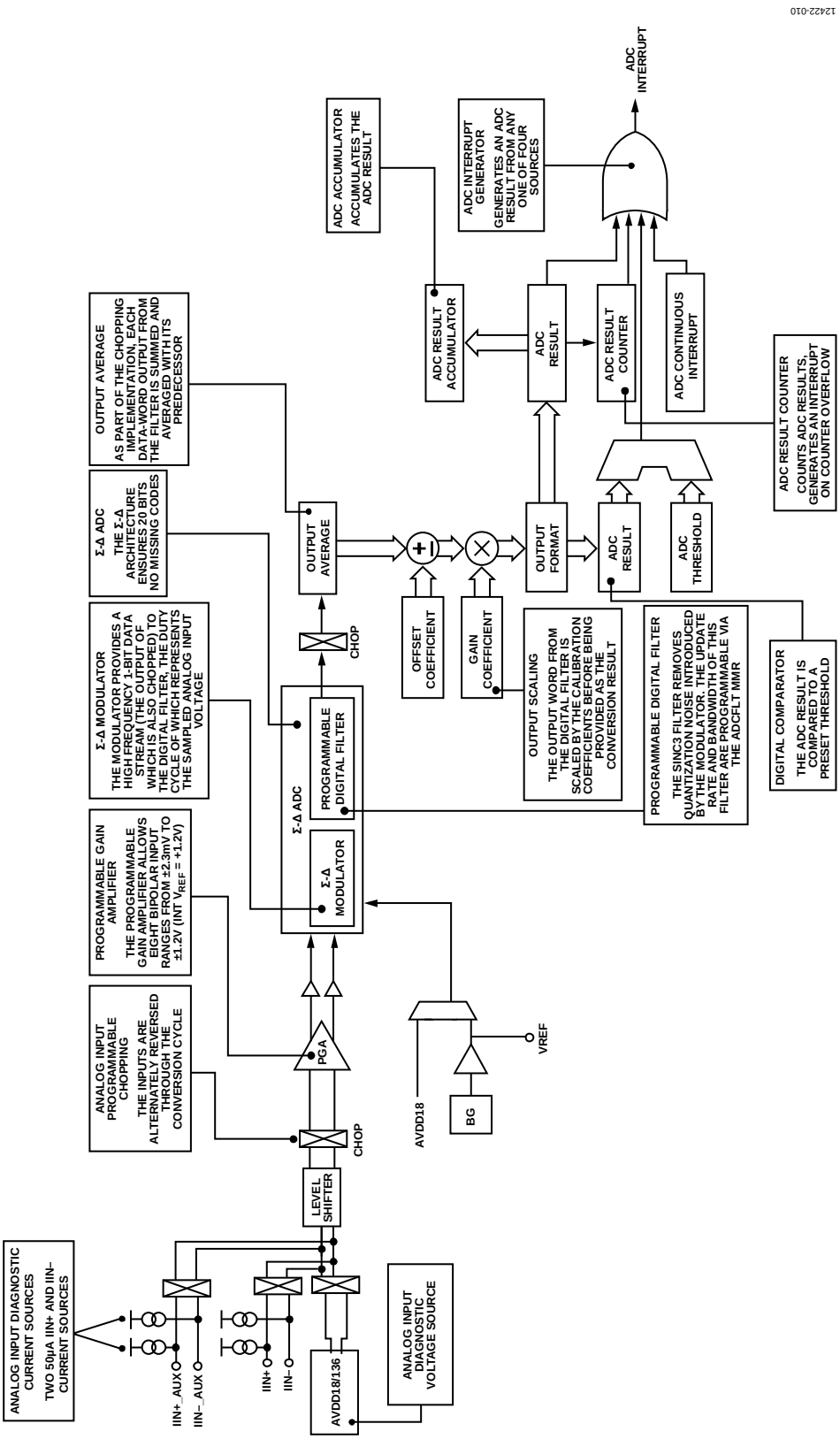

Figure 10. Current ADC Top Level Overview

Voltage/Temperature ADC (VADC/TADC)

The VADC/TADC converts additional battery parameters, such as voltage and temperature. The input to this channel can be multiplexed from an external voltage and an on-chip temperature sensor.

There are five different input signal pairs, which are selected inside the input mux:

- VBAT/AGND (VBAT attenuator selected)
- VTEMP/GND_SW (external temperature sensor)
- Vbe1/Vbe2 (internal temperature sensor)
- VINP_AUX/VINM_AUX (auxiliary voltage input)
- Vbe/GND (VIN for ADC diagnostic)

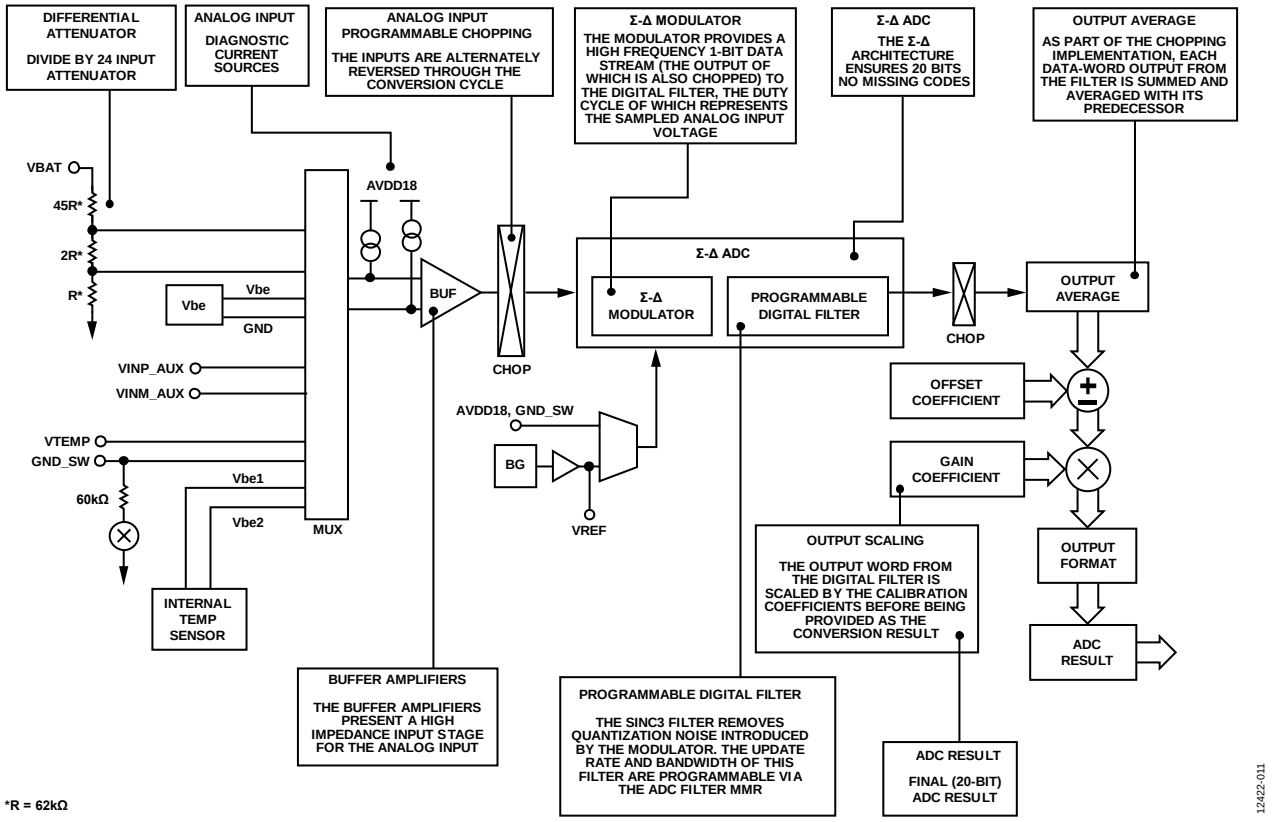

Figure 11. Voltage/Temperature ADC Top Level Overview

As with the current ADC, the VADC/TADC employs an identical Σ-Δ conversion technique, including a modified sinc3 or sinc4 low-pass filter to give a valid 20-bit data conversion result at programmable output rates from 4 Hz to 8 kHz in normal mode and 1 Hz to 656 Hz in low power mode. An external RC filter network is not required, because this is internally implemented in the voltage channel.

The external battery voltage (VBAT) is routed to the ADC input via an on-chip, high voltage (divide by 24), resistive attenuator. This ADC channel, unlike the current channel, has a fixed input range of 0 V to 28.8 V on VBAT.

The battery temperature can be derived through the on-chip temperature sensor.

By default, the time to a first valid (fully settled) result after an input channel switch on the voltage/temperature channel is three ADC conversion cycles, with chop mode turned off. A top level overview of the ADC signal chain is shown in Figure 11.

ADC Ground Switch

The ADuCM330/ADuCM331 feature an integrated ground switch pin, GND_SW. This switch allows the user to dynamically disconnect ground from external devices and allows a connection to ground using a 60 kΩ resistor, reducing the number of external components required for an NTC circuit, as shown in Figure 12. The control bit for this switch is ADCCFG[7]. The ground switch feature can be used for reducing power consumption on application specific boards.

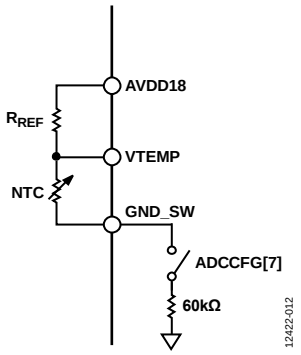

Figure 12. Internal Ground Switch Configuration

ADC Noise

Table 72 lists the output rms noise in microvolts for some typical output update rates on the IADC. The numbers are typical and are generated at a differential input voltage of 0 V. The output rms noise is specified as the standard deviation (or 1 ∑) of the distribution of ADC output codes collected when the ADC input voltage is at a dc voltage. It is expressed in microvolts rms (μV rms).

Table 72. Typical Output RMS Noise of Current ADC in Normal Power Mode[1, 2]

ADCFLT	Data Update Rate (Hz)	Gain = 4 (μV)	Gain = 8 (μV)	Gain = 16 (μV)	Gain = 32 (μV)	Gain = 64 (μV)	Gain = 512 (μV)
0x0961F	10	0.231	0.123	0.086	0.079	0.072	0.063
0x0007F	50	0.56	0.30	0.20	0.168	0.159	0.152
0x00007	1000	2.60	1.60	1.10	0.80	0.75	0.627
0x08101	994	3.40	2.10	1.50	1.00	0.80	0.840
0x18001	994	3.02	1.62	1.11	0.94	0.877	0.893
0x10001	4000	4.35	2.40	2.00	1.70	1.60	1.28
0x10000	8000	6.02	3.43	2.30	1.94	1.85	1.72

[1] PGASCALE = 1
[2] The maximum absolute input voltage allowed is −200 mV to +300 mV, relative to ground.

ADC OPERATION

Power Modes

Two power modes are available on the ADuCM330/ADuCM331:

- Normal mode
- Low power mode

The ADCs can be configured into reduced (low power) or full power (normal) mode of operation by configuring ADCMDE[3] as appropriate. The Cortex-M3 processor can also be configured in low power modes of operation (PWRMOD[2:0]). The processor power modes are independently controlled and are not related to the ADC power modes described in the following sections.

ADC Normal Power Mode

In normal mode, the current and voltage/temperature channels are fully enabled. The ADC modulator clock is 512 kHz and enables the ADCs to provide regular conversion results at a rate of between 10 Hz and 8 kHz. Both channels are under full control of the core and can be reconfigured at any time. The default ADC update rate for all channels in this mode is 1 kHz.

The IADC and VADC/TADC channels can be configured to initiate periodic, normal power mode, high accuracy, and single conversion cycles before returning to ADC full power-down mode. This flexibility is facilitated under full core control via the ADCMDE MMR, and ensures that continuous periodic monitoring of battery current, voltage, and temperature settings is feasible while ensuring that the average dc current consumption is minimized.

ADC Low Power Mode

In ADC low power mode, the IADC is enabled in a reduced power and reduced accuracy configuration. The ADC modulator clock is 128 kHz and enables the ADCs to provide regular conversion results at a rate of between 1 Hz to 656 Hz. All of the ADC peripheral functions (result counter, digital comparator, and accumulator) can be enabled in low power mode, requiring no extra power. Low power mode is designed for gains of 64 to 512 specifically.

Typically, in low power mode, the IADC only is configured to run at a low update rate and to continuously monitor battery current. The processor is in power-down mode and wakes up when the IADC interrupts the core. This interrupt occurs when the IADC detects a current conversion beyond a preprogrammed threshold, a set point, or a set number of conversions.

Bipolar/Unipolar Configuration

The analog inputs to the ADuCM330/ADuCM331 can accept either unipolar or bipolar input voltage ranges.

A bipolar voltage does not imply that the device can handle negative voltages with respect to ground. The input range can vary above or below the common-mode voltage by the value of V_{REF} as long as the absolute input voltage range is not exceeded.

Typical ADC Modes of Operation

The ADC can be configured to operate in one of four different general modes of operation:

- ADC conversion: can be continuous conversions at a fixed rate or single conversions triggered by software.
- ADC idle mode: the ADC is fully powered on but held in reset.
- ADC calibration modes: this mode is used to remove any ADC and system errors where possible.
- ADC power-down modes: the ADC is powered down to reduce overall system power consumption, particularly between single conversions.

Conversion Modes

In normal ADC operation mode, two conversion modes are possible:

- Single conversion: ADCMDE = 0x2. A single ADC conversion can be initiated in software by setting Bit 1 of the ADCMDE register. After a single conversion is completed, the ADC returns to idle mode.
- Continuous conversion: ADCMDE = 0x1. Continuous conversion mode results in the ADCxDAT register being updated at the sampling rate selected by the ADCFLT register.

When a conversion is complete in either mode, the ready flag in the ADCSTA register is asserted, indicating that the ADC result is now present in the ADC0DAT or ADC1DAT register for reading. These ready flags can be configured to flag an interrupt to the ARM Cortex-M3 processor. If an error occurs in the conversion due to an underrange or overrange error in the input voltage to either ADC, the error is set in the appropriate ADCSTA register (Bit 13 or Bit 12).

ADC Idle Mode

In idle mode, the ADC is fully powered on but held in reset. The device enters this mode after calibration or between single conversions. The current consumption is reduced relative to fully active mode.

Power-Down Mode

In power-down mode, the ADCs and the input amplifiers are fully powered off for maximum power reduction. Before entering hibernate mode, it is recommended to complete the following ADC power-down sequence:

```
disable 1.2Vref           (HRFCTRL)
disable 1.2Vrefbuffer     (IRFPD)
disable ADC operation     (all other ADC MMRs)
disable ADC interrupts    (ADCMSKI)
disable ADC1              (ADC1CON)
disable ADC0              (ADC0CON)
_DSB()
enter PWDN Mode           (ADCMDE)
```

Modifying ADC Settings

When changing ADC settings, switch the ADCs to idle mode, which holds the ADCs in reset while remaining fully settled. The user must avoid powering down the ADCs, either individually via Register ADCxCON, Bit 19, or both at the same time via Register ADCMDE, Bits[2:0].

With data output rates larger than 1 kHz, powering down the ADCs can result in the ADCs not being fully settled when the next conversion begins.

```
set Idle Mode             (ADCMDE)
change ADC1               (ADC1CON)
change ADC0               (ADC0CON)
change ADC operation      (all other ADC MMRs)
Wait(100us)
set ADC operation Mode    (ADCMDE)
```

ADC Power-Up Sequence

The following ADC power-up sequence is used after the device exits hibernate mode, or when the ADC exits power-down mode. To ensure the ADCs are operational and synchronized with the first conversion result, the following sequence is recommended:

```
config 1.2Vref            (HRFCTRL)
enable 1.2Vrefbuffer      (IRFPD)
enable ADC1               (ADC1CON)
enable ADC0               (ADC0CON)
set Idle Mode             (ADCMDE)
setup ADC operation       (all other ADC MMRs)
Wait(500us)               (settling time)
set ADC operation Mode    (ADCMDE)
```

For data rates of 4 kHz and 8 kHz with PGA gain ≥ 32, allow 10 ms settling time after the ADCs wake up from power-down mode.

ADC CALIBRATION

As shown in the top level diagrams (Figure 10 and Figure 11), the signal flow through all ADC channels can be described in the following steps:

1. An input voltage is applied through an input buffer (and PGA, in the case of the IADC) to the Σ-Δ modulator.
2. The modulator output is applied to a programmable digital decimation filter.
3. The filter output result is averaged, if chopping is used.
4. An offset value (ADCxOF) is subtracted from the result.
5. This result is scaled by a gain value (ADCxGN).
6. The result is formatted as twos complement/unipolar or clamped to ±full scale.

Each ADC channel (current, voltage, and temperature) has a specific offset and gain correction or calibration coefficient associated with it; these coefficients are stored in MMR-based offset and gain registers (ADCxOF and ADCxGN). The offset and gain registers can be used to remove system level offset and gain errors external to the device.

These registers are configured at power-on with a factory programmed calibration value. These factory calibration values vary from device to device, reflecting the manufacturing variability of internal ADC circuits. These registers can also be overwritten by user code after a calibration.

On the current channel, when a system calibration is initiated, the ADC generates its calibration coefficient based on an externally generated zero-scale voltage and full-scale voltage, which are applied to the external ADC input for the duration of the calibration cycle. The coefficients are written in the ADC0DAT MMR of the ADC channels; they are not automatically written in the ADC0OF or ADC0GN MMR. User code must copy these values to their appropriate registers.

The duration of an offset calibration is a full ADC filter settling time before returning the ADC to idle mode. When a calibration cycle is initiated, any ongoing ADC conversion is immediately halted, the calibration is automatically carried out at an ADC update rate programmed into ADCFLT, and the ADC returns to idle after any calibration cycle. It is strongly recommended that ADC calibration is initiated at as low an ADC update rate as possible (high SF value in ADCFLT) to minimize the impact of ADC noise during calibration.

On the voltage channel, a two-point calibration must be performed, because the minimum voltage specified on the input is 4 V.

The temperature channel is factory calibrated for the internal temperature sensor.

Calibrating the Voltage Channel

To calibrate the offset and gain of the voltage channel, a two-point calibration method must be used. This method consists of converting two known voltages (for example, 8 V and 16 V) to determine the slope and offset of the transfer function. The gain coefficient can be divided by the calculated slope to improve the gain error.

The offset error can be reduced by writing ⅜ of the calculated offset (in unipolar codes) into the ADC1OF MMR.

Calibrating the Current Channel

If the chop bit (ADCFLT[15]) is enabled, internal ADC offset errors are minimized, and an offset calibration may not be required. If chopping is disabled, however, an initial offset calibration is required and may need to be repeated, particularly after a large change in temperature.

A gain calibration, particularly in the context of the IADC (with internal PGA), may need to be carried out at all relevant system gain ranges, depending on system accuracy requirements.

If it is not possible to apply an external full-scale current on all gain ranges, the user can apply a lower current and scale the result produced by the calibration. For example, apply a 50% current, divide the ADC0DAT value produced by two, and write this value back into ADC0GN.

Because ADC0GN is a 16-bit register, a lower limit can be applied to the input signal for system calibration. The input span (difference between the system zero-scale value and the system full-scale value) must be greater than 40% of the nominal full-scale-input range, that is, >40% of V_{REF}/gain.

The on-chip Flash/EE memory can be used to store multiple calibration coefficients. These coefficients can be copied by user code directly into the relevant calibration registers, as appropriate, based on the system configuration.

A factory or end-of-line calibration for the IADC is a two-step procedure.

1. Apply the 0 A current. Configure the ADC in the required PGA setting and other required settings, and write to ADCMDE[2:0] to perform a system zero-scale calibration. This process writes a new offset calibration value into ADC0DAT. User code must store this value into ADC0OF or into Flash/EE memory.
2. Apply a full-scale current for the selected PGA setting. Write to ADCMDE[2:0] to perform a system full-scale calibration. This process writes a new gain calibration value into ADC0DAT. This value must be copied by user software to the ADC0GN MMR or into Flash/EE memory.

Understanding the Offset and Gain Calibration Registers

The output of the average block in the ADC signal flow can be considered a fractional number with a span for a ±full-scale input of approximately ±0.75. The span is less than ±1.0 because there is attenuation in the modulator to accommodate some overrange capacity on the input signal. The exact value of the attenuation varies slightly between devices because of manufacturing tolerances.

The offset coefficient is read from the ADC0OF calibration register. This value is a 24-bit, twos complement number.

A positive value of ADC0OF indicates that, when offset is subtracted from the output of the filter, a negative value is added. The nominal value of this register is 0x0000, indicating zero offset is to be removed. The actual offset of the ADC can vary slightly between devices and at different PGA gains. The offset within the ADC is minimized if chopping mode is active (ADCFLT[15] = 1).

The gain coefficient is a unitless scaling factor. The nominal value of this register equals 0x5555, corresponding to a multiplication factor of 1.3333. This factor scales the nominal ±0.75 signal to produce a full-scale output signal of ±1.0, which is checked for overflow/underflow and converted to twos complement or unipolar mode, as appropriate, before being output to the data register.

The actual gain, and the required scaling coefficient for zero gain error, varies slightly from device to device and at different PGA settings. The value downloaded into ADC0GN at power-on reset represents the scaling factor for a PGA gain of 8. There is some level of gain error if this value is used at different PGA settings. User code can run ADC calibrations and overwrite the calibration coefficients to correct the gain error at the current PGA setting.

The ADC transfer function, taking offset and gain calibration factors into consideration, can be described as follows.

For the current ADC, with the PGA scale enabled:

$$ADC0DAT = \left(\frac{V_{IN} \times (2^{28} - 1)}{V_{REF}} \times PGA\ Scale - ADC0OF \times \frac{4}{3} \times \frac{2^9}{2^4} \div GN \right) \times \frac{ADC0GN}{ADC0GN_{NOM}}$$

For the current ADC, with the PGA scale disabled:

$$ADC0DAT = \left(\frac{V_{IN} \times (2^{28} - 1)}{V_{REF}} - ADC0OF \times \frac{4}{3} \times \frac{2^9}{2^4} \div GN \right) \times \frac{ADC0GN}{ADC0GN_{NOM}}$$

For the voltage ADC:

$$ADC1DAT = \left(\frac{\frac{V_{IN}}{24} \times (2^{28} - 1)}{V_{REF}} - ADC1OF \times \frac{4}{3} \times \frac{2^9}{2^4} \right) \times \frac{ADC1GN}{ADC1GN_{NOM}}$$

Note that $V_{IN}/24$ is not required for the VINx_AUX inputs.

For the temperature ADC:

$$ADC2DAT = \left(\frac{V_{IN} \times (2^{28} - 1)}{V_{REF}} - ADC2OF \times \frac{4}{3} \times \frac{2^9}{2^4} \right) \times \frac{ADC2GN}{ADC2GN_{NOM}}$$

For these equations, note the following:

- $2^9/2^4$ is used to transform ADCxOF format (24-bit data) to ADCxDAT format (32-bit data).
- 4/3 is used to scale ADCxOF data back to actual data, because ADCxOF is 0.75 of the actual offset.
- $ADCxGN_{NOM}$ = 0x5555.
- ADC works in twos complement mode.

Calibration Modes

Self Offset Calibration: ADCMDE = 0x4

In this mode, an offset calibration is performed on any enabled ADC using an internally generated 0 V signal. The calibration is carried out at the user programmed ADC settings; therefore, as with a normal single ADC conversion, it takes two to three ADC conversion cycles before a fully settled calibration result is ready. The calibration result is automatically written to the ADC data register of the respective ADC; the user must transfer the results to the ADCxOF register. After a device reset, the ADCxOF register is reloaded with the factory calibration value.

Self Gain Calibration: ADCMDE = 0x5

In this mode, a gain calibration against selected reference voltage is performed on all enabled ADCs. A gain calibration is a two-stage process and takes twice the time of an offset calibration. The calibration result is automatically written to the ADC data register of the respective ADC; the user must transfer the results to the gain register. The internal full-scale calibration does not work for gain settings greater than 1. After a device reset, the ADC gain register is reloaded with the factory calibration value.

System Zero-Scale Calibration: ADCMDE = 0x6

In this mode, a zero-scale calibration is performed on enabled ADC channels against an external zero-scale voltage driven at the ADC input pins. Usually, the selected channel is shorted externally. The calibration result is automatically written to the ADC data register of the respective ADC. The user must transfer the results to the offset register.

System Full-Scale Calibration: ADCMDE = 0x7

In this mode, a full-scale calibration is performed using the enabled ADC channels against an external full-scale voltage driven at the ADC input pins. The ADC gain register is updated after a full-scale calibration sequence. After a device reset, the ADC gain register is reloaded with the factory calibration value.

Note that, when the ADC is delivering samples at a rate of $1/f_{ADC}$, take care when modifying the ADCxGN and ADCxOF registers to avoid writing to them while the ADC is reading these registers. Ensure that the ADC is in idle mode when modifying the ADCxGN and ADCxOF registers, or ensure that the ADCxGN and ADCxOF registers are modified between the start of the conversion to $1/f_{ADC}$ − 344 clock cycles (16 MHz).

It is strongly recommended that ADC calibration is initiated at the lowest ADC update rate possible (high SF value in ADCFLT) to minimize the impact of ADC noise during calibration. Do not use calibration registers for coarse scaling of input ranges.

ADC SINC3 DIGITAL FILTER RESPONSE

The overall frequency response on all ADuCM330/ADuCM331 ADCs is dominated by the low-pass filter response of the on-chip, sinc3 and sinc4 digital filters. The sinc3 and sinc4 filters are used to decimate the ADC Σ-Δ modulator output data bit stream and to generate a valid 20-bit data result. The digital filter response is identical for both ADCs and is configured via the 20-bit ADC filter (ADCFLT) register. This register determines the overall throughput rate of the ADCs. The noise resolution of the ADCs is determined by the programmed ADC throughput rate. In the case of the current ADC, the noise resolution is determined by the throughput rate and selected gain.

The overall frequency response and the ADC throughput is dominated by the configuration of the sinc3 filter decimation factor (SF) bits (ADCFLT[6:0]) and the averaging factor (AF) bits (ADCFLT[13:8]). Due to limitations on the digital filter internal data path, there are some limitations on the allowable combinations of SF and AF that can be used to generate a required ADC output rate. This restriction limits the minimum ADC update in normal power mode to 4 Hz (chop on, AF = 60, SF = 31). The calculation of the ADC throughput rate is detailed in Table 88.

By default, ADCFLT = 0x0007 configures the ADCs for a throughput rate of 1.0 kHz with all other filtering options (chop, running average, averaging factor, and sinc3 modify) disabled. A typical filter response based on this default configuration is shown in Figure 13.

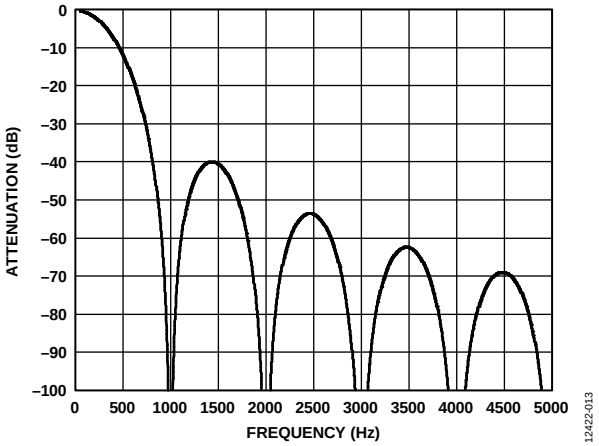

Figure 13. Typical Digital Filter Response at f_{ADC} = 1.0 kHz (ADCFLT = 0x0007)

An additional sinc3 modify bit (ADCFLT[7]) is also available in the ADCFLT register. This bit is set by user code to modify the standard sinc3 frequency response, increasing the filter stop-band rejection by approximately 5 dB. This modification is achieved by inserting a second notch (NOTCH2) at

$$f_{NOTCH2} = 1.333 \times f_{NOTCH}$$

where f_{NOTCH} is the location of the first notch in the response.

There is a slight increase in ADC noise if the sinc3 modify bit is active. Figure 14 shows the modified 1 kHz filter response when the sinc3 modify bit is active. The new notch is visible at 1.33 kHz, as is the improvement in stop-band rejection when compared to the standard 1 kHz response.

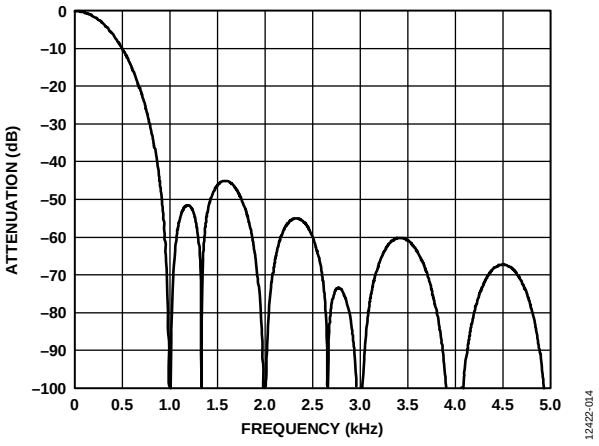

Figure 14. Modified Sinc3 Digital Filter Response at f_{ADC} = 1.0 kHz (ADCFLT = 0x0087)

At very low throughput rates, the chop bit in the ADCFLT register can be enabled to minimize offset errors and, more importantly, temperature drift in the ADC offset error.

There are two primary variables (sinc3 decimation factor and averaging factor) available to allow the user to select an optimum filter response, trading off filter bandwidth against ADC noise. For example, with the chop bit (ADCFLT[15]) set to 1, increasing the SF value (ADCFLT[6:0]) to 0x1F (31 decimal) and selecting an AF value (ADCFLT[13:8]) of 0x16 (22 decimal) results in an ADC throughput of 10 Hz. The frequency response in this case is shown in Figure 15.

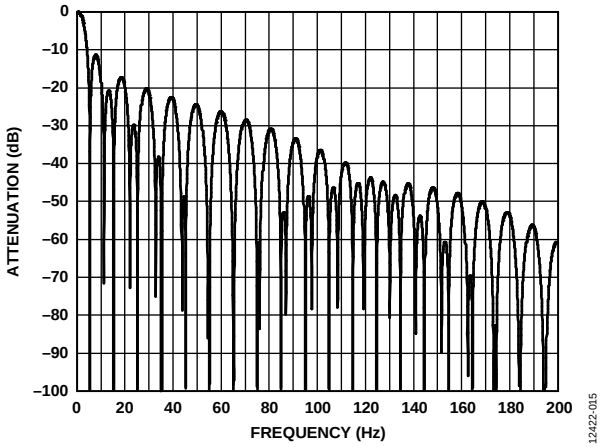

Figure 15. Typical Digital Filter Response at f_{ADC} = 10 Hz (ADCFLT = 0x961F)

In ADC low power mode, the Σ-Δ ADC modulator clock is no longer driven at 512 kHz but at 128 kHz. Subsequently, for the same ADCFLT configurations in normal mode, all filter values must be scaled by a factor of approximately four.

In general, it is possible to program different values of SF and AF in the ADCFLT register and achieve the same ADC update rate. In practical terms, the trade off with any value of ADCFLT is frequency response versus ADC noise. For optimum filter response and ADC noise when using combinations of SF and AF, the best practice is to choose an SF in the range of 16 decimal to 40 decimal, or 0x10 to 0x28, and increasing the AF value to achieve the required ADC throughput. Table 73 shows some common ADCFLT configurations.

Table 73. Common ADCFLT Configurations

ADC Mode	SF	AF	Other Configurations	ADCFLT	f_{ADC}	$t_{SETTLING}$
Normal	0x1F	0x16	Chop on	0x961F	10 Hz	0.2 sec
Normal	0x07	0x00	None	0x0007	1 kHz	3 ms
Normal	0x07	0x00	Sinc3 modify	0x0087	1 kHz	3 ms

DIGITAL FILTER OPTIONS

Example Values of SF and AF

Normal mode:

- f_{ADC} = 1 kHz: CHP = 0, AF = 0, SF = 7
- f_{ADC} = 1 kHz: CHP = 0, AF = 1, SF = 1 (single cycle settling)
- f_{ADC} = 50 Hz: CHP = 0, AF = 0, SF = 127
- f_{ADC} = 10 Hz: CHP = 1, AF = 22, SF = 31
- f_{ADC} = 4 Hz: CHP = 1, AF = 60, SF = 31 (3.97 Hz)
  Note that setting RAVG2 improves noise performance.

Low power mode:

- f_{ADC} = 10 Hz: CHP = 1, AF = 2, SF = 39 (f_{MOD} = 128 kHz)

Table 74. ADC Conversion Rates and Settling Times[1]

Sinc4_EN	Chop Enabled	Running Average	Averaging Factor	f_{ADC}	$t_{SETTLING}$
No	No	No	No	$f_{MOD} \div [64 \times (SF + 1)]$	$3 \div f_{ADC}$
No	No	Yes	No	$f_{MOD} \div [64 \times (SF + 1)]$	$4 \div f_{ADC}$
No	No	No	Yes	$f_{MOD} \div [64 \times (SF + 1) \times (3 + AF)]$	$1 \div f_{ADC}$
No	No	Yes	Yes	$f_{MOD} \div [64 \times (SF + 1) \times (3 + AF)]$	$2 \div f_{ADC}$
No	Yes	Yes/no	Yes/no	$f_{MOD} \div [64 \times (SF + 1) \times (3 + AF) + 3]$	$2 \div f_{ADC}$
Yes	No	No	Not applicable	$f_{MOD} \div [64 \times (SF + 1)]$	$4 \div f_{ADC}$
Yes	No	Yes	Not applicable	$f_{MOD} \div [64 \times (SF + 1)]$	$5 \div f_{ADC}$
Yes	Yes	Yes/no	Not applicable	$f_{MOD} \div [64 \times (SF + 1) \times 4) + 3]$	$2 \div f_{ADC}$

[1] f_{MOD} = 512 kHz in normal mode, 128 kHz in low power mode. For $t_{SETTLING}$, an additional 60 µs (approximately) per ADC is required before the first ADC result is available.

Table 75. Allowable Combinations of SF and AF

	AF Range		
SF	**0**	**1 to 7**	**8 to 63**
1:31	Yes	Yes	Yes
32:63	Yes	Yes	No
64:127	Yes	No	No

The combination limit shown in Table 75 is only for sinc3 mode.

The ADuCM330/ADuCM331 also incorporate a sinc4 digital filtering option. The sinc4 filter offers reduced noise, particularly at high (>2 kHz) output rates, and is recommended for these applications. Contact Analog Devices for a spreadsheet application showing the ADC frequency response for both sinc3 and sinc4 filtering options.

For sinc4 mode:

- AF is not applicable and always forced to be zero.
- SF must be no greater than 0x0E. If SF is set to greater than 0x0E in sinc4 mode, it is automatically forced to 0x0E.

ADC CONFIGURATION

Fast Temperature Conversion Mode

The battery temperature can be derived through the on-chip temperature sensor. By default, the time to a first valid (fully settled) result after switching the ADC input from the voltage to the temperature channel or from the temperature to the voltage channel is three ADC conversion cycles with chop mode turned off, as shown in Figure 16.

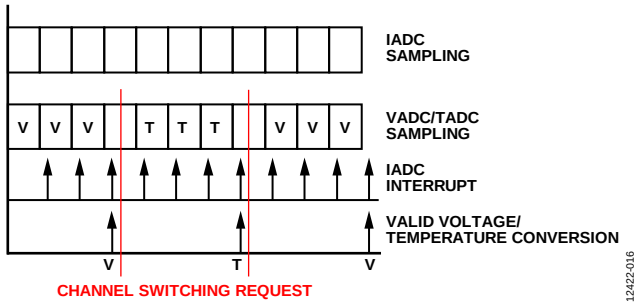

Figure 16. Default Temperature Mode, Chop Off

A fast mode is provided on the temperature channel to minimize the switching delay between voltage conversion and temperature conversions, as shown in Figure 17 and in Table 76.

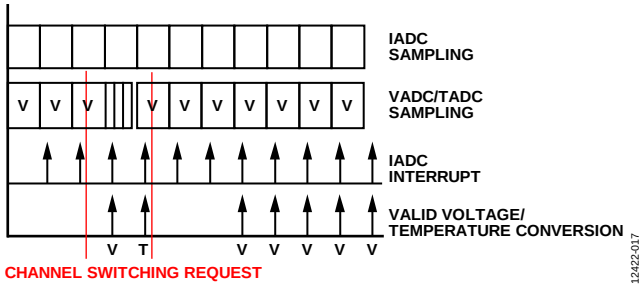

Figure 17. Fast Temperature Mode, Chop Off (ADCFLT = 0x07)

A request for a fast temperature conversion is executed with a delay of one ADC conversion. The fast temperature mode must be cleared after the temperature measurement is available and before a new temperature request.

Table 76. Fast Temperature Mode

Interrupt	Valid Flags	User code
1	I and V	Voltage = ADC1DAT.
2	I and V	Voltage = ADC1DAT. Set fast temperature request bit.
3	I and V	Voltage = ADC1DAT. This data must be read for the next temperature channel flag to be valid.
4	I and T	Temperature = ADC1DAT. Clear fast temperature request bit.
5	I	
6	I	
7	I and V	Voltage = ADC1DAT.
8	I and V	Voltage = ADC1DAT.

The fast temperature option cannot be used on the first conversion after ADC power-on. The option can only be set after at least the first ADC interrupt. Waiting for a valid ADC result is not necessary. When using the fast temperature mode, ensure that SF ≥ 1. In addition, a conversion rate of 1 ms is recommended in this mode of operation, to ensure that the fast result occurs simultaneously with the current channel result.

When changing the ADCs configuration by writing to ADCMDE, ADC0CON or ADCFLT, the fast temperature bit must also be cleared to ensure correct operation. This condition is similar to a first conversion after ADC power-on.

IADC DIAGNOSTICS

The ADuCM330/ADuCM331 feature the capability to detect open circuit conditions on the application board. This detection is accomplished using the two current sources on IIN+ and IIN−; these sources are controlled via ADC0CON[13:12].

Note that these current sources have a tolerance of ±10%.

OTHER ADC SUPPORT CIRCUITS

Internal/External 1.2 V Voltage Reference

It is possible to apply an external reference to the ADuCM330/ADuCM331. To use an external reference, the following steps are required:

- Power down the internal reference buffer by setting IRFPD = 0x01. This is also the default state.
- Apply an external reference to VREF (1.2 V).

The ADCs operate as normal.

Internal 1.2 V Reference Control Register (Address 0x40008808)

Table 77. HRFCTRL Register Bit Descriptions

Bits	Name	Description
[15:2]	RESERVED	Reserved.
1	HRFAUTOB	0: automatic mode. The 1.2 V internal reference is controlled by the ADC. HRFPD is ignored. 1: user mode. The 1.2 V internal reference is controlled by HRFPD. This is the default state.
0	HRFPD	0: enable the 1.2 V internal reference. 1: power down the 1.2 V internal reference.

Internal Reference Buffer Power Down Register (Address 0x40008824)

Table 78. IRFPD Register Bit Descriptions

Bits	Name	Description
[15:1]	RESERVED	Reserved.
0	INTREFPD	0: enable internal reference buffer. 1: gates in an external 1.2 V reference, powers down internal reference buffer. This is the default state.

Note the following:

- If using an external reference, the reference buffer must be disabled via the IRFPD register. However, the internal 1.2 V reference must be kept enabled, because it is still required internally for correct functionality.
- When using the internal 1.2 V reference for the ADCs, both the reference and the buffer must be enabled.
- When using the internal 1.2 V reference for the ADCs, ensure that the reference buffer is enabled or disabled according to ADCxCON[19].
- Before entering hibernate mode, follow the recommended ADC power-down sequence (see the Power-Down Mode section).

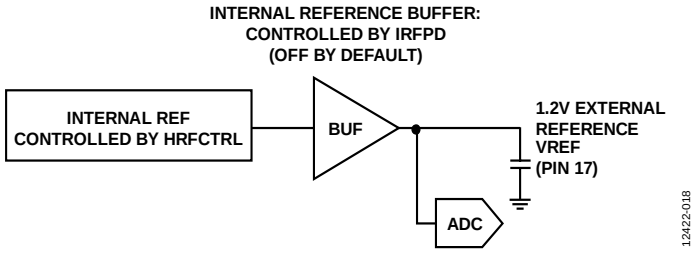

Figure 18. ADC Reference Configuration

Programmable Gain Amplifier

The primary ADC (current ADC) incorporates an on-chip, programmable gain amplifier (PGA). The gain is controlled by the ADC0CON register. The PGA allows signals of very small amplitude to be gained up while still maintaining low noise performance. Internally, the PGA requires its output range to be limited to <±1 V while still allowing a 1.2 V reference to be used.

To accommodate this requirement, there are two bits at ADC0CON[11:10] named PGASCALE. Clearing these bits (default state) activates a 13/16 × 14/16 scaling factor and maintains compatibility with gain ranges from previous IBS generations.

If the PGASCALE bits are both set to 1, ensure that the input to the PGA does not exceed 1 V/gain.

Table 79. PGASCALE

ADC0CON[3:0] = ADC0PGA	PGASCALE[11:10]	Scaling 14/16	Scaling 13/16	Resulting PGA Scaling Factor Applied
Gain	00	Gain × (14 ÷ 16)	Gain × (13 ÷ 16)	Gain × (14 ÷ 16) × (13 ÷ 16)
Gain	01	Gain × (14 ÷ 16)	1	Gain × (14 ÷ 16)
Gain	10	1	Gain × (13 ÷ 16)	Gain × (13 ÷ 16)
Gain	11	1	1	Gain × 1

ADC Comparator and Accumulator

The current ADC result can be compared to a preset threshold level (ADC0TH) configured via the ADCCFG register. An interrupt is generated if the absolute (sign independent) value of the ADC result is greater than the preprogrammed comparator threshold level. An ADC interrupt is also generated when the threshold counter reaches a preset value (ADC0RCL).

Finally, a 32-bit accumulator (ADC0ACC) contains the result of multiple primary conversions. User code can read the accumulated value directly (ADC0ACC) without any further software processing.

Diagnostic Current Sources

For diagnostic purposes, both ADCs of the ADuCM330/ADuCM331 incorporate 50 µA constant current sources. These current sources can be controlled by the appropriate bits in the ADCxCON registers. For all current input channels (IIN+/IIN− and IIN+_AUX/IIN−_AUX), independent current sources are available.

For the implementation of diagnostic functions in the application, the user must take into account the internal present RIN+/RIN− (see Figure 19). The resistors are approximately in the range of 1.8 kΩ ± 10%. In addition, the 50 µA current sources are typically ±10% accurate. Therefore, with the external recommended filter resistors, shown as R1 and R2 with a value of 221 Ω ± 1%, the minimum and maximum voltage change after switching on such a diagnostic current is typically in the range from 82 mV to 121 mV. Table 80 shows some possible fault conditions and possible tests for error detection. To conduct these tests, an appropriate gain range must be selected.

Note that for higher gain ranges (gain ≥ 8, ADC0CON[3:0] ≥ 3), in most cases, the diagnostic currents cause an underrange or overrange error status (ADCSTA[12] = 1 and ADC0DAT[0] = 1).

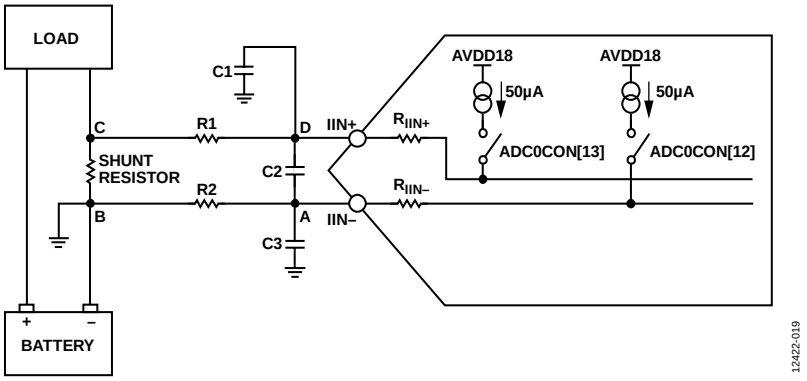

Figure 19. ADC0 Example Circuit Using Diagnostic Current Sources

Table 80. Current ADC Diagnostics

Fault Condition[1]	Tests
Short between Point C and Point D (R1 ~221 Ω shorted) or between Point A and Point D (C2 shorted) or Point D and ground (C1 shorted)	1. Turn off both current sources, and read the value from ADC0DAT. 2. Turn on the current source on IIN+, and read the value from ADC0DAT. 3. If the difference between the two measurements is notably smaller than the nominal value measured in a correct environment, one of the given fault conditions may be present. Note that depending on the actual current flowing through the shunt and the selected gain range, the diagnostic current can cause a overrange error (ADCSTA[12] = 1), and the data in ADC0DAT is not valid (ADC0DAT[0] = 1).
Short between Point A and Point B (R2 ~221 Ω shorted) or between Point A and Point D (C2 shorted) or Point A grounded (C3 shorted)	1. Turn off both current sources, and read the value from ADC0DAT. 2. Turn on the current source on input IIN−, and read the value from ADC0DAT. 3. If the difference between the two measurements is notably smaller than the nominal value measured in a correct environment, one of the given fault conditions may be present. Note that depending on the actual current flowing through the shunt and the selected gain range the diagnostic current can cause a underrange error (ADCSTA[12] = 1), and the data in ADC0DAT is not valid (ADC0DAT[0] = 1).
Open circuit between Point C and Point D (R1 unsoldered, or IIN+ unsoldered)	1. Turn on current source on input IIN+, and read the value from ADC0DAT. 2. If the value for gain ranges ≥ 2 (ADC0CON[3:0] ≥ 1) is equal to positive full scale, the IIN+ input is open circuit. Note that depending on the actual circuitry, calibration, and selected gain range, the diagnostic current can cause an overrange error (ADCSTA[12] = 1), and the data in ADC0DAT is not valid (ADC0DAT[0] = 1).
Open circuit between Point A and Point B (R2 unsoldered, or IIN− unsoldered)	1. Turn on the current source on input IIN−, and read the value from ADC0DAT. 2. If the value for gain ranges > 2 (ADC0CON[3:0] ≥ 1) is equal to negative full scale, the IIN− input is open circuit. Note that depending on the actual circuitry, calibration, and selected gain range, the diagnostic current can cause an underrange error (ADCSTA[12] = 1), and the data in ADC0DAT is not valid (ADC0DAT[0] = 1).
High resolution diagnostic allows detection of: (A) R1, R2 mismatch outside of allowed component tolerance (B) A short circuit condition on Point D or Point A, but not both (C) An open circuit condition at R1 (Point D or Point C) or at R2 (Point A or Point B), but not both resistors	1. Turn off both current sources, and read the value from ADC0DAT. 2. Turn on the current source on IIN+ only, and read the value from ADC0DAT. 3. Turn on the current source on IIN− only, and read the value from ADC0DAT. 4. If for a gain of 1 (ADC0CON[3:0] = 0) the measured changes are not symmetrical, and the differences are not in the range of the nominal measured changes, one of the given fault conditions may be present. Note that depending on the actual circuitry, calibration, and selected gain range, the diagnostic currents can cause an overrange or underrange error (ADCSTA[12] = 1), and the data in ADC0DAT is not valid (ADC0DAT[0] = 1).

[1] See Figure 19.

On the VBAT input channel, constant current sources are available on the HV die and the LV die. For the VBAT channel, the primary measurement channel, it is recommended to use the current source on the HV die for diagnostics, as shown in Figure 20. If the current source on the HV die in HVDCFG0[3] is enabled, a typical voltage of 160 mV ± 5% is added to the actual voltage measured from VBAT.

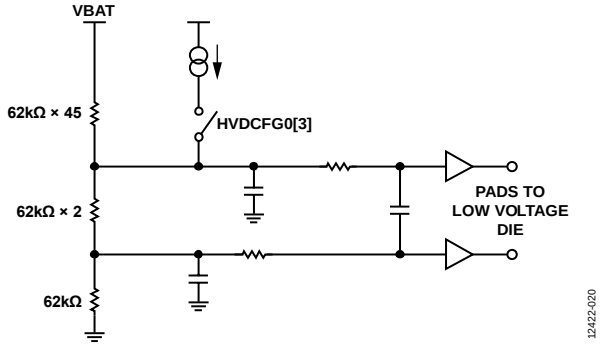

Figure 20. ADC1 Example Circuit Using Diagnostic Current Source on HV Die

ADC CHOPPING

The ADCs on the ADuCM330/ADuCM331 implement a chopping scheme whereby the ADC repeatedly reverses its inputs. Therefore, the decimated digital output values from the sinc3 or sinc4 filter have a positive and negative offset term associated with them. This results in the ADC including a final summing stage that sums and averages each value from the filter with previous filter output values. This new value is then sent to the ADC data MMR. This chopping scheme results in excellent dc offset and offset drift specifications and is extremely beneficial in applications where drift and noise rejection are required. Chopping is not active by default, meaning there is an offset error. Chopping also affects settling time, as shown in Table 74.

ADC REGISTERS

ADC Status Register (Table 83)

The various enabled ADC interrupts sources are OR'ed to produce a single ADC interrupt to the ARM core. The software then interrogates the ADCSTA MMR to determine the source of the interrupt. The ADC0RDY and ADC1RDY bits are set simultaneously if both the IADC and VADC are enabled. If a temperature channel is selected, the ADC1RDY bit is not set, but the ADC2RDY bit is set.

All the RDY bits are cleared by a read of the ADC0DAT MMR. Writing to the ADC0CON register also clears all the ADCSTA bits.

A change in the ADC1CON register only clears ADCSTA[1]. This bit is also cleared by reading the ADC1DAT register.

If the IADC is not enabled, all the RDY bits are cleared by a read of the ADC1DAT MMR. Therefore, software routines that need to access current/voltage data that is guaranteed synchronously sampled must read the ADC1DAT register before reading the ADC0DAT register; otherwise, a read of the ADC0DAT register clears the RDY1 flag, allowing the possibility of the ADC1DAT register being overwritten by new data before it is read.

Summary of How the Status Register Flags Are Reset

- The ADCxERR flags are set or cleared every time new data is written into the appropriate ADCxDAT MMR, setting the ADCxRDY bits. The flags are not cleared by a read of the ADCxDAT MMRs.
- Comparator/counter interrupts are cleared if the comparator/counter is disabled.
- The ADCxRDY bits are automatically cleared by the ADC if the ADC configuration is changed.
- The ADCxRDY flags are cleared by a read of the ADC0DAT register or a read of the ADC1DAT register if the IADC is not active.

All other interrupts and flags are also cleared if the ADC configuration is changed.

ADC Mask Register (Table 84)

This MMR allows the ADC interrupt sources in the ADCSTA register to be individually masked. The bit positions are the same as the lower five bits of ADCSTA. If a bit is set to 1, the interrupt is enabled. The default value is 0x00, that is, all ADC interrupts are inactive.

The ADCMSKI register is used to enable the lower bits of the ADCSTA register to generate an interrupt.

ADC Mode Register (Table 85)

A write to the ADCMDE register immediately resets each active ADC, including the ADCxRDY bits and other ADCSTA flags. The full digital filter settling time must elapse before the first result with the new MMR programmed configuration is available. The one exception to this condition is that an interrupt can be provided before the filter is settled if the always interrupt mode bit is set.

The results of the calibrations are written into the ADCxDAT registers. The user must copy the calibration results from the ADCxDAT registers into ADCxOF/ADCxGN.

ADC Control Register: ADC0CON (Table 86)

A write to the ADC0CON register resets the IADC and other enabled ADCs. The full digital filter settling time must elapse before a result is available.

If the two ADCs are being reconfigured at the same time, ADC0CON must be written last to ensure that the voltage/temperature channel restarts converting at the same time as the current channel. However, if the voltage/temperature ADCs are powered down, there is a power-up delay before the voltage/temperature channel can start converting, which can mean that the IADC has started converting, and the voltage/temperature channel cannot start its conversion until the IADC has provided an output.

ADC Control Register: ADC1CON (Table 87)

A write to the ADC1CON register resets the VADC/TADC. The VADC/TADC restarts at an appropriate time so that its outputs are synchronous with the IADC, that is, the IADC is not reset by a change in the VADC/TADC configuration. The full VADC/TADC digital filter settling time must elapse before a voltage or temperature result is available.

ADC Filter Register (Table 88)

The ADCFLT register controls the output speed of the ADC, which influences the noise of the ADC. A write to ADCFLT resets the corresponding ADC.

ADC Configuration Registers (Table 89)

A write to the ADCCFG register does not reset the ADC.

If the overrange or comparator interrupts are set, the interrupt can be disabled by one of the following methods:

- Clearing the interrupt mask in ADCSTA
- Disabling comparator
- Reconfiguring the ADC

Note that the comparator must be disabled for at least one full conversion period to ensure that the comparator interrupt is reset. Also, the interrupt mask in ADCMSKI (ADCMSKI[4]) must be cleared to prevent unwanted interrupts during this period.

The result counter does not produce an interrupt of its own, but only gates the ADCxRDY interrupts.

If the result counter and the always interrupt mode are both active, the ADC operation is undefined.

ADC Data Register (Table 90 and Table 91)

Figure 21 shows the formatting of the 32-bit ADC0DAT register according to the gain setting selected. The ADC1DAT register does not have this shifting of data as there is no gain control on the voltage/temperature channel.

Only the relevant data bits in the ADC0DAT register are set depending on the gain setting. This means that no software adjustment to the ADC0DAT result is required for different gain settings. The result is automatically adjusted internally based on the gain setting.

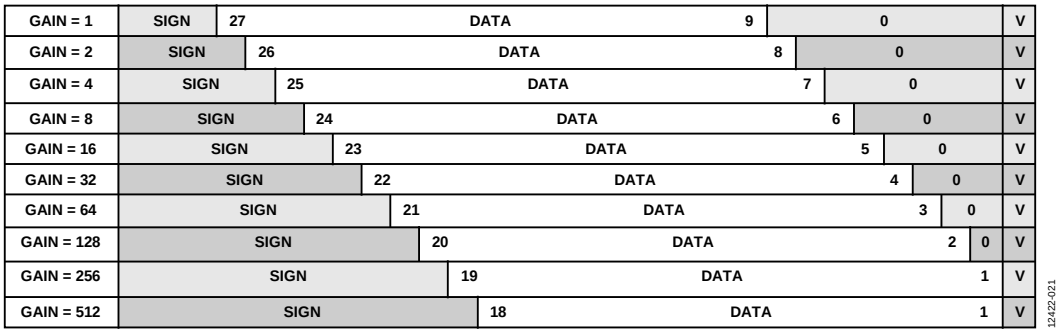

Figure 21. ADC Output (Gain = 1 to Gain = 512)

The data format of ADCxDAT is twos complement or unipolar, based on the ADCxCODE bits in the ADCxCON registers.

The ADC0DAT register output equals a useful signal range of bits from 18 to 3 at a gain of 512 and noise from Bit 2 to Bit 1.

ADCxDAT[0] indicates if valid data is available. If ADCxDAT[0] = 1, the data is invalid. If ADCxDAT[0] = 0, the data is valid.

For improved accuracy, adjust the gain to match the expected signal with the correct input range. This can be seen in Figure 21. For example, when measuring a small signal at a gain of 1, mostly noise is being measured. At a gain of 512, there are a number of bits with more useful data to give an accurate result. Note that for faster update rates, the noise increases.

Reading the ADC0DAT register clears any active RDY flags.

The ADC cannot write new data into the ADCxDAT register if the relevant ADCxRDY bits are set. This does not apply if the core is off; when the core is off, the ADCxDAT register contains the most recent ADC data.

The data registers are written simultaneously if all ADCs are active, and the ADCxRDY bits are set when the ADCxDAT MMRs are written.

If the configuration of the ADC1 is changed in the middle of a conversion, a number of ADC results occur where data is only written into the ADC0DAT data.

If the ADCxRDY bit is low, there is no guarantee that the ADCxDAT MMRs are stable if read.

If the ADC0 result counter is active, the data register (ADC0DAT) is not updated until the ADC0RDY bit is set, that is, when the ADC0RCV counter reaches the programmed limit (ADC0RCL). This condition does not hold if the core is off, in which case the ADC0DAT MMR(s) contain the most recent ADC result(s) when the core wakes up.

In calibration mode, the result of the calibration is stored in the ADCxDAT register. The user must transfer the result to the ADCxGN or ADCxOF register.

ADC Calibration Registers

In gain calibration mode, the result of the calibration is stored in ADCxDAT[15:0]. The user must transfer the result to ADCxGN[15:0].

In offset calibration mode, the result of the calibration is stored in ADCxDAT[24:0]. The user must transfer the result to ADCxOF[23:0].

ADC Offset Registers (Table 92, Table 93, and Table 94)

The offset registers are in signed twos complement form. A code of 0000 means subtract 0 offset. This form does not change to unipolar if the ADCxCODE bit in the ADCxCON register is set.

ADC0OF

If performing a manual offset calibration with an external shorted input, the ADC0DAT[31:0] result can be transferred to the ADC0OF register using the following formula:

$$ADC0OF = (ADC0DAT \div 2^5) \times PGAGN \times (0x4000 \div ADC0GN)$$

ADC1OF

If performing a manual offset calibration, the ADC1DAT[31:0] result can be transferred to the ADC1OF register using the following formula:

$$ADC1OF = (ADC1DAT \div 2^5) \times (0x4000 \div ADC1GN)$$

ADC2OF

If performing a manual offset calibration, the ADC1DAT[31:0] result can be transferred to the ADC2OF register using the following formula:

$$ADC2OF = (ADC1DAT \div 2^5) \times (0x4000 \div ADC2GN)$$

ADC Gain Registers (Table 95, Table 96, and Table 97)

The gain calibration registers are written with factory calibration values at power-on reset.

The gain register is an unsigned number, representing a scaling factor. The maximum value is 0xFFFF. The nominal value is 0x5555.

The temperature channel gain can be calibrated for an internal sensor or an external sensor, not both.

The ADC0 calibration results stored in ADC0DAT[15:0] must be transferred to the ADC0GN register. The ADC1 calibration results stored in ADC1DAT[15:0] must be transferred to the ADC1GN register. The ADC1 calibration results stored in ADC1DAT[15:0] must be transferred to the ADC2GN register.

ADC Current Channel Accumulator: ADC0ACC (Table 103)

The ADC0ACC register is read only and returns the current value of the accumulator. The ADC0ACC register updates one or two ADC clocks earlier than the ADC0DAT register.

There is no warning if the accumulator overflows. The ADC0RCL register can be used to reset the ADC0ACC register after a suitable number of samples. The number of samples is dependent on the gain range selected. For example, at gain = 32, 256 samples may be accumulated.

The accumulator is a signed twos complement or unipolar register, depending on the setting of the ADC0CODE bit.

The accumulator is reset by disabling it in the ADCCFG register (ADCCFG[5]), or by reconfiguring the ADC. If using the ADCCFG register, the enable must be low for at least one full conversion period to ensure that the accumulator is reset. The interrupt mask in the ADCMSKI register (ADCMSKI[5]) must also be cleared to prevent unwanted interrupts.

ADC MEMORY MAPPED REGISTERS

Table 81. ADC Memory Mapped Registers (Base Address 0x40030000)

Offset	Name	Description	Access	Default
0x0000	ADCSTA	ADC status register	R	0x0000
0x0004	ADCMSKI	ADC interrupt mask register	RW	0x00
0x0008	ADCMDE	ADC mode control register	RW	0x0003
0x000C	ADC0CON	Current ADC control register	RW	0x0000 0000
0x0010	ADC1CON	Voltage/temperature ADC control register	RW	0x0000 0000
0x0018	ADCFLT	ADC filter configuration register	RW	0x0000 0007
0x001C	ADCCFG	ADC configuration register	RW	0x00
0x0020	ADC0DAT	Current ADC result register	R	0x0000 0000
0x0024	ADC1DAT	Voltage/temperature ADC result register	R	0x0000 0000
0x0030	ADC0OF	Current ADC offset calibration register	RW	Calibration value
0x0034	ADC1OF	Voltage ADC offset calibration register	RW	Calibration value
0x0038	ADC2OF	Temperature ADC offset calibration register	RW	Calibration value
0x003C	ADC0GN	Current ADC gain calibration register	RW	Calibration value
0x0040	ADC1GN	Voltage ADC gain calibration register	RW	Calibration value
0x0044	ADC2GN	Temperature ADC gain calibration register	RW	Calibration value
0x0048	ADC0RCL	ADC result counter limit register	RW	0x0001
0x004C	ADC0RCV	ADC result counter value register	R	0x0000
0x0050	ADC0TH	Current ADC comparator threshold register	RW	0x0000 0000
0x0054	ADC0THC	ADC threshold counter limit register	RW	0x01
0x0058	ADC0THV	ADC threshold counter value register	R	0x00
0x005C	ADC0ACC	Current ADC accumulator register	R	0x0000 0000
0x0060	ADC0ATH	Current ADC accumulator threshold register	RW	0x0000 0000

Reference Control Registers

Table 82. ADC Reference Control Memory Mapped Registers (Base Address 0x40008800)

Offset	Name	Description	Access	Default
0x0008	HRFCTRL	Internal 1.2 V reference control register	RW	0x0002
0x0024	IRFPD	Internal reference buffer power down register	RW	0x0001

The ADC and ADC MMRs work on different time domains; for synchronization, a delay is required before valid data is available in the ADC time domain. In normal mode, the delay required is 8 µs. In low power mode, the delay required is 31 µs. To avoid this delay, the registers can be polled for valid data.

ADC Status Register

Address: 0x40030000, Reset: 0x0000, Name: ADCSTA

Table 83. ADCSTA Register Bit Descriptions

Bits	Name	Description
15	ADCxCAL	ADC calibration status 0: cleared to 0 after any of the ADCMDE, ADCFLT, or ADC0CON registers are written to. 1: set to 1 in hardware to indicate an ADC calibration cycle is complete.
14	RESERVED	Reserved.
13	ADC1ERR	ADC conversion error status bit. Error in VADC/TADC conversion. 0: cleared after reading the ADCSTA register. 1: set to interrupt when an underrange or an overrange error occurs in ADC1. This bit is set to 1 when the VADC input voltage exceeds 28.8 V. However, note that for voltages greater than 29.5 V, this bit is cleared again.
12	ADC0ERR	ADC conversion error status bit. Error in current ADC conversion. 0: clear after reading the ADCSTA register. 1: set to interrupt when an underrange or an overrange error occurs in ADC0.
[11:7]	RESERVED	Reserved.

Bits	Name	Description
6	ADCINT	ADC interrupt. 0: cleared to 0 by reading ADC0DAT. 1: if the always interrupt mode is active, this bit generates an interrupt every $1/f_{ADC}$, regardless of whether the digital filter is settled. The ADCxRDY flags must be interrogated to determine if the ADC result is valid. For example, at the default ADCFLT value of 0x0007, this bit is asserted every 1.0 ms. The first two interrupts after reconfiguring the ADC cannot have the ADCxRDY bits set, but all subsequent ones can be set. This does not guarantee a periodic time base, however, because the ADC time base is interrupted any time the current ADC is reconfigured.
5	ADC0ATHEX	ADC current accumulator comparator threshold exceeded. 0: cleared by a reconfiguration of the ADC, or the accumulator comparator is disabled. 1: set if the absolute value of the accumulator exceeds the value written in the ADC0ATH register.
4	ADC0THEX	ADC current comparator threshold exceeded. Valid only if the ADC current channel comparator is enabled. 0: cleared by a reconfiguration of the ADC, or if the ADC0TH comparator is disabled. 1: set if the absolute value of the ADC conversion result exceeds the value written in the ADC0TH register. If ADC0THC (ADC threshold counter) is also used, this bit is only set when the specified number of ADC conversions exceed the threshold.
3	ADC0OVR	ADC current channel fast overrange bit. 0: change the gain setting to clear this bit. 1: set if the IADC input is grossly (>30% approximate on some PGA settings) overrange, and the overrange detect function is enabled. This bit is updated every 125 µs.
2	ADC2RDY	Temperature conversion result ready bit. 0: cleared to 0 by reading the ADC1DAT and ADC0DAT registers (the registers must be read in this order). Reconfiguring the VADC/TADC or IADC also clears this bit. 1: set to 1 at the end of a conversion if ADC1 is enabled and a temperature channel is selected. This generates an interrupt if the ADC interrupt is enabled and the corresponding bit in the ADCMSKI register is set. If this bit is set, the ADC cannot write further data to the ADC1DAT register. If the always interrupt mode is selected, this bit is a valid flag that indicates when the ADC1 digital filter has fully settled.
1	ADC1RDY	Voltage conversion result ready bit. 0: cleared to 0 by reading the ADC1DAT and ADC0DAT registers (the registers must be read in this order). Reconfiguring the VADC/TADC or IADC also clears this bit. 1: set to 1 at the end of a conversion if ADC1 is enabled and the voltage channel is selected. This generates an interrupt if the ADC interrupt is enabled and the corresponding bit in the ADCMSKI register is set. If this bit is set, the ADC cannot write further data to the ADC1DAT register. If the always interrupt mode is selected, this bit is a valid flag that indicates when the ADC1 digital filter has fully settled.
0	ADC0RDY	Current conversion result ready bit. 0: cleared to 0 by reading the ADC0DAT register. Reconfiguring the IADC also clears this bit. 1: set to 1 at the end of a conversion if ADC0 is enabled. This generates an interrupt if the ADC interrupt is enabled and the corresponding bit in the ADCMSKI register is set. Reading the ADC0DAT register clears this bit. Reconfiguring the IADC also clears this bit. If this bit is set, the ADC cannot write further data to the ADC0DAT register. If the ADC result counter is active, RDY is only asserted after the required number of conversions has elapsed. If the always interrupt mode is selected this bit is a valid flag that indicates when the ADC digital filter has fully settled.

ADC Interrupt Mask Register

Address: 0x40030004, Reset: 0x00, Name: ADCMSKI

Table 84. ADCMSKI Register Bit Descriptions

Bits	Name	Description
7	RESERVED	Reserved.
6	ADCINT	Mask ADCSTA[6]. 0: disable the interrupt (default). 1: enable an interrupt when the ADCINT bit in the ADCSTA register is set.
5	ADCOATHEX	Mask ADCSTA[5]. 0: disable the interrupt (default). 1: enable an interrupt when the ADC current accumulator comparator threshold is exceeded. (The ADC0ATHEX bit in the ADCSTA register is set.)
4	ADC0THEX	Mask ADCSTA[4]. 0: disable the interrupt (default). 1: enable an interrupt when the ADC current comparator threshold is exceeded. (The ADC0THEX bit in the ADCSTA register is set.)
3	ADC0OVR	Mask ADCSTA[3]. 0: disable the interrupt (default). 1: enable an interrupt when the ADC0OVR bit in the ADCSTA register is set.
2	ADC2RDY	Mask ADCSTA[2]. 0: disable the interrupt (default). 1: enable an interrupt when the ADC2RDY bit in the ADCSTA register is set.
1	ADC1RDY	Mask ADCSTA[1]. 0: disable the interrupt (default). 1: enable an interrupt when the ADC1RDY bit in the ADCSTA register is set.
0	ADC0RDY	Mask ADCSTA[0]. 0: disable the interrupt (default). 1: enable an interrupt when the ADC0RDY bit in the ADCSTA register is set.

ADC Mode Control Register

Address: 0x40030008, Reset: 0x0003, Name: ADCMDE

Table 85. ADCMDE Register Bit Description

Bits	Name	Description	
[7:6]	RESERVED	Reserved.	
5	AINT	Always interrupt. 0: disable this function. 1: enables the ADC to always produce an interrupt 1/f_{ADC} after starting a new conversion, even if the digital filter is not settled. The ADCxRDY flags must be interrogated to determine the validity of the ADC data registers. Bit 6 of the ADCSTA and ADCMSKI registers must be used in conjunction with this bit.	
4	RESERVED	Reserved.	
3	ADCLP	ADC power mode. 0: enable ADC normal power mode (f_{MOD} = 512 kHz). 1: enable ADC low power mode (f_{MOD} = 128 kHz).	
[2:0]	ADCMD	ADC mode bits.	
		ADCMDE[2:0]	**Description**
		000	ADC power-down mode. The ADC circuitry is powered off. This powers down the ADC and PGA.
		001	Continuous convert mode. The enabled ADC(s) continuously produce conversions at f_{ADC}. The ADCxRDY bits must be cleared to enable new data to be written into the ADCxDAT registers.
		010	Single convert mode. This performs a single shot conversion on the enabled ADC(s). The ADC enters idle mode after the RDY bit is set.
		011	Idle mode. The ADC is powered up but held in reset. The ADC enters idle mode after calibration.
		100	Self offset calibration. An offset calibration is performed with an internally generated 0 V. The input of the ADC is a short to the negative pin of the selected channel. This can be used when chopping is off to calibrate out the ADC offset. The result is written to the ADCxDAT register of each enabled ADC. After calibration, the ADC returns to idle mode, with the ADCxRDY and ADCxCAL bits in the ADCSTA register set.
		101	Self gain calibration. A gain calibration is performed with an internal +FS voltage. On the IADC, this is only allowed at a gain of 1, and the user must first change to a gain of 1. On the VADC, a self gain calibration does not remove attenuator error. On the TADC, a self gain calibration does not remove sensor errors, and results in an uncalibrated internal temperature sensor. In these cases, use the factory calibration or system gain calibration.
		110	System zero-scale calibration (offset). ADC offset calibration on the selected channel; the channel must be shorted externally. The result is written to the ADCxDAT register of each enabled ADC.
		111	System full-scale calibration (system). A gain calibration is performed with an externally applied full-scale input voltage. The result is written to the ADCxDAT register of each enabled ADC. This must not be done if using the internal temperature sensor, because it is not possible to produce a full-scale input.

Note that the results of the calibrations are written into the ADCxDAT registers. The user must copy the calibration results from the ADCxDAT registers into the ADCxOF and ADCxGN registers to perform plausibility checks.

Current ADC Control Register

Address: 0x4003000C, Reset: 0x0000 0000, Name: ADC0CON

Set ADC0CON = 0x030000 when the ADC is not enabled, and set ADC0CON = 0x08XXXX when the ADC is enabled.

Table 86. ADC0CON Register Bit Descriptions

Bits	Name	Description		
19	ADC0EN	Enable current ADC. 0: power down the ADC, RDY bit is cleared. 1: enable the current ADC.		
18	RESERVED	Reserved.		
[17:16]	ADC0BUFBP	ADC0 buffer bypass (positive, negative).		
		ADC0CON[17:16]	**Description**	
		00	No buffer is bypassed.	
		01	Negative buffer is bypassed.	
		10	Positive buffer is bypassed.	
		11	Both buffers are bypassed.	
[15:14]	ADC0DIAG2	ADC0DIAG2[1:0]: diagnostic current bits.		
		ADC0CON[15:14]	**Description**	
		00	Current source off.	
		01	Enable 50 µA on IIN−_AUX.	
		10	Enable 50 µA on IIN+_AUX.	
		11	Enable 50 µA on IIN+_AUX and IIN−_AUX.	
[13:12]	ADC0DIAG1	ADC0DIAG1[1:0]: diagnostic current bits.		
		ADC0CON[13:12]	**Description**	
		00	Current source off.	
		01	Enable 50 µA on IIN−.	
		10	Enable 50 µA on IIN+.	
		11	Enable 50 µA on IIN+ and IIN−.	
[11:10]	PGASCALE	PGASCALE[1:0]: PGA scaling bits. If enabled, PGA scaling scales the PGA output by the selected scale factor. To obtain the correct ADC result, user software must compensate for the scaling factor selected. For example, if 14/16 PGA scale is selected, user software must compensate by multiplying the ADC0DAT result by 16/14. See the ADC formulas in the Understanding the Offset and Gain Calibration Registers section for further details.		
		ADC0CON[11:10]	**Description**	
		00	PGA gain is scaled by ((14/16) × (13/16)).	
		01	PGA gain is scaled by 14/16.	
		10	PGA gain is scaled by 13/16.	
		11	PGA scaling is disabled. No scaling.	
9	ADC0CODE	Current ADC output coding. 0: twos complement (bipolar). 1: unipolar.		
[8:6]	ADC0CH	ADC0CH[1:0] current ADC input channel select.		
		ADC0CON[8:6]	**Input Channel Select**	**Description**
		000	IIN+, IIN−	
		001	IIN−, IIN−	Internal short configuration
		010	AVDD18/136±	Voltage input to diagnostic
		011	IIN+_AUX, IIN−_AUX	Auxiliary channel
		100 to 111	Reserved	
5	RESERVED	Reserved.		
4	ADC0REF	Current ADC reference selection. 0: internal reference (1.2 V) or external reference (see Figure 18 and corresponding notes). 1: AVDD18, AGND.		

Bits	Name	Description	
[3:0]	ADC0PGA	Current ADC gain select:	
		ADC0CON[3:0]	**Gain**
		0000	1 (only use for a self gain calibration)
		0010	4
		0011	8
		0100	16
		0101	32
		0110	64
		1001	512
		Others	Reserved

Voltage/Temperature ADC Control Register

Address: 0x40030010, Reset: 0x0000 0000, Name: ADC1CON

Set ADC1CON = 0x030000 when the ADC is not enabled and ADC1CON = 0x08XXXX when the ADC is enabled.

Table 87. ADC1CON Register Bit Descriptions

Bits	Name	Description		
19	ADC1EN	Enable voltage/temperature ADC. 0: power down the ADC; ADC1RDY and ADC2RDY bits are cleared. 1: enable the voltage/temperature ADC.		
18	RESERVED	Reserved.		
[17:16]	ADC1BUFBP	ADC1 buffer bypass (positive/negative).		
		ADC1CON[17:16]	**Description**	
		00	No buffer is bypassed (recommended setting with the ADC is enabled).	
		01	Negative buffer is bypassed.	
		10	Positive buffer is bypassed.	
		11	Both buffers are bypassed (recommended setting with the ADC is disabled).	
[15:14]	RESERVED	Reserved.		
[13:12]	ADC1DIAG	ADC1DIAG[1:0]: diagnostic current bits		
		ADC1CON[13:12]	**Description**	
		00	Current source off.	
		01	Enable 50 µA to negative input.	
		10	Enable 50 µA to positive input.	
		11	Enable 50 µA to two differential inputs.	
[11:10]	RESERVED	Reserved.		
9	ADC1CODE	Voltage/temperature ADC output coding. 0: twos complement (bipolar). 1: unipolar.		
[8:6]	ADC1CH	Voltage/temperature ADC input channel select.		
		ADC1CON[8:6]	**Input Channel Select**	**Description**
		000	VBAT, AGND	VBAT attenuator
		001	VTEMP, GND_SW	External temperature sensor
		010	Vbe1, Vbe2	Internal temperature sensor
		011	Reserved	
		100	VINP_AUX, VINM_AUX	Auxiliary voltage input
		101	Reserved	
		110	Vbe, GND	V_{IN} for ADC diagnostic
		111	Reserved	
5	RESERVED	Reserved.		
4	ADC1REF	Voltage/temperature ADC reference selection 0: internal reference (1.2 V) 1: AVDD18, GND_SW		
[3:0]	RESERVED	Reserved.		

ADC Filter Configuration Register

Address: 0x40030018, Reset: 0x0000 0007, Name: ADCFLT

Table 88. ADCFLT Register Bit Descriptions

Bits	Name	Description
16	SINC4_EN	Sinc4 filter enable. 0: sinc filter is in sinc3 mode. 1: sinc filter is in sinc4 mode. Note that when the output rate is <2 kHz, this bit must not be set. For an output rate ≥2 kHz, setting this bit makes the filter a sinc4 filter. This bit is recommended when the output rate is greater than 2 kHz, because in these cases the sinc3 is insufficient to filter out the quantization noise.
15	CHOP	Enables system chopping of the ADCs. 0: disable system chopping. 1: enable system chopping. Enable chopping to give very low offset errors and drift. The settling time to a change in configuration equals one conversion period. This operates on all ADCs.
14	RAVG2	Enables a running average-by-2. 0: disable running average function. 1: enable running average function. RAVG2 implements a simple running average-by-2 function to reduce the ADC noise. It is automatically active when chopping is enabled and is an optional feature when chopping is inactive. It does not reduce the output rate with chop off, but does increase the settling time by one conversion.
[13:8]	AF	Averaging Factor[5:0], Number of Averages = AF + 1. AF implements a programmable, first order, sinc post-filter. This additional averaging factor (AF) further reduces the ADC output rate. There are restrictions on the maximum allowable values of SF and AF. If AF is less than or greater than 0, the full equation for the ADC output rate from the filter with CHOP = 0/1 is $f_{ADC} = f_{MOD}/[(SF+1) \times 64 \times (3 + AF) +3]$ Hz. The averaging factor can only be used at certain SF values.
7	NOTCH2	Inserts a notch at $f_{NOTCH2} = 1.2 \times f_{NOTCH}$. NOTCH2 modifies the frequency response of the sinc3/sinc4 filter to improve the stop-band rejection. The worst-case rejection at $f > f_{ADC}$ improves to −45 dB, compared to −40 dB with the default sinc3 filter, where f is the sampling frequency. There is a slight increase in ADC noise if this is active. The second notch is generally at (4 ÷ 3) × the main sinc notch, to generate notches at both 50 Hz and 60 Hz.
[6:0]	SF	Sinc3 filter decimation factor. Default value: 0x7. SF controls the oversampling rate/decimation factor of the sinc3/sinc4 filter. The output rate from the sinc3 filter is given by: $f_{ADC} = f_{MOD} \div [(SF+1) \times 64]$ Hz. (CHOP = 0 and AF = 0.) $f_{ADC} = f_{MOD} \div [(SF+1) \times 64 \times 3+3]$ Hz. (CHOP = 1 and AF = 0.) $f_{ADC} = f_{MOD} \div [(SF+1) \times 64 \times (AF+3) + 3]$ Hz. (CHOP = 0/1 and AF ≠ 0.) where f_{MOD} is 512 kHz in normal mode and 128 kHz in low power mode. The output rate from the sinc4 filter (AF is always 0 in sinc4) is given by: $f_{ADC} = f_{MOD} \div [(SF+1) \times 64]$ Hz. (CHOP = 0.) $f_{ADC} = f_{MOD} \div [(SF+1) \times 64 \times 4 + 3]$ Hz. (CHOP = 1.) The first notch in the frequency response of the sinc3/sinc4 filter occurs at $f_{NOTCH} = f_{ADC}$. The settling time of the filter to a change in ADC configuration (CHOP = 0) is $3 \div f_{ADC}$ for sinc3, or $4 \div f_{ADC}$ for sinc4.

ADC Configuration Register

Address: 0x4003001C, Reset: 0x00, Name: ADCCFG

Table 89. ADCCFG Register Bit Descriptions

Bits	Name	Description	
7	SWEN	Switch enable. 0: switch from GND_SW to AGND is open circuit. 1: switch from GND_SW to AGND is closed. This switch is not open circuit automatically when the ADC is placed in power-down mode.	
6	RESERVED	Reserved.	
5	ADC0ACCEN	ADC0 accumulator enable—current channel. 0: accumulator not active, resets accumulator to 0. 1: accumulator active. A negative reading decrements the accumulator. Note that the accumulator can overflow if allowed to run for more than 65,535 conversions.	
[4:3]	ADC0CMPEN	ADC0 comparator enable.	
		ADCCFG[4:3]	**Description**
		00	Comparator not active.
		01	Comparator active. Interrupt if |I| > ADC0TH. The comparator samples at f_{ADC}, that is, at the ADC output speed.
		10	Comparator + counter. Interrupt if |I|> ADC0TH for #ADC0THC conversions. A value |I| < Ith resets ADC0THCNT counter to 0, if the interrupt is low.
		11	Comparator + counter. Interrupt if |I|> ADC0TH for #ADC0THC conversions. A value |I| < Ith decrements the counter with a floor of 0, if the interrupt is low.
2	ADC0OREN	ADC overrange enable 0: disable this function. 1: enables a coarse overrange comparator on the current channel. If the current reading goes significantly above +FS or below −FS for the active PGA setting, the overrange flag (ADC0OVR) in ADCSTA[3] is set. The signal must be outside ±FS for 125 µs for this to occur. The ADC conversion is not interrupted if an overrange occurs.	
1	FASTEN	Fast conversion on internal temperature sensor. 0: disable fast conversion mode on the internal temperature sensor. 1: enable fast conversion mode on the internal temperature sensor. The VADC/TADC does not switch immediately to the temperature channel; therefore, there is one more conversion on the channel programmed in ADC1CON, then the ADC switches at the optimum time to the internal temperature sensor to minimize latency. Only a single fast conversion is performed, that is, the ADC only reacts to a 0 to 1 transition on this bit and when complete reverts to the ADC1CON channel. This bit must be cleared when the fast conversion result is read, in preparation for the next fast request. The fast conversion occurs at a fixed conversion speed (1 ms), and must only be used with certain combinations of SF and AF so that the fast result occurs simultaneous with the current channel result. This bit cannot be set before the ADC conversions start, that is, it must be set after at least one ADC interrupt (this can be an unsettled interrupt if this mode is used).	
0	ADCRCEN	ADC result counter enable. 0: counter off/reset. ADC interrupts every 1 ÷ f_{ADC}. 1: counter on. ADC interrupts if ADC0RCV = ADC0RCL. Intermediate ADC conversions are lost if the result counter is active unless the accumulator is active.	

Current ADC Result Register

Address: 0x40030020, Reset: 0x0000 0000, Name: ADC0DAT

Table 90. ADC0DAT Register Bit Descriptions

Bits	Name	Description
[31:1]	RESULT	Current ADC conversion result.
0	VALID	0: data is valid. 1: data is invalid (overrange or not ready).

Voltage/Temperature ADC Result Register

Address: 0x40030024, Reset: 0x0000 0000, Name: ADC1DAT

Table 91. ADC1DAT Register Bit Descriptions

Bits	Name	Description
[31:1]	RESULT	Voltage/temperature ADC conversion result.
0	VALID	0: data is valid. 1: data is invalid (overrange or not ready).

Current ADC Offset Calibration Register

Address: 0x40030030, Reset: Calibration value, Name: ADC0OF

Table 92. ADC0OF Register Bit Descriptions

Bits	Name	Description
[23:0]	ADC0OF	Current ADC 24-bit offset calibration coefficient for all IADC input channels. Note that Bit 23 is a sign bit. In offset calibration mode, the result of the calibration is stored in ADC0DAT[23:0]. The user must transfer the result to ADC0OF[23:0].

Voltage ADC Offset Calibration Register

Address: 0x40030034, Reset: Calibration value, Name: ADC1OF

Table 93. ADC1OF Register Bit Descriptions

Bits	Name	Description
[23:0]	ADC1OF	Voltage ADC 24-bit offset calibration coefficient for the VBAT/AGND and VINP_AUX/VINM_AUX channels. Note that Bit 23 is a sign bit. In offset calibration mode, the result of the calibration is stored in ADC1DAT[23:0]. The user must transfer the result to ADC1OF[23:0].

Temperature ADC Offset Calibration Register

Address: 0x40030038, Reset: Calibration value, Name: ADC2OF

Table 94. ADC2OF Register Bit Descriptions

Bits	Name	Description
[23:0]	ADC2OF	Temperature ADC 24-bit offset calibration coefficient for the VTEMP/GND_SW, Vbe1/Vbe2, and Vbe/GND channels. Note that Bit 23 is a sign bit. In offset calibration mode, the result of the calibration is stored in ADC1DAT[23:0]. The user must transfer the result to ADC2OF[23:0].

Current ADC Gain Calibration Register

Address: 0x4003003C, Reset: Calibration value, Name: ADC0GN

Table 95. ADC0GN Register Bit Descriptions

Bits	Name	Description
[15:0]	ADC1GN	Current ADC 16-bit gain calibration coefficient for all IADC input channels. In gain calibration mode, the result of the calibration is stored in ADC0DAT[15:0]. The user must transfer the result to ADC0GN[15:0].

Voltage ADC Gain Calibration Register

Address: 0x40030040, Reset: Calibration value, Name: ADC1GN

Table 96. ADC1 GN Register Bit Descriptions

Bits	Name	Description
[15:0]	ADC1GN	Voltage ADC 16-bit gain calibration coefficient for the VBAT/AGND and VINP_AUX/VINM_AUX channels. In gain calibration mode, the result of the calibration is stored in ADC1DAT[15:0]. The user must transfer the result to ADC1GN[15:0].

Temperature ADC Gain Calibration Register

Address: 0x40030044, Reset: Calibration value, Name: ADC2GN

Table 97. ADC2 GN Register Bit Descriptions

Bits	Name	Description
[15:0]	ADC2GN	Temperature ADC 16-bit gain calibration coefficient for the VTEMP/GND_SW, Vbe1/Vbe2, and Vbe/GND channels. In gain calibration mode, the result of the calibration is stored in ADC1DAT[15:0]. The user must transfer the result to ADC2GN[15:0].

ADC Result Counter Limit Register

Address: 0x40030048, Reset: 0x0001, Name: ADC0RCL

Table 98. ADC0RCL Register Bit Descriptions

Bits	Name	Description
[15:0]	ADC0RCL	ADC result counter limit register. This register sets the number of conversions required before an ADC interrupt is generated. This feature must be enabled via the ADCCFG MMR (ADCCFG[0]). The current channel comparator and the current channel accumulator are always active at f_{ADC}.

ADC Result Counter Value Register

Address: 0x4003004C, Reset: 0x0000, Name: ADC0RCV

Table 99. ADCRCV Register Bit Descriptions

Bits	Name	Description
[15:0]	ADCRCV	ADC result counter value register. This register holds the current number of completed conversion. This value can be used in conjunction with the accumulator to calculate the average current measured.

Current ADC Comparator Threshold Register

Address: 0x40030050, Reset: 0x0000 0000, Name: ADC0TH

Table 100. ADC0TH Register Bit Descriptions

Bits	Name	Description
[31:0]	ADC0TH	ADC 32-bit current ADC comparator threshold register. The absolute value is used by the comparator.

ADC Threshold Counter Limit Register

Address: 0x40030054, Reset: 0x01, Name: ADC0THC

Table 101. ADC0THC Register Bit Descriptions

Bits	Name	Description
[7:0]	ADC0THC	ADC threshold counter limit. This register determines how many ADC readings above ADC0TH must occur before the ADC0THEX bit is set (ADCSTA[4]), causing an interrupt.

ADC Threshold Counter Register

Address: 0x40030058, Reset: 0x00, Name: ADC0THV

Table 102. ADC0THV Register Bit Descriptions

Bits	Name	Description
[7:0]	ADC0THV	ADC threshold counter. This register contains the current value of the number of times the threshold (ADC0TH) has been exceeded.

Current ADC Accumulator Register

Address: 0x4003005C, Reset: 0x0000 0000, Name: ADC0ACC

Table 103. ADC0ACC Register Bit Descriptions

Bits	Name	Description
[31:0]	ADC0ACC	ADC accumulator. This register holds the accumulated conversion result in the same format as ADC0DAT.

Current ADC Accumulator Threshold Register

Address: 0x40030060, Reset: 0x0000 0000, Name: ADC0ATH

Table 104. ADC0ATH Register Bit Descriptions

Bits	Name	Description
[31:0]	ADC0ATH	Current ADC accumulator threshold. The result is in the same format as ADC0DAT.

Internal 1.2 V Reference Control

Address: 0x40008808, Reset: 0x0002, Name: HRFCTRL

Table 105. HRFCTRL Register Bit Descriptions

Bits	Name	Description
[15:2]	RESERVED	Reserved.
1	HRFAUTOB	0: automatic mode. The 1.2 V internal reference is controlled by the ADC. HRFPD is ignored. 1: user mode. The 1.2 V internal reference is controlled by HRFPD. This is the default state.
0	HRFPD	0: enable the 1.2 V internal reference. 1: power down the 1.2 V internal reference.

Internal Reference Buffer Power-Down Register

Address: 0x40008824, Reset: 0x0001, Name: IRFPD

Table 106. IRFPD Register Bit Descriptions

Bits	Name	Description
[15:1]	RESERVED	Reserved.
0	INTREFPD	0: enable internal reference buffer. 1: power down internal reference buffer. Gates in an external 1.2 V reference. This is the default state.

TIMERS

The ADuCM330/ADuCM331 have three timers:

- Timer 0, general-purpose timer
- Timer 2, wake-up timer
- Timer 3, watchdog timer

Timer 0 is a general-purpose, 32-bit, count up/count down timer with a programmable prescaler. Timer 0 is clocked directly by the PCLK.

Timer 2 is a 32-bit, count up/count down timer with a programmable prescaler. Timer 2 is clocked directly by the internal 32 kHz oscillator (LFOSC), UCLK, PCLK, or an external clock on P0.4 (ECLKIN).

Timer 3 is a 16-bit, count down timer with a programmable prescaler. Timer 3 is clocked by the internal 32.768 kHz oscillator (LFOSC). The watchdog timer (Timer 3) is used to recover from an illegal software state.

TIMER SYNCHRONIZATION

The synchronization block diagram (Figure 22) shows the interface between user timer MMRs and the core timer blocks. User code can access all timer MMRs directly, including TxVAL, TxCON and TxCLRI. Data must then transfer from these MMRs to the core timers (Timer 0, Timer 2, and Timer 3) within the timer subsystem. Theses core timers are buffered from the user MMR interface by the synchronization (SYNC) block. The principle of the synchronization block is to provide a method that ensures data and other required control signals can cross asynchronous clock domains correctly. An example of asynchronous clock domains is the Cortex-M3 processor running on a 16 MHz clock, and the wake-up timer (Timer 2) running on the low frequency oscillator of 32 kHz (LFOSC).

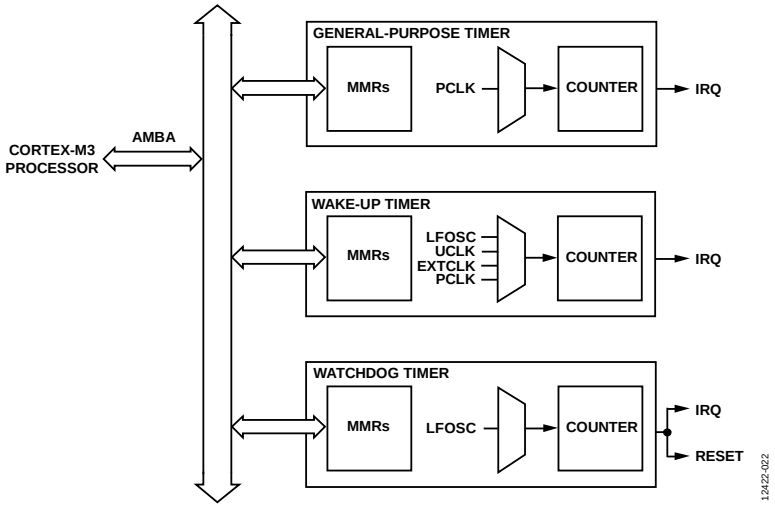

Figure 22. Timer Synchronization Block Diagram

As shown in Figure 22, the MMR logic and core timer logic reside in separate and asynchronous clock domains. Any data coming from the MMR core clock domain and being passed to the internal timer domain must be synchronized to the internal timer clock domain to ensure that it is latched correctly into the core timer clock domain. This is achieved by using two flip-flops, as shown in Figure 23, to not only synchronize but also to double buffer the data and therefore ensure data integrity in the timer clock domain.

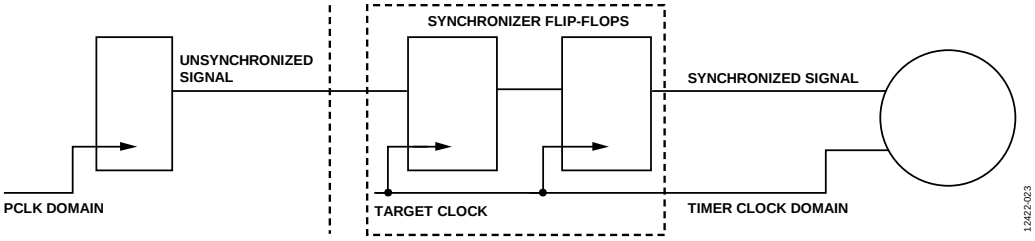

Figure 23. Synchronizer for Signals Crossing Clock Domains

Because of the synchronization block, while timer control data is latched almost immediately (with the fast, core clock) in the MMR clock domain, this data in turn cannot reach the core timer logic for at least two periods of the selected internal timer domain clock. It is recommended to wait for three periods of the timer domain clock for critical operations such as putting the core to sleep.

GENERAL-PURPOSE TIMER

GENERAL-PURPOSE TIMER (TIMER 0) FEATURES

The ADuCM330/ADuCM331 feature one general-purpose timer (Timer 0), a 32-bit, count up/count down timer with a programmable prescaler.

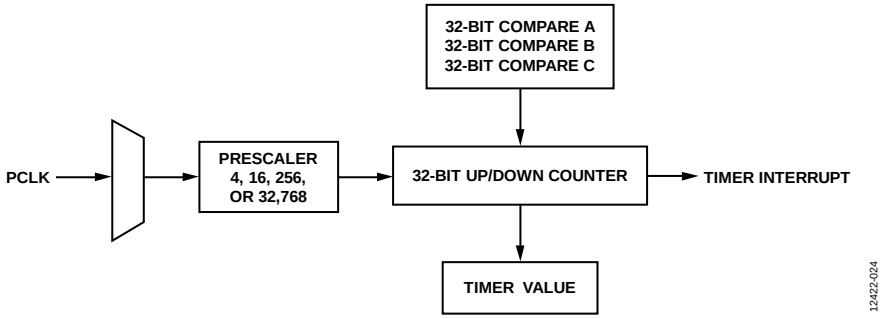

Figure 24. General-Purpose Timers Block Diagram

GENERAL-PURPOSE TIMER OVERVIEW

Timer 0 is a general-purpose, 32-bit, count up/count down timer with a programmable prescaler. The prescaler source is selectable and can be scaled by a factor of 4, 16, 256, or 32,768. Timer 0 is clocked by PCLK.

The timer can be free running or periodic. In free running mode, the counter decrements/increments from the maximum/minimum value until zero/full scale and starts again at the maximum/minimum value. In periodic mode, the counter decrements/increments until the counter matches the programmed timeout time in the Timer 0 Timeout Field C, stored in the T0TOFC0 and T0TOFC1 registers.

Timer 0 has three time field values that can be compared with the counter. The value of a counter can be read at any time by accessing its value registers (T0VALx).

GENERAL-PURPOSE TIMER OPERATION

The general-purpose timer is started by writing 1 to the enable bit (T0EN[0]).

In incrementing mode, the timer increments until the value reaches full scale (T0CON[4], FREE_RUN = 1) or until the counter matches the programmed wake-up time in T0TOFC0/T0TOFC1 (T0CON[4], FREE_RUN = 0). When this occurs, an IRQ is generated, and the timer is reset to 0.

In decrement mode, the timer starts at full scale and counts down. The timer decrements until it reaches zero (FREE_RUN = 1) or until the counter matches the programmed wake-up time in T0TOFC0/T0TOFC1 (T0CON[4], FREE_RUN = 0). When this occurs, an IRQ is generated, and the timer is reset to full scale.

The following must also be noted:

- Any timer IRQ can be cleared by writing a 1 to the correct clear bit in the T0CLRI register.
- The timer can be stopped and cleared by resetting the enable bit (T0EN[0]).
- The maximum timer value is 0xFFFF FFFF.
- The timer can be read at any time.
- The control register selects the mode and selects the prescaler value.
- There is a separate enable register to enable the timer.

Because Timer 0 is 32 bits but resides on a 16-bit bus, 2 bus reads are required to obtain the 32 bits. There are separate addresses for the upper (T0VAL1) and lower (T0VAL0) 16 bits of Timer 0. When the lower 16 bits are addressed and read, the upper 16 bits are latched and held in a separate register to be read later. The entire T0VALx registers (upper and lower) remain frozen until the upper 16 bits are read. The freeze control bit (T0CON[3]) must be set to freeze the T0VALx registers between lower and upper reads.

GENERAL-PURPOSE TIMER CONFIGURATION

Take care when configuring the general-purpose timer registers to ensure correct operation of the timer.

The configuration registers are:

- T0CON can only be written to when the timer is disabled.
- T0EN can be written to at any time.
- T0TOFA0, T0TOFA1, T0TOFB0, and T0TOFB1 can only be written to after the corresponding interrupt is disabled.
- For periodic mode, T0TOFC0 and T0TOFC1 can only be written to when the timer is disabled. For free running mode, these registers can be updated after the corresponding interrupt is disabled.
- T0IEN can be written to at any time.

On initial startup, all configuration registers must be configured before enabling the timer.

To reconfigure the timer (T0CON register or T0TOFC0/T0TOFC1 registers in periodic mode), use the following instructions:

1. T0EN = 0x0
2. Wait until (T0ISTA[8] = 0) // Wait until the enable is synchronized to the timer clock domain.
3. T0CON = {new value}
4. T0EN = 0x1

To configure the wake-up field registers while the timer is enabled as follows, the following example uses T0TOFA0.

1. T0IEN = T0ISTA & 0xE // Clear corresponding interrupt enable.
2. T0TOFA0 = {new value}
3. T0IEN = T0ISTA | 0x1 // (or 0x1) Enable corresponding interrupt.

This procedure can only be used for T0TOFC registers in free running mode.

Care must be taken when disabling the interrupts. If an interrupt is about to occur when it is disabled, the interrupt may be captured by the logic; in that case, it asserts the interrupt line after the interrupt is reenabled.

GENERAL-PURPOSE TIMER 0 MEMORY MAPPED REGISTERS

Table 107. General-Purpose Timer 0 Memory Mapped Registers (Base Address 0x40000000)

Offset	Name	Description	Access	Default
0x0000	T0VAL0	Current count value (LSB) register	R	0x0000
0x0004	T0VAL1	Current count value (MSB) register	R	0x0000
0x0008	T0CON	Control register	RW	0x0040
0x000C	T0EN	Timer enable register	RW	0x0000
0x0010	T0TOFA0	Timeout Field A (LSB) register	RW	0x1FFF
0x0014	T0TOFA1	Timeout Field A (MSB) register	RW	0x0000
0x0018	T0TOFB0	Timeout Field B (LSB) register	RW	0x2FFF
0x001C	T0TOFB1	Timeout Field B (MSB) register	RW	0x0000
0x0020	T0TOFC0	Timeout Field C (LSB) register	RW	0x3FFF
0x0024	T0TOFC1	Timeout Field C (MSB) register	RW	0x0000
0x0028	T0IEN	Interrupt enable register	RW	0x0000
0x002C	T0ISTA	Interrupt status register	R	0x0000
0x0030	T0CLRI	Clear interrupts register	W	Not applicable

General-Purpose Timer Current Count Value (LSB) Register

Address: 0x40000000, Reset: 0x0000, Name: T0VAL0

Table 108. T0VAL0 Register Bit Descriptions

Bits	Name	Description
[15:0]	VALUE	Current Timer 0 count value, Bit 15 to Bit 0 (LSB)

General-Purpose Timer Current Count Value (MSB) Register

Address: 0x40000004, Reset: 0x0000, Name: T0VAL1

Table 109. T0VAL1 Register Bit Descriptions

Bits	Name	Description
[15:0]	VALUE	Current Timer 0 count value, Bit 31 to Bit 16 (MSB)

General-Purpose Timer Control Register

Address: 0x40000008, Reset: 0x0040, Name: T0CON

Table 110. T0CON Register Bit Descriptions

Bits	Name	Description	
[15:5]	RESERVED	Reserved. These bits must be written 0.	
4	FREE_RUN	Timer free run enable. 0: periodic mode. In increment mode, the timer counts up to T0TOFC0/T0TOFC1 and returns to 0 and starts again. In decrement mode, the timer counts from 0xFFFF FFFF to T0TOFC0/T0TOFC1 and starts again at 0xFFFF FFFF. 1: free running mode. In increment mode, the timer counts from 0 to 0xFFFF FFFF and starts again at 0. In decrement mode, the timer counts from 0xFFFF FFFF to 0 and starts again at 0xFFFF FFFF.	
3	FREEZE	Freeze enable bit. 0: disable this feature (default). 1: enable the freezing of the upper 16 bits (T0VAL1) of the counter after the lower 16 bits are read from T0VAL0. This ensures that the software reads an atomic shot of the timer. T0VAL1 unfreezes after it is read.	
2	INC_DEC	Increment or decrement mode. 0: timer starts at zero and starts incrementing. 1: timer starts at full scale and starts decrementing.	
[1:0]	PRE	Prescaler.	
		T0CON[1:0]	**Description**
		00	Source clock ÷ 4 (default)
		01	Source clock ÷ 16
		10	Source clock ÷ 256
		11	Source clock ÷ 32,768

General-Purpose Timer Enable Register

Address: 0x4000000C, Reset: 0x0000, Name: T0EN

Table 111. T0EN Register Bit Descriptions

Bits	Name	Description
[15:1]	RESERVED	Reserved.
0	T0EN	Timer enable register. 0: disable the timer. Timer is in a reset state. 1: enable the timer. This bit must be low when configuring the control register or the T0TOFC0/T0TOFC1 register in periodic mode. Bit 8 of the status register must also be 0 before programming the control register (or T0TOFC0/T0TOFC1 in periodic mode).

General-Purpose Timeout Field A Registers

These registers can be written to at any time; however, the corresponding interrupt enable, Bit 0 of the T0IEN register, must be disabled. After these registers are updated, the interrupt can be reenabled.

Timeout Field A (LSB) Register

Address: 0x40000010, Reset: 0x1FFF, Name: T0TOFA0

Table 112. T0TOFA0 Register Bit Descriptions

Bits	Name	Description
[15:0]	VALUE	Lower 16 bits of Timeout Field A. This register can be written to at any time; however, the corresponding interrupt enable, Bit 0 of the T0IEN register, must be disabled. After the register is updated, the interrupt can be reenabled.

Timeout Field A (MSB) Register

Address: 0x40000014, Reset: 0x0000, Name: T0TOFA1

Table 113. T0TOFA1 Register Bit Descriptions

Bits	Name	Description
[15:0]	VALUE	Upper 16 bits of Timeout Field A. This register can be written to at any time; however, the corresponding interrupt enable, Bit 0 of the T0IEN register, must be disabled. After the register is updated, the interrupt can be reenabled.

General-Purpose Timeout Field B Registers

These registers can be written to at any time; however, the corresponding interrupt enable, Bit 1 of the T0IEN register, must be disabled. After the register is updated, the interrupt can be reenabled.

Timeout Field B (LSB) Register

Address: 0x40000018, Reset: 0x2FFF, Name: T0TOFB0

Table 114. T0TOFB0 Register Bit Descriptions

Bits	Name	Description
[15:0]	VALUE	Lower 16 bits of Timeout Field B. This register can be written to at any time; however, the corresponding interrupt enable, Bit 1 of the T0IEN register, must be disabled. After the register is updated, the interrupt can be reenabled.

Timeout Field B (MSB) Register

Address: 0x4000001C, Reset: 0x0000, Name: T0TOFB1

Table 115. T0TOFB1 Register Bit Descriptions

Bits	Name	Description
[15:0]	VALUE	Upper 16 bits of Timeout Field B. This register can be written to at any time; however, the corresponding interrupt enable, Bit 1 of the T0IEN register, must be disabled. After the register is updated, the interrupt can be reenabled.

General-Purpose Timeout Field C Registers

If the T0VAL register is not free running, it resets after reaching the value of these registers.

For periodic mode, these registers can only be written to when the timer is disabled. In free running mode, these registers can be written to while the timer is running. Before writing to this register, the corresponding interrupt enable, Bit 2 of the T0IEN register, must be disabled. After these registers are updated, the interrupt can be reenabled.

Timeout Field C (LSB) Register

Address: 0x40000020, Reset: 0x3FFF, Name: T0TOFC0

Table 116. T0TOFC0 Register Bit Descriptions

Bits	Name	Description
[1:0]	VALUE	Lower 16 bits of Timeout Field C

Timeout Field C (MSB) Register

Address: 0x40000024, Reset: 0x0000, Name: T0TOFC1

Table 117. T0TOFC1 Register Bit Descriptions

Bits	Name	Description
[15:0]	VALUE	Upper 16 bits of Timeout Field C

General-Purpose Timer Interrupt Enable Register

Address: 0x40000028, Reset: 0x0000, Name: T0IEN

The interrupt enable also acts as an enable for the Timeout Match A, Timeout Match B, and Timeout Match C. If the enable bit is not set, the corresponding status bit is also not set. The corresponding compare does not occur if the enable bit is not enabled.

Table 118. T0IEN Register Bit Descriptions

Bits	Name	Description
[15:4]	RESERVED	Reserved. These bits must be written 0.
3	ROLL	Rollover interrupt enable. Used only in free running mode. 0: disable the rollover interrupt (default). 1: generate an interrupt when the general-purpose timer rolls over.
2	TOFC	T0TOFC interrupt enable. 0: disable T0TOFC interrupt (default). 1: generate an interrupt when T0VAL reaches T0TOFC.
1	TOFB	T0TOFB interrupt enable. 0: disable T0TOFB interrupt (default). 1: generate an interrupt when T0VAL reaches T0TOFB.
0	TOFA	T0TOFA interrupt enable. 0: disable T0TOFA interrupt (default). 1: generate an interrupt when T0VAL reaches T0TOFA.

General-Purpose Timer Interrupt Status Register

Address: 0x4000002C, Reset: 0x0000, Name: T0ISTA

Table 119. T0ISTA Register Bit Descriptions

Bits	Name	Description
[15:9]	RESERVED	Reserved.
8	EN_SYNC	Indicates when a change in the enable bit is synchronized to the 32 kHz clock domain. 0: cleared automatically when the change in the enable bit has been synchronized to the 32 kHz clock domain. 1: set automatically when the enable bit in T0EN register is set or cleared.
7	FREEZE	Timer value freeze. T0CON[3] enables the freeze functionality. 0: cleared automatically when T0VAL1 is not frozen. 1: set automatically to indicate that the value in T0VAL1 is frozen.
[6:4]	RESERVED	Reserved.
3	ROLL	Rollover interrupt flag. Used only in free running mode. 0: cleared automatically after a write to T0CLRI. 1: set automatically to indicate a rollover interrupt has occurred (that is, T0VAL counter register is all 1s for increment mode and all 0s for decrement mode) and T0IEN[3] is enabled.
2	TOFC	T0TOFC interrupt flag. 0: cleared automatically after a read. 1: set automatically to indicate a comparator interrupt has occurred and T0IEN[2] is enabled.
1	TOFB	T0TOFB interrupt flag. 0: cleared automatically after a read. 1: set automatically to indicate a comparator interrupt has occurred and T0IEN[1] is enabled.
0	TOFA	T0TOFA interrupt flag. 0: cleared automatically after a read. 1: set automatically to indicate a comparator interrupt has occurred and T0IEN[0] is enabled.

General-Purpose Timer Clear Interrupt Register

Address: 0x40000030, Reset: Not applicable, Name: T0CLRI

Table 120. T0CLRI Register Bit Descriptions

Bits	Name	Description
[15:4]	RESERVED	Reserved. These bits must be written 0.
3	ROLL	Rollover interrupt clear bit. Used only in free running mode. 0: cleared automatically after synchronization. 1: clear a rollover interrupt flag.
2	TOFC	T0TOFC interrupt flag. 0: cleared automatically after synchronization. 1: clear a T0TOFC interrupt flag.
1	TOFB	T0TOFB interrupt flag. 0: cleared automatically after synchronization. 1: clear a T0TOFB interrupt flag.
0	TOFA	T0TOFA interrupt flag. 0: cleared automatically after synchronization. 1: clear a T0TOFA interrupt flag.

Ensure that the register write has fully completed before returning from the interrupt handler. Use the data synchronization barrier (DSB) instruction, if necessary.

WAKE-UP TIMER

WAKE-UP TIMER (TIMER 2) FEATURES

The wake-up timer (Timer 2) is a 32-bit, count up/count down timer with a programmable prescaler.

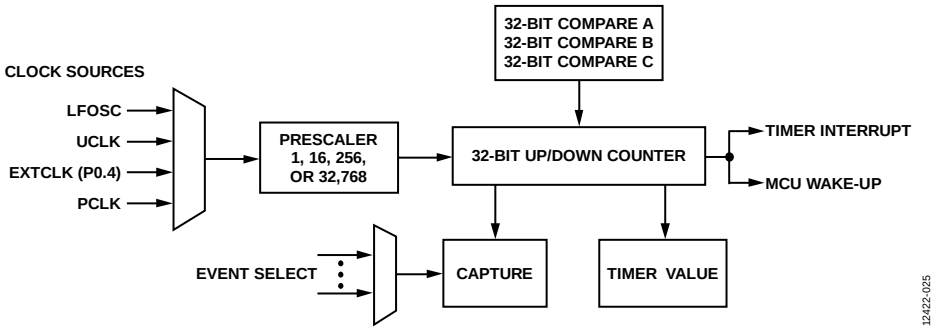

Figure 25. Wake-Up Timer Block Diagram

WAKE-UP TIMER OVERVIEW

The wake-up timer (Timer 2) is a 32-bit, count up/count down timer with a programmable prescaler. The prescaler source is selectable and can be scaled by a factor of 1, 4, 16, 256, or 32,768. Timer 2 is clocked directly by the internal 32 kHz oscillator (LFOSC), UCLK, PCLK, or an external clock on P0.4 (ECLKIN).

Timer 2 can be either free running or periodic. In free running mode, the counter decrements/increments from the maximum/minimum value until zero/full scale and starts again at the maximum/minimum value. In periodic mode, the counter decrements/increments from the value in the load register (T2LD MMR) until zero/full scale and starts again at the value stored in the load register.

Timer 2 has three time field values that can be compared with the counter.

The value of a counter can be read at any time by accessing its value register (T2VAL).

WAKE-UP TIMER OPERATION

The wake-up timer, Timer 2, is started by writing 1 to the enable bit (T2EN[0]).

In increment mode, the timer increments until the value reaches full scale (T2CON[4], FREE_RUN = 1) or until the counter matches the programmed wake-up time in T2WUFC0/T2WUFC1 (T2CON[4], FREE_RUN = 0). When this occurs, an IRQ is generated, and the timer is reset to zero.

In decrement mode, the timer starts at full scale and counts down. The timer decrements until it reaches zero (FREE_RUN = 1) or until the counter matches the programmed wake-up time in T2WUFC0/T2WUFC1 (FREE_RUN = 0). When this occurs, an IRQ is generated, and the timer is reset to full scale

The following must also be noted:

- Any timer IRQ can be cleared by writing a 1 to the correct clear bit in the T2CLRI register.
- The timer can be stopped and cleared by resetting the enable bit, T2EN[0].
- The maximum timer value is 0xFFFF FFFF.
- The timer can be read at any time.
- The control register selects the mode and selects the prescaler value.
- A separate enable register enables the timer.

Because Timer 2 is 32 bits but resides on a 16-bit bus, 2 bus reads are required to obtain the 32 bits. There are separate addresses for the upper (T2VAL1) and lower (T2VAL0) 16 bits of Timer 2. When the lower 16 bits are addressed and read, the upper 16 bits are latched and held in a separate register to be read later. The entire T2VALx registers (upper and lower) remains frozen until the upper 16 bits are read. The freeze control bit (T2CON[3]) must be set to freeze the T2VALx registers between lower and upper reads.

Timer Configuration

Take care when configuring the general-purpose timer registers to ensure correct operation of the timer. There are in total nine configuration registers:

- T2CON can only be written to when the timer is disabled.
- T2EN can be written to at any time.
- T2WUFA0, T2WUFA1, T2WUFB0, and T2WUFB1 can only be written to after the corresponding interrupt is disabled.
- For periodic mode, T2WUFC0 and T2WUFC0 can only be written to when the block is disabled. The procedure for updating these registers is detailed in the following instructions. For free running mode, these registers can be updated after the corresponding interrupt is disabled.
- T2IEN can be written to at any time.

On initial startup, all configuration registers must be configured before enabling the timer. To reconfigure the timer (T2CON register or T2WUFC registers in periodic mode), use the following procedure:

1. T2EN = 0x0
2. Wait until (T2ISTA[8] = 0) // Wait until the enable is synchronized to the timer clock domain.
3. T2CON = {new value}
4. T2EN = 0x1

To configure the wake-up field registers while the timer is enabled as follows, the following example uses T2WUFA0:

1. T2IEN = T2IEN & 0xE // Clear corresponding IRQ enable.
2. T2WUFA0 = {new value}
3. T2IEN = T2ISTA | 0x1 // (or 0x1) Enable corresponding IRQ.

This procedure can only be used for the T2WUFC registers in free running mode.

Care must be taken when disabling the interrupts. If an interrupt is about to occur when it is disabled, the interrupt may be captured by the logic; in that case, it asserts the interrupt line after the interrupt is reenabled.

Wake-Up Feature

The wake-up feature is derived from register programming that selects up to three specific time fields to compare with the corresponding wake-up timer counter.

T2WUFA to T2WUFC

After one of these three time fields (T2WUFA to T2WUFC) matches the Timer 2 counter, a wake-up event or interrupt can be generated. The top time field (T2WUFC0/T2WUFC1) value has priority. If the FREE_RUN control bit is cleared, Timer 2 is reset to zero after reaching this wake-up time (or set to full scale if in decrementing mode). The timer continues to increment/decrement, and the wake-up is periodic without software programming. If the FREE_RUN bit is set, wake-up events can be generated, but reprogramming the wake-up time field is required after one or all three time fields have been reached (unless full timer length wake-up is desired) because of the timer not being reset after each wake-up event. The interrupt enable bits for these three time values also act as wake-up pulse enable bits. If the time value of a register needs to be changed, after a wake-up has occurred, the interrupt enable bit for it must first be disabled as the match logic is in the 32 kHz time domain. This ensures that another match cannot occur. After the new value is loaded into the time field register, the interrupt can be enabled again.

WAKE-UP TIMER MEMORY MAPPED REGISTERS

Table 121. Wake-Up Timer Memory Mapped Registers (Base Address 0x40002500)

Offset	Name	Description	Access	Default
0x0000	T2VAL0	Current count value (LSB) register	R	0x0000
0x0004	T2VAL1	Current count value (MSB) register	R	0x0000
0x0008	T2CON	Control register	RW	0x0010
0x000C	T2EN	Enable register	RW	0x0000
0x0010	T2WUFA0	Wake-Up Field A (LSB) register	RW	0x1FFF
0x0014	T2WUFA1	Wake-Up Field A MSB) register	RW	0x0000
0x0018	T2WUFB0	Wake-Up Field B (LSB) register	RW	0x2FFF
0x001C	T2WUFB1	Wake-Up Field B (MSB) register	RW	0x0000
0x0020	T2WUFC0	Wake-Up Field C (LSB) register	RW	0x3FFF
0x0024	T2WUFC1	Wake-Up Field C (MSB) register	RW	0x0000
0x0028	T2IEN	Interrupt enable register	RW	Not applicable
0x002C	T2ISTA	Interrupt status register	R	0x0000
0x0030	T2CLRI	Clear interrupts register	W	0x0000
0x003C	T2CAP0	Capture event count (LSB) register	R	0x0000
0x0040	T2CAP1	Capture event count (MSB) register	R	0x0000

Wake-Up Timer Count Value Registers

Current Count Value (LSB) Register

Address: 0x40002500, Reset: 0x0000, Name: T2VAL0

Table 122. T2VAL0 Register Bit Description

Bits	Name	Description
[15:0]	VALUE	Current Timer 2 count value Bit 15 to Bit 0 (LSB)

Current Count Value (MSB) Register

Address: 0x40002504, Reset: 0x0000, Name: T2VAL1

Table 123. T2VAL1Register Bit Descriptions

Bits	Name	Description
[15:0]	VALUE	Current Timer 2 count value Bit 31 to Bit 16 (MSB)

Wake-Up Timer Control Register

Address: 0x40002508, Reset: 0x0010, Name: T2CON

Table 124. T2CON Register Bit Descriptions

Bits	Name	Description	
[15:12]	RESERVED	Reserved. These bits must be written 0.	
[11:8]	EVENT_SEL	Event select. Select one of the events for event capture.	
		T2CON[11:8]	**Description**
		0000	EXTINT0
		0001	EXTINT1
		0010	Watchdog timer
		0011	General-purpose timer
		0100	ADC
		0101	Flash
		0110	SPI
		0111	LIN0
		1000	LIN1
		1001	LIN2
		1010	HV
		1011	CALOSC
		1100	SRAM_ECC
		1101	Reserved
		1110	Reserved
		1111	Reserved
7	EVENT_ENABLE	Event capture enable. 0: disable the event capture logic. 1: enable the event capture logic.	
[6:5]	CLKSEL	Clock select.	
		T2CON[6:5]	**Description**
		00	PCLK
		01	32 kHz internal oscillator
		10	UCLK
		11	ECLKIN (external clock P0.4)
4	FREE_RUN	Timer free running enable. 0: periodic mode. In increment mode, the timer counts to T2WUFC0/T2WUFC1 and returns to 0 and starts again. In decrement mode, the timer counts from 0xFFFF FFFF to T2WUFC0/T2WUFC1 and starts again at 0xFFFF FFFF. 1: free running mode. In increment mode, the timer counts from 0 to 0xFFFF FFFF and starts again at 0. In decrement mode, the timer counts from 0xFFFF FFFF to 0 and starts again at 0xFFFF FFFF.	
3	FREEZE	Freeze enable bit. 0: cleared by user to disable this feature (default). 1: enable the freezing of the upper 16 bits (T2VAL1) of the counter after the lower bits are read from the T2VAL0 register. This ensures that the software reads an atomic shot of the timer. The T2VAL1 register unfreezes after it is read.	
2	INC_DEC	Increment or decrement mode. 0: timer starts at zero and start incrementing. 1: timer starts at full scale and start decrementing.	
[1:0]	PRE	Prescaler.	
		T2CON[1:0]	**Description**
		00	Source clock ÷ 1. If the selected clock source is UCLK/PCLK, this setting results in a prescaler of 4.
		01	Source clock ÷ 16.
		01	Source clock ÷ 16.
		10	Source clock ÷ 256.
		11	Source clock ÷ 32,768.

Wake-Up Timer Enable Register

Address: 0x4000250C, Reset: 0x0000, Name: T2EN

Table 125. T2EN Register Bit Description

Bits	Name	Description
[15:1]	RESERVED	Reserved.
0	T2EN	Timer enable register. 0: disable the timer. Timer is in a reset state. 1: enable the timer. This bit must be low when configuring the control register or the T2WUFC0/T2WUFC1 register in periodic mode. Bit 8 of the status register must also be 0 before programming the control register (or T2WUFC0/T2WUFC1 in periodic mode).

Wake-Up Field A Registers

The T2WUFAx registers can be written to at any time; however, the corresponding interrupt enable, Bit 0 of the T2IEN register, must be disabled. After these registers are updated, the interrupt can be reenabled.

Wake-Up Field A (LSB) Register

Address: 0x40002510, Reset: 0x1FFF, Name: T2WUFA0

Table 126. T2WUFA0 Register Bit Descriptions

Bits	Name	Description
[15:0]	T2WUFA0	Lower 16 bits of Wake-Up Field A. This register can be written to at any time; however, the corresponding interrupt enable, Bit 0 of the T2IEN register, must be disabled. After the register is updated, the interrupt can be reenabled.

Wake-Up Field A (MSB) Register

Address: 0x40002514, Reset: 0x0000, Name: T2WUFA1

Table 127. T2WUFA1 Register Bit Description

Bits	Name	Description
[15:0]	T2WUFA1	Upper 16 bits of Wake-Up Field A. This register can be written to at any time; however, the corresponding interrupt enable, Bit 0 of the T2IEN register, must be disabled. After the register is updated, the interrupt can be reenabled.

Wake-Up Timer Wake-Up Field B Registers

The T2WUFBx registers can be written to at any time but the corresponding interrupt enable, Bit 1 of T2IEN register, must be disabled. After these registers are updated, the interrupt can be reenabled.

Wake-Up Field B (LSB) Register

Address: 0x40002518, Reset: 0x2FFF, Name: T2WUFB0

Table 128. T2WUFB0 Register Bit Descriptions

Bits	Name	Description
[15:0]	T2WUFB0	Lower 16 bits of Wake-Up Field B. This register can be written to at any time; however, the corresponding interrupt enable, Bit 1 of the T2IEN register, must be disabled. After the register is updated, the interrupt can be reenabled.

Wake-Up Field B (MSB) Register

Address: 0x4000251C, Reset: 0x0000, Name: T2WUFB1

Table 129. T2WUFB1 Register Bit Descriptions

Bits	Name	Description
[15:0]	T2WUFB1	Upper 16 bits of Wake-Up Field B. This register can be written to at any time; however, the corresponding interrupt enable, Bit 1 of the T2IEN register, must be disabled. After the register is updated, the interrupt can be reenabled.

Wake-Up Field C Registers

If the T2VAL register is not free running, it resets after reaching the value of the T2WUFCx registers.

For periodic mode, these registers can only be written to when the timer is disabled. In free running mode, these registers can be written to while the timer is running. Before writing to these registers, the corresponding interrupt enable, Bit 2 of the T2IEN register, must be disabled. After these registers are updated, the interrupt can be reenabled.

Wake-Up Field C (LSB) Register

Address: 0x40002520, Reset: 0x3FFF, Name: T2WUFC0

Table 130. T2WUFC0 Register Bit Descriptions

Bits	Name	Description
[15:0]	T2WUFC0	Lower 16 bits of Wake-Up Field C

Wake-Up Field C (MSB) Register

Address: 0x40002524, Reset: 0x0000, Name: T2WUFC1

Table 131. T2WUFC1 Register Bit Descriptions

Bits	Name	Description
[15:0]	T2WUFC1	Upper 16 bits of Wake-Up Field C

Wake-Up Timer Interrupt Enable Register

Address: 0x40002528, Reset: Not applicable, Name: T2IEN

The interrupt enable also acts as an enable for the Wake-Up Match A, Wake-Up Match B, and Wake-Up Match C. If the enable bit is not set, the corresponding status bit cannot be set. The corresponding compare does not occur if the enable bit is not enabled.

Table 132. T2IEN Register Bit Descriptions

Bits	Name	Description
[15:4]	RESERVED	Reserved. These bits must be written 0.
3	ROLL	Rollover interrupt enable. Used only in free running mode. 0: disable the roll over interrupt (default). 1: generate an interrupt when the wake-up timer rolls over.
2	WUFC	T2WUFC interrupt enable. 0: disable T2WUFC interrupt (default). 1: generate an interrupt when T2VAL reaches T2WUFC.
1	WUFB	T2WUFB interrupt enable. 0: disable T2WUFB interrupt (default). 1: generate an interrupt when T2VAL reaches T2WUFB.
0	WUFA	T2WUFA interrupt enable. 0: disable T2WUFA interrupt (default). 1: generate an interrupt when T2VAL reaches T2WUFA.

Wake-Up Timer Interrupt Status Register

Address: 0x4000252C, Reset: 0x0000, Name: T2ISTA

Table 133. T2ISTA Register Bit Descriptions

Bits	Name	Description
[15:9]	RESERVED	Reserved.
8	EN_SYNC	Indicates when a change in the enable bit is synchronized to the 32 kHz clock domain. 0: cleared automatically when the change in the enable bit has been synchronized to the 32 kHz clock domain. 1: set automatically when the enable bit in the T2EN register is set or cleared.
7	FREEZE	Timer value freeze. T2CON[3] enables the freeze functionality. 0: cleared automatically when the T2VAL1 register is not frozen. 1: set automatically to indicate that the value in the T2VAL1 register is frozen.
6	WAKE-UP STATUS	Status of the wake-up signal to power-down control. 0: power-down of the timer can occur immediately. 1: power-down is delayed until all the bits are cleared.
5	RESERVED	Reserved.
4	CAPTURE	Capture event flag. Indicates when a capture event is pending. 0: cleared automatically when the T2CAP1 register is read. 1: set automatically when a rising edge is detected on the selected event line.
3	ROLL	Rollover interrupt flag. Used only in free running mode. 0: cleared automatically after a write to the T2CLRI register. 1: set automatically to indicate a rollover interrupt has occurred, that is, the T2VALx counter registers are all 1s for increment mode and all 0s for decrement mode.
2	WUFC	T2WUFC interrupt flag. 0: cleared automatically after a read. 1: set automatically to indicate a comparator interrupt has occurred and T0IEN[2] is enabled.
1	WUFB	T2WUFB interrupt flag. 0: cleared automatically after a read. 1: set automatically to indicate a comparator interrupt has occurred and T0IEN[1] is enabled.
0	WUFA	T2WUFA interrupt flag. 0: cleared automatically after a read. 1: set automatically to indicate a comparator interrupt has occurred and T0IEN[0] is enabled.

Wake-Up Timer Clear Interrupt Register

Address: 0x40002530, Reset: 0x0000, Name: T2CLRI

Table 134. T2CLRI Register Bit Descriptions

Bits	Name	Description
[15:5]	RESERVED	Reserved. These bits must be written 0.
4	CAPTURE_CLR	Interrupt clear bit for the capture event interrupt.
3	ROLL	Rollover interrupt clear bit. Used only in free running mode. 0: cleared automatically after synchronization. 1: clear a roll over interrupt flag.
2	WUFC	T2WUFC interrupt flag. 0: cleared automatically after synchronization. 1: clear a T2WUFC interrupt flag.
1	WUFB	T2WUFB interrupt flag. 0: cleared automatically after synchronization. 1: clear a T2WUFB interrupt flag.
0	WUFA	T2WUFA interrupt flag. 0: cleared automatically after synchronization. 1: clear a T2WUFA interrupt flag.

Ensure that the register write has fully completed before returning from the interrupt handler. Use the data synchronization barrier (DSB) instruction, if necessary.

Capture Event Count (LSB) Register

Address: 0x4000253C, Reset: 0x0000, Name: T2CAP0

Table 135. T2CAP0 Register Bit Descriptions

Bits	Name	Description
[15:0]	VALUE	Lower 16 bits of the count at which the selected event occurred

Capture Event Count (MSB) Register

Address: 0x40002540, Reset: 0x0000, Name: T2CAP01

Table 136. T2CAP01 Register Bit Descriptions

Bits	Name	Description
[15:0]	VALUE	Upper 16 bits of the count at which the selected event occurred

WATCHDOG TIMER

WATCHDOG TIMER (TIMER 3) FEATURES

The watchdog timer (Timer 3) is a 16-bit, count down timer with programmable prescaler.

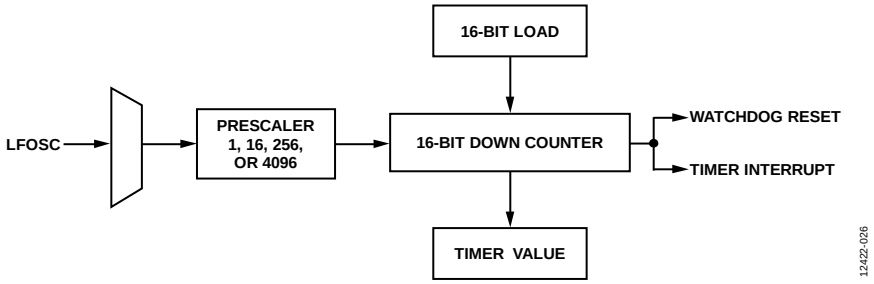

Figure 26. Watchdog Timer Block Diagram

WATCHDOG TIMER OVERVIEW

The watchdog timer (Timer 3) is a 16-bit, count down timer with a programmable prescaler. The prescaler source is selectable and can be scaled by factors of 1, 16, 256, or 4096.

Timer 3 is used to recover from an illegal software state. After being enabled by the user code, it requires periodic servicing to prevent it from forcing a reset or an interrupt of the processor.

Timer 3 is clocked by the internal 32.768 kHz oscillator (LFOSC). It is clocked at all times except during reset.

Timer 3 is active in kernel download mode.

Timer 3 timeout can generate a reset or an IRQ. The T3CON[1] bit is added to allow selecting an IRQ instead of a reset; this can be used for debug purposes. The IRQ can be cleared by writing 0xCCCC to the T3CLRI write only register.

WATCHDOG TIMER OPERATION

After a valid reset, the watchdog timer is initialized by the kernel as follows:

T3LD = 0x0400;

This initialization enables the watchdog timer with a timeout of 8 sec. This initial configuration can be modified by user code.

However, setting T3CON[5] write protects the T3CON and T3LD registers. After kernel execution, user code can disable the timer once, then reconfigure it with T3CON[5] set, once only. After this, only a full POR reset can unlock the T3CON and T3LD registers and allow reconfiguring of the timer again. If T3CON[5] is not set by user code, user code can change the T3LD register and the other bits of the T3CON register at any time. If T3CON[5] is cleared, the timer is disabled. The settings can be modified and the timer reenabled.

If the watchdog timer is set to fixed mode (T3CON[6] = 0), the watchdog timer value decrements from 0x1000 to zero, wraps around to 0x1000, and continues to decrement. To achieve a timeout value greater or less than 0x1000 (~32 sec with default prescaler = 256), periodic mode must be used (T3CON[6] = 1), and T3LD and T3CON[3:2] (prescaler) be written with the values corresponding to the desired timeout period. The maximum timeout is ~8192 sec (T3LD = 0xFFFF, prescaler = 4096).

When the watchdog timer decrements to zero, a reset (or IRQ) is generated. This reset can be prevented by writing the T3CLRI register with 0xCCCC before the expiration period. A write to the T3CLRI register causes the watchdog timer to reload with T3LD (or 0x1000 if in fixed mode) immediately to begin a new timeout period and start to count again. If any value other than 0xCCCC is written, a reset is generated (or IRQ if selected by T3CON[1]).

Power-Down Mode

If T3CON[0] is cleared to 0, the watchdog timer continues to count while the Cortex-M3 is halted. If T3CON[0] is set to 1, the watchdog timer is held at its current count while the Cortex-M3 is halted and continues from its previous value after the Cortex-M3 restarts again. This bit must be configured at the same time as T3CON[5].

Timer Configuration

Care must be taken when configuring the watchdog timer registers to ensure correct operation of the timer.

Disabling the Timer

If T3CON[5] is reset on the initial T3CON write, the timer is reset and disabled.

Reenabling the Timer

If T3CON[5] is set in a subsequent T3CON write, the watchdog timer is reset/reloaded or restarted. Depending on the state of T3CON[6], the watchdog timer restarts from 0x1000 (T3CON[6], MOD = 0) or restarts from T3LD (T3CON[6], MOD = 1).

Writing to the T3LD Register

If the T3LD register is written when MOD = 1, the watchdog timer receives the T3LD value and starts counting from there.

Write to T3CLRI to Restart the Watchdog Timer

The user writes 0xCCCC to T3CLRI within the timeout period to restart the watchdog timer normally and prevent a reset or an IRQ.

Prescaled Decrement

The enabled watchdog timer decrements when the clock divided by prescaler is reached. If the watchdog timer decrements after reaching the minimum value of 0x0000, the watchdog timer is reloaded with either 0x1000 or the T3LD value, depending on the MOD bit. This is a watchdog timeout event. A reset or IRQ is generated, depending on the IRQ bit (T3STA[0]).

WATCHDOG TIMER MEMORY MAPPED REGISTERS

Table 137. Watchdog Timer Memory Mapped Registers (Base Address 0x40002580)

Offset	Name	Description	Access	Default
0x0000	T3LD	Load value register	RW	0x0400
0x0004	T3VAL	Current count value register	R	0x1000
0x0008	T3CON	Control register	RW	0x00E9
0x000C	T3CLRI	Clear interrupt register	W	Not applicable
0x0018	T3STA	Status register	R	0x0020

Watchdog Timer Load Value Register

Address: 0x40002580, Reset: 0x0400, Name: T3LD

Table 138. T3LD Register Bit Descriptions

Bits	Name	Description
[15:0]	VALUE	Load value.

Watchdog Timer Current Count Value Register

Address: 0x40002584, Reset: 0x1000, Name: T3VAL

Table 139. T3VAL Register Bit Descriptions

Bits	Name	Description
[15:0]	VALUE	Current count value. Read only register.

Watchdog Timer Control Register

Address: 0x40002588, Reset: 0x00E9, Name: T3CON

Table 140. T3CON Register Bit Descriptions

Bits	Name	Description
[15:7]	RESERVED	Reserved.
6	MOD	Timer mode. 0: fixed mode. 1: periodic mode (default). Note that in fixed mode, the timer wraps around at 0x1000.
5	ENABLE	Timer enable. 0: disable the timer. Can only be cleared once. 1: enable the timer (default).
4	RESERVED	Reserved.

Bits	Name	Description	
[3:2]	PRE	Prescaler.	
		T3CON[3:2]	**Description**
		00	(32,768 ÷ 1) Hz
		01	(32,768 ÷ 16) Hz
		10	(32,768 ÷ 256) Hz (default)
		11	(32,768 ÷ 4096) Hz
1	IRQ	Timer interrupt. 0: generate a reset on a timeout (default). 1: generate an interrupt instead of a reset when the timer times out. This feature is provided for debug purposes and is only available in active mode.	
0	PD	Power-down off. 0: enable Timer 3 to continue counting when the Cortex-M3 is halted. 1: halt Timer 3 when the Cortex-M3 is powered down using SYSHALT or hibernate modes in the PWRMOD MMR.	

Watchdog Timer Clear Interrupt Register

Address: 0x4000258C, Reset: Not applicable, Name: T3CLRI

Table 141. T3CLRI Register Bit Descriptions

Bits	Name	Description
[15:0]	VALUE	Clear watchdog. The user writes 0xCCCC to reset/reload/restart Timer 3 or clear IRQ. A write of any other value causes a watchdog reset/IRQ. This register is write only and reads back 0.

Ensure that the register write has fully completed before returning from the interrupt handler. Use the data synchronization barrier (DSB) instruction, if necessary.

Watchdog Timer Status Register

Address: 0x40002598, Reset: 0x0020, Name: T3STA

The T3STA register is a read only status register. Because of the asynchronous relationship and frequency difference between the core clock and Timer 3 clock, changes to the watchdog timer configuration are synchronized between the two clock domains. Several of the status bits are used to signal to the core that Timer 3 clock synchronization is in progress. The user can use these bits to verify that the previous timer configuration write has taken effect in the Timer 3 clock domain, if necessary.

The status bits typically only need to be used if the watchdog timer is disabled on the initial write to the T3CON register, followed by Timer 3 configuration changes, followed by an immediate reenable of Timer 3. To ensure that all changes are in effect, do not perform the reenable until all in progress status bits are cleared.

Table 142. T3STA Register Bit Descriptions

Bits	Name	Description
[15:5]	RESERVED	Reserved.
4	LOCK	Lock status bit. 0: cleared by default until user code sets T3CON[5]. 1: set automatically in hardware if T3CON[5] has been set by user code.
3	CON_SYNC	T3CON write sync. 0: T3CON register write sync match. 1: T3CON register write sync in progress.
2	LD_SYNC	T3LD write sync. 0: T3LD register write sync match. 1: T3LD register write sync in progress.
1	CLRI_SYNC	T3CLRI write sync. 0: T3CLRI write sync not complete or inactive. 1: T3CLRI write being synced to Timer 3 clock domain. Timer 3 is restarted (if 0xCCCC was written) when sync is complete.
0	IRQ	Timer 3 IRQ. 0: Timer 3 interrupt not pending. 1: Timer 3 interrupt pending.

GENERAL-PURPOSE DIGITAL INPUTS/OUTPUTS

GENERAL-PURPOSE DIGITAL INPUTS/OUTPUTS OVERVIEW

The ADuCM330/ADuCM331 feature six, bidirectional, general-purpose input/output (GPIO) pins. All of the GPIO pins have multiple functions, configurable by user code. These features are described in Table 143.

Each GPIO pin can be configured as an input, output, or open circuit. These pins also have an internal, pull-up, programmable resistor. See the ADuCM330/ADuCM331 data sheet for logic input voltages and absolute maximum values. For optimum low power operation, the GPIOs must be configured as inputs (using GP0OEN) and with pull-up resistors enabled (using GP0PUL).

When the ADuCM330/ADuCM331 enter a power saving mode, the GPIO pins retain their state.

Note that a driving peripheral is not able to drive these pins.

GENERAL-PURPOSE DIGITAL INPUTS/OUTPUTS FEATURES

- Six, bidirectional, general-purpose input/output (GPIO) pins
- Internally multiplexed with SPI and LIN configurations available

GENERAL-PURPOSE DIGITAL PORT MULTIPLEX

This block provides control over the GPIO functionality of specified pins, because some of the pins can work as GPIOs or have other specific functions. The following tables detail the configuration modes for each GPIO.

Table 143. GPIO Multiplex

	Configuration Modes			
GPIO	**00**	**01**	**10**	**11**
P0.0	GPIO0	$\overline{CS}$	Not applicable	LIN_RX (in)[1]
P0.1	GPIO1	SCLK	Not applicable	LIN_TX (out)[1]
P0.2	GPIO2	MISO	Not applicable	Not applicable
P0.3	GPIO3/IRQ0	MOSI	LC_TX (in)[2]	LIN_TX (out)[1]
P0.4	GPIO4/IRQ1	LC_RX (out)[2]	ECLKIN	LIN_RX (in)[1]
P0.5	GPIO5[3]	LC_TX (out)[2]	Not applicable	LIN_TX (out)[1]

[1] For this mode, these GPIOs connect directly to the LIN logic interface.
[2] Used for LIN conformance testing. These GPIOs connect directly to the top die LIN Rx and Tx pads. See the LIN Interface section for more information.
[3] GPIO5 is checked by the kernel on every reset. See the Kernel section for more information.

GENERAL PURPOSE DIGITAL INPUT/OUTPUT OPERATION

Digital Port Multiplexed Configuration

The pin functions are configured using the GP0CON register. The GP0CON register configures Port 0. External interrupts and input level signals (GP0IN) are available in any configuration modes, except when the GPIOs are configured as open circuit and as outputs (GP0OE = 1 and GP0OCE = 1).

GPIO Pull-Up Enable

All GPIO pins have an internal pull-up resistor. Using the GP0PUL register, it is possible to enable/disable pull-up registers on the pins when they are configured as inputs. The pull-up resistors are automatically disabled when the pad is set as an output or when open circuit is enabled.

GPIO Data In

When configured as an input (by default), the GPIO input levels are available in GP0IN.

Open Circuit Enable

This disables the input paths if the pin is set as output. To disable the input and not drive the pin, the open circuit must be set and drive Logic 1. External interrupts interrupt are not available when open circuit is enabled.

GPIO Data Out

When the GPIOs are configured as outputs, the values in GP0OUT are reflected on the GPIOs.

Bit Set

Bit set is used to set one or more GPIO data outputs without affecting others within a port. Only the GPIO corresponding with the write data bit equal to one is set; the remaining GPIOs are unaffected.

Bit Clear

Bit clear is used to clear one or more GPIO data outputs without affecting others within a port. Only the GPIO corresponding with the write data bit equal to one is cleared; the remaining GPIOs are unaffected.

Bit Toggle

Bit toggle is used to toggle one or more GPIO data outputs without affecting others within a port. Only the GPIO corresponding with the write data bit equal to one is toggled; the remaining GPIOs are unaffected.

GPIO Data Output Enable

The data output path is enabled; the values in GP0OUT are reflected on the GPIOs.

GPIO MEMORY MAPPED REGISTERS

Table 144. GPIO Interface Memory Address (Base Address 0x40006000)

Offset	Name	Description	Access	Default[1]
0x0000	GP0CON	GPIO Port 0 configuration register	RW	0x0000
0x0004	GP0OEN	GPIO Port 0 output enable register	RW	0x00
0x0008	GP0PUL	GPIO Port 0 output pull-up enable register	RW	0x3F
0x000C	GP0OCE	GPIO Port 0 open circuit enable register	RW	0x00
0x0014	GP0INR	GPIO Port 0 input data register[2]	R	XX
0x0018	GP0OUT	GPIO Port 0 data out register	RW	0x00
0x001C	GP0SET	GPIO Port 0 data out set register	W	0x00
0x0020	GP0CLR	GPIO Port 0 data out clear register	W	0x00
0x0024	GP0TGL	GPIO Port 0 pin toggle	W	0x00

[1] X means don't care.
[2] Contents of the GP0INR register depend on the digital level on the corresponding pins.

GPIO Configuration Register

Address: 0x40006000, Reset: 0x0000, Name: GP0CON

Table 145. GP0CON Register Bit Descriptions

Bits	Name	Description
[15:12]	RESERVED	Reserved
[11:10]	CON5	Configuration bits for P0.5 (see Table 143)
[9:8]	CON4	Configuration bits for P0.4 (see Table 143)
[7:6]	CON3	Configuration bits for P0.3 (see Table 143)
[5:4]	CON2	Configuration bits for P0.2 (see Table 143)
[3:2]	CON1	Configuration bits for P0.1 (see Table 143)
[1:0]	CON0	Configuration bits for P0.0 (see Table 143)

GPIO Output Enable Register

Address: 0x40006004, Reset: 0x00, Name: GP0OEN

Table 146. GP0OEN Register Bit Descriptions

Bits	Name	Description
[7:6]	RESERVED	Reserved
[5:0]	GP0OEN	Input/output enable 0: enables the corresponding GPIO on Port 0 as an input 1: enables the corresponding GPIO on Port 0 as an output

GPIO Output Pull-Up Enable Register

Address: 0x40006008, Reset: 0x3F, Name: GP0PUL

Table 147. GP0PUL Register Bit Descriptions

Bits	Name	Description
[7:6]	RESERVED	Reserved
[5:0]	GP0PUL	Pull-up enable 0: disables the internal pull-up resistor on corresponding GPIO on Port 0 1: enables the internal pull-up resistor on corresponding GPIO on Port 0

GPIO Open Circuit Enable Register

Address: 0x4000600C, Reset: 0x00, Name: GP0OCE

Table 148. GP0OCE Register Bit Descriptions

Bits	Name	Description
[7:6]	RESERVED	Reserved.
[5:0]	GP0OCE	Open circuit enable. Sets the GPIO pads on Port 0 to open circuit mode.

GPIO Input Data Register

Address: 0x40006014, Reset: 0xX[1], Name: GP0INR

Table 149. GP0INR Register Bit Descriptions

Bits	Name	Description
[7:6]	RESERVED	Reserved
[5:0]	GP0IN	Reflects the level on the GPIO pins on Port 0 except when in configured in open circuit

[1] Contents of the GP0INR register depends on the digital level on the corresponding pins

GPIO Data Out Register

Address: 0x40006018, Reset: 0x00, Name: GP0OUT

Table 150. GP0OUT Register Bit Descriptions

Bits	Name	Description
[7:6]	RESERVED	Reserved.
[5:0]	GP0OUT	Data out register. 0: cleared by user to drive the corresponding GPIO low. 1: set by user code to drive the corresponding GPIO high. Reads back the value on GPIO outputs. For example, writing GP0OUT = 0x12 drives 1 on P0.1 and P0.4, and the remaining GPIO are low, assuming these pins are configured as outputs.

GPIO Data Out Set Register

Address: 0x4000601C, Reset: 0x00, Name: GP0SET

Table 151. GP0SET Register Bit Descriptions

Bits	Name	Description
[7:6]	RESERVED	Reserved
[5:0]	GP0SET	Data out set register 0: no action 1: set by user code to drive the corresponding GPIO high

GPIO Data Out Clear Register

Address: 0x40006020, Reset: 0x00, Name: GP0CLR

Table 152. GP0CLR Register Bit Descriptions

Bits	Name	Description
[7:6]	RESERVED	Reserved
[5:0]	GP0CLR	Data out clear register 0: cleared by user code; has no effect 1: set by user code to drive the corresponding GPIO low

GPIO Pin Toggle Register

Address: 0x40006024, Reset: 0x00, Name: GP0TGL

Table 153. GP0TGL Register Bit Descriptions

Bits	Name	Description
[7:6]	RESERVED	Reserved
[5:0]	GP0TGL	Data out toggle register 0: cleared by user code; has no effect 1: set by user code to invert the corresponding GPIO; automatically cleared after input is toggled

SERIAL PERIPHERAL INTERFACE

SPI FEATURES

The standard SPI features included are as follows:

- Serial clock phase mode (CPHA)
- Serial clock polarity mode (CPOL)
- LSB first transfer option
- Loopback mode
- Master or slave mode
- Transfer and interrupt mode
- Continuous transfer mode
- Tx/Rx first in, first out (FIFO)
- Interrupt mode, interrupt after 1, 2, 3, or 4 bytes
- Rx overflow mode and Tx underrun mode
- Open circuit data output mode
- Full-duplex communications supported

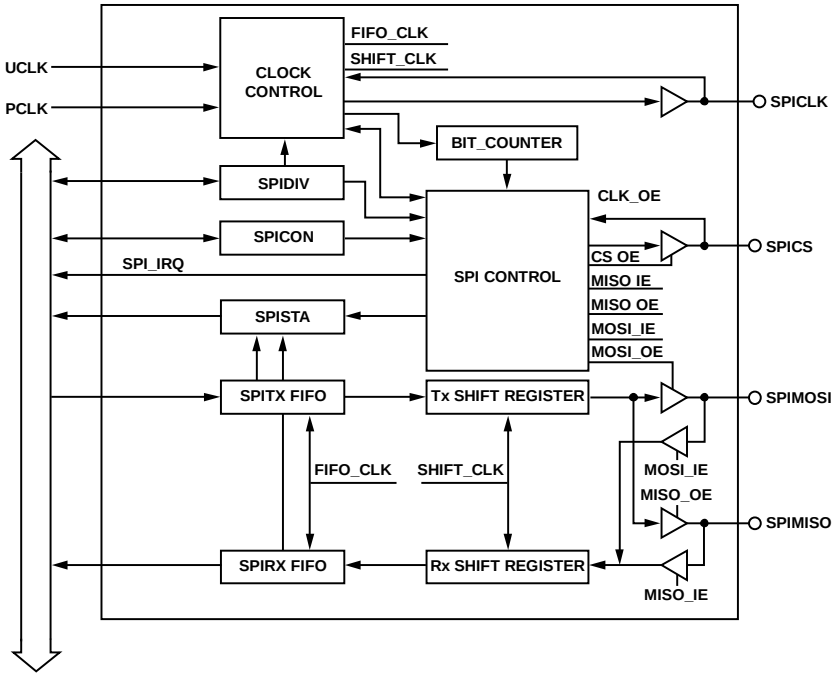

Figure 27. SPI Block Diagram

SPI OVERVIEW

The ADuCM330/ADuCM331 integrate a serial peripheral interface (SPI). The SPI is an industry standard, synchronous serial interface that allows eight bits of data to be synchronously transmitted and simultaneously received, that is, full duplex. The SPI implemented on the ADuCM330/ADuCM331 can operate to a maximum bit rate of 8 Mbps in both master and slave mode.

SPI OPERATION

The SPI port can be configured for master or slave operation and consists of four pins: MISO, MOSI, SCLK, and $\overline{CS}$. In an SPI system, the master device generates a serial clock and asserts the chip select for the appropriate slave device. $\overline{CS}$ frames the transfer. Data is simultaneously transmitted from the master to the slave using the master out, slave in (MOSI) pin and transmitted from the slave to the master using the master in, slave out (MISO) pin. In normal master operation, the MOSI pin is an output, and the MISO pin is an input. In normal slave operation, the MOSI pin is an input, and the MISO pin is an output.

Note that the GPIOs used for SPI communication must be configured in SPI mode before enabling the SPI peripheral, and that the internal pull-up resistors on the SPI pins must be disabled via the GP0PUL register when using the SPI. To enable the SPI peripheral in slave mode, the $\overline{CS}$ pin must be monitored to ensure that communication is inactive first.

MISO (Master In, Slave Out) Pin

The MISO pin is configured as an input line in master mode and as an output line in slave mode. The MISO line on the master (data in) must be connected to the MISO line in the slave device (data out). The data is transferred as byte-wide (8-bit) serial data.

MOSI (Master Out, Slave In) Pin

The MOSI pin is configured as an output line in master mode and as an input line in slave mode. The MOSI line on the master (data out) must be connected to the MOSI line in the slave device (data in). The data is transferred as byte-wide (8-bit) serial data.

SCLK (Serial Clock Input/Output) Pin

The master serial clock (SCLK) synchronizes the data being transmitted and received through the MOSI SCLK period. Therefore, a byte is transmitted/received after eight SCLK periods. The SCLK pin is configured as an output in master mode and as an input in slave mode.

In master mode, the polarity and phase of the clock are controlled by the SPICON register, and the bit rate is defined in the SPIDIV register as follows:

$$f_{SERIALCLOCK} = \frac{f_{UCLK}}{2 \times (1 + \text{SPIDIV}[5:0])}$$

The maximum data rate is 8 Mbps.

In slave mode, the SPICON register must be configured with the phase and polarity of the expected input clock. The slave accepts data from an external master up to 8 Mbps.

In both master and slave mode, data is transmitted on one edge of the SCLK signal and sampled on the other. Therefore, it is important that the polarity and phase be configured the same for the master and slave devices.

Chip Select ($\overline{CS}$ Input) Pin

In SPI slave mode, a transfer is initiated by the assertion of $\overline{CS}$, which is an active low input signal. The SPI port then transmits and receives 8-bit data until the transfer is concluded by deassertion of $\overline{CS}$. In slave mode, $\overline{CS}$ is always an input.

In SPI master mode, $\overline{CS}$ is an active low output signal. $\overline{CS}$ asserts itself automatically at the beginning of a transfer and deasserts itself upon completion.

SPI TRANSFER INITIATION

In master mode, the transfer and interrupt mode bit, SPICON[6], determines the manner in which an SPI serial transfer is initiated. If the mode bit (MOD) is set, a serial transfer is initiated after a write to the Tx FIFO occurs. If the mode bit is cleared, a serial transfer is initiated after a read of the Rx FIFO; the read must be performed while the SPI interface is idle. A read performed during an active transfer cannot initiate another transfer.

For any setting of SPICON[1] and SPICON[6], the SPI simultaneously receives and transmits data. Therefore, during data transmission, the SPI is also receiving data and filling up the Rx FIFO. If the data is not read from the Rx FIFO, the overflow interrupt occurs after the FIFO starts to overflow. If the user does not want to read the Rx data or receive overflow interrupts, SPICON[12] can be set, and the receive data cannot be saved to the Rx FIFO.

Similarly, when the user only wants to receive data and does not want to write data to the Tx FIFO. SPICON[13] can be set to avoid receiving underrun interrupts from the Tx FIFO.

Tx Initiated Transfer

For transfers initiated by a write to the Tx FIFO, the SPI begins transmitting as soon as the first byte is written to the FIFO, irrespective of the configuration in SPICON[15:14]. The first byte is immediately read from the FIFO and written to the Tx shift register, and the transfer commences.

If the continuous transfer enable bit, SPICON[11], is set, the transfer continues until no valid data is available in the Tx FIFO. There is no stall period between transfers where $\overline{CS}$ is deasserted; $\overline{CS}$ is asserted and remains asserted for the duration of the transfer until the Tx FIFO is empty. When the transfer stops does not depend on SPICON[15:14]; the transfer stops when there is no valid data left in the FIFO. Conversely, the transfer continues while there is valid data in the FIFO.

If the continuous transfer enable bit, SPICON[11], is cleared, each transfer consists of a single, 8-bit serial transfer. If valid data exists in the Tx FIFO, a new transfer is initiated after a stall period where $\overline{CS}$ is deasserted.

Rx Initiated Transfer

Transfers initiated by a read of the Rx FIFO depend on the number of bytes to be received in the FIFO. If SPICON[15:14] is set to 3, and a read to the Rx FIFO occurs, the SPI initiates a 4-byte transfer. If continuous mode is set, the four bytes occur continuously with no deassertion of $\overline{CS}$ between bytes. If continuous mode is not set, the four bytes occur with stall periods between transfers where the $\overline{CS}$ is deasserted.

If SPICON[15:14] is set to 0x2 and a read to the Rx FIFO occurs, the SPI initiates a 3-byte transfer. If SPICON[15:14] is set to 0x1 and a read to the Rx FIFO occurs, the SPI initiates a 2-byte transfer. If SPICON[15:14] is set to 0x0 and a read to the Rx FIFO occurs, the SPI initiates a 1-byte transfer.

A read of the Rx FIFO while the SPI is receiving data does not initiate another transfer after the present transfer is complete. In slave mode, a transfer is initiated by the assertion of $\overline{CS}$. The device as a slave transmits and receives 8-bit data until the transfer is concluded by the deassertion of $\overline{CS}$.

The SPI transfer protocol diagrams (Figure 28 and Figure 29) illustrate the data transfer protocol for the SPI and the effects of the CPHA (SPICON[2]) and CPOL (SPICON[3]) bits in the control register on that protocol.

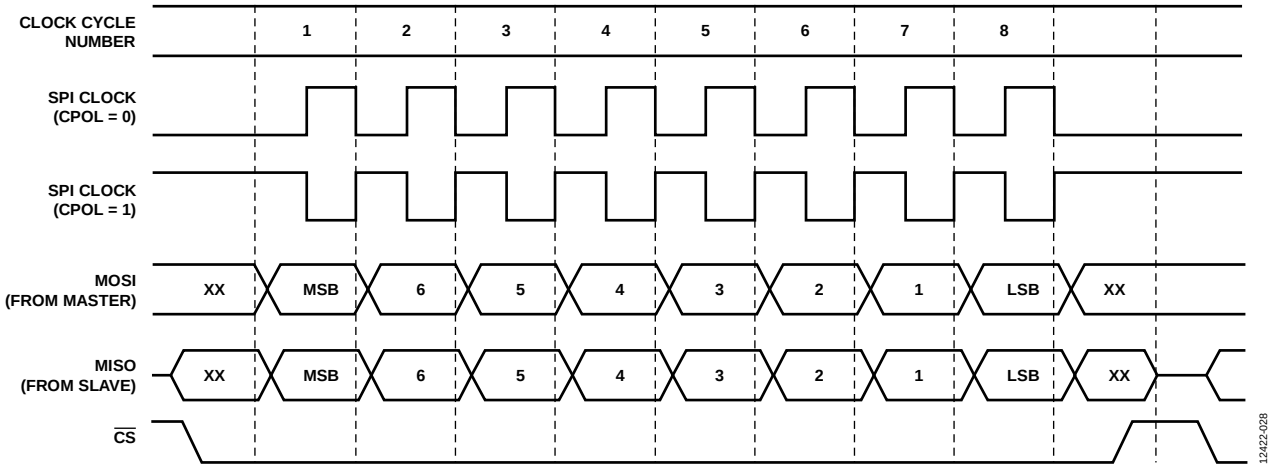

Figure 28. SPI Transfer Protocol, CPHA = 0

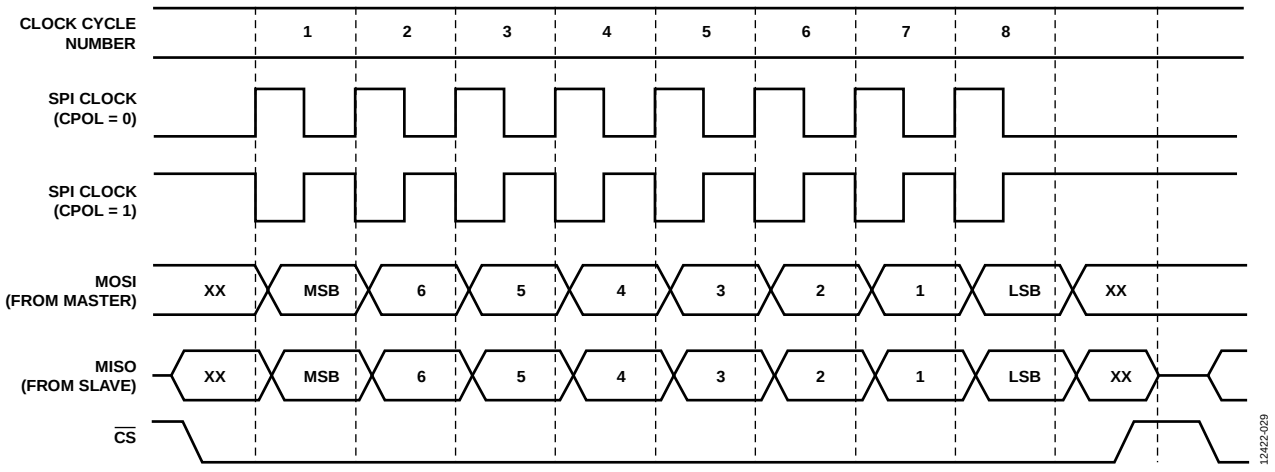

Figure 29. SPI Transfer Protocol, CPHA = 1

SPI Data Underrun and Overflow

If the Tx underrun mode bit, ZEN (SPICON[7]), is cleared, the last byte from the previous transmission is shifted out when a transfer is initiated with no valid data in the FIFO. If ZEN is set, 0s are transmitted when a transfer is initiated with no valid data in the FIFO.

If the Rx overflow overwrite enable bit, RXOF (SPICON[8]), is set, the valid data in the Rx FIFO is overwritten by the new serial byte received when there is no space left in the FIFO. If RXOF is cleared, the new serial byte received is discarded when there is no space left in the FIFO.

When valid data is being overwritten in the Rx FIFO, the oldest byte is overwritten first, followed by the next oldest byte, and so on.

SPI INTERRUPTS

There are five sources of interrupts. SPISTA[0] reflects the state of the interrupt line. SPISTA[7:4] and SPISTA[12] reflect the state of the five sources. The SPI generates either TIRQ or RIRQ. Both interrupts cannot be enabled at the same time. The appropriate interrupt is enabled using the TIM bit in the SPICON register. If TIM = 1, TIRQ is enabled. If TIM = 0, RIRQ is enabled. In addition, the CSERR bit (SPISTA[12]) is always generating an interrupt.

In master mode, interrupts are generated after the last SCLK edge for each byte transfer. In slave mode, interrupts are generated by either the first SCLK edge of the next byte transferred, or by the deassertion of $\overline{CS}$.

Tx Interrupt

If the TIM bit (SPICON[6]) is set, the Tx FIFO status causes the interrupt. SPICON[15:14] controls when the interrupt occurs, as shown in Table 154.

Table 154. SPICON[15:14] IRQ Mode Bits

SPICON[15:14]	Interrupt Condition
0x0	An interrupt is generated after each byte that is transmitted. The interrupt occurs when the byte is read from the FIFO and written to the shift register.
0x1	An interrupt is generated after every two bytes that are transmitted.
0x2	An interrupt occurs after every third byte that is transmitted.
0x3	An interrupt occurs after every fourth byte that is transmitted.

The interrupts are generated depending on the number of bytes transmitted and not on the number of bytes in the FIFO. This is unlike the Rx interrupt, which depends on the number of bytes in the Rx FIFO and not on the number of bytes received.

The transmit interrupt is cleared by a read to the status register. The status of this interrupt can be cleared by reading SPISTA[5]. The interrupt is disabled if SPICON[13] is left high.

A write to the control register, SPICON, resets the transmitted byte counter back to zero. For example, in a case where SPICON[15:14] is set to 0x3 and SPICON is written to after three bytes are transmitted, the Tx interrupt does not occur until another four bytes are transmitted.

Rx Interrupt

If the TIM bit is cleared, the Rx FIFO status causes the interrupt. SPICON[15:14] controls when the interrupt occurs. The interrupt is cleared by a read of SPISTA. The status of this interrupt can be cleared by reading SPISTA[6].

Interrupts are only generated when data is written to the FIFO. For example, if SPICON[15:14] is set to 0x0, an interrupt is generated after the first byte is received. When the status register is read, the interrupt is deactivated. If the byte is not read from the FIFO, the interrupt cannot be regenerated. Another interrupt cannot be generated until another byte is received into the FIFO.

The interrupt depends on the number of valid bytes in the FIFO and not on the number of bytes received. For example, when SPICON[15:14] is set to 0x1, an interrupt is generated after a byte is received when there are two or more bytes in the FIFO. The interrupt is not generated after every two bytes received. The interrupt is disabled if SPICON[12] is left high.

Underrun/Overflow Interrupts

SPISTA[7] and SPISTA[4] also generate SPI interrupts. When a transfer starts with no data in the Tx FIFO, SPISTA[4] is set to indicate an underrun condition. This causes an interrupt. The interrupt (and status bit) are cleared on a read of the status register. This interrupt occurs irrespective of SPICON[15:14]. This interrupt is disabled if SPICON[13] is set. Note that at the last edge of the SPICLK of the actual byte transferred, the next byte must be available in the Tx FIFO to be transferred into the Tx shift register.

When data is received and the Rx FIFO is already full, Bit 8 of the status register goes high to indicate an overflow condition. This causes an interrupt. The interrupt (and status bit) are cleared on a read of the status register. This interrupt occurs irrespective of SPICON[15:14]. This interrupt is disabled if SPICON[12] is set.

All interrupts are cleared by a read of the status register or if SPICON[0] is deasserted. The Rx and Tx interrupts are also cleared if the relevant flush bits are asserted. Otherwise, the interrupts stay active even if the SPI is reconfigured.

WIRE-OR'ED MODE (WOM)

To prevent contention when the SPI is used in a multimaster or multislave system, the MOSI and MISO data output pins can be configured to behave as open circuit drivers. An external pull-up resistor is required when this feature is selected. The WOM bit in the control register controls the pad enable outputs for the data lines. When WOM is enabled in master mode, the output drive is enabled when a 1 is being transmitted on the MOSI pin, and the output driver is disabled when a 0 is being transmitted on the MOSI pin.

Similarly, when in slave mode and WOM is enabled, the output drive is enabled when a 0 is being transmitted on the MISO pin, and the output driver is disabled when a 1 is being transmitted on the MISO pin.

CSERR CONDITION

The SPI BIT_COUNTER (see Figure 27) is reset after the completion of eight clocks of SCLK. The CSERR bit (SPISTA[12]) indicates if an erroneous deassertion of the $\overline{CS}$ signal has been detected before the completion of all the eight SCLK cycles. This bit generates an interrupt. The BIT_COUNTER stays at the value where it stopped and then continues from there when $\overline{CS}$ is asserted afterwards. This continuation of the BIT_COUNTER may cause inconsistent data transfers. To avoid inconsistent data transfers, the CSERR detection circuit is available. The CSERR detection circuit checks for $\overline{CS}$ deassertion when BIT_COUNTER is not equal to 7 (reset value). If the condition is hit, the circuit asserts a CSERR signal. If an interrupt occurs, generated by the CSERR bit (SPISTA[12]), the SPI enable bit (SPICON[0]) must be disabled and restarted to enable a clean recovery. This process ensures that subsequent transfers are error free. The BCRST bit (SPIDIV[7]) must be set at all times in both slave and master mode, except when a midbyte stall in SPI communication is required. In this case, the CSERR flag is set but can be ignored.

Note that the SPI must only be reenabled when the $\overline{CS}$ signal is high.

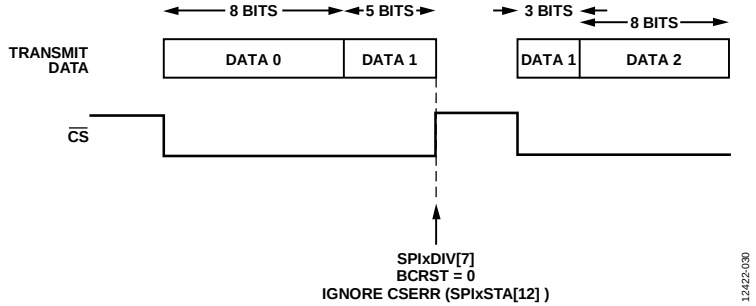

Figure 30. High Voltage Interface, Top-Level Block Diagram

SPI AND POWER-DOWN MODES

In master mode, before entering power-down mode, it is recommended to disable the SPI block in SPICON[0]. In slave mode, the $\overline{CS}$ line level must be checked via the GPIO registers to ensure that the SPI is not communicating, and the SPI block must be disabled while the $\overline{CS}$ line is high. At power-up, the SPI block can be reenabled. In slave mode, the SPI must be reenabled only if the $\overline{CS}$ line is high.

SPI MEMORY MAPPED REGISTERS

Table 155. SPI Peripheral Memory Address (Base Address 0x40004000)

Offset	Name	Description	Access	Default
0x0000	SPISTA	Status register	R	0x0000
0x0004	SPIRX	8-bit receive register	R	0x0000
0x0008	SPITX	8-bit transmit register	W	0x0000
0x000C	SPIDIV	8-bit baud rate selection register	RW	0x0000
0x0010	SPICON	16-bit configuration register	RW	0x0000

SPI Status Register

Address: 0x40004000, Reset: 0x0000, Name: SPISTA

Table 156. SPISTA Register Bit Descriptions

Bits	Name	Description	
[15:13]	RESERVED	Reserved.	
12	CSERR	Detected an abrupt $\overline{CS}$ deassertion. 0: cleared when the SPISTA register is read. 1: set when the $\overline{CS}$ line is deasserted abruptly, even before the full byte of data is transmitted completely. This bit causes an interrupt. If the CSERR bit is set, it is recommended to clear the enable bit in the SPICON register to ensure a clean recovery.	
11	RXS	SPI Rx FIFO excess bytes present. Indicates when there are more bytes in the Rx FIFO than the Rx interrupt indicated. This bit depends on SPICON[15:14].	
		SPICON[15:14]	**RXS**
		00	RXS is set if there are two or more bytes in the Rx FIFO.
		01	RXS is set if there are three or more bytes in the Rx FIFO.
		10	RXS is set if there are four or more bytes in the Rx FIFO.
		11	RXS is not set.
		Cleared to 0 when the number of bytes in the FIFO is equal or less than the number in SPICON[15:14]. This bit is not cleared when SPISTA register is read. This bit is not dependent on SPICON[6] and does not cause an interrupt.	

Bits	Name	Description	
[10:8]	RXFSTA	SPI Rx FIFO status bits.	
		SPISTA[10:8]	**Description**
		000	Rx FIFO is empty
		001	One valid byte in the FIFO
		010	Two valid bytes in the FIFO
		011	Three valid bytes in the FIFO
		100	Four valid bytes in the FIFO
7	RXOF	SPI Rx FIFO overflow status bit (interrupt). 0: cleared to 0 when the SPISTA register is read. 1: set to 1 when the Rx FIFO is already full when new data is loaded to the FIFO. This bit generates an interrupt, except when the RFLUSH bit is set in the SPICON register.	
6	RX	SPI Rx IRQ status bit. 0: cleared to 0 when the SPISTA register is read. 1: set to 1 when a receive interrupt occurs. This bit is set when the TIM bit in the SPICON register is cleared, and the required number of bytes are received.	
5	TX	SPI Tx IRQ status bit. 0: cleared to 0 when the SPISTA register is read. 1: set to 1 when a transmit interrupt occurs. This bit is set when the TIM bit in the SPICON register is set, and the required number of bytes are transmitted.	
4	TXUR	SPI Tx FIFO underrun (interrupt). 0: cleared to 0 when the SPISTA register is read. 1: set to 1 when a transmit is initiated without any valid data in the Tx FIFO. This bit generates an interrupt, except when the TFLUSH bit is set in the SPICON register.	
[3:1]	TXFSTA	Indicates how many valid bytes are in the SPI Tx FIFO.	
		SPISTA[3:1]	**Description**
		000	Rx FIFO is empty
		001	One valid byte in the FIFO
		010	Two valid bytes in the FIFO
		011	Three valid bytes in the FIFO
		100	Four valid bytes in the FIFO
		In master mode, when a byte is copied from the FIFO to the shift register, the status bits are decremented. In slave mode, the status bits are not decremented until the byte has been transmitted out of the shift register. Therefore, in master mode, the status bits reflect the number of valid bytes in the FIFO; however, in slave mode, the status bits reflect the number of valid bytes in the FIFO and also in the shift register. This is due to a byte being immediately transmitted in master mode, whereas in slave mode a valid byte can wait in the shift register for the master SPI to initiate a transfer.	
0	IRQ	SPI interrupt status bit. 0: cleared to 0 after reading the SPISTA register. 1: set to 1 when an SPI-based interrupt occurs.	

SPI 8-Bit Receive Register

Address: 0x40004004, Reset: 0x0000, Name: SPIRX

Table 157. SPIRX Register Bit Descriptions

Bits	Name	Description
[15:8]	RESERVED	These bits are reserved and must be written 0 by user code.
[7:0]	VALUE	8-bit receive register. A read of the Rx FIFO returns the next byte to be read from the FIFO. A read of the FIFO when it is empty returns 0s.

SPI 8-Bit Transmit Register

Address: 0x40004008, Reset: 0x0000, Name: SPITX

Table 158. SPITX Register Bit Descriptions

Bits	Name	Description
[15:8]	RESERVED	These bits are reserved and must be written 0 by user code.
[7:0]	VALUE	8-bit transmit register. A write to the Tx FIFO address space writes data to the next available location in the Tx FIFO. If the FIFO is full, the oldest byte of data in the FIFO is overwritten. A read from this address location returns 0s.

SPI 8-Bit Baud Rate Selection Register

Address: 0x4000400C, Reset: 0x0000, Name: SPIDIV

Table 159. SPIDIV Register Bit Descriptions

Bits	Name	Description
[15:8]	RESERVED	These bits are reserved and must be written 0 by user code.
7	BCRST	Reset mode for CSERR. This bit is used to configure the expected behavior of the SPI Interface logic after an abrupt deassertion of $\overline{CS}$. 0: SPI interface logic continues from where it stopped. The SPI can receive the remaining bits when $\overline{CS}$ is asserted, and user code must ignore the CSERR interrupt. 1: SPI interface logic is reset after a CSERR condition, and user code must clear the SPI enable bit in the SPICON register.
6	RESERVED	Reserved.
[5:0]	DIV	Factor used to divide UCLK to generate the serial clock. $f_{SERIAL\ CLOCK} = f_{UCLK} \div (2 \times (1 + SPIDIV[5:0]))$ Maximum frequency for the serial clock is ½ the UCLK frequency. These bits are only used for master mode. In slave mode, there is no need to set the serial clock frequency; the slave receives the clock from the master.

Note that when setting the SPI serial clock, the PCLK frequency must be taken into account. The PCLK frequency can be no less than half the SPI serial clock frequency. For example, if the SPI clock divide register is set to 0x0000 (SCLK frequency = ½ UCLK frequency), the maximum that the CD bits can be set to is 2 (PCLK frequency = ¼ UCLK frequency).

SPI 16-Bit Configuration Register

Address: 0x40004010, Reset: 0x0000, Name: SPICON

Table 160. SPICON Register Bit Descriptions

Bits	Name	Description	
[15:14]	MOD	SPI IRQ mode bits. When the TIM bit is set, these bits configure when the Tx/Rx interrupts occur in a transfer.	
		SPICON[15:14]	**Description**
		00	Tx interrupt occurs when one byte has been transferred. Rx interrupt occurs when one or more bytes have been received into the FIFO.
		01	Tx interrupt occurs when two bytes has been transferred. Rx interrupt occurs when two or more bytes have been received into the FIFO.
		10	Tx interrupt occurs when three bytes has been transferred. Rx interrupt occurs when three or more bytes have been received into the FIFO.
		11	Tx interrupt occurs when four bytes has been transferred. Rx interrupt occurs when the Rx FIFO is full, or four bytes are present.
13	TFLUSH	SPI Tx FIFO flush enable bit. 0: disable Tx FIFO flushing. 1: flush the Tx FIFO. This bit does not clear itself and must be toggled if a single flush is required. If this bit is left high, either the last transmitted value or 0x00 is transmitted, depending on the ZEN bit. Any writes to the Tx FIFO are ignored while this bit is set.	
12	RFLUSH	SPI Rx FIFO flush enable bit. 0: disable Rx FIFO flushing. 1: flush the Rx FIFO. This bit does not clear itself and must be toggled if a single flush is required. If this bit is set, all incoming data is ignored, and no interrupts are generated. If this bit is set and TIM = 0, a read of the Rx FIFO initiates a transfer.	
11	CON	Continuous transfer enable. 0: disable continuous transfer. Each transfer consists of a single, 8-bit serial transfer. If valid data exists in the SPITX register, a new transfer is initiated after a stall period of one serial clock cycle. The $\overline{CS}$ line is deactivated for this one serial clock cycle. 1: enable continuous transfer. In master mode, the transfer continues until no valid data is available in the SPITX register. $\overline{CS}$ is asserted and remains asserted for the duration of each 8-bit serial transfer until SPITX is empty.	
10	LOOPBACK	Loopback enable bit. 0: normal mode. 1: connect MISO to MOSI; data transmitted from the SPITX register is looped back to the SPIRX register. The MASEN bit must be set for loopback mode to work.	

Bits	Name	Description
9	OEN	Slave MISO output enable bit. 0: disable the output driver on the MISO pin. The MISO pin is open circuit when this bit is clear. 1: MISO operates as normal.
8	RXOF	SPIRX overflow overwrite enable. 0: the new serial byte received is discarded. 1: the valid data in the SPIRX register is overwritten by the new serial byte received.
7	ZEN	SPI transmit 0s when the Tx FIFO is empty. 0: transmit the last transmitted value when there is no valid data in the Tx FIFO. 1: transmit 0x00 when there is no valid data in the Tx FIFO.
6	TIM	SPI transfer and interrupt mode. 0: initiate transfer with a read of the SPIRX register. The read must be done while the SPI interface is idle. Interrupt only occurs when Rx is full. 1: initiate transfer with a write to the SPITX register. Interrupt only occurs when Tx is empty.
5	LSB	LSB first transfer enable bit. 0: MSB is transmitted first. 1: LSB is transmitted first.
4	WOM	SPI wired or mode enable bit. 0: normal output levels. 1: enable open circuit data output. Master mode: when a 0 is being transmitted on the MOSI pin, the output driver is enabled. When a 1 is being transmitted on the MOSI pin, the output driver is disabled, and an external pull-up resister is required to pull the pin high. Typical resistor value is 1 kΩ. Slave mode: when a 0 is being transmitted on the MISO pin, the output driver is enabled. When a 1 is being transmitted on the MISO pin, the output driver is disabled, and an external pull-up resister is required to pull the pin high. Typical resistor value is 1 kΩ.
3	CPOL	Serial clock polarity mode bit. 0: serial clock idles low. 1: serial clock idles high.
2	CPHA	Serial clock phase mode bit. 0: serial clock pulses at the middle of the first data bit transfer. 1: serial clock pulses at the start of the first data bit.
1	MASEN	Master mode enable bit. 0: enable slave mode. 1: enable master mode.
0	ENABLE	SPI enable bit. 0: disable the SPI. Clearing this bit also reset all the FIFO related logic and BIT_COUNTER to enable a clean start. 1: enable the SPI.

Note that, when changing the configuration, take care not to change it during a data transfer to avoid corrupting the data. It is recommended to change the configuration when the module is disabled (disable the SPI, ENABLE = 0), then reconfigure and reenable the SPI (ENABLE = 1). When reconfiguring from slave mode to master mode, or vice versa, both FIFOs must be empty. It is recommended in slave mode to reenable the SPI only if the $\overline{CS}$ line is high.

HIGH VOLTAGE PERIPHERAL CONTROL INTERFACE

HIGH VOLTAGE PERIPHERAL CONTROL INTERFACE OVERVIEW

The top and the bottom die are connected by a simple 3-wire interface: DCLK, DATA, and INTR. The bottom voltage die is the master to the top die slave. DCLK is clocked at UCLK ÷ 8.

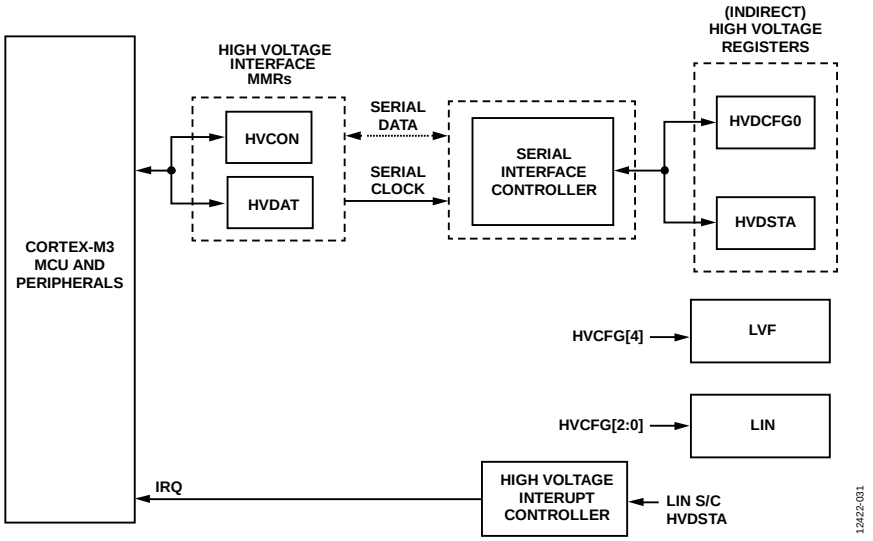

Figure 31. High Voltage Interface, Top-Level Block Diagram

HIGH VOLTAGE PERIPHERAL CONTROL INTERFACE OPERATION

Two methods are implemented for communicating data to and from the top die of the ADuCM330/ADuCM331:

- The keyhole approach that is similar to previous intelligent battery sensor devices from Analog Devices.
- A new, direct MMR programming method.

The user can decide which method is used.

Keyhole Approach

To read or write data to the high voltage (HV) top die, a keyhole configuration is implemented in the master side of the interface. This means that the user must use the HVDAT and HVCON registers on the bottom die for accessing the registers in the top die.

Two types of operation are allowed:

- Read. The registers in the HV side can be accessed from the master by implementing a read command. After the high voltage to low voltage transmission is finished, the data appears in the lower eight bits of the HVDAT MMR.
- Write. The process to perform a write is as follows:

    1. Place the data to be written to the HV die into the HVDAT MMR.
    2. Write the desired write command into the HVCON MMR.

Note that on an HV interrupt, the contents of the HVDSTA register are copied into the HVDAT register automatically.

Direct MMR Programming

This method uses a single register for writing to the HV die. HVDAT and HVCON registers are not necessary for writing because the top die HVDCFG0 register is shadowed on the low voltage die MMR map. Writing a value to the HVDCFG0 register on the low voltage die automatically triggers a write to the HVDCFG0 register on the high voltage die. The HVDCFG0 register on the low voltage MMR is then automatically updated to the readback value from the top die.

The HVDSTA register on the high voltage die is also shadowed on the low voltage MMR map. However, this shadowed register on the low voltage die is only updated on a high voltage interrupt or when the HVDSTA register on the top die is read back via the keyhole approach.

Note that a delay of 20 µs is required when programming the MMRs to ensure that data has been transferred correctly. To avoid this delay, the user can poll the register to check if the information has been written correctly.

Table 161. High Voltage MMR Mapping Approach Control Registers

Address	Name
0x40003010	HVDCFG0
0x40003018	HVDSTA

HIGH VOLTAGE MEMORY MAPPED REGISTERS

Table 162. High Voltage Control Interface Memory Mapped Registers (Base Address 0x40003000)

Offset	Name	Description	Access	Default
0x0000	RESERVED	Reserved		
0x0004	HVCON	Write commands register (used by keyhole method)	RW	0x00000000
0x0008	RESERVED	Reserved		
0x000C	HVDAT	Write data register (data to be transmitted, used by keyhole method)	RW	0x00000000
0x0010	HVDCFG0	Configuration register (used by direct MMR method)	RW	0x00
0x0018	HVDSTA	Status register (used by direct MMR method)	R	0x00

High Voltage Control Write Commands Register

Address: 0x40003004, Reset: 0x0000 0000, Name: HVCON

If using the keyhole approach, the HVCON and HVDAT registers are used together.

To write data to the HVDCFG0 register on the top die, first data is placed into the HVDAT register, and then 0x08 is written into the HVCON register.

To read data from the top die registers (HVDFCG0 or HVDSTA), 0x00 or 0x02 is placed in the HVCON register. A data read is then triggered from the top die, and the selected register contents are copied down into the HVDAT register. Care must be taken to observe the relevant busy, P_BIT, and ACK_BIT bits of the HVCON register.

The HVCON register accepts the following commands, as shown in Table 163, when written to.

Table 163. HVCON Commands

Group	Command	Description
Read Commands	0x00	Read back HV register HVDCFG0 into HVDAT
	0x02	Read back HV register STATUS into HVDAT
Write Commands	0x08	Write to HV register HVDCFG0 from HVDAT
	Other	Reserved

When read, the HVCON register returns a 3-bit value.

Table 164. HVCON Register Read Descriptions

Bits	Name	Description
[31:4]	RESERVED	Reserved.
3	MMRMODE_BUSY	0: no mirrored MMR write in progress. Previous mirrored MMR writes are completed. 1: mirrored MMR transaction in progress. Do not write to HV die using keyhole method when this bit is set to 1.
2	ACK_BIT	0: HV slave Tx NACK. 1: HV slave Tx ACK.
1	P_BIT	0: Rx parity failed. 1: Rx parity achieved.
0	BUSY	1: indicates communication in progress. Further commands must not be issued. 0: indicates communication not progress.

High Voltage Write Data Register

Address: 0x4000300C, Reset: 0x0000 0000, Name: HVDAT

HVDAT[7:0] stores the data read from or written to the top die (depending on the HVCON register).

Table 165. HVDAT Register Bit Descriptions

Bits	Name	Description
[31:12]	RESERVED	Reserved.
[11:8]	COM	Current command. Contains the last write to HVCON.
[7:0]	DATA	Value of shift register in the communications. When transmissions occur, data is stored here.

High Voltage Configuration Register

Address: 0x40003010, Reset: 0x00, Name: HVDCFG0

HVDCFG0 is a shadowed MMR of the HVDCFG0 register on the top die. Bit locations are identical.

Table 166. HVDCFG0 Register Bit Descriptions

Bits	Name	Description
[7:5]	RESERVED	Reserved.
4	LVFE	Low voltage flag enable. 0: disable the low voltage flag. 1: enable the low voltage flag.
3	VE	Voltage attenuator diagnostic enable. 0: disable the VBAT attenuator diagnostic function. 1: enable the VBAT attenuator diagnostic function. Switches in 1.29 µA current, which adds 160 mV of differential voltage to the voltage channel measurement.
2	IOE	Reenable after short circuit detection. In the event of a short on LIN, the bus is disabled. 0: automatically cleared by hardware after being set. 1: reenable the driver.
1	DLSCP	Disable LIN short circuit protection. 0: allow short circuit detection, and LIN driver automatically disabling. 1: disable the LIN short circuit protection functions. The LSCI bit in the status register still indicates that the short-circuit current limit has been exceeded for LIN. An interrupt is generated, which the user can choose to ignore. The drivers are not automatically shut down when this bit is set.
0	LIN_EN	LIN enable. 0: clear to 0 to disable the top die LIN transceiver. 1: set to 1 to enable the top die LIN transceiver.

High Voltage Status Register

Address: 0x40003018, Reset: 0x00, Name: HVDSTA

The HVDSTA register is a shadowed MMR of the HVDSTA register on the top die. Bit locations are identical.

Table 167. HVDSTA Register Bit Descriptions

Bits	Name	Description
[7:3]	RESERVED	Reserved.
2	LVF	Low voltage flag. 0: cleared to 0 automatically by hardware if the regulated supply drops below the specified limit. See the ADuCM330/ADuCM331 data sheet. 1: set to 1 if the supply voltage has not dropped below the specified limit. Valid only if HVDCFG0[4] is enabled.
1	LSCS	LIN short-circuit status. 0: cleared to 0 automatically by hardware. 1: set and cleared automatically by hardware due to a short circuit detected on LIN. Indicates status of LIN short-circuit detect.
0	LSCI	LIN short-circuit interrupt. 0: cleared to 0 automatically by hardware. 1: set automatically by hardware due to a short circuit detected on LIN, bus disabled.

LIN INTERFACE

LIN OVERVIEW

The ADuCM330/ADuCM331 implement a low overhead LIN interface that is compliant with LIN 1.3 and LIN 2.2/SAE J2602-2 specifications. The device operates as a slave only interface, operating from 1 kbps to 20 kbps. All lower rates are interpreted as 1 kbps.

The interface consists of a two die solution. As shown in Figure 32, an integrated LIN transceiver is present on the top die that communicates directly to the LIN logic on the bottom die. The logic uses three internal Cortex-M3 interrupts, LININT0, LININT1, and LININT2, to signify to the core that a LIN event has occurred.

LIN FEATURES

- Efficient LIN data handling with storage for up to eight data bytes
- Built in parity check for PID
- Framing error checking
- Running checksum (classic or enhanced) on received bytes
- Partial PID decoding to determine diagnostic master request frame
- Break symbol detection at any time
- Collision detection
- UART mode
- Automatic transceiver delay time compensation

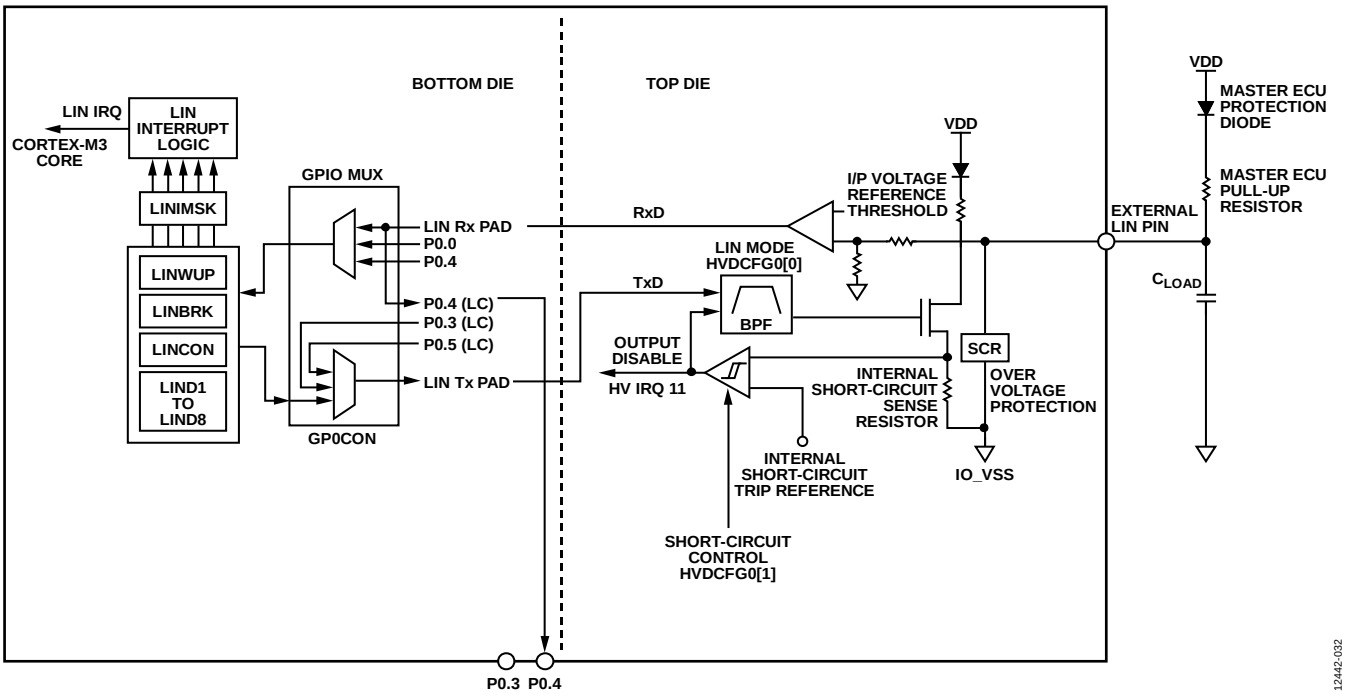

Figure 32. LIN Block Diagram

LIN USER OPERATION

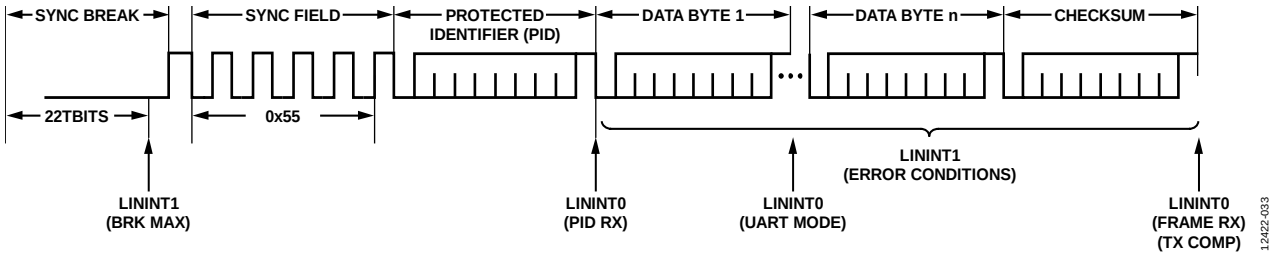

Figure 33. LIN Interrupts

LIN Transmission

A maskable software interrupt (LININT0) occurs on the reception of a PID. After identifying the PID, the user code determines if the device must respond, receive, or ignore the PID. If a transmission is required, the user code must do the following:

- Set the number of bytes to transmit in the LINCNT register.
- Place all of the data bytes to transmit into their individual LINDx registers.
- Set LINCON[1] to 1 to signify begin transmission.

When the number of data bytes transmitted is equal to the value held in LINCNT, the checksum (which is automatically calculated) is then transmitted. After the checksum is transmitted, the frame is considered complete, and the LININT0 interrupt is set again.

LIN Reception

The LIN interface automatically changes to receive mode on reception of a break symbol.

The LININT0 interrupt indicates that a PID has been received. If the software decides that full message reception is required, the expected number of bytes is placed in the LINCNT register, and the interface begins to receive all the data bytes from the master. When all expected bytes are received, the checksum calculation is started, and if it is correct, LININT0 is triggered again.

If the software determines that the PID does not require a response, the LINCNT register must be set to 0 indicating to the hardware to ignore the rest of the frame.

Sleep/Wake Functionality

The ADuCM330/ADuCM331 LIN interface has integrated LIN sleep/wake functionality. The LININT2 interrupt is used to notify the user that a wake or sleep event has occurred. When the LININT2 interrupt is triggered, the user must interrogate the LINSTA register to determine the source of the interrupt.

The two LININT2 interrupt sources are as follows:

1. The LIN interface has measured a period of inactivity on the LIN bus corresponding to the LINSLP register setting. The LINSLP register allows flexible configuring of the bus idle timeout time from 0.5 sec to 16 sec.

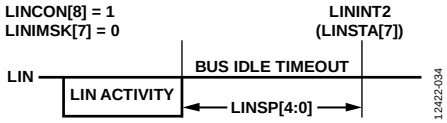

Figure 34. Sleep Interrupt

2. The LIN interface has received a LIN wake-up frame during power-down mode. A valid wakeup is a dominant state on the LIN bus for a time set by the LINWUP register, followed by a rising edge. The LIN bus must remain high for at least two 32 kHz clock periods for the LININT2 interrupt to be generated.

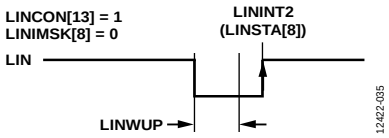

Figure 35. Wake Interrupt

LIN Diagnostic Frame Support

The ADuCM330/ADuCM331 LIN interface has intelligent support for diagnostic command frames (PID 0x3C) from a LIN master. In the event a 0x3C PID is sent from the LIN master and LINCON[14] is cleared to 0, the ADuCM330/ADuCM331 cannot generate a LININT0 PID interrupt. The ADuCM330/ADuCM331 then generates a LININT0 frame received interrupt only on the following conditions:

- Byte 0: contains NAD of the IBS slave.
- Byte 0: contains 0x00 (sleep frame).
- Byte 0: contains 0x7F (broadcast frame).

All other conditions cause the entire LIN diagnostic frame to be ignored, and no interrupt is generated on the LIN frame. This feature means there is minimum software intervention in servicing LIN diagnostic frames, and ensures that the ADuCM330/ADuCM331 only interrupts the user due to frames that are of relevance to the IBS.

This feature can be disabled by setting LINCON[14] to 1. This setting ensures that a header received interrupt occurs on all 0x3C PID headers that occur on the LIN bus.

Error Handling

The ADuCM330/ADuCM331 LIN interface is capable of handling multiple error events. The LINSTA register reflects the source of the error.

The following errors can produce an interrupt:

- Collision detection: in the event of a collision during transmission, the hardware automatically ceases transmission, and a maskable LININT1 interrupt is generated.
- Maximum negative edges: if the interface receives more than the maximum negative edges expected in a LIN frame, a maskable LININT1 interrupt occurs.
- PID parity error flag: a LIN PID has two parity bits. If the received PID does not correctly match the LIN2.2 parity scheme specification, this error is flagged. A new break symbol clears this flag.
- Framing error flag: in the event of a frame error, this error is flagged. A new break symbol clears this flag.
- Checksum error: if the received checksum does not match the calculated checksum, this error condition occurs.

LIN MEMORY MAPPED REGISTERS

Table 168. LIN Memory Mapped Registers (Base Address 0x40005000)

Offset	Name	Description	Access	Default
0x0000	LINCON	LIN control register	RW	0x1000
0x0004	LINIMSK	LIN interrupt mask register	RW	0x0000
0x0008	LINBR	Baud rate count register	RW	0x0326
0x000C	LINBRK	Break symbol count register	RW	0x0454
0x0014	LINSAMP	LIN sampling delay count register	RW	0x00
0x0018	LINFORCE	Forces LIN low while its value > 0	RW	0x0000
0x001C	LINWUP	Minimum low time needed to wake up the device	W	0x0007
0x0020	LINCNT	Byte count register for the entire LIN frame	RW	0x18
0x0024	LINTRDLY	Transceiver compensation delay time	RW	0x00
0x0028	LINSLP	Bus idle timeout register	RW	0x08
0x002C	LINLCNT	Number of bytes in the last response frame	R	0x0000
0x0030	LINSTA	LIN status register	RW	0x0000
0x0038	LINSTA1	LIN Status 1 register	R	0x0000
0x003C	LINID	PID received register	R	0x0000
0x0040	LIND1	LIN Data Byte 1 for Rx or Tx	RW	0x00
0x0044	LIND2	LIN Data Byte 2 for Rx or Tx	RW	0x00
0x0048	LIND3	LIN Data Byte 3 for Rx or Tx	RW	0x00
0x004C	LIND4	LIN Data Byte 4 for Rx or Tx	RW	0x00
0x0050	LIND5	LIN Data Byte 5 for Rx or Tx	RW	0x00
0x0054	LIND6	LIN Data Byte 6 for Rx or Tx	RW	0x00
0x0058	LIND7	LIN Data Byte 7 for Rx or Tx	RW	0x00
0x005C	LIND8	LIN Data Byte 8 for Rx or Tx	RW	0x00
0x0060	LINFCS	LIN frame checksum	R	0x00
0x0064	LINCCS	Current calculated checksum	R	0xFF
0x0068	LINNAD	Node ID address	RW	0xFF
0x006C	LININAD	Initial node ID	RW	0x00
0x0070	LINFID	Initial function ID	RW	0x0000
0x0074	LINVID	Variant ID	RW	0x0000
0x0078	LINSUPID	Supplier ID	RW	0x00
0x007C	LINSID	Service ID	RW	0x00

LIN Control Register

Address: 0x40005000, Reset: 0x1000, Name: LINCON

Table 169. LINCON Register Bit Descriptions

Bits	Name	Description
15	TRDLY	Transceiver delay compensation. This bit, in association with the LINTRDLY register, is used to adjust the sampling point during Tx mode. It is used to compensate for transceiver Tx to Rx delay time. 0: cleared to 0 to enable automatic tracking of transceiver delay time between Tx and Rx (default). The LINTRDLY register is automatically updated on the beginning of every byte bit measuring the falling edge transition of the start bit. 1: set to 1 to disable automatic delay compensation and enable manual programming of the LINTRDLY register. Software can write an appropriate value to compensate transceiver delay time.
14	PIDRXINT	PID received interrupt (PID = 0x3C) enable. 0: cleared to 0 to disable header received interrupt for diagnostic frame with PID as 0x3C (default). 1: set to 1 to enable header received interrupt for diagnostic frame with PID as 0x3C.
13	WUPEN	Wake-up enable. 0: cleared to 0 to enable wake-up detection only when the core is in hibernate mode (default). 1: set to 1 to enable wake-up detection in hibernate, active, and SYS_HALT power modes. If wake-ups are not required to generate an interrupt in active or SYSHALT mode, this bit must be cleared.
12	TXTIMEOUT	Tx dominant timeout check. The Tx signal is checked by a running counter. If the Tx signal is held low for more than 100 ms, this counter logic forces it high. 0: cleared to 0 to disable dominant timeout check on Tx signal. 1: set to 1 to enable dominant timeout check on Tx signal (default).
11	SYNC	Sync error check. The duration of the bits in the sync symbol are measured with respect to the duration of the start bit of the sync symbol. If the duration of successive bits in sync symbol are either more than twice or less than half then frame reception is aborted. 0: cleared to 0 to enable sync symbol error check. 1: set to 1 to disable sync symbol error check.
10	NAD	NAD match. 0: cleared to 0 to enable NAD match check. Diagnostic frame is only recognized if NAD matches. 1: set to 1 to disable NAD match check. Diagnostic frame is always recognized.
9	COLL	Collision check. This bit is used to disable collision check using the sampled RX line and TX data value. Collisions are checked only at the sampling instant in the response field if the direction bit, LINCON[1], is set. 0: cleared to 0 to enable collision check. This is the default state. 1: set to 1 to disable collision check.
8	SLEEPEN	Enable sleep counter. This bit enables the counter that monitors the LIN line for inactivity. The bus idle timeout period is determined by the LINSLP register. 0: cleared to 0 to disable sleep counter. 1: set to 1 to enable sleep counter.
7	CSCLR	Checksum clear. 1: this is a write only bit and when set to 1, clears the calculated checksum. This bit reads back a 0.
6	UARTEN	UART mode enable. This mode is used to transmit/receive data one byte at a time and is not LIN protocol compliant. LIND1 is the only register that participates in this mode. LINCON[1] must be cleared to allow receive of data bytes. 0: cleared to 0 for normal LIN protocol mode of operation. 1: set to 1 to allow transmission without receiving the frame header (UART mode).
5	BYPASSEN	LIN bypass enable. 0: cleared to 0 for LIN normal mode. 1: set to 1 allows the user to take control of the LIN output of the LIN block (to perform LIN conformance tests).
4	SYNCTIM	Timing of sync symbol bit 0 (not required in a single slave system). This bit ensures that if a second break is transmitted, it is recognized as such and not timed as part of the sync symbol. If the start symbol is more than the number of clock ticks dictated by this bit, the device assumes it is now receiving a break and continues to count the low cycle to verify if the break meets the minimum time required for a break, as defined in the LINBRK MMR. 0: cleared by user. The start bit of the sync symbol must be less than 8872 clock ticks. 1: set by user. The start bit of the sync symbol must be less than 1209 clock ticks.

Bits	Name	Description
3	LEN	Stop bit length. 0: if cleared, one stop bit is sent while transmitting. 1: if set, two stop bits are sent while transmitting.
2	CSCALC	Checksum calculation. This bit must be modified simultaneously with LINCON[0]. 0: cleared to 0 to calculate an enhanced checksum, PID included. 1: set to 1 to calculate automatically a classic checksum, PID excluded.
1	RXTXMODE	Receive/transmit mode. 0: cleared to 0 when a break symbol is received or when Tx is complete. Also cleared when 0 is written to LINCON[0]. 1: set to 1 to transmit data bytes after decoding the PID.
0	LINENABLE	LIN enable bit. 0: cleared to 0 by user code to disable the LIN interface or to reset the interface. 1: set to 1 by user code to enable the LIN interface. Note that clearing this bit resets all LIN interface registers to their default values, except for the following: LINSTA[4:0], LINSTA[13:11], LINCON[8:7], LINCON[13:12], and LINCON[15].

LIN Interrupt Mask Register

Address: 0x40005004, Reset: 0x0000, Name: LINIMSK

Table 170. LINIMSK Register Bit Descriptions

Bits	Name	Description
15	STOPSTART	Start/stop bit error interrupt mask. 0: cleared to 0 by user to enable the start/stop bit error detection. 1: set by user to disable the start/stop bit error detection.
14	SYNC	Sync field error interrupt mask. 0: cleared to 0 by user to enable the interrupt on sync field error detection on LININT1. 1: set by user to disable the interrupt sync field error detection on LININT1.
[13:12]	RESERVED	Reserved.
11	UARTMODE	UART mode interrupt mask. 0: cleared to 0 by user to enable the interrupt in UART mode on LININT0. 1: set to 1 by user to disable the interrupt in UART mode on LININT0.
10	MAXBRK	Maximum break time interrupt mask. 0: cleared to 0 by the user to enable the maximum break time interrupt on LININT1. 1: set to 1 by the user to disable the maximum break time interrupt on LININT1.
9	MAXNEGEDGE	Maximum negative edges in the frame interrupt mask. An interrupt is generated if more than 57 falling edges are detected in a frame. 0: cleared to 0 by the user to enable the maximum negative edges in the frame interrupt on LININT1. 1: set to 1 by the user to disable the maximum negative edges in the frame interrupt on LININT1.
8	WAKEUP	Wake-up interrupt mask. 0: cleared by the user to enable the wake-up interrupt on LININT2. 1: set to 1 by the user to disable the wake-up interrupt on LININT2.
7	SLEEP	Sleep interrupt mask. 0: cleared by the user to enable the bus idle timeout sleep interrupt on LININT2. 1: set to 1 by the user to disable the bus idle timeout sleep interrupt on LININT2.
6	FRAMEERR	Frame error interrupt mask. An interrupt is generated if framing error is detected. 0: cleared to 0 by the user to enable the framing error interrupt on LININT1. 1: set to 1 by the user to disable the framing error interrupt on LININT1.
5	COLLDETECT	Collision detect interrupt mask. 0: cleared to 0 by the user to enable the collision detect interrupt on LININT1. 1: set to 1 by the user to disable the collision detect interrupt on LININT1.
4	CSERR	Checksum error detected interrupt mask. 0: cleared to 0 by the user to enable the checksum error detected interrupt on LININT1. 1: set to 1 by the user to disable the checksum error detected interrupt on LININT1.

Bits	Name	Description
3	PIDPARITY	PID parity error interrupt mask. 0: cleared to 0 by user to enable the PID parity error interrupt on LININT1. 1: set to 1 by the user to disable the PID parity error interrupt on LININT1.
2	TXCOMP	Transmit complete interrupt mask. 0: cleared to 0 by the user to enable the transmit complete interrupt on LININT0. 1: set to 1 by the user to disable the transmit complete interrupt on LININT0.
1	FRAMERX	Frame received interrupt mask. An interrupt is generated when a complete frame has been received. 0: cleared to 0 by the user to enable the frame received interrupt on LININT0. 1: set to 1 by the user to disable the frame received interrupt on LININT0.
0	HEADERRX	Header received interrupt mask. 0: cleared to 0 by the user to enable the header received interrupt on LININT0. 1: set to 1 by the user to disable the header received interrupt on LININT0. Note that an interrupt is not generated for the master request frame (PID = 0x3C) if LINCON[14] is cleared.

LIN Baud Rate Count Register

Address: 0x40005008, Reset: 0x0326, Name: LINBR

Table 171. LINBR Register Bit Descriptions

Bits	Name	Description
[15:0]	BAUDRATE	Current baud rate value. User software must check this register to ensure that the baud rate is within expected limits. Baud rate = PCLK/LINBR.

LIN Break Symbol Count Register

Address: 0x4000500C, Reset: 0x0454, Name: LINBRK

Table 172. LINBRK Register Bit Descriptions

Bits	Name	Description
[15:0]	LINBRK	Contains the value after which the first break is considered valid. The value represents the time taken for 11 bits to be transmitted at 20 kbps (clock at 16 MHz). Break period = (LINBRK × 8 +7) × t_{PCLK}.

LIN Sampling Delay Count Register

Address: 0x40005014, Reset: 0x00, Name: LINSAMP

Table 173. LINSAMP Register Bit Descriptions

Bits	Name	Description
[7:0]	LINSAMP	Contains the count of clock ticks after which LIN input line is sampled. Three such samples are taken, and the majority value of these three samples decides the value of the bit. Sample spacing = (LINSAMP + 1) × t_{PCLK}.

LIN Force Low Register

Address: 0x40005018, Reset: 0x0000, Name: LINFORCE

Table 174. LINFORCE Register Bit Descriptions

Bits	Name	Description
[15:0]	LINFORCE	Low period = LINFORCE × 8 × t_{PCLK}. If LINCON[12] is set and a Tx timeout has already occurred, a break must be detected to allow the LINFORCE register to force the LIN line low again.

LIN Wake-Up Register

Address: 0x4000501C, Reset: 0x0007, Name: LINWUP

Table 175. LINWUP Register Bit Descriptions

Bits	Name	Description
[15:0]	LINWAKEUP	Wake-up period > (LINWUP) × t_{LFOSC}. This is the minimum low period to guarantee a wake-up.

LIN Byte Count Register

Address: 0x40005020, Reset: 0x18, Name: LINCNT

Table 176. LINCNT Register Bit Descriptions

Bits	Name	Description
[7:4]	COUNT	This field gives status information about the number of complete bytes sent/received. This number is updated at the end of the stop symbol of each byte.
[3:0]	NUMBYTES	Indicates the number of data bytes in the frame. Reset to 8 after the break is detected.

LIN Transceiver Delay Compensation Register

Address: 0x40005024, Reset: 0x00, Name: LINTRDLY

Table 177. LINTRDLY Register Bit Descriptions

Bits	Name	Description
[7:0]	TRDLY	This field can be manually configured by software or is filled automatically by hardware, depending on LINCON[15]. If LINCON[15] is cleared, this register is automatically programmed by hardware to compensate for the transceiver delay time between Tx and Rx (default). If LINCON[15] is set, manual programming of this register is required to compensate for the Tx to Rx delay time. Software must write an appropriate value to compensate for the transceiver delay time. Delay adjustment = (LINTRDLY + 1) × t_{PCLK}.

Bus Idle Timeout Register

Address: 0x40005028, Reset: 0x08, Name: LINSLP

Table 178. LINSLP Register Bit Descriptions

Bits	Name	Description
[4:0]	SLPVAL	This register defines the bus idle timeout period. When LINCON[8] is set, the hardware counts down from the timeout set by this register and generates a maskable interrupt, LININT2. Default value reflects a bus idle timeout of 4.096 sec. Bus idle period = LINSLP × 2^{14} × 31.25 µs.

LIN Last Byte Count Register

Address: 0x4000502C, Reset: 0x0000, Name: LINLCNT

Table 179. LINLCNT Register Bit Descriptions

Bits	Name	Description
[3:0]	NUMLBYTES	This register indicates the number of data bytes in the previous LIN frame for both Rx and Tx mode. Increment occurs for data bytes and CSUM byte. This register has a maximum value of 9, and the next counter iteration results in a value of 1. This register is not reset by hardware; software in the LININT0 PID interrupt is expected to write any value to clear this register.

LIN Status Register

Address: 0x40005030, Reset: 0x0000, Name: LINSTA

Note that data written to the LINSTA register requires two LFOSC clock periods before it is guaranteed to be latched.

Table 180. LINSTA Register Bit Descriptions

Bits	Name	Description
15	STOPSTART	Start/stop bit error 0: cleared to 0 by a break detect or by writing 1 to this bit location. This bit is also cleared when 0 is written to LINCON[0]. 1: set to 1 if the received signal start bit is sampled high or the stop bit is sampled low. This check is done both while receiving or transmitting.
14	SYNCERR	Sync field error status. Generates a maskable interrupt, LININT1. 0: cleared to 0 by a break detect or by writing 1 to this bit location. This bit is also cleared when 0 is written to LINCON[0]. 1: set to 1 when error is detected in the sync field. Bits are timed by the duration of the start bit. If bits are less than half or more than twice the duration of the start bit, the frame is aborted and this bit is set. This bit is also set if the overall duration of the sync field exceeds a maximum value.
13	COLL	Collisions bit value. This bit has meaning only if LINSTA[5] is set and indicates the status of Tx bit when collision occurred. This bit is cleared by writing 1 to this bit location.
12	UARTDATA	UART data byte received status. Generates a maskable interrupt, LININT0. 0: cleared to 0 by reading the LIND1 register or by writing 1 to this bit location. 1: set to 1 when a byte is received into the LIND1 register in UART receive mode.
11	UARTTX	UART transmit complete status. Generates a maskable interrupt, LININT0. 0: cleared to 0 on a write to the LIND1 register or by writing 1 to this bit location. This bit is cleared when 0 is written to LINCON[0]. 1: set to 1 when the contents of the LIND1 register is transmitted onto the LIN bus in UART transmit mode, and new data can be written to the LIND1 register.
10	BRKTIME	Break time maximum status. Generates a maskable interrupt, LININT1. 0: this bit is never cleared and is only valid on the first break symbol. This bit is reset if the first break symbol after enabling the LIN interface ends before the maximum count is reached, or the bit can be reset by writing 1 to this bit location. 1: set to 1 if the first break symbol after enabling the LIN interface is longer than 22 bits. This maximum break time is based on the contents of the LINBRK register and can change if the contents of the LINBRK register change. Frame reception continues if this bit is set.
9	MAXNEGEDGE	Negative edge maximum error status. Generates a maskable interrupt, LININT1. 0: cleared to 0 if the number of negative edges allowed in a frame is not surpassed. This bit is also cleared when 0 is written to LINCON[0]. 1: set to 1 if the number of negative edges is 57 or more. This bit is not required to be written by software and is cleared when a break is detected.
8	WAKESTA	Wake-up status. Generates a maskable interrupt, LININT2. 0: this bit is cleared when 0 is written to LINCON[0] or by writing 1 to this bit location. This bit must be cleared by software. 1: set to 1 when in sleep (power-down) mode if LIN line is detected to be low for a longer time than programmed by LINWUP followed by a rising edge.
7	SLEEPSTA	Sleep status. Generates a maskable interrupt on LININT2. 0: this bit is cleared when 0 is written to LINCON[0] or by writing 1 to this bit location. This bit must be cleared by software. 1: set to 1 if LIN bus remains at the same state for more than the value indicated by the LINSLP register.
6	FRMERR	Framing error status in receive mode. Generates a maskable interrupt, LININT1. 0: cleared to 0 when 0 is written to LINCON[0]. This bit is cleared by writing 1 to this bit location. 1: set to 1 if a valid start/stop is not detected in receive mode, or after receiving a sync, it is determined that the break was less than 11 Tbits long.
5	COLLDETECT	LIN collision detect status. Generates a maskable interrupt, LININT1. 0: cleared to 0 by break detect or when 0 is written to LINCON[0]. 1: set to 1 automatically by the hardware if the device has stopped transmission due to a collision on the bus. Write 1 via the software to clear this bit. This bit is only set on collision of data bits.

Bits	Name	Description
4	CSMATCH	Checksum match status. An error condition generates maskable interrupt, LININT1 0: cleared to 0 by break detect or by writing 1 to this bit location. 1: set to 1 when the checksum does not match. This is an error condition at the end of the frame, which generates an interrupt.
3	PIDPARITY	PID parity error status. Generates maskable interrupt, LININT1. 0: cleared to 0 by break detect or by writing 1 to this bit location. 1: set to 1 by hardware if the received PID byte does not correctly match the parity scheme for a PID, as described in the LIN2.2 specifications.
2	TXCOMP	Transmit complete. Generates a maskable interrupt, LININT0. 0: this bit is cleared by break detect or by writing 1 to this bit location. 1: this bit is set when the last bit of the checksum byte is transmitted after transmitting all data bytes.
1	FRAMERX	Frame received status. Generates maskable interrupt, LININT0. 0: cleared to 0 by break detect or by writing 1 to this bit location. 1: set to 1 if the full frame has been received. If PID = 0x3C, NAD matches (or broadcast address). For all frames, checksum must match.
0	HEADERRX	Header received status. Generates maskable interrupt, LININT0. 0: cleared to 0 by break detect or by writing 1 to this bit location. 1: set to 1 when a header has been received. For PID = 0x3C, this status bit setting also depends on LINCON[14]. PID parity check must also pass.

LIN Status 1 Register

Address: 0x40005038, Reset: 0x0000, Name: LINSTA1

Table 181. LINSTA1 Register Bit Descriptions

Bits	Name	Description
[15:1]	RESERVED	Reserved. These bits return 0 when read.
0	FRMINCOMP	Incomplete frame received. 0: cleared to 0 by writing 1 to this bit location. 1: set to 1 when a break is detected and the expected number of bytes in the previous frame (LINCNT[3:0] + 1) is not the same as the actual number of bytes received (LINCNT[7:4]). This bit is not set if LINCNT[3:0] is equal to zero. Note that the plus one is to account for the checksum byte. This status refers to the previous frame and is set for the first frame after enabling LIN. Note that this register is superseded by the LINLCNT register. It is recommended to use the LINLCNT register to determine the completion status of the previous frame.

LIN PID Received Register

Address: 0x4000503C, Reset: 0x0000, Name: LINPID

Table 182. LINPID Register Bit Descriptions

Bits	Name	Description
[15:0]	PIDRX	LIN protected ID (PID) received. Updated when the stop bit of the PID has been received. Parity bits are not saved but checked in hardware.

LIN Data Byte Registers

LIN Data Byte 1 for Rx or Tx

Address: 0x40005040, Reset: 0x00, Name: LIND1

Table 183. LIND1 Register Bit Descriptions

Bits	Name	Description
[7:0]	DATABYTE1	Data Byte 1 for receive or transmit. Also used in UART mode. See LINCON[6] in Table 169.

LIN Data Byte 2 for Rx or Tx

Address: 0x40005044, Reset: 0x00, Name: LIND2

Table 184. LIND2 Register Bit Descriptions

Bits	Name	Description
[7:0]	DATABYTE2	Data Byte 2 for receive or transmit

LIN Data Byte 3 for Rx or Tx

Address: 0x40005048, Reset: 0x00, Name: LIND3

Table 185. LIND3 Register Bit Descriptions

Bits	Name	Description
[7:0]	DATABYTE3	Data Byte 3 for receive or transmit

LIN Data Byte 4 for Rx or Tx

Address: 0x4000504C, Reset: 0x00, Name: LIND4

Table 186. LIND4 Register Bit Descriptions

Bits	Name	Description
[7:0]	DATABYTE4	Data Byte 4 for receive or transmit

LIN Data Byte 5 for Rx or Tx

Address: 0x40005050, Reset: 0x00, Name: LIND5

Table 187. LIND5 Register Bit Descriptions

Bits	Name	Description
[7:0]	DataByte5	Data Byte 5 for receive or transmit

LIN Data Byte 6 for Rx or Tx

Address: 0x40005054, Reset: 0x00, Name: LIND6

Table 188. LIND6 Register Bit Descriptions

Bits	Name	Description
[7:0]	DATABYTE6	Data Byte 6 for receive or transmit

LIN Data Byte 7 for Rx or Tx

Address: 0x40005058, Reset: 0x00, Name: LIND7

Table 189. LIND7 Register Bit Descriptions

Bits	Name	Description
[7:0]	DATABYTE7	Data Byte 7 for receive or transmit

LIN Data Byte 8 for Rx or Tx

Address: 0x4000505C, Reset: 0x00, Name: LIND8

Table 190. LIND8 Register Bit Descriptions

Bits	Name	Description
[7:0]	DATABYTE8	Data Byte 8 for receive or transmit

LIN Frame Checksum Register

Address: 0x40005060, Reset: 0x00, Name: LINFCS

Table 191. LINFCS Register Bit Descriptions

Bits	Name	Description
[7:0]	LINFCS	While receiving a frame, a received checksum byte is saved into this register.

LIN Current Calculated Checksum Register

Address: 0x40005064, Reset: 0xFF, Name: LINCCS

Table 192. LINCCS Register Bit Descriptions

Bits	Name	Description
[7:0]	LINCCS	Current calculated checksum value

LIN Node ID Address Register

Address: 0x40005068, Reset: 0xFF, Name: NODEID

Table 193. LINNAD Register Bit Descriptions

Bits	Name	Description
[7:0]	NODEID	Software is expected to update this register per the configuration diagnostic frame (assign node address, and so on). The configuration can also be initialized (read from NVM) and written into this register before the LIN block is enabled.

LIN Initial Node ID Register

Address: 0x4000506C, Reset: 0x00, Name: LININAD

Table 194. LININAD Register Bit Descriptions

Bits	Name	Description
[7:0]	INITNODEID	LIN initial node ID. Software is expected to generate the initial node address (for example, a read from NVM) and write the address into this register before the LIN block is enabled.

LIN Initial Function ID Register

Address: 0x40005070, Reset: 0x0000, Name: LINFID

Table 195. LINFID Register Bit Descriptions

Bits	Name	Description
[15:0]	FUNCID	LIN function ID. This register can be kept fixed, read only, or initialized like LININAD after reading from NVM.

LIN Variant ID Register

Address: 0x40005074, Reset: 0x0000, Name: LINVID

Table 196. LINVID Register Bit Descriptions

Bits	Name	Description
[15:0]	VARID	LIN variant ID. This register can be kept fixed, read only, or initialized like LININAD after reading from NVM.

LIN Supplier ID Register

Address: 0x40005078, Reset: 0x00, Name: LINSUPID

Table 197. LINSUPID Register Bit Descriptions

Bits	Name	Description
[7:0]	SUPID	LIN supplier ID. This register can be used to store supplier information.

LIN Service ID Register

Address: 0x4000507C, Reset: 0x00, Name: LINSID

Table 198. LINSID Register Bit Descriptions

Bits	Name	Description
[7:0]	SERVID	LIN service ID. This register is updated by the software when a diagnostic master request frame is received and is used to decide the response field of the diagnostic slave response frame.

DEVICE IDENTIFICATION

For traceability, device identification is available at power-up. This traceability information is contained across four registers: R4, FEEDATL, System Serial ID 0, and System Serial ID 1.

The assembly lot number is part of the branding on the package, as shown in Table 199. Note that these values are an example and may not be a final representation of the branding.

Table 199. Branding Example

Line	Branding
Line 1	ADuCM330/ADuCM331
Line 2	WDCPZ
Line 3	L6x[1] number date code
Line 4	Assembly lot number

[1] x is a number that changes with the kernel revision.

R4

After kernel execution, this 32-bit register, R4, holds the assembly lot ID. This information matches Line 4 branding on the unit. This information must be saved to a known SRAM location before user code is executed.

FEEDATL

After kernel execution, the FEEDATL MMR contains information that can be used to identify the ADuCM330/ADuCM331. This information must be saved to a known SRAM location before user code is executed.

Table 200. FEEDATL Register Bit Descriptions (Address 0x40018010)

Bits	Description
[31:16]	Identity of the ADuCM330/ADuCM331
[15:0]	Wafer lot fabrication ID. Same as SYSSER1[15:0].

SYSTEM SERIAL ID 0

At power-on, this 32-bit location holds the value of the original manufacturing lot number from which this specific ADuCM330/ADuCM331 device was manufactured (bottom die only). Used in conjunction with SYSSER1, this lot number allows the full manufacturing history of this device to be traced (bottom die only).

Table 201. System Serial ID 0 Register (Address 0x0002 07F0)

Offset	Name	Description	Access	Default
0x0000	SYSSER0	Holds the value of the original manufacturing lot number	R	0x0000 0000

Table 202. SYSSER0 Register Bit Descriptions

Bits	Description
[31:27]	Wafer number. The five bits read from this location give the wafer number (1 to 24) from the wafer fabrication lot ID (from which this device originated).
[26:22]	Wafer lot fabrication plant. The five bits read from this location reflect the manufacturing plant associated with this wafer lot.
[21:16]	Wafer lot fabrication ID. The six bits read from this location form part of the wafer lot fabrication.
[15:0]	Wafer lot fabrication ID. These 16 LSBs hold a 16-bit number to be interpreted as the wafer fabrication lot ID number.

SYSTEM SERIAL ID 1

At power-on, this 32-bit location holds the values of the part ID number, silicon mask revision number, and kernel revision number of the bottom die only, as detailed in Table 204.

Table 203. System Serial ID 0 Register (Address 0x0002 07F4)

Offset	Name	Description	Access	Default
0x0000	SYSSER1	Holds the values of the part ID number, silicon mask revision number, and kernel revision number	R	0x0000 0000 (updated by kernel at power-on)

Table 204. SYSSER1 Register Bit Descriptions

Bits	Description
[31:28]	Silicon mask revision ID. The four bits read from this nibble reflect the silicon mask ID number. Specifically, the hexadecimal value in this nibble must be decoded as the lower nibble, reflecting the ASCII characters in the range of A to O. For example, if Bits[31:28] = 0001 = 0x1, this value is interpreted as 41, which is ASCII Character A corresponding to Silicon Mask Revision A. If Bits[31:28] = 1011 = 0xB, the number is interpreted as 4B, which is ASCII Character K, corresponding to Silicon Mask Revision K. The allowable range for this value is 1 to 15, which is interpreted as 41 to 4F or ASCII Character A to Character O.
[27:20]	Kernel revision ID. This byte contains the hexadecimal number, which is interpreted as an ASCII character indicating the revision of the kernel firmware embedded in the on-chip Flash/EE memory. For example, reading 0x31 from this byte is interpreted as 1, indicating a Revision 1 kernel is on chip.
[19:16]	These bits refer to the kernel minor revision number of the device.
[15:0]	Part ID. These 16 LSBs hold a 16-bit number that is interpreted as the part ID number. When used in conjunction with the value in SYSSER0 (that is, the manufacturing lot ID), this number is a unique identifier for the device.

COMPLETE MMR LISTING

In the following MMR tables, addresses are listed in hexadecimal code. Access types include R for read, W for write, and RW for read and write.

Table 205. ADuCM330/ADuCM331 Memory Map Summary

Base Address	Peripheral
0x000207F0	System serial ID
0x40000000	General-Purpose Timer 0
0x40002000	SRAM
0x40002400	Interrupt detection
0x40002500	Wake-up timer
0x40002580	Watchdog timer
0x40003000	High voltage control interface
0x40004000	SPI
0x40005000	LIN
0x40005C00	HFOSC calibration
0x40006000	GPIO
0x40008824	Reference control
0x40009C00	LFOSC calibration
0x40018000	Flash controller
0x40030000	ADC
0xE000E004	Cortex-M3 and NVIC

Table 206. System Serial ID Memory Mapped Registers (Base Address 0x000207F0)

Offset	Name	Description	Access	Default
0x0000	SYSSER0	System Serial ID 0	R	0x0000 0000
0x0004	SYSSER1	System Serial ID 1	R	0x0000 0000

Table 207. SRAM Memory Mapped Registers (Base Address 0x40002000)

Offset	Name	Description	Access	Default
0x002C	SRAMCTRL	SRAM control register	RW	0x0001
0X0030	SRAMERR	Error location	R	0x0000

Table 208. General-Purpose Timer 0 Memory Mapped Registers (Base Address 0x40000000)

Offset	Name	Description	Access	Default
0x0000	T0VAL0	Current count value (LSB) register	R	0x0000
0x0004	T0VAL1	Current count value (MSB) register	R	0x0000
0x0008	T0CON	Control register	RW	0x0040
0x000C	T0EN	Timer enable register	RW	0x0000
0x0010	T0TOFA0	Timeout Field A (LSB) register	RW	0x1FFF
0x0014	T0TOFA1	Timeout Field A (MSB) register	RW	0x0000
0x0018	T0TOFB0	Timeout Field B (LSB) register	RW	0x2FFF
0x001C	T0TOFB1	Timeout Field B (MSB) register	RW	0x0000
0x0020	T0TOFC0	Timeout Field C (LSB) register	RW	0x3FFF
0x0024	T0TOFC1	Timeout Field C (MSB) register	RW	0x0000
0x0028	T0IEN	Interrupt enable register	RW	0x0000
0x002C	T0ISTA	Interrupt status register	R	0x0000
0x0030	T0CLRI	Clear interrupts register	W	Not applicable

Table 209. Clock Control Memory Mapped Register

Address	Name	Description	Access	Default
0x40002000	CLKCON	System clocks control register	RW	0x00E0

Table 210. Power Mode Memory Mapped Registers (Base Address 0x40002400)

Offset	Name	Description	Access	Default
0x0000	PWRMOD	Power mode register	RW	0x00
0x0004	PWRKEY	Power modes key register	RW	Not applicable

Table 211. Interrupt Detection Unit Memory Mapped Registers (Base Address 0x40002400)

Offset	Name	Description	Access	Default
0x0020	EI0CFG	External Interrupt Configuration Register 0	RW	0x0000
0x0030	EICLR	External interrupt clear register	RW	0x0000
0x0040	RSTSTA	Reset status register	R	Depends on type of reset
0x0040	RSTCLR	Reset clear register	W	Not applicable

Table 212. Wake-Up Timer Memory Mapped Registers (Base Address 0x40002500)

Offset	Name	Description	Access	Default
0x0000	T2VAL0	Current count value (LSB) register	R	0x0000
0x0004	T2VAL1	Current count value (MSB) register	R	0x0000
0x0008	T2CON	Control register	RW	0x0010
0x000C	T2EN	Enable register	RW	0x0000
0x0010	T2WUFA0	Wake-Up Field A (LSB) register	RW	0x1FFF
0x0014	T2WUFA1	Wake-Up Field A MSB) register	RW	0x0000
0x0018	T2WUFB0	Wake-Up Field B (LSB) register	RW	0x2FFF
0x001C	T2WUFB1	Wake-Up Field B (MSB) register	RW	0x0000
0x0020	T2WUFC0	Wake-Up Field C (LSB) register	RW	0x3FFF
0x0024	T2WUFC1	Wake-Up Field C (MSB) register	RW	0x0000
0x0028	T2IEN	Interrupt enable register	RW	Not applicable
0x002C	T2ISTA	Interrupt status register	R	0x0000
0x0030	T2CLRI	Clear interrupts register	W	0x0000
0x003C	T2CAP0	Capture event count (LSB) register	R	0x0000
0x0040	T2CAP1	Capture event count (MSB) register	R	0x0000

Table 213. Watchdog Timer Memory Mapped Registers (Base Address 0x40002580)

Offset	Name	Description	Access	Default
0x0000	T3LD	Load value register	RW	0x0400
0x0004	T3VAL	Current count value register	R	0x1000
0x0008	T3CON	Control register	RW	0x00E9
0x000C	T3CLRI	Clear interrupt register	W	Not applicable
0x0018	T3STA	Status register	R	0x0020

Table 214. High Voltage Control Interface Memory Mapped Registers (Base Address 0x40003000)

Offset	Name	Description	Access	Default
0x0000	RESERVED	Reserved	Not applicable	Not applicable
0x0004	HVCON	Write commands register (used by keyhole method)	RW	0x0000 0000
0x0008	RESERVED	Reserved	Not applicable	Not applicable
0x000C	HVDAT	Write data register (data to be transmitted, used by keyhole method)	RW	0x0000 0000
0x0010	HVDCFG0	Configuration register (used by direct MMR method)	RW	0x00
0x0018	HVDSTA	Status register (used by direct MMR method)	R	0x00

Table 215. SPI Peripheral Memory Address (Base Address 0x40004000)

Offset	Name	Description	Access	Default
0x0000	SPISTA	Status register	R	0x0000
0x0004	SPIRX	8-bit receive register	R	0x0000
0x0008	SPITX	8-bit transmit register	W	0x0000
0x000C	SPIDIV	8-bit baud rate selection register	RW	0x0000
0x0010	SPICON	16-bit configuration register	RW	0x0000

Table 216. LIN Memory Mapped Registers (Base Address 0x40005000)

Offset	Name	Description	Access	Default
0x0000	LINCON	LIN control register	RW	0x1000
0x0004	LINIMSK	LIN interrupt mask register	RW	0x0000
0x0008	LINBR	Baud rate count register	RW	0x0326
0x000C	LINBRK	Break symbol count register	RW	0x0454
0x0014	LINSAMP	LIN sampling delay count register	RW	0x00
0x0018	LINFORCE	Forces LIN low while its value > 0	RW	0x0000
0x001C	LINWUP	Minimum low time needed to wake up the device	W	0x0007
0x0020	LINCNT	Byte count register for the entire LIN frame	RW	0x18
0x0024	LINTRDLY	Transceiver compensation delay time	RW	0x00
0x0028	LINSLP	Bus idle timeout register	RW	0x08
0x002C	LINLCNT	Number of bytes in the last response frame	R	0x0000
0x0030	LINSTA	LIN status register	RW	0x0000
0x0038	LINSTA1	LIN Status 1 register	R	0x0000
0x003C	LINID	PID received register	R	0x0000
0x0040	LIND1	LIN Data Byte 1 for Rx or Tx	RW	0x00
0x0044	LIND2	LIN Data Byte 2 for Rx or Tx	RW	0x00
0x0048	LIND3	LIN Data Byte 3 for Rx or Tx	RW	0x00
0x004C	LIND4	LIN Data Byte 4 for Rx or Tx	RW	0x00
0x0050	LIND5	LIN Data Byte 5 for Rx or Tx	RW	0x00
0x0054	LIND6	LIN Data Byte 6 for Rx or Tx	RW	0x00
0x0058	LIND7	LIN Data Byte 7 for Rx or Tx	RW	0x00
0x005C	LIND8	LIN Data Byte 8 for Rx or Tx	RW	0x00
0x0060	LINFCS	LIN frame checksum	R	0x00
0x0064	LINCCS	Current calculated checksum	R	0xFF
0x0068	LINNAD	Node ID address	RW	0xFF
0x006C	LININAD	Initial node ID	RW	0x00
0x0070	LINFID	Initial function ID	RW	0x0000
0x0074	LINVID	Variant ID	RW	0x0000
0x0078	LINSUPID	Supplier ID	RW	0x00
0x007C	LINSID	Service ID	RW	0x00

Table 217. HFOSC Calibration Memory Mapped Registers (Base Address 0x40005C00)

Offset	Name	Description	Access	Default[1]
0x00	LINCALCON	LIN calibration control register	RW	0x0000
0x04	LINCALVAL0	User calibration value (low precision mode)	RW	0x0000
0x08	LINCALVAL1	User calibration value (high precision mode)	RW	0x0000
0x0C	LINCALMAXL	Maximum control window [15:0]	RW	0x0000
0x10	LINCALMAXH	Maximum control window [18:16]	RW	0x0000
0x14	LINCALMINL	Minimum control window [15:0]	RW	0x0000
0x18	LINCALMINH	Minimum control window [18:16]	RW	0x0000
0x1C	LINCALSTA	System calibration status	R	0x0000
0x20	LINCALVAL2	System calibration value(low precision mode)	R	0xXXXX
0x24	LINCALVAL3	System calibration value(high precision mode)	R	0xXXXX
0x28	LINCALOCK	Calibration lock register	RW	0x0000

[1] X means don't care.

Table 218. GPIO Interface Memory Address (Base Address 0x40006000)

Offset	Name	Description	Access	Default
0x0000	GP0CON	GPIO Port 0 configuration register	RW	0x0000
0x0004	GP0OEN	GPIO Port 0 output enable register	RW	0x00
0x0008	GP0PUL	GPIO Port 0 output pull-up enable register	RW	0x3F
0x000C	GP0OCE	GPIO Port 0 open circuit enable register	RW	0x00
0x0014	GP0INR	GPIO Port 0 input data register	R	XX[1]
0x0018	GP0OUT	GPIO Port 0 data out register	RW	0x00
0x001C	GP0SET	GPIO Port 0 data out set register	W	0x00
0x0020	GP0CLR	GPIO Port 0 data out clear register	W	0x00
0x0024	GP0TGL	GPIO Port 0 pin toggle	W	0x00

[1] Contents of the GP0INR register depend on the digital level on the corresponding pins.

Table 219. Reference Control Memory Address (Base Address 0x40008800)

Offset	Name	Description	Access	Default
0x0008	HRFCTRL	Internal 1.2 V reference control register	RW	0x0002
0x0024	IRFPD	Internal reference buffer power down register	RW	0x0001

Table 220. LFOSC Trim Memory Mapped Registers (Base Address 0x 0x40009C00)

Offset	Name	Description	Access	Default
0x0000	TRMSTA	Status register	R	0x00
0x0004	TRMCON	Control register	RW	0x00
0x0008	TRMMXC	Maximum calibration value register	R	0x3F
0x000C	TRMMNC	Minimum calibration value register	RW	0x00
0x0010	TRMVAL	Oscillator trim value	RW	Calibration value
0x0014	TRM32TGT	LFOSC target count	RW	0x0
0x0018	TRM32CNT	LFOSC current count	R	0x0
0x001C	TRMUCTGT	UCLK target count	RW	0x0000
0x0020	TRMUCTCNT	UCLK current count	R	0x0000

Table 221. Flash Controller Memory Mapped Registers (Base Address 0x40018000)

Offset	Name	Description	Access	Default
0x0000	FEESTA	Status register	R	0x0000 0000
0x0004	FEECON0	Command control register	RW	0x0010
0x0008	FEECMD	Command register	RW	0x0000
0x000C	FEEADR	Flash address keyhole register	RW	0x0000 0000
0x0010	FEEDATL	Flash data register (lower)	RW	0x0000 0000
0x0014	FEEDATH	Flash data register (upper)	RW	0x0000 0000
0x0018	FEEADR1L	Lower page address register	RW	0x0000 0000
0x001C	FEEADR1H	Upper page address register	RW	0x0000 0000
0x0020	FEEKEY	Key register	W	0x0000
0x0028	FEEPROP	Program flash write protection register	RW	0xFFFF FFFF
0x002C	FEEPROD	Data flash write protection register	RW	0xFF
0x0030	FEEECC	Data flash ECC disable register	RW	0x0000 0000
0x0034	FEESIGN	Signature	R	0x0000 0000
0x0038	FEECON1	Serial wire control register	RW	0x0001
0X0040	FEEABORT	Write abort address register	R	0x0000 0000
0x0048	FEEAEN0	Abort enable register	RW	0x0000
0x0068	USERFAKEY0	USERFAKEY low register [31:0]	RW	0x0000 0000
0x006C	USERFAKEY1	USERFAKEY high register [63:32]	RW	0x0000 0000
0x0074	FEEPECC	Program flash address for ECC error	R	0x0000 0000
0x0078	FEEDECC	Data flash address for ECC error	R	0x0000 0000

Table 222. ADC Memory Mapped Registers (Base Address 0x40030000)

Offset	Name	Description	Access	Default
0x0000	ADCSTA	ADC status register	R	0x0000
0x0004	ADCMSKI	ADC interrupt mask register	RW	0x00
0x0008	ADCMDE	ADC mode control register	RW	0x0003
0x000C	ADC0CON	Current ADC control register	RW	0x0000 0000
0x0010	ADC1CON	Voltage/temperature ADC control register	RW	0x0000 0000
0x0018	ADCFLT	ADC filter configuration register	RW	0x0000 0007
0x001C	ADCCFG	ADC configuration register	RW	0x00
0x0020	ADC0DAT	Current ADC result register	R	0x0000 0000
0x0024	ADC1DAT	Voltage/temperature ADC result register	R	0x0000 0000
0x0030	ADC0OF	Current ADC offset calibration register	RW	Calibration value
0x0034	ADC1OF	Voltage ADC offset calibration register	RW	Calibration value
0x0038	ADC2OF	Temperature ADC offset calibration register	RW	Calibration value
0x003C	ADC0GN	Current ADC gain calibration register	RW	Calibration value
0x0040	ADC1GN	Voltage ADC gain calibration register	RW	Calibration value
0x0044	ADC2GN	Temperature ADC gain calibration register	RW	Calibration value
0x0048	ADC0RCL	ADC result counter limit register	RW	0x0001
0x004C	ADC0RCV	ADC result counter value register	R	0x0000
0x0050	ADC0TH	Current ADC comparator threshold register	RW	0x0000 0000
0x0054	ADC0THC	ADC threshold counter limit register	RW	0x01
0x0058	ADC0THV	ADC threshold counter value register	R	0x00
0x005C	ADC0ACC	Current ADC accumulator register	R	0x0000 0000
0x0060	ADC0ATH	Current ADC accumulator threshold register	RW	0x0000 0000

Table 223. Cortex-M3 and NVIC Registers

Address	Analog Devices Header File Name	Description	Access
0xE000E004	ICTR	Shows the number of interrupt lines that the NVIC supports	R
0xE000E010	STCSR	SYSTICK control and status register	RW
0xE000E014	STRVR	SYSTICK reload value register	RW
0xE000E018	STCVR	SYSTICK current value register	RW
0xE000E01C	STCR	SYSTICK calibration value register	R
0xE000E100	ISER0	Set IRQ0 to IRQ13 enable	RW
0xE000E180	ICER0	Clear IRQ0 to IRQ13 enable	RW
0xE000E200	ISPR0	Set IRQ0 to IRQ13 pending	RW
0xE000E280	ICPR0	Clear IRQ0 to IRQ13 pending	RW
0xE000E300	IABR0	IRQ0 to IRQ13 active bits	RW
0xE000E400	IPR0	IRQ0 to IRQ3 priority	RW
0xE000E404	IPR1	IRQ4 to IRQ7 priority	RW
0xE000E408	IPR2	IRQ8 to IRQ11 priority	RW
0xE000E40C	IPR3	IRQ12 to IRQ13 priority	RW
0xE000ED00	CPUID	CPUID base register	R
0xE000ED04	ICSR	Interrupt control and status register	RW
0xE000ED08	VTOR	Vector table offset register	RW
0xE000ED0C	AIRCR	Application interrupt/reset control register	RW
0xE000ED10	SCR	System control register	RW
0xE000ED14	CCR	Configuration control register	RW
0xE000ED18	SHPR1	System Handlers Register 1	RW
0xE000ED1C	SHPR2	System Handlers Register 2	RW
0xE000ED20	SHPR3	System Handlers Register 3	RW
0xE000ED24	SHCSR	System handler control and state	RW
0xE000ED28	CFSR	Configurable fault status	RW
0xE000ED2C	HFSR	Hard fault status	RW
0xE000ED34	MMAR	Memory manage address	RW
0xE000ED38	BFAR	Bus fault address	RW
0xE000EF00	STIR	Software trigger interrupt register	W