ADuCM355 Hardware Reference Manual

SCOPE

This reference manual provides a detailed description of the functionality and features of the ADuCM355.

Full specifications on the ADuCM355 are available in the product data sheet. Consult the data sheet in conjunction with this reference manual when working with this device. Refer to the ADuCM355 data sheet for the functional block diagram.

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

REVISION HISTORY

6/2023—Rev. B to Rev. C

Changed Master to Initiator, Slave to Target, ac to AC, and dc to DC (Throughout) ... 1
Changes to Scope Section ... 1
Change to User Space Metadata Section ... 184
Changes to Reading Flash Section ... 185
Changes to Run-Time Configuration Section ... 188
Changes to Metadata Configuration Section ... 188
Changes to Write Protection Register Section and Table 234 ... 199

USING THE ADUCM355 HARDWARE REFERENCE MANUAL

NUMBER NOTATIONS

Table 1. Number Notations

Notation	Description
Bit N	Bits are numbered in little endian format, where the least significant bit of a number is referred to as Bit 0.
V[x:y]	A range from Bit x to Bit y of a value or a field (V) is represented in bit field format, V[x:y].
0xNN	Hexadecimal (Base 16) numbers are preceded by the 0x prefix.
0bNN	Binary (Base 2) numbers are preceded by the 0b prefix.

REGISTER ACCESS CONVENTIONS

Table 2. Register Access Conventions

Mode	Description
R/W	Memory location has read and write access.
RC	Memory location is cleared after reading the location.
R	Memory location is read access only. A read always returns 0, unless otherwise specified.
W	Memory location is write access only.
R/W1C	Memory location has read access. To clear to 0, write 1 once to the memory location.

Memory mapped register (MMR) bits that are not documented are reserved. When writing to MMRs with reserved bits, the reserved bits must be written with the value in the reset column of the relevant MMR description, unless otherwise specified.

Note that, throughout this hardware reference manual, multifunction pins, such as P0.0/SPI0_CLK, are referred to either by the entire pin name or by a single function of the pin, for example, P0.0, when only that function is relevant.

In header files, registers are grouped in stacks, such as CLKG0_CLK and AFECON. These stack names serve as pointers to each element in the associated structure and are used to access specific registers. For example, to access the STAT0 register, the user must enter the following:

```
pADI_CLKG0_CLK->STAT0
```

Not all registers have stacks, in which case the register name is not preceded by a stack name in this hardware reference manual.

ADUCM355 OVERVIEW

MAIN FEATURES OF THE ADUCM355

The analog-to-digital converter (ADC) includes the following features:

- 16-bit multichannel successive approximation register (SAR) ADC.
- Up to 34 input channels, programmable via input mux.
- Low noise transimpedance amplifiers (TIAs) for accurate current measurements.
- Low noise programmable gain amplifier (PGA) for accurate measurement of small input voltage signals.
- Internal 1.82 V and 2.5 V voltage reference sources.
- Hardware accelerators, including digital discrete Fourier transform (DFT) calculations.

The digital-to-analog converters (DACs) include the following features:

- Low power, low noise amplifiers designed for biasing external electrochemical sensors.
- Two dual output, low power DACs designed to set external sensor bias voltage and to support pulse and voltammetry electrochemical techniques.
- Programmable switch configuration to support interfacing to a variety of sensors and external pins.
- High bandwidth DAC and excitation amplifier designed to generate an excitation signal for impedance measurements up to 200 kHz.

The ADuCM355 contains the following communication features:

- Industry-standard, 16450/16550 universal asynchronous receiver/transmitter (UART) peripheral and support for direct memory access (DMA). Also supports wakeup from hibernate mode via the UART receive input.
- I^2C, 2-byte transmit and receive first in, first out (FIFOs) for the initiator and target, and support for DMA.
- Two serial peripheral interfaces (SPIs) with initiator or target mode, separate 4-byte receive and transmit FIFOs, and receive and transmit DMA channels.
- Multiple general-purpose input/output (GPIO) pins.

The processor of the ADuCM355 operates using the following:

- Arm® Cortex™-M3 processor, operating from an internal 26 MHz system clock.
- 128 kB Flash/EE memory, 64 kB static random-access memory (SRAM) on digital die.
- In circuit programming and debug via serial wire.

The on-chip peripherals are as follows:

- Three general-purpose timers on the digital die. There are two general-purpose timers on the analog die.
- Wake-up timer on the digital die. There is an optional wake-up timer also on the analog die.
- Independent watchdog timer on the analog die.

The package is 6 mm × 5 mm, 72-lead land grid array (LGA) package, and the temperature range is −40°C to +85°C. A low cost development system and a third party compiler and emulator tool support are included in the ADuCM355 evaluation kit.

Applications of the ADuCM355 include the following:

- Gas detection.
- Food quality.
- Environmental sensing (air, water, and soil).
- Blood glucose meters.
- Life sciences and biosensing analysis.
- Bioimpedance measurements.
- General amperometry, voltammetry, and impedance.

The memory organization of the device operates using the following:

- Arm Cortex-M3 memory system features include a predefined memory map, support for bit band operation for atomic operations, and unaligned data access.

ADUCM355 OVERVIEW

- On-chip peripherals are accessed via memory-mapped registers, situated in the bit band region.
- User memory size options are 128 kB Flash/EE memory and 64 kB SRAM.
- On-chip kernel for booting device with manufacturer data.

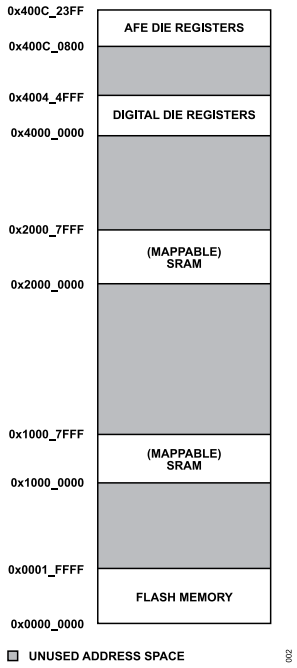

Figure 1. Arm Cortex-M3 Memory Map Diagram

CLOCKING ARCHITECTURE

CLOCKING ARCHITECTURE OPERATION

The ADuCM355 contains two internal dice. Therefore, there are two independent clock systems: a digital die clock system and an analog die clock system. Figure 2 shows the overall clock architecture.

REQUIRED CLOCK RATIO BETWEEN DIGITAL DIE AND ANALOG DIE SYSTEM CLOCKS

To maintain reliable communications between the digital die and the analog die, the ratio of the digital die system clock frequency to the analog die system clock frequency must be within the range of 3:1 and 1:3. For example, if the digital die system clock is set to 6.5 MHz, the analog die system clock must be >2.2 MHz but <19 MHz. If this ratio is not maintained, the digital die can lose its communication link to the analog die.

DIGITAL DIE CLOCK FEATURES

At power-up, the processor executes from the internal 26 MHz oscillator, with the oscillator output divided by four to give a clock to the central processing unit (CPU) of 6.5 MHz. User code can select the clock source for the digital die system clock and can divide the clock by a factor of 1 to 32, where the clock divider bits are controlled by Bits[5:0] in the CTL1 register, which allows slower code execution and reduced power consumption.

When switching clock sources, a stable clock must always be connected to the core. Otherwise, the system can halt before connecting to the new clock. The digital die clocks include the following:

- The low frequency oscillator is a 32 kHz internal oscillator.
- The high frequency oscillator is a 26 MHz internal oscillator.
- External, 16 MHz and 32 MHz crystal options, routed through the analog die.
- External clock input option, routed through the analog die.
- The root clock is divided into several internal clocks.
- The reference clock (RCLK) clocks the reference timer in the flash controller. The RCLK controls the time for flash erase and write operations. By default, RCLK is always connected to a 13 MHz clock source. The clock source is generated by a ½ divider connected to the 26 MHz high frequency oscillator. Therefore, the default values of the flash timer registers correspond to a 13 MHz clock.
- The high-power buck regulator clocks the high-power buck module. When the high-power buck regulator is enabled, this clock source is always 200 kHz. The high-power buck regulator is enabled and disabled by the CTL1 register in the power management unit (PMU).

ANALOG DIE CLOCK FEATURES

At power-up, the internal high frequency oscillator is selected as the analog front end (AFE) system clock with a 16 MHz setting. User code can divide the clock by a factor of 1 to 32, where the clock divider bits are controlled by CLKCON0, Bits[5:0], which allows reduced power consumption.

The system performance of the analog die has only been validated with the system clock = 16 MHz. The analog die clocks include the following:

- AFE low frequency oscillator is a 32 kHz internal oscillator used for the analog die watchdog timer.
- AFE high frequency oscillator is a 16 MHz or 32 MHz internal oscillator. The 32 MHz setting is only intended for clocking the ADC when measuring signals >80 kHz, especially for high frequency impedance measurements. If the 32 MHz setting is used, ensure that CLKCON0 Bits[5:0] = 2 to limit the digital die clock sources to 16 MHz. To select the 32 MHz oscillator option, use the following sequence:

```
pADI_AFECON->CLKEN1 |= BITM_AFECON_CLKEN1_ACLKDIS;  // Temporarily disable ACLK
pADI_AFE->HPOSCCON &= (~0x4);  // Clear HPOSCCON[2] = 0 to select 32 MHz output
pADI_AFECON->CLKEN1 &= (~BITM_AFECON_CLKEN1_ACLKDIS); // Re-enable ACLK
```

  The high-power oscillator configuration register is key protected, and the temporary disabling of the analog clock (ACLK) ensures a safe switchover of the high frequency oscillator from 16 MHz to 32 MHz or from 32 MHz to 16 MHz.
- External 16 MHz and 32 MHz crystal option. If the 32 MHz crystal is used, ensure that CLKCON0, Bits[5:0] = 2 to limit the digital die clock sources to 16 MHz.
- External clock input option. If the 32 MHz crystal is used, ensure that CLKCON0, Bits[5:0] = 2 to limit the digital die clock sources to 16 MHz.
- When using the 32 MHz oscillator on the analog die, the PMBW register in AFECON must be configured for high-power mode.
- Note, the ADC clock cannot be divided. It runs at the same speed as the high frequency oscillator selected in CLKSEL, Bits[3:2].

CLOCKING ARCHITECTURE

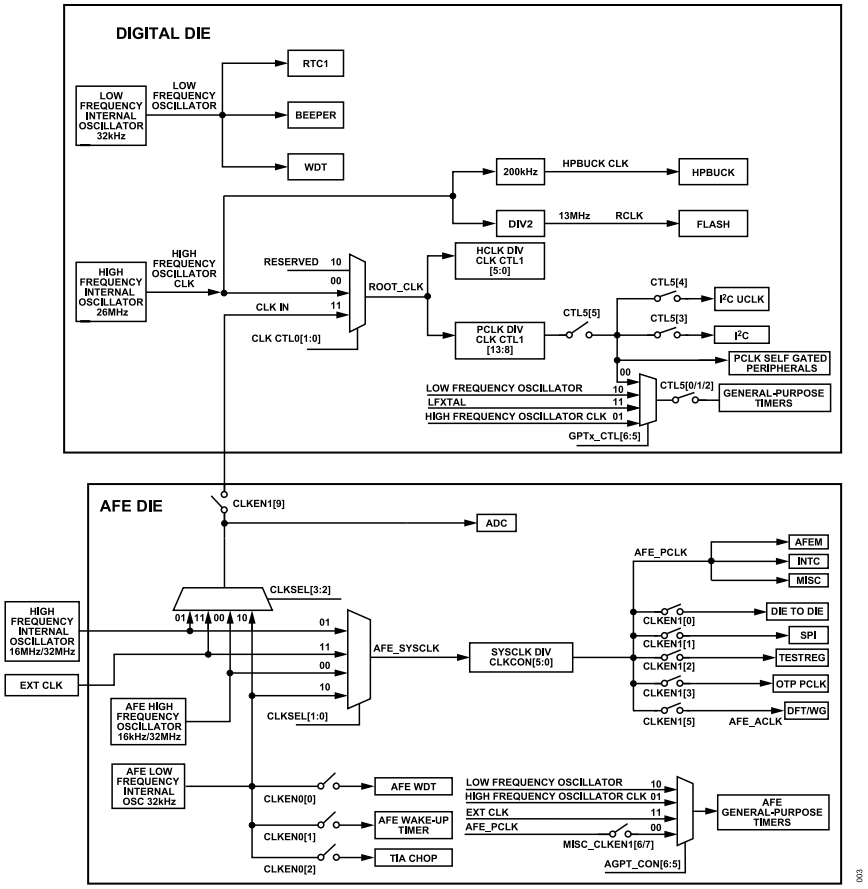

Figure 2. Clock Architecture Block Diagram

CLOCK GATING

In the case of certain clocks, clocks can be individually gated depending on the power mode or register settings. For more information about clock gating and power modes, refer to the Power Management Unit section.

On the digital die, the clock gates of the peripheral clocks are user-controllable in certain power modes. Register CTL5 in CLKG0_CLK can be programmed to turn off certain clocks, depending on the user application. Set the appropriate bits in the CTL5 register to 1 to disable the clock to individual blocks.

On the analog die, use the CLKEN0 register and the CLKEN1 register to disable the system clock to different peripherals on the analog die.

CONNECTING AFE DIE CLOCK TO DIGITAL DIE CLOCK INPUT

The AFE die 16 MHz oscillator is a more accurate oscillator than the 26 MHz high frequency oscillator on the digital die. For UART communications, select the AFE die 16 MHz oscillator as the input clock to the digital die. Internally, the AFE system clock can be connected to an internal pad, P2.2, on the AFE. There is an internal bond wire connecting this AFE die pad to the digital die pad, P1.10, on the digital die that can be configured as the external clock input for the digital die.

To connect and select the AFE die 16 MHz oscillator as the external clock input for the digital die, perform the following steps:

1. Enable AFE die Pad P2.2 as an output.

```
pADI_AGPIO2->OEN |= 0x4;
```

CLOCKING ARCHITECTURE

2. Configure the internal digital die Pad P1.10 as an input and configure its mode as EXT_CLKIN.

```
DioCfgPin(pADI_GPIO1,PIN10,2);              // External Clock mode for Digital die P1.10
DioIenPin(pADI_GPIO1,PIN10,1);              // Enable p1.10 input path
```

3. Clear CLKEN1, Bits[9:8]. The user is required to close the switch on the AFE die to connect the AFE die clock to the P2.2 pad.

```
pADI_AFECON->CLKEN1 &= 0x0FF;                           // Clear CLKEN1 bits 9:8
```

4. Select the digital die clock source as the external clock from the AFE die.

```
DigClkSel(DIGCLK_SOURCE_AFE);
```

If the clock source is a 32 MHz external crystal, ensure the clock to the digital die is 16 MHz by setting CLKCON[9:6] = 2.

Hibernate Mode and AFE Die Clock Selected on Digital Die

Switch the digital die clock source back to a digital die clock before entering hibernate mode. The device does not wake up if both dice are in hibernate mode and the AFE die clock is used by both dice, because, on waking from hibernate mode, the digital die wakes first. The digital die then must read or write to an AFE die register to wake the AFE die. If the AFE die is the clock source to both dice, the wake-up sequence does not complete.

To enter hibernate mode, use the following suggested sequence:

```
DigClkSel(DIGCLK_SOURCE_HFOSC);       // Switch digital die clock back to its own oscillator
pADI_UART0->COMDIV = 0;               // Clear COMDIV to ensure UART operates after wake-up
EnterHibernateMode();                 // Enter Hibernate mode
```

To exit hibernate mode to switch back to using only the AFE clock, use the following suggested sequence:

```
uiDummyRead =  pADI_AFE->LPDACCON0;   // Dummy read to wake-up AFE die
delay_10us(2000);                     // Wait 20mS
DigClkSel(DIGCLK_SOURCE_AFE);
UrtCfg(pADI_UART0,B57600,             // Re-Initialize the UART
  (BITM_UART_COMLCR_WLS|3),0);        // Configure UART for 57600 baud rate
printf("Digital Die clocked by AFE die's 16MHz oscillator \r\n");
```

REGISTER SUMMARY: CLOCK ARCHITECTURE

Table 3. Digital Die System Clock Register Summary (CLKG0_CLK Stack)

Address	Name	Description	Reset	Access
0x4004C10C	KEY	Key protection for CTL register	0x00000000	R/W
0x4004C110	CTL	Oscillator control	0x00000302	R/W
0x4004C300	CTL0	Clock Control 0	0x00000078	R/W
0x4004C304	CTL1	Clock dividers	0x00100404	R/W
0x4004C314	CTL5	User clock gating control	0x0000001F	R/W
0x4004C318	STAT0	Clocking status	0x00000000	R/W

Table 4. Analog Die Clock Register Summary (AFECON Stack)

Address	Name	Description	Reset	Access
0x400C0408	CLKCON0	Clock divider configuration	0x0441	R/W
0x400C0410	CLKEN1	Clock gate enable	0x010A	R/W
0x400C0414	CLKSEL	Clock select	0x0000	R/W
0x400C041C	GPIOCLKMUXSEL	GPIO clock mux select to GPIO1 pin	0x0000	R/W
0x400C0420	CLKCON0KEY	Key protection for CLKCON0	0x0000	R/W
0x400C0A70	CLKEN0	Clock control of low-power TIA chop, watchdog, and wake-up timers	0x0000	R/W
0x400C0A0C	OSCKEY	Key protection for OSCCON	0x0000	R/W
0x400C0A10	OSCCON	Oscillator control	0x0303	R/W
0x400C20BC	HPOSCCON	High-power oscillator configuration	0x00000024	R/W
0x400C22F0	PMBW	Power mode configuration register	0x00000000	R/W

REGISTER DETAILS: CLOCK ARCHITECTURE

KEY PROTECTION FOR CTL REGISTER

Address: 0x4004C10C, Reset: 0x00000000, Name: KEY

Table 5. Bit Descriptions for KEY

Bits	Bit Name	Settings	Description	Reset	Access
[31:16]	Reserved		Reserved.	0x0000	R
[15:0]	VALUE	0xCB14	Oscillator Key. The CTL register is key protected. To unlock this protection, write 0xCB14 to KEY before writing to CTL. A write to any other register on the Arm peripheral bus before writing to CTL returns the protection to the lock state.	0x0000	W

OSCILLATOR CONTROL REGISTER

Address: 0x4004C110, Reset: 0x00000302, Name: CTL

The CTL register is key protected. To unlock this protection, write 0xCB14 to KEY before writing to the CTL register. A write to any other register on the Arm peripheral bus before writing to the CTL register returns the protection to the lock state.

Table 6. Bit Descriptions for CTL

Bits	Bit Name	Settings	Description	Reset	Access
[31:10]	Reserved		Reserved.	0x0	R
9	HFOSCOK		Status of High Frequency Oscillator on Digital Die. This bit indicates when the oscillator is stable after it is enabled. This bit is not a monitor and does not indicate a subsequent loss of stability.	0x1	R
		0	Oscillator is not yet stable or is disabled.		
		1	Oscillator is enabled, stable, and ready for use.		
8	LFOSCOK		Status of Low Frequency Oscillator on Digital Die. This bit indicates when the oscillator is stable after it is enabled. This bit is not a monitor and does not indicate a subsequent loss of stability.	0x1	R
		0	Oscillator is not yet stable or is disabled.		
		1	Oscillator is enabled, stable, and ready for use.		
[7:2]	Reserved		Reserved.	0x0	R
1	HFOSCEN		High Frequency Internal Oscillator Enable. This bit is used to enable and disable the high frequency oscillator on the digital die. The oscillator must be stable before use. This bit must be set before the SYSRESETREQ system reset can be initiated. SYSRESETREQ is a bit in the Arm Cortex-M3 AIRCR register.	0x1	R/W
		0	The high frequency oscillator is disabled and placed in a low-power state.		
		1	The high frequency oscillator is enabled.		
0	Reserved		Reserved. This bit must always be 0.	0x0	R/W

CLOCK CONTROL 0 REGISTER

Address: 0x4004C300, Reset: 0x00000078, Name: CTL0

The Clock Control 0 register is used to configure clock sources used by various systems, such as the core, memories, and peripherals. All unused bits are read only and return a value of 0. Writing unused bits has no effect.

Table 7. Bit Descriptions for CTL0

Bits	Bit Name	Settings	Description	Reset	Access
[31:2]	Reserved		Reserved.	0x0000000	R
[1:0]	CLKMUX		Clock Mux Select. Determines which single shared clock source is used by the gated system clock (peripheral clock (PCLK) on digital die) and high-speed clock (HCLK) dividers.	0x0	R/W
		00	High frequency internal oscillator is selected.		
		01, 10	Reserved.		
		11	External clock routed from analog die. Can be external crystal or clock source.		

REGISTER DETAILS: CLOCK ARCHITECTURE

CLOCK DIVIDERS REGISTER

Address: 0x4004C304, Reset: 0x00100404, Name: CTL1

The clock dividers register is used to set the divide rates for the HCLK, PCLK, and ACLK dividers. This register can be written to at any time. All unused bits are read only, returning a value of 0. Writing to unused bits has no effect.

Table 8. Bit Descriptions for CTL1

Bits	Bit Name	Settings	Description	Reset	Access
[31:24]	Reserved		Reserved.	0x0	R
[23:16]	ACLKDIVCNT		ACLK Clock Divider. This bit determines the ACLK rate based on the equation ACLK = root clock/ACLKDIVCNT. For example, if the root clock is 26 MHz and ACLKDIVCNT = 0x1, ACLK operates at 26 MHz. The value of ACLKDIVCNT takes effect after a write access to this register and typically takes one ACLK cycle. This register can be read at any time and can be written to at any time. The value range is from 1 to 32. Values larger than 32 are saturated to 32. Value 0 and Value 1 have the same results as divided by 1. The default value of this register is configured such that ACLK = 1.625 MHz.	0x10	R/W
[15:14]	Reserved		Reserved.	0x0	R
[13:8]	PCLKDIVCNT		PCLK Clock Divider. Determines the PCLK rate based on the equation PCLK = root clock/ PCLKDIVCNT. For example, if the root clock is 26 MHz and PCLKDIVCNT = 0x2, PCLK operates at 13 MHz. The value of PCLKDIVCNT takes effect after a write access to this register and typically takes 2 to 4 PCLK cycles. This register can be read at any time and can be written to at any time. The value range is from 1 to 32. Values larger than 32 are saturated to 32. Value 0 and Value 1 have the same results as divided by 1. The default value of this register is configured such that PCLK frequency = 6.5 MHz. It is recommended to only use the 0x1, 0x2, or 0x4 value and matching PCLKDIVCNT with HCLKDIVCNT.	0x4	R/W
[7:6]	Reserved		Reserved.	0x0	R
[5:0]	HCLKDIVCNT		HCLK Divide Count. Determines the HCLK rate based on the equation: HCLK = root clock/ HCLKDIVCNT. For example, if the root clock is 26 MHz and HCLKDIVCNT = 0x1, HCLK operates at 26 MHz. The value of HCLKDIVCNT takes effect after a write access to this register and typically takes 2 to 4 PCLK cycles (not HCLK cycles). This register can be read at any time and can be written to at any time. The value range is from 1 to 32. Values larger than 32 are saturated to 32. Value 0 and Value 1 have the same results as divided by 1. The default value of this register is configured such that HCLK = 6.5 MHz. It is recommended to only use the values 0x1, 0x2, or 0x4 and match HCLKDIVCNT with PCLKDIVCNT.	0x4	R/W

USER CLOCK GATING CONTROL REGISTER

Address: 0x4004C314, Reset: 0x0000001F, Name: CTL5

User clock gating control is used to control the clock gating to peripherals.

Table 9. Bit Descriptions for CTL5

Bits	Bit Name	Settings	Description	Reset	Access
[31:6]	Reserved		Reserved.	0x0	R
5	PERCLKOFF		Peripheral Clocks Off. This bit is used to disable all clocks connected to all peripherals. After setting this bit, any read or write to any of the peripheral registers automatically resets PERCLKOFF to 0, and that read or write transaction is honored. After setting PERCLKOFF = 1, if the user reads the CTL5 register, PERCLKOFF is automatically cleared and PERCLKOFF reads as 0. The user must ensure that DMA transactions are done and no more transactions are expected from the DMA. Ensure that the PERCLKOFF bit write is the last write and no writes or reads to any of peripheral registers are performed after setting this bit. Otherwise, the PERCLKOFF bit clears.	0x0	R/W
		0	Clocks to all peripherals are active.		
		1	Clocks to all peripherals are gated off.		
4	GPIOCLKOFF		GPIO Clock Control. This bit disables the GPIO clock and controls the gate on the ACLK out from ACLK divider. This ACLK control is available in active mode and Flexi™ mode. In hibernate mode, the ACLK is always off and this bit has no effect. This bit does not automatically clear. Explicitly enable or disable this bit to control ACLK out. Before programming the ACLKDIVCNT bits in the CTL1 register, clear this bit to 0. Otherwise, the ACLKDIVCNT bit is not taken into effect.	0x1	R/W

REGISTER DETAILS: CLOCK ARCHITECTURE

Table 9. Bit Descriptions for CTL5 (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		0	GPIO clock is enabled.		
		1	GPIO clock is disabled.		
3	UCLKI2COFF		I^2C Clock User Control. This bit disables the I^2C universal clock (UCLK) and controls the gate on the I^2C UCLK in active and flexi modes. In hibernate mode, the I^2C UCLK is always off and this bit has no effect. This bit is automatically cleared if the user code accesses any of the I^2C registers.	0x1	R/W
		0	I^2C clock is enabled.		
		1	I^2C clock is disabled.		
2	GPTCLK2OFF		General-Purpose Timer 2 User Control. This bit disables the General-Purpose Timer 2 clock (muxed version) and controls the gate in active and flexi modes. In hibernate mode, the General-Purpose Timer 2 clock is always off and this bit has no effect. This bit is automatically cleared if user code accesses any of the General-Purpose Timer 2 registers.	0x1	R/W
		0	Timer 2 clock is enabled.		
		1	Timer 2 clock is disabled.		
1	GPTCLK1OFF		General-Purpose Timer 1 User Control. This bit disables the General-Purpose Timer 1 clock (muxed version) and controls the gate in active and flexi modes. In hibernate mode, the General-Purpose Timer 1 clock is always off and this bit has no effect. This bit is automatically cleared if user code accesses any of the General-Purpose Timer 1 registers.	0x1	R/W
		0	Timer 1 clock is enabled.		
		1	Timer 1 clock is disabled.		
0	GPTCLK0OFF		General-Purpose Timer 0 User Control. This bit disables the General-Purpose Timer 0 clock (muxed version) and controls the gate in active and flexi modes. In hibernate mode, the General-Purpose Timer 0 clock is always off and this bit has no effect. This bit is automatically cleared if user code accesses any of the General-Purpose Timer 0 registers.	0x1	R/W
		0	Timer 0 clock is enabled.		
		1	Timer 0 clock is disabled.		

CLOCKING STATUS REGISTER

Address: 0x4004C318, Reset: 0x00000000, Name: STAT0

Table 10. Bit Descriptions for STAT0

Bits	Bit Name	Settings	Description	Reset	Access
[15:3]	Reserved		Reserved. Do not write to this bit.	0	R
2	SPLLUNLK		System Phase-Locked Loop (PLL) Unlock Status. Write a 1 to this bit to clear it.		R/W1C
		0	No loss of PLL lock detected.	0	
		1	A PLL loss of lock is detected.		
1	SPLLLK		System PLL Lock Status. Write a 1 to this bit to clear it.	0	R/W1C
		0	No PLL lock event detected.		
		1	A PLL lock event is detected.		
0	SPLL		System PLL Status.	0	R
		0	PLL is not locked, do not use PLL.		
		1	PLL is locked and ready for use.		

CLOCK DIVIDER CONFIGURATION REGISTER

Address: 0x400C0408, Reset: 0x0441, Name: CLKCON0

User must write CLKCON0KEY = 0xA815 before writing to CLKCON0.

Table 11. Bit Descriptions for CLKCON0

Bits	Bit Name	Settings	Description	Reset	Access
[15:6]	Reserved		Reserved. Do not write to this bit.	0x1	R/W
[5:0]	SYSCLKDIV		System Clock Divider Configuration. The system clock divider is used to provide a divided clock from the root clock, which drives the peripheral bus, die to die interface, and most digital	0x1	R/W

REGISTER DETAILS: CLOCK ARCHITECTURE

Table 11. Bit Descriptions for CLKCON0 (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
			peripherals. System clock frequency (f_{SYS}) = root clock/SYSCLKDIV. Value range is from 1 to 32. Values larger than 32 are saturated to 32. Value 0 and Value 1 have the same results as divide by 1. f_{SYS} must be ≤16 MHz. Characterization is completed only with analog die system clock of 4 MHz, 8 MHz, and 16 MHz.		

CLOCK GATE ENABLE REGISTER

Address: 0x400C0410, Reset: 0x010A, Name: CLKEN1

Table 12. Bit Descriptions for CLKEN1

Bits	Bit Name	Settings	Description	Reset	Access
[15:10]	Reserved		Reserved.	0x0	R
9	AFECLKDIS		AFE Die Clock Enable to AFE P2.2 Pad.	0x0	R/W
		0	Connect AFE clock to AFE P2.2 pad.		
		1	Disconnect AFE clock from AFE P2.2 pad.		
8	AFECLKSTA		Reflects Status of CLKEN1 Bit, Read Only.	0x0	R
		0	AFE clock connected to AFE die P2.2 pad.		
		1	AFE clock disconnected from AFE die P2.2 pad.		
7	GPT1DIS		General-Purpose Timer 1 (GPT1) Clock Enable. This bit controls pulse width modulation (PWM) Timer 1 clocks.	0x1	R/W
		0	Turn on GTP1 clock.		
		1	Turn off GPT1 clock.		
6	GPT0DIS		General-Purpose Timer 0 (GPT0) Clock Enable. This bit controls PWM Timer 0 clocks.	0x1	R/W
		0	Turn on GPT0 clock.		
		1	Turn off GPT0 clock.		
5	ACLKDIS		ACLK Clock Enable. This bit controls the clock to the DFT and the waveform generator blocks control clock, including analog interface and digital signal processing.	0x0	R/W
		0	Turn on ACLK clock.		
		1	Turn off ACLK clock.		
4	Reserved		Reserved. Never write to this bit. Leave this bit cleared to 0.	0x0	R/W
3	Reserved		Reserved. Never write to this bit.	0x0	R/W
2	Reserved		Reserved. Never write to this bit. Leave this bit cleared to 0.	0x0	R/W
1	Reserved		Reserved. Never write to this bit.	0x0	R/W
0	Reserved		Reserved. Never write to this bit. Leave this bit cleared to 0.	0x0	R/W

CLOCK SELECT REGISTER

Address: 0x400C0414, Reset: 0x0000, Name: CLKSEL

Table 13. Bit Descriptions for CLKSEL

Bits	Bit Name	Settings	Description	Reset	Access
[15:4]	Reserved		Reserved.	0x0	R
[3:2]	ADCCLKSEL		Select ADC Clock Source. To configure the GPIO1 pin for an external clock,	0x0	R/W
			`pADI_AGPIO2->CON |= 3<<2;  // EXT_CLK pADI_AGPIO2->IEN |= 1<<1;  //AGPIO2.1(PWM1) input`		
		0	Internal high frequency oscillator clock.		
		1	External high frequency crystal (XTAL) clock.		
		10	Internal low frequency oscillator clock. Not recommended.		
		11	External clock.		
[1:0]	SYSCLKSEL		Select System Clock Source. To configure the GPIO1 pin for an external clock,	0x0	R/W

REGISTER DETAILS: CLOCK ARCHITECTURE

Table 13. Bit Descriptions for CLKSEL (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
			`pADI_AGPIO2->CON |= 3<<2; // EXT_CLK pADI_AGPIO2->IEN |= 1<<1; //AGPIO2.1(PWM1) input`		
		0	Internal high frequency oscillator clock.		
		1	External high frequency XTAL clock.		
		10	Internal low frequency oscillator clock. Not recommended.		
		11	External clock.		

GPIO CLOCK MUX SELECT TO GPIO1 PIN REGISTER

Address: 0x400C041C, Reset: 0x0000, Name: GPIOCLKMUXSEL

Select which digital clock is output to GPIO1 for observation.

Table 14. Bit Descriptions for GPIOCLKMUXSEL

Bits	Bit Name	Settings	Description	Reset	Access
[15:3]	Reserved		Reserved.	0x0	R
[2:0]	SEL		Configure Clock Mux Out to GPIO1.	0x0	R/W
		0	System clock.		
		1	Power gate low frequency clock.		
		10	PCLK.		
		11	Wake-up timer (WUT) on analog die clock.		
		100	GPT0 clock.		
		101	GPT1 clock.		
		110	ADC clock.		
		111	ADC control clock.		

KEY PROTECTION FOR CLKCON0 REGISTER

Address: 0x400C0420, Reset: 0x0000, Name: CLKCON0KEY

This register provides key protection for the CLKCON0 register.

Table 15. Bit Descriptions for CLKCON0KEY

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	KEY		Key to Allow Read and Write Access to the CLKCON0 Register. Write 0xA815 to this register before accessing CLKCON0.	0x0	W

CLOCK CONTROL OF LOW-POWER TIA CHOP, WATCHDOG, AND WAKE-UP TIMERS REGISTER

Address: 0x400C0A70, Reset: 0x0000, Name: CLKEN0

Table 16. Bit Descriptions for CLKEN0

Bits	Bit Name	Settings	Description	Reset	Access
[15:3]	Reserved		Reserved.	0x0	R
2	TIACHPDIS		TIA Chop Clock Disable.	0x1	R/W
		0	Turn on TIA chop clock.		
		1	Turn off TIA chop clock.		
1	SLPWUTDIS		Sleep and Wake-Up Timer Clock Disable.	0x0	R/W
		0	Turn on sleep wake-up timer clock.		
		1	Turn off sleep wake-up timer clock.		
0	WDTDIS		Watchdog Timer Clock Disable.	0x0	R/W
		0	Turn on watchdog timer clock.		
		1	Turn off watchdog timer clock.		

REGISTER DETAILS: CLOCK ARCHITECTURE

KEY PROTECTION FOR OSCCON REGISTER

Address: 0x400C0A0C, Reset: 0x0000, Name: OSCKEY

Table 17. Bit Descriptions for OSCKEY

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	OSCKEY	0xCB14	Oscillator Control Key Register. The OSCCON register is key-protected. Write 0xCB14 to OSCKEY before accessing the OSCCON register. A write to any other register before writing to OSCCON returns the protection to the lock state.	0x0	R/W

OSCILLATOR CONTROL REGISTER

Address: 0x400C0A10, Reset: 0x0303, Name: OSCCON

The OSCCON register is key protected. To unlock this protection, write 0xCB14 to the OSCKEY register before writing to the OSCCON register. A write to any other register before writing to the OSCCON register returns the protection to the lock state.

Table 18. Bit Descriptions for OSCCON

Bits	Bit Name	Settings	Description	Reset	Access
[15:11]	Reserved		Reserved.	0x0	R
10	HFXTALOK		Status of HFXTAL Oscillator. This bit indicates when the oscillator is stable after it is enabled. This bit is not a monitor and does not indicate a subsequent loss of stability.	0x0	R
		0	Oscillator is not yet stable or is disabled.		
		1	Oscillator is enabled, stable, and ready for use.		
9	HFOSCOK		Status of High Frequency Oscillator. This bit indicates when the oscillator is stable after it is enabled. This bit is not a monitor and does not indicate a subsequent loss of stability.	0x1	R
		0	Oscillator is not yet stable or is disabled.		
		1	Oscillator is enabled, stable, and ready for use.		
8	LFOSCOK		Status of Low Frequency Oscillator. This bit indicates when the oscillator is stable after it is enabled. This bit is not a monitor and does not indicate a subsequent loss of stability.	0x1	R
		0	Oscillator is not yet stable or is disabled.		
		1	Oscillator is enabled, stable, and ready for use.		
[7:3]	Reserved		Reserved.	0x0	R
2	HFXTALEN		HFXTAL Oscillator Enable. This bit is used to enable or disable the oscillator. The oscillator must be stable before use. This bit must be set before the SYSRESETREQ system reset can be initiated.	0x0	R/W
		0	The HFXTAL oscillator is disabled and placed in a low-power state.		
		1	The HFXTAL oscillator is enabled.		
1	HFOSCEN		High Frequency Oscillator Enable. This bit is used to enable or disable the oscillator. The oscillator must be stable before use. This bit must be set before the SYSRESETREQ system reset can be initiated.	0x1	R/W
		0	The high frequency oscillator is disabled and placed in a low-power state.		
		1	The high frequency oscillator is enabled.		
0	LFOSCEN		Low Frequency Oscillator Enable. This bit is used to enable or disable the oscillator. The oscillator must be stable before use.	0x1	R/W
		0	The low frequency oscillator is disabled and placed in a low-power state.		
		1	The low frequency oscillator is enabled.		

HIGH-POWER OSCILLATOR CONFIGURATION REGISTER

Address: 0x400C20BC, Reset: 0x00000024, Name: HPOSCCON

Table 19. Bit Descriptions for HPOSCCON

Bits	Bit Name	Settings	Description	Reset	Access
[31:3]	Reserved		Reserved.	0x0	R

REGISTER DETAILS: CLOCK ARCHITECTURE

Table 19. Bit Descriptions for HPOSCCON (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
2	CLK32MHZEN		16 MHz or 32 MHz Output Selector Signal. Select an output of 32 MHz or 16 MHz. The ADC can run at 32 MHz, but the system clock cannot run at 32 MHz. Divide the system clock by 2 first before switching the oscillator to 32 MHz. Refer to CLKCON0, Bits[5:0].	0x1	R/W
		0	Select 32 MHz output.		
		1	Select 16 MHz output.		
[1:0]	Reserved		Reserved.	0x0	R

POWER MODE CONFIGURATION REGISTER

Address: 0x400C22F0, Reset: 0x00000000, Name: PMBW

This register configures the high and low-power system mode for the high-speed DAC and ADC circuits.

Table 20. Bit Descriptions for PMBW

Bits	Bit Name	Settings	Description	Reset	Access
[31:4]	Reserved		Reserved.	0x0000000	R
[3:2]	SYSBW		Configure System Bandwidth. Configures the bandwidth of the high-speed DAC reconstruction filter and the ADC antialias filter.	0x0	R/W
		00	Reserved.		
		01	50 kHz, −3 dB bandwidth.		
		10	100 kHz, −3 dB bandwidth.		
		11	250 kHz, −3 dB bandwidth.		
1	Reserved		Reserved.	0x0	R
0	SYSHP		Set High-Speed DAC and ADC in High-Power Mode.	0x0	R/W
		0	Low-power mode. Clear this bit for impedance measurements <80 kHz.		
		1	High-power mode. Set this bit for impedance measurements >80 kHz.		

POWER MANAGEMENT UNIT

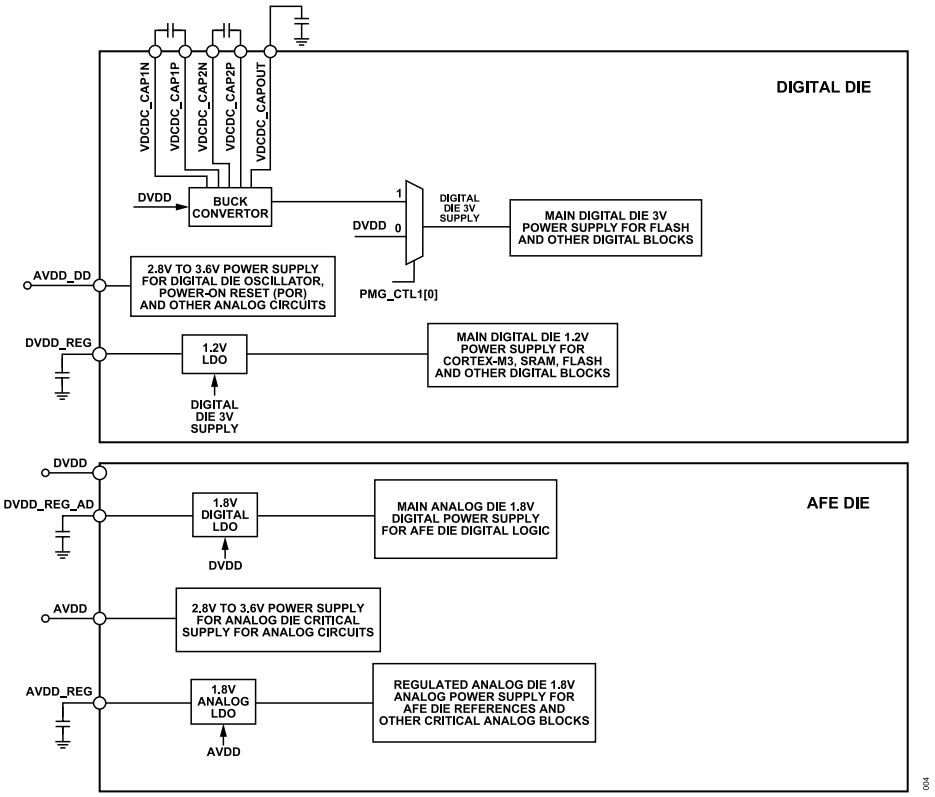

Figure 3. Power Supply Architecture Block Diagram

POWER MANAGEMENT UNIT FEATURES

The ADuCM355 contains two separate PMUs, one for each die. The PMUs control the different power modes of each ADuCM355 die.

The power management features of the device include the following:

- High efficiency buck converters to reduce power on the digital die.
  - Buck converter for active mode, which requires external flying capacitors, as shown in the Figure 3. The buck converter is disabled by default. For optimal performance of the analog peripherals, leave the converter disabled.
  - The DC-DC converter shown on the digital die is optional. If disabled, the three capacitors connected to the VDCDC_x pins are not required and these five pins can be left unconnected.
- Customized clock gating for active modes.
- Power gating to reduce leakage in sleep modes.
- Voltage monitoring.
- Flexible sleep modes with smart peripherals.
- Deep sleep modes with no retention.

Three power modes are available: active mode, flexi mode, and hibernate mode. The power mode control for each die is separate.

Active Mode

The Arm Cortex-M3 is executing from flash and SRAM on the digital die. PMG0 PWRMOD, Bits[1:0] = 00. The analog die circuitry is in active mode. ALLON PWRMOD, Bits[1:0] = 01.

Flexi Mode

In this mode, the Arm Cortex-M3 is disabled. The user selects the peripherals to be enabled, for example, SPI for DMA or I^2C for DMA.

POWER MANAGEMENT UNIT

Hibernate Mode

On the digital die, the system is power gated. 8 kB of SRAM is always retained. Up to an additional 24 kB SRAM can be selected to be retained. PMG0 PWRMOD, Bits[1:0] = 10.

On the analog die, the high-speed oscillator and high-speed clock source are powered down so that all blocks clocked by these clock sources are clock gated. The 32 kHz oscillator and the analog die watchdog timer remains active. Optionally, the low-power DACs, low-power reference, and low-power amplifiers can remain active to keep an external sensor biased. ALLON PWRMOD, Bits[1:0] = 10.

POWER MANAGEMENT UNIT OPERATION

The debug tools can prevent the Arm Cortex-M3 from fully entering power saving modes by setting bits in the debug logic. Only a power-on reset (POR) can reset the debug logic. Therefore, the device must be power cycled after using the serial wire debug with an application code containing the wait for interrupt (WFI) instruction.

Active Mode, Mode 0

The system is fully active. Memories and all user enabled peripherals are clocked, and the Arm Cortex-M3 processor executes instructions. The Arm Cortex-M3 processor manages its internal clocks and can be in a partial clock gated state. This clock gating affects only the internal Arm Cortex-M3 processing core. Automatic clock gating is used on all blocks except the I^2C, UART, and general-purpose timers. These blocks are manually clock gated using the CLKG CTL5 register for the digital die and the CLKEN1 register for the analog die.

The user code can use a WFI command to put the Arm Cortex-M3 processor into sleep mode. The processor is independent of the power mode settings of the PMU.

Writing 1 to CLKG CTL5, Bits[5:0] or AFE CLKEN1, Bits[9:0] stops the corresponding clock to peripherals. For the digital die peripherals, after the clock stops, if the user or software accesses any register in that peripheral, the clock is automatically enabled. In addition, writing 0 to these bits in the CLKG CTL5 or AFE CLKEN1 registers enables the corresponding clock to the peripheral.

Chip power-up with the LDO regulator is on by default on both dice. The buck regulator can be enabled to save power consumption by writing 1 to the PMG0 CTL1, Bit 0. When enabled, the input to the on-chip 1.2 V LDO regulator is the buck converter output, which is typically 1.6 V.

When the ADuCM355 wakes up from any of the low-power modes, the device returns to Mode 0.

Flexi Mode, Mode 1 (Digital Die Only)

In flexi mode, the system gates the clock to the Arm Cortex-M3 core after the Arm Cortex-M3 enters power-down by executing the WFI instruction. The rest of the system remains active, and no instructions can be executed. However, DMA transfers can continue to occur between peripherals and memories. The Arm Cortex-M3 processor, RCLK (clock to flash), is active, and the device wakes up using the nested vectored interrupt controller (NVIC).

Hibernate Mode, Mode 2

To enter hibernate mode, the user must first configure the analog die for hibernate by setting ALLON PWRMOD, Bits[1:0] = 10, and then, the user can place the digital die into hibernate mode. To minimize current consumption, configure unused digital GPIO pins as tristate on both dice.

On the digital die, in hibernate mode, the Arm Cortex-M3 and all digital peripherals are turned off. The SRAM can be programmed to retain up to 32 kB. The user can select the following:

- The amount of SRAM to retain. This amount is in addition to the 8 kB of SRAM always retained in the hibernate mode. This selection is controlled using SRAMRET, Bits[1:0].
- Control battery monitoring during hibernate mode. The regulated 1.2 V supply is always monitored to guarantee that data is never corrupted by the supply going below the minimum retention voltage. If the regulated supply falls below 1 V, the chip resets before any data is corrupted. Though the regulated supply is always monitored, there is an option to also monitor the AVDD_DD pin (2.8 V to 3.6 V supply) in hibernate mode by clearing PMG0 PWRMOD, Bit 3 = 0.

On the analog die, in hibernate mode, the AFE high-speed clock circuits are powered down, causing all blocks clocked by these circuits to enter a low-power, clock gated state. After setting ALLON PWRMOD, Bits[1:0] = 10, do not read back the value of the register because this read can halt the entry of the analog die into hibernate mode.

POWER MANAGEMENT UNIT

CODE EXAMPLES

Enter Power Saving Mode

The following function configures the analog die operating mode:

```
uint32_t AfePwrCfg(uint16_t iMode)
{
  // PSWFULLCON[14:13]= [11]b
  //Close switches NL and NL2. PSWFULLCON[11:10]= [11]b
   pADI_AFE-> PSWFULLCON|=0x6C00;  // Close PL2, PL1, P12, P11 switches to tie HSTIA N and D
//terminals to 1.8 V LDO
   pADI_ALLON->PWRKEY = 0x4859;
   pADI_ALLON->PWRKEY = 0xF27B;
   pADI_ALLON->PWRMOD = (pADI_ALLON->PWRMOD&(~BITM_ALLON_PWRMOD_PWRMOD))|iMode;
   //========================================
   //            IMPORTANT
   // If Chip is going into hibernate mode, you
   // cannot read PWRMOD after configuring it.
   // For safety reasons, return value set
   // to 0 directly.
   //========================================
   //return   pADI_ALLON->PWRMOD;
   return   0;
}
```

The following function configures the digital die operating mode:

```
int PwrCfg(int iMode,int iMonVbBat,int iSramRet)
{
   int32_t index = 0;
   uint32_t savedWDT;
   savedWDT = pADI_WDT0->CTL;  //None of the watchdog timer registers are retained in hibernate mode
   if (iMode > 3)    // Check for invalid sleep mode value
   {
      iMode = 0;
      return 1;
   }
   if ((iMode == 2) || (iMode == 3))
   {
      SCB->SCR = 0x04; // sleepdeep mode - write to the Cortex-m3 System Control register bit2
   }
   pADI_PMG0->PWRKEY = 0x4859;    // key1
   pADI_PMG0->PWRMOD = iMode|iMonVbBat;
   for (index=0;index<2;index++);
   __WFI();
   for (index=0;index<2;index++);
   pADI_WDT0->CTL = savedWDT;  //restore WDT control register.
   return 1;
}
```

POWER MANAGEMENT UNIT

Wake-Up Sequence

The digital die wake-up mechanism is different for each power-down mode. The wake-up is triggered by an interrupt or a reset. The sequence for the wake-up is different depending on the power mode. If the wake-up is triggered by a coming interrupt, the system first executes the interrupt routine.

The analog die exits hibernate mode when the digital die tries to read or write to any analog die register. When the Arm Cortex-M3 executes an instruction to access any analog die register when the analog die is in hibernate mode, the CPU halts until that instruction is complete. The CPU must wait for the analog die to complete its wake-up sequence before resuming.

The user must reset the ALLON PWRMOD register to active mode after the wake-up sequence is complete. This reset ensures that the analog die exits hibernate mode correctly when required in the user application.

This following code example shows how to wake up the analog die after the digital die has exited hibernate mode:

```
uiDummyRead = pADI_AFE->LPDACCON0; // read any analog die register to wake-up analog die
AfePwrCfg(AFE_ACTIVE);             // reset  pADI_ALLON->PWRMOD[1:0] = 0b01
```

The ADC reference requires 110 µs to settle after the analog die exits hibernate mode. Do not start any ADC conversions until this period has elapsed.

MONITOR VOLTAGE CONTROL

The user code must monitor the chip power supply voltages. The AFE die peripherals are not specified to operate at voltages <2.8 V. The ADuCM355 provides a number of features to help user code monitor the AVDD and DVDD supply rails of the ADuCM355.

On the digital die, voltage supervisory circuits are enabled at all times to guarantee that the AVDD_DD supply (2.8 V to 3.6 V) and the regulated supply are always within operating levels. The circuit monitoring these supplies is called the PMU.

The main features for the PMU circuit during active mode are as follows:

- Monitors DVDD voltage. Generates a reset to the chip if AVDD_DD supply is below 1.6 V. The analog die generates a reset at a higher voltage. Refer to the specifications section in the ADuCM355 data sheet for details.
- Monitors the state of the AVDD_DD. Generates AVDD_DD power supply monitor (PSM) interrupts between 3.6 V and 2.75 V, and AVDD_DD PSM interrupts between 2.75 V and 2.3 V. These ranges of interrupts are enabled by the PMG0 IEN register.
- Monitors regulated supply.
  - Generates an interrupt if the DVDD_REG regulated supply is greater than 1.32 V (overvoltage).
  - Generates an interrupt if the DVDD_REG regulated supply is less than 1.1 V (undervoltage).
  - Generates a reset if the DVDD_REG regulated supply is below 1.08 V.

The main features for the PMU during hibernate mode are as follows:

- Monitors battery voltage.
  - Generates an alarm to the processor if the supply voltage is less than 1.83 V (optional).
  - Generates a reset to the chip if the supply is less than 1.6 V (optional).
  - Monitors the state of the optional battery monitor feature. The PMU also provides optional battery monitoring between 3.6 V and 2.75 V, and battery between 2.75 V and 2.3 V.
- Monitors regulated supply. Generates a reset if the DVDD_REG regulated supply is below 1.08 V.

On the analog die, the ADC input mux on the analog die allows the user to measure a number of the input and regulated supply pins. These channels include the following:

- AVDD supply to the analog die.
- DVDD supply to the AFE and digital die.
- AVDD_REG regulated 1.8 V analog supply voltage.

The analog die has its own POR circuit that generates a full chip reset if the supply voltage drops below its brownout voltage. Refer to the ADuCM355 data sheet for details.

REGISTER SUMMARY: POWER MANAGEMENT UNIT

Table 21. Digital Die Power Management Register Summary (PMG0 Stack)

Address	Name	Description	Reset	Access
0x4004C000	IEN	Power supply monitor interrupt enable	0x00000000	R/W
0x4004C004	PSM_STAT	Power supply monitor status	0x2100	W1C
0x4004C008	PWRMOD	Power mode	0x00000000	R/W
0x4004C00C	PWRKEY	Key protection for PWRMOD and SRAMRET	0x00000000	W
0x4004C014	SRAMRET	Control for retention SRAM during hibernate mode	0x00000000	R/W
0x4004C044	CTL1	High-power buck control	0x00000000	R/W
0x4004C260	SRAM_CTL	Control for SRAM parity and instruction SRAM	0x80000000	R/W
0x4004C264	SRAM_INITSTAT	Initialization status	0x00000001	R/W

Table 22. Analog Die Power Management Register Summary (ALLON Stack)

Address	Name	Description	Reset	Access
0x400C0A00	PWRMOD	Power modes	0x0001	R/W
0x400C0A04	PWRKEY	Key protection for PWRMOD	0x0000	R/W

REGISTER DETAILS: POWER MANAGEMENT UNIT

POWER SUPPLY MONITOR INTERRUPT ENABLE REGISTER

Address: 0x4004C000, Reset: 0x00000000, Name: IEN

Table 23. Bit Descriptions for IEN

Bits	Bit Name	Settings	Description	Reset	Access
[31:11]	Reserved		Reserved.	0x00000	R
10	IENBAT		Interrupt Enable for AVDD_DD Range. Set this bit if an interrupt must be generated for the RANGEBAT bit. Configure RANGEBAT for the interrupt to be generated and then set IENBAT. An interrupt is generated if the DVDD_AD pin falls in the RANGEBAT bit. For example, if the battery is within the required range and the user wishes to monitor the battery, the user must configure the RANGE2 bit in the PSM_STAT register. Clear all PSM_STAT flags and then enable this interrupt. Otherwise, the RANGE1 bit of the PSM_STAT register of battery keeps issuing the interrupt.	0x0	R/W
		0	Disable IENBAT as an interrupt source.		
		1	Enable IENBAT as an interrupt source.		
9	Reserved		Reserved.	0x0	R
8	RANGEBAT		Battery Monitor Range. Configure the appropriate RANGEBAT bit setting to generate the interrupt.	0x0	R/W
		0	Configure to generate interrupt if AVDD_DD > 2.75 V.		
		1	Configure to generate interrupt if AVDD_DD is between 2.75 V and 1.6 V.		
[7:3]	Reserved		Reserved.	0x00	R
2	VREGOVR		Enable Interrupt When DVDD_REG is Greater Than 1.32 V (Overvoltage).	0x0	R/W
		0	Disable VREGOVR as an interrupt source.		
		1	Enable VREGOVR as an interrupt source.		
1	VREGUNDR		Enable Interrupt when DVDD_REG is Less Than 1 V (Undervoltage). If enabled, the interrupt connects to nonmaskable interrupt (NMI).	0x0	R/W
		0	Disable VREGUNDR as an interrupt source.		
		1	Enable VREGUNDR as an interrupt source.		
0	VBAT		Enable Interrupt for AVDD_DD. If enabled, the interrupt connects to NMI and generates an interrupt if AVDD_DD < 1.83 V.	0x0	R/W
		0	Disable AVDD_DD as an interrupt source.		
		1	Enable AVDD_DD as an interrupt source.		

POWER SUPPLY MONITOR STATUS REGISTER

Address: 0x4004C004, Reset: 0x2100, Name: PSM_STAT

Table 24. Bit Descriptions for PSM_STAT

Bits	Bit Name	Settings	Description	Reset	Access
[31:15]	Reserved		Reserved.	0x0000	R
14	RORANGE2		AVDD_DD Range 2 (2.75 V and 2.3 V) Read Only Status Bit.	0x0	R
		1	AVDD_DD in the range specified.		
		0	AVDD_DD not in the range specified.		
13	RORANGE1		Battery Voltage Range 1 (>2.75 V) Read Only Status Bit.	0x1	R
		1	AVDD_DD in the range specified.		
		0	AVDD_DD not in the range specified.		
[12:10]	Reserved		Reserved.	0x0	R
9	RANGE2		AVDD_DD Range 2 (2.75 V and 2.3 V). This is a write one to clear the status bit that indicates the relevant DVDD_AD range. Generates the AVDD_DD range interrupt if IEN, Bit 10 is set. The status bit sets again even after 1 is written to the flag to clear it if AVDD_DD falls in the range specified.	0x0	R/W1C
		1	AVDD_DD in the range specified.		
		0	AVDD_DD not in the range specified.		
8	RANGE1		AVDD_DD Range 1 (>2.75 V). This is a write one to clear status bit indicating the relevant AVDD_DD range. Generates the AVDD_DD range interrupt if IEN, Bit 10 is set. The status bit sets again even after 1 is written to the flag to clear it if AVDD_DD falls in the specified range.	0x1	R/W1C

REGISTER DETAILS: POWER MANAGEMENT UNIT

Table 24. Bit Descriptions for PSM_STAT (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		1	AVDD_DD in the specified range.		
		0	AVDD_DD not in the specified range.		
7	WICENACK		Wake-Up Interrupt Controller (WIC) Enable Acknowledge from Cortex.	0x0	R
[6:3]	Reserved		Reserved.	0x0	R
2	VREGOVR		Status Bit for Alarm Indicating Overvoltage for DVDD_REG. Bit set if DVDD_REG (LDO regulator output) > 1.32 V. Generates an interrupt if IEN, Bit 2 is set. This is write one to clear this bit. The status bit sets again even after 1 is written to the flag to clear it if DVDD_REG is > 1.32 V.	0x0	R/W1C
1	VREGUNDR		Status Bit for Alarm Indicating DVDD_REG is Less Than 1 V. Generates an interrupt if IEN Bit 1 is set. This bit sets if DVDD_REG < 1 V. This is a write one to clear bit. The status bit sets again even after 1 is written to the flag to clear it if DVDD_REG is < 1 V.	0x0	R/W1C
0	VBATUNDR		Status Bit Indicating an Alarm that AVDD_DD is Less than 1.8 V. Generates an interrupt if IEN Bit 0 is set. This bit sets if AVDD_DD < 1.83 V. This is a write one to clear bit. The status bit sets again even after 1 is written to the flag to clear it if AVDD_DD is < 1.83 V.	0x0	R/W1C

POWER MODE REGISTER

Address: 0x4004C008, Reset: 0x00000000, Name: PWRMOD

Table 25. Bit Descriptions for PWRMOD

Bits	Bit Name	Settings	Description	Reset	Access
[31:4]	Reserved		Reserved.	0x0	R
3	MONVBATN		Monitor AVDD_DD During Hibernate Mode. Monitors AVDD_DD by default. DVDD_REG (1.2 V LDO regulator). Monitoring cannot be disabled.	0x0	R/W
		0	AVDD_DD monitor enabled in PMU block. Default.		
		1	AVDD_DD monitor disabled in PMU block.		
2	Reserved		Reserved.	0x0	R/W
[1:0]	MODE		Power Mode Bits.	0x0	R/W
		00	Flexi mode.		
		01	Reserved.		
		10	Hibernate mode.		
		11	Reserved.		

KEY PROTECTION FOR PWRMOD AND SRAMRET REGISTER

Address: 0x4004C00C, Reset: 0x00000000, Name: PWRKEY

Table 26. Bit Descriptions for PWRKEY

Bits	Bit Name	Settings	Description	Reset	Access
[31:16]	Reserved		Reserved.	0x0	R
[15:0]	VALUE		Power Control Key Register. The PMG0 PWRMOD and PMG0 SRAMRET registers are key protected. One write to the key is necessary to change the value in the PMG0 PWRMOD and PMG0 SRAMRET registers. Write 0x4859 to PMG0 PWRKEY before writing to PMG0 PWRMOD or PMG0 SRAMRET register. A write to any other register on the Arm peripheral bus before writing to PMG0 PWRMOD or PMG0 SRAMRET returns the protection to the lock state.	0x0000	W

CONTROL FOR RETENTION SRAM DURING HIBERNATE MODE REGISTER

Address: 0x4004C014, Reset: 0x00000000, Name: SRAMRET

Table 27. Bit Descriptions for SRAMRET

Bits	Bit Name	Settings	Description	Reset	Access
[31:2]	Reserved		Reserved.	0x0	R
1	BNK2EN		Enable Retention Bank 2 (16 kB). Bank address is 0x10000000 to 0x10003FFF if SRAM_CTL, Bit 31 = 1. Bank address is 0x20004000 to 0x20007FFF if SRAM_CTL, Bit 31 = 0.	0x0	R/W

REGISTER DETAILS: POWER MANAGEMENT UNIT

Table 27. Bit Descriptions for SRAMRET (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		0	Disable retention of SRAM Bank 2.		
		1	Enable retention of SRAM Bank 2 during hibernate mode. This option consumes more power.		
0	BNK1EN		Enable Retention Bank 1 (8 kB). Bank address is 0x20002000 to 0x20003FFF.	0x0	R/W
		0	Disable retention of SRAM Bank 1.		
		1	Enable retention of SRAM Bank 1 during hibernate mode. This option consumes more power.		

HIGH-POWER BUCK CONTROL REGISTER

Address: 0x4004C044, Reset: 0x00000000, Name: CTL1

Table 28. Bit Descriptions for CTL1

Bits	Bit Name	Settings	Description	Reset	Access
[31:1]	Reserved		Reserved.	0x00000000	R/W
0	HPBUCKEN		Enable High-Power Buck.	0x0	R/W
		0	Buck regulator is disabled.		
		1	Buck regulator is enabled.		

CONTROL FOR SRAM PARITY AND INSTRUCTION SRAM REGISTER

Address: 0x4004C260, Reset: 0x80000000, Name: SRAM_CTL

Table 29. Bit Descriptions for SRAM_CTL

Bits	Bit Name	Settings	Description	Reset	Access
31	INSTREN		Enables Instruction SRAM.	0x1	R/W
		1	CPU instructions use SRAM address range of 0x10000000 to 0x10003FFF.		
		0	SRAM used for data.		
[30:22]	Reserved		Reserved.	0x000	R
21	PENBNK5		Enable Parity Check for SRAM Bank 5.	0x0	R/W
		0	Disable parity check of this bank of SRAM.		
		1	Enable parity check of this bank of SRAM.		
20	PENBNK4		Enable Parity Check for SRAM Bank 4.	0x0	R/W
		0	Disable parity check of this bank of SRAM.		
		1	Enable parity check of this bank of SRAM.		
19	PENBNK3		Enable Parity Check for SRAM Bank 3.	0x0	R/W
		0	Disable parity check of this bank of SRAM.		
		1	Enable parity check of this bank of SRAM.		
18	PENBNK2		Enable Parity Check for SRAM Bank 2. SRAM Address 0x10000000 to Address 0x10003FFF if SRAM_CTL, Bit 31 = 1. Address range is 0x20004000 to 0x20007FFF if SRAM_CTL, Bit 31 = 0. Parity is checked when data is read and when a byte or half word data is written to this SRAM area. If a parity error is detected, a bus error is generated and the execution vectors to the bus fault interrupt.	0x0	R/W
		0	Disable parity check of this bank of SRAM.		
		1	Enable parity check of this bank of SRAM.		
17	PENBNK1		Enable Parity Check for SRAM Bank 1. SRAM Address 0x20002000 to Address 0x20003FFF. Parity is checked when data is read and when a byte or half word data is written to this SRAM area. If a parity error is detected, a bus error is generated and the execution vectors to the bus fault interrupt.	0x0	R/W
		0	Disable parity check of this bank of SRAM.		
		1	Enable parity check of this bank of SRAM.		
16	PENBNK0		Enable Parity Check for SRAM Bank 0. SRAM Address 0x20000000 to Address 0x20001FFF. Parity is checked when data is read and when a byte or half word data is written to this SRAM area. If a parity error is detected, a bus error is generated and the execution vectors to the bus fault interrupt.	0x0	R/W
		0	Disable parity check of this bank of SRAM.		
		1	Enable parity check of this bank of SRAM.		

REGISTER DETAILS: POWER MANAGEMENT UNIT

Table 29. Bit Descriptions for SRAM_CTL (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
15	ABTINIT		Abort Current Initialization. Self cleared.	0x0	R/W
14	AUTOINIT		Automatic Initialization on Wake-Up from Hibernate Mode.	0x0	R/W
13	STARTINIT		Write One to Trigger Initialization. Self cleared.	0x0	R/W
[12:6]	RESERVED		Reserved.	0x00	R
5	BNK5EN		Enable Initialization of SRAM Bank 5.	0x0	R/W
		0	Disable initialization of this bank of SRAM.		
		1	Enable initialization of this bank of SRAM.		
4	BNK4EN		Enable Initialization of SRAM Bank 4.	0x0	R/W
		0	Disable initialization of this bank of SRAM.		
		1	Enable initialization of this bank of SRAM.		
3	BNK3EN		Enable Initialization of SRAM Bank 3.	0x0	R/W
		0	Disable initialization of this bank of SRAM.		
		1	Enable initialization of this bank of SRAM.		
2	BNK2EN		Enable Initialization of SRAM Bank 2. SRAM Address 0x10000000 to Address 0x10003FFF if SRAM_CTL, Bit 31 = 1. Address range is 0x20004000 to 0x20007FFF if SRAM_CTL, Bit 31 = 0. Initialization is necessary on exiting hibernate mode if the SRAM is not retained and parity checking is enabled.	0x0	R/W
		0	Disable initialization of this bank of SRAM.		
		1	Enable initialization of this bank of SRAM.		
1	BNK1EN		Enable Initialization of SRAM Bank 1. SRAM Address 0x20002000 to Address 0x20003FFF. Initialization is necessary on exiting hibernate mode if the SRAM is not retained and parity checking is enabled.	0x0	R/W
		0	Disable initialization of this bank of SRAM.		
		1	Enable initialization of this bank of SRAM.		
0	BNK0EN		Enable Initialization of SRAM Bank 0. SRAM Address 0x20000000 to Address 0x20001FFF. Initialization is necessary on exiting hibernate mode if the SRAM is not retained and parity checking is enabled.	0x0	R/W
		0	Disable initialization of this bank of SRAM.		
		1	Enable initialization of this bank of SRAM.		

INITIALIZATION STATUS REGISTER

Address: 0x4004C264, Reset: 0x00000001, Name: SRAM_INITSTAT

Table 30. Bit Descriptions for SRAM_INITSTAT

Bits	Bit Name	Settings	Description	Reset	Access
[31:6]	Reserved		Reserved.	0x0000000	R/W
5	BNK5		Initialization Status of SRAM Bank 5.	0x0	R
		0	Not initialized.		
		1	Initialization completed.		
4	BNK4		Initialization Status of SRAM Bank 4.	0x0	R
		0	Not initialized.		
		1	Initialization completed.		
3	BNK3		Initialization Status of SRAM Bank 3.		R
		0	Not initialized.	0x0	
		1	Initialization completed.		
2	BNK2		Initialization Status of SRAM Bank 2.	0x0	R
		0	Not initialized.		
		1	Initialization completed.		
1	BNK1		Initialization Status of SRAM Bank 1.	0x0	R
		0	Not initialized.		
		1	Initialization completed.		

REGISTER DETAILS: POWER MANAGEMENT UNIT

Table 30. Bit Descriptions for SRAM_INITSTAT (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
0	BNK0		Initialization Status of SRAM Bank 0.	0x1	R
		0	Not initialized.		
		1	Initialization completed.		

POWER MODES REGISTER

Address: 0x400C0A00, Reset: 0x0001, Name: PWRMOD

Table 31. Bit Descriptions for PWRMOD

Bits	Bit Name	Settings	Description	Reset	Access
[15:4]	Reserved		Reserved.	0x0	R
3	SEQSLPEN		Autosleep Function by Sequencer Command.	0x0	R/W
		0	Disables the sequencer autosleep function.		
		1	Enables the sequencer autosleep function.		
2	TMRSLPEN		Autosleep Function by Sleep and Wake-Up Timer.	0x0	R/W
		0	Disables the sleep and wake-up timer autosleep function.		
		1	Enables the sleep and wake-up timer autosleep function.		
[1:0]	PWRMOD		Power Mode Control Bits. When read, these bits contain the last power mode value entered by user code.	0x1	R/W
		00, 11	Reserved. Do not enter this mode.		
		01	Active Mode. Normal working mode. All digital circuits powered up. User can optionally power down blocks by disabling their input clock.		
		10	Hibernate Mode. Digital core powered down, most analog die blocks are powered down. Low-power DACs or references can remain active to bias an external sensor. The high-speed clock is powered down. Only the low-speed clock is powered up.		

KEY PROTECTION FOR PWRMOD REGISTER

Address: 0x400C0A04, Reset: 0x0000, Name: PWRKEY

Table 32. Bit Descriptions for PWRKEY

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	PWRKEY		PWRMOD Key Register. The PWRMOD register is key protected. Two writes to the key are necessary to change the value in the PWRMOD register: first 0x4859, then 0xF27B. A write to any other register before writing to PWRMOD returns the protection to the lock state.	0x0	R/W

ARM CORTEX-M3 PROCESSOR

The ADuCM355 contains an embedded Arm Cortex-M3 processor. The Arm Cortex-M3 processor provides a high performance, low cost platform that meets the system requirements of minimal memory implementation, reduced pin count, and low-power consumption, and delivers computational performance and system response to interrupts.

ARM CORTEX-M3 PROCESSOR FEATURES

The high performance features of the Arm Cortex-M3 processor are as follows:

- A 26 MHz maximum clock speed.
- 128 kB of embedded flash memory with error correction code (ECC).
- 32 kB system SRAM with parity.
- 32 kB user configurable instruction or data SRAM with parity. 4 kB of SRAM can be used as cache memory to reduce active power consumption by reducing access to flash memory.
- 1.25 Dhrystone million instructions per second (DMIPS)/MHz.
- Many instructions are single cycle, including multiply.
- Separate data and instruction buses allow simultaneous data and instruction accesses to be performed.
- Optimized for single cycle flash usage.
- A flexible RTC that supports a wide range of wake-up times.
- Three general-purpose timers and one watchdog timer.
- Programmable GPIOs, each with optional input interrupt capability.

The low power features are as follows:

- PMU.
- POR and PSM.
- Buck converter for improved efficiency during active state.
- The core is implemented using advanced clock gating so that only the actively used logic consumes dynamic power.
- Power saving mode support (hibernate mode). The design has separate clocks to allow unused parts of the processor to be stopped.

The advanced interrupt handling features are as follows:

- The NVIC supports up to 240 interrupts. The ADuCM355 supports 64 of these interrupts. The vectored interrupt feature greatly reduces interrupt latency because there is no need for software to determine which interrupt handler to serve. Additionally, there is no need to have software to set up nested interrupt support.
- The Arm Cortex-M3 processor automatically pushes registers onto the stack at the entry interrupt and retrieves them at the exit interrupt. Pushing and retrieving reduces interrupt handling latency and allows interrupt handlers to be normal C functions.
- Dynamic priority control for each interrupt.
- Latency reduction using late arrival interrupt acceptance and tail chain interrupt entry.
- Immediate execution of an NMI request for safety critical applications.

The system features are as follows:

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

The debug support features are as follows:

- Serial wire debug (SWD) port.
- Flash patch and breakpoint (FPB) unit for implementing breakpoints. Limited to two hardware breakpoints.
- Data watchpoint and trigger (DWT) unit for implementing watchpoint trigger resources and system profiling. Limited to one hardware watchpoint. The DWT does not support data matching for watchpoint generation because it has only one comparator.

ARM CORTEX-M3 PROCESSOR OPERATION

Several Arm Cortex-M3 processor components are flexible in their implementation. This section details the implementation of these components in the ADuCM355.

ARM CORTEX-M3 PROCESSOR

Serial Wire Debug

The ADuCM355 only supports the serial wire interface via the SWCLK and SWDIO pins. The device does not support the 5-wire, Joint Action Test Group (JTAG) interface. The SWCLK pin is driven by the debug probe. The SWDIO signal is a bidirectional signal that can be driven by the debug probe or target, depending on the protocol phase.

NVIC

The Arm Cortex-M3 processor includes an NVIC, which offers several features, as follows:

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

In addition, the NVIC has an NMI input. The NVIC is implemented on the ADuCM355, and more details are available in the System Exceptions and Peripheral Interrupts section.

Wake-Up Interrupt Controller

The ADuCM355 has a modified WIC that provides the lowest possible power-down current. See the Power Management Unit section for details.

It is not recommended to enter power saving mode when servicing an interrupt. However, if the device enters power saving mode when servicing an interrupt, it can only be woken up by a higher priority interrupt source.

ARM CORTEX-M3 PROCESSOR RELATED DOCUMENTS

The following list contains documentation related to the Arm Cortex-M3:

- Arm Cortex-M3 Processor Technical Reference Manual Revision r2p1 (DDI 0337)
- Arm Processor Cortex-M3 (AT420) and Cortex-M3 with ETM (AT425) Software Developers Errata Notice
- Armv7-M Architecture Reference Manual (DDI 0403) with Errata Markups
- Arm Debug Interface Architecture Specification ADIV5.0 to ADIV5.2 (IHI 0031)
- PrimeCell µDMA Controller (PL230) Technical Reference Manual, Revision r0p0 (DDI 0417)

SYSTEM RESETS

There are three primary reset sources on the digital die, as shown in Figure 4. A digital die watchdog timer is also available, but is disabled by default. These reset sources are as follows:

- External reset.
- POR.
- Software reset.

In addition, there are four similar reset sources on the analog die, as follows:

- External reset.
- POR.
- Software reset of digital device.
- Watchdog timer reset.

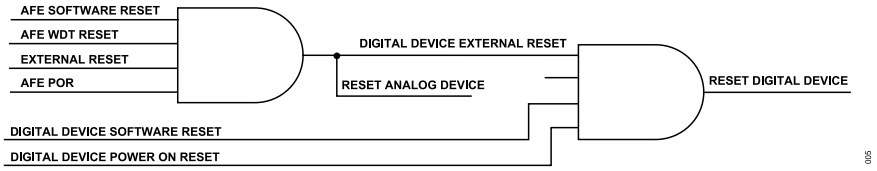

Figure 4. Reset Sources of the ADuCM355

The analog die POR and digital die POR are differentiated due to separate power supplies. Two AFE reset sources are also reported to the digital die as an external reset.

On both dice, a POR circuit is present. If either POR circuit triggers, the entire chip resets and all control registers and circuitry return to their default state. Likewise, both dice have an external reset. The external pin reset or hardware reset occurs when the external $\overline{\text{RESET}}$ pin is pulled low. All circuits and control registers return to their default states. The analog die watchdog timer reset circuit generates a reset if it is not refreshed correctly. By default, a watchdog timer reset resets all circuits and control registers.

The RST_STAT register indicates the source of the last reset to the digital die. This register can be used during a reset exception service routine to identify the source of the reset to the digital die. The RSTSTA register indicates the source of the last reset to the analog die. The RSTSTA register can be used during a reset exception service routine to identify the source of the reset to the analog die.

After a reset sequence, the on-chip kernel completes a self check of the analog die. Though problems are unlikely, if an issue occurs, AFEDIESTA, Bit 0 sets to 1. After every reset, read this bit. If this bit is read as 1, do not continue with analog die operations.

DIGITAL DIE RESET OPERATION

The software system reset is provided as part of the Arm Cortex-M3 processor. To generate a software system reset, call the NVIC_SystemReset() function. This function effectively writes 0x05FA to the 16 MSBs of the Cortex register, AIRCR, Address 0xE000ED0C. The NVIC_SystemReset() function, along with other useful functions, are defined in the Cortex microcontroller software interface standard (CMSIS) header files that are provided with the tools from third party vendors. The NVIC_SystemReset() function is defined in the **core_cm3.h** file.

The RST_STAT register stores the cause for the reset until it is cleared by writing to the same register. The RST_STAT register can be used during a reset exception service routine to identify the source of the reset. The watchdog timer is enabled by default after a reset. Table 33 lists details of all reset types.

Table 33. Digital Die Reset Implications

Reset	Reset External Pins to Default State	Execute Kernel	Reset All MMRs Except RST_STAT Register	Reset All Peripherals	Valid SRAM	RST_STAT Register After Reset Event
Software	Yes[1]	Yes	Yes	Yes	Yes/No[2]	RST_STAT, Bit 3 = 1
Watchdog	Yes	Yes	Yes	Yes	Yes/No[2]	RSTSTA, Bit 2 = 1
External Reset Pin	Yes	Yes	Yes	Yes	Yes/No[2]	RST_STAT, Bit 1 = 1

SYSTEM RESETS

Table 33. Digital Die Reset Implications (Continued)

Reset	Reset External Pins to Default State	Execute Kernel	Reset All MMRs Except RST_STAT Register	Reset All Peripherals	Valid SRAM	RST_STAT Register After Reset Event
POR	Yes	Yes	Yes	Yes	No	RST_STAT, Bit 0 = 1 and RST_STAT, Bits[5:4] have information on the cause of POR reset

[1] The GPIOx pins return to their default state (same as a POR event).

[2] Random access memory (RAM) is not valid in the case of a reset following a UART download.

Software Reset Command

The ARM Cortex-M3 supports a software reset command. Call the reset by using the following CMSIS library function:

```
NVIC_SystemReset();               // Call Software reset
```

This function involves a write the SYSRESETREQ bit in the Cortex AIRCR register. See the ARM Cortex-M3 user manual for details on this register. When a software reset of the digital die is called, the chip is reset and all registers on both dice return to their default states.

REGISTER SUMMARY: SYSTEM RESETS

Table 34. Digital Die Reset Register Summary

Address	Name	Description	Reset	Access
0x4004C040	RST_STAT	Digital die reset status	0x000000XX	R/W1C

Table 35. Always On Register Summary

Address	Name	Description	Reset	Access
0x400C0A40	RSTSTA	Always on reset status	0x000X	R/W1C

Table 36. Analog Die Status Register Summary

Address	Name	Description	Reset	Access
0x40007008	AFEDIESTA	Analog die status	0x0000	R

REGISTER DETAILS: SYSTEM RESETS

DIGITAL DIE RESET STATUS REGISTER

Address: 0x4004C040, Reset: 0x000000XX, Name: RST_STAT

Table 37. Bit Descriptions for RST_STAT

Bits	Bit Name	Settings	Description	Reset	Access
[15:6]	Reserved		Reserved.	0x0	R
[5:4]	PORSRC		POR Source for Digital Die. This bit contains additional details after a POR occurs.		R/W
		00	POR triggered because DVDD drops below POR threshold.		
		01	POR triggered because DVDD drops below POR threshold.		
		10	POR triggered because DVDD_REG supply drops below 1.08 V.		
		11	POR triggered because DVDD_REG drops below fail-safe.		
3	SWRST		Software Reset. Set automatically to 1 when the Arm Cortex-M3 system reset is generated. Cleared by writing 1 to the bit.	0x0	R/W1C
2	Reserved		Reserved.	0x0	R/W1C
1	EXTRST		External Reset. Set automatically to 1 when an external reset occurs. Cleared by writing 1 to the bit.	0x0	R/W1C
0	POR		POR. Set automatically when a POR occurs. Cleared by writing 1 to the bit.	0x0	R/W1C

ALWAYS ON RESET STATUS REGISTER

Address: 0x400C0A40, Reset: 0x000X, Name: RSTSTA

Table 38. Bit Descriptions for RSTSTA

Bits	Bit Name	Settings	Description	Reset	Access
[15:3]	Reserved		Reserved.	0xX	R
2	WDRST		Watchdog Timeout. Set automatically to 1 when a watchdog timeout occurs. Cleared by writing 1 to the bit.	0xX	R/W1C
1	EXTRST		External Reset. Set automatically to 1 when an external reset occurs. Cleared by writing 1 to the bit.	0xX	R/W1C
0	POR		AFE POR. Set automatically when a POR occurs. Cleared by writing 1 to the bit.	0xX	R/W1C

ANALOG DIE STATUS REGISTER

Address: 0x40007008, Reset: 0x0000, Name: AFEDIESTA

Table 39. Bit Descriptions for AFEDIESTA

Bits	Bit Name	Settings	Description	Reset	Access
[15:1]	Reserved		Reserved.	0x0	R
0	AFEDIESTA		Status of AFE to Digital Die Communication Status After a Reset Sequence. Always read this bit after every reset.	0x0	R
		0	Analog die trimming and calibration has been confirmed.		
		1	Analog die is not configured correctly and must not be used.		

PROGRAMMING, PROTECTION, AND DEBUG

BOOTING

The processor supports the following boot modes:

- Booting from internal flash. Pin setting is 1.
- If the BM/P1.1 pin = 0, user code is prevented from executing. Control stays within the bootloader code with the serial wire enabled. The user flash space cannot be viewed in this mode if the space is protected by the user flash metadata.

SECURITY FEATURES

The ADuCM355 processor provides a combination of hardware and software protection mechanisms that lock out access to the device in secure mode but grant access in open mode. During startup, the system is clocked from an internal on-chip oscillator. A reset computes a hardware checksum of the information area and then permits the CPU to execute the Analog Devices bootloader in the flash information area if the checksum passes. The Analog Devices bootloader inspects a GPIO boot pin, which determines whether user code is allowed to execute or not.

The ADuCM355 features read protection, which protects device contents (flash, SRAM, CPU registers, and peripheral registers) from being read through an external interface by an unauthorized user.

SAFETY FEATURES

The ADuCM355 processor provides a number of features that help achieve certain levels of system safety and reliability. The level of safety is mainly dominated by system considerations, and the following safety features are provided to enhance robustness.

Multiparity Bit Protected L1 Memories

In the SRAM and cache L1 memory space, each word is protected by multiple parity bits to detect the single event upsets that occur in all RAMs.

Debug Features

The SWCLK and SWDIO pins are used for debugging and programming. By default, an internal pull-up resistor to DVDD is provided. To save power, users can disable the internal pull-up resistor and reconfigure these pins as tristate. Take care when reconfiguring these pins from their default state, because this reconfiguration disables debug access to the ADuCM355. It is recommended to only disable the SWCLK and SWDIO feature of these pins in user code when code development and debugging are almost complete. If the SWCLK and SWDIO pins are accidently converted to tristate, mass erase the device by resetting the device. Set BM/P1.1 = 0, which allows the user to call the mass erase function of a serial wire-based debugger.

The ADuCM355 can also utilize the analog die watchdog timer for debugging. Disable the analog die watchdog timer when developing or debugging user code. Enable the timer only toward the end of the development cycle. If left enabled during debug sessions, unexpected watchdog timer resets can crash source level debugger programs and cause unexpected resets to user sensor interface.

SYSTEM EXCEPTIONS AND PERIPHERAL INTERRUPTS

CORTEX-M3 AND FAULT MANAGEMENT

The ADuCM355 integrates an Arm Cortex-M3 processor, which supports several system exceptions and interrupts generated by peripherals. Table 40 lists the Arm Cortex-M3 processor system exceptions.

Table 40. System Exceptions

Exception Number	Type	Priority	Description
1	Reset	−3 (highest)	Any reset.
2	NMI	−2	Nonmaskable interrupt connected to a combination of logical ORs of DVDD_REG pin undervoltage or AVDD_DD pin undervoltage. See Table 23.
3	Hard fault	−1	All fault conditions if the corresponding fault handler is not enabled.
4	Memory management fault	Programmable	Access to invalid locations.
5	Bus fault	Programmable	Prefetch fault, memory access fault, data abort, and other address or memory related faults.
6	Usage fault	Programmable	Same as undefined instruction executed or invalid state transition attempt.
7 to 10	Reserved	Not applicable	Reserved.
11	SVCALL	Programmable	System service call with supervisor mode call (SVC) instruction. Used for system function calls.
12	Debug monitor	Programmable	Debug monitor for breakpoint, watchpoint, or external debug requests.
13	Reserved	Not applicable	Reserved.
14	PENDSV	Programmable	Pendable request for system service. Used for queuing system calls until other tasks and interrupts are serviced.
15	SYSTICK	Programmable	System tick timer.

The NVIC controls the peripheral interrupts, which are listed in Table 41. All interrupt sources can wake up the Arm Cortex-M3 core from flexi mode. Only a limited number of interrupts can wake up the processor from hibernate mode, as shown in Table 41. When the device is woken up from flexi or hibernate mode, it returns to active mode. If the processor enters flexi or hibernate mode while the processor is in an interrupt handler, only an interrupt source with a higher priority than the current interrupt can wake up the device. Higher priority means having a higher value in a bit setting in the Cortex IPRx registers.

Two steps are usually required to configure an interrupt as follows:

1. Configure a peripheral to generate an interrupt request to the NVIC.
2. Configure the NVIC for that peripheral request.

Table 41. Interrupt Vectors

			Wake Up From	
Exception Number	IRQx	Vector	Flexi	Hibernate
16	IRQ0	Digital Die Real-Time Clock 1, wake-up timer, hibernate RTC	Yes	Yes
17	IRQ1	Reserved	Not applicable	Not applicable
18	IRQ2	External Interrupt 1 (SYS_WAKE)	Yes	Yes
19	IRQ3	Reserved	Not applicable	Not applicable
20	IRQ4	External Interrupt 3, UART receive wake-up interrupt, and INTCxxx register interrupt	Yes	Yes
21	IRQ5	Reserved	Yes	No
22	IRQ6	Digital Die DVDD_REG pin overrange	Yes	No
23	IRQ7	DVDD pin voltage range	Yes	Yes
24	IRQ8	Reserved	Not applicable	Not applicable
25	IRQ9	GPIO Interrupt A	Yes	No
26	IRQ10	GPIO Interrupt B	Yes	No
27	IRQ11	Digital Die General-Purpose Timer 0	Yes	No
28	IRQ12	Digital Die General-Purpose Timer 1	Yes	No
29	IRQ13	Flash controller	Yes	No
30	IRQ14	UART0	Yes	No
31	IRQ15	SPI0	Yes[1]	No
32	IRQ16	SPI1	Yes[1]	No

SYSTEM EXCEPTIONS AND PERIPHERAL INTERRUPTS

Table 41. Interrupt Vectors (Continued)

Exception Number	IRQx	Vector	Wake Up From	
			Flexi	Hibernate
33	IRQ17	I^2C target	Yes[1]	No
34	IRQ18	I^2C initiator	Yes[1]	No
35	IRQ19	DMA Error	Yes	No
36	IRQ20	DMA Channel 0 done	Yes	No
37	IRQ21	DMA Channel 1 done	Yes	No
38	IRQ22	DMA Channel 2 done	Yes	No
39	IRQ23	DMA Channel 3 done	Yes	No
40	IRQ24	DMA Channel 4 done	Yes	No
41	IRQ25	DMA Channel 5 done	Yes	No
42	IRQ26	DMA Channel 6 done	Yes	No
43	IRQ27	DMA Channel 7 done	Yes	No
44	IRQ28	DMA Channel 8 done	Yes	No
45	IRQ29	DMA Channel 9 done	Yes	No
46	IRQ30	DMA Channel 10 done	Yes	No
47	IRQ31	DMA Channel 11 done	Yes	No
48	IRQ32	DMA Channel 12 done	Yes	No
49	IRQ33	DMA Channel 13 done	Yes	No
50	IRQ34	DMA Channel 14 done	Yes	No
51	IRQ35	DMA Channel 15 done	Yes	No
52	IRQ36	Reserved	Not applicable	Not applicable
53	IRQ37	Reserved	Not applicable	Not applicable
54	IRQ38	Reserved	Yes	No
55	IRQ39	Reserved	Not applicable	Not applicable
56	IRQ40	Digital Die General-Purpose Timer 2	Yes	No
57	IRQ41	Digital die crystal oscillator	Yes	No
58	IRQ42	Reserved	Not applicable	Not applicable
59	IRQ43	Reserved	Not applicable	Not applicable
60	IRQ44	Reserved	Yes	No
61 to 63	IRQ45 to IRQ47	Reserved	Not applicable	Not applicable
64	IRQ48	Analog die ADC	Yes	No
65 to 67	IRQ49 to IRQ51	Reserved	Not applicable	Not applicable
68	IRQ52	Analog die watchdog timer	Yes	No
69	IRQ53	Reserved	Not applicable	Not applicable
70	IRQ54	Analog Die General-Purpose Timer 0	Yes	No
71	IRQ55	Analog Die General-Purpose Timer 1	Yes	No
72	IRQ56	Reserved	Not applicable	Not applicable
73	IRQ57	DMA analog die data FIFO (DMA Channel 17)	Yes	No
74	IRQ58	DMA Channel 18 done	Yes	No
75	IRQ59	DMA Channel 19 done	Yes	No
76	IRQ60	DMA Channel 20 done	Yes	No
77	IRQ61	DMA Channel 21 done	Yes	No
78	IRQ62	DMA Channel 22 done	Yes	No
79	IRQ63	DMA Channel 23 done	Yes	No

[1] The corresponding PCLK is required to generate the interrupt.

SYSTEM EXCEPTIONS AND PERIPHERAL INTERRUPTS

Internal to the Arm Cortex-M3 processor, the highest user-programmable priority (0) is treated as fourth priority after a reset, an NMI, or a hard fault. The ADuCM355 implements three priority bits, which means that eight priority levels are available as programmable priorities. The default for all the programmable priorities is 0. If the same priority level is assigned to two or more interrupts, their hardware priority (the lower the position number) determines the order in which the processor activates them. For example, if both Digital Die General-Purpose Timer 0 and Digital Die General-Purpose Timer 1 are priority Level 1, Digital Die General-Purpose Timer 0 has higher priority.

To enable an interrupt for any peripheral from IRQ0 to IRQ31, set the appropriate bit in the Cortex ISER0 register. ISER0 is a 32-bit register, and each bit corresponds to the first 32 entries in Table 41.

For example, to enable the External Interrupt 0 interrupt source in the NVIC, set ISER0, Bit 2 = 1. Similarly, to disable External Interrupt 1, SYS_WAKE, set ICER0, Bit 2 = 1.

To enable an interrupt for any peripheral from IRQ32 to IRQ63, set the appropriate bit in the Cortex ISER1 register. ISER1 is a 32-bit register, and Bit 0 to Bit 31 in the ISER1 register correspond to IRQ32 to IRQ63.

For example, to enable the General-Purpose Timer 2 interrupt source in the NVIC, set ISER1, Bit 8 = 1. Similarly, to disable the General-Purpose Timer 2 interrupt, set ICER1, Bit 8 = 1.

Alternatively, CMSIS provides a number of useful NVIC functions in the core_cm3.h file. The NVIC_EnableIRQ (TMR2_EVT_IRQn) function enables the General-Purpose Timer 2 interrupt. The interrupt can be disabled by calling the NVIC_DisableIRQ (TMR2_EVT_IRQn) function.

To set the priority of a peripheral interrupt, set the Cortex IPRx registers appropriately or call the NVIC_SetPriority() function. For example, NVIC_SetPriority (TMR2_EVT_IRQn, 2) configures the General-Purpose Timer 2 interrupt with a priority level of 2.

Table 43 lists the registers to enable and disable relevant interrupts and set the priority levels. The registers in Table 43 are defined in the CMSIS core_cm3.h file, which is provided with tools from third party vendors.

INTERRUPT SOURCES FROM THE ANALOG DIE

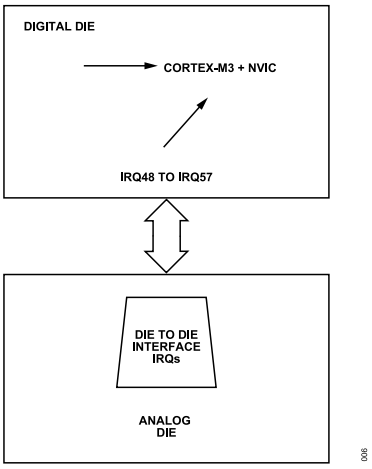

Figure 5. Analog Die Interrupt Connections to Digital Die

The analog die peripherals provide different interrupt sources to the NVIC. These sources can be selected via the standard internal die to die interface. The analog die interrupts connect to NVIC IRQ48 to IRQ57, as shown in Figure 5. Table 42 lists all the analog die interrupt sources (interrupt enable register bit and interrupt status bit) for IRQ48, Exception Number 64, via the interrupt enable register, ADCINTIEN.

Table 42. Analog Die Interrupts List

Exception Number	IRQx	Interrupt Enable Register	Interrupt Enable Register Bit	Interrupt Status Register	Interrupt Status Register Bit
64	IRQ48	ADCINTIEN	ADCRDYIEN	ADCINTSTA	ADCRDY
			DFTRDYIEN		DFTRDY
			SINC2RDYIEN		SINC2RDY
			TEMPRDYIEN		TEMPRDY
			ADCMINFAILIEN		ADCMINERR
			ADCMAXFAILIEN		ADCMAXERR

SYSTEM EXCEPTIONS AND PERIPHERAL INTERRUPTS

Table 42. Analog Die Interrupts List (Continued)

Exception Number	IRQx	Interrupt Enable Register	Interrupt Enable Register Bit	Interrupt Status Register	Interrupt Status Register Bit
			ADCDELTAFAILIEN		ADCDIFFERR
			MEANIEN		MEANRDY

Two steps are required to generate an interrupt request to the NVIC on the digital die as follows:

1. Configure the corresponding peripheral to generate an interrupt request to the NVIC. Refer to the interrupt enable register bit column in Table 42.
2. Configure the NVIC for that peripheral request. Call NVIC_EnableIRQ(IRQn).

The following example function shows how to select the ADC ready (ADCRDY) interrupt source for the die to die interrupt source to the NVIC:

```
void AfeAdcInterruptSetup(void)
{
     AfeAdcIntCfg(BITM_AFE_ADCINTIEN_ADCRDYIEN);     // Select ADCReady as interrupt source
   NVIC_EnableIRQ(AFE_ADC_IRQn);                     // Enable AFE_ADC interrupt source in NVIC
}
void SIP0_IRQHandler()
{
    uiIntSta = AfeAdcIntSta();
    if (uiIntSta & BITM_AFE_ADCINTSTA_ADCRDY)
    {
        szADCSamples[i]= AfeAdcRd(RAWADC);
    }
}
```

CLEARING ANALOG DIE INTERRUPT SOURCES

IRQ48, IRQ52, IRQ54, IRQ55, and IRQ57 are interrupt sources from the analog die.

Ensure that the interrupt is fully serviced and the associated interrupt status flags are fully cleared before exiting the interrupt service routine. If user code exits the interrupt service routine before the interrupt flag is fully cleared, the CPU program counter can repeatedly vector back into the same interrupt service routine.

In the case of timer interrupts, add a short delay of 30 AFE_SYSCLK periods after clearing the timeout status bit but before exiting the timer interrupt service routine.

The following is an example interrupt service routine for the Analog Die General-Purpose Timer 0:

```
// AFE General-Purpose Timer0 Interrupt handler.
void AfeGpTimer0_Int_Handler()
{
  ucSecondTimer = 1;
  pADI_AGPT0->CLRI0 = 0x1;                    // Clear Timeout IRQ
  delay(100);                        // Ensure the delay equates to 10 >=30x analog die System clocks
}
```

CORTEX-M3 NVIC REGISTER LIST

The registers in Table 43 are found in the Arm Cortex-M3.

Table 43. 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	System tick timer control and status.	R/W

SYSTEM EXCEPTIONS AND PERIPHERAL INTERRUPTS

Table 43. NVIC Registers (Continued)

Address	Analog Devices Header File Name	Description	Access
0xE000E014	STRVR	System tick timer reload value.	R/W
0xE000E018	STCVR	System tick timer current value.	R/W
0xE000E01C	STCR	System tick timer calibration value.	R
0xE000E100	ISER0	Set IRQ0 to IRQ31 enable. Each bit corresponds to IRQ0 to IRQ31 in Table 41.	R/W
0xE000E104	ISER1	Set IRQ32 to IRQ63 enable. Each bit corresponds to IRQ32 to IRQ63 in Table 41.	R/W
0xE000E180	ICER0	Clear IRQ0 to IRQ31 by setting the appropriate bit. Each bit corresponds to IRQ0 to IRQ31 in Table 41.	R/W
0xE000E184	ICER1	Clear IRQ32 to IRQ63 by setting the appropriate bit. Each bit corresponds to IRQ32 to IRQ63 in Table 41.	R/W
0xE000E200	ISPR0	Set IRQ0 to IRQ31 pending. Each bit corresponds to IRQ0 to IRQ31 in Table 41.	R/W
0xE000E204	ISPR1	Set IRQ32 to IRQ63 pending. Each bit corresponds to IRQ32 to IRQ63 in Table 41.	R/W
0xE000E280	ICPR0	Clear IRQ0 to IRQ31 pending. Each bit corresponds to IRQ0 to IRQ31 in Table 41.	R/W
0xE000E284	ICPR1	Clear IRQ32 to IRQ63 pending. Each bit corresponds to IRQ32 to IRQ63 in Table 41.	R/W
0xE000E300	IABR0	IRQ0 to IRQ31 active bits.	R/W
0xE000E304	IABR1	IRQ32 to IRQ63 active bits.	R/W
0xE000E400	IPR0	IRQ0 to IRQ3 priority.	R/W
0xE000E404	IPR1	IRQ4 to IRQ7 priority.	R/W
0xE000E408	IPR2	IRQ8 to IRQ11 priority.	R/W
0xE000E40C	IPR3	IRQ12 to IRQ15 priority.	R/W
0xE000E410	IPR4	IRQ16 to IRQ19 priority.	R/W
0xE000E414	IPR5	IRQ20 to IRQ23 priority.	R/W
0xE000E418	IPR6	IRQ24 to IRQ27 priority.	R/W
0xE000E41C	IPR7	IRQ28 to IRQ31 priority.	R/W
0xE000E420	IPR8	IRQ32 to IRQ35 priority.	R/W
0xE000E424	IPR9	IRQ36 to IRQ39 priority.	R/W
0xE000E428	IPR10	IRQ40 to IRQ43 priority.	R/W
0xE000E42C	IPR11	IRQ44 to IRQ47 priority.	R/W
0xE000E430	IPR12	IRQ48 to IRQ51 priority.	R/W
0xE000E434	IPR13	IRQ52 to IRQ55 priority.	R/W
0xE000E438	IPR14	IRQ56 to IRQ59 priority.	R/W
0xE000E43C	IPR15	IRQ60 to IRQ63 priority.	R/W
0xE000ED00	CPUID	CPU ID base.	R
0xE000ED04	ICSR	Interrupt control and status.	R/W
0xE000ED08	VTOR	Vector table offset.	R/W
0xE000ED0C	AIRCR	Application interrupt and reset control.	R/W
0xE000ED10	SCR	System control.	R/W
0xE000ED14	CCR	Configuration control.	R/W
0xE000ED18	SHPR1	System Handler 1.	R/W
0xE000ED1C	SHPR2	System Handler 2.	R/W
0xE000ED20	SHPR3	System Handler 3.	R/W
0xE000ED24	SHCRS	System handler control and state.	R/W
0xE000ED28	CFSR	Configurable fault status.	R/W
0xE000ED2C	HFSR	Hard fault status.	R/W
0xE000ED34	MMAR	Memory manage fault address.	R/W
0xE000ED38	BFAR	Bus fault address.	R/W
0xE000EF00	STIR	Software trigger interrupt.	W

EXTERNAL INTERRUPT CONFIGURATION

Two external interrupts are implemented, separate from those described in the Digital Inputs and Outputs section. One of these external interrupts is the P1.0/SYS_WAKE pin. The other is the interrupt source from the analog die that connects to External Interrupt 3 line of the

SYSTEM EXCEPTIONS AND PERIPHERAL INTERRUPTS

digital die, which can be connected to the UART input pin (P0.11/UART_SIN). These two external interrupts can be separately configured to detect any combination of the following type of events:

- Edge: rising edge, falling edge, or both rising and falling edges. An interrupt signal (pulse) is sent to the NVIC upon detecting a transition from low to high, high to low, or on either high to low or low to high.
- Level: high or low. An interrupt signal is generated and remains asserted in the NVIC until the conditions generating the interrupt deassert. The level must be maintained for a minimum of one core clock cycle to be detected.

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

Ensure that the associated GPxIE register bits are enabled for the required external interrupt input. The GPxIE registers enable the input path circuit for the external interrupt.

For example, for External Interrupt 1 (SYS_WAKE), the following code disables the P1.0/SYS_WAKE output pin and enables the input path. The appended code also enables the External Interrupt 1 NVIC interrupt source.

```
pADI_GPIO1->OEN &= 0xFFFE;          //Disable P1.0 output.
pADI_GPIO1->IEN |= 0x0001;          //Enable input path for P1.0 input.
pADI_XINT0->CFG0 |=0x80;            //External IRQ1 enabled.
NVIC_EnableIRQ(XINT_EVT1_IRQn);     //Enable External Interrupt 1 source.
```

REGISTER SUMMARY: SYSTEM EXCEPTIONS AND PERIPHERAL INTERRUPTS

Table 44. Digital Die External Interrupts Register Summary

Address	Name	Description	Reset	Access
0x4004C080	XINT_CFG0	External Interrupt Configuration 0	0x00200000	R/W
0x4004C084	XINT_EXT_STAT	External wake-up interrupt status	0x00000000	R
0x4004C090	XINT_CLR	External interrupt clear	0x00000000	R/W
0x4004C094	XINT_NMICLR	Nonmaskable interrupt clear	0x00000000	R/W

Table 45. Analog Die Global Interrupt Enable Register Summary

Address	Name	Description	Reset	Access
0x400C0A28	EI2CON	Analog die interrupt enable	0x0000	R/W

REGISTER DETAILS: SYSTEM EXCEPTIONS AND PERIPHERAL INTERRUPTS

EXTERNAL INTERRUPT CONFIGURATION 0 REGISTER

Address: 0x4004C080, Reset: 0x00200000, Name: XINT_CFG0

Table 46. Bit Descriptions for XINT_CFG0

Bits	Bit Name	Settings	Description	Reset	Access
[31:24]	Reserved		Reserved.	0x0	R
[23:21]	UART_RX_MDE		External Interrupt Using P0.11/UART_SIN Wake-Up Mode.	0x1	R/W
		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).		
20	UART_RX_EN		External Interrupt Enable Bit. This bit enables the P0.11/UART_SIN pin to generate an interrupt on IRQ4. Refer to Table 41.	0x0	R/W
		1	P0.11/UART_SIN wake-up interrupt is enabled.		
		0	P0.11/UART_SIN wake-up interrupt is disabled.		
[19:16]	Reserved		Reserved.	0x0	R/W
15	IRQ3EN		External Interrupt 3 Enable Bit.	0x0	R/W
		0	External Interrupt 3 disabled.		
		1	External Interrupt 3 enabled.		
[14:12]	IRQ3MDE		External Interrupt 3 Mode.	0x0	R/W
		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).		
[11:8]	Reserved		Reserved.	0x0	R/W
7	IRQ1EN		External Interrupt 1 Enable Bit.	0x0	R/W
		0	External Interrupt 1 disabled.		
		1	External Interrupt 1 enabled.		
[6:4]	IRQ1MDE		External Interrupt 1 Mode.	0x0	R/W
		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).		
[3:0]	Reserved		Reserved.	0x0	R/W

REGISTER DETAILS: SYSTEM EXCEPTIONS AND PERIPHERAL INTERRUPTS

EXTERNAL WAKE-UP INTERRUPT STATUS REGISTER

Address: 0x4004C084, Reset: 0x00000000, Name: XINT_EXT_STAT

Table 47. Bit Descriptions for XINT_EXT_STAT

Bits	Bit Name	Settings	Description	Reset	Access
[31:6]	Reserved		Reserved.	0x0	R
5	STAT_UART_RXWKUP		Interrupt Status Bit for P0.11/UART_SIN Wake-Up Interrupt. Read only register bit. Cleared by writing 1 to XINT_CLR, Bit 5.	0x0	R
		0	P0.11/UART_SIN wakeup did not generate the interrupt.		
		1	P0.11/UART_SIN wakeup generated the interrupt.		
4	RESERVED		Reserved.	0x0	R
3	STAT_EXTINT3		Interrupt Status Bit for External Interrupt 3. This bit is valid if there is an INTC interrupt from the AFE die to the digital die.	0x0	R
		0	External Interrupt 3 did not generate the interrupt.		
		1	External Interrupt 3 generated the interrupt.		
2	RESERVED		Reserved.	0x0	R
1	STAT_EXTINT1		Interrupt Status Bit for External Interrupt 1. This bit is valid if there is an interrupt asserted on SYS_WAKE. Cleared by writing 1 to XINT_CLR, Bit 1. Read only register bit.	0x0	R
		0	External Interrupt 1 did not generate the interrupt.		
		1	External Interrupt 1 generated the interrupt.		
0	Reserved		Reserved.	0x0	R

EXTERNAL INTERRUPT CLEAR REGISTER

Address: 0x4004C090, Reset: 0x00000000, Name: XINT_CLR

Table 48. Bit Descriptions for XINT_CLR

Bits	Bit Name	Settings	Description	Reset	Access
[31:6]	Reserved		Reserved.	0x0	R
5	UART_RX_CLR		External Interrupt Clear for P0.11/UART_SIN Wake-Up Interrupt. Set to 1 to clear the interrupt status flag. Cleared automatically by hardware.	0x0	R/W
4	IRQ3		External Interrupt 3. Set to 1 to clear the interrupt status flag. Cleared automatically by hardware.	0x0	R/W
[3:2]	Reserved		Reserved.	0x0	R/W
1	IRQ1		External Interrupt 1. Set to 1 to clear the interrupt status flag. Cleared automatically by hardware.	0x0	R/W
0	Reserved		Reserved.	0x0	R

NONMASKABLE INTERRUPT CLEAR REGISTER

Address: 0x4004C094, Reset: 0x00000000, Name: XINT_NMICLR

Table 49. Bit Descriptions for XINT_NMICLR

Bits	Bit Name	Settings	Description	Reset	Access
[31:1]	Reserved		Reserved.	0x0	R
0	CLR		NMI Clear. Set to 1 to clear an interrupt status flag when the NMI interrupt is set. Cleared automatically by hardware.	0x0	R/W

ANALOG DIE INTERRUPT ENABLE REGISTER

Address: 0x400C0A28, Reset: 0x0000, Name: EI2CON

Table 50. Bit Descriptions for EI2CON

Bits	Bit Name	Settings	Description	Reset	Access
[15:4]	Reserved		Reserved.	0x0	R
3	BUSINTEN		Bus Interrupt Detection Enable Bit. Set before entering hibernate to enable the AFE wakeup via any analog die access.	0x0	R/W

REGISTER DETAILS: SYSTEM EXCEPTIONS AND PERIPHERAL INTERRUPTS

Table 50. Bit Descriptions for EI2CON (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		0	Die interface interrupt wakeup disabled.		
		1	Die interface interrupt wakeup enabled.		
[2:0]	Reserved		Reserved. Leave as 0.	0x0	R/W

ANALOG DIE CIRCUITRY SUMMARY

The ADuCM355 analog die includes the following eight main blocks:

- ADC. The ADC is a high-speed SAR ADC with a wide range of voltage and current input channels. See the ADC Circuit section.
- Low-power potentiostat and TIA. This block also includes the low-power DACs used to set the DC bias voltage of an external electrochemical sensor. See the Low-Power Potentiostat Amplifiers and Low-Power TIAs section.
- High-speed TIA. The high-speed TIA is intended for measuring AC current signals with the ADC, especially during impedance measurements. The high-speed TIA supports a wider input signal bandwidth than the low-power TIA. The current consumption of the high-speed TIA is higher than that of the low-power TIA. See the High-Speed TIA Circuits section.
- High-speed DAC circuits. The high-speed DAC is designed to support AC impedance measurements with its specially designed output excitation amplifier. The AC signal from the output of the high-speed DAC can be coupled on the DC sensor bias voltage set by the low-power DAC via the excitation amplifier. See the High-Speed DAC Circuits section.
- Programmable switches connecting external sensor to the high-speed DAC and the high-speed TIA. The ADuCM355 provides flexibility in connecting external pins to the high-speed TIA and excitation amplifier terminals. See the Programmable Switches Connecting the External Sensor to the High-Speed DAC and High-Speed TIA section.
- Analog die digital circuits. This block includes optional programmable timers. See the Analog Die General-Purpose Timers section.
- Use case configurations. The Use Case Configurations section describes typical electrochemical sensor use cases and the configuration of the ADuCM355 for each use case.
- Sequencer (see the Sequencer section).

ADC, HIGH-SPEED DAC, AND ASSOCIATED AMPLIFIERS OPERATING MODE CONFIGURATION

The ADC and high-speed DAC circuits are flexible in trading current consumption vs. signal bandwidth. If the ADC and high-speed DACs are used to measure and generate signals <80 kHz for low frequency impedance measurements, these blocks can be configured for low-power mode by clearing PMBW, Bit 0 = 0. This configuration minimizes power consumption. If the ADC and or high-speed DAC are used to measure and generate signals >80 kHz for high frequency impedance measurements, set PMBW, Bit 0 = 1.

SYSTEM BANDWIDTH CONFIGURATION

The user must configure the bandwidth setting for the reconstruction filter of the high-speed DAC, the antialias filter of the ADC, and the bandwidth of the high-speed TIA, in addition to configuring PMBW Bit 0. PMBW, Bits[3:2] allow the user to set this configuration. For the HSTIACON register, ensure that HSTIACON, Bits[5:1] = 00000 in low-power mode and HSTIACON, Bits[5:1] = 11111 in high-power mode. See Table 20 for more details on the PMBW register.

ANALOG DIE CIRCUITRY SUMMARY

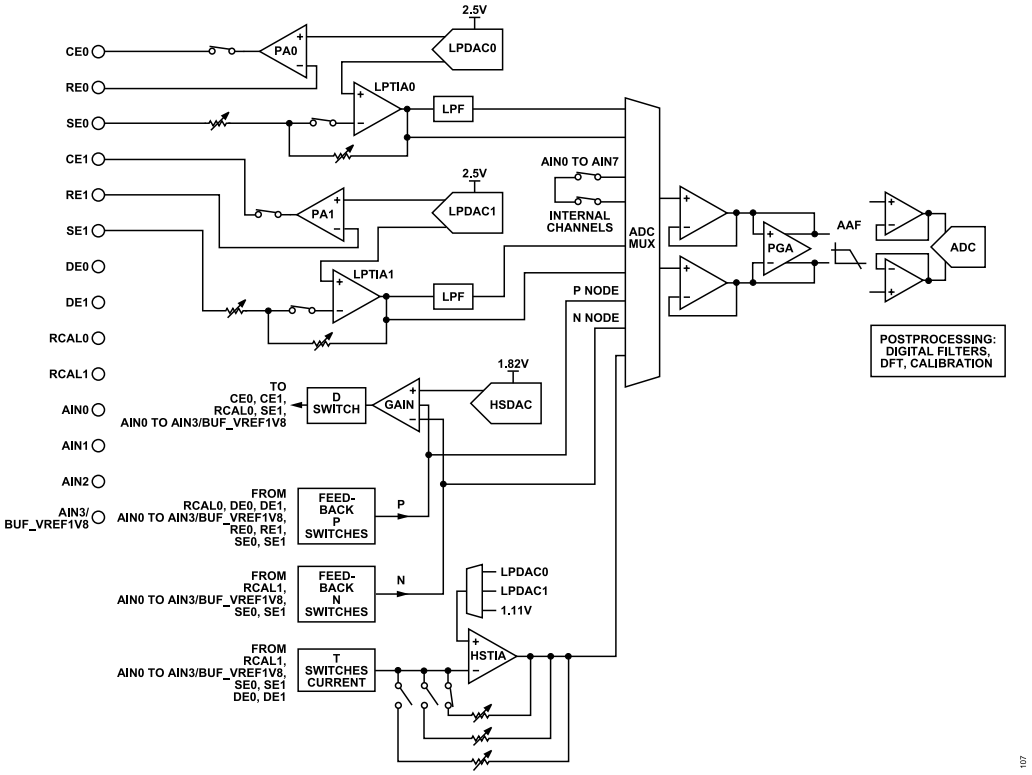

Figure 6. Block Level Overview of AFE Die Analog Circuitry and Connection to External Pins

REGISTER SUMMARY: ANALOG DIE CIRCUITRY

Table 51. Analog Die Circuitry Register Summary

Address	Name	Description	Reset	Access
0x400C2000	AFECON	Analog configuration	0x00080000	R/W

REGISTER DETAILS: ANALOG DIE CIRCUITRY

AFE CONFIGURATION REGISTER

Address: 0x400C2000, Reset: 0x00080000, Name: AFECON

Specific bits in these registers are relevant to particular blocks in the analog die. The relevant bits for each block are as follows:

- Bits relevant to the ADC block are Bit 16, Bit 15, Bit 13, Bit 12, Bit 8, Bit 7, and Bit 5.
- Bits relevant to the high-speed TIA block are Bit 11 and Bit 5.
- Bits relevant to the high-speed DAC block are Bit 21, Bit 20, Bit 14, Bit 10, Bit 9, and Bit 5.

Table 52. Bit Descriptions for AFECON

Bits	Bit Name	Settings	Description	Reset	Access
[31:22]	Reserved		Reserved.	0x0	R
21	HSDACBUFEN		Enable DC DAC Buffer. Enable the buffer for the high impedance output of the DC DAC.	0x0	R/W
		0	Disable DC DAC buffer.		
		1	Enable DC DAC buffer.		
20	HSDACREFEN		High-Speed DAC Reference Enable.	0x0	R/W
		0	Reference disable. Clear to 0 to disable the high-speed DAC reference.		
		1	Reference enable. Set to 1 to enable the high-speed DAC reference.		
19	ALDOILIMITEN		Analog LDO Buffer Current Limiting. Enable the AFE analog LDO buffer current limiting. If enabled, LDO buffer current limiting limits the current drawn from an external battery when charging the capacitor on AVDD_REG.	0x1	R/W
		0	Analog LDO buffer current limiting enabled.		
		1	Analog LDO buffer current limiting disabled.		
[18:17]	Reserved		Reserved.	0x0	R
16	SINC2EN		ADC Output 50 Hz or 60 Hz Filter Enable. Enable 50 Hz or 60 Hz supply rejection filter. When the sinc2 digital filter is used, clear this bit and set it before restarting ADC conversions.	0x0	R/W
		0	Supply rejection filter disabled. Disable sinc2 (50 Hz/60 Hz digital filter). Disable this bit for impedance measurements.		
		1	Supply rejection filter enabled. Enable sinc2 (50 Hz/60 Hz digital filter).		
15	DFTEN		DFT Hardware Accelerator Enable. Enable the DFT hardware acceleration block.	0x0	R/W
		0	DFT hardware accelerator disabled.		
		1	DFT hardware accelerator enabled.		
14	WAVEGENEN		Waveform Generator Enable. Enable waveform generator.	0x0	R/W
		0	Waveform generator disabled.		
		1	Waveform generator enabled.		
13	TEMPCONVEN0		ADC Temperature Sensor 0 Convert Enable. Enables ADC temperature channel conversion. When the temperature conversion is complete, the result is available in the TEMPSENSDAT0 register. After the conversion, this bit is reset to 0.	0x0	R/W
		0	Temperature Channel 0 reading disabled.		
		1	Temperature Channel 0 reading enabled.		
12	TEMPSENSEN0		ADC Temperature Sensor 0 Channel Enable. Enable temperature sensor.	0x0	R/W
		0	Temperature sensor disabled. The temperature sensor is powered down.		
		1	Temperature sensor enabled. The temperature sensor is powered up but no temperature readings are performed unless the TEMPCONVEN0 bit = 1.		
11	HSTIAEN		Enable High Speed TIA.	0x0	R/W
		0	High-speed TIA disabled.		
		1	High-speed TIA enabled.		
10	INAMPEN		Enable Excitation Instrumentation Amplifier on the High-Speed DAC Output. Enables instrumentation amplifier.	0x0	R/W
		0	High-speed DAC programmable instrumentation amplifier disabled.		
		1	High-speed DAC programmable instrumentation amplifier enabled.		

REGISTER DETAILS: ANALOG DIE CIRCUITRY

Table 52. Bit Descriptions for AFECON (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
9	EXBUFEN		Enable Excitation Buffer on High-Speed DAC Output.	0x0	R/W
		0	High-speed DAC excitation buffer disabled.		
		1	High-speed DAC excitation buffer enabled.		
8	ADCCONVEN		ADC Conversion Start Enable.	0x0	R/W
		0	ADC idle. ADC powered on but not converting.		
		1	ADC conversions enabled.		
7	ADCEN		ADC Power Enable. Enable ADC.	0x0	R/W
		0	ADC disabled. ADC is powered off.		
		1	ADC enabled. ADC is powered on. The ADCCONVEN bit must be set to start conversions.		
6	HSDACEN		High-Speed DAC Enable. Enable the high-speed DAC and its reconstruction filter. This bit only enables the analog block, not including the DAC waveform generator.	0x0	R/W
		0	High-speed DAC disabled.		
		1	High-speed DAC enabled.		
5	HPREFDIS		Disable High-Power Reference. This is the power-down signal for the high-power reference.	0x0	R/W
		0	High-power reference enabled. Must be cleared to 0 for the ADC and high-speed DAC to operate.		
		1	High-power reference disabled. Power down the reference.		
4	Reserved		Reserved.	0x0	R/W
[3:0]	Reserved		Reserved.	0x0	R

ADC CIRCUIT

ADC CIRCUIT OVERVIEW

The SAR ADC circuit is implemented on the analog die. The die operates from a 2.8 V to 3.6 V power supply. The Arm Cortex-M3 processor interfaces to the ADC via an internal die-to-die interface. The ADC uses a precision, low drift, factory calibrated 1.82 V reference. An external reference can also be connected to the VREF_1.82V pin. ADC conversions can be triggered by writing directly to the AFECON register.

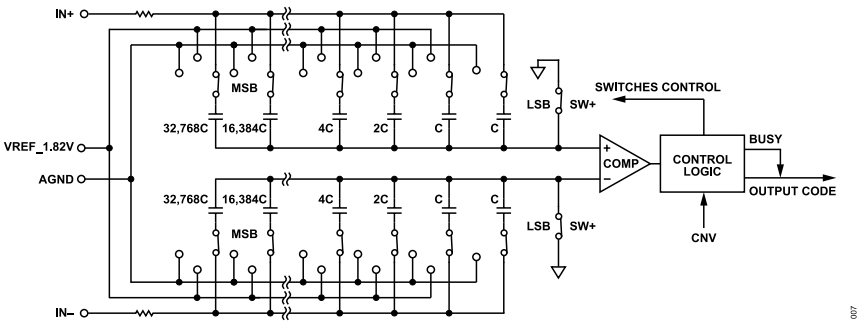

Figure 7. ADC Core Block Diagram

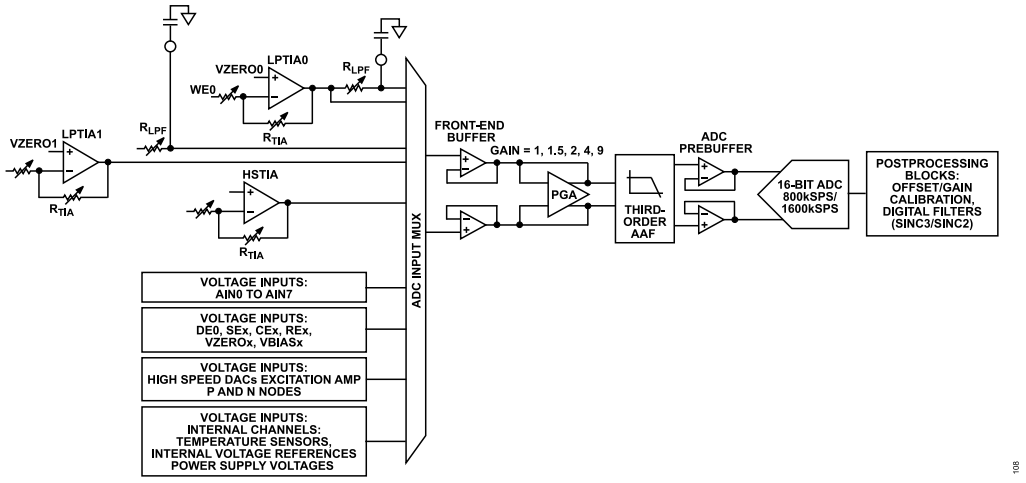

Figure 8. ADC Input Stages

ADC CIRCUIT FEATURES

The ADuCM355 includes a fast multichannel, 16-bit ADC. The input multiplexer supports a number of external and internal channels. There are up to 34 user-selectable channels, including the following:

- Four low-power current measurement channels. These channels are designed to measure the sense, working, and diagnostic electrode outputs of electrochemical sensors. The current channels feed into a programmable load resistor (R_{LOAD}).
  - Includes two low-power TIAs, each containing a programmable gain resistor to convert very small currents to a voltage signal that the ADC can measure.
  - The low-power current channels can be configured to sample with a low-pass filter in place. Bypassing the low-pass filter can be useful for diagnostic operations on an electrochemical sensor output.
- One high-speed current input channel for performing impedance measurements up to 200 kHz. This channel has dedicated TIAs with a programmable gain resistor.
- Multiple external voltage inputs.
  - Eight dedicated voltage input channels, AIN0 to AIN7.
  - The sensor electrode pins can also be measured as ADC voltage inputs.
  - The SE0, SE1, DE0, RE0, RE1, CE0, and CE1 pins are included. Divide by two options are available for the CE0 and CE1 inputs.
- Internal ADC channels.
  - AVDD, DVDD, and AVDD_REG are power supply measurement channels.

ADC CIRCUIT

- ADC, high-speed DAC, and low power reference voltage inputs.
- Two internal die temperature sensors.
- Four low-power DAC output voltages: VBIAS0, VZERO0, VBIAS1, and VZERO1.

The ADC has a number of postprocessing features, as follows:

- Digital filtering of sinc2 and sinc3, and 50 Hz or 60 Hz power supply rejection.
- DFT, used with impedance measurements to automatically calculate magnitude and phase values.
- Programmable averaging of ADC results.
- Programmable statistics option for calculating the mean.
- Multiple calibration options to support system calibration of the current, voltage, and temperature channels.

The ADC input stage provides an input buffer to support low input current and low input leakage specifications on all channels.

To support a range of current and voltage based input ranges, the ADC front end provides a PGA and programmable TIAs. The PGA supports gains of 1, 1.5, 2, 4, and 9. The low-power TIAs support programmable gain resistors ranging from 200 Ω to 512 kΩ. The high-speed TIA, used for impedance measurement, supports programmable gain resistors ranging from 200 Ω to 160 kΩ.

The default reference source of the ADC is a precision, low drift, internal 1.8 V reference source. Optionally, connect an external reference to the VREF_1.82V and AGND_REF pins.

The ADC supports averaging and digital filtering options. With these options, the user can trade off speed and precision. The highest ADC update rate is 800 kHz in low-power mode or 1.6 MSPS in high-power mode with no digital filtering. The ADC filtering options also include a 50 Hz or 60 Hz mains power supply filter. When the mains power supply filter is enabled, the ADC update rate is typically 900 Hz. If no filtering is selected, the supported resolution reduces to 14 bits.

The ADC supports a number of postprocessing features. These features include a DFT calculator. The DFT is intended for impedance measurements to remove the processing requirements from the microcontroller. Minimum, maximum, and mean value detections are also supported.

ADC CIRCUIT OPERATION

The SAR ADC is based on a charge redistribution DAC. The capacitive DAC consists of two identical arrays of 16 binary weighted capacitors that are connected to the two inputs of the comparator.

The ADC block operates from the 16 MHz clock in normal operation. This clock ensures a maximum ADC update rate of 800 kSPS with no filtering. For high-power mode, select the 32 MHz oscillator as the ADC clock source. The ADC maximum update rate is 1.6 MSPS with higher power consumption. For normal mode and high-power mode, it is strongly recommended to enable the sinc3 filter option at a minimum, which results in a 200 kSPS.

ADC TRANSFER FUNCTION

The transfer function in Figure 9 shows the ADC output codes on the y-axis and the differential voltage into the ADC. The ADC negative input channel is the 1.11 V voltage source (ADCCON, Bits[12:8] = 0b01000). The positive input channel is any voltage input to the ADC after the TIA, PGA, and input buffer stages.

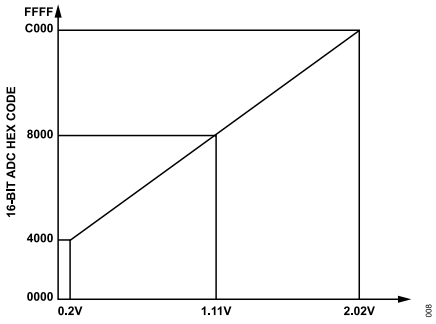

Figure 9. Ideal ADC Transfer Function, Voltage Input to ADC vs. Output Codes, Where Input is ADCVBIAS_CAP at 1.11 V

ADC CIRCUIT

To calculate the input voltage (V_{IN}), use the following equation to convert ADC codes to a voltage:

For PGA gain = 1,

$$V_{IN} = (VREF \times (((ADCDAT - 0x8000)/2^{15}))) + ADCVBIAS\_CAP \quad (1)$$

For PGA gain = 1.5,

$$V_{IN} = (VREF/1.5 \times (((ADCDAT - 0x8000)/2^{15}) \times 1.835/1.82)) + ADCVBIAS\_CAP \quad (2)$$

For PGA gain = 2, 4, or 9,

$$V_{IN} = (VREF/PGA\_GAIN \times (((ADCDAT - 0x8000)/2^{15}))) + ADCVBIAS\_CAP \quad (3)$$

where:
VREF = 1.82 V (typical).
ADCDAT is the ADC conversion result.
ADCVBIAS_CAP = 1.11 V (typical).
PGA_GAIN is the PGA gain setting minus one. This gain is set by ADCCON, Bits[18:16].

ADC LOW-POWER CURRENT INPUT CHANNELS

Figure 10 shows the low-power TIA0 input current channel, low-power TIA0. The low-power TIA1 input current channel, low-power TIA1, is identical to low-power TIA0. The output of the low-power TIA is the voltage proportional to the input current measured by the ADC. Details on how to configure the R_{LOAD0} resistor, R_{TIA0} resistor, and low-pass filter programmable resistor (R_{FILTER}) values can be found in the Low-Power TIAs section.

To select the low-power TIA input channel for calibration and measurement, refer to ADCCON, Bits[12:0] in Table 63. For low-power TIA0, the ADC positive input is selected by setting ADCCON, Bits[5:0] = 0b000010. For low power TIA1, the ADC positive input is selected by setting ADCCON, Bits[5:0] = 0b000011.

Figure 10 also shows the low-power DAC (V_{ZERO}) signal controlling the low-power TIA positive input voltage level.

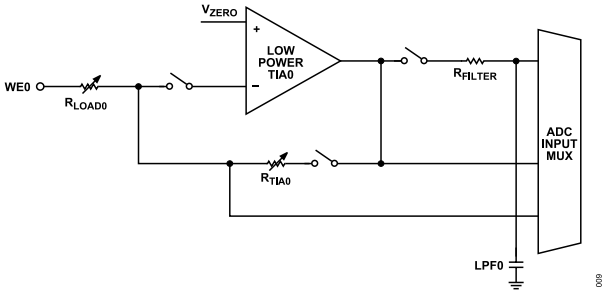

Figure 10. Low-Power TIA0 Current Input Channel to the ADC

The low-power TIA outputs have a low-pass filter. The resistor connecting the TIA output to the input mux is typically 1 MΩ with a recommended external low-pass filter capacity of 4.7 µF resulting in a very low cutoff frequency. This resistor is labeled RFILTER in Figure 10. For more details on how to set up the sensor bias voltage, see the Low-Power DACs section.

ADC INPUT CIRCUIT

Figure 8 shows a basic view of the ADC input stages from the ADC mux to the ADC.

The ADC input mux is programmable to select from up to 34 positive input channels. Features of the input mux include the following:

- The positive input can be selected via ADCCON, Bits[5:0].
- The negative input is nominally expected to be the 1.11 V reference source, selected via ADCCON, Bits[12:8] = 0b01000.
- An optional PGA can be selected to amplify the positive voltage input. The PGA or instrumentation amplifier are enabled via AFECON, Bit 10. The gain setting is configured via ADCCON, Bits[18:16].

ADC CIRCUIT

- The output of the gain stage goes through an antialias filter. The cutoff frequency of the antialias filter is set by PMBW, Bits[3:2]. Set the cutoff frequency to suit the input signal bandwidth. See the ADC, High-Speed DAC, and Associated Amplifiers Operating Mode Configuration section for more details.

The ADC output code is calibrated with an offset and gain correction factor. This digital adjustment factor occurs automatically. The offset and gain correction register used depends on the ADC input channel selected. See the ADC Calibration section for more details.

ADC POSTPROCESSING FILTER OPTIONS

Figure 11 shows an overview of the postprocessing options for ADC results.

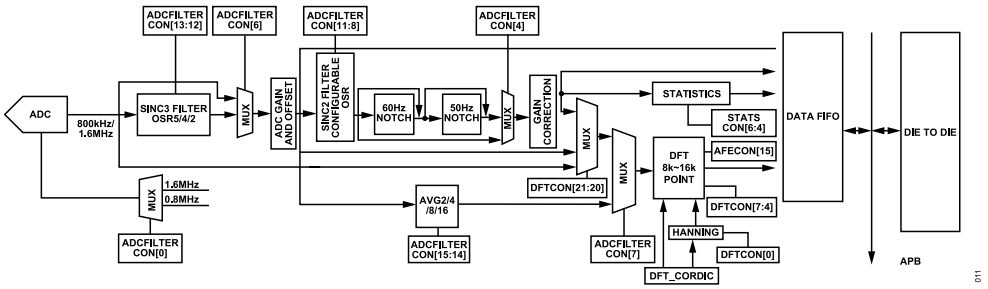

Figure 11. ADC Postprocessing Filter Options

Sinc3 Filter

The input is the raw ADC codes at a rate of 800 kHz if the 16 MHz oscillator is selected on the analog die, or 1.6 MHz if the 32 MHz oscillator is selected on the analog die.

To enable the sinc3 filter, ensure that ADCFILTERCON, Bit 6 = 0. The filter decimation rate is programmable. The recommended decimation or oversampling rate (OSR) is 4. The OSR is controlled by ADCFILTERCON, Bits[13:12]. The input rate is 200 kSPS in normal mode if the 16 MHz oscillator is selected, and 400 kSPS in high-speed mode if the 32 MHz oscillator is selected.

If selected, the sinc3 filter output can be read via the ADCDAT register. When changing the ADC input channels or when changing the ADC update rates via ADCFILTERCON, the user must reset the sinc3 filter. If sinc3 filter is not reset, ADC samples with the new settings are inaccurate.

The following are example instructions:

```
pADI_AFE->AFECON &= (~(BITM_AFE_AFECON_ADCCONVEN)); // Clear AFECON[8]
pADI_AFE->AFECON |= BITM_AFE_AFECON_ADCCONVEN; // Set AFECON[8]
```

Sinc2 Filter

The input is the gain or offset adjusted codes from the sinc3 filter or from the ADC. The input rate is 800 kSPS if coming directly from the ADC when the 16 MHz oscillator is selected, 1.6 MSPS if coming directly from the ADC when the 32 MHz oscillator is selected, and 400 kSPS or 160 kSPS if coming directly from the sinc3 filter output.

To enable the sinc2 filter, set AFECON, Bit 16 = 1. The output rate from the sinc2 filter is programmable via ADCFILTERCON, Bits[11:8]. The decimation rate is a minimum of divide by 22 to a maximum of divide by 1333. If selected, the sinc2 filter output can be read via the SINC2DAT register.

When changing ADC input channels, restarting the ADC, or changing the ADC update rates via the ADCFILTERCON register, reset the sinc2 filter. If the sinc2 filter is not reset, the ADC samples with the new settings are inaccurate.

The following are example instructions:

```
pADI_AFE->AFECON &= (~(BITM_AFE_AFECON_SINC2EN)); // Clear AFECON[16]
pADI_AFE->AFECON |= BITM_AFE_AFECON_SINC2EN; // Set AFECON[16]
```

ADC CIRCUIT

Power Supply Rejection Filter (50 Hz or 60 Hz Mains Filter)

To enable the 50 Hz or 60 Hz notch filter for filtering mains noise, clear ADCFILTERCON, Bit 4 = 0 and set AFECON, Bit 16 = 1. The input is the sinc2 filter output. The input rate is dependent on the sinc3 and sinc2 settings. If selected, the power supply rejection filter output can be read via the SINC2DAT register. Table 53 describes the digital filter settings that support simultaneous 50 Hz or 60 Hz mains rejection.

Table 53. Digital Filter Settings for 50 Hz or 60 Hz Mains Rejection

ADCFILTERCON Bits[13:8] Value	Power Mode (PMBW Bit 0)	ADC Clock Setting (MHz)	Sinc3 Oversampling Setting	Sinc2 Oversampling Setting	Final ADC Output Update Rate in Samples per Second	Filter Settling Time (ms)
0b000011	0 (low-power mode)	16	5	178	900	37
0b100111	0 (low-power mode)	16	2	667	600	37
0b101011	0 (low-power mode)	16	2	1333	300	37
0b101011	1 (high-power mode)	32	2	1333	600	37

Gain Correction

The gain correction of the sinc2 filter and power supply rejection filter occurs in the gain correction block. The block is automatically enabled in hardware, and no user configuration is required.

Digital DFT

The DFT accelerator is intended for use during impedance measurements. To enable the DFT block, set AFECON, Bit 15 = 1. The input can be the raw ADC results, sinc2 output, sinc3 output (ADCDAT register), or the power supply filter output (SINC2DAT register).

Configure the DFT using DFTCON, Bits[21:20]. The DFT outputs a complex number (real and imaginary terms) that represents the overall DFT result for the selected number of ADC samples of the applied AC waveform. See the DFT Result, Real Part Register section and the DFT Result, Imaginary Part Register section. The number of samples used by the DFT is configurable via DFTCON, Bits[7:4] (see the AFE DSP Configuration Register section). A Hanning window (raised cosine window) option is available. To enable the Hanning window, set DFTCON, Bit 0 to 1. If enabled, values outside the selected interval are set to 0. It is recommended to enable the Hanning window.

AVERAGING, STATISTICS, AND OUTLIER DETECTION OPTIONS

Averaging Option

The ADuCM355 supports averaging of the sinc3 output. The number of samples to average is configured via ADCFILTERCON, Bits[15:14].

Statistics Option

The ADuCM355 supports the calculation of the mean value for a programmable sample size of the sinc3 output. This calculation is controlled by the STATSCON register. The number of samples used for statistics is configured via STATSCON, Bits[6:4].

Outlier Detection Options

The ADuCM355 provides outlier detection. Use the ADCMIN and ADCMAX registers to trigger an interrupt if the ADCDAT result from the ADC is outside the limits of the value in the ADCMIN and ADCMAX registers.

The hysteresis values are also programmable. For more details, see the Minimum Value Check Register section, the Maximum Value Check Register section, and the Delta Check Register section.

INTERNAL TEMPERATURE SENSOR CHANNELS

The ADuCM355 analog die contains two internal temperature sensor channels.

ADC CIRCUIT

Temperature Sensor 0

The temperature sensor outputs a voltage proportional to die temperature. This voltage is linear relative to temperature. This internal channel is measured via the ADC by selecting the temperature sensor channels as the positive and negative inputs from the mux. The die temperature is calculated by the following:

$$(TEMPSENSDAT0/(PGA\ Gain \times K)) - 273.15 \quad (4)$$

where $K = 8.13$.

For improved accuracy, configure the temperature sensor in chop mode via TEMPCON0, Bits[3:1]. If chopping is selected, the user must ensure an even number of ADC conversions take place on the temperature sensor channel and that these results are averaged.

Dedicated calibration registers for the temperature sensor channel are also available. When the ADC selects the temperature sensor as the positive input, the calibration values in the ADCOFFSETTEMPSENS0 and ADCGAINTEMPSENS0 registers are automatically used.

To enable the internal temperature sensor, set AFECON, Bit 12 = 1. Select ADC input channels as follows:

- ADCCON, Bits[12:8] = 01011 to select the ADC negative input channel.
- ADCCON, Bits[5:0] = 001011 to select the positive input channel.

To start an ADC conversion of the temperature sensor channel, set AFECON, Bit 13 and AFECON, Bit 8 to 1. For optimal temperature sensor results, enable chop mode of the temperature sensor with the 6.25 kHz chopping frequency. Then, average an even number of ADC temperature sensor results to eliminate any inaccuracies due to the chopping clock.

The following code snippets demonstrate how to set up and use the Temperature Sensor 0:

```
void InitAfeADC(void)
{
   AfeAdcPwrUp(BITM_AFE_AFECON_ADCEN);          // power up the ADC
   AfeAdcCfg(GNPGA_1_5, 0);                    // Configure ADC to measure Temp. output
   AfeAdcChan(MUXSELP_TEMP,MUXSELN_TEMPN);      // Select Temp sensor inputs to the ADC
   AfeAdcIntCfg(
            BITM_AFE_ADCINTIEN_ADCRDYIEN| // Select ADCReady and Temp sensor channels as interrupt
sources
       BITM_AFE_ADCINTIEN_TEMPRDYIEN);
   pADI_AFE->AFECON |=
     BITM_AFE_AFECON_TEMPSENSEN0;               // Enable the ADC temp sensor channel
   pADI_AFE-> TEMPCON0 = 0x2;                              // Turn on Temp sensor chopping with
6.25kHz chop clock
   pADI_AFE->REPEATADCCNV = 0x11;
   NVIC_EnableIRQ(AFE_ADC_IRQn);               // Enable AFE_ADC interrupt source in NVIC
}
// ADC Interrupt handler
void AfeAdc_Int_Handler()
{
    uiIntSta = AfeAdcIntSta();
    if (uiIntSta & BITM_AFE_ADCINTSTA_ADCRDY)
    {
    }
     if (uiIntSta & BITM_AFE_ADCINTSTA_TEMPRDY) // Check for Temp sensor flag
    {
      TEMP_RESULT= AfeAdcRd(TEMPSENSOR);
       pADI_AFE->ADCINTSTA |=
         BITM_AFE_ADCINTSTA_TEMPRDY;           // Clear interrupt flag
        }
    }
void CalculateTemp(void)  // PGA GAIN of 1.5x assumed
{
```

ADC CIRCUIT

```
        fTemp = (float)(( TEMP_RESULT/(1.5*8.13))-273.15);  // ((Temperature reading/
(PGA_Gain*8.13)-273.15
}
```

Temperature Sensor 1

A second backup temperature sensor is provided on the ADuCM355 analog die for functional safety purposes. Only use Temperature Sensor 1 to crosscheck the Temperature Sensor 0 channel. To calculate the die temperature with the Temperature Sensor 1 channel, 21 different ADC measurements are required.

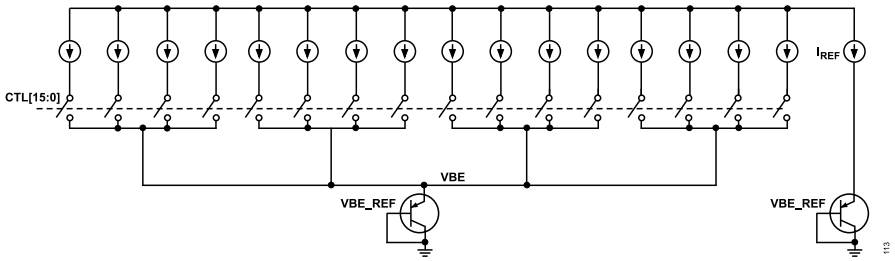

Figure 12. Temperature Sensor 1 Channel Overview

In Figure 12, the base emitter voltage (VBE) of the transistor is connected to 16 different current sources. The transistor labeled VBE_REF is a reference transistor. The base emitter voltage of both transistors varies with temperature. This variation is used to calculate the die temperature.

The voltage of VBE is measured relative to the voltage of VBE_REF via the ADC as a differential voltage. The conversion result is placed in the ADCDAT or SINC2DAT register, depending on the digital filter settings used. To select the Temperature Sensor 1 ADC inputs, set the ADCCON register to the following values:

- ADCCON, Bits[12:8] = 010101 selects the VBE_REF voltage as the ADC negative input.
- ADCCON, Bits[5:0] = 001011 selects the VBE voltage as the ADC positive input.

The 16 switches connecting different current sources to the emitter terminal of the VBE transistor are controlled by the TEMPCON1 Bits[15:0]. Providing separate current sources to the VBE transistor means that a more accurate base emitter voltage can be extracted. To extract the base emitter voltage, take the following steps:

1. Close each switch individually and measure these 16 VBE voltages individually. Calculate the average of 16 measurements to give an overall voltage (VBE1).
2. Close the switches in groups of four and measure these four VBE voltages individually. Calculate the average of four measurements to give an overall voltage (VBE2).
3. Close all switches to connect all current sources to the VBE transistor and measure the VBE voltage to obtain VBE3.

Equation 5 calculates the final temperature sensor value. The value 2768.231 is calculated using the charge of an electron (1.602×10^{-19} C) and the Boltzmann constant (1.381×10^{-23}).

$$T(°C) = 2768.231 \times (4(VBE2 - VBE1) - (VBE3 - VBE2)) - 273.15 \quad (5)$$

ADC INITIALIZATION

The following steps are required to initialize the ADC:

1. Power up the ADC by writing to the ADCEN bit of the AFECON register.
2. Configure the ADC input buffers as per the recommendations for low-power or high-power mode.

```
pADI_AFE->ADCBUFCON = 0x005F3D04; // for Low Power mode, input signals <80 kHz
pADI_AFE->ADCBUFCON = 0x005F3D0F; // for high Power mode, input signals >80 kHz
```

3. Configure the PGA gain setting as required. Write to ADCCON, Bits[18:16]. Setting these bits to 001 configures the PGA for a gain of 1.5.
4. Configure the ADC update rate by writing to ADCFILTERCON. If the sinc2 filter is required, also write to AFECON, Bit 16.
5. Select the ADC positive and negative input channels by configuring ADCCON, Bits[12:0].

ADC CIRCUIT

6. If ADC interrupts are required, enable the interrupts by setting the required ADC bits in the ADCINTIEN register. Enable the ADC global interrupt to the Cortex-M3 core.

ADC CALIBRATION

Because of the multiple input types of the ADuCM355, there are multiple offset and gain calibration options. The ADC must be recalibrated when switching from low-power mode to high-power mode, regardless of the gain change. For optimal performance, calibrate the ADC in low-power mode and high-power mode, if both power modes are used. An error occurs in the high-speed TIA offset and gain when switching from low power to high power. Table 54 to Table 57 detail the registers associated with ADC voltage and gain calibration.

The current input channels (low-power TIA0, low-power TIA1, and high-speed TIA) have an extra ADC calibration stage to that detailed in Table 54. The extra stage is determined by the gain error introduced by each TIA gain resistor. When a current channel is selected by the ADC, its calibration involves the use of a voltage measurement port relative to the PGA setting (as detailed in Table 54), and the current selection relates to the TIA channel.

Example functions are provided with the EVAL-ADuCM355QSPZ to demonstrate how to calibrate the ADC.

Table 54. Voltage Channel Offset and Gain Calibration Registers

PGA Gain Setting	Low-Power Mode and High-Power Mode Offset Registers	Low-Power Mode and High-Power Mode Gain Registers
1	ADCOFFSETGN1	ADCGAINGN1
1.5	ADCOFFSETGN1P5	ADCGAINGN1P5
2	ADCOFFSETGN2	ADCGAINGN2
4	ADCOFFSETGN4	ADCGAINGN4
9	ADCOFFSETGN9	ADCGAINGN9

Table 55. Low-Power TIA0 Channel Offset and Gain Calibration Registers

PGA Gain Setting	Offset Registers	Gain Registers
1	ADCOFFSETGN1, ADCOFFSETLPTIA0	ADCGAINGN1, ADCGNLPTIA0
1.5	ADCOFFSETGN1P5, ADCOFFSETLPTIA0	ADCGAINGN1P5, ADCGNLPTIA0
2	ADCOFFSETGN2, ADCOFFSETLPTIA0	ADCGAINGN2, ADCGNLPTIA0
4	ADCOFFSETGN4, ADCOFFSETLPTIA0	ADCGAINGN4, ADCGNLPTIA0
9	ADCOFFSETGN9, ADCOFFSETLPTIA0	ADCGAINGN9, ADCGNLPTIA0

Table 56. Low-Power TIA1 Channel Offset and Gain Calibration Registers, Low-Power Mode Only

PGA Gain Setting	Offset Registers	Gain Registers
1	ADCOFFSETGN1, ADCOFFSETLPTIA1	ADCGAINGN1, ADCGNLPTIA1
1.5	ADCOFFSETGN1P5, ADCOFFSETLPTIA1	ADCGAINGN1P5, ADCGNLPTIA1
2	ADCOFFSETGN2, ADCOFFSETLPTIA1	ADCGAINGN2, ADCGNLPTIA1
4	ADCOFFSETGN4, ADCOFFSETLPTIA1	ADCGAINGN4, ADCGNLPTIA1
9	ADCOFFSETGN9, ADCOFFSETLPTIA1	ADCGAINGN9, ADCGNLPTIA1

Table 57. High-Speed TIA Channel Offset and Gain Calibration Registers

PGA Gain Setting	Offset Registers	Gain Registers
1	ADCOFFSETGN1, ADCOFFSETHSTIA	ADCGAINGN1, ADCGNHSTIA
1.5	ADCOFFSETGN1P5, ADCOFFSETHSTIA	ADCGAINGN1P5, ADCGNHSTIA
2	ADCOFFSETGN2, ADCOFFSETHSTIA	ADCGAINGN2, ADCGNHSTIA
4	ADCOFFSETGN4, ADCOFFSETHSTIA	ADCGAINGN4, ADCGNHSTIA
9	ADCOFFSETGN9, ADCOFFSETHSTIA	ADCGAINGN9, ADCGNHSTIA

When calibrating the gain error for the ADC voltage channels during Analog Devices production testing, the value loaded to the ADCGAINGN1P5 calibration register is ≥0x4000. To ensure this value, the target ADC result is higher than normal. The factory trim value for the ADC reference is 1.82 V, but for calibration purposes, the target voltage is 1.835 V.

When calculating a real voltage from an ADC conversion on a channel using the factory gain calibration, the K factor of 1.835 or 1.82 must be taken into account, as shown in Equation 1. If user ADC gain calibration targets a reference voltage of 1.82 V, then the K portion of Equation 1 is not required.

ADC CIRCUIT

ADC DIGITAL SIGNAL PROCESSOR (DSP) BUILT IN SELF TEST

It is possible to verify the digital logic blocks on the analog die related to the ADC.

The digital waveform generator can be used to create a digital pattern that is connected directly to the output filters of the ADC, bypassing the ADC itself. As a new value is outputted from the digital waveform generator block to the sinc3 digital filter, the digital value is shifted through the calibration block and other digital filter blocks until the digital value reaches the ADC filter result register. This value can then be fed to the cyclic redundancy check (CRC) accelerator block on the digital die.

By completing the self test for a large number of digital waveform values, the CRC accelerator can compute a final CRC result. By comparing this CRC result with a known and previously verified CRC result, all of the digital blocks shown in Figure 13 can be checked for errors. Example code demonstrating how to enable this feature is provided with the EVAL-ADuCM355QSPZ.

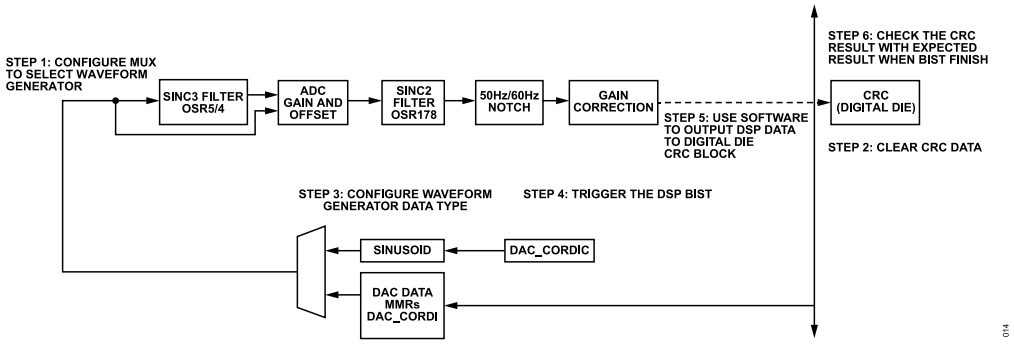

Figure 13. ADC Built In Self Test Feature

VOLTAGE REFERENCE OPTIONS

The ADC features internal 1.82 V and 2.5 V voltage reference sources, as shown in Figure 14.

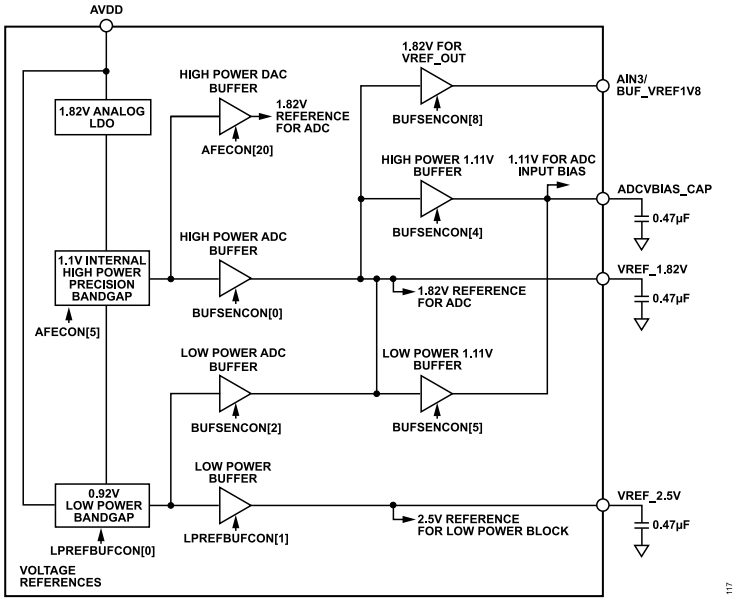

Figure 14. ADC Voltage Reference Options

REGISTER SUMMARY: ADC CIRCUIT

Table 58. ADC Control Register Summary

Address	Name	Description	Reset	Access
0x400C21A8	ADCCON	ADC configuration register	0x00000000	R/W
0x400C2044	ADCFILTERCON	ADC output filters configuration	0x00000301	R/W
0x400C2074	ADCDAT	Raw result	0x00000000	R/W
0x400C2078	DFTREAL	DFT result, real part	0x00000000	R/W
0x400C207C	DFTIMAG	DFT result, imaginary part	0x00000000	R/W
0x400C2080	SINC2DAT	Sinc2 and supply rejection filter result	0x00000000	R/W
0x400C2084	TEMPSENSDAT0	Temperature Sensor 0 result	0x00000000	R/W
0x400C2088	ADCINTIEN	Analog capture interrupt enable	0x00000000	R/W
0x400C2098	ADCINTSTA	Analog capture interrupt	0x00000000	R/W
0x400C20D0	DFTCON	AFE DSP configuration	0x00000090	R/W
0x400C2174	TEMPCON0	Temperature Sensor 0 configuration	0x00000000	R/W
0x400C2180	BUFSENCON	High-power and low-power buffer control	0x00000037	R/W
0x400C21F0	REPEATADCCNV	Number of repeat ADC conversions	0x00000160	R/W
0x400C238C	ADCBUFCON	Buffer configuration	0x005F3D00	R/W

Table 59. ADC Calibration Register Summary

Address	Name	Description	Reset	Access
0x400C2230	CALDATLOCK	Calibration lock	0x00000000	R/W
0x400C2288	ADCOFFSETLPTIA0	Offset calibration low-power TIA0 channel	0x00000000	R/W
0x400C228C	ADCGNLPTIA0	Gain calibration for low-power TIA0 channel	0x00004000	R/W
0x400C22C0	ADCOFFSETLPTIA1	Offset calibration low-power TIA1 channel	0x00000000	R/W
0x400C22C4	ADCGNLPTIA1	Gain calibration for low-power TIA1 channel	0x00004000	R/W
0x400C2234	ADCOFFSETHSTIA	Offset calibration high-speed TIA channel	0x00000000	R/W
0x400C2284	ADCGNHSTIA	Gain calibration for high-speed TIA channel	0x00004000	R/W
0x400C2244	ADCOFFSETGN1	Offset calibration voltage channel (PGA gain = 1)	0x00000000	R/W
0x400C2240	ADCGAINGN1	Gain calibration voltage input channel (PGA gain = 1)	0x00004000	R/W
0x400C22CC	ADCOFFSETGN1P5	Offset calibration voltage input channel (PGA gain = 1.5)	0x00000000	R/W
0x400C2270	ADCGAINGN1P5	Gain calibration voltage input channel (PGA gain = 1.5)	0x00004000	R/W
0x400C22C8	ADCOFFSETGN2	Offset calibration voltage input channel (PGA gain = 2)	0x00000000	R/W
0x400C2274	ADCGAINGN2	Gain calibration voltage input channel (PGA gain = 2)	0x00004000	R/W
0x400C22D4	ADCOFFSETGN4	Offset calibration voltage input channel (PGA gain = 4)	0x00000000	R/W
0x400C2278	ADCGAINGN4	Gain calibration voltage input channel (PGA gain = 4)	0x00004000	R/W
0x400C22D0	ADCOFFSETGN9	Offset calibration voltage input channel (PGA gain = 9)	0x00000000	R/W
0x400C2298	ADCGAINGN9	Gain calibration voltage input channel (PGA gain = 9)	0x00004000	R/W
0x400C223C	ADCOFFSETTEMPSENS0	Offset calibration temperature sensor Channel 0	0x00000000	R/W
0x400C2238	ADCGAINTEMPSENS0	Gain calibration temperature sensor Channel 0	0x00004000	R/W

Table 60. ADC Digital Postprocessing Register Summary (Optional)

Address	Name	Description	Reset	Access
0x400C20A8	ADCMIN	Minimum value check	0x00000000	R/W
0x400C20AC	ADCMINSM	Minimum slow moving value	0x00000000	R/W
0x400C20B0	ADCMAX	Maximum value check	0x00000000	R/W
0x400C20B4	ADCMAXSMEN	Maximum slow moving	0x00000000	R/W
0x400C20B8	ADCDELTA	Delta check	0x00000000	R/W

Table 61. ADC Statistics Register Summary (Optional)

Address	Name	Description	Reset	Access
0x400C21C4	STATSCON	Statistics module configuration	0x00000000	R/W
0x400C21C8	STATSMEAN	Mean output	0x00000000	R

REGISTER SUMMARY: ADC CIRCUIT

Table 62. ADC Digital Logic Test Register Summary (Optional)

Address	Name	Description	Reset	Access
0x400C0434	MKEY	Key access for DSPUPDATEEN register	0x00000000	W
0x400C0438	DSPUPDATEEN	Digital logic test enable	0x00000000	R/W
0x400C2374	TEMPCON1	Temperature Sensor 1 control	0x00020000	R/W

REGISTER DETAILS: ADC CIRCUIT

ADC CONFIGURATION REGISTER

Address: 0x400C21A8, Reset: 0x00000000, Name: ADCCON

Table 63. Bit Descriptions for ADCCON

Bits	Bit Name	Settings	Description	Reset	Access
[31:19]	Reserved		Reserved.	0x0	R
[18:16]	GNPGA		PGA Gain Setup.	0x0	R/W
		000	Gain = 1.		
		001	Gain = 1.5.		
		010	Gain = 2.		
		011	Gain = 4.		
		1XX	Gain = 9.		
15	GNOFSELPGA		Internal Offset/Gain Cancellation.	0x0	R/W
		0	DC offset cancellation disabled.		
		1	Enable DC offset cancellation. When PGA is enabled, only gain = 4 supported.		
[14:13]	Reserved		Reserved.	0x0	R/W
[12:8]	MUXSELN		Select Signals for the ADC Input Multiplexer N Input (Negative Input Signal).	0x0	R/W
		00000	Floating input.		
		00001	High-speed TIA inverting input.		
		00010	Low-power TIA0 inverting input.		
		00011	Low-power TIA1 inverting input.		
		00100	AIN0.		
		00101	AIN1.		
		00110	AIN2.		
		00111	AIN3/BUF_VREF1V8.		
		01000	ADCVBIAS_CAP.		
		01001	Reserved.		
		01010	Reserved.		
		01011	Temperature Sensor 0 negative input.		
		01100	AIN4_LPF0.		
		01101	AIN5.		
		01110	AIN6.		
		01111	Reserved.		
		10000	VZERO0.		
		10001	VBIAS0.		
		10010	VZERO1.		
		10011	VBIAS1.		
		10100	N node of excitation amplifier.		
		10101	Temperature Sensor 1 negative input.		
		10110	Test signal.		
[7:6]	Reserved		Reserved.	0x0	R
[5:0]	MUXSELP		Select Positive Mux (Positive ADC Input).	0x0	R/W
		00000	Floating input.		
		00001	High-speed TIA output.		
		00010	Low-power TIA0 output after the low-pass filter.		
		00011	Low-power TIA1 output after the low-pass filter.		
		00100	AIN0.		
		00101	AIN1.		
		00110	AIN2.		
		00111	AIN3/BUF_VREF1V8.		

REGISTER DETAILS: ADC CIRCUIT

Table 63. Bit Descriptions for ADCCON (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		01000	AVDD/2.		
		01001	DVDD/2.		
		01010	AVDD_REG/2.		
		01011	Temperature Sensor 0 positive input.		
		01100	ADCVBIAS_CAP.		
		01101	DE0.		
		01110	SE0.		
		01111	SE1.		
		010000	VREF_2.5V/2. Low power 2.5 V reference divided by 2.		
		010001	Reserved.		
		010010	VREF_1.82V. ADC and high-speed DAC 1.82 V voltage channel.		
		010011	Temperature Sensor 0 negative input.		
		010100	AIN4_LPF0.		
		010101	AIN5.		
		010110	AIN6.		
		010111	VZERO0.		
		011000	VBIAS0.		
		011001	CE0 pin voltage.		
		011010	RE0 pin voltage.		
		011011	VZERO1.		
		011100	VBIAS1.		
		011101	CE1 pin voltage.		
		011110	RE1 pin voltage.		
		011111	CE0 pin divided by 2 voltage.		
		100000	CE1 pin divided by 2 voltage.		
		100001	LPTIA0 output before the low-pass filter.		
		100010	LPTIA1 output before the low-pass filter.		
		100011	Reserved.		
		100100	P node of excitation amplifier.		
		100101	Temperature Sensor 1 positive input.		
		100110	Test signal.		

ADC OUTPUT FILTERS CONFIGURATION REGISTER

Address: 0x400C2044, Reset: 0x00000301, Name: ADCFILTERCON

Table 64. Bit Descriptions for ADCFILTERCON

Bits	Bit Name	Settings	Description	Reset	Access
[31:16]	Reserved		Reserved.	0x0	R
[15:14]	AVRGNUM		These bits set the number of samples used by the averaging function. The average output is fed directly to the DFT block and the DFT source is automatically changed to the average output. Set the AVRGEN bit to 1 to use these bits.	0x0	R/W
		0	2 ADC samples used for the average function.		
		1	4 ADC samples used for the average function.		
		10	8 ADC samples used for the average function.		
		11	16 ADC samples used for the average function.		
[13:12]	SINC3OSR		Sinc3 Filter Oversampling Rate.	0x0	R/W
		0	Oversampling rate of 5. Use for 160 kHz sinc3 filter output update rate. Use when ADC update rate is 800 kSPS (default).		
		1	Oversampling rate of 4. Use for 400 kHz sinc3 filter output update rate. Use when ADC update rate is 1.6 MSPS. High-power option.		

REGISTER DETAILS: ADC CIRCUIT

Table 64. Bit Descriptions for ADCFILTERCON (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		10	Oversampling rate of 2. Use for 400 kHz sinc3 filter output update rate. Use when ADC update rate is 800 kSPS.		
		11	Reserved. Do not use this setting.		
[11:8]	SINC2OSR		Sinc2 Filter Oversampling Rates.	0x3	R/W
		0	22 samples for this OSR setting.		
		1	44 samples for this OSR setting.		
		10	89 samples for this OSR setting.		
		11	178 samples for this OSR setting.		
		100	267 samples for this OSR setting.		
		101	533 samples for this OSR setting.		
		110	640 samples for this OSR setting.		
		111	667 samples for this OSR setting.		
		1000	800 samples for this OSR setting.		
		1001	889 samples for this OSR setting.		
		1010	1067 samples for this OSR setting.		
		1011	1333 samples for this OSR setting.		
7	AVRGEN		Enable ADC Average Function.	0x0	R/W
		0	Disable average.		
		1	Enable average. Average result feeds to next stage.		
6	SINC3BYP		Sinc3 Filter Bypass.	0x0	R/W
		0	Sinc3 filter active. Enable sinc3 filter.		
		1	Bypass sinc3 filter. Raw 800 kHz or 1.6 MHz ADC output data is fed directly to gain offset adjustment stage. If the sinc3 filter is bypassed, the 200 kHz sine wave can be handled directly by DFT block without amplitude attenuation. If the sinc3 filter is bypassed and ADC raw data rate is 800 kHz, the gain offset block output is used as DFT input.		
5	Reserved		Reserved.	0x0	R
4	LPFBYPEN		50 Hz or 60 Hz Low-Pass Filter. Bypass both 50 Hz and 60 Hz notch filter.	0x0	R/W
		1	Bypass both 50 Hz notch and 60 Hz notch filters.		
		0	Enable 50 Hz notch and 60 Hz notch filters. ADC result is written to the SINC2DAT register.		
[3:1]	Reserved		Reserved.	0x0	R
0	ADCCLK		ADC Data Rate. Unfiltered ADC output rate.	0x1	R/W
		1	800 kHz.		
		0	1.6 MHz. If ADC sample rate is 1.6 MHz, ADC clock frequency must be 32 MHz.		

RAW RESULT REGISTER

Address: 0x400C2074, Reset: 0x00000000, Name: ADCDAT

This register is the ADC result register for raw ADC output or when sinc3 filter options are selected.

Table 65. Bit Descriptions for ADCDAT

Bits	Bit Name	Settings	Description	Reset	Access
[31:16]	Reserved		Reserved.	0x0	R
[15:0]	DATA		ADC Result. Register contains the ADC conversion result. Depending on user configuration, result can reflect raw or sinc3 filter outputs. The result is a 16-bit unsigned number.	0x0	R/W

REGISTER DETAILS: ADC CIRCUIT

DFT RESULT, REAL PART REGISTER

Address: 0x400C2078, Reset: 0x00000000, Name: DFTREAL

Table 66. Bit Descriptions for DFTREAL

Bits	Bit Name	Settings	Description	Reset	Access
[31:18]	Reserved		Reserved.	0x0	R
[17:0]	DATA		DFT Real. DFT hardware accelerator returns a complex number. This register returns the 18-bit real part of the complex number from the DFT result. The DFT result is represented in twos complement.	0x0	R/W

DFT RESULT, IMAGINARY PART REGISTER

Address: 0x400C207C, Reset: 0x00000000, Name: DFTIMAG

Table 67. Bit Descriptions for DFTIMAG

Bits	Bit Name	Settings	Description	Reset	Access
[31:18]	Reserved		Reserved.	0x0	R
[17:0]	DATA		DFT Imaginary. DFT hardware accelerator returns a complex number. This register returns the 18-bit imaginary part of the complex number from the DFT result. DFT result is represented in twos complement.	0x0	R/W

SINC2 AND SUPPLY REJECTION FILTER RESULT REGISTER

Address: 0x400C2080, Reset: 0x00000000, Name: SINC2DAT

Table 68. Bit Descriptions for SINC2DAT

Bits	Bit Name	Settings	Description	Reset	Access
[31:16]	Reserved		Reserved.	0x0	R
[15:0]	RESULT		Sinc2 and Low-Pass Filter Result. Sinc2 and power supply rejection filter, ADC output result. Data output from 50 Hz or 60 Hz rejection filter. When new data is available, ADCINTSTA, Bit 2 is set to 1.	0x0	R/W

TEMPERATURE SENSOR 0 RESULT REGISTER

Address: 0x400C2084, Reset: 0x00000000, Name: TEMPSENSDAT0

Table 69. Bit Descriptions for TEMPSENSDAT0

Bits	Bit Name	Settings	Description	Reset	Access
[31:16]	Reserved		Reserved.	0x0	R
[15:0]	DATA		Temperature Sensor. ADC temperature sensor Channel 0 result.	0x0	R/W

ANALOG CAPTURE INTERRUPT ENABLE REGISTER

Address: 0x400C2088, Reset: 0x00000000, Name: ADCINTIEN

Table 70. Bit Descriptions for ADCINTIEN

Bits	Bit Name	Settings	Description	Reset	Access
[31:8]	Reserved		Reserved.	0x0	R
7	MEANIEN		Mean Interrupt. Mean result ready interrupt enable.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
6	ADCDELTAFAILIEN		ADC Delta Value Check Fail Interrupt Enable. When set, this bit generates an interrupt if the difference between two consecutive ADC samples is greater than the value in the ADC delta range.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
5	ADCMAXFAILIEN		ADC Maximum Value Check Fail Interrupt Enable. When set, this bit generates an interrupt if the ADC result is greater than the value in the ADCMAX register.	0x0	R/W
		0	Interrupt disabled.		

REGISTER DETAILS: ADC CIRCUIT

Table 70. Bit Descriptions for ADCINTIEN (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		1	Interrupt enabled.		
4	ADCMINFAILIEN		ADC Minimum Value Check Fail Interrupt Enable. When set, this bit generates an interrupt if the ADC result is less than the value in the ADCMIN register.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
3	TEMPRDYIEN		Temperature Sensor 0 ADC Result Ready Interrupt Enable. The TEMPSENSDAT0 register is ready for reading.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
2	SINC2RDYIEN		Low-Pass Filter Result Interrupt. Supply rejection filter result ready for interrupt enable. The SINC2DAT register is ready for reading.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
1	DFTRDYIEN		DFT Result Ready Interrupt. The DFTREAL and DFTIMAG registers are ready for reading.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
0	ADCRDYIEN		ADC Result Ready Interrupt Enable. The ADCDAT register is ready for reading.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		

ANALOG CAPTURE INTERRUPT REGISTER

Address: 0x400C2098, Reset: 0x00000000, Name: ADCINTSTA

The bits in this register are sticky when set. Each bit is cleared by writing a 1 to its location. Writing a 0 has no effect. If simultaneously the interrupt source is asserted and the core is attempting to clear a bit, the interrupt remains set. A read of 1 means the interrupt source has been asserted since the last time the bit was cleared. A read of 0 means the interrupt source has not been asserted since the last time the bit was cleared.

Table 71. Bit Descriptions for ADCINTSTA

Bits	Bit Name	Settings	Description	Reset	Access
[31:8]	Reserved		Reserved.	0x0	R
7	MEANRDY		Mean Result Ready. If STATSCON, Bit 0 is set to 1, this bit indicates the status of the STATSMEAN register. The user must write 1 to this bit to clear it. Writing 0 has no effect.	0x0	R/W1C
		0	Interrupt not asserted.		
		1	Interrupt asserted. The STATSMEAN register is ready for reading. This bit generates an interrupt if ADCINTIEN, Bit 7 = 1.		
6	ADCDIFFERR		ADC Delta Ready. ADC delta value check fail. User must write 1 to this bit to clear it. Writing 0 has no effect.	0x0	R/W1C
		0	Interrupt not asserted.		
		1	Interrupt asserted. When set, this bit indicates that the difference between two consecutive ADCDAT results was greater than the value specified by the ADCDELTA register. This bit generates an interrupt if ADCINTIEN, Bit 6 = 1.		
5	ADCMAXERR		ADC Maximum Value Check Fail. User must write 1 to this bit to clear it. Writing 0 has no effect.	0x0	R/W1C
		0	Interrupt not asserted.		
		1	Interrupt asserted. When set, indicates that ADCDAT result was greater than the value specified by the ADCMAX register. This bit generates an interrupt if ADCINTIEN, Bit 5 = 1. User must write 1 to this bit to clear it. Writing 0 has no effect.		
4	ADCMINERR		ADC Minimum Value Check Fail. Writing 0 has no effect.	0x0	R/W1C
		0	Interrupt not asserted.		
		1	Interrupt asserted. When set, indicates ADCDAT result was less than the value specified by ADCMIN. This bit generates an interrupt if ADCINTIEN, Bit 4 = 1.		

REGISTER DETAILS: ADC CIRCUIT

Table 71. Bit Descriptions for ADCINTSTA (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
3	TEMPRDY		Temperature Sensor Result Ready Interrupt. User must write 1 to this bit to clear it. Writing 0 has no effect.	0x0	R/W1C
		0	Interrupt not asserted.		
		1	Interrupt asserted. When set, indicates TEMPSENSDAT0 result is ready for reading. This bit generates an interrupt if ADCINTIEN, Bit 3 = 1.		
2	SINC2RDY		Low-Pass Filter Result Status. Supply rejection filter result ready interrupt. User must write 1 to this bit to clear it. Writing 0 has no effect.	0x0	R/W1C
		0	Interrupt not asserted.		
		1	Interrupt asserted. When set, indicates SINC2DAT result is ready for reading. This bit generates an interrupt if ADCINTIEN, Bit 2 = 1.		
1	DFTRDY		DFT Result Ready Status. DFT result ready interrupt. User must write 1 to this bit to clear it. Writing 0 has no effect.	0x0	R/W1C
		0	Interrupt not asserted.		
		1	Interrupt asserted. When set, indicates DFTREAL and DFTIMAG registers are ready for reading. This bit generates an interrupt if ADCINTIEN, Bit 1 = 1.		
0	ADCRDY		ADC Result Ready Status. ADC result ready interrupt. User must write 1 to this bit to clear it. Writing 0 has no effect.	0x0	R/W1C
		0	Interrupt not asserted.		
		1	Interrupt asserted. When set, indicates ADCDAT register is ready for reading. This bit generates an interrupt if ADCINTIEN, Bit 0 = 1.		

AFE DSP CONFIGURATION REGISTER

Address: 0x400C20D0, Reset: 0x00000090, Name: DFTCON

Table 72. Bit Descriptions for DFTCON

Bits	Bit Name	Settings	Description	Reset	Access
[31:22]	Reserved		Reserved.	0x0	R
[21:20]	DFTINSEL		DFT Input Select. ADCFILTERCON, Bit 7 is the highest priority. If ADCFILTERCON, Bit 7 is 1, output of average block is used as the DFT input, regardless of the DFTINSEL setting.	0x0	R/W
		00	Supply filter output. Select output from low-pass supply filter.		
		01	Gain offset output with or without sinc3. Select output from ADC gain and offset correction stage. If ADCFILTERCON, Bit 6 is 1 (filter is bypassed), ADC raw data through gain or offset correction is the DFT input. If the SINC3BYP bit in the ADCFILTERCON register is 0 (filter is not bypassed), the sinc3 output through gain or offset correction is the DFT input.		
		10	ADC raw data. Select output direct from the ADC, no offset or gain correction. Only supported for 800 kHz sample rate of ADC.		
		11	Supply filter output. Select output from low-pass supply filter. Same as 00.		
[19:8]	Reserved		Reserved.	0x0	R
[7:4]	DFTNUM		ADC Samples Used. DFT number is 4 to 16,384.	0x9	R/W
		0	DFT point number is 4. DFT uses 4 ADC samples.		
		1	DFT point number is 8. DFT uses 8 ADC samples.		
		10	DFT point number is 16. DFT uses 16 ADC samples.		
		11	DFT point number is 32. DFT uses 32 ADC samples.		
		100	DFT point number is 64. DFT uses 64 ADC samples.		
		101	DFT point number is 128. DFT uses 128 ADC samples.		
		110	DFT point number is 256. DFT uses 256 ADC samples.		
		111	DFT point number is 512. DFT uses 512 ADC samples.		
		1000	DFT point number is 1024. DFT uses 1024 ADC samples.		
		1001	DFT point number is 2048. DFT uses 2048 ADC samples.		
		1010	DFT point number is 4096. DFT uses 4096 ADC samples.		
		1011	DFT point number is 8192. DFT uses 8192 ADC samples.		

REGISTER DETAILS: ADC CIRCUIT

Table 72. Bit Descriptions for DFTCON (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		1100	DFT point number is 16384. DFT uses 16,384 ADC samples.		
[3:1]	Reserved		Reserved.	0x0	R
0	HANNINGEN		Hanning Window Enable.	0x0	R/W
		0	Disable Hanning window.		
		1	Enable Hanning window.		

TEMPERATURE SENSOR 0 CONFIGURATION REGISTER

Address: 0x400C2174, Reset: 0x00000000, Name: TEMPCON0

Table 73. Bit Descriptions for TEMPCON0

Bits	Bit Name	Settings	Description	Reset	Access
[31:4]	Reserved		Reserved.	0x0	R
[3:2]	CHOPFRESEL		Chop Mode Frequency Setting. Sets frequency of chop mode switching.	0x0	R/W
		11	Chop switch frequency is 200 kHz.		
		10	Chop switch frequency is 100 kHz.		
		00	Chop switch frequency is 6.25 kHz.		
		01	Chop switch frequency is 25 kHz.		
1	CHOPCON		Temperature Sensor Chop Mode. Temperature sensor channel chop control signal.	0x0	R/W
		0	Disable chop.		
		1	Enable chop. If chopping is enabled, take two consecutive samples and average the results to obtain a final temperature sensor channel reading. Chopping helps to reduce the offset error associated with this channel.		
0	Reserved		Reserved.	0x0	R/W

HIGH-POWER AND LOW-POWER BUFFER CONTROL REGISTER

Address: 0x400C2180, Reset: 0x00000037, Name: BUFSENCON

Table 74. Bit Descriptions for BUFSENCON

Bits	Bit Name	Settings	Description	Reset	Access
[31:9]	Reserved		Reserved.	0x0	R
8	V1P8THERMSTEN		Buffered Reference Output. Buffered output to the AIN3/BUF_VREF1V8 pin.	0x0	R/W
		0	Disable 1.8 V buffered reference output.		
		1	Enable 1.8 V buffered reference output.		
7	Reserved		Reserved.	0x0	R
6	V1P1LPADCCHGDIS		Controls Decoupling Capacitor Discharge Switch. This switch connects the 1.1 V internal reference for ADC common-mode voltage to an internal discharging circuit. Ensure that the switch is open for normal operation to maintain the reference voltage on the external 1.1 V decoupling capacitor.	0x0	R/W
		0	Open Switch. Recommended value. Leave the switch open to maintain the charge on external decoupling capacitor for the 1.1 V reference.		
		1	Close Switch. Close this switch to connect the 1.1 V reference to the discharging circuit.		
5	V1P1LPADCEN		ADC 1.1 V Low-Power Common-Mode Buffer. Optional. Use either high-power or low-power reference buffer.	0x1	R/W
		0	Disable ADC 1.1 V low-power reference buffer.		
		1	Enable ADC 1.1 V low-power reference buffer.		
4	V1P1HPADCEN		Enable 1.1 V High-Power Common-Mode Buffer. Controls buffer for 1.1 V. Common-mode voltage source to ADC input stage.	0x1	R/W
		0	Disable 1.1 V high-power common-mode buffer.		
		1	Enable 1.1 V high-power common-mode buffer. Recommended value for normal ADC operation.		

REGISTER DETAILS: ADC CIRCUIT

Table 74. Bit Descriptions for BUFSENCON (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
3	V1P8HPADCCHGDIS		Controls Decoupling Capacitor Discharge Switch. This switch connects the 1.8 V internal ADC reference to an internal discharging circuit. Ensure that the switch is open for normal operation to maintain the reference voltage on the external decoupling capacitor.	0x0	R/W
		0	Open switch. If opened, the voltage on the external decoupling capacitor for the reference is maintained. Recommended setting.		
		1	Close Switch. Close this switch to connect the reference to the discharge circuit.		
2	V1P8LPADCEN		ADC 1.8 V Low-Power Reference Buffer.	0x1	R/W
		0	Disable low-power 1.8 V reference buffer.		
		1	Enable low-power 1.8 V reference buffer. Recommended value. Speeds up settling time when exiting power-down states.		
1	V1P8HPADCILIMITEN		High-Power ADC Input Current Limit. Protects ADC input buffer.	0x1	R/W
		0	Disable buffer current limit.		
		1	Enable buffer current limit. Recommended setting.		
0	V1P8HPADCEN		High-Power 1.8 V Reference Buffer. Enable for normal ADC conversions.	0x1	R/W
		0	Disable 1.8 V high-power ADC reference buffer.		
		1	Enable 1.8 V high-power ADC reference buffer.		

NUMBER OF REPEAT ADC CONVERSIONS REGISTER

Address: 0x400C21F0, Reset: 0x00000160, Name: REPEATADCCNV

Table 75. Bit Descriptions for REPEATADCCNV

Bits	Bit Name	Settings	Description	Reset	Access
[31:5]	Reserved		Reserved.	0x00016	R
[4]	NUM		Write 1 to this bit to enable single or continuous conversion.	0x0	R
[3:1]	Reserved		Reserved.	0x0	R
0	EN		Enable Repeat ADC Conversions.	0x0	R/W
		0	Disable repeat ADC conversions.		
		1	Enable repeat ADC conversions.		

BUFFER CONFIGURATION REGISTER

Address: 0x400C238C, Reset: 0x005F3D00, Name: ADCBUFCON

The recommended value for this register is 0x005F3D0F in high-power mode and 0x005F3D04 in low-power mode.

Table 76. Bit Descriptions for ADCBUFCON

Bits	Bit Name	Settings	Description	Reset	Access
[31:4]	Reserved		Reserved.	0x0	R
3	CHOPDIS		Configure Offset Cancellation Buffer Chop.	0x0	R/W
		0	Enable chop.		
		1	Disable chop.		
2	CHOPDIS		Configure ADC Buffer Chop.	0x0	R/W
		0	Enable chop.		
		1	Disable chop.		
1	CHOPDIS		Configure PGA Chop.	0x0	R/W
		0	Enable chop.		
		1	Disable chop.		
0			Configure Front-End Buffer Chop.		
		0	Enable chop.		
		1	Disable chop.		

REGISTER DETAILS: ADC CIRCUIT

CALIBRATION LOCK REGISTER

Address: 0x400C2230, Reset: 0x00000000, Name: CALDATLOCK

Table 77. Bit Descriptions for CALDATLOCK

Bits	Bit Name	Settings	Description	Reset	Access
[31:0]	KEY		Password for Calibration Data Registers. Prevents overwriting of data after the calibration phase.	0x0	R/W
		0xDE87A5AF	Calibration data registers read/write.		

OFFSET CALIBRATION LOW-POWER TIA0 CHANNEL REGISTER

Address: 0x400C2288, Reset: 0x00000000, Name: ADCOFFSETLPTIA0

Table 78. Bit Descriptions for ADCOFFSETLPTIA0

Bits	Bit Name	Settings	Description	Reset	Access
[31:15]	Reserved		Reserved.	0x0	R
[14:0]	VALUE		Offset Calibration for Low-Power TIA0. Represented as a twos complement number. The calibration resolution is 0.25 LSBs of the ADCDAT LSB size.	0x0	R/W
		0x3FFF	4095.75 (maximum positive offset calibration value).		
		0x0001	0.25 (minimum positive offset calibration value).		
		0x0000	0 (no offset adjustment).		
		0x7FFF	−0.25 (minimum negative offset calibration value).		
		0x4000	−4096.0 (maximum negative offset calibration value).		

GAIN CALIBRATION FOR LOW-POWER TIA0 CHANNEL REGISTER

Address: 0x400C228C, Reset: 0x00004000, Name: ADCGNLPTIA0

Table 79. Bit Descriptions for ADCGNLPTIA0

Bits	Bit Name	Settings	Description	Reset	Access
[31:15]	Reserved		Reserved.	0x0	R
[14:0]	VALUE		Gain Error Calibration for Low-Power TIA0. ADC offset correction in TIA measurement mode, represented as a twos complement number.	0x4000	R/W
		0x7FFF	2 (maximum positive gain adjustment).		
		0x4001	1.000061 (minimum positive gain adjustment).		
		0x4000	1.0. ADC result multiplied by 1. No gain adjustment. Default value.		
		0x3FFF	0.999939 (minimum negative gain adjustment).		
		0x2000	0.5. ADC result multiplied by 0.5.		
		0x0001	0.000061 (maximum negative gain adjustment).		
		0x0000	0. Invalid value. Results in an ADC result of 0.		

OFFSET CALIBRATION LOW-POWER TIA1 CHANNEL REGISTER

Address: 0x400C22C0, Reset: 0x00000000, Name: ADCOFFSETLPTIA1

Table 80. Bit Descriptions for ADCOFFSETLPTIA1

Bits	Bit Name	Settings	Description	Reset	Access
[31:15]	Reserved		Reserved.	0x0	R
[14:0]	VALUE		Offset Calibration for Low-Power TIA1. Represented as a twos complement number. The calibration resolution is 0.25 LSBs of the ADCDAT LSB size.	0x0	R/W
		0x3FFF	4095.75 (maximum positive offset calibration value).		
		0x0001	0.25 (minimum positive offset calibration value).		
		0x0000	0 (no adjustment).		
		0x7FFF	−0.25 (minimum negative offset calibration value).		
		0x4000	−4096.0 (maximum negative offset calibration value).		

REGISTER DETAILS: ADC CIRCUIT

GAIN CALIBRATION FOR LOW-POWER TIA1 CHANNEL REGISTER

Address: 0x400C22C4, Reset: 0x00004000, Name: ADCGNLPTIA1

Table 81. Bit Descriptions for ADCGNLPTIA1

Bits	Bit Name	Settings	Description	Reset	Access
[31:15]	Reserved		Reserved.	0x0	R
[14:0]	VALUE		Gain Error Calibration for Low-Power TIA1. ADC offset correction in TIA measurement mode, represented as a twos complement number.	0x4000	R/W
		0x7FFF	2 (maximum positive gain adjustment).		
		0x4001	1.000061 (minimum positive gain adjustment).		
		0x4000	1.0. ADC result multiplied by 1. No gain adjustment. Default value.		
		0x3FFF	0.999939 (minimum negative gain adjustment).		
		0x2000	0.5. ADC result multiplied by 0.5.		
		0x0001	0.000061 (maximum negative gain adjustment).		
		0x0000	0. Invalid value. Results in ADC result of 0.		

OFFSET CALIBRATION HIGH-SPEED TIA CHANNEL REGISTER

Address: 0x400C2234, Reset: 0x00000000, Name: ADCOFFSETHSTIA

Table 82. Bit Descriptions for ADCOFFSETHSTIA

Bits	Bit Name	Settings	Description	Reset	Access
[31:15]	Reserved		Reserved.	0x0	R
[14:0]	VALUE		High-Speed TIA Offset Calibration. ADC offset correction for high-speed TIA measurement mode, represented as a twos complement number. The calibration resolution is 0.25 LSBs of the ADCDAT LSB size.	0x0	R/W
		0x3FFF	4095.75 (maximum positive offset calibration value).		
		0x0001	0.25 (minimum positive offset calibration value).		
		0x0000	0 (no offset correction).		
		0x7FFF	−0.25 (minimum negative offset correction).		
		0x4000	−4096.0 (maximum negative offset correction).		

GAIN CALIBRATION FOR HIGH-SPEED TIA CHANNEL REGISTER

Address: 0x400C2284, Reset: 0x00004000, Name: ADCGNHSTIA

Table 83. Bit Descriptions for ADCGNHSTIA

Bits	Bit Name	Settings	Description	Reset	Access
[31:15]	Reserved		Reserved.	0x0	R
[14:0]	VALUE		Gain Error Calibration High-Speed TIA Channel.	0x4000	R/W
		0x7FFF	2 (maximum positive gain adjustment).		
		0x4001	1.000061 (minimum positive gain adjustment).		
		0x4000	1.0. ADC result multiplied by 1. No gain adjustment. Default value.		
		0x3FFF	0.999939 (minimum negative gain adjustment).		
		0x2000	0.5. ADC result multiplied by 0.5.		
		0x0001	0.000061 (maximum negative gain adjustment).		
		0x0000	0. Invalid value. Results in ADC result of 0.		

OFFSET CALIBRATION VOLTAGE CHANNEL (PGA GAIN = 1) REGISTER

Address: 0x400C2244, Reset: 0x00000000, Name: ADCOFFSETGN1

Table 84. Bit Descriptions for ADCOFFSETGN1

Bits	Bit Name	Settings	Description	Reset	Access
[31:15]	Reserved		Reserved.	0x0	R

REGISTER DETAILS: ADC CIRCUIT

Table 84. Bit Descriptions for ADCOFFSETGN1 (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
[14:0]	VALUE		Offset Calibration Gain 1. ADC offset correction for voltage channel with gain = 1, represented as a twos complement number. The calibration resolution is 0.25 LSBs of the ADCDAT LSB size.	0x0	R/W
		0x3FFF	4095.75 (maximum positive offset calibration value).		
		0x0001	0.25 (minimum positive offset calibration value).		
		0x0000	0 (no offset adjustment).		
		0x7FFF	−0.25 (minimum negative offset calibration value).		
		0x4000	−4096 (maximum negative offset calibration value).		

GAIN CALIBRATION VOLTAGE INPUT CHANNEL (PGA GAIN = 1) REGISTER

Address: 0x400C2240, Reset: 0x00004000, Name: ADCGAINGN1

Table 85. Bit Descriptions for ADCGAINGN1

Bits	Bit Name	Settings	Description	Reset	Access
[31:15]	Reserved		Reserved.	0x0	R
[14:0]	VALUE		Gain Calibration PGA Gain 1. ADC gain correction for voltage input channels. Stored as a signed number. Bit 14 is the sign bit and Bits[13:0] represent the fractional part.	0x4000	R/W
		0x7FFF	2 (maximum positive gain adjustment).		
		0x4001	1.000061 (minimum positive gain adjustment).		
		0x4000	1.0. ADC result multiplied by 1. No gain adjustment. Default value.		
		0x3FFF	0.999939 (minimum negative gain adjustment).		
		0x2000	0.5. ADC result multiplied by 0.5.		
		0x0001	0.000061 (maximum negative gain adjustment).		
		0x0000	0. Invalid value. Results in ADC result of 0.		

OFFSET CALIBRATION VOLTAGE CHANNEL (PGA GAIN = 1.5) REGISTER

Address: 0x400C22CC, Reset: 0x00000000, Name: ADCOFFSETGN1P5

Table 86. Bit Descriptions for ADCOFFSETGN1P5

Bits	Bit Name	Settings	Description	Reset	Access
[31:15]	Reserved		Reserved.	0x0	R
[14:0]	VALUE		Offset Calibration Gain 1.5. ADC offset correction with PGA gain = 1.5.	0x0	R/W
		0x3FFF	4095.75 (maximum positive offset calibration value).		
		0x0001	0.25 (minimum positive offset calibration value).		
		0x0000	0 (no offset adjustment).		
		0x7FFF	−0.25 (minimum negative offset calibration value).		
		0x4000	−4096 (maximum negative offset calibration value).		

GAIN CALIBRATION VOLTAGE INPUT CHANNEL (PGA GAIN = 1.5) REGISTER

Address: 0x400C2270, Reset: 0x00004000, Name: ADCGAINGN1P5

Table 87. Bit Descriptions for ADCGAINGN1P5

Bits	Bit Name	Settings	Description	Reset	Access
[31:15]	Reserved		Reserved.	0x0	R
[14:0]	VALUE		Gain Calibration PGA Gain 1.5. ADC gain correction for voltage input channels. Stored as a signed number. Bit 14 is the sign bit and Bits[13:0] represent the fractional part.	0x4000	R/W
		0x7FFF	2 (maximum positive gain adjustment).		
		0x4001	1.000061 (minimum positive gain adjustment).		
		0x4000	1.0. ADC result multiplied by 1. No gain adjustment. Default value.		
		0x3FFF	0.999939 (minimum negative gain adjustment).		
		0x2000	0.5. ADC result multiplied by 0.5.		

REGISTER DETAILS: ADC CIRCUIT

Table 87. Bit Descriptions for ADCGAINGN1P5 (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		0x0001	0.000061 (maximum negative gain adjustment).		
		0x0000	0. Invalid value. Results in ADC result of 0.		

OFFSET CALIBRATION VOLTAGE INPUT CHANNEL (PGA GAIN = 2) REGISTER

Address: 0x400C22C8, Reset: 0x00000000, Name: ADCOFFSETGN2

Table 88. Bit Descriptions for ADCOFFSETGN2

Bits	Bit Name	Settings	Description	Reset	Access
[31:15]	Reserved		Reserved.	0x0	R
[14:0]	VALUE		Offset Calibration Voltage Channel Gain 2. ADC offset correction for inputs using PGA gain = 2, represented as a twos complement number. The calibration resolution is 0.25 LSBs of the ADCDAT LSB size.	0x0	R/W
		0x3FFF	4095.75 (maximum positive offset calibration value).		
		0x0001	0.25 (minimum positive offset calibration value).		
		0x0000	0 (no offset adjustment).		
		0x7FFF	−0.25 (minimum negative offset calibration value).		
		0x4000	−4096 (maximum negative offset calibration value).		

GAIN CALIBRATION VOLTAGE INPUT CHANNEL (PGA GAIN = 2) REGISTER

Address: 0x400C2274, Reset: 0x00004000, Name: ADCGAINGN2

Table 89. Bit Descriptions for ADCGAINGN2

Bits	Bit Name	Settings	Description	Reset	Access
[31:15]	Reserved		Reserved.	0x0	R
[14:0]	VALUE		Gain Calibration PGA Gain 2. ADC gain correction for voltage input channels. Stored as a signed number. Bit 14 is the sign bit and Bits[13:0] represent the fractional part.	0x4000	R/W
		0x7FFF	2 (maximum positive gain adjustment).		
		0x4001	1.000061 (minimum positive gain adjustment).		
		0x4000	1.0. ADC result multiplied by 1. No gain adjustment. Default value.		
		0x3FFF	0.999939 (minimum negative gain adjustment).		
		0x2000	0.5. ADC result multiplied by 0.5.		
		0x0001	0.000061 (maximum negative gain adjustment).		
		0x0000	0. Invalid value. Results in ADC result of 0.		

OFFSET CALIBRATION VOLTAGE INPUT CHANNEL (PGA GAIN = 4) REGISTER

Address: 0x400C22D4, Reset: 0x00000000, Name: ADCOFFSETGN4

Table 90. Bit Descriptions for ADCOFFSETGN4

Bits	Bit Name	Settings	Description	Reset	Access
[31:15]	Reserved		Reserved.	0x0	R
[14:0]	VALUE		Offset Calibration Gain 4. ADC offset correction with PGA gain = 4.	0x0	R/W
		0x3FFF	4095.75 (maximum positive offset calibration value).		
		0x0001	0.25 (minimum positive offset calibration value).		
		0x0000	0 (no offset adjustment).		
		0x7FFF	−0.25 (minimum negative offset calibration value).		
		0x4000	−4096 (maximum negative offset calibration value).		

REGISTER DETAILS: ADC CIRCUIT

GAIN CALIBRATION VOLTAGE INPUT CHANNEL (PGA GAIN = 4) REGISTER

Address: 0x400C2278, Reset: 0x00004000, Name: ADCGAINGN4

Table 91. Bit Descriptions for ADCGAINGN4

Bits	Bit Name	Settings	Description	Reset	Access
[31:15]	Reserved		Reserved.	0x0	R
[14:0]	VALUE		Gain Calibration PGA Gain 4. ADC gain correction for voltage input channels. Stored as a signed number. Bit 14 is the sign bit and Bits[13:0] represent the fractional part.	0x4000	R/W
		0x7FFF	2 (maximum positive gain adjustment).		
		0x4001	1.000061 (minimum positive gain adjustment).		
		0x4000	1.0. ADC result multiplied by 1. No gain adjustment. Default value.		
		0x3FFF	0.999939 (minimum negative gain adjustment).		
		0x2000	0.5. ADC result multiplied by 0.5.		
		0x0001	0.000061 (maximum negative gain adjustment).		
		0x0000	0. Invalid value. Results in ADC result of 0.		

OFFSET CALIBRATION VOLTAGE INPUT CHANNEL (PGA GAIN = 9) REGISTER

Address: 0x400C22D0, Reset: 0x00000000, Name: ADCOFFSETGN9

Table 92. Bit Descriptions for ADCOFFSETGN9

Bits	Bit Name	Settings	Description	Reset	Access
[31:15]	Reserved		Reserved.	0x0	R
[14:0]	VALUE		Offset Calibration Gain 9. ADC offset correction with PGA gain = 9.	0x0	R/W
		0x3FFF	4095.75 (maximum positive offset calibration value).		
		0x0001	0.25 (minimum positive offset calibration value).		
		0x0000	0 (no offset adjustment).		
		0x7FFF	−0.25 (minimum negative offset calibration value).		
		0x4000	−4096 (maximum negative offset calibration value).		

GAIN CALIBRATION VOLTAGE INPUT CHANNEL (PGA GAIN = 9) REGISTER

Address: 0x400C2298, Reset: 0x00004000, Name: ADCGAINGN9

Table 93. Bit Descriptions for ADCGAINGN9

Bits	Bit Name	Settings	Description	Reset	Access
[31:15]	Reserved		Reserved.	0x0	R
[14:0]	VALUE		Gain Calibration PGA Gain 9. ADC gain correction for voltage input channels. Stored as a signed number. Bit 14 is the sign bit and Bits[13:0] represent the fractional part.	0x4000	R/W
		0x7FFF	2 (maximum positive gain adjustment).		
		0x4001	1.000061 (minimum positive gain adjustment).		
		0x4000	1.0. ADC result multiplied by 1. No gain adjustment. Default value.		
		0x3FFF	0.999939 (minimum negative gain adjustment).		
		0x2000	0.5. ADC result multiplied by 0.5.		
		0x0001	0.000061 (maximum negative gain adjustment).		
		0x0000	0. Invalid value. Results in ADC result of 0.		

OFFSET CALIBRATION TEMPERATURE SENSOR CHANNEL 0 REGISTER

Address: 0x400C223C, Reset: 0x00000000, Name: ADCOFFSETTEMPSENS0

Table 94. Bit Descriptions for ADCOFFSETTEMPSENS0

Bits	Bit Name	Settings	Description	Reset	Access
[31:15]	Reserved		Reserved.	0x0	R

REGISTER DETAILS: ADC CIRCUIT

Table 94. Bit Descriptions for ADCOFFSETTEMPSENS0 (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
[14:0]	VALUE		Offset Calibration Temperature Sensor. ADC offset correction for temperature sensor channel, represented as a twos complement number. The calibration resolution is 0.25 LSBs of the ADCDAT LSB size.	0x0	R/W
		0x3FFF	4095.75 (maximum positive offset calibration value).		
		0x0001	0.25 (minimum positive offset calibration value).		
		0x0000	0 (no offset adjustment).		
		0x7FFF	−0.25 (minimum negative offset calibration value).		
		0x4000	−4096 (maximum negative offset calibration value).		

GAIN CALIBRATION TEMPERATURE SENSOR CHANNEL 0 REGISTER

Address: 0x400C2238, Reset: 0x00004000, Name: ADCGAINTEMPSENS0

Table 95. Bit Descriptions for ADCGAINTEMPSENS0

Bits	Bit Name	Settings	Description	Reset	Access
[31:15]	Reserved		Reserved.	0x0	R
[14:0]	VALUE		Gain Calibration Temperature Sensor Channel. ADC gain correction for temperature sensor channel. Stored as a signed number. Bit 14 is the sign bit and Bits[13:0] represent the fractional part.	0x4000	R/W
		0x7FFF	2 (maximum positive gain adjustment).		
		0x4001	1.000061 (minimum positive gain adjustment).		
		0x4000	1.0. ADC result multiplied by 1. No gain adjustment. Default value.		
		0x3FFF	0.999939 (minimum negative gain adjustment).		
		0x2000	0.5. ADC result multiplied by 0.5.		
		0x0001	0.000061 (maximum negative gain adjustment).		
		0x0000	0. Invalid value. Results in ADC result of 0x8000.		

MINIMUM VALUE CHECK REGISTER

Address: 0x400C20A8, Reset: 0x00000000, Name: ADCMIN

Table 96. Bit Descriptions for ADCMIN

Bits	Bit Name	Settings	Description	Reset	Access
[31:16]	Reserved		Reserved.	0x0	R
[15:0]	MINVAL		ADC Minimum Value Threshold. This value is a low ADCDAT threshold value. If a value less than ADCMIN is measured by the ADC, the ADCINTSTA, Bit 4 is set to 1.	0x0	R/W

MINIMUM SLOW MOVING VALUE REGISTER

Address: 0x400C20AC, Reset: 0x00000000, Name: ADCMINSM

Table 97. Bit Descriptions for ADCMINSM

Bits	Bit Name	Settings	Description	Reset	Access
[31:16]	Reserved		Reserved.	0x0	R
[15:0]	MINCLRVAL		ADCMIN Hysteresis Value. If a value less than ADCMIN is measured by the ADC, ADCINTSTA, Bit 4 is set. ADCINTSTA, Bit 4 is set until ADCDAT > ADCMIN + ADCMINSM, Bits[15:0].	0x0	R/W

MAXIMUM VALUE CHECK REGISTER

Address: 0x400C20B0, Reset: 0x00000000, Name: ADCMAX

Table 98. Bit Descriptions for ADCMAX

Bits	Bit Name	Settings	Description	Reset	Access
[31:16]	Reserved		Reserved.	0x0	R

REGISTER DETAILS: ADC CIRCUIT

Table 98. Bit Descriptions for ADCMAX (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	MAXVAL		ADC Maximum Threshold. Optional maximum ADCDAT threshold. If a value greater than ADCMAX is measured by the ADC, ADCINTSTA, Bit 5 is set to 1.	0x0	R/W

MAXIMUM SLOW MOVING REGISTER

Address: 0x400C20B4, Reset: 0x00000000, Name: ADCMAXSMEN

Table 99. Bit Descriptions for ADCMAXSMEN

Bits	Bit Name	Settings	Description	Reset	Access
[31:16]	Reserved		Reserved.	0x0	R
[15:0]	MAXSWEN		ADC Maximum Hysteresis Value. If a value >ADCMAX is measured by the ADC, ADCINTSTA, Bit 5 is set. ADCINTSTA, Bit 5 is set until ADCDAT < ADCMAX − ADCMAXSMEN, Bits[15:0].	0x0	R/W

DELTA CHECK REGISTER

Address: 0x400C20B8, Reset: 0x00000000, Name: ADCDELTA

Table 100. Bit Descriptions for ADCDELTA

Bits	Bit Name	Settings	Description	Reset	Access
[31:16]	Reserved		Reserved.	0x0	R
[15:0]	DELTAVAL		ADCDAT Code Differences Limit Option. If two consecutive ADCDAT results have a difference greater than ADCDELTA, Bits[15:0], an error flag is set via ADCINTSTA, Bit 6.	0x0	R/W

STATISTICS MODULE CONFIGURATION REGISTER

Address: 0x400C21C4, Reset: 0x00000000, Name: STATSCON

Includes mean and outlier detection blocks.

Table 101. Bit Descriptions for STATSCON

Bits	Bit Name	Settings	Description	Reset	Access
[31:7]	Reserved		Reserved.	0x0	R
[6:4]	SAMPLENUM		Sample Size. Sets the number of ADC samples used for each statistic calculation.	0x0	R/W
		0	128 samples.		
		1	64 samples.		
		10	32 samples.		
		11	16 samples.		
		100	8 samples.		
[3:1]	Reserved		Reserved.	0x0	R/W
0	STATSEN		Statistics Enable.	0x0	R/W
		0	Disable statistics.		
		1	Enable statistics.		

MEAN OUTPUT REGISTER

Address: 0x400C21C8, Reset: 0x00000000, Name: STATSMEAN

Table 102. Bit Descriptions for STATSMEAN

Bits	Bit Name	Settings	Description	Reset	Access
[31:16]	Reserved		Reserved.	0x0	R
[15:0]	MEAN		Mean Output. Mean value calculated for the number of ADC samples set by STATSCON, Bits[6:4].	0x0	R

REGISTER DETAILS: ADC CIRCUIT

KEY ACCESS FOR DSPUPDATEEN REGISTER

Address: 0x400C0434, Reset: 0x00000000, Name: MKEY

Table 103. Bit Descriptions for MKEY

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	KEY		Key Access for DSPUPDATEEN Register. To access the DSPUPDATEEN register, write 0xA51F to this register first. After writing to DSPUPDATEEN, write 0x0000 to this register to lock the key again.	0x0	W

DIGITAL LOGIC TEST ENABLE REGISTER

Address: 0x400C0438, Reset: 0x00000000, Name: DSPUPDATEEN

Table 104. Bit Descriptions for DSPUPDATEEN

Bits	Bit Name	Settings	Description	Reset	Access
[31:1]	Reserved		Reserved.	0x0	R/W
0	DSPLOOP		ADC Digital Logic Test Enable. Allows high-speed DAC waveform generator to create digital values that connect to the output digital logic of the ADC.	0x0	R/W
		0	Disables digital logic test function.		
		1	Enable digital logic test feature.		

TEMPERATURE SENSOR 1 CONTROL REGISTER

Address: 0x400C2374, Reset: 0x00020000, Name: TEMPCON1

Table 105. Bit Descriptions for TEMPCON1

Bits	Bit Name	Settings	Description	Reset	Access
[31:18]	Reserved		Reserved.	0x0	R
17	PWD		Power-Down Control. Power down Temperature Sensor 1 channel.	0x1	R/W
		0	Enable Temperature Sensor 1 in active mode.		
		1	Power down Temperature Sensor 1 channel.		
16	EN		Test Signal Enable. Enables Temperature Sensor 1.	0x0	R/W
		0	Turn on. Enable conversions.		
		1	Turn off. Disable conversions.		
[15:0]	ISWCON		Bias Current Selection. Switch control register. Each bit controls the switches connecting to the VBE transistor.	0x0	R/W

LOW-POWER POTENTIOSTAT AMPLIFIERS AND LOW-POWER TIAS

The ADuCM355 features two low-power TIAs and two low-power potentiostat amplifiers. This section details the operation of these components.

LOW-POWER POTENTIOSTAT AMPLIFIERS

The ADuCM355 has two low-power potentiostat amplifiers designed to set the bias voltage of an external electrochemical sensor. The bias voltage is the voltage between the sense electrode and reference electrode. Depending on the electrochemical sensor used, a specific bias voltage is required. The bias voltage is set by the low-powers DACs. See the Low-Power DACs section for details.

Figure 15 shows the potentiostat amplifier connected to a 3-lead electrochemical sensor. The potentiostat amplifier (labeled PA in Figure 15) has the V_{BIAS} output of the dual DAC as its noninverting input. The amplifier output is connected to the counter electrode. The reference electrode is connected to the inverting input of the potentiostat. As such, the voltage on the reference electrode is determined by the V_{BIAS} DAC output voltage via the potentiostat amplifier.

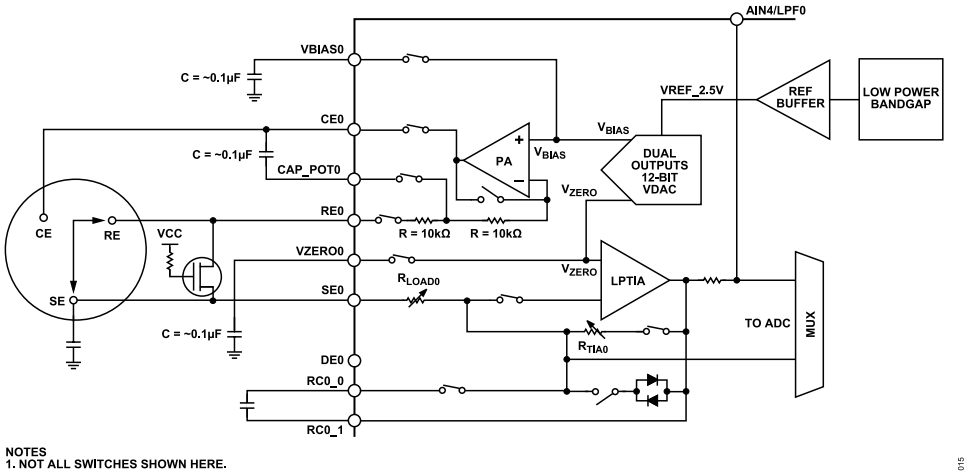

Figure 15. Low-Power Potentiostat and Low-Power TIA and DAC Connected to One Electrochemical Sensor

LOW-POWER TIAS

Two low-power TIA channels are available on the ADuCM355. The load resistor and gain resistor values are specified in the Lx registers. Select the TIA gain resistor that maximizes the ADC input voltage range for the selected PGA gain setting. For example, if the PGA gain setting is 1, select a TIA gain resistor to maximize the ±900 mV range. To calculate the required gain resistor, use the following equation:

$$I_{MAX} = 0.9/R_{TIA} \quad (6)$$

where:
I_{MAX} is the expected full-scale input current.
R_{TIA} is the TIA gain resistor selected by LPTIACONx, Bits[9:5].

A number of operation modes are selectable by user code. The different modes are selected by configuring a series of switches. Figure 16 shows the various switches for Channel 0. These switches are controlled within the LPTIASW0 register. The switches in Figure 17 are controlled by the LPTIASW1 register for Channel 1. Switch 0 (SW0) to Switch 13 (SW13) are the same for both channels. Channel 1 does not have SW15 or SW14. The LPTIASW0 register, Bit 0, controls SW0, and the LPTIASW0 register, Bit 1, controls SW1.

Low-Power TIA Protection Diodes

Figure 16 shows back to back protection diodes connected parallel to the R_{TIA0} gain resistor. These diodes can be connected or disconnected by controlling SW0, which in turn is controlled by LPTIASWx, Bit 0. These diodes are intended for use when switching R_{TIA} gain settings to amplify small currents to prevent saturation of the TIA. These diodes have a leakage current specification that is dependent on the voltage across them. The leakage current is large if the differential voltage across the diode is >200 mV. The leakage current can be >1 nA and several microamperes if >500 mV.

LOW-POWER POTENTIOSTAT AMPLIFIERS AND LOW-POWER TIAS

Close SW0 when changing the R_{TIA} value, and open SW0 again when the change is complete. If high currents are detected on the low-power TIA input path when using an oxygen electrochemical sensor, close the shorting switch, SW1, to protect the low-power TIA input circuitry. SW1 is controlled by LPTIASWx, Bit 1. For full details of the high-speed TIA in Figure 16 and Figure 17, refer to the High-Speed TIA Circuits section.

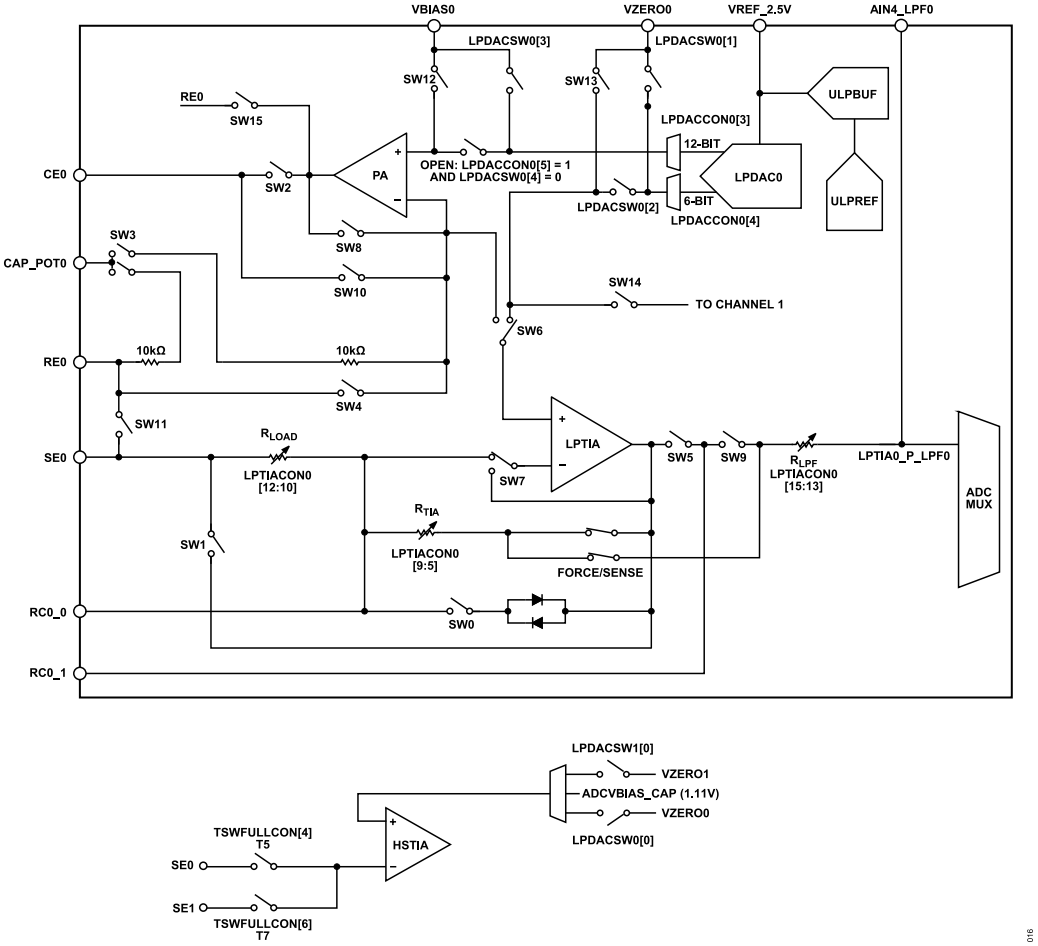

Figure 16. Low-Power TIA, Low-Power Potentiostat, and Low-Power DAC Switches for Channel 0

LOW-POWER POTENTIOSTAT AMPLIFIERS AND LOW-POWER TIAS

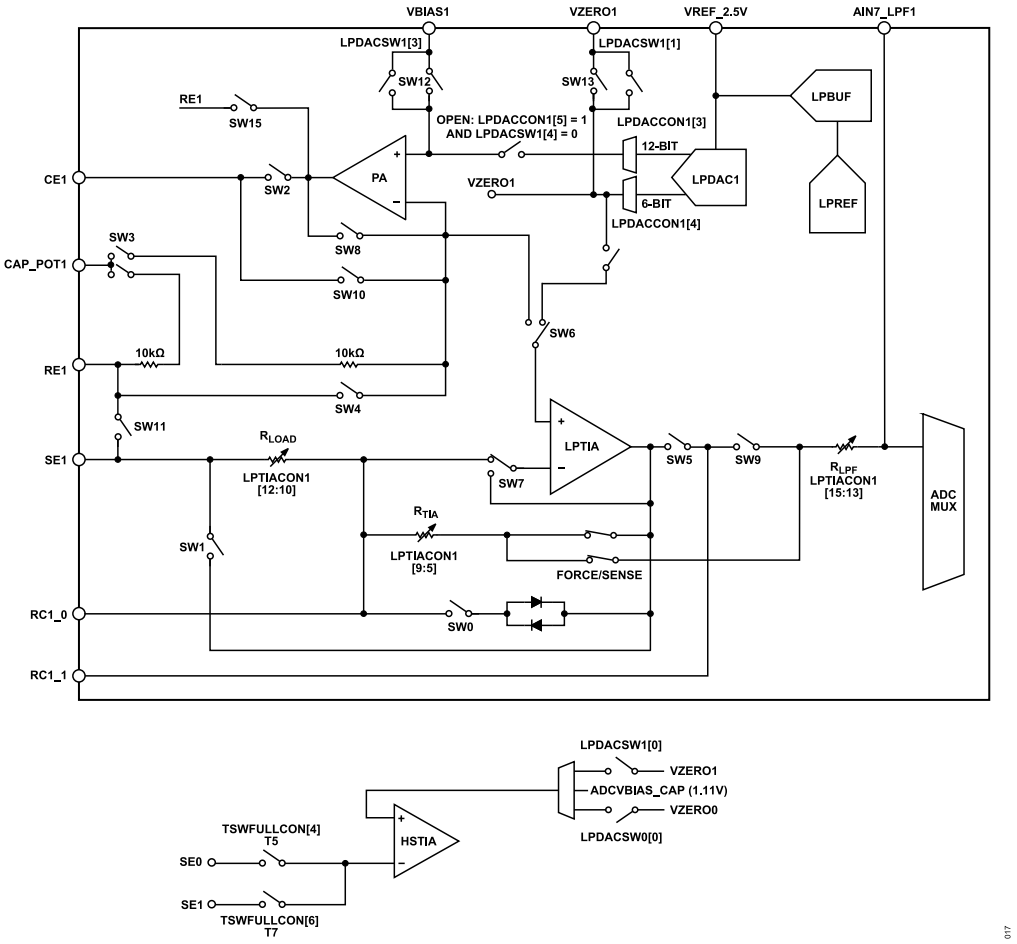

Figure 17. Low-Power TIA, Low-Power Potentiostat, and Low-Power DAC Switches for Channel 1

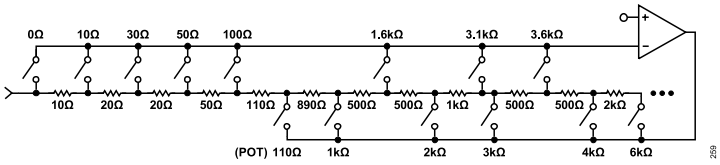

Figure 18. Low-Power TIA R_{LOAD} and R_{GAIN} Configuration

Figure 18 shows the relationship between the R_{LOAD} and R_{GAIN} settings for the low-power TIA. R_{LOAD} is configured by setting LPTIACON0, Bits[12:10]. R_{GAIN} is configured by LPTIACON0, Bits[9:5]. When R_{LOAD} is large, it uses resistors from the R_{GAIN} bank, which reduces the size of R_{GAIN}. See the descriptions in the LPTIACON0 bit fields for details.

Low-Power TIA and Potentiostat Amplifiers Current Limit Feature

In addition to the protection diode, the low-power TIAs also have a built in current limiting feature. If the current sourced or sunk from the low-power TIAs is greater than the overcurrent limit protection specified in the ADuCM355 data sheet, the amplifiers clamp the current to this limit. If a sensor on startup attempts to source or sink more than the overcurrent limit, the amplifier clamps the output current. Do not use this feature more frequently or for longer than specified in the data sheet. See the ADuCM355 data sheet for full specifications.

Low-Power TIA Force and Sense Feature

LPTIACONx, Bits[9:5] select different gain resistor values for the low-power TIA, labeled as R_{TIA} in Figure 16 and Figure 17. The force and sense connections shown on the feedback path of the low-power TIA are used to avoid voltage (I × R) drops on the switches used to select different R_{TIA} settings for the internal R_{TIA}.

LOW-POWER POTENTIOSTAT AMPLIFIERS AND LOW-POWER TIAS

External R_{TIA} Gain Resistor with Low-Power TIA Amplifiers

To use an external R_{TIA} gain resistor, follow this process:

1. Connect the external R_{TIA} across the RC0_0 and RC0_1 pins for Channel 0. For Channel 1, connect the external R_{TIA} across the RC1_0 and RC1_1 pins.
2. Clear LPTIACONx, Bits[9:5] = 0b00000 to disconnect the internal R_{TIA} from the TIA output terminal.
3. Close the SW9 switch by setting LPTIASWx, Bit 9 = 1. When using the internal R_{TIA}, open the SW9 switch.

LOW-POWER DACS

The low-power DACs are designed to set the sensor bias voltage. In Figure 15, the sensor bias voltage is the voltage difference between the reference electrode and sense electrode.

Each low-power DAC has two outputs, an output with 12-bit resolution (VBIAS0 pin) and an output with 6-bit resolution (VZERO0 pin). The low-power DACs are made up of two 6-bit string DACs. The main 6-bit string DAC provides the VZERO0 DAC output, and is made up of 63 resistors. Each resistor is the same value.

The main 6-bit string with the 6-bit subDAC provides the VBIAS0 DAC output. In 12-bit mode, the MSBs select a resistor from the main string DAC. The top end of this resistor is selected as the top of the 6-bit subDAC, and the bottom end of the selected resistor is connected to the bottom of the 6-bit subDAC string.

The resistor matching between the 12-bit and 6-bit subDAC means 64 LSB 12-bit (VBIAS0 pin) is equal to one LSB 6-bit (VZERO0 pin).

Figure 19 shows the low-power DAC structure. The main DAC string shows the 64 resistors that make up the six bits of the VZERO0 output and the six MSBs of the VBIAS output. The main DAC makes up the 6-bit DAC output (VZERO0 output), and is controlled by LPDACDATx, Bits[17:12]. The output range is 0.2 V to 2.36615 V. The right side of the main DAC block shows the six MSBs of the 12-bit DAC output (VBIAS0 output), which are controlled by LPDACDATx, Bits[11:0]. The voltage steps of these bits are equal to the 6-bit VZERO0 output steps. The 12-bit output is capable of reaching 2.4 V.

The subDAC block shows the 64-resistor string used to generate the six LSBs of the 12-bit output. The low-power DACs are controlled by LPDACCONx, Bits[5:0].

LOW-POWER POTENTIOSTAT AMPLIFIERS AND LOW-POWER TIAS

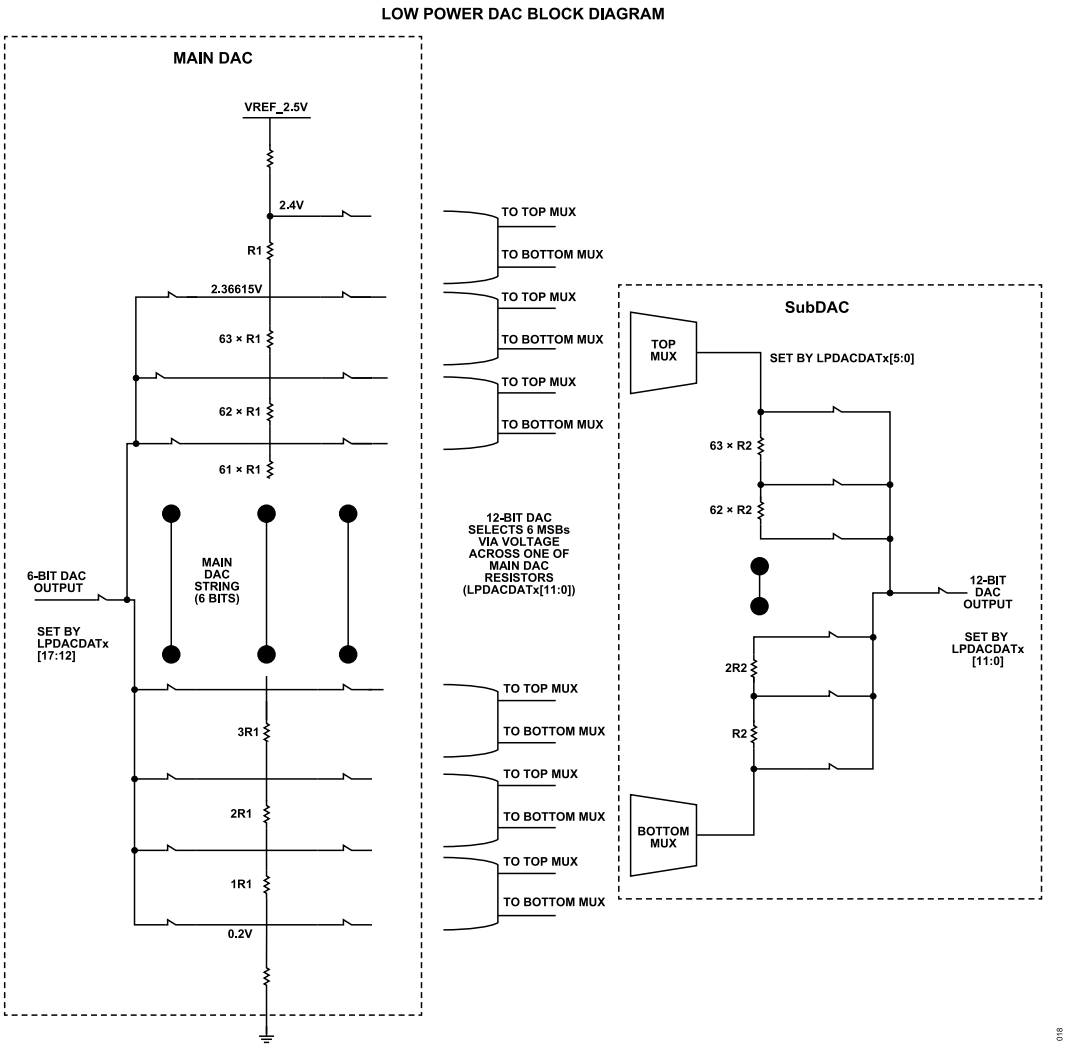

Figure 19. Block Diagram of Low-Power DACs

Low-Power DAC Switch Options

There are a number of switch options available that allow the user to configure the low-power DACs for various modes of operation. These switches facilitate a number of different use cases, such as electrochemical impedance spectroscopy. Figure 16 shows the location of the switches controlled by LPDACSWx, Bits[4:0]. These switches are controlled either automatically via LPDACCONx, Bit 5 or individually via the LPDACSWx registers.

When LPDACCONx, Bit 5 is cleared, the switches are configured for normal mode. SW2 and SW3 are closed and SW0, SW1, and SW4 are open. When LPDACCONx, Bit 5 is set, the switches are configured for diagnostic mode. SW0 and SW4 are closed and SW1, SW2, and SW3 are open. This feature is designed for electrochemical use cases in normal mode, where the low-power TIAs are used to measure the sense electrode. In diagnostic mode, the high-speed TIA is used to measure the sense electrode. By switching the VZEROx output from the low-power TIA to the high-speed TIA, the effective bias on the sensor (V_{BIAS} to V_{ZERO}) is unaffected. Using the high-speed TIA facilitates high bandwidth measurements such as impedance, pulse, and cyclic voltammetry.

To control the switches individually, use the LPDACSWx registers. LPDACSWx, Bit 5 must be set to 1 so that each switch can be individually controlled via LPDACSWx, Bits[4:0].

Relationship Between 12-Bit and 6-Bit Outputs, Hardware Compensation Enabled

The 12-bit and 6-bit outputs are mostly independent. However, the selected 12-bit value does have a loading effect on the 6-bit output that must be compensated for in user code, particularly when the 12-bit output level is greater than the 6-bit output.

LOW-POWER POTENTIOSTAT AMPLIFIERS AND LOW-POWER TIAS

When the 12-bit output < 6-bit output,

12-Bit DAC Output Voltage = 0.2 V + (*LPDACDATx, Bits[11:0]* × 0.54 mV) (7)

6-Bit DAC Output Voltage = 0.2 V + (*LPDACDATx, Bits[17:12]* × 34.38 mV) (8)

When the 12-bit output is greater than or equal to 6-bit output,

12-Bit DAC Output Voltage = 0.2 V + (*LPDACDATx, Bits[11:0]* × 0.54 mV) + 0.54 mV (9)

6-Bit DAC Output Voltage = 0.2 V + (*LPDACDATx, Bits[17:12]* × 34.38 mV) (10)

where 0.54 mV is approximately 1 LSB of the 12-bit DAC and 34.38 mV is approximately 1 LSB of the 6-bit DAC.

In user code, it is recommended to add the following:

```
12BITCODE = LPDACDATx[11:0];
6BITCODE = LPDACDATx[17:12];
if (12BITCODE > (6BITCODE *64))
    LPDACDATx[11:0] = (12BITCODE - 1);
```

If LPDACDATx, Bits[11:0] = 4095, the minimum voltage on the 12-bit output is 2.39946 V, because LPDACDATx, Bits[11:0] = 4095 has the same effect as LPDACDATx, Bits[11:0] = 4094.

Low-Power DAC Use Cases

Electrochemical Amperometric Measurement

In an electrochemical measurement, the 12-bit output sets the voltage on the reference electrode pin via the potentiostat circuit shown in Figure 20. The voltage on the CE0 and RE0 pins is V_{BIAS}. The 6-bit output sets the bias voltage on the low-power TIA positive pin, which in turn sets the voltage on SE0. This voltage is V_{ZERO}. The bias voltage on the sensor effectively becomes the difference between the 12-bit output and the 6-bit output.

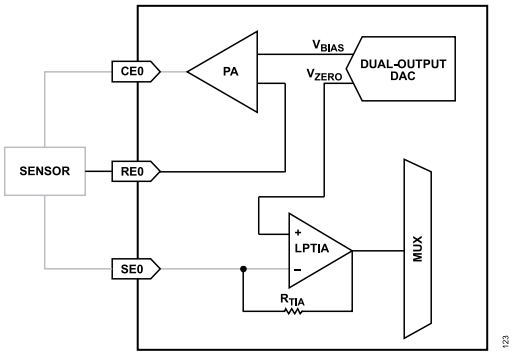

Figure 20. Electrochemical Standard Configuration

Recommended Switch Settings for Various Operating Modes

Table 106 details the recommended switch settings in the low-power potentiostat loop for various measurement types. For all measurement types, setting to 1 closes the switch and setting to 0 opens the switch. LPTIASWx, Bits[13:0] control SW13 to SW0 in Figure 16 and Figure 17.

Table 106. Recommended Switch Settings in Low-Power Potentiostat Loop

Measurement Name	LPDACCONx Bit 5 Setting	LPDACSWx Bits[5:0] Setting	LPTIASWx Bits[13:0] Setting	Description
Amperometric Mode	0	0xXX (don't care)	0x302C or 0b11 0000 0010 1100	Normal DC current measurement. External capacitors to VBIASx and VZEROx DACs are connected.

LOW-POWER POTENTIOSTAT AMPLIFIERS AND LOW-POWER TIAS

Table 106. Recommended Switch Settings in Low-Power Potentiostat Loop (Continued)

Measurement Name	LPDACCONx Bit 5 Setting	LPDACSWx Bits[5:0] Setting	LPTIASWx Bits[13:0] Setting	Description
Amperometric Mode with Diode Protection	0	0xXX (don't care)	0x302D or 0b11 0000 0010 1101	Normal DC current measurement with low-power TIA back to back diode protection enabled. External capacitors to VBIASx and VZEROx DACs connected.
Amperometric Mode with Short Switch Enabled	0	0xXX (don't care)	0x302E or 0b11 0000 0010 1110	Normal DC current measurement with short switch protection enabled. SW1 is closed to connect the sense electrode input to the output of the low-power TIA. External capacitors to VBIASx and VZEROx DACs connected. Useful if external sensor must be charged after a power-up and many currents flow in or out of the SEx pin.
Amperometric Mode for Zero-Biased Sensor	0	0xXX (don't care)	0x306C or 0b11 0000 0110 1100	Amperometric mode with SW6 configured to set sensors reference electrode and sense electrode to VBIASx level. Potentiostat amplifier inverting and low-power TIA noninverting inputs shorted. Gives optimal noise performance for zero bias voltage sensors.
Amperometric Mode for 2-Lead Sensor	0	0xXX (don't care)	0x342C or 0b11 0100 0010 1100	Amperometric mode with SW10 closed to short counter electrode to reference electrode internally.
Chronoamperometry (Low-Power Pulse Test) Using Low-Power TIA	1	0x32	0x0014 or 0b00 0000 0001 0100	VBIASx output generates pulse to counter electrode. Capacitors on low-power DACs are disconnected. Low-power TIA measures sense electrode current response.
Chronoamperometry (Full Power Pulse Test) Using High-Speed TIA on Sense Electrode	1	0x31	0x0094 or 0b00 0000 1001 0100	VBIASx output generates pulse to counter electrode. Capacitors on low-power DACs are disconnected. High-speed TIA measures sense electrode current response.
Voltammetry (Full Power Pulse Test) Using High-Speed TIA	1	0x31	0x0094 or 0b00 0000 1001 0100	VBIASx output generates pulse to counter electrode. Capacitors on low-power DACs are disconnected. High-speed TIA measures sense electrode or the DEx pin current response. High-speed TIA resistors and switches are configured separately.
Potentiostat Amplifier and Low-Power TIA in Unity-Gain Mode (Test Mode)	0	0xXX (don't care)	0x04A4 or 0b00 0100 1010 0100	Potentiostat amplifier in unity gain mode, output to CEx pin. Low-power TIA in unity gain mode, output to RCx_1 pin. Useful for checking VBIASx or VZEROx DAC outputs.

Electrochemical Impedance Spectroscopy

In many electrochemical applications, there is significant value to carrying out a diagnostic measurement. A typical diagnostic technique is carrying out an impedance measurement on the sensor. For some sensor types, the DC bias on the sensor must be maintained when carrying out the impedance measurement. The ADuCM355 facilitates this measurement. To maintain the DC bias on the sensor, set LPDACCONx, Bit 5 = 1. VZEROx is set to the input of the high-speed TIA, and the high-speed DAC is used to generate an AC signal. The level of the AC signal is set via the VBIASx output of the low-power DAC. The voltage on the SE0 and SE1 pins is maintained by VZEROx. The high-speed DAC DC buffers must also be enabled by setting AFECON, Bit 21.

REGISTER SUMMARY: LOW-POWER TIA/POTENTIOSTAT AND DAC CIRCUITS

Table 107. Low-Power Potentiostat and TIA Control Register Summary

Address	Name	Description	Reset	Access
0x400C20EC	LPTIACON0	Low-power TIA control bits Channel 0	0x00000003	R/W
0x400C20E4	LPTIASW0	Low-power TIA switch configuration for Channel 0	0x00000000	R/W
0x400C20E8	LPTIACON1	Low-power TIA control bits Channel 1	0x00000003	R/W
0x400C20E0	LPTIASW1	Low-power TIA switch configuration for Channel 1	0x00000000	R/W

Table 108. Low-Power DAC Control Register Summary

Address	Name	Description	Reset	Access
0x400C2120	LPDACDAT0	Low-power DAC0 data out	0x00000000	R/W
0x400C2124	LPDACSW0	Low-power DAC0 switch control	0x00000000	R/W
0x400C2128	LPDACCON0	Low-power DAC0 control	0x00000002	R/W
0x400C212C	LPDACDAT1	Low-power DAC1 data out	0x00000000	R/W
0x400C2130	LPDACSW1	Low-power DAC1 switch control	0x00000000	R/W
0x400C2134	LPDACCON1	Low-power DAC1 control	0x00000002	R/W
0x400C2050	LPREFBUFCON	Low-power reference control	0x00000000	R/W

REGISTER DETAILS: LOW POWER TIA/POTENTIOSTAT AND DAC CIRCUITS

LOW-POWER TIA CONTROL BITS CHANNEL 0 REGISTER

Address: 0x400C20EC, Reset: 0x00000003, Name: LPTIACON0

Table 109. Bit Descriptions for LPTIACON0

Bits	Bit Name	Settings	Description	Reset	Access
[31:16]	Reserved		Reserved.	0x0	R
[15:13]	TIARFILT		Set Low-Pass Filter Resistor. Configures TIA output low-pass filter cutoff frequency.	0x0	R/W
		0	Disconnect TIA output from the AIN4_LPF0 pin. Useful for diagnostics where fast response is required from the ADC. This bit setting disconnects the low-power TIA output from the low-pass filter capacitor.		
		1	Bypass resistor. 0 Ω option.		
		10	20 kΩ.		
		11	100 kΩ.		
		100	200 kΩ.		
		101	400 kΩ.		
		110	600 kΩ.		
		111	1 MΩ. Recommended value for optimal DC current measurement performance. Lowest cutoff frequency setting for low-pass filter.		
[12:10]	TIARL		Set R_{LOAD}.	0x0	R/W
		0	0 Ω.		
		1	10 Ω.		
		10	30 Ω.		
		11	50 Ω.		
		100	100 Ω.		
		101	1.6 kΩ. R_{TIA} gain resistor must be ≥2 kΩ.		
		110	3.1 kΩ. R_{TIA} gain resistor must be ≥4 kΩ.		
		111	3.6 kΩ. R_{TIA} gain resistor must be ≥4 kΩ.		
[9:5]	TIAGAIN		Set R_{TIA}.	0x0	R/W
		0	Disconnect R_{TIA}.		
		1	200 Ω. Intended for oxygen sensor. R_{TIA} is combination of R_{LOAD} and a fixed series 110 Ω. Assume R_{LOAD} = 10 Ω. Set by the TIARL bit. R_{TIA} gain setting = 100 Ω − R_{LOAD} + 110 Ω fixed. Overall TIA gain is 200.		
		10	1 kΩ. If R_{LOAD} ≤ 100 Ω, R_{TIA} gain = (100 Ω − R_{LOAD}) + 1 kΩ. If R_{LOAD} > 100 Ω, R_{TIA} gain is not supported with a 1 kΩ R_{TIA}.		
		11	2 kΩ. R_{TIA} gain = 2 kΩ − (R_{LOAD} − 100 Ω).		
		100	3 kΩ. R_{TIA} gain = 3 kΩ − (R_{LOAD} − 100 Ω).		
		101	4 kΩ. R_{TIA} gain = 4 kΩ − (R_{LOAD} − 100 Ω).		
		110	6 kΩ. R_{TIA} gain = 6 kΩ − (R_{LOAD} − 100 Ω).		
		111	8 kΩ. R_{TIA} gain = 8 kΩ − (R_{LOAD} − 100 Ω).		
		1000	10 kΩ. R_{TIA} gain = 10 kΩ − (R_{LOAD} − 100 Ω).		
		1001	12 kΩ. R_{TIA} gain = 12 kΩ − (R_{LOAD} − 100 Ω).		
		1010	16 kΩ. R_{TIA} gain = 16 kΩ − (R_{LOAD} − 100 Ω).		
		1011	20 kΩ. R_{TIA} gain = 20 kΩ − (R_{LOAD} − 100 Ω).		
		1100	24 kΩ. R_{TIA} gain = 24 kΩ − (R_{LOAD} − 100 Ω).		
		1101	30 kΩ. If R_{LOAD} ≤ 100 Ω, R_{TIA} gain = (100 Ω − R_{LOAD}) + 30 Ω. If R_{LOAD} > 100 Ω, R_{TIA} gain = 30 kΩ − (R_{LOAD} − 100 Ω).		
		1110	32 kΩ. R_{TIA} gain = 32 kΩ − (R_{LOAD} − 100 Ω).		
		1111	40 kΩ. R_{TIA} gain = 40 kΩ − (R_{LOAD} − 100 Ω).		
		10000	48 kΩ. R_{TIA} gain = 48 kΩ − (R_{LOAD} − 100 Ω).		
		10001	64 kΩ. R_{TIA} gain = 64 kΩ − (R_{LOAD} − 100 Ω).		
		10010	85 kΩ. R_{TIA} gain = 85 kΩ − (R_{LOAD} − 100 Ω).		
		10011	96 kΩ. R_{TIA} gain = 96 kΩ − (R_{LOAD} − 100 Ω).		

REGISTER DETAILS: LOW POWER TIA/POTENTIOSTAT AND DAC CIRCUITS

Table 109. Bit Descriptions for LPTIACON0 (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		10100	100 kΩ. R_{TIA} gain = 100 kΩ − (R_{LOAD} − 100 Ω).		
		10101	120 kΩ. R_{TIA} gain = 120 kΩ − (R_{LOAD} − 100 Ω).		
		10110	128 kΩ. R_{TIA} gain = 128 kΩ − (R_{LOAD} − 100 Ω).		
		10111	160 kΩ. R_{TIA} gain = 160 kΩ − (R_{LOAD} − 100 Ω).		
		11000	196 kΩ. R_{TIA} gain = 196 kΩ − (R_{LOAD} − 100 Ω).		
		11001	256 kΩ. R_{TIA} gain = 256 kΩ − (R_{LOAD} − 100 Ω).		
		11010	512 kΩ. R_{TIA} gain = 512 kΩ − (R_{LOAD} − 100 Ω).		
[4:3]	IBOOST		Current Boost Control.	0x0	R/W
		00	Normal mode.		
		01	Increase amplifier output stage current to quickly charge external capacitor load. Intended for use with high current sensors, such as oxygen electrochemical sensor.		
		10	Double TIA and potentiostat amplifiers overall quiescent current. Increases amplifier bandwidth. Useful for diagnostic tests.		
		11	Double TIA and potentiostat amplifiers overall quiescent current and increase output stage current. Increases amplifier bandwidth and output current capability. Useful for diagnostic tests with high current sensors like the oxygen electrochemical sensor.		
2	HALFPWR		Half Power Mode Select. Control bit to reduce active power consumption of TIA and potentiostat amplifiers for Sensor Channel 0.	0x0	R/W
		0	Normal mode. Default value.		
		1	Reduce potentiostat amplifier and TIA current by half. Degrades performance.		
1	PAPDEN		Potentiostat Amplifier Power Down. Low-power Potentiostat Amplifier 0 power-down control bit.	0x1	R/W
		0	Power up.		
		1	Power down.		
0	TIAPDEN		TIA Power Down. Low-power TIA0 power-down control bit.	0x1	R/W
		0	Power up.		
		1	Power down.		

LOW-POWER TIA SWITCH CONFIGURATION FOR CHANNEL 0 REGISTER

Address: 0x400C20E4, Reset: 0x00000000, Name: LPTIASW0

See Figure 16 for details on the switches mentioned in this register.

Table 110. Bit Descriptions for LPTIASW0

Bits	Bit Name	Settings	Description	Reset	Access
[31:16]	Reserved		Reserved.	0x0	R
15	RECAL		TIA SW15 Control. Active high.	0x0	R/W
		0	Disconnect potentiostat amplifier for Channel 0 output from the RE0 pin.		
		1	Connect Potentiostat Amplifier 0 output to the RE0 pin.		
14	VZEROSHARE		TIA SW14 Control. Active high.	0x0	R/W
		0	Turn off switch. Default.		
		1	Short TIA0 input to TIA1 input and share the VZEROx pin.		
13	TIABIASSEL		TIA SW13 Control. Active high.	0x0	R/W
		0	Disconnect TIA bias voltage from the VZERO0 pin.		
		1	Connect TIA bias voltage to the VZERO0 pin.		
12	PABIASSEL		TIA SW12 Control. Active high.	0x0	R/W
		0	Disconnect potentiostat amplifier bias voltage from the VBIAS0 pin.		
		1	Connect potentiostat amplifier bias voltage to the VBIAS0 pin.		
11	SW11		SW11 Switch Control Active High.	0x0	R/W
		0	Open switch.		
		1	Close switch.		

REGISTER DETAILS: LOW POWER TIA/POTENTIOSTAT AND DAC CIRCUITS

Table 110. Bit Descriptions for LPTIASW0 (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
10	SW10		SW10 Switch Control Active High.	0x0	R/W
		0	Open switch.		
		1	Close switch.		
9	SW9		SW9 Switch Control Active High.	0x0	R/W
		0	Open switch.		
		1	Close switch.		
8	SW8		SW8 Switch Control Active High.	0x0	R/W
		0	Open switch.		
		1	Close switch.		
7	SW7		SW7 Switch Control Active High.	0x0	R/W
		0	Open switch.		
		1	Close switch.		
6	SW6		SW6 Switch Control Active High.	0x0	R/W
		0	Open switch.		
		1	Close switch.		
5	SW5		SW5 Switch Control Active High. Close to connect external capacitor or R_{TIA} between the RC0_0 and RC0_1 pins.	0x0	R/W
		0	Open switch.		
		1	Close switch.		
4	SW4		SW4 Switch Control Active High.	0x0	R/W
		0	Open switch.		
		1	Close switch.		
3	SW3		SW3 Switch Control Active High.	0x0	R/W
		0	Open switch.		
		1	Close switch.		
2	SW2		SW2 Switch Control Active High.	0x0	R/W
		0	Open switch.		
		1	Close switch.		
1	SW1		SW1 Switch Control Active High.	0x0	R/W
		0	Open switch.		
		1	Close switch.		
0	SW0		SW0 Switch Control Active High.	0x0	R/W
		0	Open switch.		
		1	Close switch.		

LOW-POWER TIA CONTROL BITS CHANNEL 1 REGISTER

Address: 0x400C20E8, Reset: 0x00000003, Name: LPTIACON1

Table 111. Bit Descriptions for LPTIACON1

Bits	Bit Name	Settings	Description	Reset	Access
[31:16]	Reserved		Reserved.	0x0	R
[15:13]	TIARFILT		Set Low-Pass Filter Resistor.	0x0	R/W
		0	Disconnect TIA output from AIN7_LPF1 pin. Useful for diagnostics where fast response is required from ADC. Disconnects the low-power TIA output from the low-pass filter capacitor.		
		1	Bypass resistor.		
		10	20 kΩ.		
		11	100 kΩ.		
		100	200 kΩ.		
		101	400 kΩ.		

REGISTER DETAILS: LOW POWER TIA/POTENTIOSTAT AND DAC CIRCUITS

Table 111. Bit Descriptions for LPTIACON1 (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		110	600 kΩ.		
		111	1 MΩ. Recommended value for best DC current measurement performance. Lowest cutoff frequency setting for low-pass filter.		
[12:10]	TIARL		Set R_{LOAD}.	0x0	R/W
		0	0 Ω.		
		1	10 Ω.		
		10	30 Ω.		
		11	50 Ω.		
		100	100 Ω.		
		101	1.6 kΩ. R_{TIA} must be ≥2 kΩ.		
		110	3.1 kΩ. R_{TIA} must be ≥4 kΩ.		
		111	3.6 kΩ. R_{TIA} must be ≥4 kΩ.		
[9:5]	TIAGAIN		Set R_{TIA}.	0x0	R/W
		0	Disconnect R_{TIA}.		
		1	200 Ω. Intended for oxygen sensor. TIA gain resistor is combination of R_{LOAD} and fixed series 110 Ω. Assume R_{LOAD} = 10 Ω. Set by the TIARL bit. R_{TIA} gain setting = 100 Ω − R_{LOAD} + 110 Ω fixed. Overall TIA gain is 200.		
		10	1 kΩ. If R_{LOAD} ≤ 100 Ω, R_{TIA} gain = (100 Ω − R_{LOAD}) + 1 kΩ. If R_{LOAD} > 100 Ω, R_{TIA} gain is not supported with a 1 kΩ R_{TIA}.		
		11	2 kΩ. R_{TIA} gain = 2 kΩ − (R_{LOAD} − 100 Ω).		
		100	3 kΩ. R_{TIA} gain = 3 kΩ − (R_{LOAD} − 100 Ω).		
		101	4 kΩ. R_{TIA} gain = 4 kΩ − (R_{LOAD} − 100 Ω).		
		110	6 kΩ. R_{TIA} gain = 6 kΩ − (R_{LOAD} − 100 Ω).		
		111	8 kΩ. R_{TIA} gain = 8 kΩ − (R_{LOAD} − 100 Ω).		
		1000	10 kΩ. R_{TIA} gain = 10 kΩ − (R_{LOAD} − 100 Ω).		
		1001	12 kΩ. R_{TIA} gain = 12 kΩ − (R_{LOAD} − 100 Ω).		
		1010	16 kΩ. R_{TIA} gain = 16 kΩ − (R_{LOAD} − 100 Ω).		
		1011	20 kΩ. R_{TIA} gain = 20 kΩ − (R_{LOAD} − 100 Ω).		
		1100	24 kΩ. R_{TIA} gain = 24 kΩ − (R_{LOAD} − 100 Ω).		
		1101	30 kΩ. If R_{LOAD} ≤ 100 Ω, R_{TIA} gain = (100 Ω − R_{LOAD}) + 30 kΩ. If R_{LOAD} > 100 Ω, R_{TIA} gain = 30 kΩ − (R_{LOAD} − 100 Ω).		
		1110	32 kΩ. R_{TIA} gain = 32 kΩ − (R_{LOAD} − 100 Ω).		
		1111	40 kΩ. R_{TIA} gain = 40 kΩ − (R_{LOAD} − 100 Ω).		
		10000	48 kΩ. R_{TIA} gain = 48 kΩ − (R_{LOAD} − 100 Ω).		
		10001	64 kΩ. R_{TIA} gain = 64 kΩ − (R_{LOAD} − 100 Ω).		
		10010	85 kΩ. R_{TIA} gain = 85 kΩ − (R_{LOAD} − 100 Ω).		
		10011	96 kΩ. R_{TIA} gain = 96 kΩ − (R_{LOAD} − 100 Ω).		
		10100	100 kΩ. R_{TIA} gain = 100 kΩ − (R_{LOAD} − 100 Ω).		
		10101	120 kΩ. R_{TIA} gain = 120 kΩ − (R_{LOAD} − 100 Ω).		
		10110	128 kΩ. R_{TIA} gain = 128 kΩ − (R_{LOAD} − 100 Ω).		
		10111	160 kΩ. R_{TIA} gain = 160 kΩ − (R_{LOAD} − 100 Ω).		
		11000	196 kΩ. R_{TIA} gain = 196 kΩ − (R_{LOAD} − 100 Ω).		
		11001	256 kΩ. R_{TIA} gain = 256 kΩ − (R_{LOAD} − 100 Ω).		
		11010	512 kΩ. R_{TIA} gain = 512 kΩ − (R_{LOAD} − 100 Ω).		
[4:3]	IBOOST		Current Boost Control.	0x0	R/W
		00	Normal mode.		
		01	Increase amplifier output stage current to quickly charge external capacitor load. Intended for use with high current sensors like oxygen electrochemical sensor.		
		10	Double TIA and potentiostat amplifiers overall quiescent current. Increases amplifier bandwidth. Useful for diagnostic tests.		

REGISTER DETAILS: LOW POWER TIA/POTENTIOSTAT AND DAC CIRCUITS

Table 111. Bit Descriptions for LPTIACON1 (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		11	Double TIA and potentiostat amplifiers overall quiescent current and increase output stage current. Increases amplifier bandwidth and output current capability. Useful for diagnostic tests with high current sensors like the oxygen electrochemical sensor.		
2	HALFPWR		Half Power Mode Select. Control bit to reduce active power consumption of TIA and potentiostat amplifiers for sensor Channel 1.	0x0	R/W
		0	Normal mode. Default value.		
		1	Reduce potentiostat amplifier and TIA current by half. Degrades performance.		
1	PAPDEN		Potentiostat Amplifier Power-Down. Low-power Potentiostat Amplifier 1 power-down control bit.	0x1	R/W
		0	Power-up.		
		1	Power-down.		
0	TIAPDEN		TIA Power-Down. Low-power TIA1 power-down control bit.	0x1	R/W
		0	Power up.		
		1	Power down.		

LOW-POWER TIA SWITCH CONFIGURATION FOR CHANNEL 1 REGISTER

Address: 0x400C20E0, Reset: 0x00000000, Name: LPTIASW1

See Figure 17 for details on the switches mentioned in this register.

Table 112. Bit Descriptions for LPTIASW1

Bits	Bit Name	Settings	Description	Reset	Access
[31:14]	Reserved		Reserved.	0x0	R
13	TIABIASSEL		TIA SW13 Control Active High.	0x0	R/W
		0	Disconnect TIA bias voltage from the VZERO0 pin.		
		1	Connect TIA bias voltage to the VZERO0 pin.		
12	PABIASSEL		TIA SW12 Control Active High.	0x0	R/W
		0	Disconnect potentiostat amplifier bias voltage from the VBIAS1 pin.		
		1	Connect potentiostat amplifier bias voltage to the VBIAS1 pin.		
11	SW11		SW11 Switch Control Active High.	0x0	R/W
		0	Open switch.		
		1	Close switch.		
10	SW10		SW10 Switch Control Active High.	0x0	R/W
		0	Open switch.		
		1	Close switch.		
9	SW9		SW9 Switch Control Active High.	0x0	R/W
		0	Open switch.		
		1	Close switch.		
8	SW8		SW8 Switch Control Active High.	0x0	R/W
		0	Open switch.		
		1	Close switch.		
7	SW7		SW7 Switch Control Active High.	0x0	R/W
		0	Open switch.		
		1	Close switch.		
6	SW6		SW6 Switch Control Active High.	0x0	R/W
		0	Open switch.		
		1	Close switch.		
5	SW5		SW5 Switch Control Active High. Close to connect external capacitor or R_{TIA} resistor between the RC1_0 and RC1_1 pins.	0x0	R/W
		0	Open switch.		
		1	Close switch.		

REGISTER DETAILS: LOW POWER TIA/POTENTIOSTAT AND DAC CIRCUITS

Table 112. Bit Descriptions for LPTIASW1 (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
4	SW4		SW4 Switch Control Active High.	0x0	R/W
		0	Open switch.		
		1	Close switch.		
3	SW3		SW3 Switch Control Active High.	0x0	R/W
		0	Open switch.		
		1	Close switch.		
2	SW2		SW2 Switch Control Active High.	0x0	R/W
		0	Open switch.		
		1	Close switch.		
1	SW1		SW1 Switch Control Active High.	0x0	R/W
		0	Open switch.		
		1	Close switch.		
0	SW0		SW0 Switch Control Active High.	0x0	R/W
		0	Open switch.		
		1	Close switch.		

LPDAC0 DATA OUT REGISTER

Address: 0x400C2120, Reset: 0x00000000, Name: LPDACDAT0

Table 113. Bit Descriptions for LPDACDAT0

Bits	Bit Name	Settings	Description	Reset	Access
[31:18]	Reserved		Reserved.	0x0	R
[17:12]	DACIN6		6-Bit Value, 1 LSB = 34.375 mV. A low-power DAC0 6-bit output data register value between 0 and 0x3F is expected to set 6-bit output voltage.	0x0	R/W
		0	0.2 V.		
		111111	2.366 V.		
[11:0]	DACIN12		12-Bit Value, 1 LSB = 537 µV. A low-power DAC0 12-bit output data register value between 0 and 0xFFF is expected to set 12-bit output voltage.	0x0	R/W
		0	0.2 V.		
		0xFFF	2.4 V.		

LPDAC0 SWITCH CONTROL REGISTER

Address: 0x400C2124, Reset: 0x00000000, Name: LPDACSW0

Table 114. Bit Descriptions for LPDACSW0

Bits	Bit Name	Settings	Description	Reset	Access
[31:6]	Reserved		Reserved.	0x0	R
5	LPMODEDIS		Switch Control. Controls switches connected to the output of low-power DAC0.	0x0	R/W
		0	Switches connected to output of low-power DAC configured via LPDACCON0, Bit 5. Default.		
		1	Overrides LPDACCON0, Bit 5. Switches connected to the low-power DAC0 output are controlled via LPDACSW0, Bits[4:0].		
4	SW4		LPDAC0 SW4 Control.	0x0	R/W
		0	Disconnect direct connection of VBIAS0 DAC output to positive input of low-power Amplifier 0. Default.		
		1	Connect VBIAS0 DAC output directly to positive input of low-power Amplifier 0.		
3	SW3		LPDAC0 SW3 Control.	0x0	R/W
		0	Disconnect VBIAS0 DAC output from low-pass filter and VBIAS0 pin.		
		1	Connect VBIAS0 DAC output to the low-pass filter and VBIAS0 pin. Default.		
2	SW2		LPDAC0 SW2 Control.		R/W
		0	Disconnect direct connection of VZERO0 DAC output to low-power TIA0 positive input. Default.	0x0	

REGISTER DETAILS: LOW POWER TIA/POTENTIOSTAT AND DAC CIRCUITS

Table 114. Bit Descriptions for LPDACSW0 (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		1	Connect VZERO0 DAC output directly to low-power TIA0 positive input.		
1	SW1		LPDAC0 SW1 Control.	0x0	R/W
		0	Disconnect VZERO0 DAC output from the low-pass filter and VZERO0 pin.		
		1	Connect VZERO0 DAC output to the low-pass filter and VZERO0 pin. Default.		
0	SW0		LPDAC0 SW0 Control.	0x0	R/W
		0	Disconnect VZERO0 DAC output from high-speed TIA positive input. Default.		
		1	Connect VZERO0 DAC output to the high-speed TIA positive input.		

LPDAC0 CONTROL REGISTER

Address: 0x400C2128, Reset: 0x00000002, Name: LPDACCON0

Table 115. Bit Descriptions for LPDACCON0

Bits	Bit Name	Settings	Description	Reset	Access
[31:7]	Reserved		Reserved.	0x0	R
6	WAVETYPE		Low-Power DAC Source.	0x0	R/W
		0	Direct from the LPDACDAT0 register.		
		1	Waveform generator.		
5	DACMDE		LPDAC0 Switch Settings. Control bit for LPDAC0 output switches.	0x0	R/W
		0	LPDAC0 switches set for normal mode. Clear to 0 for normal output switch operation. Default.		
		1	LPDAC0 switches set for diagnostic mode.		
4	VZEROMUX		VZERO0 Mux Select. Select which DAC0 output connects to the VZERO0 node. Ensure that the same value is written to LPDACCON0, Bit 3 = 1. If this bit is cleared to 0, VBIASMUX must be cleared to 0.	0x0	R/W
		0	VZERO0 6-bit. Default. Clear to 0 for VZERO0 output to be 6-bit.		
		1	VZERO0 12-bit. Set to 1 for VZERO0 output to be 12-bit.		
3	VBIASMUX		VBIAS Mux Select. Select which DAC0 output connects to the VBIAS0 node. Ensure that the same value is written to LPDACCON, Bit 4. If this bit is set to 1, the VZEROMUX bit must be set to 1. If this bit is cleared to 0, VZEROMUX bit must be cleared to 0.	0x0	R/W
		0	VBIAS0 12-bit. Default. 12-bit DAC connect to VBIAS0.		
		1	VBIAS0 6-bit. 6-bit DAC connect to VBIAS0.		
2	REFSEL		LPDAC0 Reference Select.	0x0	R/W
		0	VREF_2.5V Reference 0. Selects the low-power 2.5 V reference as the LPDAC0 reference source. Default.		
		1	AVDD Reference 1. Set to 1 to select AVDD as the low-power DAC0 reference.		
1	PWDEN		LPDAC0 Power-Down. Power-down control bit for low-power DAC.	0x1	R/W
		0	LPDAC0 powered on. Power on LPDAC0.		
		1	LPDAC0 powered off. Default. Power down LPDAC0 to open all switches on LPDAC0 output.		
0	RSTEN		Enable Writes to LPDAC0. Enables writes to LPDACDAT0 register.	0x0	R/W
		0	Disable LPDAC0 writes. Default. If cleared to 0, LPDACDAT0 is always 0. Writes to LPDACDAT0 are disabled.		
		1	Enable LPDAC0 writes. Set to 1 to enable writes to the LPDACDAT0 register.		

LPDAC1 DATA OUT REGISTER

Address: 0x400C212C, Reset: 0x00000000, Name: LPDACDAT1

Table 116. Bit Descriptions for LPDACDAT1

Bits	Bit Name	Settings	Description	Reset	Access
[31:18]	Reserved		Reserved.	0x0	R
[17:12]	DACIN6		6-Bit Value, 1 LSB = 34.375 mV. A low-power DAC1 6-bit output data register values between 0 and 0x3F is expected to set 6-bit output voltage.	0x0	R/W
		000000	0.2 V.		

REGISTER DETAILS: LOW POWER TIA/POTENTIOSTAT AND DAC CIRCUITS

Table 116. Bit Descriptions for LPDACDAT1 (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		111111	2.366 V.		
[11:0]	DACIN12		12-Bit Value, 1 LSB = 537 µV. A low-power DAC1 12-bit output data register value between 0 and 0xFFF is expected to set 12-bit output voltage.	0x0	R/W
		0x000	0.2 V.		
		0xFFF	2.4 V.		

LPDAC1 SWITCH CONTROL REGISTER

Address: 0x400C2130, Reset: 0x00000000, Name: LPDACSW1

Table 117. Bit Descriptions for LPDACSW1

Bits	Bit Name	Settings	Description	Reset	Access
[31:6]	Reserved		Reserved.	0x0	R
5	LPMODEDIS		Switch Control. Controls switches connected to the output of LPDAC1.	0x0	R/W
		0	Switches connected to output of low-power DAC configured via LPDACCON1, Bit 5. Default.		
		1	Overrides LPDACCON1, Bit 5. Switches connected to LPDAC1 output are controlled via LPDACSW1, Bits[4:0].		
4	SW4		LPDAC1 SW4 Control.	0x0	R/W
		0	Disconnect direct connection of VBIAS0 DAC output to positive input of low-power Amplifier 1. Default.		
		1	Connect VBIAS1 DAC output directly to positive input of low-power Amplifier 1.		
3	SW3		LPDAC1 SW3 Control.	0x0	R/W
		0	Disconnect VBIAS1 DAC output from low-pass filter and VBIAS1 pin.		
		1	Connect VBIAS1 DAC output to the low-pass filter and VBIAS1 pin. Default.		
2	SW2		LPDAC1 SW2 Control.		R/W
		0	Disconnect direct connection of VZERO0 DAC output to the low-power TIA0 positive input. Default.	0x0	
		1	Connect VZERO0 DAC output directly to the low-power TIA0 positive input.		
1	SW1		Low-Power DAC1 SW1 Control.	0x0	R/W
		0	Disconnect VZERO1 DAC output from the low-pass filter and VZERO1 pin.		
		1	Connect VZERO1 DAC output to the low-pass filter and VZERO1 pin. Default.		
0	SW0		Low-Power DAC1 SW0 Control.	0x0	R/W
		0	Disconnect VZERO1 DAC output from the high-speed TIA positive input. Default.		
		1	Connect VZERO1 DAC output to the high-speed TIA positive input.		

LPDAC1 CONTROL REGISTER

Address: 0x400C2134, Reset: 0x00000002, Name: LPDACCON1

Table 118. Bit Descriptions for LPDACCON1

Bits	Bit Name	Settings	Description	Reset	Access
[31:7]	Reserved		Reserved.	0x0	R
6	WAVETYPE		Low-Power DAC Source.	0x0	R/W
		0	Direct from LPDACDAT1.		
		1	Waveform generator.		
5	DACMDE		Low-Power DAC1 Switch Settings. Control bit for the low-power DAC1 output switches.	0x0	R/W
		0	Low-Power DAC1 switches set for normal mode.		
		1	Low-Power DAC1 switches set for diagnostic mode.		
4	VZEROMUX		VZERO1 Output. Select which low-power DAC1 output connects to the VZERO1 node. Ensure same value is written to the VBIASMUX bit. If this bit is set to 1, LPDACCON1, Bit 3 = 1. If this bit is cleared to 0, LPDACCON1, Bit 3 = 0.	0x0	R/W
		0	VZERO1 6-bit. Default. Clear to 0 for VZERO1 output to be 6-bit.		
		1	VZERO1 12-bit. Set to 1 for VZERO1 output to be 12-bit.		

REGISTER DETAILS: LOW POWER TIA/POTENTIOSTAT AND DAC CIRCUITS

Table 118. Bit Descriptions for LPDACCON1 (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
3	VBIASMUX		Bit Select. Select which low-power DAC1 output connects to the VBIAS1 node. Ensure same value is written to the VZEROMUX bit. If this bit is set to 1, LPDACCON1, Bit 4 = 1. If this bit is cleared to 0, LPDACCON1, Bit 4 = 0.	0x0	R/W
		0	12-bit output. Default. 12-bit DAC connects to the VBIAS1 pin.		
		1	6-bit output. 6-bit DAC connects to the VBIAS1 pin.		
2	REFSEL		Low-Power DAC1 Reference Select.	0x0	R/W
		0	Select 2.5 V from low-power buffer as DAC reference. Default.		
		1	Select AVDD power supply as DAC reference.		
1	PWDEN		Low-Power DAC1 Power. Power-down control bit for low-power DAC1.	0x1	R/W
		0	Low-Power DAC1 powered on. Clear to 0 to power on low-power DAC1.		
		1	Low-Power DAC1 powered off. Default. Power down low-power DAC1 and open all switches on the low-power DAC1 output.		
0	RSTEN		Enable Writes to LPDACDAT1.	0x0	R/W
		0	Disable low-power DAC1 writes. Default. If cleared to 0, LPDACDAT1 is always 0. Writes to LPDACDAT1 are disabled.		
		1	Enable low-power DAC1 writes.		

LOW-POWER REFERENCE CONTROL REGISTER

Address: 0x400C2050, Reset: 0x00000000, Name: LPREFBUFCON

Table 119. Bit Descriptions for LPREFBUFCON

Bits	Bit Name	Settings	Description	Reset	Access
[31:3]	Reserved		Reserved.	0x0	R
2	BOOSTCURRENT		Set this Bit when Using both Channel 0 and Channel 1 Potentiostat Channels.	0x0	R/W
		0	Option to clear to 0 when using only one potentiostat channel to support only one low-power DAC and save power.		
		1	Set to 1 to boost bias current of low-power reference buffer to support driving two low-power DACs.		
1	LPBUF2P6DIS		Low-Power Band Gap Output Buffer. Normally cleared to enable the low-power reference buffer.	0x0	R/W
		0	Enable low-power 2.5 V buffer.		
		1	Power down low-power 2.5 V buffer.		
0	LPREFDIS		Low-Power Band Gap Power-Down Bit. Normally cleared to enable the low-power reference.	0x0	R/W
		0	Low-power reference enabled.		
		1	Low-power reference powered down.		

HIGH-SPEED TIA CIRCUITS

The high-speed TIA is intended for measuring wide bandwidth input signals up to 200 kHz. The output of the high-speed TIA transfers to the main ADC mux, where the high-speed TIA can be selected as the ADC input channel. The high-speed TIA is especially designed for impedance measurements in conjunction with the high-speed DAC and excitation amplifier. To connect the high-speed TIA to the external sensor pins, the user must configure the transmit switches as described in the Tx Switches section.

KEY FEATURES

To turn on the high-speed TIA, set AFECON, Bit 11 = 1. Clear to 0 to power down the high-speed TIA. The high-speed TIA is off by default.

The high-speed TIA has programmable flexibility built into its input signal selection, R_{TIA} selection, R_{LOAD} selection, and common-mode voltage source. The input signal options are as follows:

- SE0 input pin from sense electrode of Channel 0 sensor.
- SE1 input pin from sense electrode of Channel 1 sensor.
- AIN0, AIN1, AIN2, and AIN3/BUF_VREF1V8 input pins.
- DE0 input pin. This is the Diagnostic Electrode 0 pin (DE0). This pin has its own R_{LOAD03} and R_{TIA2_03} options configurable via the DE0RESCON register.
- DE1 input pin. This is the Diagnostic Electrode 1 pin (DE1). This pin has its own R_{LOAD05} and R_{TIA2_05} options configurable via the DE1RESCON register.

The R_{TIA2_x} options are in the 50 Ω to 160 kΩ.1 kΩ range for the DE0 and DE1 inputs. For all other pins, the gain range is 200 Ω to 160 kΩ.

The R_{LOADx} option for the SE0 and SE1 channels is a fixed 100 Ω resistor. However, for the DE0 and DE1 input pins, the R_{LOADx} is programmable. The R_{LOADx} values are 0 Ω, 10 Ω, 30 Ω, 50 Ω, and 100 Ω.

Table 120. Configuration of R_{LOAD} and R_{TIA} of High-Speed TIA When Using DE0 and DE1 Electrodes

DE0RESCON, Bits[7:0] (DE0) and DE1RESCON, Bits[7:0] (DE1) Setting	R_{LOAD03} and R_{LOAD05} Value (Ω)	R_{TIA2_03} and R_{TIA2_05} Value
0xFF	Disconnected	Disconnected
0x00	0	50 Ω
0x18	0	100 Ω
0x38	0	200 Ω
0x58	0	1.1 kΩ
0x60	0	5.1 kΩ
0x68	0	10.1 kΩ
0x70	0	20.1 kΩ
0x78	0	40.1 kΩ
0x80	0	80.1 kΩ
0x88	0	160.1 kΩ
0x9	10	50 Ω
0x21	10	100 Ω
0x39	10	190 Ω
0x59	10	1.09 Ω
0x61	10	5.09 kΩ
0x69	10	10.09 kΩ
0x71	10	20.09 kΩ
0x79	10	40.09 kΩ
0x81	10	80.09 kΩ
0x89	10	160.09 kΩ
0x12	30	50 Ω
0x2A	30	100 Ω
0x4A	30	210 Ω
0x5A	30	1.07 kΩ
0x62	30	5.07 kΩ
0x6A	30	10.07 kΩ

HIGH-SPEED TIA CIRCUITS

Table 120. Configuration of R_{LOAD} and R_{TIA} of High-Speed TIA When Using DE0 and DE1 Electrodes (Continued)

DE0RESCON, Bits[7:0] (DE0) and DE1RESCON, Bits[7:0] (DE1) Setting	R_{LOAD03} and R_{LOAD05} Value (Ω)	R_{TIA2_03} and R_{TIA2_05} Value
0x72	30	20.07 kΩ
0x7A	30	40.07 kΩ
0x82	30	80.07 kΩ
0x8A	30	160.07 kΩ
0x1B	50	50 Ω
0x33	50	100 Ω
0x4B	50	190 Ω
0x5B	50	1.05 kΩ
0x63	50	5.05 kΩ
0x6B	50	10.05 kΩ
0x73	50	20.05 kΩ
0x7B	50	40.05 kΩ
0x83	50	80.05 kΩ
0x8B	50	160.05 kΩ
0x34	100	50 Ω
0x3C	100	100 Ω
0x54	100	200 Ω
0x5C	100	1 kΩ
0x64	100	5 kΩ
0x6C	100	10 kΩ
0x74	100	20 kΩ
0x7C	100	40 kΩ
0x84	100	80 kΩ
0x8C	100	160 kΩ

The high-speed TIA common-mode voltage setting (positive input to the high-speed TIA amplifier) is configurable via HSTIACON, Bits[1:0]. The configuration options are as follows:

- Internal 1.1 V reference source (same as ADCVBIAS_CAP pin voltage).
- Sensor Channel 0 low-power DAC output (VZERO0 pin). See Table 110 and Table 115 for details.
- Sensor Channel 1 low-power DAC output (VZERO1 pin). See Table 112 and Table 118 for details.

Figure 21 shows the high-speed TIA connections to external pins and the programmable switch and resistor locations in this part of the receive signal circuitry.

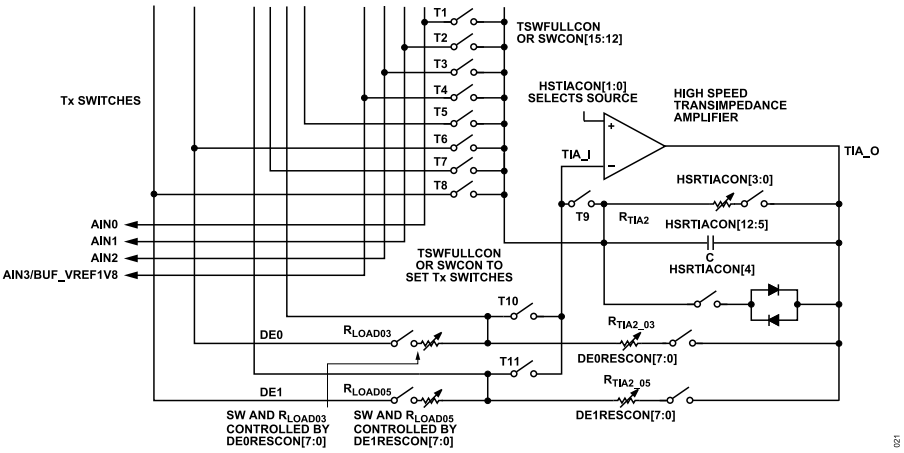

Figure 21. High-Speed TIA Circuitry

HIGH-SPEED TIA CIRCUITS

USING DE0 AND DE1 INPUTS WITH THE HIGH-SPEED TIA

To use DE0 as the input of the high-speed TIA, set the following register values:

- DE0RESCON = value required to set R_{LOAD03} and R_{TIA2_03}. See Table 120.
- HSRTIACON, Bits[3:0] = 0xF to disconnect R_{TIA2} from the high-speed TIA.
- DE1RESCON = 0xFF, to disconnect R_{TIA2_05} from the high-speed TIA.

To use DE1 as the output of the high-speed TIA, set the following register values:

- DE1RESCON = value required to set R_{LOAD05} and R_{TIA2_05}. See Table 120.
- HSRTIACON, Bits[3:0] = 0xF to disconnect R_{TIA2} from the high-speed TIA.
- DE1RESCON = 0xFF to disconnect R_{TIA2_03} from the high-speed TIA.

High-Speed TIA Current Limit Feature

As well as the protection diode, the high-speed TIA also has a built in current limiting feature. If the current sourced or sunk from the TIA is greater than the overcurrent limit protection, the amplifier clamps the current to this limit. The current clamp typically clamps at approximately 17 mA. Refer to the ADuCM355 data sheet for full specifications. Do not use this feature more frequently or for longer than specified in the data sheet.

EXTERNAL R_{TIA} SELECTION

The high-speed TIA has the option of selecting an R_{TIA} instead of the internal R_{TIA2}, R_{TIA2_03}, or R_{TIA2_05} gain options. The DE0 pin or DE1 pin can be connected to one side of the R_{TIA}. This R_{TIA} can be connected to the output of high-speed TIA. The AIN0, AIN1, AIN2, or AIN3/BUF_VREF1V8 pin can be connected to the other side of the external gain resistor, as shown in Figure 22.

To use the DE0 and AIN0 (input) pins to connect the high-speed TIA with an R_{TIA}, set the following register values:

- DE0RESCON = 0x97 for 0 Ω R_{LOAD03} and 0 Ω R_{TIA2_03} in series with an external R_{TIA}.
- DE1RESCON = 0xFF.
- HSRTIACON, Bits[3:0] = 0xF.
- Close the T1 and T10 switches (configure the TSWFULLCON register).

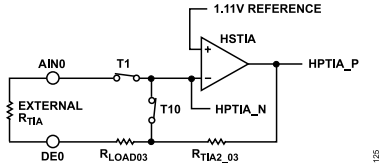

Figure 22. Connecting External R_{TIA} Across the High-Speed TIA

REGISTER SUMMARY: HIGH-SPEED TIA CIRCUITS

AFECON, Bit 11 and AFECON, Bit 5 are relevant to the high-speed TIA block. See Table 52 for more details.

Table 121. High-Speed TIA Circuit Register Summary

Address	Name	Description	Reset	Access
0x400C20F0	HSRTIACON	High-speed R_{TIA} configuration	0x0000000F	R/W
0x400C20F4	DE1RESCON	DE1 high-speed TIA resistor configuration	0x000000FF	R/W
0x400C20F8	DE0RESCON	DE0 high-speed TIA resistor configuration	0x000000FF	R/W
0x400C20FC	HSTIACON	High-speed TIA amplifier configuration	0x00000000	R/W

REGISTER DETAILS: HIGH-SPEED TIA CIRCUITS

HIGH-SPEED R_{TIA} CONFIGURATION REGISTER

Address: 0x400C20F0, Reset: 0x0000000F, Name: HSRTIACON

This register controls the high-speed TIA R_{TIA}, current protection diode, and feedback capacitor.

Table 122. Bit Descriptions for HSRTIACON

Bits	Bit Name	Settings	Description	Reset	Access
[31:13]	Reserved		Reserved.	0x0	R
[12:5]	CTIACON		Configure Capacitor in Parallel with R_{TIA}. This capacitor improves the amplifier loop stability. When the bit is set, the capacitor is added in parallel with R_{TIA}. This capacitor forms a resistor/capacitive (RC) filter with R_{TIA} value selected by the RTIACON bits. Ensure that the cutoff setting is larger than input signal frequency. For best stability, use as large a capacitor value as possible (up to 31 pF).	0x0	R/W
		00000000	0 pF.		
		00000001	1 pF.		
		00000010	2 pF.		
		00000100	4 pF.		
		00001000	8 pF.		
		00010000	16 pF.		
		00100000	Reserved.		
		x1xxxxxx	Not used.		
4	TIASW6CON		SW6 Control. Use SW6 to select whether to use diode in parallel with R_{TIA} or not.	0x0	R/W
		0	SW6 off, diode is not in parallel with R_{TIA}.		
		1	SW6 on, diode is in parallel with R_{TIA}.		
[3:0]	RTIACON		Configure General R_{TIA} Value. To use this R_{TIA}, set TSWFULLCON, Bit 8 = 1 and TSWFULLCON, Bits[10:9] = 00.	0xF	R/W
		0000	R_{TIA} = 200 Ω.		
		0001	R_{TIA} = 1 kΩ.		
		0010	R_{TIA} = 5 kΩ.		
		0011	R_{TIA} = 10 kΩ.		
		0100	R_{TIA} = 20 kΩ.		
		0101	R_{TIA} = 40 kΩ.		
		0110	R_{TIA} = 80 kΩ.		
		0111	R_{TIA} = 160 kΩ.		
		1000 to 1111	R_{TIA} is open.		

DE1 HIGH-SPEED TIA RESISTOR CONFIGURATION REGISTER

Address: 0x400C20F4, Reset: 0x000000FF, Name: DE1RESCON

Table 123. Bit Descriptions for DE1RESCON

Bits	Bit Name	Settings	Description	Reset	Access
[31:8]	Reserved		Reserved.	0x0	R
[7:0]	DE1RCON		DE1 R_{LOAD05} and R_{TIA2_05} Setting. DE1 high-speed TIA resistor settings. To use this R_{LOAD}, open T9 and T10 switches, but close T11 by setting TSWFULLCON, Bits[10:8] = 0b100. To set R_{LOAD05} and R_{TIA2_05} resistor values, see Table 120.	0xFF	R/W

DE0 HIGH-SPEED TIA RESISTOR CONFIGURATION REGISTER

Address: 0x400C20F8, Reset: 0x000000FF, Name: DE0RESCON

Table 124. Bit Descriptions for DE0RESCON

Bits	Bit Name	Settings	Description	Reset	Access
[31:8]	Reserved		Reserved.	0x0	R

REGISTER DETAILS: HIGH-SPEED TIA CIRCUITS

Table 124. Bit Descriptions for DE0RESCON (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
[7:0]	DE0RCON		DE0 R_{LOAD03} and R_{TIA2_03} Setting. DE0 high-speed TIA resistor settings. To use this R_{LOAD}, open the switches T9 and T11, but close T10 by setting TSWFULLCON, Bits[10:8] = 0b010. To set the R_{LOAD03} and R_{TIA2_03} resistor values, see Table 120.	0xFF	R/W

HIGH-SPEED TIA AMPLIFIER CONFIGURATION REGISTER

Address: 0x400C20FC, Reset: 0x00000000, Name: HSTIACON

Table 125. Bit Descriptions for HSTIACON

Bits	Bit Name	Settings	Description	Reset	Access
[31:2]	Reserved		Reserved.	0x0	R
[1:0]	VBIASSEL		Select High-Speed TIA Positive Input.	0x0	R/W
		00	ADCVBIAS_CAP. 1.1 V voltage source.		
		01	V_{ZERO} output from low-power DAC0.		
		10	V_{ZERO} output from low-power DAC1.		
		11	Reserved.		

HIGH-SPEED DAC CIRCUITS

The 12-bit high-speed DAC generates an AC excitation signal when measuring the impedance of an external sensor. The DAC output signal can be controlled directly by writing to a data register or by the automated waveform generator block. The high-speed DAC signal is fed to an excitation amplifier designed specifically to couple the AC signal on top of the normal DC bias voltage of the sensor. Alternatively, the high-speed DAC can be used as a normal voltage source. See the Calibrating the High-Speed DAC section for more details.

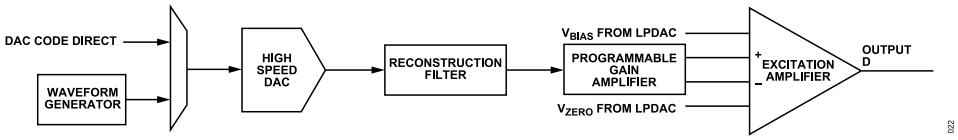

Figure 23. Overview of High-Speed DAC Blocks

HIGH-SPEED DAC OUTPUT SIGNAL GENERATION

There are two ways of setting the high-speed DAC output voltage, which are as follows:

- Direct write to the DAC code register. Write to the HSDACDAT register, a 12-bit register where the MSB is a sign bit. A value of 0x800 results in a 0 V output. 0xFFF is positive full scale. 0x000 is negative full scale.
- Use the automatic waveform generator. The waveform generator can be programmed to generate fixed frequency and fixed amplitude signals. If the user selects the sine wave, options exist to adjust the offset and phase of the output signal.

To use the waveform generator to generate a sine wave, follow these steps:

1. Set AFECON, Bit 14 = 1 to turn on the waveform generator.
2. Set WGCON, Bits[2:1] = 10 to select sine waveforms.
3. Set WGAMPLITUDE, Bits[10:0] to set up the sine wave amplitude. The sine wave automatically swings above or below the common-mode voltage. As such, there are only 11 bits required for the amplitude control.
4. Set WGFCW, Bits[23:0] to set the sine wave output frequency. For output frequencies higher than 80 kHz, the high-speed DAC must be configured for high-power mode. See the Power Mode Configuration Register section for more details. For this configuration, use the equation

$$f_{OUT} = f_{ACLK} \times \frac{WGFCW\ Bits[23:0]}{2^{30}} \quad (11)$$

where:
f_{OUT} is the output frequency.
f_{ACLK} is the analog clock frequency, 16 MHz.

HIGH-SPEED DAC CORE POWER MODES

The reference source of the high-speed DAC is an internal 1.8 V precision reference voltage.

There are three basic modes of operation of the high-speed DAC that trade power consumption and output speed.

Low-Power Mode

When configuring the high-speed DAC for low-power mode, take note of the following requirements and features:

- Clear PMBW, Bit 0 = 0 to minimize current consumption. This setting is recommended when the high-speed DAC output frequency must be ≤80 kHz.
- In low-power mode, the system clock to the DAC and the ADC is 16 MHz.
- Ensure that CLKSEL, Bits[1:0] selects a 16 MHz clock source. For example, an internal high-speed oscillator is selected if CLKSEL, Bits[1:0] = 00. Ensure that the system clock divide ratio is 1 (CLKCON0, Bits[5:0] = 0 or 1).
- If the internal high-speed oscillator is selected as the system clock source, ensure that the 16 MHz option is selected. Set HPOSCCON, Bit 2 = 1.

HIGH-SPEED DAC CIRCUITS

High-Power Mode

When configuring the high-speed DAC for high-power mode, take note of the following requirements and features:

- Increases the bandwidth supported by the high-speed DAC amplifiers.
- Set PMBW, Bit 0 = 1. Power consumption is increased, but the output signal bandwidth increases to a maximum of 200 kHz.
- In this mode, the system clock to the DAC and the ADC must be set to 32 MHz.
- Ensure CLKSEL, Bits[1:0] selects a 32 MHz clock source. For example, an internal high-speed oscillator is selected if CLKSEL, Bits[1:0] = 00. Ensure that the system clock divide ratio is 1 (CLKCON0, Bits[5:0] = 0 or 1).
- If the internal high-speed oscillator is selected as the system clock source, ensure that the 32 MHz option is selected. Clear HPOSCCON, Bit 2 = 0.

Hibernate Mode

When configuring the high-speed DAC for hibernate mode, take note of the following requirements and features:

- When the ADuCM355 enters hibernate mode, the clocks to the high-speed DAC circuits are clock gated to save power.
- When the ADuCM355 is in active mode but the high-speed DAC is not required, disable the high-speed DAC circuits to save power. To do this, clear Bit 20, Bit 14, Bit 10, Bit 9, and Bit 6 in the AFECON register. Leave Bit 5 set if the ADC is in use, because Bit 5 controls the high-power reference source.

RECOMMENDED CONFIGURATION IN HIBERNATE MODE

To minimize leakage on the switches connected to the excitation amplifier P and N nodes and to minimize leakage on the high-speed TIA, tie the switches to the internal 1.8 V LDO regulator generated voltage, as follows:

- Close the PL and PL2 switches. PSWFULLCON, Bits[14:13] = 11.
- Close the NL and NL2 switches. NSWFULLCON, Bits[11:10] = 11.

In hibernate mode, assume that only the DC bias voltage from the low-power amplifiers is required for the sensor.

HIGH-SPEED DAC FILTER OPTIONS

The high-speed DAC has a configurable reconstruction filter on its output stage. It is important that this filter is configured appropriately depending on the output signal frequency of the DAC. PMBW, Bits[3:2] configure the 3 dB cutoff frequency of the filters. Ensure that the cutoff frequency is higher than the required DAC output frequency. The output filter cutoff frequency details are as follows:

- PMBW, Bits[3:2] = 01 for optimal performance if the DAC output signal frequency is <80 kHz.
- PMBW, Bits[3:2] = 10 for optimal performance if the DAC output signal frequency is <100 kHz.
- PMBW, Bits[3:2] = 11 for optimal performance if the DAC output signal frequency is <250 kHz.

HIGH-SPEED DAC OUTPUT ATTENUATION OPTIONS

Scaling options for the high-speed DAC output exist to modify the output signal amplitude to the sensor. The output of the 12-bit DAC string before any attenuation or gain is approximately ±300 mV. At the DAC output, there is a 1 or 0.2 gain stage that is controlled by HSDACCON, Bit 0. At the PGA stage, there is a 2 or 0.25 gain option that is controlled by HSDACCON, Bit 12.

It is recommended to only use the maximum and minimum overall gain options with the lowest DAC code of 0x200 and maximum DAC code of 0xE00. The characterized DAC output ranges are as follows:

- HSDACCON, Bit 12 = 0 and HSDACCON, Bit 0 = 0. The overall gain is 2. This setting gives a full-scale voltage of approximately ±607 mV to the sensor for HSDACDAT Code 0x200 to Code 0xE00.
- HSDACCON, Bit 12 = 1 and HSDACCON, Bit 0 = 1. This setting gives a full-scale voltage of approximately ±15.1 mV to the sensor for HSDACDAT Code 0x200 to Code 0xE00.

HIGH-SPEED DAC CIRCUITS

COUPLING AN AC SIGNAL FROM HIGH-SPEED DAC ONTO THE DC LEVEL SET BY LOW-POWER DAC

The ADuCM355 contains two independent low-power potentiostat channels that configure two separate electrochemical sensors. In normal operation, the bias voltage of the sensor between the reference electrode and working electrode is set directly by the low-power DAC outputs, VBIASx and VZEROx. See Figure 15 for the setup.

In normal operation, the high-speed DAC circuits are not used. However, to connect an AC signal onto the counter electrode, the potentiostat amplifier must be disconnected from the sensor and the whole signal must be applied from the high-speed DAC excitation amplifier output. The bias voltage setting of the sensor must also be completed by the high-speed TIA, rather than the low-power TIA. The AC signal generated by the high-speed DAC is coupled onto the DC voltage level set by the low-power DAC for the channel under test. The DACDCBUFCON register, Bit 1 selects LPDAC0 or LPDAC1 as the DC level voltage source that couples to the high-speed DAC.

The DAC DC buffers shown in Figure 24 are enabled by setting AFECON, Bit 21 = 1. This setting feeds the sensor DC bias voltage to the excitation amplifier. For the appropriate LPDACx channel, set LPTIASWx, Bits[11:0] = 0x180 to set the low-power TIA and potentiostat switches for AC impedance measurement mode.

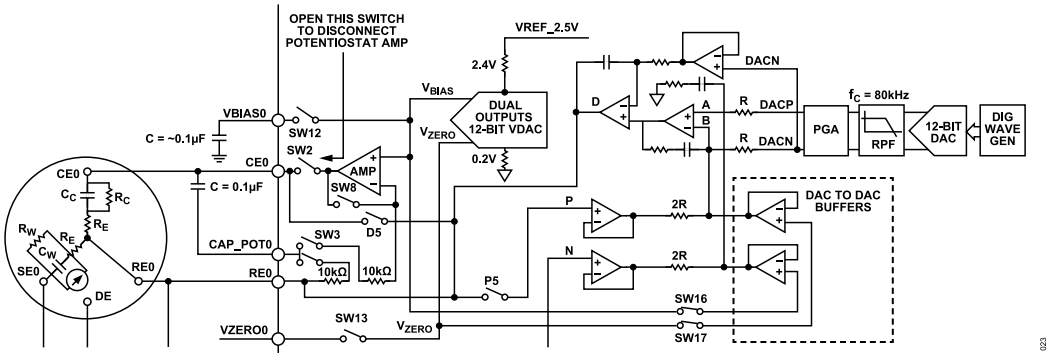

Figure 24. Signal Path for AC Signal Coupled onto DC Level Set by Low-Power DAC

AVOIDING INCOHERENCY ERRORS BETWEEN EXCITATION AND MEASUREMENT FREQUENCIES DURING IMPEDANCE MEASUREMENTS

Table 126 details the recommended settings to avoid incoherency errors between excitation frequencies and measurement frequencies during impedance measurements.

Table 126. Recommended Settings to Avoid Incoherency Errors

Parameter	Recommended Settings
Hanning Window	Always on (DFTCON, Bit 0 = 1). Enabling the Hanning window avoids issues due to incoherency. Disabling the Hanning window can result in degraded performance.
High-Speed DAC Update Rate	In low-power mode, the typical value is 16 MHz or 27 MHz. (HSDACCON, Bits[8:1] = 0x1B). In high-power mode, the typical value is 32 MHz or 7 MHz. (HSDACCON, Bits[8:1] = 0x7).
ADC Sampling Rate	Low-power mode, 800 kSPS, high frequency oscillator = 16 MHz. High-power mode, 1.6 MSPS, high frequency oscillator = 32 MHz.

CALIBRATING THE HIGH-SPEED DAC

The high-speed DAC is not calibrated during production testing. This section describes the calibration of the high-speed DAC for all gain settings in low and high-power modes. Calibrate the high-speed DAC if it is intended to generate an excitation signal to a sensor. If an offset error exists on the excitation signal, and a current or voltage output must be measured, the DAC output voltage can exceed the headroom of the selected TIA or ADC input buffer and PGA setting.

Calibrate the high-speed DAC with the required HSDACCON, Bit 12 and HSDACCON, Bit 0 settings. For example, if the DAC is calibrated with HSDACCON, Bit 12 = 0 and HSDACCON, Bit 0 = 0, and the user changes HSDACCON, Bit 12 = 1, an error is introduced if the DACOFFSET register or DACOFFSETHP register is not recalibrated for the new output range.

The high-speed DAC is a differential output DAC that swings on the voltage applied to the excitation N node of the amplifier. Figure 26 shows the connections of the high-speed DAC to the external calibration resistor (R_{CAL}) and internally to the ADC.

HIGH-SPEED DAC CIRCUITS

To calibrate the offset, ensure that the differential voltage measured across R_{CAL} is 0 V. It is important to ensure the offset error is calibrated for the chosen high-speed DAC output range and power mode.

Gain calibration is optional and adjusts the peak-to-peak voltage swing. The peak-to-peak voltage swing can also be adjusted by changing the minimum and maximum DAC codes.

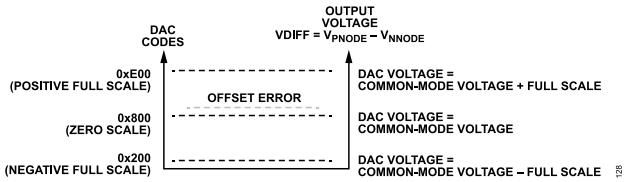

Figure 25. High-Speed DAC Transfer Function

The high-speed DAC transfer function is shown in Figure 25. Note that in Figure 26 the common-mode voltage setting for the calibration circuit is set by the noninverting input of the high speed TIA.

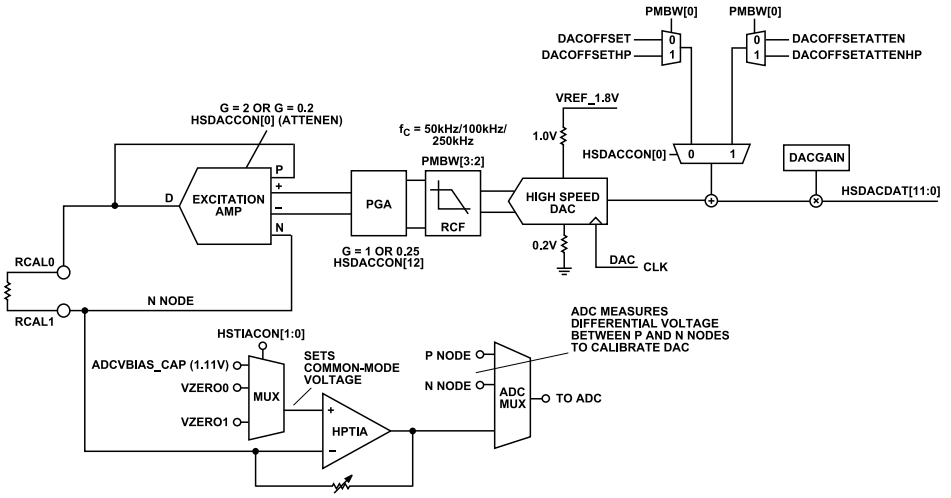

Figure 26. High-Speed DAC Calibration Circuit Using R_{CAL}

The example functions provided in the EVAL-ADuCM355QSPZ kit demonstrate how to use the ADC to measure the differential voltage across R_{CAL} and how to adjust the appropriate offset calibration resistor until this differential voltage is approximately 0 V.

Table 127 shows the appropriate calibration for each high-speed DAC output setting and the selected power mode of the analog die.

Table 127. High-Speed Calibration Register Assignment for Each Output Range

			Relevant Calibration Registers		
HSDACCON, Bit 12 Setting	HSDACCON, Bit 0 Setting	Typical Output Range (Code 0x200 to Code 0xE00)	Low-Power Mode	High-Power Mode	Low-Power Mode/ High-Power Mode
0	0	±607 mV	DACOFFSET	DACOFFSETHP	DACGAIN
1	0	±75 mV	DACOFFSET	DACOFFSETHP	DACGAIN
1	1	±15.14 mV	DACOFFSETATTEN	DACOFFSETATTENHP	DACGAIN
0	1	±121.2 mV	DACOFFSETATTEN	DACOFFSETATTENHP	DACGAIN

REGISTER SUMMARY: HIGH-SPEED DAC CIRCUITS

Table 128. High-Speed DAC Control Register Summary

Address	Name	Description	Reset	Access
0x400C2010	HSDACCON	High-speed DAC configuration	0x0000001E	R/W
0x400C2048	HSDACDAT	Direct write to DAC output control value	0x00000800	R/W
0x400C2104	DACDCBUFCON	DAC DC buffer configuration	0x00000000	R/W
0x400C2260	DACGAIN	DAC gain	0x00000800	R/W
0x400C2264	DACOFFSETATTEN	DAC offset with attenuator enabled (low-power mode)	0x00000000	R/W
0x400C2268	DACOFFSET	DAC offset with attenuator disabled (low-power mode)	0x00000000	R/W
0x400C22B8	DACOFFSETATTENHP	DAC offset with attenuator enabled (high-power mode)	0x00000000	R/W
0x400C22BC	DACOFFSETHP	DAC offset with attenuator disabled (high-power mode)	0x00000000	R/W

Table 129. Waveform Generator for High-Speed DAC Register Summary

Address	Name	Description	Reset	Access
0x400C2014	WGCON	Waveform generator configuration	0x00000030	R/W
0x400C2030	WGFCW	Waveform generator for sinusoid frequency control word	0x00000000	R/W
0x400C2034	WGPHASE	Waveform generator for sinusoid phase offset	0x00000000	R/W
0x400C2038	WGOFFSET	Waveform generator for sinusoid offset	0x00000000	R/W
0x400C203C	WGAMPLITUDE	Waveform generator for sinusoid amplitude	0x00000000	R/W

REGISTER DETAILS: HIGH-SPEED DAC CIRCUITS

HIGH-SPEED DAC CONFIGURATION REGISTER

Address: 0x400C2010, Reset: 0x0000001E, Name: HSDACCON

Table 130. Bit Descriptions for HSDACCON

Bits	Bit Name	Settings	Description	Reset	Access
[31:13]	Reserved		Reserved.	0x0	R
12	INAMPGNMDE		Excitation Amplifier Gain Control. Selects the gain of excitation amplifier.	0x0	R/W
		0	Gain = 2.		
		1	Gain = 0.25. HSDACCON, Bit 0 must also be set to 1 for this option.		
[11:9]	Reserved		Reserved.	0x0	R/W
[8:1]	RATE		DAC Update Rate. DAC update rate = ACLK/HSDACCON, Bits[8:1]. ACLK can be a high-speed oscillator at 16 MHz or 32 MHz or a low-power oscillator at 32 kHz.	0xF	R/W
0	ATTENEN		PGA Stage Gain Attenuation. Enable the PGA attenuator at the output of the DAC.	0x0	R/W
		0	DAC attenuator disabled. Gain of 1.		
		1	DAC attenuator enabled. Gain of 0.2. HSDACCON, Bit 12 must also be set to 1 for this option.		

DIRECT WRITE TO DAC OUTPUT CONTROL VALUE REGISTER

Address: 0x400C2048, Reset: 0x00000800, Name: HSDACDAT

Table 131. Bit Descriptions for HSDACDAT

Bits	Bit Name	Settings	Description	Reset	Access
[31:12]	Reserved		Reserved.	0x0	R
[11:0]	DACDAT		DAC Code. Written directly to DAC. Minimum code is 0x000 and maximum code is 0xFFF. Midscale (0x800) corresponds to an output voltage of 0 V.	0x800	R/W

DAC DC BUFFER CONFIGURATION REGISTER

Address: 0x400C2104, Reset: 0x00000000, Name: DACDCBUFCON

Table 132. Bit Descriptions for DACDCBUFCON

Bits	Bit Name	Settings	Description	Reset	Access
[31:2]	Reserved		Reserved.	0x0	R
1	CHANSEL		DAC DC Channel Selection. Selects the low-power DAC output channel onto which to couple a high-speed DAC AC signal.	0x0	R/W
		0	LPDAC0 sets DC level. Selects LPDAC0 as excitation amplifier common-mode level.		
		1	LPDAC1 sets DC level. Selects LPDAC1 as excitation amplifier common-mode level.		
0	Reserved		Reserved.	0x0	R/W

DAC GAIN REGISTER

Address: 0x400C2260, Reset: 0x00000800, Name: DACGAIN

Protected by CALDATLOCK. Valid for all settings of HSDACCON Bit 12 and HSDACCON Bit 0.

Table 133. Bit Descriptions for DACGAIN

Bits	Bit Name	Settings	Description	Reset	Access
[31:12]	Reserved		Reserved.	0x0	R
[11:0]	VALUE		High-Speed DAC Gain Correction Factor. Unsigned number.	0x800	R/W
		0x000	Maximum negative gain adjustment occurs.		
		0x800	No gain adjustment.		
		0xFFF	Maximum positive gain adjustment occurs.		

REGISTER DETAILS: HIGH-SPEED DAC CIRCUITS

DAC OFFSET WITH ATTENUATOR ENABLED (LOW-POWER MODE) REGISTER

Address: 0x400C2264, Reset: 0x00000000, Name: DACOFFSETATTEN

The LSB adjustment is typically 4.9 µV for HSDACCON, Bit 12 = 1 and HSDACCON, Bit 0 = 1. The LSB adjustment is typically 24.7 µV for HSDACCON, Bit 12 = 1 and HSDACCON, Bit 0 = 0.

Table 134. Bit Descriptions for DACOFFSETATTEN

Bits	Bit Name	Settings	Description	Reset	Access
[31:12]	Reserved		Reserved.	0x0	R
[11:0]	VALUE		DAC Offset Correction Factor. Signed number represented in twos complement format with a 0.5 LSB precision. Used when attenuator is enabled.	0x0	R/W
		0x7FF	2^{10} − 0.5. Maximum positive adjustment. Results in positive full scale/2 − 0.5 LSB adjustment.		
		0x001	0.5. Results in 0.5 LSB adjustment.		
		0x000	0. No offset adjustment.		
		0xFFF	−0.5. Results in −0.5 LSB adjustment.		
		0x800	-2^{10}. Maximum negative adjustment. Results in negative full scale/2 adjustment.		

DAC OFFSET WITH ATTENUATOR DISABLED (LOW-POWER MODE) REGISTER

Address: 0x400C2268, Reset: 0x00000000, Name: DACOFFSET

The LSB adjustment is typically 197.7 µV for HSDACCON Bit 12 = 0 and HSDACCON Bit 0 = 0. The LSB adjustment is typically 39.5 µV for HSDACCON, Bit 12 = 0 and HSDACCON, Bit 0 = 1.

Table 135. Bit Descriptions for DACOFFSET

Bits	Bit Name	Settings	Description	Reset	Access
[31:12]	Reserved		Reserved.	0x0	R
[11:0]	VALUE		DAC Offset Correction Factor. Signed number represented in twos complement format with a 0.5 LSB precision. Used when attenuator is disabled.	0x0	R/W
		0x7FF	2^{10} − 0.5. Maximum positive adjustment. Results in positive full scale/2 − 0.5 LSB adjustment.		
		0x001	0.5. Results in 0.5 LSB adjustment.		
		0x000	0. No offset adjustment.		
		0xFFF	−0.5. Results in −0.5 LSB adjustment.		
		0x800	-2^{10}. Maximum negative adjustment. Results in negative full scale/2 adjustment.		

DAC OFFSET WITH ATTENUATOR ENABLED (HIGH-POWER MODE) REGISTER

Address: 0x400C22B8, Reset: 0x00000000, Name: DACOFFSETATTENHP

Protected by the CALDATLOCK register. The LSB adjustment is typically 4.9 µV for HSDACCON, Bit 12 = 1 and HSDACCON, Bit 0 = 1. The LSB adjustment is typically 24.7 µV for HSDACCON, Bit 12 = 1 and HSDACCON, Bit 0 = 0.

Table 136. Bit Descriptions for DACOFFSETATTENHP

Bits	Bit Name	Settings	Description	Reset	Access
[31:12]	Reserved		Reserved.	0x0	R
[11:0]	VALUE		DAC Offset Correction Factor. Signed number represented in twos complement format with a 0.5 LSB precision. Used when attenuator is enabled.	0x0	R/W
		0x7FF	2^{10} − 0.5. Maximum positive adjustment. Results in positive full scale/2 − 0.5 LSB adjustment.		
		0x001	0.5. Results in 0.5 LSB adjustment.		
		0x000	0. No offset adjustment.		
		0xFFF	−0.5. Results in −0.5 LSB adjustment.		
		0x800	-2^{10}. Maximum negative adjustment. Results in negative full scale/2 adjustment.		

REGISTER DETAILS: HIGH-SPEED DAC CIRCUITS

DAC OFFSET WITH ATTENUATOR DISABLED (HIGH-POWER MODE) REGISTER

Address: 0x400C22BC, Reset: 0x00000000, Name: DACOFFSETHP

This register is protected by the CALDATLOCK register. The LSB adjustment is typically 197.7 µV for HSDACCON, Bit 12 = 0 and HSDACCON, Bit 0 = 0. The LSB adjustment is typically 39.5 µV for HSDACCON, Bit 12 = 0 and HSDACCON, Bit 0 = 1.

Table 137. Bit Descriptions for DACOFFSETHP

Bits	Bit Name	Settings	Description	Reset	Access
[31:12]	Reserved		Reserved.	0x0	R
[11:0]	VALUE		DAC Offset Correction Factor. Signed number represented in twos complement format with a 0.5 LSB precision. Used when attenuator is disabled.	0x0	R/W
		0x7FF	2^{10} − 0.5. Maximum positive adjustment. Results in positive full scale/2 − 0.5 LSB adjustment.		
		0x001	0.5. Results in 0.5 LSB adjustment.		
		0x000	0. No offset adjustment.		
		0xFFF	−0.5. Results in −0.5 LSB adjustment.		
		0x800	-2^{10}. Maximum negative adjustment. Results in negative full scale/2 adjustment.		

WAVEFORM GENERATOR CONFIGURATION REGISTER

Address: 0x400C2014, Reset: 0x00000030, Name: WGCON

Table 138. Bit Descriptions for WGCON

Bits	Bit Name	Settings	Description	Reset	Access
[31:5]	Reserved		Reserved.	0x0	R
5	DACGAINCAL		DAC Gain Enable. Use the DAC gain calculated during the Analog Devices factory trim and stored in the DACGAIN register.	0x1	R/W
		0	Bypass DAC gain correction.		
		1	Enable DAC gain correction using value in the DACGAIN register.		
4	DACOFFSETCAL		Bypass DAC Offset. Use the DAC offset calculated during the calibration routine.	0x1	R/W
		0	Bypass DAC offset correction.		
		1	Enable DAC offset correction. The offset value is in the DACOFFSET register or DACOFFSETHP register for low-power mode and high-power mode when HSDACCON, Bit 0 = 0. The offset value is in the DACOFFSETATTEN register or the DACOFFSETATTENHP register for low-power mode and high-power mode when HSDACCON, Bit 0 = 1.		
3	Reserved		Reserved.	0x0	R
[2:1]	TYPESEL		Selects the Type of Waveform.	0x0	R/W
		00	Direct write to DAC. User code writes to the HSDACDAT register directly.		
		10	Sinusoid. Set AFECON, Bit 4 to 1, set this bit to 10, and the DAC outputs a sine wave.		
		11, 01	Reserved.		
0	Reserved		Reserved. Clear to 0 always.	0x0	W

WAVEFORM GENERATOR FOR SINUSOID FREQUENCY CONTROL WORD REGISTER

Address: 0x400C2030, Reset: 0x00000000, Name: WGFCW

Table 139. Bit Descriptions for WGFCW

Bits	Bit Name	Settings	Description	Reset	Access
[31:24]	Reserved		Reserved.	0x0	R
[23:0]	SINEFCW		Sinusoid Generator Frequency Control Word. Selects the output frequency of the sinusoid waveform. By default, the output frequency ACLK frequency × (SINEFCW/2^{30}).	0x0	R/W

REGISTER DETAILS: HIGH-SPEED DAC CIRCUITS

WAVEFORM GENERATOR FOR SINUSOID PHASE OFFSET REGISTER

Address: 0x400C2034, Reset: 0x00000000, Name: WGPHASE

Table 140. Bit Descriptions for WGPHASE

Bits	Bit Name	Settings	Description	Reset	Access
[31:20]	Reserved		Reserved.	0x0	R
[19:0]	SINEOFFSET		Sinusoid Phase Offset. SINOFFSET, Bits[19:0] = phase (degrees)/360 × 2^{20}. For example, to achieve a 45° phase offset, SINOFFSET, Bits[19:0] = 45/360 × 2^{20}. This MMR must be set before setting WGCON, Bits[2:1] and AFECON, Bit 14.	0x0	R/W

WAVEFORM GENERATOR FOR SINUSOID OFFSET REGISTER

Address: 0x400C2038, Reset: 0x00000000, Name: WGOFFSET

Table 141. Bit Descriptions for WGOFFSET

Bits	Bit Name	Settings	Description	Reset	Access
[31:12]	Reserved		Reserved.	0x0	R
[11:0]	SINEOFFSET		Sinusoid Offset. Added to the waveform generator output in sinusoid mode. Signed number represented in twos complement format. This MMR must be set before setting WGCON, Bits[2:1] and AFECON, Bit 14.	0x0	R/W

WAVEFORM GENERATOR FOR SINUSOID AMPLITUDE REGISTER

Address: 0x400C203C, Reset: 0x00000000, Name: WGAMPLITUDE

Table 142. Bit Descriptions for WGAMPLITUDE

Bits	Bit Name	Settings	Description	Reset	Access
[31:11]	Reserved		Reserved.	0x0	R
[10:0]	SINEAMPLITUDE		Sinusoid Amplitude. Unsigned number. Scales the waveform generator in sinusoid mode. The DAC output voltage is determined by the value in HSDACCON, Bit 0 and HSDACCON, Bit 12. This MMR must be set before setting WGCON, Bits[2:1] and AFECON, Bit 14.	0x0	R/W

PROGRAMMABLE SWITCHES CONNECTING THE EXTERNAL SENSOR TO THE HIGH-SPEED DAC AND HIGH-SPEED TIA

The ADuCM355 provides flexibility for connecting external pins to the high-speed DAC excitation amplifier and to the high-speed TIA inverting input. The external pin connections provide many options for impedance measurements to different sensor types and allow an AC signal to be coupled onto the DC bias voltage of a sensor. When configuring these switches, the user must account for the switch settings on the output of the low-power amplifiers. See Figure 16. On power-up, all switches are open to disconnect the sensor. For details on the recommended configuration in hibernate mode, see the Recommended Configuration in Hibernate Mode section.

DX SWITCHES

These switches select the pin to connect to the high-speed DAC excitation amplifier output. For an electrochemical gas sensor impedance measurement, this pin is the CE0 or CE1 pin. The Dx switches can be connected to an external calibration resistor via the RCAL0 pin if the DR0 switch is closed.

PX SWITCHES

These switches select the pin to connect to the high-speed DAC excitation amplifier P input. For an electrochemical gas sensor, this pin is typically RE0 or RE1. The Px switches can be connected to an external R_{CAL} via the RCAL0 pin if the PR0 switch is closed.

NX SWITCHES

These switches select the pin to connect to the high-speed DAC excitation amplifier N input. The Nx switches can be connected to an external R_{CAL} via the RCAL1 pin if the NR1 switch is closed.

TX SWITCHES

These switches select the pin to connect to the high-speed TIA inverting input. The Tx switches can be connected to an external R_{CAL} via the RCAL1 pin if the TR1 switch is closed.

OPTIONS FOR CONTROLLING ALL SWITCHES

Figure 27 shows all of the switches connected to the high-speed DAC excitation amplifier and to the inverting input of the high-speed TIA.

PROGRAMMABLE SWITCHES CONNECTING THE EXTERNAL SENSOR TO THE HIGH-SPEED DAC AND HIGH-SPEED TIA

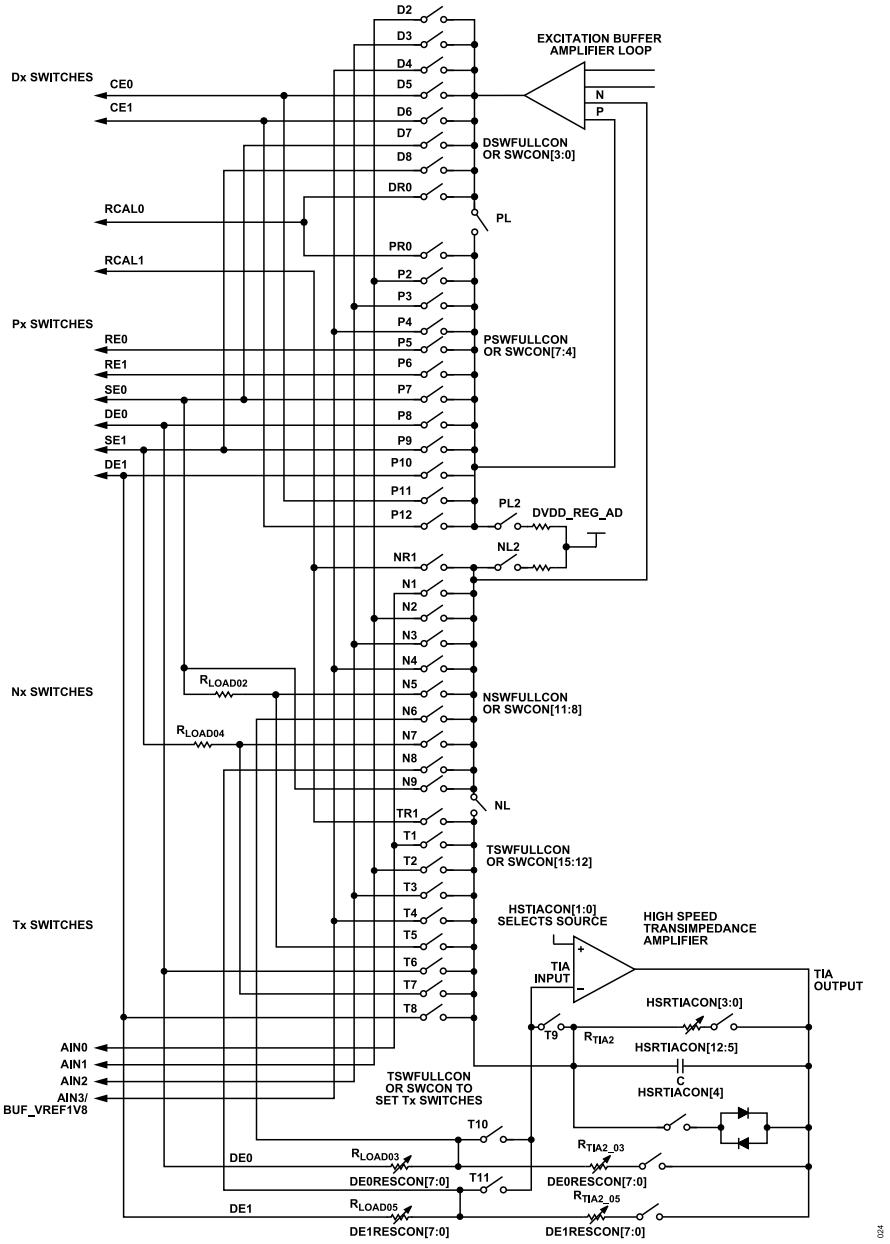

Figure 27. Switch Matrix Block Diagram, All Switches Connected to High-Speed DAC Excitation Amplifier and Inverting Input of High-Speed TIA

There are two options to control the programmable switches to the high-speed DAC excitation amplifier and the high-speed TIA inverting input: SWCON, Bit 16 = 0 and SWCON, Bit 16 =1.

For SWCON, Bit 16 = 0, the bit selects the SWCON register as the control source for these switches. The transmit, Nx, Px, and Dx switches are controlled in groups as follows:

- Transmit switches are controlled via SWCON, Bits[15:12]. A single external pin is selected as the T input to the inverting input of the high-speed TIA.
- Nx switches are controlled via SWCON, Bits[11:8]. A single external pin is selected as the N input to the excitation amplifier of the high-speed DAC.
- Px switches are controlled via SWCON, Bits[7:4]. A single external pin is selected as the P input to the excitation amplifier of the high-speed DAC.
- Dx switches are controlled via SWCON, Bits[3:0]. A single external pin is selected as the D output of the excitation amplifier of the high-speed DAC.

PROGRAMMABLE SWITCHES CONNECTING THE EXTERNAL SENSOR TO THE HIGH-SPEED DAC AND HIGH-SPEED TIA

For SWCON, Bit 16 = 1, each switch can be individually configured as follows:

- The Dx switches are controlled via the DSWFULLCON register bits.
- The Px switches are controlled via the PSWFULLCON register bits.
- The Nx switches are controlled via the NSWFULLCON register bits.
- The transmit switches (including T9, T10, and T11) are controlled via the TSWFULLCON register bits.

Perform the following steps if SWCON, Bit 16 = 1 and if the xSWFULLCON registers are used:

1. Clear SWCON, Bit 16 = 0.
2. Write to DSWFULLCON, TSWFULLCON, NSWFULLCON, and PSWFULLCON.
3. Set SWCON, Bit 16 = 1. This final write ensures all switches update simultaneously.

The status of all switches can be read from the switch status registers at any time. These registers indicate whether each switch is open or closed. The full switch status registers are available in the Register Details: Programmable Switches section.

REGISTER SUMMARY: PROGRAMMABLE SWITCHES

The status of all the switches can be read from the switch status registers at any time. These statuses indicate whether each switch is open or closed. The switch status registers are DSWSTA, PSWSTA, NSWSTA, and TSWSTA.

Table 143. Programmable Switch Matrix Register Summary

Address	Name	Description	Reset	Access
0x400C200C	SWCON	Switch matrix configuration	0x0000FFFF	R/W
0x400C2150	DSWFULLCON	Dx switch matrix full configuration	0x00000000	R/W
0x400C2154	NSWFULLCON	Nx switch matrix full configuration	0x00000000	R/W
0x400C2158	PSWFULLCON	Px switch matrix full configuration	0x00000000	R/W
0x400C215C	TSWFULLCON	Tx switch matrix full configuration	0x00000000	R/W
0x400C21B0	DSWSTA	Dx switch matrix status	0x00000000	R
0x400C21B4	PSWSTA	Px switch matrix status	0x00000000	R
0x400C21B8	NSWSTA	Nx switch matrix status	0x00000000	R
0x400C21BC	TSWSTA	Tx switch matrix status	0x00000000	R

REGISTER DETAILS: PROGRAMMABLE SWITCHES

SWITCH MATRIX CONFIGURATION REGISTER

Address: 0x400C200C, Reset: 0x0000FFFF, Name: SWCON

Table 144. Bit Descriptions for SWCON

Bits	Bit Name	Settings	Description	Reset	Access
[31:20]	Reserved		Reserved.	0x0	R
19	T11CON		Control of T11 Switch.	0x0	R/W
		1	T11 closed.		
		0	T11 open.		
18	T10CON		Control of T10 Switch.	0x0	R/W
		1	T10 closed.		
		0	T10 open.		
17	T9CON		Control of T9 Switch.	0x0	R/W
		1	T9 closed.		
		0	T9 open.		
16	SWSOURCESEL		Switch Control Select. Select registers to control programmable switches.	0x0	R/W
		1	Switch control source. Switches controlled by DSWFULLCON, TSWFULLCON, PSWFULLCON, and NSWFULLCON registers.		
		0	Dx, Tx, Px, and Nx switches controlled as groups via SWCON register.		
[15:12]	TMUXCON		Control of Tx Switch Mux. Does not Include control of T11 to T9 switches.	0xF	R/W
		0000	All switches open.		
		0001	T1 closed, others open.		
		0010	T2 closed, others open.		
		0011	T3 closed, others open.		
		0100	T4 closed, others open.		
		0101	T5 closed, others open.		
		0110	Reserved.		
		0111	T7 closed, others open.		
		1000	TR1 closed, others open.		
		1001	All switches closed.		
		1010 to 1111	All switches open.		
[11:8]	NMUXCON		Control of Nx Switch Mux.	0xF	R/W
		0000	NL2 closed, others open.		
		0001	N1 closed, others open.		
		0010	N2 closed, others open.		
		0011	N3 closed, others open.		
		0100	N4 closed, others open.		
		0101	N5 closed, others open.		
		0110	N6 closed, others open.		
		0111	N7 closed, others open.		
		1000	N8 closed, others open.		
		1001	N9 closed, others open.		
		1010	NR1 closed, others open.		
		1011 to 1110	NL closed, others open.		
		1111	All switches open.		
[7:4]	PMUXCON		Control of Px Switch Mux.	0xF	R/W
		0000	PL2 closed, others open.		
		0001	PR0 closed, others open.		
		0010	P2 closed, others open.		
		0011	P3 closed, others open.		

REGISTER DETAILS: PROGRAMMABLE SWITCHES

Table 144. Bit Descriptions for SWCON (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		0100	P4 closed, others open.		
		0101	P5 closed, others open.		
		0110	P6 closed, others open.		
		0111	P7 closed, others open.		
		1000	P8 closed, others open.		
		1001	P9 closed, others open.		
		1010	P10 closed, others open.		
		1011	P11 closed, others open.		
		1100	P12 closed, others open.		
		1101 to 1110	PL closed, others open.		
		1111	All switches open.		
[3:0]	DMUXCON		Control of Dx Switch Mux.	0xF	R/W
		0000	All switches open.		
		0001	DR0 closed, others open.		
		0010	D2 closed, others open.		
		0011	D3 closed, others open.		
		0100	D4 closed, others open.		
		0101	D5 closed, others open.		
		0110	D6 closed, others open.		
		0111	D7 closed, others open.		
		1000	D8 closed, others open.		
		1001	All switches closed.		
		1010 to 1111	All switches open.		

DX SWITCH MATRIX FULL CONFIGURATION REGISTER

Address: 0x400C2150, Reset: 0x00000000, Name: DSWFULLCON

This register allows individual control of the Dx switches. The bit names are the same as the switch names shown in Figure 27. SWCON, Bit 16 must be set to 1 after writing to this register for the new switch settings to take effect.

Table 145. Bit Descriptions for DSWFULLCON

Bits	Bit Name	Settings	Description	Reset	Access
[31:8]	Reserved		Reserved.	0x0	R
7	D8		Control of D8 Switch. Connects the D node of the excitation amplifier to the SE1 pin.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
6	D7		Control of D7 Switch. Connects the D node of the excitation amplifier to the SE0 pin.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
5	D6		Control of D6 Switch. Connects the D node of the excitation amplifier to the CE1 pin.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
4	D5		Control of D5 Switch. Connects the D node of the excitation amplifier to the CE0 pin.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
3	D4		Control of D4 Switch. Connects the D node of the excitation amplifier to the AIN3/BUF_VREF1V8 pin.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
2	D3		Control of D3 Switch. Connects the D node of the excitation amplifier to the AIN2 pin.	0x0	R/W

REGISTER DETAILS: PROGRAMMABLE SWITCHES

Table 145. Bit Descriptions for DSWFULLCON (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		0	Switch open.		
		1	Switch closed.		
1	D2		Control of D2 Switch. Connects the D node of the excitation amplifier to the AIN1 pin.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
0	DR0		Control of DR0 Switch. Connects excitation amplifier D node to the RCAL0 pin.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		

NX SWITCH MATRIX FULL CONFIGURATION REGISTER

Address: 0x400C2154, Reset: 0x00000000, Name: NSWFULLCON

This register allows individual control of the Nx switches. The bit names are the same as the switch names shown in Figure 27. SWCON, Bit 16 must be set to 1 after writing to this register for the new switch settings to take effect.

Table 146. Bit Descriptions for NSWFULLCON

Bits	Bit Name	Settings	Description	Reset	Access
[31:12]	Reserved		Reserved.	0x0	R
11	NL2		Control of NL2 Switch. Set to close the NL2 switch, clear to open.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
10	NL		Control of NL Switch. Set to close the NL switch, clear to open. This switch shorts the N node of the excitation amplifier to the high-speed TIA inverting input.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
9	NR1		Control of NR1 Switch. Set to close the NR1 switch, clear to open. Connects the N node of the excitation amplifier to the RCAL1 pin.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
8	N9		Control of N9 Switch. Set to close the N9 switch, clear to open. Connects the N node of the excitation amplifier to the SE0 pin (bypasses R_{LOAD02}).	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
7	N8		Control of N8 Switch. Set to close the N8 switch, clear to open. Connects the N node of the excitation amplifier to R_{LOAD05}/T11.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
6	N7		Control of N7 Switch. Set to close the N7 switch, clear to open. Connects the N node of the excitation amplifier to the SE1 pin via R_{LOAD04}.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
5	N6		Control of N6 Switch. Set to close the N6 switch, clear to open. Connects the N node of the excitation amplifier to R_{LOAD03}/T10.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
4	N5		Control of N5 Switch. Set to close the N5 switch, clear to open. Connects excitation amplifier N node to the SE0 pin via R_{LOAD02}.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		

REGISTER DETAILS: PROGRAMMABLE SWITCHES

Table 146. Bit Descriptions for NSWFULLCON (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
3	N4		Control of N4 Switch. Set to close the N4 switch, clear to open. Connects the N node of the excitation amplifier to the AIN3/BUF_VREF1V8 pin.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
2	N3		Control of N3 Switch. Set to close the N3 switch, clear to open. Connects the N node of the excitation amplifier to the AIN2 pin.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
1	N2		Control of N2 Switch. Set to close the N2 switch, clear to open. Connects the N node of the excitation amplifier to the AIN1 pin.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
0	N1		Control of N1 Switch. Set to close the N1 switch, clear to open. Connects the N node of the excitation amplifier to the AIN0 pin.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		

PX SWITCH MATRIX FULL CONFIGURATION REGISTER

Address: 0x400C2158, Reset: 0x00000000, Name: PSWFULLCON

This register allows individual control of the Px switches. The bit names are the same as the switch names shown in Figure 27. SWCON, Bit 16 must be set to 1 after writing to this register for the new switch settings to take effect.

Table 147. Bit Descriptions for PSWFULLCON

Bits	Bit Name	Settings	Description	Reset	Access
[31:15]	Reserved		Reserved.	0x0	R
14	PL2		Control of PL2 Switch. Control of PL2 switch.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
13	PL		Control of PL Switch. Shorts the D node and P node of the excitation amplifiers together.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
12	Reserved		Reserved.	0x0	R/W
11	P12		Control of P12 Switch. Connects the P node of the excitation amplifier to the CE1 pin.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
10	P11		Control of P11 Switch. Connects the P node of the excitation amplifier to the CE0 pin.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
9	P10		Control of P10 Switch. Connects the P node of the excitation amplifier to the DE1 pin.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
8	P9		Control of P9 Switch. Connects the P node of the excitation amplifier to the SE1 pin.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
7	P8		Control of P8 Switch. Connects the P node of the excitation amplifier to the DE0 pin.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
6	P7		Control of P7 Switch. Connects the P node of the excitation amplifier to the SE0 pin.	0x0	R/W
		0	Switch open.		

REGISTER DETAILS: PROGRAMMABLE SWITCHES

Table 147. Bit Descriptions for PSWFULLCON (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		1	Switch closed.		
5	P6		Control of P6 Switch. Connects the P node of the excitation amplifier to the RE1 pin.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
4	P5		Control of P5 Switch. Connects the P node of the excitation amplifier to the RE0 pin.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
3	P4		Control of P4 Switch. Connects the P node of the excitation amplifier to the AIN3/BUF_VREF1V8 pin.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
2	P3		Control of P3 Switch. Connects the P node of the excitation amplifier to the AIN2 pin.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
1	P2		Control of P2 Switch. Connects the P node of the excitation amplifier to the AIN1 pin.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
0	PR0		Control of PR0 Switch. Connects the P node of the excitation amplifier to the RCAL0 pin.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		

TX SWITCH MATRIX FULL CONFIGURATION REGISTER

Address: 0x400C215C, Reset: 0x00000000, Name: TSWFULLCON

This register allows individual control of the Tx switches. The bit names are the same as the switch names shown in Figure 27. Note that SWCON, Bit 16 must be set to 1 after writing to this register for the new switch settings to take effect.

Table 148. Bit Descriptions for TSWFULLCON

Bits	Bit Name	Settings	Description	Reset	Access
[31:12]	Reserved		Reserved.	0x0	R
11	TR1		Control of TR1 Switch. Connects the RCAL1 pin to the high-speed TIA inverting input.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
10	T11		Control of T11 Switch. Connects the DE1 pin to the high-speed TIA inverting input.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
9	T10		Control of T10 Switch. Connects the DE0 pin to the high-speed TIA inverting input.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
8	T9		Control of T9 Switch. Used in conjunction with T10 and T11.	0x0	R/W
		0	Switch open. When open, the high-speed TIA inverting input can be DE0 or DE1 via T10 and T11.		
		1	Switch closed. Ensure T10 and T11 are open. The high-speed TIA inverting input is determined by T1, T2, T3, T4, T5, and T7.		
7	T8		Control of T8 Switch. Allows connection of the R_{CAL} path onto the DE1 input to calibrate the R_{TIA2_05} resistor.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
6	T7		Control of T7 Switch. Connects high-speed TIA inverting input to the SE1 pin via T9 and R_{LOAD04}.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		

REGISTER DETAILS: PROGRAMMABLE SWITCHES

Table 148. Bit Descriptions for TSWFULLCON (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
5	T6		Control of T6 Switch. Allows connection of the R_{CAL} path on to the DE0 input to calibrate the R_{LOAD03} and R_{TIA2_03} resistors.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
4	T5		Control of T5 Switch. Connects high-speed TIA inverting input to the SE0 pin via T9 and R_{LOAD02}.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
3	T4		Control of T4 Switch. Connects high-speed TIA inverting input to the AIN3/BUF_VREF1V8 pin via T9.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
2	T3		Control of T3 Switch. Connects high-speed TIA inverting input to the AIN2 pin via T9.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
1	T2		Control of T2 Switch. Connects high-speed TIA inverting input to the AIN1 pin via T9.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		
0	T1		Control of T1 Switch. Connects high-speed TIA inverting input to the AIN0 pin via T9.	0x0	R/W
		0	Switch open.		
		1	Switch closed.		

DX SWITCH MATRIX STATUS REGISTER

Address: 0x400C21B0, Reset: 0x00000000, Name: DSWSTA

This gives the status of the Dx switches shown in Figure 27.

Table 149. Bit Descriptions for DSWSTA

Bits	Bit Name	Settings	Description	Reset	Access
[31:8]	Reserved		Reserved.	0x0	R
7	D8STA		Status of D8 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
6	D7STA		Status of D7 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
5	D6STA		Status of D6 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
4	D5STA		Status of D5 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
3	D4STA		Status of D4 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
2	D3STA		Status of D3 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
1	D2STA		Status of D2 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		

REGISTER DETAILS: PROGRAMMABLE SWITCHES

Table 149. Bit Descriptions for DSWSTA (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
0	D1STA		Status of DR0 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		

PX SWITCH MATRIX STATUS REGISTER

Address: 0x400C21B4, Reset: 0x00000000, Name: PSWSTA

This register gives the status of the Px switches shown in Figure 27.

Table 150. Bit Descriptions for PSWSTA

Bits	Bit Name	Settings	Description	Reset	Access
[31:15]	Reserved		Reserved.	0x0	R
14	PL2STA		PL2 Switch Control.	0x0	R
		0	Switch open.		
		1	Switch closed.		
13	PLSTA		PL Switch Control.	0x0	R
		0	Switch open.		
		1	Switch closed.		
12	P13STA		Status of P13 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
11	P12STA		Status of P12 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
10	P11STA		Status of P11 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
9	P10STA		Status of P10 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
8	P9STA		Status of P9 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
7	P8STA		Status of P8 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
6	P7STA		Status of P7 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
5	P6STA		Status of P6 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
4	P5STA		Status of P5 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
3	P4STA		Status of P4 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
2	P3STA		Status of P3 Switch.	0x0	R

REGISTER DETAILS: PROGRAMMABLE SWITCHES

Table 150. Bit Descriptions for PSWSTA (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		0	Switch open.		
		1	Switch closed.		
1	P2STA		Status of P2 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
0	PR0STA		PR0 Switch Control.	0x0	R
		0	Switch open.		
		1	Switch closed.		

NX SWITCH MATRIX STATUS REGISTER

Address: 0x400C21B8, Reset: 0x00000000, Name: NSWSTA

This register gives the status of the Nx switches shown in Figure 27.

Table 151. Bit Descriptions for NSWSTA

Bits	Bit Name	Settings	Description	Reset	Access
[31:12]	Reserved		Reserved.	0x0	R
11	NL2STA		Status of NL2 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
10	NLSTA		Status of NL Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
9	NR1STA		Status of NR1 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
8	N9STA		Status of N9 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
7	N8STA		Status of N8 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
6	N7STA		Status of N7 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
5	N6STA		Status of N6 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
4	N5STA		Status of N5 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
3	N4STA		Status of N4 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
2	N3STA		Status of N3 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
1	N2STA		Status of N2 Switch.	0x0	R
		0	Switch open.		

REGISTER DETAILS: PROGRAMMABLE SWITCHES

Table 151. Bit Descriptions for NSWSTA (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		1	Switch closed.		
0	N1STA		Status of N1 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		

TX SWITCH MATRIX STATUS REGISTER

Address: 0x400C21BC, Reset: 0x00000000, Name: TSWSTA

This register gives the status of the Tx switches shown in Figure 27.

Table 152. Bit Descriptions for TSWSTA

Bits	Bit Name	Settings	Description	Reset	Access
[31:12]	Reserved		Reserved.	0x0	R
11	TR1STA		Status of TR1 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
10	T11STA		Status of T11 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
9	T10STA		Status of T10 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
8	T9STA		Status of T9 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
7	T8STA		Status of T8 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
6	T7STA		Status of T7 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
5	T6STA		Status of T6 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
4	T5STA		Status of T5 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
3	T4STA		Status of T4 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
2	T3STA		Status of T3 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
1	T2STA		Status of T2 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		
0	T1STA		Status of T1 Switch.	0x0	R
		0	Switch open.		
		1	Switch closed.		

SEQUENCER

SEQUENCER FEATURES

The features of the ADuCM355 sequencer are as follows:

- Programmable for cycle accurate applications.
- Four separate command sequences.
- 6 kB SRAM to store sequences.
- FIFO to store measurement results.
- Control via the wake-up timer or direct register write.
- Various interrupts from user maskable sources.

SEQUENCER OVERVIEW

The role of the sequencer is to allow offloading of the low level AFE operations from the digital die and to provide cyclic accurate control over the analog DSP blocks. The sequencer handles timing critical operations without being subject to system load.

Four command sequences are supported by hardware on the ADuCM355. These sequences can be stored in the SRAM to switch between different measurement procedures. Only one sequence can be executed by the sequencer at a time. However, the user can configure which sequences the sequencer executes and the order in which they are executed.

The sequencer reads commands from the sequence that is stored in the command memory and, depending on the command, either waits a certain amount of time or writes a value to an MMR. The execution is sequential with no branching. The sequencer cannot read MMR values or signals from the analog or DSP blocks.

To enable the sequencer, set the SEQEN bit in the SEQCON register. To disable the sequencer, write 0 to this bit.

The rate at which the sequencer commands are executed is provided in the SEQWRTMR bits in the SEQCON register. When a write command is executed by the sequencer, the sequencer performs the MMR write and then waits SEQWRTMR clock cycles before fetching the next command in the sequence. The effect is the same as a write command followed by a wait command. The main purpose of this setup is to reduce code size when generating arbitrary waveforms. The SEQWRTMR bits do not have any effect following a wait or timeout command.

In addition to a single write command being followed by a wait command, multiple write commands can be executed in succession followed by a wait command. Any configuration can be rapidly set up by the sequencer, regardless of the number of register writes followed by a precisely executed delay.

The sequencer can also be paused by setting the SEQHALT bit in the SEQCON register. This option applies to each function, including FIFO operations, internal timers, and waveform generation. Reads from the MMRs are allowed when the sequencer is paused. This mode is intended for debugging during software development.

The number of commands executed by the sequencer can be read from the SEQCNT register. Each time a command is read from command memory and executed, the counter increments by 1. To reset the counter, perform a write to the SEQCNT register.

The sequencer calculates the cyclic redundancy check (CRC) of all commands it executes. The algorithm used is the CRC-8, using the $x^8 + x^2 + x + 1$ polynomial. The CRC-8 algorithm performs on 32-bit input data (sequencer instructions). Each 32-bit input is processed in one clock cycle and the result is available immediately for reading by the host controller. The CRC value can be read from the SEQCRC register. To reset this register by the same mechanism as the command count, write to the SEQCNT register. The SEQCRC register resets to a seed value of 0x01. SEQCRC is a read only register.

SEQUENCER COMMANDS

Two types of commands can be executed by the sequencer: write commands and timer commands, which include wait commands and timeout commands.

Write Command

Use a write instruction to write data to a register. The register address must lie between 0x400C0000 and 0x400C21FC. Figure 28 shows the format of the instruction. The MSB is equal to 1, which indicates a write command.

In Figure 28, ADDR is the write address and data is the write data to be written to the MMR. All write instructions finish within one cycle.

SEQUENCER

The address field is seven bits wide to allow access to registers from Address 0x0 to Address 0x1FC in the AFE register block. All MMR accesses are 32 bits. Byte and half word accesses are forbidden. All accesses are write only. There is a direct mapping between the address field and the MMR address. In Figure 28, ADDR corresponds to Bits[8:2] of the 16-bit MMR address.

For example, when writing to the WGCON register directly through the die to die interface, Address 0x400C2014 is used. To write to the same register using the sequencer, the address field must be 0b0000101 (Bits[8:2] of the address used by the external controller).

The data field is 24 bits wide and only allows writing to the MMR bits, Bits[23:0]. It is not possible to write to the full 32 bits of the MMRs via the sequencer. However, Bits[31:24] are not used by any of the MMRs. Therefore, all assigned MMR bits can be written by the sequencer.

Timer Command

There are two timer commands in the sequencer, each with a separate hardware counter.

The wait command introduces wait states in the sequencer execution. When the programmed counter reaches 0, the execution resumes by reading the next command from command memory.

Figure 29 shows the format of the timer command, and Figure 30 shows the format of the wait command.

The timeout command starts a counter that operates independently of the sequencer flow. When the timer elapses, one of two interrupts is generated: a sequencer timeout error interrupt (INTSEL17) or a sequencer timeout finished interrupt (INTSEL16). Both interrupts are configured in the INTCSELx registers. The sequence timeout finished interrupt is asserted at the end of the timeout period. The sequence timeout error interrupt is asserted if the sequencer does not reach the end of execution at the end of the timeout period. These interrupts are cleared by writing to the corresponding bits in the INTCCLR register. The current value of the counter can be read by the host controller at any time through the SEQTIMEOUT register.

The timeout counter is not reset when the sequencer execution is stopped as a result of a sequencer write command. However, the counter resets if the host controller writes a 0 to the SEQEN bit in the SEQCON register. This reset applies to situations when the host must abort the sequence.

The time unit for both timer commands is one ACLK period. For a clock frequency of 16 MHz, the timer resolution is 62.5 ns and the maximum timeout is 67.1 sec. These values are still true when the SEQWRTMR bits in the SEQCON register are nonzero.

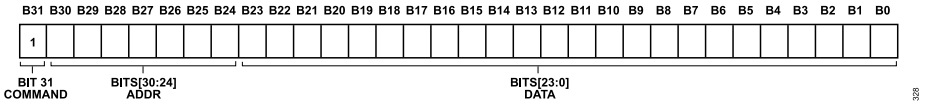

Figure 28. Sequencer Write Command

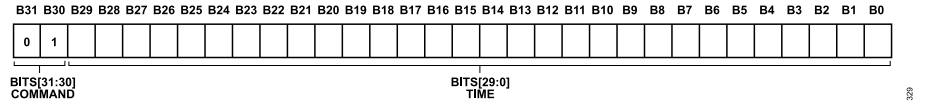

Figure 29. Sequencer Timer Command

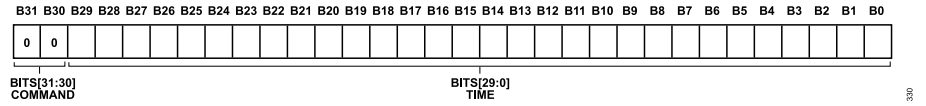

Figure 30. Sequencer Wait Command

SEQUENCER OPERATION

Figure 31 shows the typical steps required to set up the sequencer to take measurements. When the device is booted up, take the following steps to configure the sequencer, command memory, and data FIFO:

1. Configure the command memory.
2. Load the sequences into the SRAM.
3. Set the Sequence 0 (SEQ0) to Sequence 3 (SEQ3) information sequences.
4. Configure the data FIFO.

SEQUENCER

5. Configure the sleep and wake-up timer.
6. Configure the interrupts.
7. Configure the sleep and wake-up method.

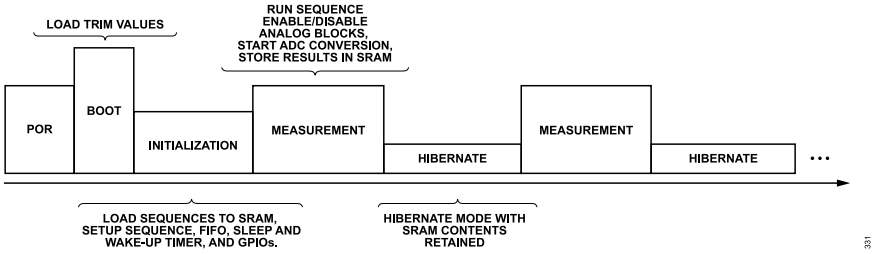

Figure 31. Run Sequence

Command Memory

The command memory stores the sequence commands and provides a link between the external microcontroller and the sequencer. The command memory can be configured to use the 2 kB, 4 kB, and 6 kB SRAM memory sizes, which are selected using the CMDDATACON register, Bits[2:0].

The large amount of memory available for the command memory facilitates the creation of larger, more complex sequences.

Determine the number of commands in a sequence by reading the SEQxINFO register, Bits[26:16].

The command memory is unidirectional. The host microcontroller specifies the destination address of the command by writing to the CMDFIFOWADDR register and writes the command contents to the CMDFIFOWRITE register. The sequencer reads the commands from memory for execution.

There are a number of interrupts associated with the command FIFO, including the FIFO threshold interrupt, the FIFO empty interrupt, and the FIFO full interrupt. Refer to the AFE Interrupts section for more information.

Loading Sequences

The sequence commands are written to the SRAM by writing to two registers. The address in SRAM for the command is written to the CMDFIFOWADDR register. The command content is written to the CMDFIFOWRITE register. When all commands are written to the SRAM, write to the SEQxINFO registers to set the SEQ0 to SEQ3 information sequences.

Each information sequence from SEQ0 to SEQ3 requires a start address in SRAM and a total number of commands for that sequence. The number of commands is written to the SEQxINFO register, Bits[26:16]. The start address is written to the SEQxINFO register, Bits[10:0]. Ensure that there is no overlap between the four sequences. There is no hardware mechanism in place to warn the user of overlapping sequences.

The interrupt sources associated with the sequencer include the following:

- Sequence timeout error.
- Sequencer timeout command finished.
- End of sequence interrupt. For this interrupt to be asserted, the SEQCON register, Bit 0, must be cleared at the end of the sequencer command.

Data FIFO

The data FIFO provides a buffer for the output of the analog and DSP blocks before the FIFO is read by the external controller.

The memory available for the data FIFO can be selected in the DATA_MEM_SEL bits in the CMDDATACON register. The available memory options are 2 kB, 4 kB, and 6 kB. The data FIFO and command memory share the same block of 6 kB SRAM. Therefore, ensure that there is no overlap between the command memory and the data FIFO.

The data FIFO can be configured in FIFO mode or stream mode via the CMDDATACON register, Bits[11:9]. In stream mode, when the FIFO is full, old data is discarded to make room for new data. In FIFO mode, when the FIFO is full, new data is discarded. Do not let the FIFO overflow when in FIFO mode. If there is overflow, all new data are lost.

SEQUENCER

The data FIFO is always unidirectional. A selectable source in the AFE block writes data and the external microcontroller reads data from the DATAFIFORD register (see Table 220).

Select the data source for the data FIFO in the FIFOCON register, Bits[15:13] (see Table 219). The available source options include ADC data, DFT result, sinc2 filter result, and statistic block mean result.

The interrupt flags associated with the data FIFO include empty, full, overflow, underflow, and threshold. These interrupts are user readable using the INTCFLAGx registers (see the AFE Interrupts section for more details). Each flag has an associated maskable interrupt.

The overflow and underflow flags only activate for one clock period.

The data FIFO is enabled by writing a 1 to the FIFOCON register, Bit 11. The data FIFO threshold value is set by writing to the DATAFIFOTHRES register. At any time, the host microcontroller can read the number of words in the data FIFO by reading the FIFOCNTSTA register, Bits[26:16].

Reading data from the data FIFO when the FIFO is empty returns 0x00000000. The underflow flag, the FLAG27 bit, in the INTCFLAGx register is also asserted.

Data FIFO Word Format

The format of data FIFO words is shown in Figure 32. Each word in the data FIFO is 32 bits. The seven MSBs are the ECC required for functional safety applications. Bits[24:23] of the data FIFO word form the sequence identification (ID) and indicate which sequence, from SEQ0 to SEQ3, the result came from.

Bits[22:16] of the data FIFO word contain the channel ID and indicate the source for the data (see Table 153).

The 16 LSBs of the data FIFO word are the actual data (see Figure 32).

When the data source is the DFT result, the data is 18 bits wide and is in twos complement format. The format is shown in Figure 33. The channel ID is five bits wide, with 5'b11111 indicating the DFT results.

Sequencer and the Sleep and Wake-Up Timer

See the Sleep and Wake-Up Timer section for more information.

Sequencer Conflicts

If a conflict between sequences arises, for example, when SEQ0 is running and the SEQ1 request arrives, SEQ1 is ignored and SEQ0 completes. An interrupt is generated to indicate that the Sequence 1 is ignored.

Reading back registers does not cause resource conflicts. Writes to the MMRs by the processor are not allowed when the sequencer is enabled. If conflicts arise, the sequencer has the priority. If the sequencer and the processor write at the same time, the host controller is ignored. There is no error report for this conflict. Do not write to a register when the sequencer is running. However, there are registers that are exceptions and can be written to freely without any conflict. The SEQCON register allows ending a sequence execution (SEQEN bit) and halting a sequence (SEQHALT bit).

Table 153. Channel ID Description

Data FIFO Word, Bits[22:16]	Description
11111XX[1]	DFT result
11110XX	Mean from statistics block
1XXXXXX	Sinc2 filter result, XXXXXX is the ADC multiplexer positive setting (ADCCON register, Bits[5:0])
0XXXXXX	Sinc3 filter result, XXXXXX is the ADC multiplexer positive setting (ADCCON register, Bits[5:0])

[1] X means don't care.

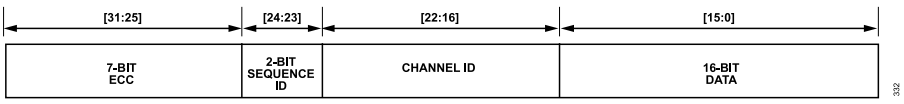

Figure 32. Data FIFO Word Format

SEQUENCER

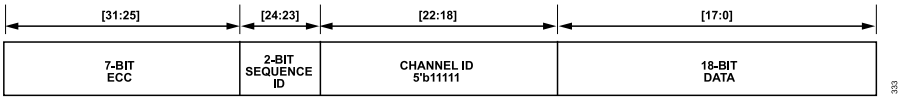

Figure 33. Data FIFO DFT Word Format

SEQUENCER AND FIFO REGISTERS

Table 154. Sequence and FIFO Registers Summary

Address	Name	Description	Reset	Access
0x400C2004	SEQCON	Sequencer configuration register	0x00000002	R/W
0x400C2008	FIFOCON	FIFO configuration register	0x00001010	R/W
0x400C2060	SEQCRC	Sequencer CRC value register	0x00000001	R
0x400C2064	SEQCNT	Sequencer command count register	0x00000000	R/W
0x400C2068	SEQTIMEOUT	Sequencer timeout counter register	0x00000000	R
0x400C206C	DATAFIFORD	Data FIFO read register	0x00000000	R
0x400C2070	CMDFIFOWRITE	Command FIFO write register	0x00000000	W
0x400C2118	SEQSLPLOCK	Sequencer sleep control lock register	0x00000000	R/W
0x400C211C	SEQTRGSLP	Sequencer trigger sleep register	0x00000000	R/W
0x400C21CC	SEQ0INFO	Sequence 0 information register	0x00000000	R/W
0x400C21D0	SEQ2INFO	Sequence 2 information register	0x00000000	R/W
0x400C21D4	CMDFIFOWADDR	Command FIFO write address register	0x00000000	R/W
0x400C21D8	CMDDATACON	Command data control register	0x00000410	R/W
0x400C21E0	DATAFIFOTHRES	Data FIFO threshold register	0x00000000	R/W
0x400C21E4	SEQ3INFO	Sequence 3 information register	0x00000000	R/W
0x400C21E8	SEQ1INFO	Sequence 1 information register	0x00000000	R/W
0x400C2200	FIFOCNTSTA	Command and data FIFO internal data count register	0x00000000	R
0x400C0430	TRIGSEQ	Trigger sequence register	0x0000	R/WS

Sequencer Configuration Register

Address: 0x400C2004, Reset: 0x00000002, Name: SEQCON

Table 155. Bit Descriptions for SEQCON Register

Bits	Bit Name	Settings	Description	Reset	Access
[31:16]	Reserved		Reserved.	0x0	R
[15:8]	SEQWRTMR		Sequencer Write Commands Timer. These bits act as a clock divider affecting only the write commands, not the wait commands. This divider is useful to reduce the code size when generating arbitrary waveforms. The clock source for the timer is ACLK.	0x0	R/W
[7:5]	Reserved		Reserved.	0x0	R
4	SEQHALT		Halt Sequence Debugging Feature. This bit provides a way to halt the AFE interface.	0x0	R/W
		0	Normal execution.		
		1	Execution halted.		
[3:2]	Reserved		Reserved.	0x0	R
1	SEQHALTFIFOEMPTY		Halt Sequencer. This bit controls whether the sequencer stops when attempting to read when the command FIFO is empty (in an underflow condition).	0x1	R/W
		1	Sequencer stops if command FIFO is empty and sequencer attempts to read (in an underflow condition).		
		0	Sequencer continues to attempt to read, even if the FIFO is empty.		
0	SEQEN		Enable Sequencer. When this bit is set to 1, the sequencer reads from the command FIFO and executes the commands.	0x0	R/W
		0	Sequencer disabled (default).		
		1	Sequencer enabled.		

SEQUENCER

FIFO Configuration Register

Address: 0x400C2008, Reset: 0x00001010, Name: FIFOCON

Table 156. Bit Descriptions for FIFOCON Register

Bits	Bit Name	Settings	Description	Reset	Access
[31:16]	RESERVED		Reserved.	0x0	R
[15:13]	DATAFIFOSRCSEL		Data FIFO Source Select.	0x0	R/W
		000, 001, 110, or 111	ADC data. The ADC data is the output of the sinc3 filter.		
		010	DFT data. The real part is 18 bits, and the imaginary part is 18 bits. The lowest two bits are fractional because the ADC is 16 bits.		
		011	Sinc2 filter output, data is 16 bits.		
		100	Statistic variance output.		
		101	Mean result, mean is 16 bits of data.		
12	Reserved		Reserved.	0x1	R/W
11	DATAFIFOEN		Data FIFO Enable.	0x0	R/W
		0	FIFO is reset, no data transfers can take place. This setting sets the read and write pointers to the default values (FIFO empty). The status indicates that the FIFO is empty.		
		1	Normal operation, the FIFO is not reset.		
[10:0]	Reserved		Reserved.	0x0	R/W

Sequencer CRC Value Register

Address: 0x400C2060, Reset: 0x00000001, Name: SEQCRC

The SEQCRC register forms the checksum value calculated from all the commands executed by the sequencer.

Table 157. Bit Descriptions for SEQCRC Register

Bits	Bit Name	Description	Reset	Access
[31:8]	Reserved	Reserved.	0x0	R
[7:0]	CRC	Sequencer Command CRC Value. The algorithm used is CRC-8.	0x1	R

Sequencer Command Count Register

Address: 0x400C2064, Reset: 0x00000000, Name: SEQCNT

The SEQCNT register forms the command count, which is incremented by 1 each time the sequencer executes a command. This register is not key protected.

Table 158. Bit Descriptions for SEQCNT Register

Bits	Bit Name	Description	Reset	Access
[31:16]	Reserved	Reserved.	0x0	R
[15:0]	Count	Sequencer Command Count. This count is incremented by 1 each time the sequencer executes a command. To reset to 0 or clear the SEQCRC register, write 1 to this register.	0x0	R/W1

Sequencer Timeout Counter Register

Address: 0x400C2068, Reset: 0x00000000, Name: SEQTIMEOUT

Table 159. Bit Descriptions for SEQTIMEOUT Register

Bits	Bit Name	Description	Reset	Access
[31:30]	Reserved	Reserved.	0x0	R
[29:0]	Timeout	Sequencer Timeout Counter Current Value.	0x0	R

SEQUENCER

Data FIFO Read Register

Address: 0x400C206C, Reset: 0x00000000, Name: DATAFIFORD

Table 160. Bit Descriptions for DATAFIFORD Register

Bits	Bit Name	Description	Reset	Access
[31:0]	DATAFIFOOUT	Data FIFO Read. When the data FIFO is empty, a read of this register returns 0x00000000. See Figure 32 and Figure 33 for word format details.	0x0	R

Command FIFO Write Register

Address: 0x400C2070, Reset: 0x00000000, Name: CMDFIFOWRITE

Table 161. Bit Descriptions for CMDFIFOWRITE Register

Bits	Bit Name	Description	Reset	Access
[31:0]	CMDFIFOIN	Command FIFO Write. If the command FIFO is written when full, the write is ignored and no current commands are affected.	0x0	W

Sequencer Sleep Control Lock Register

Address: 0x400C2118, Reset: 0x00000000, Name: SEQSLPLOCK

The SEQSLPLOCK register protects the SEQTRGSLP register.

Table 162. Bit Descriptions for SEQSLPLOCK Register

Bits	Bit Name	Settings	Description	Reset	Access
[31:20]	Reserved		Reserved.	0x0	R
[19:0]	SEQ_SLP_PW		SEQTRGSLP Register Password. These bits prevent the sequencer from accidentally triggering a sleep state.	0x0	R/W
		0x400C	Write any value other than 0xA47E5 to lock the SEQTRGSLP register.		
		0xA47E5	Write this value to this register to unlock the SEQTRGSLP register.		

Sequencer Trigger Sleep Register

Address: 0x400C211C, Reset: 0x00000000, Name: SEQTRGSLP

The SEQTRGSLP register is protected by the SEQSLPLOCK register.

Table 163. Bit Descriptions for SEQTRGSLP Register

Bits	Bit Name	Description	Reset	Access
[31:1]	Reserved	Reserved.	0x0	R
0	TRGSLP	Trigger Sleep by Sequencer. Write to the SEQSLPLOCK register first. Put this command at the end of a sequence. Set this command to 1 if entering sleep at the end of a sequence.	0x0	R/W

Sequence 0 Information Register

Address: 0x400C21CC, Reset: 0x00000000, Name: SEQ0INFO

Table 164. Bit Descriptions for SEQ0INFO Register

Bits	Bit Name	Description	Reset	Access
[31:27]	Reserved	Reserved.	0x0	R
[26:16]	SEQ0INSTNUM	SEQ0 Instruction Number.	0x0	R/W
[15:11]	Reserved	Reserved.	0x0	R
[10:0]	SEQ0STARTADDR	SEQ0 Start Address.	0x0	R/W

SEQUENCER

Sequence 2 Information Register

Address: 0x400C21D0, Reset: 0x00000000, Name: SEQ2INFO

Table 165. Bit Descriptions for SEQ2INFO Register

Bits	Bit Name	Description	Reset	Access
[31:27]	Reserved	Reserved.	0x0	R
[26:16]	SEQ2INSTNUM	SEQ2 Instruction Number.	0x0	R/W
[15:11]	Reserved	Reserved.	0x0	R
[10:0]	SEQ2STARTADDR	SEQ2 Start Address.	0x0	R/W

Command FIFO Write Address Register

Address: 0x400C21D4, Reset: 0x00000000, Name: CMDFIFOWADDR

Table 166. Bit Descriptions for CMDFIFOWADDR Register

Bits	Bit Name	Description	Reset	Access
[31:11]	Reserved	Reserved.	0x0	R
[10:0]	WADDR	Write Address. These bits are the address in SRAM in which the command is stored.	0x0	R/W

Command Data Control Register

Address: 0x400C21D8, Reset: 0x00000410, Name: CMDDATACON

Table 167. Bit Descriptions for CMDDATACON Register

Bits	Bit Name	Settings	Description	Reset	Access
[31:12]	Reserved		Reserved.	0x0	R
[11:9]	DATAMEMMDE		Data FIFO Mode Select.	0x2	R/W
		10	FIFO mode.		
		11	Stream mode.		
[8:6]	DATA_MEM_SEL		Data FIFO Size Select.	0x0	R/W
		000	Reserved.		
		001	2 kB SRAM.		
		010	4 kB SRAM.		
		011	6 kB SRAM.		
[5:3]	CMDMEMMDE		Command FIFO Mode.	0x2	R/W
		01	Memory mode.		
		10	Reserved.		
		11	Reserved.		
[2:0]	CMD_MEM_SEL		Command Memory Select.	0x0	R/W
		0x0	Reserved.		
		0x1	2 kB SRAM.		
		0x2	4 kB SRAM.		
		0x3	6 kB SRAM.		

Data FIFO Threshold Register

Address: 0x400C21E0, Reset: 0x00000000, Name: DATAFIFOTHRES

Table 168. Bit Descriptions for DATAFIFOTHRES Register

Bits	Bit Name	Description	Reset	Access
[31:27]	Reserved	Reserved.	0x0	R
[26:16]	HIGHTHRES	High Threshold.	0x0	R/W
[15:0]	Reserved	Reserved.	0x0	R

SEQUENCER

Sequence 3 Information Register

Address: 0x400C21E4, Reset: 0x00000000, Name: SEQ3INFO

Table 169. Bit Descriptions for SEQ3INFO Register

Bits	Bit Name	Description	Reset	Access
[31:27]	Reserved	Reserved.	0x0	R
[26:16]	INSTNUM	SEQ3 Instruction Number.	0x0	R/W
[15:11]	Reserved	Reserved.	0x0	R
[10:0]	STARTADDR	SEQ3 Start Address.	0x0	R/W

Sequence 1 Information Register

Address: 0x400C21E8, Reset: 0x00000000, Name: SEQ1INFO

Table 170. Bit Descriptions for SEQ1INFO Register

Bits	Bit Name	Description	Reset	Access
[31:27]	Reserved	Reserved.	0x0	R
[26:16]	SEQ1INSTNUM	SEQ1 Instruction Number.	0x0	R/W
[15:11]	Reserved	Reserved.	0x0	R
[10:0]	SEQ1STARTADDR	SEQ1 Start Address.	0x0	R/W

Command and Data FIFO Internal Data Count Register

Address: 0x400C2200, Reset: 0x00000000, Name: FIFOCNTSTA

Table 171. Bit Descriptions for FIFOCNTSTA Register

Bits	Bit Name	Description	Reset	Access
[31:27]	Reserved	Reserved.	0x0	R
[26:16]	DATAFIFOCNTSTA[10:0]	Current Number of Words in the Data FIFO.	0x0	R
[15:0]	Reserved	Reserved.	0x0	R

Trigger Sequence Register

Address: 0x400C0430, Reset: 0x0000, Name: TRIGSEQ

Table 172. Bit Descriptions for TRIGSEQ Register

Bits	Bit Name	Description	Reset	Access
[15:4]	Reserved	Reserved.	0x0	R
3	TRIG3	Trigger Sequence 3.	0x0	R/W
2	TRIG2	Trigger Sequence 2.	0x0	R/W
1	TRIG1	Trigger Sequence 1.	0x0	R/W
0	TRIG0	Trigger Sequence 0.	0x0	R/WS

AFE INTERRUPTS

There are interrupt options available on the ADuCM355 analog front end that can be configured to toggle the internal GPIO pin on the digital die. The GPIO pin is connected internally and is not bonded out of the LGA package.

INTERRUPT CONTROLLER INTERRUPTS

The interrupt controller is divided into two blocks. Each block consists of an INTCSELx register and an INTCFLAGx register. Only INTCSEL0 interrupts are connected to the digital die by the internal GPIO pin. The INTCSEL1 interrupts are only used for polling. The INTCPOL and INTCCLR registers are common to both blocks. When an interrupt is enabled in the INTCSELx register, the corresponding bit in the INTCFLAGx register is set. The available interrupt sources are shown in Table 173.

CONFIGURING THE INTERRUPTS

The first step to configure the INTC interrupts is to configure the digital die GPIO pin that connects internally to the AFE interrupt controller output. To configure the GPIO pin, take the following steps:

1. Configure the internal GPIO2 Pin 1 as a GPIO as follows:

```
DioCfgPin(pADI_GPIO2, PIN1, 0);
```

2. Configure the internal P2.1 pin as an input as follows:

```
DioIenPin(pADI_GPIO2, PIN1, 1); /* Enable P2.1 input path. */
```

3. Enable External Interrupt 3 and configure for falling edge as follows:

```
EiCfg(EXTINT3, INT_EN, INT_FALL); /* Falling edge. */
```

4. Enable the interrupt in the NVIC as follows:

```
NVIC_EnableIRQ(AFE_EVT3_IRQn);
```

When these steps are complete, the digital die is configured for INTC interrupts. To configure the AFE INTC interrupts, first write to the INCT0POL to configure the polarity. To enable the required interrupt, write to the INTCSEL0 register. To clear an interrupt source, write to the corresponding bit in the INTCCLR register.

CUSTOM INTERRUPTS

Four custom interrupt sources are selectable by the user in the INTCSELx register, Bits[12:9]. For these custom interrupts to generate an interrupt event, write to the corresponding bit in the AFEGENINTSTA register. It is only possible to write to this register via the sequencer.

Table 173. Interrupt Sources Summary

INTCFLAGx Register Flag Name	Interrupt Source Description
FLAG0	ADC result IRQ status.
FLAG1	DFT result IRQ status.
FLAG2	Sinc2 filter result ready IRQ status.
FLAG3	Temperature result IRQ status.
FLAG4	ADC minimum fail IRQ status.
FLAG5	ADC maximum fail IRQ status.
FLAG6	ADC delta fail IRQ status.
FLAG7	Mean IRQ status.
FLAG13	Bootload done IRQ status.
FLAG15	End of sequence IRQ status.
FLAG16	Sequencer timeout finished IRQ status (see the Timer Command section).
FLAG17	Sequencer timeout command error IRQ status (see the Timer Command section).
FLAG23	Data FIFO full IRQ status.
FLAG24	Data FIFO empty IRQ status.
FLAG25	Data FIFO threshold IRQ status, threshold value set in DATAFIFOTHRES register.

AFE INTERRUPTS

Table 173. Interrupt Sources Summary (Continued)

INTCFLAGx Register Flag Name	Interrupt Source Description
FLAG26	Data FIFO overflow IRQ status.
FLAG27	Data FIFO underflow IRQ status.
FLAG29	Outlier IRQ status, detects when an outlier is detected.
FLAG31	Attempt to break IRQ status. This interrupt is set if a Sequence B request occurs when Sequence A is running. This interrupt indicates that Sequence B is ignored.

INTERRUPT REGISTERS

Table 174. Interrupt Registers Summary

Address	Name	Description	Reset	Access
0x400C3000	INTCPOL	Interrupt polarity register	0x00000000	R/W
0x400C3004	INTCCLR	Interrupt clear register	0x00000000	W
0x400C3008	INTCSEL0	Interrupt controller select register	0x00002000	R/W
0x400C300C	INTCSEL1	Interrupt controller select register	0x00002000	R/W
0x400C3010	INTCFLAG0	Interrupt controller flag register	0x00000000	R
0x400C3014	INTCFLAG1	Interrupt controller flag register	0x00000000	R
0x400C209C	AFEGENINTSTA	Analog generation interrupt register	0x00000010	R/W1C

Interrupt Polarity Register

Address: 0x400C3000, Reset: 0x00000000, Name: INTCPOL

Table 175. Bit Descriptions for INTCPOL Register

Bits	Bit Name	Settings	Description	Reset	Access
[31:1]	Reserved		Reserved.	0x0	R
0	INTPOL		Interrupt Polarity.	0x0	R/W
		0	Output negative edge interrupt.		
		1	Output positive edge interrupt.		

Interrupt Clear Register

Address: 0x400C3004, Reset: 0x00000000, Name: INTCCLR

Table 176. Bit Descriptions for INTCCLR Register

Bits	Bit Name	Description	Reset	Access
31	INTCLR31	Attempt to Break Interrupt (IRQ). Write 1 to clear.	0x0	W
30	Reserved	Reserved.	0x0	W
29	INTCLR29	Outlier IRQ. Write 1 to clear.	0x0	W
28	Reserved	Reserved.	0x0	W
27	INTCLR27	Data FIFO Underflow IRQ. Write 1 to clear.	0x0	W
26	INTCLR26	Data FIFO Overflow IRQ. Write 1 to clear.	0x0	W
25	INTCLR25	Data FIFO Threshold IRQ. Write 1 to clear.	0x0	W
24	INTCLR24	Data FIFO Empty IRQ. Write 1 to clear.	0x0	W
23	INTCLR23	Data FIFO Full IRQ. Write 1 to clear.	0x0	W
22	Reserved	Reserved.	0x0	W
17	INTCLR17	Sequencer Timeout Error IRQ. Write 1 to clear.	0x0	W
16	INTCLR16	Sequencer Timeout Finished IRQ. Write 1 to clear.	0x0	W
15	INTCLR15	End of Sequence IRQ. Write 1 to clear.	0x0	W
14	Reserved	Reserved.	0x0	W
13	INTCLR13	Boot Load Done IRQ. Write 1 to clear.	0x0	W
12	INTCLR12	Custom Interrupt 3. Write 1 to clear.	Not applicable	Not applicable

AFE INTERRUPTS

Table 176. Bit Descriptions for INTCCLR Register (Continued)

Bits	Bit Name	Description	Reset	Access
11	INTCLR11	Custom Interrupt 2. Write 1 to clear.	Not applicable	Not applicable
10	INTCLR10	Custom Interrupt 1. Write 1 to clear.	Not applicable	Not applicable
9	INTCLR9	Custom Interrupt 0. Write 1 to clear.	Not applicable	Not applicable
8	Reserved	Reserved.	0x0	W
7	INTCLR7	Mean IRQ. Write 1 to clear.	0x0	W
6	INTCLR6	ADC Delta Fail IRQ. Write 1 to clear.	0x0	W
5	INTCLR5	ADC Maximum Fail IRQ. Write 1 to clear.	0x0	W
4	INTCLR4	ADC Minimum Fail IRQ. Write 1 to clear.	0x0	W
3	INTCLR3	Temperature Result IRQ. Write 1 to clear.	0x0	W
2	INTCLR2	Sinc2 Filter Result Ready IRQ. Write 1 to clear.	0x0	W
1	INTCLR1	DFT Result IRQ. Write 1 to clear.	0x0	W
0	INTCLR0	ADC Result IRQ. Write 1 to clear.	0x0	W

Interrupt Controller Select Registers

Address: 0x400C3008, Reset: 0x00002000, Name: INTCSEL0

Address: 0x400C300C, Reset: 0x00002000, Name: INTCSEL1

Table 177. Bit Descriptions for INTCSEL0 and INTCSEL1 Registers

Bits	Bit Name	Settings	Description	Reset	Access
31	INTSEL31		Attempt to Break IRQ Enable.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
30	Reserved		Reserved.	0x0	R/W
29	INTSEL29		Outlier IRQ Enable.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
28	Reserved		Reserved.	0x0	R/W
27	INTSEL27		Data FIFO Underflow IRQ Enable.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
26	INTSEL26		Data FIFO Overflow IRQ Enable.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
25	INTSEL25		Data FIFO Threshold IRQ Enable.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
24	INTSEL24		Data FIFO Empty IRQ Enable.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
23	INTSEL23		Data FIFO Full IRQ Enable.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
[22:18]	Reserved		Reserved.	0x0	R/W
17	INTSEL17		Sequencer Timeout Error IRQ Enable.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
16	INTSEL16		Sequencer Timeout Finished IRQ Enable.	0x0	R/W

AFE INTERRUPTS

Table 177. Bit Descriptions for INTCSEL0 and INTCSEL1 Registers (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		0	Interrupt disabled.		
		1	Interrupt enabled.		
15	INTSEL15		End of Sequence IRQ Enable.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
14	Reserved		Reserved.	0x0	R/W
13	INTSEL13		Bootloader Done IRQ Enable.	0x1	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
12	INTSEL12		Custom Interrupt 3 Enable.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
11	INTSEL11		Custom Interrupt 2 Enable.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
10	INTSEL10		Custom Interrupt 1 Enable.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
9	INTSEL9		Custom Interrupt 0 Enable.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
8	Reserved		Reserved.	0x0	R/W
7	INTSEL7		Mean IRQ Enable.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
6	INTSEL6		ADC Delta Fail IRQ Enable.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
5	INTSEL5		ADC Maximum Fail IRQ Enable.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
4	INTSEL4		ADC Minimum Fail IRQ Enable.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
3	INTSEL3		Temperature Result IRQ Enable.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
2	INTSEL2		Sinc2 Filter Result Ready IRQ Enable.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
1	INTSEL1		DFT Result IRQ Enable.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
0	INTSEL0		ADC Result IRQ Enable.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		

AFE INTERRUPTS

Interrupt Controller Flag Registers

Address: 0x400C3010, Reset: 0x00000000, Name: INTCFLAG0

Address: 0x400C3014, Reset: 0x00000000, Name: INTCFLAG1

Table 178. Bit Descriptions for INTCFLAG0 and INTCFLAG1 Registers

Bits	Bit Name	Settings	Description	Reset	Access
31	FLAG31		Attempt to Break IRQ Status. This bit is set if a Sequence B request arrives when Sequence A is running, indicating that Sequence B is ignored.	0x0	R
		0	Interrupt not asserted.		
		1	Interrupt asserted.		
30	Reserved		Reserved.	0x0	R
29	FLAG29		Outlier IRQ Status.	0x0	R
		0	Interrupt not asserted.		
		1	Interrupt asserted.		
28	Reserved		Reserved.	0x0	R
27	FLAG27		Data FIFO Underflow IRQ Status.	0x0	R
		0	Interrupt not asserted.		
		1	Interrupt asserted.		
26	FLAG26		Data FIFO Overflow IRQ Status.	0x0	R
		0	Interrupt not asserted.		
		1	Interrupt asserted.		
25	FLAG25		Data FIFO Threshold IRQ Status.	0x0	R
		0	Interrupt not asserted.		
		1	Interrupt asserted.		
24	FLAG24		Data FIFO Empty IRQ Status.	0x0	R
		0	Interrupt not asserted.		
		1	Interrupt asserted.		
23	FLAG23		Data FIFO Full IRQ Status.	0x0	R
		0	Interrupt not asserted.		
		1	Interrupt asserted.		
[22:18]	Reserved		Reserved.	0x0	R
17	FLAG17		Sequencer Timeout Error IRQ Status.	0x0	R
		0	Interrupt not asserted.		
		1	Interrupt asserted.		
16	FLAG16		Sequencer Timeout Finished IRQ Status.	0x0	R
		0	Interrupt not asserted.		
		1	Interrupt asserted.		
15	FLAG15		End of Sequence IRQ Status.	0x0	R
		0	Interrupt not asserted.		
		1	Interrupt asserted.		
14	Reserved		Reserved.	0x0	R
13	FLAG13		Bootload Done IRQ Status.	0x0	R
		0	Interrupt not asserted.		
		1	Interrupt asserted.		
12	FLAG12		Custom Interrupt 3 Status.	0x0	R
		0	Interrupt not asserted.		
		1	Interrupt asserted.		
11	FLAG11		Custom Interrupt 2 Status.	0x0	R
		0	Interrupt not asserted.		
		1	Interrupt asserted.		

AFE INTERRUPTS

Table 178. Bit Descriptions for INTCFLAG0 and INTCFLAG1 Registers (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
10	FLAG10		Custom Interrupt 1 Status.	0x0	R
		0	Interrupt not asserted.		
		1	Interrupt asserted.		
9	FLAG9		Custom Interrupt 0 Status.	0x0	R
		0	Interrupt not asserted.		
		1	Interrupt asserted.		
8	Reserved		Reserved.	0x0	R
7	FLAG7		Mean IRQ Status.	0x0	R
		0	Interrupt not asserted.		
		1	Interrupt asserted.		
6	FLAG6		ADC Delta Fail IRQ Status. When this bit is set, it indicates that the difference between two consecutive ADC results is greater than the value specified by the ADCDELTA register. If this bit is clear, it indicates that no difference between two consecutive ADC values greater than the limit is detected since the last time this bit was cleared.	0x0	R
		0	Interrupt not asserted.		
		1	Interrupt asserted.		
5	FLAG5		ADC Maximum Fail IRQ Status. When this bit is set, it indicates that an ADC result is above the maximum value specified by the ADCMAX register. If this bit is clear, it indicates that no ADC value above the maximum is detected.	0x0	R
		0	Interrupt not asserted.		
		1	Interrupt asserted.		
4	FLAG4		ADC Minimum Fail IRQ Status. When this bit is set, it indicates that an ADC result is below the minimum value as specified by the ADCMIN register. If this bit is clear, it indicates that no ADC value below the limit is detected since the last time this bit was cleared.	0x0	R
		0	Interrupt not asserted.		
		1	Interrupt asserted.		
3	FLAG3		Temperature Result IRQ Status.	0x0	R
		0	Interrupt not asserted.		
		1	Interrupt asserted.		
2	FLAG2		Sinc2 Filter Result Ready IRQ Status.	0x0	R
		0	Interrupt not asserted.		
		1	Interrupt asserted.		
1	FLAG1		DFT Result IRQ Status.	0x0	R
		0	Interrupt not asserted.		
		1	Interrupt asserted.		
0	FLAG0		ADC Result IRQ Status.	0x0	R
		0	Interrupt not asserted.		
		1	Interrupt asserted.		

Analog Generation Interrupt Register

Address: 0x400C209C, Reset: 0x00000010, Name: AFEGENINTSTA

The AFEGENINTSTA register provides custom interrupt generation. Writing to this register is only possible using the sequencer.

Table 179. Bit Descriptions for AFEGENINTSTA Register

Bits	Bit Name	Description	Reset	Access
[31:4]	Reserved	Reserved.	0x1	R
3	CUSTOMINT3	General-Purpose Custom Interrupt 3. Set this bit manually using the sequencer program. Write 1 to this bit to trigger an interrupt.	0x0	R/W1C

AFE INTERRUPTS

Table 179. Bit Descriptions for AFEGENINTSTA Register (Continued)

Bits	Bit Name	Description	Reset	Access
2	CUSTOMINT2	General-Purpose Custom Interrupt 2. Set this bit manually using the sequencer program. Write 1 to this bit to trigger an interrupt.	0x0	R/W1C
1	CUSTOMINT1	General-Purpose Custom Interrupt 1. Set this bit manually using the sequencer program. Write 1 to this bit to trigger an interrupt.	0x0	R/W1C
0	CUSTOMINT0	General-Purpose Custom Interrupt 0. Set this bit manually using the sequencer program. Write 1 to this bit to trigger an interrupt.	0x0	R/W1C

SLEEP AND WAKE-UP TIMER

SLEEP AND WAKE-UP TIMER FEATURES

The ADuCM355 integrates a 20-bit sleep and wake-up timer. The sleep and wake-up timer provides automated control of the sequencer and can run up to eight sequences sequentially in any order from SEQ0 to SEQ3. Each sequence has a defined sleep period (SEQxSLEEPx) and a defined active period (SEQxWUPx). The timer is clocked from the internal 32 kHz oscillator clock source.

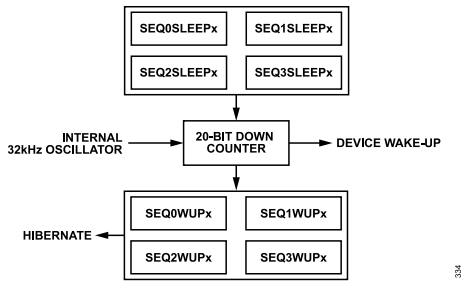

Figure 34. Sleep and Wake-Up Timer Block Diagram

SLEEP AND WAKE-UP TIMER OVERVIEW

The sleep and wake-up timer block consists of a 20-bit timer that counts down. The source clock is the 32 kHz, internal, low frequency oscillator.

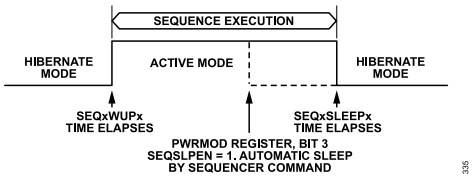

Figure 35. Sleep and Wake-Up Timing Diagram

When the timer elapses, the device wakes up and runs a sequence automatically. Up to eight sequences can run sequentially.

When the timer elapses, the device returns to sleep. If the timer elapses before the sequence completes execution, the remaining commands in the sequence are ignored. Therefore, the user code must ensure that the values in the SEQxSLEEPx registers are large enough to allow sequences to execute all commands.

It is recommended to use the wake-up timer to disable the timer sleep function (AFE PWRMOD, Bit 2 = 0) and use the sequencer to enter hibernate mode. Set the AFE PWRMOD register, Bit 3 = 1 to enable the sequencer to put the device in hibernate mode.

CONFIGURING A DEFINED SEQUENCE ORDER

The sleep and wake-up timer provides a feature that allows a specific order of sequences to execute periodically. The order in which the sequences are executed is defined in the SEQORDER register. There are eight available slots in this register (A to H). Each slot can be configured with any one of the four sequences. Figure 36 shows an example diagram of this feature. There are four defined sequences executed, SEQ0, SEQ1, SEQ2, and SEQ3, as shown in Figure 36.

To configure the ADuCM355 to implement this sequence order, implement the following register settings:

1. SEQORDER, Bits[1:0] = 00
2. SEQORDER, Bits[3:0] = 01
3. SEQORDER, Bits[5:4] = 10
4. SEQORDER, Bits[7:6] = 11
5. CON, Bits[3:1] = 011

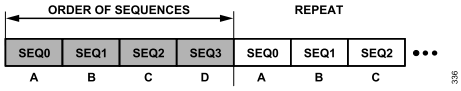

Figure 36. Sequence Order Diagram

SLEEP AND WAKE-UP TIMER

RECOMMENDED SLEEP AND WAKE-UP TIMER OPERATION

When using the sleep and wake-up timer to optimize performance and power consumption, the following procedure is recommended:

1. Set the AFE PWRMOD register, Bit 2 to disable the timer sleep function. Note that the sleep and wake-up timer does not place the device in hibernate mode. Instead, write to the SEQTRGSLP register at the end of the sequence to place the device in sleep mode. Sleep mode optimizes power consumption.
2. Set the TMRCON register, Bit 0 to enable the timer wake-up function.
3. Set the AFE PWRMOD register, Bits[3:1] and the SEQSLPLOCK register to 0xA47E5 to enable the sequencer to trigger sleep.
4. Set the final sequence in the CON register, Bits[3:1]. If only one sequence is used, select that sequence.
5. Write the sleep time and wake-up time to the SEQxSLEEPH, SEQxSLEEPL, SEQxWUPH, and SEQxWUPL registers.
6. Use the SEQORDER register to configure the order in which sequences are triggered.
7. Write 1 to the CON register, Bit 0, to enable the timer.

When Bit 0 of the CON register = 1, the timer loads the values from the SEQxWUPH and SEQxWUPL registers and begins counting down. When the timer reaches zero, the device wakes up and executes sequences in the order specified in the SEQORDER register, Bits[1:0]. The timer loads the values from the SEQxSLEEPH and SEQxSLEEPL registers and begins counting down again when the sequencer is running. When the timer elapses, the ADuCM355 returns to sleep mode if the TMRCON register, Bit 0 = 1. If the AFE PWRMOD register, Bit 3 = 1, the ADuCM355 returns to sleep mode at the end of the last sequence.

The maximum hibernate time is 32 seconds when using the internal 32 kHz oscillator.

To calculate the code for the SEQxWUPx and SEQxSLEEPx registers, use the following equation:

$$Code = ClkFreq \times Time$$

where:
Code is the code value for the SEQxWUPx register.
ClkFreq is the frequency of the internal oscillator in Hz.
Time is the required timeout duration in seconds.

SLEEP AND WAKE-UP TIMER REGISTERS

Table 180. Sleep and Wake-Up Timer Registers Summary

Address	Name	Description	Reset	Access
0x400C0800	CON	Timer control register	0x0000	R/W
0x400C0804	SEQORDER	Order control register	0x0000	R/W
0x400C0808	SEQ0WUPL	Sequence 0 wake-up time register (LSB)	0xFFFF	R/W
0x400C0818	SEQ1WUPL	Sequence 1 wake-up time register (LSB)	0xFFFF	R/W
0x400C0828	SEQ2WUPL	Sequence 2 wake-up time register (LSB)	0xFFFF	R/W
0x400C0838	SEQ3WUPL	Sequence 3 wake-up time register (LSB)	0xFFFF	R/W
0x400C080C	SEQ0WUPH	Sequence 0 wake-up time register (MSB)	0x000F	R/W
0x400C081C	SEQ1WUPH	Sequence 1 wake-up time register (MSB)	0x000F	R/W
0x400C082C	SEQ2WUPH	Sequence 2 wake-up time register (MSB)	0x000F	R/W
0x400C083C	SEQ3WUPH	Sequence 3 wake-up time register (MSB)	0x000F	R/W
0x400C0810	SEQ0SLEEPL	Sequence 0 sleep time register (LSB)	0xFFFF	R/W
0x400C0820	SEQ1SLEEPL	Sequence 1 sleep time register (LSB)	0xFFFF	R/W
0x400C0830	SEQ2SLEEPL	Sequence 2 sleep time register (LSB))	0xFFFF	R/W
0x400C0840	SEQ3SLEEPL	Sequence 3 sleep time register (LSB)	0xFFFF	R/W
0x400C0814	SEQ0SLEEPH	Sequence 0 sleep time register (MSB)	0x000F	R/W
0x400C0824	SEQ1SLEEPH	Sequence 1 sleep time register (MSB)	0x000F	R/W
0x400C0834	SEQ2SLEEPH	Sequence 2 sleep time register (MSB)	0x000F	R/W
0x400C0844	SEQ3SLEEPH	Sequence 3 sleep time register (MSB)	0x000F	R/W
0x400C0A1C	TMRCON	Timer wake-up configuration register	0x0000	R/W

SLEEP AND WAKE-UP TIMER

Timer Control Register

Address: 0x400C0800, Reset: 0x0000, Name: CON

The CON register is the wake-up timer control register.

Table 181. Bit Descriptions for CON Register

Bits	Bit Name	Settings	Description	Reset	Access
[15:7]	Reserved		Reserved.	0x0	R
6	MSKTRG		Mask Sequence Trigger From Sleep and Wake-Up Timer. This bit masks the sequence trigger from the sleep and wake-up timer. When the trigger is masked, it does not go to the sequencer.	0x0	R/W
[5:4]	RESERVED		Reserved.	0x0	R
[3:1]	ENDSEQ		End Sequence. These bits select one of the SEQORDER bits to end the timing sequence.	0x0	R/W
		000	The sleep and wake-up timer stops at Sequence A and then goes back to Sequence A.		
		001	The sleep and wake-up timer stops at Sequence B and then goes back to Sequence A.		
		010	The sleep and wake-up timer stops at Sequence C and then goes back to Sequence A.		
		011	The sleep and wake-up timer stops at Sequence D and then goes back to Sequence A.		
		100	The sleep and wake-up timer stops at Sequence E and then goes back to Sequence A.		
		101	The sleep and wake-up timer stops at Sequence F and then goes back to Sequence A.		
		110	The sleep and wake-up timer stops at Sequence G and then goes back to Sequence A.		
		111	The sleep and wake-up timer stops at Sequence H and then goes back to Sequence A.		
0	EN		Sleep and Wake-Up Timer Enable Bit.	0x0	R/W
		0	Disables the sleep and wake-up timer.		
		1	Enables the sleep and wake-up timer.		

Order Control Register

Address: 0x400C0804, Reset: 0x0000, Name: SEQORDER

The SEQORDER register controls the command sequence execution order.

Table 182. Bit Descriptions for SEQORDER Register

Bits	Bit Name	Settings	Description	Reset	Access
[15:14]	SEQH		Sequence H Configuration. These bits select SEQ0, SEQ1, SEQ2, or SEQ3 for Timer Sequence H.	0x0	R/W
		00	Fills in SEQ0.		
		01	Fills in SEQ1.		
		10	Fills in SEQ2.		
		11	Fills in SEQ3.		
[13:12]	SEQG		Sequence G Configuration. These bits select SEQ0, SEQ1, SEQ2, or SEQ3 for Timer Sequence G.	0x0	R/W
		00	Fills in SEQ0.		
		01	Fills in SEQ1.		
		10	Fills in SEQ2.		
		11	Fills in SEQ3.		
[11:10]	SEQF		Sequence F Configuration. These bits select SEQ0, SEQ1, SEQ2, or SEQ3 for Timer Sequence F.	0x0	R/W
		00	Fills in SEQ0.		
		01	Fills in SEQ1.		
		10	Fills in SEQ2.		
		11	Fills in SEQ3.		
[9:8]	SEQE		Sequence E Configuration. These bits select SEQ0, SEQ1, SEQ2, or SEQ3 for Timer Sequence E.	0x0	R/W
		00	Fills in SEQ0.		
		01	Fills in SEQ1.		
		10	Fills in SEQ2.		
		11	Fills in SEQ3.		

SLEEP AND WAKE-UP TIMER

Table 182. Bit Descriptions for SEQORDER Register (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
[7:6]	SEQD		Sequence D Configuration. These bits select SEQ0, SEQ1, SEQ2, or SEQ3 for Timer Sequence D.	0x0	R/W
		00	Fills in SEQ0.		
		01	Fills in SEQ1.		
		10	Fills in SEQ2.		
		11	Fills in SEQ3.		
[5:4]	SEQC		Sequence C Configuration. These bits select SEQ0, SEQ1, SEQ2, or SEQ3 for Timer Sequence C.	0x0	R/W
		00	Fills in SEQ0.		
		01	Fills in SEQ1.		
		10	Fills in SEQ2.		
		11	Fills in SEQ3.		
[3:2]	SEQB		Sequence B Configuration. These bits select SEQ0, SEQ1, SEQ2, or SEQ3 for Timer Sequence B.	0x0	R/W
		00	Fills in SEQ0.		
		01	Fills in SEQ1.		
		10	Fills in SEQ2.		
		11	Fills in SEQ3.		
[1:0]	SEQA		Sequence A Configuration. These bits select SEQ0, SEQ1, SEQ2, or SEQ3 for Timer Sequence A.	0x0	R/W
		00	Fills in SEQ0.		
		01	Fills in SEQ1.		
		10	Fills in SEQ2.		
		11	Fills in SEQ3.		

Sequence 0 to Sequence 3 Wake-Up Time Registers (LSB)

Address: 0x400C0808, Reset: 0xFFFF, Name: SEQ0WUPL

Address: 0x400C0818, Reset: 0xFFFF, Name: SEQ1WUPL

Address: 0x400C0828, Reset: 0xFFFF, Name: SEQ2WUPL

Address: 0x400C0838, Reset: 0xFFFF, Name: SEQ3WUPL

These registers set the sequence sleep time. The counter is 20 bits. These registers set the 16 LSBs. When this timer elapses, the device wakes up.

Table 183. Bit Descriptions for SEQxWUPL Registers

Bits	Bit Name	Description	Reset	Access
[15:0]	WAKEUPTIME0[15:0]	Sequence and Sleep Period. This register defines the length of time in which the device stays in sleep mode. When this time elapses, the device wakes up.	0xFFFF	R/W

Sequence 0 to Sequence 3 Wake-Up Time Registers (MSB)

Address: 0x400C080C, Reset: 0x000F, Name: SEQ0WUPH

Address: 0x400C081C, Reset: 0x000F, Name: SEQ1WUPH

Address: 0x400C082C, Reset: 0x000F, Name: SEQ2WUPH

Address: 0x400C083C, Reset: 0x000F, Name: SEQ3WUPH

These registers set the sequence sleep time. The counter is 20 bits. These registers set the 4 MSBs. When this timer elapses, the device wakes up.

Table 184. Bit Descriptions for SEQxWUPH Registers

Bits	Bit Name	Description	Reset	Access
[15:4]	Reserved	Reserved.	0x0	R

SLEEP AND WAKE-UP TIMER

Table 184. Bit Descriptions for SEQxWUPH Registers (Continued)

Bits	Bit Name	Description	Reset	Access
[3:0]	WAKEUPTIME0[19:16]	Sequence and Sleep Period. This register defines the length of time in which the device stays in sleep mode. When this time elapses, the device wakes up.	0xF	R/W

Sequence 0 to Sequence 3 Sleep Time Registers (LSB)

Address: 0x400C0810, Reset: 0xFFFF, Name: SEQ0SLEEPL

Address: 0x400C0820, Reset: 0xFFFF, Name: SEQ1SLEEPL

Address: 0x400C0830, Reset: 0xFFFF, Name: SEQ2SLEEPL

Address: 0x400C0840, Reset: 0xFFFF, Name: SEQ3SLEEPL

The SEQxSLEEPL registers define the device active time for SEQ0 to SEQ3. The counter is 20 bits. These registers set the 16 LSBs.

Table 185. Bit Descriptions for SEQxSLEEPL Registers

Bits	Bit Name	Description	Reset	Access
[15:0]	SLEEPTIME0[15:0]	Sequence and Active Period. This register defines the length of time in which the device stays in active mode. When this time elapses, the device returns to a sleep state.	0xFFFF	R/W

Sequence 0 to Sequence 3 Sleep Time Registers (MSB)

Address: 0x400C0814, Reset: 0x000F, Name: SEQ0SLEEPH

Address: 0x400C0824, Reset: 0x000F, Name: SEQ1SLEEPH

Address: 0x400C0834, Reset: 0x000F, Name: SEQ2SLEEPH

Address: 0x400C0844, Reset: 0x000F, Name: SEQ3SLEEPH

The SEQxSLEEPH registers define the device active time for SEQ0 to SEQ3. The counter is 20 bits. These registers set the four MSBs.

Table 186. Bit Descriptions for SEQxSLEEPH Registers

Bits	Bit Name	Description	Reset	Access
[15:4]	Reserved	Reserved.	0x0	R
[3:0]	SLEEPTIME0[19:16]	Sequence and Active Period. These bits define the length of time in which the device stays in active mode. When this time elapses, the device returns to a sleep state.	0xF	R/W

Timer Wake-Up Configuration Register

Address: 0x400C0A1C, Reset: 0x0000, Name: TMRCON

Table 187. Bit Descriptions for TMRCON Register

Bits	Bit Name	Settings	Description	Reset	Access
[15:1]	Reserved		Reserved.	0x0	R
0	TMRINTEN		Wake-Up Timer Enable. Set this bit before entering hibernate mode to enable the ability of the sleep and wake-up timer to wake up the chip.	0x0	R/W
		0	Wake-up timer disabled.		
		1	Wake-up timer enabled.		

USE CASE CONFIGURATIONS

The ADuCM355 is primarily designed for controlling and measuring electrochemical sensors. This section gives suggested setup details for the main use cases of the ADuCM355 with an electrochemical sensor.

HIBERNATE MODE WHILE MAINTAINING A DC BIAS TO THE SENSOR

In this mode, the digital die and analog die are in hibernate mode.

Digital Die

On the digital die, the low-power LDO regulator and low power, 32 kHz oscillator are active. This LDO regulator and oscillator support retention of the SRAM contents and maintains the configuration of the digital input/output pins while minimizing current consumption.

Analog Die

On the analog die, the low-power LDO regulator and low-power, 32 kHz oscillator are active. The low-power reference, low-power DAC, low-power potentiostat amplifier, and low-power TIAs are also active to maintain a DC bias voltage to an external sensor. The switches connecting the low-power amplifiers to the external sensor are also unaffected when entering or exiting this mode. For code examples on how to enter and exit hibernate mode, see the Hibernate Mode, Mode 2 section.

To maintain a bias voltage to an electrochemical sensor, the recommended analog die configuration setting for the low-power potentiostat amplifier and low-power TIA is LPTIASWx = 0x302C. Figure 38 shows how the switches are configured in the low-power potentiostat amplifier with this value.

Minimize leakage in the unused switches in the switch matrix by tying an unused pin to the internal 1.8 V digital LDO regulator rail. Tying the unused pin also ties the excitation amplifier P, N, and D nodes to a fixed level. See the PL, PL2, NL2, and NL switches in Figure 37 to understand which signal paths must be tied to 1.8 V to minimize leakage current effects. To close these switches, use the following code:

```
SWCON &= 0xEFFFF;      // Clear SWCON[16] to allow access to NSWFULLCON/PSWFULLCON
NSWFULLCON |= 0xC00;     // Close NL2 and NL switches
PSWFULLCON |= 0x6000;   // Close PL2 and PL switches
SWCON |= 0x10000;     // Set SWCON[16] to update switches
```

USE CASE CONFIGURATIONS

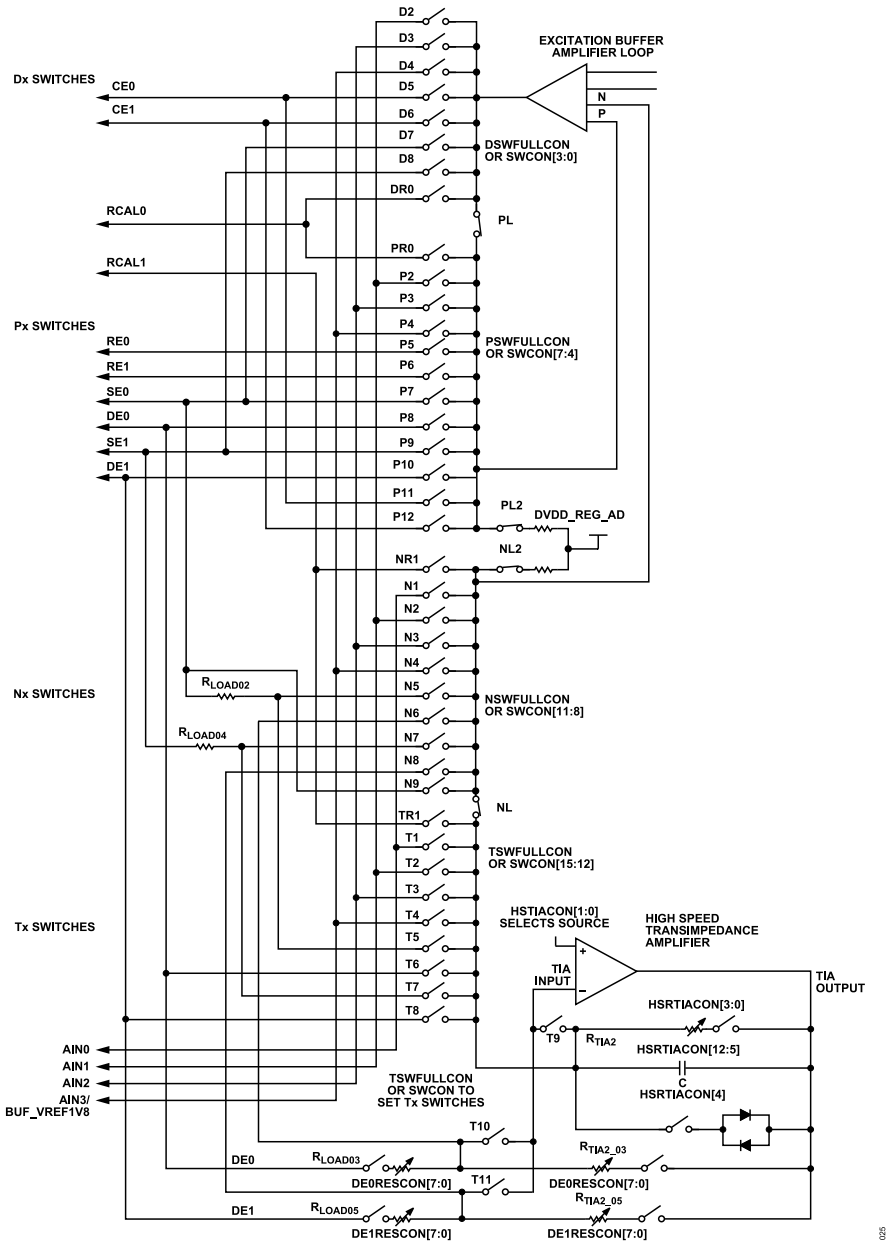

Figure 37. Recommended Switch Settings to Minimize Leakage in Hibernate Mode of Unused AFE Circuits

MEASURING A DC CURRENT OUTPUT

When measuring a DC current output, the ADC is powered on and set to measure the low-power TIA input channel. The user has the option to average the ADC results before reading to the digital die, or to use DMA mode to directly move the ADC results to the digital die SRAM.

Configure the Potentiostat Loop

Configure the low-power DACs to match the required sensor bias voltage of the external electrochemical in use. For example, for a sensor that requires a 0 V bias voltage between the working electrode and reference electrode, the V_{BIAS} and V_{ZERO} DAC outputs must be set to the same voltage. Likewise, LPDACDAT0 = 0x1A680 sets both VBIAS0 and VZERO0 outputs to approximately 1.1 V.

Additional guidelines are as follows:

- Configure the LPDACCON0 register as required. Set LPDACCON0, Bit 5 = 0 for normal operation of the low-power DAC switches.

USE CASE CONFIGURATIONS

- LPTIASW0, Bits[13:12] = 11 connects the V_{BIAS}/V_{ZERO} voltage sources to an external capacitor for noise filtering purposes.
- Configure the other low-power potentiostat switches appropriately.
  - For LPTIA0, set LPTIASW0 = 0x302C to set the low-power TIA0 switches for normal mode.
  - See Figure 38 for details about how these settings affect the low-power potentiostat loop.
- Configure the low-power TIA R_{LOAD} as per the ADuCM355 specifications via LPTIACON0, Bits[12:10]. For example, as follows:

```
pADI_AFE->LPTIACON0 = 0xE680;              // Enable LPTIA, 1 MΩ Low-Pass Filter resistor, 10 Ω
RLOAD, 100 K Gain resistor
```

After setting the sensor bias voltage, wait the settling period of the sensor before beginning to measure its output current.

Configure ADC for DC Measurement Mode

The ADC mux has two measurement options for the low-power TIA0. One is through the low-pass filter (ADCCON, Bits[5:0] = 0x21), and the other is the bypassed low-pass filter option (ADCCON, Bits[5:0] = 0x02). Select the low-pass filter as the positive input channel of the ADC for lowest noise. Select the LPTIA0 inverting input (ADCCON, Bits[12:8] = 0x2) as the ADC negative input channel.

Using the provided software libraries, the following function call selects LPTIA0 as the ADC input:

```
AfeAdcChan(MUXSELP_LPTIA0_LPF,
MUXSELN_VZERO0);               // Select LPTIA0_LPF input versus VZERO0 to the ADC
```

To select LPTIA1 as the ADC input, use the following function call:

```
AfeAdcChan(MUXSELP_LPTIA1_LPF,
MUXSELN_VZERO1);               // Select LPTIA1_LPF input versus VZERO1 to the ADC
```

For optimal performance, do not perform any read or write accesses to the analog die while the ADC is converting to minimize noise from the die to die interface coupling onto the ADC circuits. Optionally, halt the Arm Cortex-M3 core until the ADC conversions are complete.

The ADCCON and ADCBUFCON registers must also be configured. For example, as follows:

```
ADCCON = 0x010XXX; // PGA gain =1.5x
ADCBUFCON[3:0] = 0x4;                // Enables chopping of the ADC input - default value
```

Configure LPTIACON, Bits[15:13] to select the resistor value for the low-pass filter on the TIA output. This setting configures the programmable resistor, R_F, which is between the low-power TIA output and the ADC mux. The programmable resistor combines with the external capacitor on the AIN4_LPF0 and AIN7_LPF1 pins to form a low-pass filter on the low-power TIA outputs. The user must trade off the required ADC measurement settling time with the cutoff frequency on the low-power TIA output filter.

The ADuCM355 ADC provides optional digital postprocessing and filtering that can improve measurement accuracy of the sensor measurement. The ADCFILTERCON register provides programmable options for averaging, oversampling, and 50 Hz and 60 Hz mains rejection filtering. Trade the ADC update rate vs. required ADC noise performance when configuring the ADCFILTERCON register.

USE CASE CONFIGURATIONS

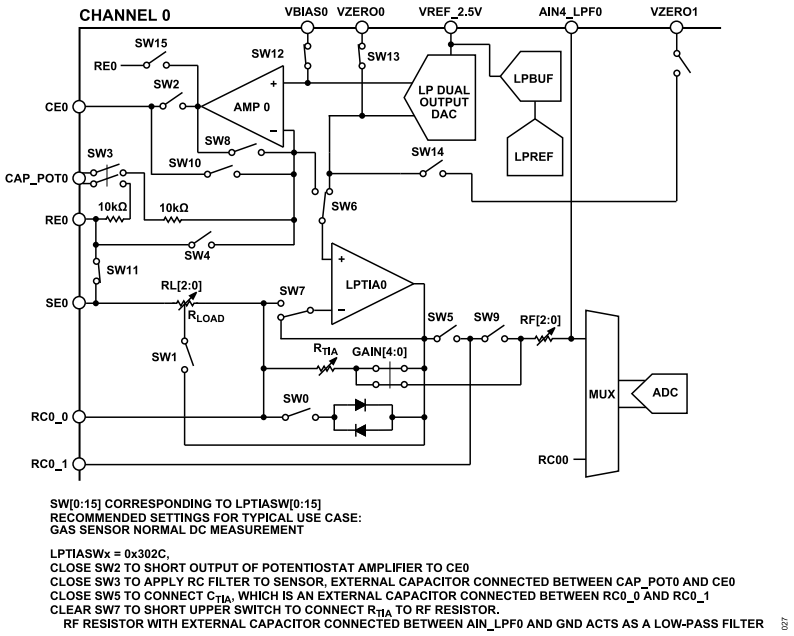

Figure 38. Switches for Low-Power Potentiostat Amplifier and Low-Power TIA to Measure DC Current from SE0 Node Using Low-Power TIA (3-Wire Electrochemical Sensor Configuration)

PULSE TEST (CHRONOAMPEROMETRY)

The pulse test involves disturbing the normal bias voltage of an electrochemical sensor and monitoring its output current response. The pulse test is usually used to check the responsiveness of the sensor. In the case of an electrochemical gas sensor, the pulse test checks that the passage of charge between electrodes through the internal electrolyte during oxidation and reduction is operating properly. Typically, the current increases sharply and quickly with the step in the sensor bias voltage. If the current step response is slow, there can be an issue with the sensor electrolyte.

Implementing Pulse Test Using Low-Power TIA

The potentiostat and the ADC are assumed to be initially configured using the steps in the Measuring a DC Current Output section. Current measurement before, during, and after the step is also described in this section.

The V_{BIAS} DAC level is stepped by 5 mV to 10 mV higher than it typically is to stimulate the sensor. A typical step duration is 100 ms.

To configure the switches in potentiostat mode for the pulse test, set LPTIASWx, Bits[11:0] = 0x014 to use the low-power TIA, write to the LPDACDAT0 register to change the VBIAS0 output voltage, and write to the LPDACDAT1 register to change the VBIAS1 output voltage.

Implementing Pulse Test Using High-Speed TIA

In this test, the user has the option of using the high-speed TIA channel to measure the current of the sensor. In this case, the SE0 or SE1 pin is routed to the high-speed TIA instead of the low-power TIA. Figure 40 shows the signal path.

The following code connects the SE0 pin to the high-speed TIA and disconnects the SE0 pin from the low-power TIA0:

```
    pADI_AFE->LPTIASW0 &= 0xF000;    // Mask SW12 to 15 control bits
    pADI_AFE->LPTIASW0 = 0x94;       // Disconnect SE from LPTIA -ve input and connect to HSTIA
                                     // Configure SW0 to SW11 for HSTIA PULSE/RAMP test setting
    pADI_AFE->LPDACCON0 |= 0x20;     // Configure LPDAC0 switches for Diagnostic mode
    pADI_AFE->LPDACSW0 = 0x32;       // Disconnect the VBIAS0 and VZERO0 from external caps
    pADI_AFE->SWCON &=
      (!BITM_AFE_SWCON_SWSOURCESEL); // Step 1: to write to T-Switch control register
    pADI_AFE->TSWFULLCON = 0x110;    // Step 2: Close T9 & T5. Leave T10, T11 open
    pADI_AFE->SWCON |=
```

USE CASE CONFIGURATIONS

```
       BITM_AFE_SWCON_SWSOURCESEL;    // Step 3: to write to T-Switch control register
AfeHSTIACon(AMPPOWER_NORM,
            HSTIABIAS_VZERO0); // Set common-mode source as Vzero0 if HSTIA  with Chan0 required
  AfeHSTIASeCfg(HSTIA_RTIA_80K,       // RTIA setting
                BITM_HSTIA_CTIA_1PF,  // internal load of 1pF
                0);                   // protection diodes disconnected
```

If the high-speed TIA is selected as the ADC input channel, it is as follows:

```
AfeAdcChan(MUXSELP_HSTIA_P,
           MUXSELN_HSTIA_N);      // Select HSTIA output as ADC input versus HSTIA_N to the ADC
```

If the sensor output pin is connected to the diagnostic electrodes (DE0 or DE1), the user can reduce the R_{LOAD} value to <100 Ω.

To generate sharp voltage transients on the sensor bias voltage, open SW12 and SW13 (as shown in Figure 16) to disconnect the low-power DAC outputs, V_{BIAS}, and V_{ZERO} from the external 100 nF filter capacitors. Configure LPTIASWx, Bits[13:12] = 00, LPDACCON0, Bit 5 = 1, and LPDACSW0 = 0x32.

To configure the switches in the potentiostat for the pulse test, set LPTIASWx, Bits[11:0] = 0x094 to use the high-speed TIA. See the Exiting Cyclic Voltammetry Mode section for recommendations to decrease the settling time of the sensor when exiting pulse testing.

CYCLIC VOLTAMMETRY

Cyclic voltammetry is similar to the pulse test, except that both the V_{BIAS} and V_{ZERO} outputs of the low-power DAC can change to set the sensor bias voltage and the common-mode voltage of the sensor to different levels. Use the high-speed TIA for the current output measurement from the sensor, rather than the low-power TIA.

To generate sharp voltage transients on the sensor bias voltage, open SW12 and SW13 (as shown in Figure 16) to disconnect the low-power DAC outputs and V_{BIAS} and V_{ZERO} from the external 100 nF filter capacitors. Configure LPTIASWx, Bits[13:12] = 00, LPDACCON0, Bit 5 = 1, and LPDACSW0 = 0x32 to open SW12 and SW13.

To configure the switches in the potentiostat loop for the ramp test with the high-speed TIA, set LPTIASWx, Bits[11:0] = 0x094. See Figure 39 for switch setup as a result of this configuration.

To capture the full current transient of each step in the voltammetry cycle, it may be necessary to optimize the ADC filter settings for speed rather than noise performance.

USE CASE CONFIGURATIONS

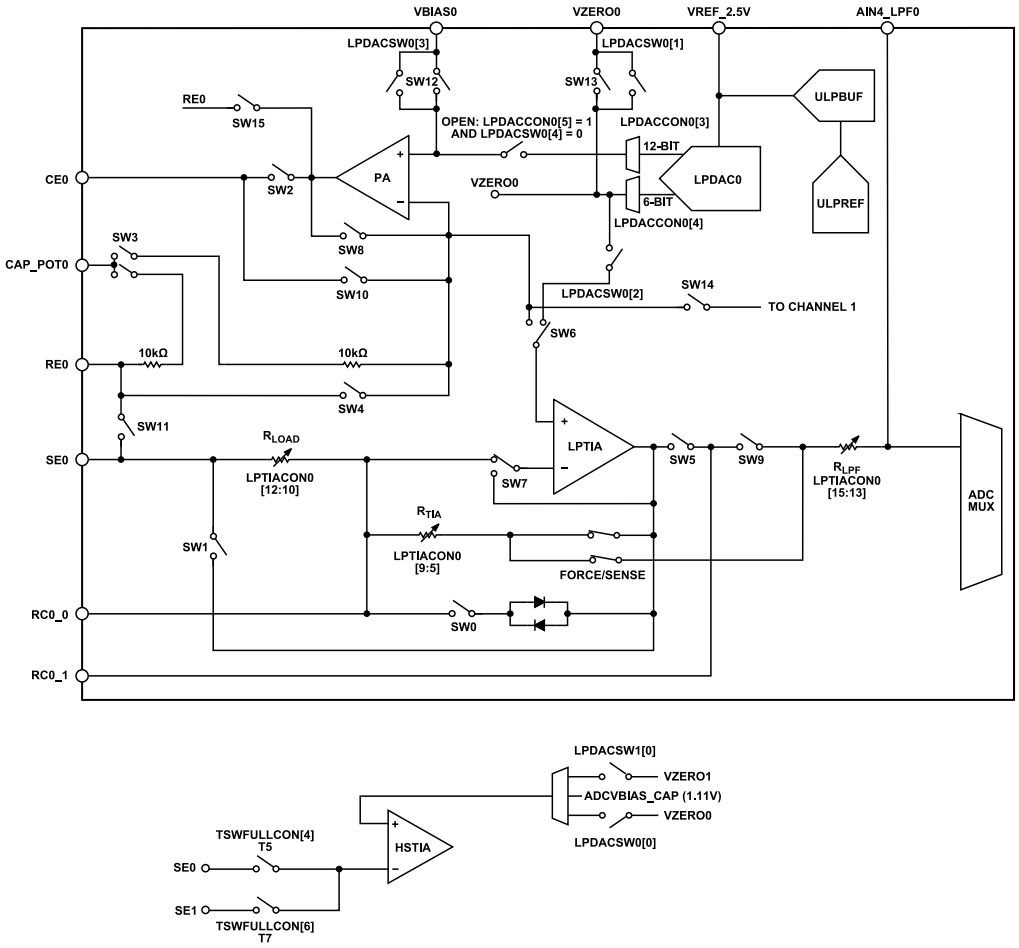

Figure 39. Switches for Low-Power Potentiostat, Low-Power TIA, and Switch Matrix to Perform Cyclic Voltammetry or Pulse Test on SE0 Node Using High-Speed TIA

USE CASE CONFIGURATIONS

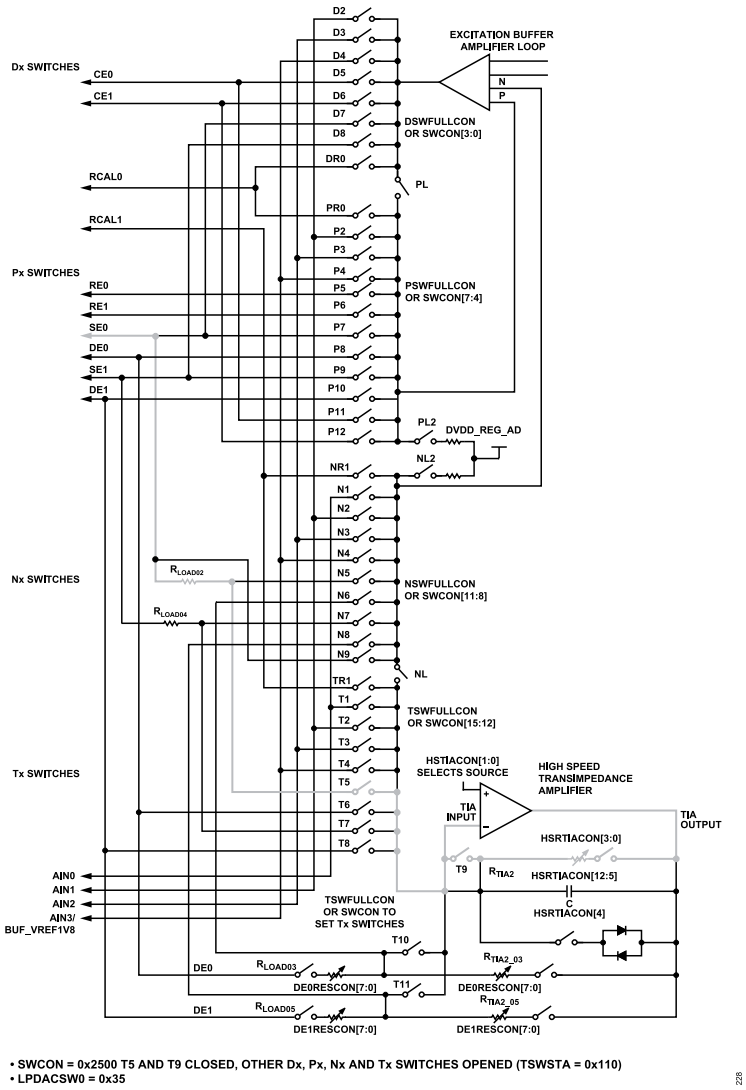

Figure 40. Signal Path for Low-Power Potentiostat, Low-Power TIA, and Switch Matrix to Perform Cyclic Voltammetry or Pulse Test on SE0 Node Using High-Speed TIA

Exiting Cyclic Voltammetry Mode

When exiting voltammetry mode to return to biasing the sensor normally or to resume taking DC measurements, take care to minimize the sensor settling time, as follows:

1. Before adjusting the switches for a normal DC bias, reconfigure the low-power DAC outputs to their required DC levels. Write to the LPDACDAT0 register or the LPDACDAT1 register.
2. Disconnect the high-speed TIA circuitry from the sensor. Open the T10 and T11 switches of the switch matrix by clearing TSWFULLCON, Bits[10:9] = 0b00. Disconnect the high-speed TIA gain resistors by setting DE0RESCON and DE1RESCON to 0xFF. Optionally, to save power, power down the high-speed TIA by clearing AFECON, Bit 11 = 0.
3. Configure the low-power DAC switches for normal DC measurements. Write LPDACSWx = 0. Clear LPDACCONx, Bit 5 = 0 for normal switch operation around the low-power DACs. Set up LPDACCONx, Bits[4:3] to set V_{BIAS} and V_{ZERO} to 12-bit or 6-bit mode.
4. After configuring the low-power DAC, set up the low-power TIA. Set the low-power TIA switches for normal operation. Optionally, the SW0 and SW1 switches around the low-power TIA can be closed for a short duration to allow the low-power TIA to charge up the sensor, allowing the SE0 node of the sensor to settle faster to the VZEROx output bias voltage setting. See Figure 16 to locate SW0 and SW1 in the low-power TIA circuitry.

USE CASE CONFIGURATIONS

AC IMPEDANCE MEASUREMENT WHILE MAINTAINING DC BIAS TO THE SENSOR

The following sections detail an example configuration setup for an AC impedance measurement of <80 kHz. The impedance measurement technique is a ratiometric measurement where an impedance measurement is completed on a known, fixed external R_{CAL} separately to the measurement of the impedance of the sensor.

In this example configuration, the impedance measurement is taken via the SE0 electrode using Electrochemical Sensor Channel 0. In Figure 41, Figure 42, and Figure 43, an AC signal of amplitude ±10 mV p-p is coupled onto a DC sensor biased to 0 V ($V_{BIAS} - V_{ZERO} = 0$). However, the DC sensor common-mode voltage is 1.1 V. The AC signal amplitude can be increased to 15 mV.

The high-speed DAC full-scale output with the attenuator on is approximately ±607 mV/40 = ±15.1 mV p-p. The voltage to the ADC is calculated as ±15.1 mV/R_{LOAD} × R_{TIA}. R_{LOAD02} is fixed at 100 Ω, which gives a current of approximately 150 µA across R_{TIA}. The testing featured in this reference manual is designed for an ADC voltage of ±750 mV. As such, set R_{TIA} = 5 kΩ.

The impedance measurement is performed in five steps, detailed in the following sections. The following steps assume a sensor with a 0 V bias requirement between the reference electrode and working electrode of a 3-electrode electrochemical sensor.

Step 1: Initialize ADuCM355 for Impedance Measurement

The electrochemical sensor remains biased via the low-power potentiostat loop. To configure the ADC and high-speed DAC operating mode, perform the following steps:

1. Configure the ADC and DAC circuits for low-power mode to minimize current consumption by clearing PMBW, Bit 0 = 0.
2. Set Bit 20, Bit 15, Bit 14, and Bits[11:5] of the AFECON register to 1 to enable high-speed DAC and ADC references, the high-speed DAC excitation amplifier and buffer, and DFT hardware accelerator. The waveform generator also must be enabled. Bit 21 is set when using a sensor with a DC bias voltage >0 V.
3. Enable chop mode on the ADC input buffer when measuring signals <80 kHz. ADCBUFCON, Bits[3:0] = 0x4 enable the ADC front-end buffer and PGA chop. When measuring signals >80 kHz (as in high-power mode), disable chopping on the ADC input buffer. ADCBUFCON, Bits[3:0] = 0xF disable ADC input chopping.

To set up the ADC, configure and calibrate the ADC. Ideally, calibrate the ADC as a current input (high-speed TIA) with the desired R_{TIA} and ADC PGA gain settings. See the ADC Calibration section for further details. Configure the ADC output data to go to the DFT block and configure the number of samples used by the DFT block in the DFTCON register.

To set up the high-speed DAC, first turn on the high-speed DAC. Use the waveform generator to generate a sine wave of the desired frequency and amplitude by appropriately configuring the following registers:

- HSDACCON. Configure the gain settings.
- PMBW, Bits[3:2]. Use this register to configure the reconstruction filter settings.
- WGCON. Main waveform control register.
- WGFCW. Configures the frequency of the AC sine wave. If necessary, adjust the WGPHASE, WGOFFSET, and WGAMPLITUDE registers.
- DACDCBUFCON. Select low-power DAC0 or low-power DAC1 as the DC level of the common-mode voltage excitation amplifiers.

After turning on the high-speed DAC, calibrate the high-speed DAC output if necessary (optional). The high-speed DAC can be calibrated to remove the offset error by setting the output code to 0x800, as follows:

- Connect the excitation amplifier to R_{CAL}.
- Measure the differential voltage across R_{CAL} by selecting the ADC inputs as the N node and P node of the excitation amplifier: ADCCON, Bits[12:0] = 0x1424. There are four offset calibration registers, DACOFFSET, DACOFFSETATTEN, DACOFFSETHP, and DACOFFSETATTENHP. The relevant register depends on the excitation amplifier gain setting and whether the device is in low or high-power mode. See Table 127.

To set up the potentiostat circuit for impedance measurement, ensure that the low-power DACs are on, with the VBIAS0 and VZERO0 outputs set to give the same voltage on the RE0 and SE0 pins. By default, leave the electrochemical sensor fully biased by the low-power potentiostat loop. Using a potentiostat circuit for impedance measurement uses the same settings as hibernate mode and for measuring a DC current output (LPTIASW0 = 0x302C).

USE CASE CONFIGURATIONS

The following code is an example of how to configure the switches in the low-power loop:

```
AfeLpTiaSwitchCfg (channel,
               SWMODE_NORM);  // Low Power Loop Normal switch settings (0x302C)
```

To set up the high-speed TIA for impedance measurement, perform the following steps:

1. Select the R_{TIA} values, power setting, and bias voltage source for the high-speed TIA. The HSRTIACON register configures the high-speed TIA parallel capacitor and the main R_{TIA} resistor. The HSTIACON register configures the power mode for the high-speed TIA. Clear HSTIACON, Bits[6:2] = 00000 for impedance measurements ≤80 kHz. Bits[1:0] are 00.
2. Select the high-speed TIA as the ADC input, as modeled in the following code.

```
AfeAdcChan(MUXSELP_HSTIA_P,
            MUXSELN_HSTIA_N);       //Select HSTIA output as ADC input versus HSTIA_N to the ADC
```

Step 2: Measure R_{LOAD02} and External Sensor (R_{SENSOR})

The electrochemical sensor remains biased during this step, but the working electrode voltage is set by the high-speed TIA instead of the low-power TIA.

The R_{LOAD02} is a fixed value load resistor (100 Ω). In Figure 41, the reference electrode is the sensor impedance. The waveform generator and the high-speed DAC generate a 10 mV amplitude sine waveform on the DC bias voltage that is required for the reference electrode and the sensing electrode sensor nodes. DACN is the bias voltage input to the excitation loop from the high-speed DAC. The sine wave generated by the high-speed DAC is added at the DACP node. N, P, DACN, and DACP are four inputs of the excitation amplifier. The differential voltage between the P node and the N node is the same as the differential voltage between DACP and DACN.

The sensor is biased, meaning that the required DC voltage is applied between the reference electrode and the working electrode. The AC signal is added to the sensor, and the DC bias voltage is also maintained. The D node provides the correct voltage and current. As such, the stimulus sine waveform is added between the electrochemical sensor reference electrode and the high-speed TIA input. The reference electrode + R_{LOAD02} is included in this AC excitation loop. At this point, the high-speed TIA output is measured via the ADC signal chain. The programmed number of ADC samples are fed to the DFT block, which outputs a complex number (real or imaginary) that reflects the DFT result for the ADC measurements of the reference electrode + R_{LOAD02}. By using the RCAL impedance measurement result determined after Step 4 and ratiometric measurements, it is possible to obtain an accurate reference electrode + R_{LOAD02} impedance value.

Configure the Tx, Dx, Nx, and Px switches appropriately, as per the following example code:

```
AfeSwitchDPNT(SWID_D5_CE0,SWID_P5_RE0,SWID_N5_SE0RLOAD,SWID_T5_SE0RLOAD|SWID_T9);
// Connect Excitation Amplifier D to the LP
// Connect Excitation Amplifier P to RE0
// Connect Excitation Amplifier N to SE0 via RLOAD02
// Connect HSTIA to SE0 via RLOAD02. Close T9
```

USE CASE CONFIGURATIONS

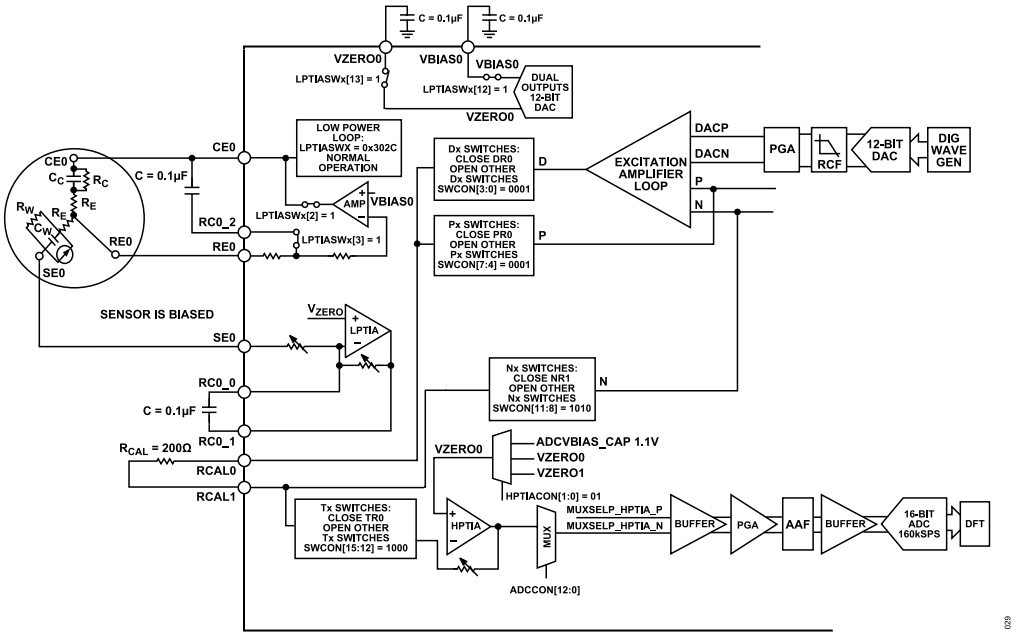

Figure 41. Step Three of Impedance Measurement, R_{LOAD02} + R_{SENSOR} Measurement

Step 3: Measure R_{LOAD02}

The counter electrode and reference electrode of the electrochemical sensor are floating during this stage of the measurement sequence.

The main differences between Step 2 and Step 3 are as follows:

- The reference electrode is disconnected from the excitation amplifier P node.
- The counter electrode is disconnected from the excitation amplifier D node and is connected directly to R_{LOAD02}.
- The excitation amplifier D node is connected to the working electrode node.

Therefore, the excitation signal is applied to R_{LOAD02} and the sensor is floating, as shown in Figure 42. The AC excitation loop D node, P node, and R_{LOAD02} are shorted. The AC excitation loop N node, TIA T node, and R_{LOAD02} are shorted. Measure the stimulus via the high-speed TIA. The ADC signal chain and DFT are used to calculate the impedance of the R_{LOAD02} resistor.

Configure the Tx, Dx, Nx, and Px switches appropriately, as described in the following example code:

```
AfeSwitchDPNT(SWID_D7_WE0,SWID_P7_WE0,SWID_N5_SE0RLOAD,SWID_T5_SE0RLOAD|SWID_T9);
// Connect Excitation Amplifier D to the WE Electrode
//  Connect Excitation Amplifier P to SE0
// Connect Excitation Amplifier N to SE0 via RLOAD02
// Connect HSTIA to SE0 via RLOAD02. Close T9
```

USE CASE CONFIGURATIONS

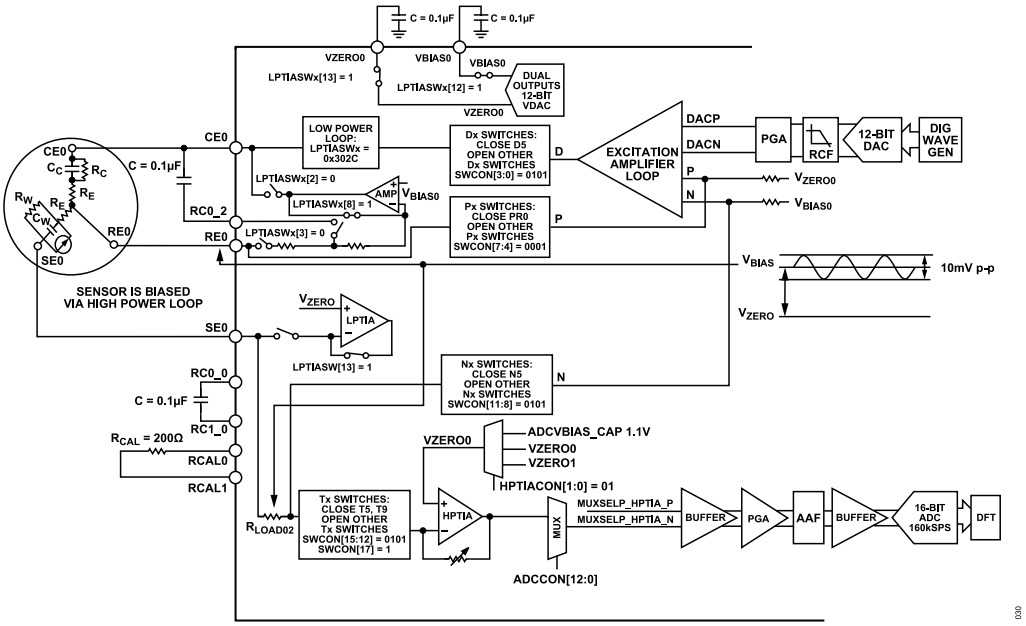

Figure 42. R_{LOAD02} Measurement

Step 4: Measure R_{CAL} via the Impedance Measurement Engine

R_{CAL} is a known external fixed resistor, typically 200 Ω, but can be other values. The electrochemical sensor DC bias voltage is set by the low-power potentiostat loop.

Configure the Tx, Dx, Nx, and Px switches appropriately, as per the following example code:

```
AfeSwitchDPNT(SWID_D7_WE0,SWID_P7_WE0,SWID_N5_SE0RLOAD,SWID_T5_SE0RLOAD|SWID_T9);
//  Connect Excitation Amplifier D to RCAL0
//  Connect Excitation Amplifier P to RCAL0
//  Connect Excitation Amplifier N to RCAL1
//  Connect HSTIA to RCAL1. Close T9
```

USE CASE CONFIGURATIONS

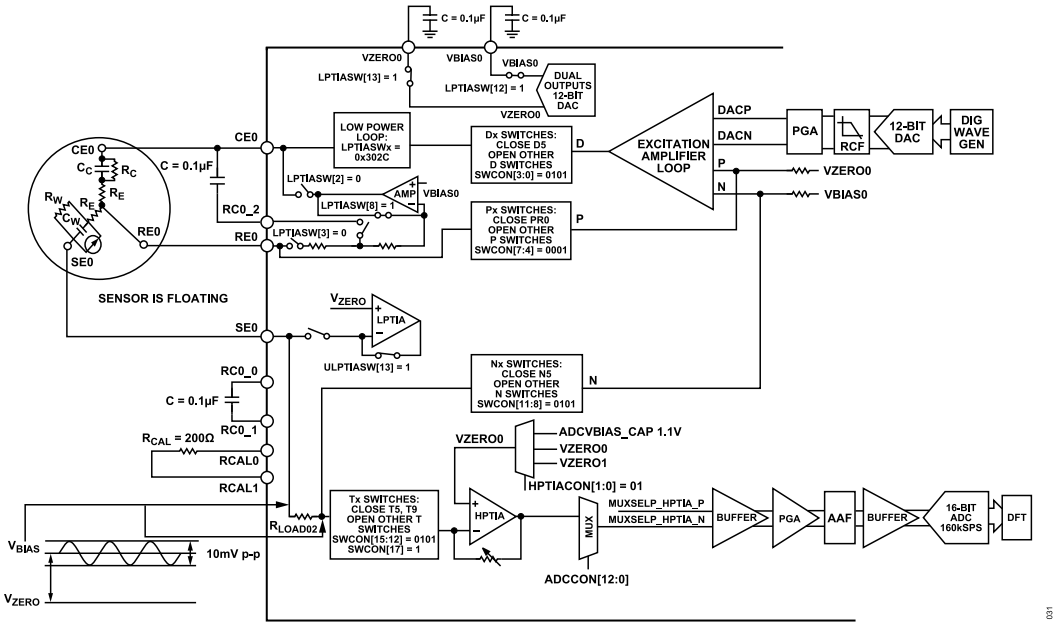

Figure 43. R_{CAL} Measurement

The R_{CAL} measurement is used to calibrate the R_{SENSOR} + R_{LOAD} measurement in Step 3 by using ratiometric measurements. As shown in Figure 43, the electrochemical sensor is biased via the low-power potentiostat and low-power TIA amplifiers for Step 4. The waveform generator and the high-speed DAC generate a 10 mV amplitude sine waveform.

The AC excitation loop D node, P node, and RCAL0 pin are shorted by setting the Dx switch and Px switch to close the DR0 and PR0 switches (SWCON, Bits[7:0] = 0x11). Other Dx switches and Px switches are open. The AC excitation loop N node, high-speed TIA T node, and RCAL1 pin are shorted by setting the Nx switches and Tx switches to close the NR1 and TR1 switches (SWCON, Bits[15:8] = 0x85). Close T9 to select R_{TIA02} (SWCON Bit 17 = 1). Other Nx switches and Tx switches are open. The stimulus is added to the 200 Ω R_{CAL} and the high-speed TIA. The ADC signal chain and DFT block are used to calculate the impedance of R_{CAL}.

Set the low-power loop switches to allow the excitation amplifier and the high-speed TIA to set the DC bias voltage level of the sensor. The following example code demonstrates how to configure the switches in the low-power loop:

```
AfeLpTiaSwitchCfg (channel, SWMODE_AC);   //Low Power Loop for AC impedance switch settings (0x3180)
```

USE CASE CONFIGURATIONS

Step 5: Calculate the Impedance of Electrochemical Sensor Sensing Electrode Node

Calculate the magnitude and phase of Step 2, Step 3, and Step 4. The following code example and equations show how to calculate the magnitude and phase of each step:

```
   // Get RMS result of each step based on DFT REAL and IMAG outputs
        DFT_Mag[i] = DFTREAL* DFTREAL+ DFTIMAG* DFTIMAG;
        DFT_Mag[i] = sqrt(DFT_Mag[i]);
   // Use atan2 function to get phase result of each step based on DFT REAL and IMAG outputs
         Phase[i] = atan2(DFTIMAG, DFTREAL);  // returns value between -pi to +pi (radians) of
ATAN2(IMAG/Real)
```

In Step 4,

$$\frac{V(s)}{Z_{CAL(magnitude)}} = I(s)_{CAL} \tag{12}$$

where:
V(s) is the signal voltage.
$I(s)_{CAL}$ is the calibration current.
$Z_{CAL(magnitude)}$ is the magnitude of the impedance of the calibration.

Therefore,

$$V(s) = I(s) \times Z_{CAL} \tag{13}$$

In Step 2,

$$\frac{V(s)}{Z_{RE\ +\ RLOAD02}} = I(s)_{RE\ +\ RLOAD02} \tag{14}$$

where:
$Z_{RE + RLOAD02}$ is the impedance of the sensor and Load Resistor 2.
$I(s)_{RE + RLOAD02}$ is the current flowing through the sensor and Load Resistor 2.

Therefore,

$$Z_{RE\ +\ RLOAD02} = \frac{V(s)}{I(s)_{RE\ +\ RLOAD02}} \tag{15}$$

In Step 3,

$$\frac{V(s)}{Z_{RLOAD02}} = I(s)_{RLOAD02} \tag{16}$$

where:
$Z_{RLOAD02}$ is the impedance of Load Resistor 2.
$I(s)_{RLOAD02}$ is the current flowing through Load Resistor 2.

Therefore,

$$Z_{RLOAD02} = \frac{V(s)}{I(s)_{RLOAD02}} \tag{17}$$

Therefore, based on Step 2 through Step 4,

$$\begin{aligned} Z_{RE} &= Z_{RE+RLOAD02} - Z_{RLOAD02} = \frac{V(s)}{I(s)_{RE+RLOAD02}} - \frac{V(s)}{I(s)_{RLOAD02}} \\ &= I(s)_{CAL} \times Z_{CAL} \times \left(\frac{1}{I(s)_{RE+RLOAD02}} - \frac{1}{I(s)_{RLOAD02}}\right) \\ &= I(s)_{CAL} \times Z_{CAL} \times \frac{I(s)_{RLOAD02} - I(s)_{RE+RLOAD02}}{I(s)_{RE+RLOAD02} \times I(s)_{RLOAD02}} \end{aligned} \tag{18}$$

USE CASE CONFIGURATIONS

If Z_{CAL}(magnitude) = 200 Ω and Z_{CAL} (phase) = 0, it is possible to obtain the following equations:

$$Z_{RE}(magnitude) = |Z_{RE+RLOAD02} - Z_{RLOAD02}| = 200 \times |I(s)_{CAL}| \times \frac{|I(s)_{RLOAD02} - I(s)_{RE+RLOAD02}|}{|I(s)_{RE+RLOAD02}| \times |I(s)_{RLOAD02}|} \quad (19)$$

where:
Z_{RE}*(magnitude)* is the magnitude of the impedance of the sensor.
I(s) values can be replaced by DFT results.

$$\begin{aligned} Z_{RE}(phase) &= \mathrm{Ang}_{(Z_{RE+RLOAD02} - Z_{RLOAD02})} \\ &= 0 + \mathrm{Ang}_{I(s)_{CAL}} + \mathrm{Ang}_{I(s)_{RLOAD02} - I(s)_{RE+RLOAD02}} - \mathrm{Ang}_{I(s)_{RE+RLOAD02}} - \mathrm{Ang}_{I(s)_{RLOAD02}} \end{aligned} \quad (20)$$

where:
Z_{RE}*(phase)* is the phase part of the impedance measurement of the sensor.
Ang() is the function to calculate phase.

DMA CONTROLLER

The DMA controller is used to perform data transfer tasks between peripherals and memory locations to offload these tasks from the microcontroller unit (MCU). Data can be moved quickly by the DMA without CPU actions, keeping the CPU free for other operations.

DMA FEATURES

The ADuCM355 provides dedicated and independent DMA channels. There are two programmable priority levels for each DMA channel. Each priority level arbitrates using a fixed priority that is determined by the DMA channel number. Channels with lower numbers have higher priority. For example, SPI0 transmit has the highest priority, and the next highest priority is the SPI0 receive.

Each DMA channel can access a primary or alternate channel control structure. Multiple DMA transfer types are supported, such as the following:

- Memory to memory.
- Memory to peripheral.
- Peripheral to memory.

DMA OVERVIEW

The DMA controller has 20 channels in total. The 20 channels are dedicated to managing DMA requests from specific peripherals. Channels are assigned, as shown in Table 188.

Table 188. DMA Channel Assignment

Channel Number	Peripheral Description
0	SPI1 transmit
1	SPI1 receive
2, 3, 6, 7, 13, 14	Reserved
4	SPI0 transmit
5	SPI0 receive
8	UART0 transmit
9	UART0 receive
10	I2C target transmit
11	I2C target receive
12	I2C initiator
15	Flash
16	Software DMA
17	AFE die ADC
18 to 23	Software DMA

DMA ANALOG DIE

The ADC on the AFE die can be connected to the DMA controller. An eight-word FIFO is provided to buffer. The output of the FIFO is the DMA controller. The user can select from the following inputs:

- ADC sinc3 result (16 bits)
- DFT result, real first followed by imaginary part (18 bits for each device)
- ADC sinc2 and low-pass filter result (16 bits)
- ADC mean result (16 bits)

AFE Die Data FIFO

DMA Channel 17 is associated with the AFE die data FIFO. To enable this die, write 1 to FIFOCON, Bit 12. When the FIFO is enabled (FIFOCON, Bit 11 = 1) and FIFOCON, Bit 12 = 1, the FIFO issues a DMA request any time the FIFO is not empty.

DMA CONTROLLER

Program Flow

After performing the following steps, the data FIFO issues DMA requests whenever the FIFO receives data. If the number of bytes transferred matches the value specified by Bits[13:4] of the control data configuration register (CFG), the DMA_DONE internal interrupt is asserted. To set up the DMA controller for a particular DMA channel, follow these steps:

1. Enable the DMA controller by setting Bit 0 of CFG to 1.
2. Enable the DMA data FIFO channel and set Bit 17 of CFG to EN_SET.
3. Configure the DMA control description for the data FIFO channel. Refer to the Channel Control Data Structure section.
4. Select the source for data FIFO (FIFOCON, Bits[15:13]).
5. Set FIFOCON, Bit 11 and FIFOCON, Bit 12 to enable the FIFO and DMA requests.
6. Enable the DMA_DONE interrupt.

DMA ARCHITECTURAL CONCEPTS

The DMA channel provides a means to transfer data between memory spaces or between memory and a peripheral using the system interface. The DMA channel provides an efficient means of distributing data throughout the system, freeing up the core for other operations. Each peripheral that supports DMA transfers has its own dedicated DMA channel or channels with their own register sets that configure and control the operating modes of the DMA transfers.

DMA OPERATING MODES

The DMA controller has two buses, one connected to the system bus shared with the Cortex-M3 core and the other connected to 16-bit peripherals. The DMA request can stop CPU access to the system bus for several bus cycles, such as when the CPU and DMA target the same destination (memory or peripheral). The DMA controller fetches channel control data structures located in the system memory to perform data transfers.

DMA capable peripherals, when enabled to use the DMA, can request the DMA controller for a transfer. At the end of the programmed number of DMA transfers for a channel, the DMA controller generates a single cycle DMA_DONE interrupt corresponding to that channel. The DMA_DONE interrupt indicates the completion of the DMA transfer. Separate interrupt enable bits are available in the NVIC for each of the DMA channels.

CHANNEL CONTROL DATA STRUCTURE

Every channel has two associated control data structures: primary and alternate. For simple transfer modes, the DMA controller uses either the primary or the alternate data structure. For more complex data transfer modes, such as ping pong or scatter gather, the DMA controller uses both the primary and alternate data structures. Both control data structures occupy four 32-bit locations in the memory, as detailed in Table 189. The entire channel control data structure is described in Table 190.

Before the controller can perform a DMA transfer, the data structure related to the DMA channel must be programmed at the designated location in system memory, SRAM. The programming determines the source and destination data size, number of transfers, and the number of arbitrations. The contents of the designated memory locations are as follows:

- The source end pointer memory location contains the end address of the source data.
- The destination end pointer memory location contains the end address of the destination data.
- The control data configuration memory location contains the channel configuration control data.

Table 189. Channel Control Data Structure

Offset Address	Offset Register Name	Description
0x00	SRC_END_PTR	Source end pointer
0x04	DST_END_PTR	Destination end pointer
0x08	CHNL_CFG	Control data configuration
0x0C	Reserved	Reserved

Table 190. Memory Map of Primary and Alternate DMA Structures[1]

	Primary Structures		Alternate Structures	
Channel Number	Register Description	Offset Address	Register Description	Offset Address
Channel 23	Reserved, set to 0	0x17C	Reserved, set to 0	0x1DC

DMA CONTROLLER

Table 190. Memory Map of Primary and Alternate DMA Structures[1] (Continued)

	Primary Structures		Alternate Structures	
Channel Number	Register Description	Offset Address	Register Description	Offset Address
	Control	0x178	Control	0x1D8
	Destination end pointer	0x174	Destination end pointer	0x1D4
	Source end pointer	0x170	Source end pointer	0x1D0
…	…	…	…	…
Channel 1	Reserved, set to 0	0x01C	Reserved, set to 0	0x11C
	Control	0x018	Control	0x118
	Destination end pointer	0x014	Destination end pointer	0x114
	Source end pointer	0x010	Source end pointer	0x110
Channel 0	Reserved, set to 0	0x00C	Reserved, set to 0	0x10C
	Control	0x008	Control	0x108
	Destination end pointer	0x004	Destination end pointer	0x104
	Source end pointer	0x000	Source end pointer	0x100

[1] The row with ellipses (…) indicates all channels between Channel 23 and Channel 1. These channels follow the same register naming contentions and offset address pattern.

The user must define the DMA structures in their source code, as shown in the Example Code: Define DMA Structures section. After the structure has been defined, its start address must be assigned to the DMA base address pointer register, PDBPTR. Each register for each DMA channel is then at the offset address (as specified in Table 190) plus the value in the PDBPTR register.

When the DMA controller receives a request for a channel, it reads the corresponding data structure from the system memory into its internal cache. Any update to the descriptor in the system memory until the DMA_DONE interrupt is received does not guarantee expected behavior. It is recommended that the user not update the descriptor before receiving DMA_DONE.

Example Code: Define DMA Structures

To define DMA structures, use the following code:

```
memset(dmaChanDesc,0x0,sizeof(dmaChanDesc));     // Set up the DMA base address pointer register.
uiBasPtr = (unsigned int)&dmaChanDesc;          // Set up the DMA base pointer.
pADI_DMA->CFG = 1;                 // Enable DMA controller
pADI_DMA->PDBPTR = uiBasPtr;
```

SOURCE DATA END POINTER

The SRC_END_PTR memory location stores the address of the last location from which data is read as part of a DMA transfer. This memory location must be programmed with the end address of the source data before the controller can perform a DMA transfer. The controller reads this memory location when it starts the first DMA data transfer. The DMA controller does not write to this memory location.

Table 191. Source Data End Pointer

Bits	Name	Description
[31:0]	SRC_END_PTR	The end address of the source data

DESTINATION DATA END POINTER

The DST_END_PTR memory location stores the address of the last location to which data is written as part of a DMA transfer. This memory location must be programmed with the end address of the destination data before the controller can perform a DMA transfer. The controller reads this memory location when it starts the first DMA data transfer. The DMA controller does not write to this memory location.

Table 192. Destination Data End Pointer

Bits	Name	Description
[31:0]	DST_END_PTR	The end address of the source data

DMA CONTROLLER

CONTROL DATA CONFIGURATION

For each DMA transfer, the CHNL_CFG memory location provides the control information for the DMA transfer to the controller.

Table 193. CHNL_CFG Control Data Configuration

Bit(s)	Name	Source Data Width	Setting	Description
[31:30]	DST_INC			Destination Address Increment. The address increment depends on the source data width.
		Byte		
			00	Source address increment is byte.
			01	Source address increment is half word.
			10	Source address increment is word.
			11	No increment. Address remains set to the value contained in the DST_END_PTR memory location.
		Half word		
			00	Reserved.
			01	Source address increment is half word.
			10	Source address increment is word.
			11	No increment. Address remains set to the value contained in the DST_END_PTR memory location.
		Word		
			00	Reserved.
			01	Reserved.
			10	Source address increment is word.
			11	No increment. Address remains set to the value contained in the DST_END_PTR memory location.
[29:28]	Reserved			Undefined. Write as zero.
[27:26]	SRC_INC			Source Address Increment. The address increment depends on the source data width.
		Byte		
			00	Source address increment is byte.
			01	Source address increment is half word.
			10	Source address increment is word.
			11	No increment. Address remains set to the value contained in the SRC_END_PTR memory location.
		Half word		
			00	Reserved.
			01	Source address increment is half word.
			10	Source address increment is word.
			11	No increment. Address remains set to the value contained in the SRC_END_PTR memory location.
		Word		
			00	Reserved.
			01	Reserved.
			10	Source address increment is word.
			11	No increment. Address remains set to the value contained in the SRC_END_PTR memory location.
[25:24]	SRC_SIZE			Size of the Source Data.
			00	Byte.
			01	Half word.
			10	Word.
			11	Reserved.
[23:18]	Reserved			Undefined. Write as 0.
[17:14]	R_POWER			DMA Transfers Before Rearbitration. Set these bits to control the number of DMA transfers can occur before the controller rearbitrates. These bits must be set to 0000 for all DMA transfers involving peripherals. Operation of the DMA is indeterminate if a value other than 0000 is programmed in this location for DMA transfers involving peripherals.
			0000	1
			0001	2

DMA CONTROLLER

Table 193. CHNL_CFG Control Data Configuration (Continued)

Bit(s)	Name	Source Data Width	Setting	Description
			0010	4
			0011	8
			0100	16
			0101	32
			0110	64
			0111	128
			1000	256
			1001	512
			1010 to 1111	1024
[13:4]	N_MINUS_1			Number of Configured Transfers Minus 1 for that Channel. The 10-bit value indicates the number of DMA transfers (not the total number of bytes) minus one.
			0x000	1 DMA transfer.
			0x001	2 DMA transfers.
			0x002	3 DMA transfers.
			…	
			0x3FF	1024 DMA transfers.
3	Reserved			Undefined. Write as 0.
[2:0]	CYCLE_CTRL			Transfer Types of DMA Cycle.
			000	Stop (invalid).
			001	Basic.
			010	Autorequest.
			011	Ping pong.
			100	Memory scatter gather primary.
			101	Memory scatter gather alternate.
			110	Peripheral scatter gather primary.
			111	Peripheral scatter gather alternate.

During the DMA transfer process, if any error occurs during the data transfer, CHNL_CFG is written back to the system memory, with the N_MINUS_1 bits updated to reflect the number of transfers yet to be completed. When a full DMA cycle is complete, the CYCLE_CTRL bits are made invalid to indicate the completion of the transfer.

DMA PRIORITY

The priority of a channel is determined by its number and priority level. Each channel can have two priority levels: default or high.

All channels at the high priority level have higher priority than channels at the default priority level. At the same priority level, a channel with a lower channel number has a higher priority. The DMA channel priority levels can be changed by writing to the appropriate bit in the PRI_SET register.

DMA TRANSFER TYPES

The DMA controller supports several types of DMA transfers. The various types are selected by programming the appropriate values into the CYCLE_CTRL bits (Bits[2:0]) in the CHNL_CFG location of the control data structure.

Invalid (CHNL_CFG, Bits[2:0] = 000)

CHNL_CFG, Bits[2:0] = 000 means that no DMA transfer is enabled for the channel. After the controller completes a DMA cycle, it sets the cycle type to invalid to prevent it from repeating the same DMA cycle.

DMA CONTROLLER

Basic (CHNL_CFG, Bits[2:0] = 001)

In basic mode, the controller can be configured to use either the primary or alternate data structure. The peripheral must present a request for every data transfer. After the channel is enabled, when the controller receives a request, it performs the following operations:

1. The controller performs a transfer. If the number of transfers remaining is zero, skip to Step 3.
2. The controller arbitrates. If a higher priority channel is requesting service, the controller services that channel. If the peripheral or software signals a request to the controller, the controller returns to Step 1.
3. At the end of the transfer, the controller generates the corresponding DMA_DONE channel interrupt in the NVIC.

Autorequest (CHNL_CFG, Bits[2:0] = 010)

When the controller operates in autorequest mode, it is only necessary for the controller to receive a single request to enable it to complete the entire DMA cycle. As such, a large data transfer can occur without significantly increasing the latency for servicing higher priority requests or requiring multiple requests from the processor or peripheral. Autorequest mode is very useful for a memory to memory copy application.

In autorequest mode, the controller can be configured to use either the primary or alternate data structure. After the channel is enabled, when the controller receives a request, it performs the following operations:

1. The controller performs a minimum ($2^{\text{CHNL\_CFG Bits[17:14]}}$, N) transfer for the channel, where N is the number of transfers. If the number of transfers remaining is zero, skip to Step 3.
2. A request for the channel is automatically generated. The controller arbitrates. If the channel has the highest priority, the DMA cycle returns to Step 1.
3. At the end of the transfer, the controller generates an interrupt for the corresponding DMA channel.

Ping Pong (CHNL_CFG, Bits[2:0] = 011)

In ping pong mode, the controller performs a DMA cycle using one of the data structures and then performs a DMA cycle using the other data structure. The controller continues to switch between using the primary and alternate data structures until it reads a data structure that is invalid, or the MCU disables the channel.

Ping pong mode is useful for transferring data using different buffers in a memory. In a typical application, the host must configure both primary and alternate data structures before starting the transfer. As the transfer progresses, the host can subsequently configure primary or alternate control data structures in the interrupt service routine when the corresponding transfer ends.

The DMA controller interrupts the MCU using the DMA_DONE interrupt after the completion of transfers associated with each control data structure. The individual transfers using either the primary or alternate control data structure work the same as a basic DMA transfer.

Software Ping Pong DMA Transfer (CHNL_CFG, Bits[2:0] = 011)

In this mode, if the DMA request comes from the software, a request is generated automatically after each arbitration cycle until the completion of primary or alternate descriptor tasks. This final descriptor must use an autorequest transfer type. This mode is shown in Figure 44.

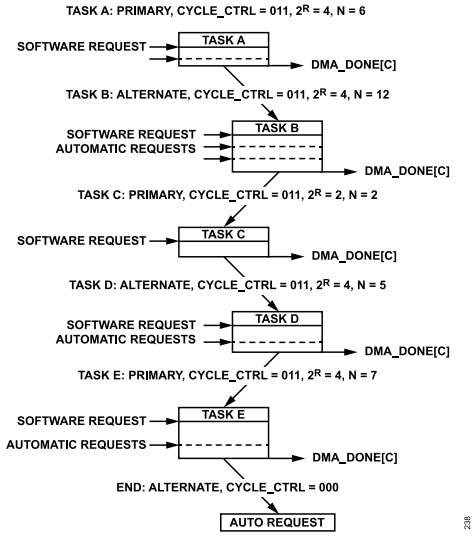

Figure 44. Software Ping Pong DMA Transfer

Peripheral Ping Pong DMA Transfer (CHNL_CFG, Bits[2:0] = 011)

In this mode, if the DMA request is from a peripheral, the peripheral must send DMA requests after every data transfer to complete primary or alternate descriptor tasks and the final descriptor must be programmed to use a basic transfer type. This mode is shown in Figure 45.

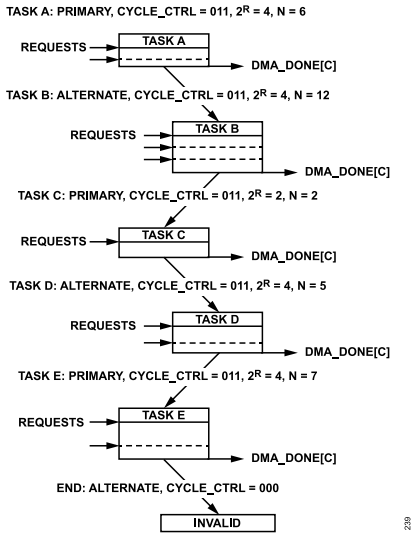

Figure 45. Peripheral Ping Pong DMA Transfer

Memory Scatter Gather (CHNL_CFG, Bits[2:0] = 100 or 101)

In memory scatter gather mode, the controller must be configured to use both the primary and alternate data structures. The controller uses the primary data structure to program the control configuration for the alternate data structure. The alternate data structure is used for actual data transfers, which are similar to an autorequest DMA transfer. The controller arbitrates after every primary transfer. The controller requires only one request to complete the entire transfer. This mode is used when performing multiple memory to memory copy tasks. The MCU can configure all of the tasks simultaneously and does not need to intervene in between each task. The controller generates the corresponding DMA channel interrupt in the NVIC when the entire scatter gather transaction completes using a basic cycle.

In memory scatter gather mode, the controller receives an initial request and then performs four DMA transfers using the primary data structure to program the control structure of the alternate data structure. After these transfers are completed, the controller starts a DMA cycle using the alternate data structure. After the cycle completes, the controller performs another four DMA transfers using the primary data structure.

DMA CONTROLLER

The controller continues to alternate between using the primary and alternate data structures until the processor configures the alternate data structure for a basic cycle or the DMA reads an invalid data structure.

Table 194 details the fields of the CHNL_CFG memory location for the primary data structure, which must be programmed with constant values for the memory scatter gather mode. This mode is also shown in Figure 46.

Table 194. CHNL_CFG for Primary Data Structure in Memory Scatter Gather Mode, CHNL_CFG, Bits[2:0] = 100

Bit(s)	Name	Description
[31:30]	DST_INC	Set to 10, configures the controller to use word increments for the address.
[29:28]	Reserved	Undefined. Write as 0.
[27:26]	SRC_INC	Set to 10, configures the controller to use word increments for the address.
[25:24]	SRC_SIZE	Set to 10, configures the controller to use word transfers.
[23:18]	Reserved	Undefined. Write as 0.
[17:14]	R_POWER	Set to 0010, indicates that the DMA controller is ready to perform four transfers.
[13:4]	N_MINUS_1	Configures the controller to perform N DMA transfers, where N is a multiple of four.
3	Reserved	Undefined. Write as 0.
[2:0]	CYCLE_CTRL	Set to 100, configures the controller to perform a memory scatter gather DMA cycle.

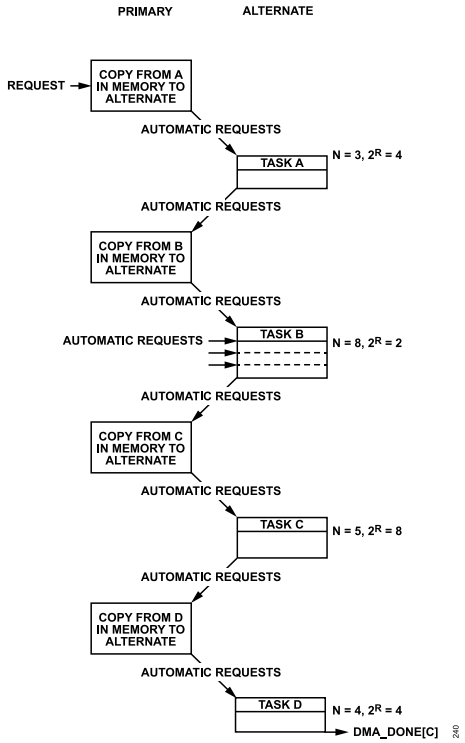

Figure 46. Memory Scatter Gather DMA Transfer

Peripheral Scatter Gather (CHNL_CFG, Bits[2:0] = 110 or 111)

In peripheral scatter gather mode, the controller must be configured to use both the primary and alternate data structures. The controller uses the primary data structure to program the control structure of the alternate data structure. The alternate data structure is used for actual data transfers, and each transfer takes place using the alternate data structure with a basic DMA transfer. The controller does not arbitrate after every primary transfer. The peripheral scatter gather mode is used when there are multiple peripheral to memory DMA tasks to be performed. The Cortex-M3 can configure all of the tasks simultaneously and does not need to intervene in between each task.

The peripheral scatter gather mode is very similar to the memory scatter gather mode except for the arbitration and request requirements. The MCU generates the corresponding DMA_DONE channel interrupt in the NVIC when the entire scatter gather transaction completes using a basic cycle.

DMA CONTROLLER

In peripheral scatter gather mode, the controller receives an initial request from a peripheral and then performs four DMA transfers using the primary data structure to program the alternate control data structure. The controller then immediately starts a DMA cycle using the alternate data structure without rearbitrating.

After this cycle completes, the controller rearbitrates, and if it receives a request from the peripheral that has the highest priority, the controller performs another four DMA transfers using the primary data structure. The controller then immediately starts a DMA cycle using the alternate data structure without rearbitrating. The controller continues to alternate between using the primary and alternate data structures until the processor configures the alternate data structure for a basic cycle, or the DMA reads an invalid data structure.

Table 195 lists the fields of the CHNL_CFG memory location for the primary data structure, which must be programmed with constant values for the peripheral scatter gather mode. This mode is shown in Figure 47.

Table 195. CHNL_CFG for Primary Data Structure in Peripheral Scatter Gather Mode, CHNL_CFG, Bits[2:0] = 110

Bit(s)	Name	Description
[31:30]	DST_INC	Set to 10, configures the controller to use word increments for the address.
[29:28]	Reserved	Undefined. Write as 0.
[27:26]	SRC_INC	Set to 10, configures the controller to use word increments for the address.
[25:24]	SRC_SIZE	Set to 10, configures the controller to use word transfers.
[23:18]	Reserved	Undefined. Write as 0.
[17:14]	R_POWER	Set to 0010, indicates that the DMA controller performed four transfers without rearbitration.
[13:4]	N_MINUS_1	Configures the controller to perform N DMA transfers, where N is a multiple of four.
3	Reserved	Undefined. Write as 0.
[2:0]	CYCLE_CTRL	Set to 110, configures the controller to perform a peripheral scatter gather DMA cycle.

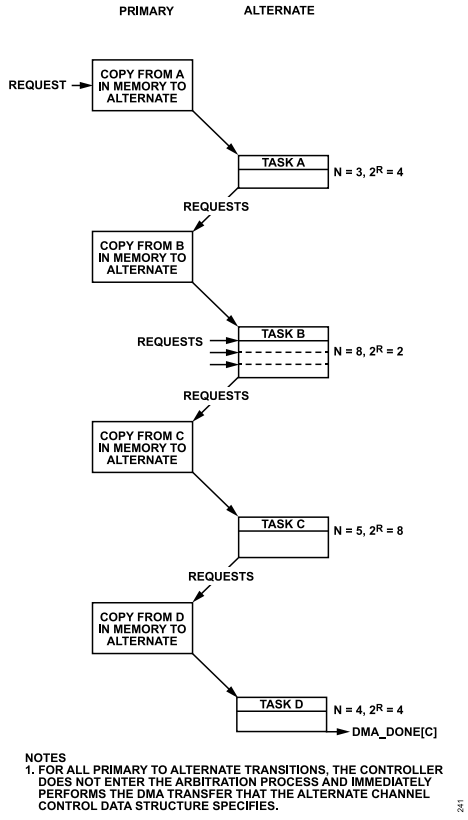

Figure 47. Peripheral Scatter Gather DMA Transfer

DMA CONTROLLER

DMA INTERRUPTS AND EXCEPTIONS

Error Management

The DMA controller generates an error interrupt to the core when a DMA error occurs. A DMA error can occur due to a bus error or an invalid descriptor fetch. A bus error can occur when fetching the descriptor or performing a data transfer. A bus error can occur when a read from or a write to a reserved address location occurs.

When a bus error occurs, the faulty channel is automatically disabled, and the corresponding status bit in the ERRCHNL_CLR register is set. If the DMA error is enabled in the NVIC, the error also generates an interrupt. In addition, the CHNL_CFG data structure for the corresponding channel is updated with the latest N count. The user can check the N count to determine how many data transfers occurred before the bus error.

When the controller fetches an invalid descriptor, the faulty channel is automatically disabled, and the corresponding status bit in the INVALIDDESC_CLR register is set. If the DMA error is enabled in NVIC, the error also generates an interrupt.

Address Calculation

The DMA controller calculates the source read address based on the content of SRC_END_PTR, the source address increment setting in CHNL_CFG, and the current value of CHNL_CFG, Bits[13:4] (N_MINUS_1). Similarly, the destination write address is calculated based on the content of DST_END_PTR, the destination address increment setting in CHNL_CFG, and the current value of CHNL_CFG, Bits[13:4]. In the following code examples, N_MINUS_1 is the number of configured transfers minus 1 for that channel:

```
Source Read Address = SRC_END_PTR - (N_MINUS_1 << (SRC_INC)) for SRC_INC = 0, 1, 2
Source Read Address = SRC_END_PTR for SRC_INC = 3
Destination Write Address = DST_END_PTR - (N_MINUS_1 << (DST_INC)) for DST_INC = 0, 1, 2
Destination Write Address = DST_END_PTR for DST_INC = 3
```

Address Decrement

The address decrement can be enabled for source and destination addresses. Source address decrement can be enabled for channels by setting the appropriate bits in the SRCADDR_SET register. Similarly, destination address decrement can be enabled for channels by setting the required bits in the DSTADDR_SET register. The values written into the source data end pointer (SRC_END_PTR) and destination data end pointer (DST_END_PTR) are still used as the addresses for the last transfer as part of the DMA cycle. However, the start address is computed differently than the address increment scheme for either source read or destination write.

In the following calculations, N_MINUS_1 is the current count of transfers to be completed. Additionally, byte swap and address decrement must not be used together for any channel. If used together, the DMA data transfer operation is unpredictable. Figure 48 shows all the combinations of source and destination decrementing and their data movement direction.

If the source decrement bit is set in the SRCADDR_SET register for a channel, its source address is calculated as follows:

```
Source Read Address = SRC_END_PTR + (N_MINUS_1 << (SRC_INC)) for SRC_INC = 0, 1, 2
```

or

```
Source Read Address = SRC_END_PTR for SRC_INC = 3
```

If the destination decrement bit is set in the DSTADDR_SET register for a channel, its source address is computed as follows:

```
Destination Write Address = DST_END_PTR + (N_MINUS_1 << (DST_INC)) for DST_INC = 0, 1, 2
```

or

```
Destination Write Address = DST_END_PTR for DST_INC = 3
```

DMA CONTROLLER

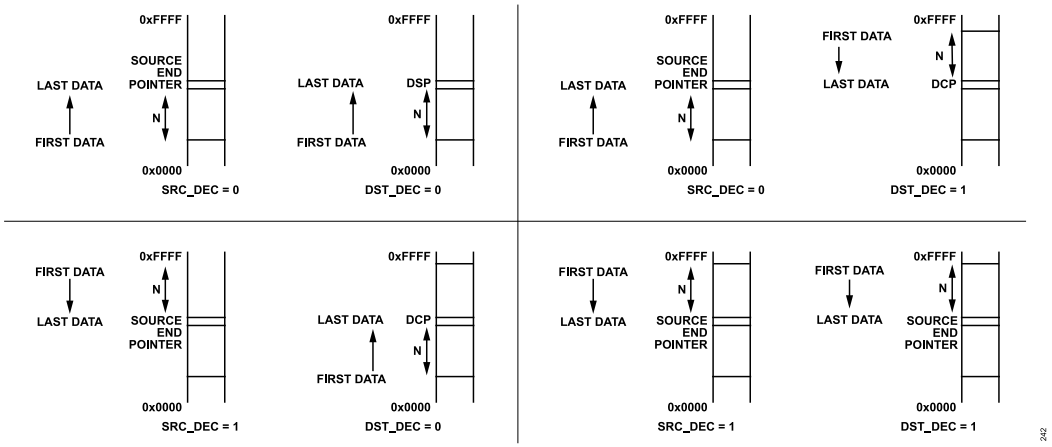

Figure 48. Image Decrement

Aborting DMA Transfers

It is possible to abort a DMA transfer that is in progress by writing to the bit in the EN_CLR register that corresponds to the channel that must be aborted. Do not set CFG to 0, because this action can corrupt the DMA structures.

ENDIAN OPERATION

The DMA controller performs a transfer by default by using a little endian approach. However, this default behavior can be changed by setting the corresponding channel bit in the BS_SET register. The endian operation is referred to as byte swap.

Byte Swap Disabled

Byte swap is disabled by default, in which case, the data transfer is considered to be little endian. Data arriving from a peripheral is placed in sequence starting from the LSB of a 32-bit word. For example, if 16 bytes of data arrive at the SPI as 0x01 (start), 0x02, 0x03, 0x04 … 0x0F, 0x10, it is stored by the DMA in memory as follows:

```
04_03_02_01
08_07_06_05
0C_0B_0A_09
10_0F_0E_0D
```

Byte Swap Enabled

Byte swap happens on 32-bit data boundaries. The transfer size must be a multiple of four. Byte swap and address decrement cannot be used together for any channel. If used together, DMA data transfer operation is unpredictable. When using byte swap, ensure that the source data address is constant for the full data transfer. Byte swap functionality is independent of DMA transfer size and can be 8-bit, 16-bit, or 32-bit.

If 16 bytes of data arrive at the SPI as 0x01(start), 0x02, 0x03, 0x04 … 0x0F, 0x10, the data is stored by the DMA in memory as follows:

```
01_02_03_04
05_06_07_08
09_0A_0B_0C
0D_0E_0F_10
```

DMA CHANNEL ENABLE AND DISABLE

Before issuing a DMA request, the DMA channel must be enabled. Otherwise, the DMA request for the corresponding channel is driven as a DMA_DONE interrupt. Any DMA channel can be enabled by writing to the corresponding bit in the EN_SET register. The DMA controller disables the channel when the corresponding DMA_DONE interrupt is generated. However, the user can disable any channel by writing to the corresponding bit in the EN_CLR register.

DMA CONTROLLER

Whenever a channel is disabled, based on the current state of the DMA controller, the channel does one of the following:

- If the user disables the channel and there is no request pending for that channel, it is disabled immediately.
- If the user disables the channel that is not being serviced, but its request is posted, its pending request is cleared, and the channel is disabled immediately.
- If the user disables a channel that has been selected after arbitration but has yet to start transfers, the controller completes the arbitration cycle and then disables the channel.
- If the user disables the channel when it is being serviced, the controller completes the current arbitration cycle.

DMA INITIATOR ENABLE

CFG, Bit 0 acts as a soft reset to the DMA controller. Any activity in the DMA controller can be performed only when this bit is set to 1. Clearing this bit to 0 clears all cached descriptors within the controller and resets the controller.

POWER-DOWN CONSIDERATIONS

Complete all ongoing DMA transfers before powering down the chip to hibernate mode. However, if the user decides to hibernate as quickly as possible (current data transfers are ignored), the DMA controller must be disabled by clearing the CFG, Bit 0 before entering hibernate mode. If hibernate mode is selected when a DMA transfer is in progress, the transfer discontinues. The DMA returns to the disabled state. After hibernate or a POR, the DMA must be enabled again by setting the CFG, Bit 0.

The following DMA registers are retained in hibernate mode:

- PDBPTR
- ADBPTR
- RMSK_SET
- RMSK_CLR
- PRI_SET
- PRI_CLR
- BS_SET
- BS_CLR
- SRCADDR_SET
- SRCADDR_CLR
- DSTADDR_SET
- DSTADDR_CLR

REGISTER SUMMARY: DMA

Table 196. DMA Register Summary

Address	Name	Description	Reset	Access
0x40010000	STAT	Status	0x00180000	R
0x40010004	CFG	Configuration	0x00000000	W
0x40010008	PDBPTR	Channel primary control data base pointer	0x00000000	R/W
0x4001000C	ADBPTR	Channel alternate control data base pointer	0x00000200	R
0x40010014	SWREQ	Channel software request	0x00000000	W
0x40010020	RMSK_SET	Channel request mask set	0x00000000	R/W
0x40010024	RMSK_CLR	Channel request mask clear	0x00000000	W
0x40010028	EN_SET	Channel enable set	0x00000000	R/W
0x4001002C	EN_CLR	Channel enable clear	0x00000000	W
0x40010030	ALT_SET	Channel primary alternate set	0x00000000	R/W
0x40010034	ALT_CLR	Channel primary alternate clear	0x00000000	W
0x40010038	PRI_SET	Channel priority set	0x00000000	W
0x4001003C	PRI_CLR	Channel priority clear	0x00000000	W
0x40010048	ERR_CLR	Bus error clear	0x00000000	R/W
0x4001004C	ERRCHNL_CLR	Per channel bus error	0x00000000	R/W
0x40010050	INVALIDDESC_CLR	Per channel invalid descriptor clear	0x00000000	R/W
0x40010800	BS_SET	Channel bytes swap enable set	0x00000000	R/W
0x40010804	BS_CLR	Channel bytes swap enable clear	0x00000000	W
0x40010810	SRCADDR_SET	Channel source address decrement enable set	0x00000000	R/W
0x40010814	SRCADDR_CLR	Channel source address decrement enable clear	0x00000000	W
0x40010818	DSTADDR_SET	Channel destination address decrement enable set	0x00000000	R/W
0x4001081C	DSTADDR_CLR	Channel destination address decrement enable clear	0x00000000	W
0x400C2008	FIFO_CON	FIFO configuration	0x00001010	R/W
0x400C206C	DATAFIFORD	Data FIFO read	0x00000000	R

REGISTER DETAILS: DMA

STATUS REGISTER

Address: 0x40010000, Reset: 0x00180000, Name: STAT

Table 197. Bit Descriptions for STAT

Bits	Bit Name	Settings	Description	Reset	Access
[31:21]	Reserved		Reserved.	0x000	R
[20:16]	CHANM1		Number of Available DMA Channels Minus 1. With 24 channels available, the register reads back 0x17.	0x18	R
[15:1]	Reserved		Reserved.	0x000	R
0	MEN		Enable Status Of The Controller.	0x0	R
		0	Controller is disabled.		
		1	Controller is enabled.		

CONFIGURATION REGISTER

Address: 0x40010004, Reset: 0x00000000, Name: CFG

Table 198. Bit Descriptions for CFG

Bits	Bit Name	Settings	Description	Reset	Access
[31:1]	Reserved		Reserved.	0x0000000	R
0	MEN		Controller Enable.	0x0	W
		0	Disable controller.		
		1	Enable controller.		

CHANNEL PRIMARY CONTROL DATA BASE POINTER REGISTER

Address: 0x40010008, Reset: 0x00000000, Name: PDBPTR

The PDBPTR register must be programmed to point to the channel primary control data base pointer in the system memory. The amount of system memory that must be assigned to the DMA controller depends on the number of DMA channels used and whether the alternate channel control data structure is used. This register cannot be read when the DMA controller is in the reset state.

Table 199. Bit Descriptions for PDBPTR

Bits	Bit Name	Settings	Description	Reset	Access
[31:0]	ADDR		Pointer to the Base Address of the Primary Data Structure. 5 + $\log_2$M LSBs are reserved and must be written as 0, where M is number of channels.	0x00000000	R/W

CHANNEL ALTERNATE CONTROL DATA BASE POINTER REGISTER

Address: 0x4001000C, Reset: 0x00000200, Name: ADBPTR

The ADBPTR read only register returns the base address of the alternate channel control data structure. This register removes the necessity for application software to calculate the base address of the alternate data structure. This register cannot be read when the DMA controller is in the reset state.

Table 200. Bit Descriptions for ADBPTR

Bits	Bit Name	Settings	Description	Reset	Access
[31:0]	ADDR		Base Address of the Alternate Data Structure.	0x200	R

CHANNEL SOFTWARE REQUEST REGISTER

Address: 0x40010014, Reset: 0x00000000, Name: SWREQ

The SWREQ register enables the generation of a software DMA request. Each bit of the register represents the corresponding channel number in the DMA controller. M is the number of DMA channels.

Table 201. Bit Descriptions for SWREQ

Bits	Bit Name	Settings	Description	Reset	Access
[31:24]	Reserved		Reserved.	0x00	R

REGISTER DETAILS: DMA

Table 201. Bit Descriptions for SWREQ (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
[23:0]	CHAN		Generate Software Request. Set the appropriate bit to generate a software DMA request on the corresponding DMA channel. Bit 0 corresponds to DMA Channel 0. Bit M − 1 corresponds to the DMA Channel M – 1. These bits are automatically cleared by the hardware after the corresponding software request completes.	0x000000	W
		0	DMA request is not created for Channel C.		
		1	DMA request is created for Channel C.		

CHANNEL REQUEST MASK SET REGISTER

Address: 0x40010020, Reset: 0x00000000, Name: RMSK_SET

Table 202. Bit Descriptions for RMSK_SET

Bits	Bit Name	Settings	Description	Reset	Access
[31:24]	Reserved		Reserved.	0x00	R
[23:0]	CHAN		Mask Requests from DMA Channels. This register disables DMA requests from peripherals. Each bit of the register represents the corresponding channel number in the DMA controller. Set the appropriate bit to mask the request from the corresponding DMA channel. Bit 0 corresponds to DMA Channel 0. Bit M – 1 corresponds to DMA Channel M – 1.	0x000000	R/W
		0	When read as 0, requests are enabled for Channel C. When written as 0, no effect.		
		1	When read as 1, requests are disabled for Channel C. When written as 1, peripherals associated with Channel C are disabled from generating DMA requests.		

CHANNEL REQUEST MASK CLEAR REGISTER

Address: 0x40010024, Reset: 0x00000000, Name: RMSK_CLR

Table 203. Bit Descriptions for RMSK_CLR

Bits	Bit Name	Settings	Description	Reset	Access
[31:24]	Reserved		Reserved.	0x00	R
[23:0]	CHAN		Clear Request Mask Set Bits. This register enables DMA requests from peripherals by clearing the mask set in the RMSK_SET register. Each bit of the register represents the corresponding channel number in the DMA controller. Set the appropriate bit to clear the corresponding bit in RMSK_SET, Bits[23:0]. Bit 0 corresponds to DMA Channel 0. Bit M – 1 corresponds to DMA Channel M – 1.	0x000000	W
		0	No effect. Use the RMSK_SET register to disable DMA requests.		
		1	Peripherals associated with Channel C are enabled to generate DMA requests.		

CHANNEL ENABLE SET REGISTER

Address: 0x40010028, Reset: 0x00000000, Name: EN_SET

Table 204. Bit Descriptions for EN_SET

Bits	Bit Name	Settings	Description	Reset	Access
[31:24]	Reserved		Reserved.	0x00	R
[23:0]	CHAN		Enable DMA Channels. This register allows the enabling of DMA channels. Reading the register returns the enable status of the channels. Each bit of the register represents the corresponding channel number in the DMA controller. Set the appropriate bit to enable the corresponding channel. Bit 0 corresponds to DMA Channel 0. Bit M – 1 corresponds to DMA Channel M – 1.	0x000000	R/W
		0	When read as 0, Channel C is disabled. When written as 0, no effect. Use the EN_CLR register to disable this channel.		
		1	When read as 1, Channel C is enabled. When written as 1, Channel C is enabled.		

REGISTER DETAILS: DMA

CHANNEL ENABLE CLEAR REGISTER

Address: 0x4001002C, Reset: 0x00000000, Name: EN_CLR

Table 205. Bit Descriptions for EN_CLR

Bits	Bit Name	Settings	Description	Reset	Access
[31:24]	Reserved		Reserved.	0x00	R
[23:0]	CHAN		Disable DMA Channels. This register allows the disabling of DMA channels. This register is write only. Each bit of the register represents the corresponding channel number in the DMA controller. The controller disables a channel automatically by setting the appropriate bit when it completes the DMA cycle. Set the appropriate bit to disable the corresponding channel. Bit 0 corresponds to DMA Channel 0. Bit M – 1 corresponds to DMA Channel M – 1.	0x000000	W
		0	No effect. Use the EN_SET register to enable the channel.		
		1	Disables Channel C.		

CHANNEL PRIMARY ALTERNATE SET REGISTER

Address: 0x40010030, Reset: 0x00000000, Name: ALT_SET

The ALT_SET register enables the user to configure the appropriate DMA channel to use the alternate control data structure. Reading the register returns the status of which data structure is in use for the corresponding DMA channel. Each bit of the register represents the corresponding channel number in the DMA controller. The DMA controller sets and clears these bits automatically as necessary for ping pong, memory scatter gather, and peripheral scatter gather transfers.

Table 206. Bit Descriptions for ALT_SET

Bits	Bit Name	Settings	Description	Reset	Access
[31:24]	Reserved		Reserved.	0x00	R
[23:0]	CHAN		Control Structure Status and Select Alternate Structure. Returns the channel control data structure status or selects the alternate data structure for the corresponding DMA channel. Bit 0 corresponds to DMA Channel 0. Bit M – 1 corresponds to DMA Channel M – 1.	0x000000	R/W
		0	When read as 0, DMA Channel C uses the primary data structure. When written as 0, no effect. Use the ALT_CLR register to set Channel C to 0.		
		1	When read as 1, DMA Channel C uses the alternate data structure. When written as 1, the alternate data structure for Channel C is selected.		

CHANNEL PRIMARY ALTERNATE CLEAR REGISTER

Address: 0x40010034, Reset: 0x00000000, Name: ALT_CLR

The ALT_CLR write only register enables the user to configure the appropriate DMA channel to use the primary control data structure. Each bit of the register represents the corresponding channel number in the DMA controller. The DMA controller sets and clears these bits automatically as necessary for ping pong, memory scatter gather, and peripheral scatter gather transfers.

Table 207. Bit Descriptions for ALT_CLR

Bits	Bit Name	Settings	Description	Reset	Access
[31:24]	Reserved		Reserved.	0x00	R
[23:0]	CHAN		Select Primary Data Structure. Set the appropriate bit to select the primary data structure for the corresponding DMA channel. Bit 0 corresponds to DMA Channel 0. Bit M – 1 corresponds to DMA Channel M – 1.	0x000000	W
		0	No effect. Use the ALT_SET register to select the alternate data structure.		
		1	Selects the primary data structure for Channel C.		

CHANNEL PRIORITY SET REGISTER

Address: 0x40010038, Reset: 0x00000000, Name: PRI_SET

Table 208. Bit Descriptions for PRI_SET

Bits	Bit Name	Settings	Description	Reset	Access
[31:24]	Reserved		Reserved.	0x00	R

REGISTER DETAILS: DMA

Table 208. Bit Descriptions for PRI_SET (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
[23:0]	CHAN		Configure Channel Priority. This register enables the user to configure a DMA channel to use the high priority level. Reading the register returns the status of the channel priority mask. Each bit of the register represents the corresponding channel number in the DMA controller. This register returns the channel priority mask status or sets the channel priority to high. Bit 0 corresponds to DMA Channel 0. Bit M – 1 corresponds to DMA Channel M – 1.	0x000000	W
		0	When read as 0, DMA Channel C uses the default priority level. When written as 0, no effect. Use the PRI_CLR register to set Channel C to the default priority level.		
		1	DMA Channel C uses a high priority level.		

CHANNEL PRIORITY CLEAR REGISTER

Address: 0x4001003C, Reset: 0x00000000, Name: PRI_CLR

Table 209. Bit Descriptions for PRI_CLR

Bits	Bit Name	Settings	Description	Reset	Access
[31:24]	Reserved		Reserved.	0x00	R
[23:0]	CHPRICLR		Configure Channel for Default Priority Level. This write only register enables the user to configure a DMA channel to use the default priority level. Each bit of the register represents the corresponding channel number in the DMA controller. Set the appropriate bit to select the default priority level for the specified DMA channel. Bit 0 corresponds to DMA Channel 0. Bit M – 1 corresponds to DMA Channel M – 1.	0x000000	W
		0	No effect. Use the PRI_SET register to set Channel C to the high priority level.		
		1	Channel C uses the default priority level.		

BUS ERROR CLEAR REGISTER

Address: 0x40010048, Reset: 0x00000000, Name: ERR_CLR

Table 210. Bit Descriptions for ERR_CLR

Bits	Bit Name	Settings	Description	Reset	Access
[31:24]	Reserved		Reserved.	0x00	R
[23:0]	CHAN		Bus Error Status. This register is used to read and clear the DMA bus error status. The error status is set if the controller encountered a bus error while performing a transfer or when it reads an invalid descriptor (whose cycle control is 0b000). If a bus error occurs or an invalid cycle control is read on a channel, that channel is automatically disabled by the controller. The other channels are unaffected. Write 1 to clear bits.	0x000000	R/W1C
		0	When read, no bus error or an invalid cycle control has occurred. When written, no effect.		
		1	When read, a bus error or invalid cycle control is pending. When written, bit is cleared.		

PER CHANNEL BUS ERROR REGISTER

Address: 0x4001004C, Reset: 0x00000000, Name: ERRCHNL_CLR

Table 211. Bit Descriptions for ERRCHNL_CLR

Bits	Bit Name	Settings	Description	Reset	Access
[31:24]	Reserved		Reserved.	0x00	R
[23:0]	CHAN		Per Channel Bus Error Status and Per Channel Bus Error Clear. This register is used to read and clear the per channel DMA bus error status. The error status is set if the controller encountered a bus error while performing a transfer. If a bus error occurs on a channel, that channel is automatically disabled by the controller. The other channels are unaffected. Write 1 to clear bits.	0x000000	R/W1C
		0	When read as 0, no bus error has occurred. When written as 0, no effect.		
		1	When read as 1, a bus error control is pending. When written as 1, bit is cleared.		

REGISTER DETAILS: DMA

PER CHANNEL INVALID DESCRIPTOR CLEAR REGISTER

Address: 0x40010050, Reset: 0x00000000, Name: INVALIDDESC_CLR

Table 212. Bit Descriptions for INVALIDDESC_CLR

Bits	Bit Name	Settings	Description	Reset	Access
[31:24]	Reserved		Reserved.	0x00	R
[23:0]	CHAN		Per Channel Invalid Descriptor Status and Per Channel Invalid Descriptor Status Clear. This register is used to read and clear the per channel DMA invalid descriptor status. The per channel invalid descriptor status is set if the controller reads an invalid descriptor (whose cycle control is 0b000). If the controller reads invalid cycle control for a channel, that channel is automatically disabled by the controller. The other channels are unaffected. Write 1 to clear bits.	0x000000	R/W1C
		0	When read as 0, no invalid cycle control has occurred. When written as 0, no effect.		
		1	When read as 1, an invalid cycle control is pending. When written as 1, bit is cleared.		

CHANNEL BYTES SWAP ENABLE SET REGISTER

Address: 0x40010800, Reset: 0x00000000, Name: BS_SET

Table 213. Bit Descriptions for BS_SET

Bits	Bit Name	Settings	Description	Reset	Access
[31:24]	Reserved		Reserved.	0x00	R
[23:0]	CHAN		Byte Swap Status. This register is used to configure a DMA channel to use byte swap. Each bit of the register represents the corresponding channel number in the DMA controller. Bit 0 corresponds to DMA Channel 0. Bit M − 1 corresponds to DMA Channel M – 1.	0x000000	R/W
		0	When read as 0, Channel C byte swap is disabled. When written as 0, no effect.		
		1	When read as 1, Channel C byte swap is enabled. When written as 1, byte swap on Channel C is enabled.		

CHANNEL BYTES SWAP ENABLE CLEAR REGISTER

Address: 0x40010804, Reset: 0x00000000, Name: BS_CLR

Table 214. Bit Descriptions for BS_CLR

Bits	Bit Name	Settings	Description	Reset	Access
[31:24]	Reserved		Reserved.	0x00	R
[23:0]	CHAN		Disable Byte Swap. The BS_CLR write only register enables the user to configure a DMA channel to not use byte swapping and use the default operation. Each bit of the register represents the corresponding channel number in the DMA controller. Bit 0 corresponds to DMA Channel 0. Bit M − 1 corresponds to DMA Channel M – 1.	0x000000	W
		0	No effect. Use the BS_SET register to enable byte swap on Channel C.		
		1	Disables byte swap on Channel C.		

CHANNEL SOURCE ADDRESS DECREMENT ENABLE SET REGISTER

Address: 0x40010810, Reset: 0x00000000, Name: SRCADDR_SET

Table 215. Bit Descriptions for SRCADDR_SET

Bits	Bit Name	Settings	Description	Reset	Access
[31:24]	Reserved		Reserved.	0x00	R
[23:0]	CHAN		Source Address Decrement Status and Configure Source Address Decrement. The SRCADDR_SET register is used to configure the source address of a DMA channel to decrement the address instead of incrementing the address after each access. Each bit of the register represents the corresponding channel number in the DMA controller. Bit 0 corresponds to DMA Channel 0. Bit M − 1 corresponds to DMA Channel M – 1.	0x000000	R/W
		0	When read as 0, Channel C source address decrement is disabled. When written as 0, no effect. Use the SRCADDR_CLR register to disable source address decrement on Channel C.		

REGISTER DETAILS: DMA

Table 215. Bit Descriptions for SRCADDR_SET (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		1	When read as 1, Channel C source address decrement is enabled. When written as 1, source address decrement on Channel C is enabled.		

CHANNEL SOURCE ADDRESS DECREMENT ENABLE CLEAR REGISTER

Address: 0x40010814, Reset: 0x00000000, Name: SRCADDR_CLR

Table 216. Bit Descriptions for SRCADDR_CLR

Bits	Bit Name	Settings	Description	Reset	Access
[31:24]	Reserved		Reserved.	0x00	R
[23:0]	CHAN		Disable Source Address Decrement. This register enables the user to configure a DMA channel to use the default source address in increment mode. Each bit of the register represents the corresponding channel number in the DMA controller. Bit 0 corresponds to DMA Channel 0. Bit M − 1 corresponds to DMA Channel M – 1.	0x000000	W
		0	No effect. Use the SRCADDR_SET register to enable source address decrement on Channel C.		
		1	Disables address source decrement on Channel C.		

CHANNEL DESTINATION ADDRESS DECREMENT ENABLE SET REGISTER

Address: 0x40010818, Reset: 0x00000000, Name: DSTADDR_SET

Table 217. Bit Descriptions for DSTADDR_SET

Bits	Bit Name	Settings	Description	Reset	Access
[31:24]	Reserved		Reserved.	0x00	R
[23:0]	CHAN		Destination Address Decrement Status and Configure Destination Address Decrement. The DSTADDR_SET register is used to configure the destination address of a DMA channel to decrement the address instead of incrementing the address after each access. Each bit of the register represents the corresponding channel number in the DMA controller. Bit 0 corresponds to DMA Channel 0. Bit M − 1 corresponds to DMA Channel M – 1.	0x000000	R/W
		0	When read as 0, Channel C destination address decrement is disabled. When written as 0, no effect. Use the DSTADDR_CLR register to disable destination address decrement on Channel C.		
		1	When read as 1, Channel C destination address decrement is enabled. When written as 1, destination address decrement on Channel C is enabled.		

CHANNEL DESTINATION ADDRESS DECREMENT ENABLE CLEAR REGISTER

Address: 0x4001081C, Reset: 0x00000000, Name: DSTADDR_CLR

Table 218. Bit Descriptions for DSTADDR_CLR

Bits	Bit Name	Settings	Description	Reset	Access
[31:24]	Reserved		Reserved.	0x00	R
[23:0]	CHAN		Disable Destination Address Decrement. This register enables the user to configure a DMA channel to use the default destination address in increment mode. Each bit of the register represents the corresponding channel number in the DMA controller. Bit 0 corresponds to DMA Channel 0. Bit M − 1 corresponds to DMA Channel M – 1.	0x000000	W
		0	No effect. Use the DSTADDR_SET register to enable destination address decrement on Channel C.		
		1	Disables destination address decrement on Channel C.		

FIFO CONFIGURATION REGISTER

Address: 0x400C2008, Reset: 0x00001010, Name: FIFOCON

Table 219. Bit Descriptions for FIFOCON

Bits	Bit Name	Settings	Description	Reset	Access
[31:16]	Reserved		Reserved.	0x0	R

REGISTER DETAILS: DMA

Table 219. Bit Descriptions for FIFOCON (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
[15:13]	DATAFIFOSRCSEL		Selects the Source for the Data FIFO.	0x0	R/W
		000, 001, 110, 111	ADC data. ADC data is output of sinc3 filter.		
		010	DFT data, 18-bit real part and 18-bit imaginary part.		
		011	Sinc2 output.		
		100	Statistic variance output.		
		101	Statistics mean result.		
12	DATAFIFODMAREQEN		Enable Data FIFO DMA Channel.	0x1	R/W
		0	Disable DMA requests for data FIFO.		
		1	Enable DMA requests for data FIFO.		
11	DATAFIFOEN		Data FIFO Enable.	0x0	R/W
		0	FIFO is reset. No data transfers may take place. Sets the read and write pointers to the default values (empty FIFO). Status indicates if FIFO is empty.		
		1	Normal operation. FIFO is not reset.		
[10:0]	Reserved		Reserved.	0x10	R

DATA FIFO READ REGISTER

Address: 0x400C206C, Reset: 0x00000000, Name: DATAFIFORD

Table 220. Bit Descriptions for DATAFIFORD

Bits	Bit Name	Settings	Description	Reset	Access
[31:25]	ECC		ECC of Lower 25 Bits.	0x0	R
[24:18]	Reserved		Reserved.	0x0	R
[17:0]	DATAFIFOOUT		Data FIFO Read. If data FIFO is empty, a read of this register returns 0x00000000.	0x0	R

FLASH CONTROLLER

FLASH CONTROLLER FEATURES

The ADuCM355 processor includes 128 kB of embedded flash memory available for access through the flash controller. The embedded flash has a 72-bit wide data bus, providing two 32-bit words of data and one corresponding 8-bit ECC byte per access.

The flash controller is coupled with a cache controller module that provides two Arm high speed bus (AHB) ports: one port for reading data (DCode), and the other for reading instructions (ICode). A prefetch mechanism is implemented in the flash controller to optimize ICode read performance.

Flash writes are supported by a keyhole mechanism through Arm peripheral bus (APB) writes to memory mapped registers. The flash controller includes support for DMA-based keyhole writes.

FLASH CONTROLLER OVERVIEW

Key flash controller features include the following:

- The prefetch buffer provides optimal performance when reading consecutive addresses on the ICode interface.
- Simultaneous ICode and DCode read accesses. DCode has priority on contention. Simultaneous reads are possible if ICode returns buffered data from prefetch.
- DMA based keyhole writes, including address autoincrement for sequential accesses.
- ECC for error detection and correction. Errors and corrections may be reported as bus errors on the ICode or DCode bus as interrupts or ignored.

SUPPORTED COMMANDS

The following is a summary of the commands supported by the flash controller:

- Read. Supported through the ICode and DCode interfaces.
- Write. Provided by a keyhole mechanism through memory mapped registers.
- Mass erase. Clears all user data and program code.
- Page erase. Clears user data or program code from a 2 kB page in flash.
- Signature. Generates and verifies signatures for any set of contiguous whole pages of user data or program code.
- Abort. Terminates a command in progress.

PROTECTION AND INTEGRITY FEATURES

Protection and integrity features include the following:

- A fixed user key is required for running protected commands including mass erase and page erase.
- Optional and user definable write protection for user accessible memory.
- Optional 8-bit ECC. Enabled by user code, off by default.

FLASH CONTROLLER OPERATION

A flash block of 128 kB is available for user code and data, from 0 to 0x1FFFF. It is not possible to execute flash while erasing a flash page at the same time. The flash provides a 64-bit data bus, plus eight bits for corresponding ECC metadata. The memory is organized as a number of pages, each 2 kB in size, as well as 256 bytes reserved for ECC. See Figure 49 for details on flash, and Figure 50 for a full over overview of the flash page structure.

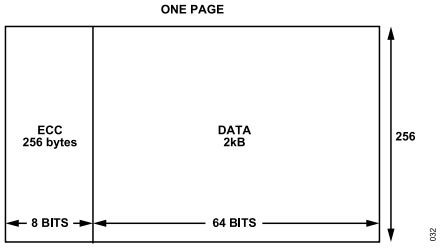

Figure 49. ADuCM355 Flash Page Structure

FLASH CONTROLLER

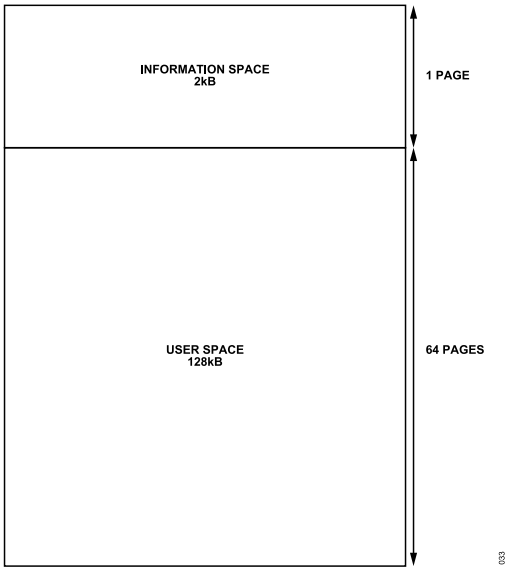

Figure 50. ADuCM355 Flash Memory Structure

FLASH MEMORY STRUCTURE

The memory pages are categorically divided into two sections: information space and user space. Total device storage is generally described as the size of user space alone.

Information Space

Information space is reserved for use by Analog Devices and generally stores a bootloader (kernel), several trim and calibration values, and other device specific metadata. All but the top 128 bytes of information space are readable by user code, but attempted erasures and writes are denied. Bus errors are also generated if user code attempts to read the protected range of information space. Other read only metadata can be made available to the user within the scope of the information space, but it is software defined. As such, the addresses and data types are not defined by the flash controller.

Besides the top 128 bytes of protected space, the remainder of the information space is freely readable by user code. Information space cannot be programmed or erased by the user, and these commands are denied. User code can perform a mass erase command on the ADuCM355 without affecting the content of information space. This mass erase provides a mechanism to upload new user firmware and data to a device without affecting the bootloader. User accessible information space is shown in Figure 51.

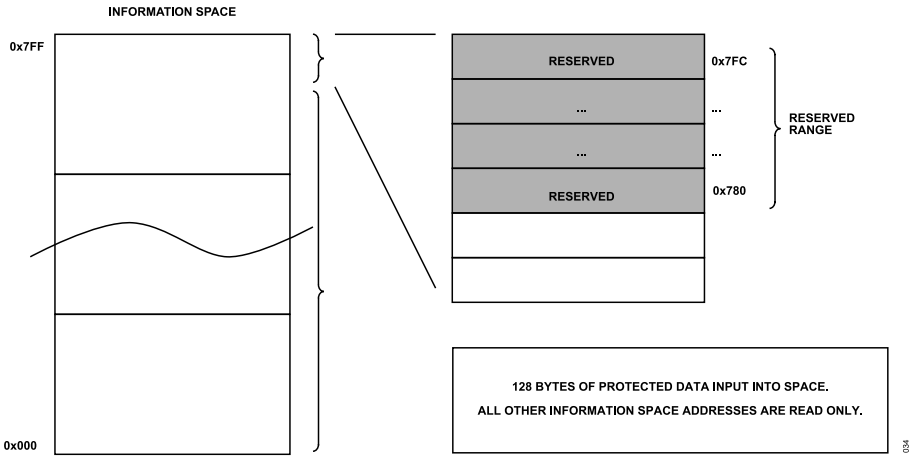

Figure 51. Information Space User Accessible Area

FLASH CONTROLLER

User Space Metadata

User space is the portion of flash memory intended for user data and program code. Several small address ranges in user space are used by the flash controller as metadata to enable various protection and integrity features.

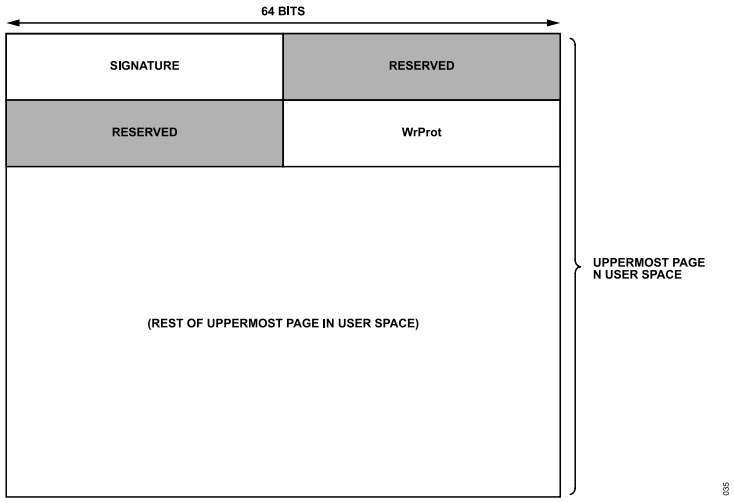

Figure 52. ADuCM355 Flash Memory Structure, Top of Page 64, Address 0x1F800 to 0x1FFFF

When writing to these locations, the user must always write 0xFFFFFFFF to the reserved locations. If the user intends to write to either location at run time, ensure that these reserved locations remain all 1s (0xFFFFFFFF). If data is stored to these reserved locations prior to run-time modification of the neighboring metadata spurious, ECC errors are likely generated when this metadata is read.

The top reserved words of metadata are utilized by the flash controller to enable protection and integrity features. For more details about these features, see the Protection and Integrity section.

FLASH ACCESS

Flash memory can be read, written, and erased by user code. Read access is provided through the cache controller using two AHB ports (as shown in Figure 53): ICode for instructions and DCode for data. Write access is provided through keyhole writes using APB control of memory mapped registers. The keyhole write implementation includes support for both DMA-based and manual user initiated writes.

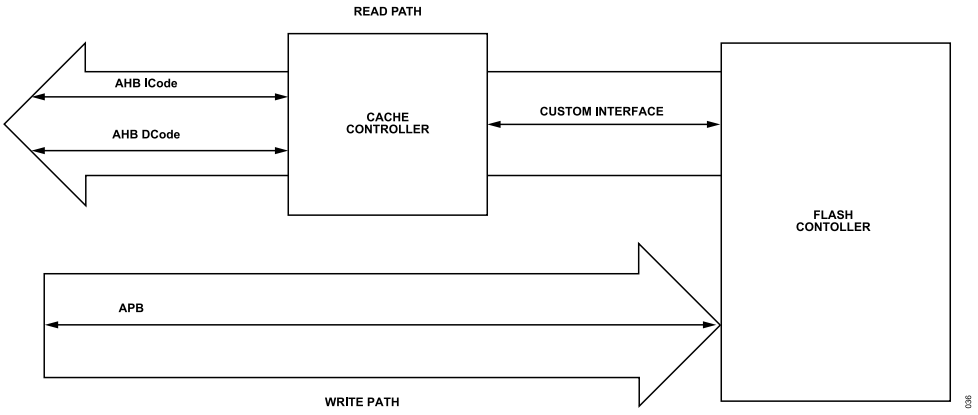

Figure 53. Flash Memory Read and Write Data Paths

Bus errors are generated if user code reads from a protected or an out of bounds address. Writes or erasures of protected addresses result in appropriate error flags set in the status register. The address setup for writes and erasures is automatically constrained to the flash address range.

FLASH CONTROLLER

READING FLASH

The flash controller provides two interfaces for reading nonvolatile storage: ICode and DCode. The ICode and DCode interfaces are accessed through the cache controller module through the AHB. The flash controller includes a prefetch buffer for ICode, by which it is possible to return data on both ICode and DCode interfaces in the same cycle.

Flash memory is available to read only after an automatic initialization process. Attempts to read during the flash controller initialization lead to a stall. Reads also stall if the flash controller is already busy performing another command (such as writing the flash), unless the prefetch buffer satisfies those reads.

ERASING FLASH

The flash controller provides page level granularity when erasing user space through the erase page command. Alternatively, user code can erase the entirety of user space at once using the mass erase command. These two commands have the same execution time.

Write protected pages cannot be erased, and the command is denied. Pages that are write protected also deny a mass erase command.

WRITING FLASH

Flash memory operates by settings bits to 1 when erased and selectively clearing bits to 0 when writing data. No write operation is capable of setting any bit to 1. For this reason, generalized write accesses must be prefixed by an erase operation.

Initial uploading of user content generally occurs immediately following a mass erase operation. Subsequent modification of already written locations in flash generally follows selective page erase operations with user code. During this process, the user must copy a full page to SRAM, erase the affected page, modify the in-memory content, and then finally write the page back to flash.

User space protections may prevent a page erase operation (see the Protection and Integrity section). All user space protections are cleared automatically after a mass erase or blank check. Note that a blank check passes only if all user space is already erased. In these two cases, there is no user space content to protect.

KEYHOLE WRITES

A keyhole write is an indirect write operation wherein user code programs memory mapped registers with a target address and data values, then commands the flash controller to perform a write operation in the background. The flash controller supports write access to the flash memory only through keyhole writes. This constraint on write access enables the flash controller to guarantee that writes occur properly as atomic data word (64-bit) operations with an associated ECC byte, if enabled (see the Protection and Integrity section).

If ECC is enabled, multiple writes to a single data word location cannot be performed without erasing the affected page between writes. Otherwise, ECC errors are reported. A maximum of two total writes are permitted to a single flash location (data word) between erasures regardless of ECC or data values, as per the flash intellectual property (IP) specifications. Writing any location more than twice per erasure can damage the nonvolatile memory or reduce its useful life. If multiple writes per location are required, disable ECC for some region of flash and target that region for these write operations.

Keyhole operations consist of writes to the following bits:

- KH_ADDR, Bits[18:3]. The target address in flash (for example, 0x104). The flash controller automatically trims lower bits to make the address data word aligned.
- KH_DATA0, Bits[31:0]. The bottom 32 bits of the 64-bit data word to be written (for example, 0x76543210).
- KH_DATA1, Bits[31:0]. The top 32 bits of the 64-bit data word to be written (for example, 0xFEDCBA98).
- CMD, Bits[3:0]. Assert the write command in the flash.

After the write command is asserted, the flash controller initiates a 64-bit dual word write to the address provided in KH_ADDR, Bits[18:3]. Word (32-bit), half word (16-bit), and byte (8-bit) writes are not supported. Because only 0 is written to the flash, masking can be used to write individual bits or bytes as necessary, provided the user takes the ECC into account.

Do not write to any of the keyhole registers when DMA access is enabled (UCFG, Bit 0 is set). Writing to these registers manually while DMA is enabled may result in spurious flash writes and can put the DMA and flash controllers out of synchronization, potentially stalling the DMA controller for long periods of time (approximately 20 µs to 40 µs).

FLASH CONTROLLER

BURST WRITES

Each 2 kB page of flash memory consists of eight rows of 256 bytes. Design constraints for programming the flash IP enable back to back writes within a single row to complete more quickly than the equivalent writes across row boundaries. For optimizing writes, attempt to write flash memory in aligned blocks of up to 256 bytes. For example, Row 0 is from flash offset Address 0x00 to Address 0xFC. Similarly, Row 1023 is from 0x3FF00 to 0x3FFFC.

To benefit from this write performance gain, the second and subsequent 64-bit write must be requested before the flash controller completes the first write. The STAT, Bit 3 flag is set when the first 64-bit write operation is nearing completion. This setting provides user code with a manageable window of time in which to assert the next write request. This flag can be polled or an interrupt can optionally be generated when it is asserted. Further descriptions of STAT, Bit 3 (WRALCOMP) and other available status flags are in Table 221.

Table 221. Status Flags

Flag Name	Description
CMDBUSY	Command busy. High when the controller is processing a command from the CMD register.
WRCLOSE	Keyhole closed. High during the first half of a write command, cleared when keyhole registers are free to be programmed for a subsequent write command.
CMDCOMP	Command complete. Sticky flag, set when a command is completed. Write 1 to clear.
WRALCOMP	Write almost completed. Sticky flag, set when approximately 20 µs to 40 µs remains for an ongoing write command. Write 1 to clear.

The following list outlines the procedure for performing multiple writes with the potential to burst within a row:

1. Wait for the STAT bit, CMDBUSY flag, to clear to ensure the prior command is ongoing.
2. Disable all interrupts so that sequenced writes to the KH_ADDR register, KH_DATA0 register, KH_DATA1 register, and CMD register are not interrupted by an interrupt service routine (ISR), which may also have flash writes.
3. Request the first 64-bit write through the keyhole access.
4. The flash controller starts the write process after the write command is written to the CMD register.
5. Check if the write command was accepted by reading the STAT register to see if any error flags are set. If the command results in an error, the write is not be performed.
6. Set IEN, Bit 1 to enable interrupt generation when WRALCOMP is asserted, and reenable other interrupts.
7. Continue the user program. The WRALCOMP interrupt vectors to an interrupt service routine to request the next write at the appropriate time.
8. The BurstWriteISR interrupt is called when an interrupt is generated by WRALCOMP.
9. Disable all interrupts so that sequenced writes to the KH_ADDR register, KH_DATA0 register, KH_DATA1 register, and CMD register are not interrupted by another ISR, which may also have flash writes.
10. Read the STAT register to verify the state of several status flags. Clear the flags by writing back the same value and ensure that the status bits are set appropriately. The WRALCOMP flag is set, indicating that there is still time for the next write to occur for a burst write. CMDBUSY is set, indicating that the current write operation is still ongoing. CMDCOMP is not set, indicating that a command has already completed, and, therefore, the command currently in progress is not the earlier write command (another function has initiated a new command). WRCLOSE is not set. If WRCLOSE is set, the keyhole registers are closed and cannot be written. WRCLOSE clears when WRALCOMP is asserted.
11. If no further writes are required, wait for STAT, Bit 2 to be set, clear the bit, and then exit the subroutine.
12. Request the next 64-bit dual word write through the keyhole access by writing the KH_ADDR register, KH_DATA0 register, KH_DATA1 register, and CMD register.
13. Check if the write command was accepted by reading the STAT register to see if any error flags are set.
14. Reenable interrupts.
15. Exit the ISR.

To prevent corrupting the flash, execute code with many write accesses to flash from SRAM if possible. Write operations cause ICode reads to stall, resulting in degraded performance when ICode access is required to fetch the next instruction. This degraded performance can be partially alleviated by the cache controller.

During a burst write, the flash controller overlaps the write operations and, therefore, any reported write failures (such as access denied due to write protection) may reflect the status of either of the two overlapped writes. Interpret a burst write failure as a failure for both the current and the prior write requests.

FLASH CONTROLLER

DMA WRITES

Keyhole writes generally require writing four memory mapped registers per flash write access. An optional address autoincrement feature can reduce the APB traffic required per flash write transaction to three register writes (KH_DATA0, KH_DATA1, and CMD) for all but the first in a series of sequential writes. The first write requires setting up the start address. DMA writes reduce the number of APB transactions to two. Every pair of KH_DATA0 and KH_DATA1 register writes results in a single flash write command executing and the address automatically incrementing to the next data word. Regardless of the value of autoincrement, DMA writes always increment the address in this manner.

To perform DMA-based writes, user code must first configure the DMA controller (a separate peripheral module) for basic access. DMA mode supports transferring data to and from peripherals, including the flash controller. After the DMA controller module is set up, it sits idle until a DMA request signal is asserted by the flash controller.

The user code must manually write KH_ADDR to set up the initial target address for DMA writes. The DMA controller must be configured to write all output data to KH_DATA1. Each pair of DMA writes to KH_DATA1 executes a single flash write command and waits for a corresponding delay before the next DMA request signal is made.

To start the flash controller process of requesting data from the DMA controller, the flash controller must be configured for DMA access. To enable DMA mode, user code must set UCFG, Bit 0. After DMA mode is enabled and the flash controller is idle, the flash controller drives the DMA request signal to the DMA module. The DMA request signal is driven at the appropriate time to support burst writes. New requests are made after the current write operation is nearly complete. For more information about DMA functionality, refer to the DMA Controller section.

PROTECTION AND INTEGRITY

Integrity of Information Space

User code does not have any control over the content of information space. If the integrity check fails on a POR, the check is attempted repeatedly until a preset number of attempts has occurred or the integrity check passes, providing some recoverability from power faults that occur during device power-up. For all other resets (except software resets, which do not reinitialize the flash controller), the integrity check is attempted just once.

In the event of an information space integrity check failure, it is expected that the device has failed. Dispose the device or return the device to Analog Devices for failure analysis. If the information space integrity check does not pass, the flash controller enters a special purpose debugging mode. In this mode, user space protection is automatically asserted and the flash controller local JTAG protection is set to allow the JTAG (or other serial wire type interface) to interact with the ICode, DCode, and APB interfaces.

In this special purpose debugging mode, the following occurs:

- All ICode reads return bus errors. Code is not executed from the flash memory without passing the integrity check.
- All DCode reads to user space return bus errors.
- All DCode reads to information space, except the top 128 bytes, are permitted. Reads of the top 128 bytes return bus errors.
- All write commands are denied. Write attempts set the appropriate error bits in the flash cache controller STAT register.
- User space protections can be bypassed only by satisfying the security requirements.

The status of the signature check after reset is read from the STAT, Bit 13. The status of ECC during the signature check is available in the STAT, Bits[16:15]. These values are read through a normal JTAG read if the JTAG interface is enabled.

User Space Protection

Two layers of user space protections are provided. Access protection protects user space from all read or write operations. This protection mechanism can be manually triggered but is typically automatically asserted in the event of a system failure or the serial wire debug interface being enabled.

Write protection is a user feature that enables blocks of user space pages to be protected against all write or erase commands. Write protection can be set by the user at run time, or by an Analog Devices bootloader. If protection is set by the user, the user stores the desired value in flash for the bootloader to consume during startup.

FLASH CONTROLLER

Access Protection

Access protection prevents third parties from reading or tampering with user data and program code through the JTAG or serial wire. Access protection applies to the entirety of user space. Access protection is enabled either when the serial wire debug is enabled or when flash initialization (information space sign check) fails. Enabling of serial wire debug and flash initialization are automatic features, and the user does not need to perform any actions for these mechanisms to enable access protection.

When access protection is enabled, all user space reads return bus errors, writes are denied, and erases are subject to WRPROT, Bits[31:0].

Access protection can be bypassed by executing a MASSERASE or BLANKCHECK command. The MASSERASE command is not allowed in the event that the WRPROT register has been modified from its reset value. The BLANKCHECK command is always permitted to execute, but only passes if all user space is already in an erased state.

Write Protection

User definable regions of user space can be configured in such a way that the flash controller refuses any attempts to modify them, affecting both write and erase commands. Write protection can be configured at run time or can be stored in user space metadata to be loaded by the Analog Devices bootloader during device start-up.

Run-Time Configuration

Write protection is configured by modifying the WRPROT memory mapped register. The word bit in the WRPROT register is a 32-bit wide bit field, but only 16 bits are required for 128 kB flash, representing the write protection state for 16 similarly sized blocks of user space pages. The flash memory is divided into 64 pages of user space storage. For write protection, these pages are logically divided into 16 blocks of four pages each. Write protection is independently controlled for each of these 16 blocks, and each bit of WRPROT controls the protection mechanism for a unique block of four pages of user space. The least significant bits of WRPROT correspond to the least significant pages of user space.

The bits in the WRPROT register are active low. 0 represents active write protection and 1 represents no protection for the corresponding block of pages. The WRPROT register is sticky at 0. After protection is enabled, it cannot be disabled without resetting the device. User code can assert write protection for any block of pages by clearing the appropriate bit in WRPROT, Bits[31:0] at any time. Assert write protection as early as possible in user code, write protect block zero (flash Page 0 to Page 3), and place user boot and integrity checking code in this block. By taking these actions, the user can fully control the write protection without relying on the Analog Devices bootloader to set up the WRPROT register.

Metadata Configuration

The address 0x0019C of user space contains a single 32-bit field, with 16 bits representing a set of 1-bit write protect flags for each of the 16 logical blocks, matching the functionality of the WRPROT register. See the User Space Metadata section for details.

The write protection bits are read from the flash by the Analog Devices bootloader and stored in the WRPROT register after a reset operation. The default (erased) state of flash memory is all 1s. As such, the default WRPROT register value disables protection for all pages in user space. Each bit of WRPROT, Bits[15:0] corresponds to the protection state for one block of the four user space pages.

User code can clear bits in the WRPROT register metadata word at run time, or this word can be included in the initial upload of user data and program code. Writing the WRPROT register metadata word at run time does not immediately affect the write protection state. If immediate protection is required, user code must also write the same values to the WRPROT register. When writing the WRPROT metadata, consider including write protection of the least significant page, which protects the metadata from a page erase or other modification. As with all flash locations, do not perform repeated writes without intermediate erasures, because repeated writes tend to cause ECC errors during readback.

After protection is enabled (the corresponding bit has been cleared in WRPROT), it cannot be disabled without resetting the device. For this reason, after write protection is enabled for any block of pages through the user space metadata, it cannot be disabled without erasing the least significant page of user space or otherwise affecting the flow of the bootloader.

The following sequence outlines the process of programming the write protection metadata word in flash:

1. Ensure that the least significant page of user space has been erased since the last time the metadata was written.
2. Write the desired value for the WRPROT metadata word directly to flash. Write 0 to a particular bit to enable protection for the corresponding block.
3. Verify that the write completed by polling the status register.

4. Reset the device. WRPROT, Bits[31:0] are automatically uploaded by the bootloader from user space to the WRPROT register and are used to enforce the protection scheme.

Signatures

Signatures are used to check the integrity of the flash device contents. Signature calculations do not include the ECC portion of the flash data bus or the most significant word read from the set of pages being signed. The most significant word is considered metadata and is intended to hold the expected signature value for the given set of pages.

The flash controller implements its own standalone CRC engine for generating and verifying signatures. However, this implementation matches the CRC accelerator peripheral with an initial value of 0xFFFFFFFF. For more information about the CRC algorithm used, refer to the Cyclic Redundancy Check section.

Signatures can be included in the initial upload of user data and program code generated before being uploaded, or can be generated and stored to flash at run time. Generation at run time utilizes the CRC accelerator or calls on the signature generation logic of the flash controller.

ECC bytes correspond to 64-bit data words in the flash memory. As such, if ECC is enabled, the most significant 64 bits (including the 32-bit signature word) must be written all at once. Otherwise, the ECC byte is corrupted by the second write. If using the flash controller to generate the signature value, leave the unused 32 bits paired with the signature word in their erased state (0xFFFFFFFF). Otherwise, spurious ECC errors may occur after device reset (depending on WRPROT configuration) and the device may become unusable.

The sign command generates a signature for all data from the start page to the end page, excluding the signature metadata word. The user must define the start and end pages by writing PAGE_ADDR0, Bits[18:10] and PAGE_ADDR1, Bits[18:10], respectively.

To generate or verify a signature, perform the following steps:

1. Write the start address of a contiguous set of pages to PAGE_ADDR0, Bits[18:10]. If out of bounds, the command is denied.
2. Write the end address of a contiguous set of pages to PAGE_ADDR1, Bits[18:10]. If out of bounds, the command is denied.
3. Write to Key, Bits[31:0]. Write the user key value (0x676C7565) to the key register.
4. Write to CMD, Bits[3:0]. Write the sign command (0x3) to the CMD register.
5. Wait. When the command is completed, STAT, Bit 2 is set.
6. If using the flash controller to generate a signature to write into the flash metadata, the signature value can be read from the signature register.
7. The generated signature is automatically compared with the data stored in the most significant 32-bit word of the region being signed. If the generated result does not match the stored value, a fail is reported in the STAT register by asserting a verify error in the STAT, Bit 2 field.

While the signature is being computed, all other accesses to the flash are stalled. Generating and verifying the signature for a 256 kB block (full user space) results in a stall duration of 32 kB flash reads (64 bits per read operation) or approximately 64,000 HCLK periods. The PAGE_ADDR0, Bits[18:10] and PAGE_ADDR1, Bits[18:10] addresses can be written as byte addresses but are consumed by the flash controller as page addresses. When this occurs, the lower 10 address bits are ignored. The sign command is denied if PAGE_ADDR1, Bits[18:10] is less than the PAGE_ADDR0, Bits[18:10]. Signatures are always performed at a page level granularity over continuous address ranges.

KEY REGISTER

To prevent spurious and potentially damaging flash accesses, some commands and registers are key protected. The user key is not a security element and is not meant to be a secret. Instead, this key protects users from negative consequences of software bugs, especially during early software development.

User Key

This key serves to prevent accidental access to some flash features and addresses. The key value is 0x676C7565. This key must be entered to run protected user commands (erase page, sign, mass erase, and abort) or to enable write access to the UCFG register. When entered, the key remains valid until an incorrect value is written to the key register, or a command is written to the CMD register. When any command is requested, this key is automatically cleared. If this key is entered to enable write access to the UCFG register, it is recommended to clear the key immediately after updating the registers.

FLASH CONTROLLER

ECC

The flash controller provides ECC-based error detection and correction for flash reads. ECC is enabled by default for information space, and thus provides assurance that flash initialization functions work properly. Information space signature check unconditionally considers ECC. The flash controller uses an 8-bit Hamming modified code to correct 1-bit errors or detect 2-bit errors for any dual word, 64-bit flash data access.

When enabled, the ECC engine is active during signature checks (refer to the Protection and Integrity section). User code can request a signature check of the entirety of user space and then check STAT, Bits[8:7] to determine if any single or dual bit data corruptions are present in user space.

Defaults and Configuration

In user space, ECC is off by default but can be selectively enabled by using the user code. Enabling ECC requires setting ECC_CFG, Bit 0. When enabled, ECC can apply to the entirety of user space or can be configured to apply only to a limited range. A single page address pointer (ECC_CFG, Bits[31:8]) is used to define the start address for ECC. All flash addresses from the start page through the top of user space (inclusive) have ECC enabled when ECC_CFG, Bit 0 is set.

ECC errors can be optionally reported as bus errors for ICode or DCode reads or can generate interrupts. Independent error reporting options are available for 1-bit corrections and 2-bit error detections by writing IEN, Bits[7:6] and IEN, Bits[5:4].

Error Handling

The impact of ECC errors during the information space signature check is described in the Signatures section. On any read operation, if the ECC engine observes a 1-bit error, the error is corrected automatically. In this case, the 1-bit error is either in the ECC byte itself or in the 64-bit dual word being read by the user. If a 2-bit error is observed, the ECC engine can only report the detection event. 2-bit errors cannot be corrected.

Depending on when the read happens (for example, during an ICode or DCode read, or as part of a built in command such as a signature check), appropriate flags are set in the status register. See the Status Register section for details.

If interrupt generation is enabled in IEN, Bits[7:6] or IEN, Bits[5:4], the source address of the ECC error causing the interrupt is available in the ECC_ADDR register for the interrupt service routine to read.

ECC Errors During Execution of Sign Command

ECC errors observed during signature checks generate the appropriate status register flags after completion, but do not populate the ECC_ADDR register.

Concurrent Errors

If ECC errors occur on DCode and ICode simultaneously (for example, from an ICode prefetch match and a DCode flash read), ECC error status information is prioritized. In first priority, 2-bit ECC errors are given priority over 1-bit ECC errors or corrections. For example, if a 2-bit ECC error is observed on a DCode read in the same cycle as a 1-bit ECC error or correction on an ICode read, the ECC error status is updated for DCode only.

In second priority, ICode is given priority over DCode. For example, if a 2-bit error is observed on an ICode read and a DCode read in the same cycle, the ECC error status is updated for ICode only.

Read of Erased Location

When erased, the flash memory holds a value of all 1s, including the ECC byte appended to every 6-bit data word. The proper ECC metadata for 64 1s is not 0xFF. As such, in its erased state, the flash memory holds data and ECC metadata representing some number of bit errors. For this reason, any flash reads of erased locations automatically bypass the ECC engine. If user code reads a location with all 1s in both the 64-bit data word and the ECC byte, the read returns data without indicating any ECC errors.

CLOCK AND TIMINGS

The flash controller is preconfigured to provide safe timing parameters for all flash operations for core clock frequencies of 26 MHz or less, and a reference clock frequency of 13 MHz.

FLASH CONTROLLER

FLASH OPERATING MODES

The flash memory used by the ADuCM355 processor supports the following power optimizing features.

Sleep Mode

The user code can put the flash IP into a low-power sleep mode by writing the sleep command to the CMD register. The flash controller wakes the flash IP from sleep automatically on the first flash access following a sleep command. The user code can observe the sleep state of flash by reading STAT, Bit 6.

A flash wake-up event is triggered by the cache controller, DMA reads, user code, or other peripherals, which may also attempt to read flash memory at any time. Ensure that user code occasionally polls STAT, Bit 6 to verify that the flash IP is still sleeping when the user expects it to be.

The flash controller does not honor any new commands while the flash IP is in sleep mode. The only supported commands in this mode are idle and abort. DMA write requests are stalled automatically by entering sleep mode.

System interrupt-based aborts (as configured through the ABORT_EN_LO register) are generally used to abort any ongoing flash commands in the event of an enabled system interrupt. However, such an interrupt does not wake the flash from sleep mode. If the system interrupt can be serviced by accessing the flash, it remains in sleep mode. If servicing the interrupt requires accessing the flash, the flash access itself serves to wake the flash IP.

Waking from sleep incurs an approximately 5 µs latency before executing any reads or commands. This latency is a requirement of the flash IP. User code can wake the flash IP early by executing an idle command. This command wakes the flash IP without any other effect on the controller.

For consistency, the abort command can also be used to wake the flash IP. Waking with the abort command differs from waking by the idle command only in that the status register reports STAT, Bits[5:4] = 11 to match expected user code checks for status register values.

Power-Down Mode Support

The ADuCM355 processor automatically powers down the flash IP when the device hibernates. To support this feature, the flash controller operates with the power management unit and delays hibernation until any ongoing flash accesses are completed. User code is responsible for reading and evaluating flash status registers prior to entering hibernate mode because status registers are not retained in hibernate mode. The abort command can be used to abruptly end an ongoing flash command. Use abort commands sparingly to avoid eventual damage to the flash array.

Clock Gating

A series of clock gates are inserted into the flash controller to automatically gate off unused components of the module. No user configuration or control is required. Unused portions of the flash controller are automatically gated off when appropriate. For example, while in sleep mode, the majority of the flash controller is gated off to save power.

Flash Interrupts and Exceptions

The flash controller can selectively generate interrupts for many events. Table 222 outlines the events that may generate interrupts and bit fields of the IEN register used to control interrupt generation for each event.

Table 222. Interrupts and Bit Fields

Name (Bit Field)	Description
ECC_ERROR	IRQ is generated when 2-bit ECC errors are observed and when this field is set to 2.
ECC_CORRECT	IRQ is generated when 1-bit ECC corrections are observed and when this field is set to 2.
CMDFAIL	IRQ generated when a command or write operation completes with an error status.
WRALCMPLT	IRQ generated when an active flash write is nearly complete and the keyhole registers are open for another write. If fulfilled in time, a burst write occurs.
CMDCMPLT	IRQ generated when a command or flash write operation completes.

FLASH CONTROLLER

Flash Programming Model

This list provides an example sequence to execute the page erase command using the flash controller. The same sequence can be used for other commands with some modifications:

1. Program the PAGE_ADDR0 or PAGE_ADDR1 register with the address of the page that must be erased.
2. Write the flash user key to the key register.
3. Write the command to be executed into the CMD register.
4. Poll for STAT, Bit 2 to be set in the flash cache controller STAT register.

REGISTER SUMMARY: FLASH CACHE CONTROLLER

Table 223. FLCC Register Summary

Address	Name	Description	Reset	Access
0x40018000	STAT	Status	0x00000000	R/W
0x40018004	IEN	Interrupt enable	0x40	R/W
0x40018008	CMD	Command	0x00000000	R/W
0x4001800C	KH_ADDR	Write address	0x00000000	R/W
0x40018010	KH_DATA0	Write lower data	0xFFFFFFFF	R/W
0x40018014	KH_DATA1	Write upper data	0xFFFFFFFF	R/W
0x40018018	PAGE_ADDR0	Lower page address	0x00000000	R/W
0x4001801C	PAGE_ADDR1	Upper page address	0x00000000	R/W
0x40018020	KEY	Key	0x00000000	W
0x40018024	WR_ABORT_ADDR	Write abort address	0xXXXXXXXX	R
0x40018028	WRPROT	Write protection	0xFFFFFFFF	R/W
0x4001802C	SIGNATURE	Signature	0xXXXXXXXX	R
0x40018030	UCFG	User configuration	0x00000000	R/W
0x4001803C	ABORT_EN_LO	IRQ abort enable (lower bits)	0x00000000	R/W
0x40018040	ABORT_EN_HI	IRQ abort enable (upper bits)	0x00000000	R/W
0x40018044	ECC_CFG	ECC configuration	0x00000002	R/W
0x40018048	ECC_ADDR	ECC status (address)	0x00000000	R
0x40018050	ADI_POR_SEC	Analog Devices flash security	0x00000000	R/W

REGISTER DETAILS: FLASH CACHE CONTROLLER (FLCC)

STATUS REGISTER

Address: 0x40018000, Reset: 0x00000000, Name: STAT

This register provides information on current command states, error detection, and correction.

Table 224. Bit Descriptions for STAT

Bits	Bit Name	Settings	Description	Reset	Access
[31:30]	Reserved		Reserved.	0x0	R
29	CACHESRAMPERR		SRAM Parity Errors in Cache Controller. This register provides details for AHB errors generated due to cache SRAM parity error on the ICode bus.	0x0	R
[28:27]	ECCDCODE		DCode AHB Error ECC Status. Provides details for AHB errors generated due to ECC errors or corrections on the DCode bus.	0x0	R/W1C
		00	No error. No errors or corrections reported since reset or the register was last cleared.		
		01	2-bit error. 2-bit ECC error has been detected and reported on AHB read access.		
		10	1-bit correction. 1-bit ECC correction has been detected and reported on AHB read access.		
		11	Reserved.		
[26:25]	ECCICODE		ICode AHB Error ECC Status. Provides details for AHB errors generated due to ECC errors or corrections on the ICode bus.	0x0	R/W1C
		00	No error. No errors or corrections reported since reset or the register was last cleared.		
		01	2-bit error. 2-bit ECC error has been detected and reported on AHB read access.		
		10	1-bit correction. 1-bit ECC correction has been detected and reported on AHB read access.		
		11	Reserved.		
[24:20]	Reserved		Reserved.	0x0	R
[19:17]	ECCERRCNT		ECC Correction Counter. This counter keeps track of overlapping ECC 1-bit correction reports. When configured to generate IRQs or AHB errors in the event of an ECC correction event, this field counts the number of ECC corrections that occur after the first reported correction. The counter remains at full scale when it overflows and clears automatically when clearing either the ECCICODE or ECCDCODE status bits.	0x0	R/W1C
[16:15]	ECCINFOSIGN		ECC Status of Flash Initialization. ECC status after the end of automatic signature check on information space.	0x0	R
		00	No error. No errors reported.		
		01	2-bit error. One or more 2-bit ECC errors detected during signature check. Signature check has failed.		
		10	1-bit error. One or more 1-bit ECC corrections performed during signature check. Signature check passes if checksum still matches.		
		11	1-bit and 2-bit error. At least one of each ECC event (1-bit correction and 2-bit error) were detected during signature check. Signature check fails.		
14	INIT		Flash Controller Initialization in Progress. Flash controller initialization is in progress. Until this bit deasserts, AHB accesses stall and APB commands are ignored.	0x0	R
13	SIGNERR		Signature Check Failure During Initialization. Indicates an automatic signature check has failed during flash controller initialization. The register value is valid only after the signature check has completed.	0x0	R
12	Reserved		Reserved.	0x0	R
11	OVERLAP		Overlapping Command. This bit is set when a command is requested while another command is busy. Overlapping commands are ignored.	0x0	R/W1C
[10:9]	ECCRDERR		ECC IRQ Cause. This field reports the cause of recently generated interrupts. The controller can be configured to generate interrupts for 1-bit or 2-bit ECC events by writing the appropriate values to IEN, Bits[7:6]. These bits are sticky high until cleared by user code.	0x0	R/W1C
		00	No error.		
		01	2-bit error. ECC engine detected a noncorrectable 2-bit error during AHB read access.		
		10	1-bit correction. ECC engine corrected a 1-bit error during AHB read access.		
		11	1-bit and 2-bit events. ECC engine detected both 1-bit and 2-bit data corruptions, which triggered IRQs. A single read can only report one type of event. This status indicates that a subsequent AHB read access incurred the alternate ECC error event. By default, 1-bit ECC		

REGISTER DETAILS: FLASH CACHE CONTROLLER (FLCC)

Table 224. Bit Descriptions for STAT (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
			corrections are reported as IRQs and 2-bit ECC errors are reported as bus faults. It is not recommended to report both types as IRQs, because the status bits become ambiguous when trying to diagnose which fault came first.		
[8:7]	ECCERRCMD		ECC Errors Detected During User Issued Sign Command. ECC errors, if produced during signature commands, are reported by these bits. To generate interrupts based on these bits, set the corresponding bits in the IEN register.	0x0	R/W1C
		00	No error, completed flash read operation during signature check.		
		01	2-bit error. During signature commands, 2-bit error is detected on one or more flash locations, not corrected.		
		10	1-bit error. 1-bit error is corrected for one or more flash locations while performing signature commands.		
		11	1-bit or 2-bit error. During signature commands, 1-bit and 2-bit errors are detected on one or more flash locations.		
6	SLEEPING		Flash Array is in Low-Power (Sleep) Mode. Indicates that the flash array is in a low-power (sleep) mode. The flash controller automatically wakes the flash when required for another data transaction. The user can wake the flash at any time by writing the idle command to the CMD register. Flash wake-up times vary but are typically approximately 5 µs. When possible, it is recommended that the user begin waking the flash approximately 5 µs before it is used for performance optimization.	0x0	R
[5:4]	CMDFAIL		Provides Information on Command Failures. This field indicates the status of a command upon completion. If multiple commands are executed without clearing these bits, only the first error encountered is stored.	0x0	R/W1C
		00	Complete. Completion of a command.		
		01	Ignored. Attempted access of a protected or out of memory location is ignored.		
		10	Verify error. Read verify error occurred. This status returns for both failed erasure and failed signature check. In the case of failed erasure, after erasing flash page(s), the controller reads the corresponding word(s) to verify that the erasure completed. If data persists, the erasure has failed and this field reports the failure. In the case of a failed signature check, if the sign command is executed and the resulting signature does not match the data stored in the most significant 32-bit word of the sign checked block, the sign check has failed and this field reports the failure.		
		11	Abort. Indicates a command was aborted either by user code or by a system interrupt. See the IRQ Abort Enable (Lower Bits) Register section and the IRQ Abort Enable (Upper Bits) Register section for more details.		
3	WRALCOMP		Write Almost Complete. Write data registers are reopened for access as an ongoing write nears completion. Requesting another write operation before the CMDCOMP bit asserts results in a burst write. Burst writes take advantage of low level protocols of the flash memory and result in significant performance gains (approximately 15 µs saved from each write operation). The performance gain of a burst write only applies to back to back writes within the same row of the flash array.	0x0	R/W1C
2	CMDCOMP		Command Complete. This bit asserts when a command completes. It automatically clears when a new command is requested. Following a POR, the flash controller performs a number of operations, such as verifying the integrity of code in information space. At the conclusion of this process, the controller sets this bit to indicate that the process has completed.	0x0	R/W1C
1	WRCLOSE		Write Registers are Closed. The write data registers (KH_DATA0 and KH_DATA1), address register (KH_ADDR), and command register (CMD) are closed for access. This bit is asserted part of the time while a write is in progress. If this bit is high, the related registers are in use by the flash controller and cannot be written. This bit clears when the WRALCOMP flag goes high, indicating that the ongoing write command has consumed the associated data and these registers can now be overwritten with new data.	0x0	R
0	CMDBUSY		Command Busy. This bit is asserted when the flash block is actively executing any command entered via the command register. There is a slight delay between requesting a command and this bit asserting. Watch the CMDCOMP bit rather than this bit when polling for command completion.	0x0	R

REGISTER DETAILS: FLASH CACHE CONTROLLER (FLCC)

INTERRUPT ENABLE REGISTER

Address: 0x40018004, Reset: 0x40, Name: IEN

Used to specify when interrupts are generated.

Table 225. Bit Descriptions for IEN

Bits	Bit Name	Settings	Description	Reset	Access
[31:8]	Reserved		Reserved.	0x0	R
[7:6]	ECC_ERROR		Control Whether to Generate Bus Errors, Interrupts, or Neither in Response to 2-Bit ECC Error Events.	0x1	R/W
		0	Do not generate a response to ECC events.		
		1	Generate bus errors in response to ECC events.		
		10	Generate IRQs in response to ECC events.		
[5:4]	ECC_CORRECT		Control Whether to Generate Bus Errors, Interrupts, or Neither in Response to 1-Bit ECC Correction Events.	0x0	R/W
		0	Do not generate a response to ECC events.		
		1	Generate bus errors in response to ECC events.		
		10	Generate IRQs in response to ECC events.		
3	Reserved		Reserved.	0x0	R
2	CMDFAIL		Command Fail Interrupt Enable. If this bit is set, an interrupt is generated when a command or flash write completes with an error status.	0x0	R/W
1	WRALCMPLT		Write Almost Complete Interrupt Enable.	0x0	R/W
0	CMDCMPLT		Command Complete Interrupt Enable. When set, an interrupt is generated when a command or flash write completes.	0x0	R/W

COMMAND REGISTER

Address: 0x40018008, Reset: 0x00000000, Name: CMD

Write this register to execute a specified command. The user key must first be written to the key register for most command requests to be honored. See the Key Register section for details.

Table 226. Bit Descriptions for CMD

Bits	Bit Name	Settings	Description	Reset	Access
[31:4]	Reserved		Reserved.	0x0	R
[3:0]	VALUE		Commands. Write command values to this register to begin a specific operation. All commands except write and idle first require writing the user key to the key register.	0x0	R/W
		0x0	Idle. No key is required. No command executed. If flash IP is in the sleep state, this command wakes the flash by returning command complete and no error codes.		
		0x1	Abort. User key is required. Use the abort command sparingly to gain access to the flash IP during time sensitive events. For example, a low voltage alarm can abort an ongoing flash write or erase command to enable user code to shut down the device. The flash array can be damaged by overuse of the abort command. If this command is issued, any command currently in progress stops abruptly, if possible. The status indicates a command was completed with an abort error. Abort is the only command that can be issued while another command is already in progress, with the exception of stacking one write command on top of a single ongoing write command. All other overlapping command combinations are invalid unless the new command is an abort. If a write or erase is aborted, some flash IP timing requirements can be violated and it is not possible to determine if the write or erase completed. Read the affected locations to determine the outcome of the aborted commands. An aborted command can result in a weak programmed flash. It is always advisable to erase the affected region and reprogram it. Depending on how far along the flash controller is in the process of performing a command, an abort is not always possible. Some flash IP timing parameters must not be violated. It is difficult to predict what these parameters are in software. Therefore, consider aborts to be a request that may have no effect on actual command duration.		
		0x2	Requests flash to enter sleep mode. User key is required. When sleeping, any ICode, DCode, or DMA transaction wakes the flash automatically. The wake-up process takes approximately 5 µs. If user code can predict approximately 5 µs ahead of time that the flash is required, the user can write an idle command to		

REGISTER DETAILS: FLASH CACHE CONTROLLER (FLCC)

Table 226. Bit Descriptions for CMD (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
			the CMD register to manually wake the flash. An abort command is also respected for waking the device and returns the appropriate status bits, indicating that the sleep command was aborted. When awoken for any reason, the device remains awake until user code asserts a sleep command.		
		0x3	Sign. User key is required. Use this command to generate a signature for a block of data. Signatures can be generated for blocks of whole pages only. Write the address of the start page to the PAGE_ADDR0 register, write the address of the end page to the PAGE_ADDR1 register, and then write this code to the CMD register to start the signature generation. When the command has completed, the signature is readable from the signature register.		
		0x4	Write. No key is required. This command takes the address and data from the KH_ADDR register, KH_DATA0 register, and KH_DATA1 register and executes a single 64-bit write operation targeting the specified address. More information can be found in the Writing Flash section and the Write Protection Register section.		
		0x5	Checks all user space and fails if any bits in user space are cleared. User key is required. Performs a blank check on all user space. If any bits in user space are cleared, the command fails with a read verify status. If all of user space is 0xFF, the command passes. This command is intended to support early customer software development. When an unprogrammed device boots with security features preventing reads and writes of user space, this command can be used to verify that the user space contains no proprietary information. If this command passes, read and write protection of user space is cleared.		
		0x6	Erase page. User key is required. Write the address of the page to be erased to the PAGE_ADDR0 register, then write this code to the CMD register. When the erase has completed, the full page is verified automatically to ensure a complete erasure. If there is a read verify error, it is indicated in the STAT register. To erase multiple pages, wait until a previous page erase has completed. Check the status, then issue a command to start the next page erase.		
		0x7	Mass erase. User key is required. Erase all of flash user space. When the erase has completed, the full user space is verified automatically to ensure a complete erasure. If there is a read verify error, it is indicated in the status register.		

WRITE ADDRESS REGISTER

Address: 0x4001800C, Reset: 0x00000000, Name: KH_ADDR

This register writes the byte address of any byte of a 64-bit, dual-word flash location to be targeted by a write command. All writes target 64-bit, dual-word elements in the flash array. User code can mask byte data to emulate byte, half word, or word writes. Flash IP specifications warn against writing to any location more than twice between erasures. When writing a location more than once, be aware that ECC metadata cannot be updated appropriately. Use code to disable ECC for the relevant region of flash. Writing any address above the valid range of flash memory saturates the address to prevent aliasing. Take care to target valid flash address locations.

Table 227. Bit Descriptions for KH_ADDR

Bits	Bit Name	Settings	Description	Reset	Access
[31:19]	Reserved		Reserved.	0x0	R
[18:3]	VALUE		Address to Be Written on a Write Command.	0x0	R/W
[2:0]	Reserved		Reserved.	0x0	R

WRITE LOWER DATA REGISTER

Address: 0x40018010, Reset: 0xFFFFFFFF, Name: KH_DATA0

This register contains the lower half of 64-bit dual-word data to be written to flash.

Table 228. Bit Descriptions for KH_DATA0

Bits	Bit Name	Settings	Description	Reset	Access
[31:0]	VALUE		Lower Half of 64-Bit Dual-Word Data to Be Written on a Write Command.	0xFFFFFFFF	R/W

REGISTER DETAILS: FLASH CACHE CONTROLLER (FLCC)

WRITE UPPER DATA REGISTER

Address: 0x40018014, Reset: 0xFFFFFFFF, Name: KH_DATA1

This register contains the upper half of 64-bit dual-word data to be written to flash.

Table 229. Bit Descriptions for KH_DATA1

Bits	Bit Name	Settings	Description	Reset	Access
[31:0]	VALUE		Upper Half of 64-Bit Dual-Word Data to Be Written on a Write Command. If DMA is enabled, this register acts as a FIFO. Writes to this register push the old data to the lower half of 64-bit data (KH_DATA0). When this register is written twice in DMA mode, the FIFO becomes full and a flash write command is automatically executed.	0xFFFFFFFF	R/W

LOWER PAGE ADDRESS REGISTER

Address: 0x40018018, Reset: 0x00000000, Name: PAGE_ADDR0

Write a byte address to this register to select the page in which that byte exists.

The selected page can be used for an erase page command (selecting which page to erase) or for a sign command (selecting the start page for a block on which a signature is calculated). For commands using both the PAGE_ADDR0 register and PAGE_ADDR1 register, ensure that PAGE_ADDR0 is always less than or equal to PAGE_ADDR1, or else the command is denied.

Writing any address above the valid range of flash memory saturates the address register to prevent aliasing in the flash memory space.

Table 230. Bit Descriptions for PAGE_ADDR0

Bits	Bit Name	Settings	Description	Reset	Access
[31:19]	Reserved		Reserved.	0x0	R
[18:10]	VALUE		Lower Address Bits of the Page Address.	0x0	R/W
[9:0]	Reserved		Reserved.	0x0	R

UPPER PAGE ADDRESS REGISTER

Address: 0x4001801C, Reset: 0x00000000, Name: PAGE_ADDR1

Write a byte address to this register to select the page in which that byte exists.

The selected page can be used for a sign command (selecting the end page for a block on which a signature is calculated). For commands using both the PAGE_ADDR0 register and PAGE_ADDR1 register, ensure that PAGE_ADDR0 is always less than or equal to PAGE_ADDR1, or else the command is denied.

Writing any address above the valid range of flash memory saturates the address register to prevent aliasing in the flash memory space.

Table 231. Bit Descriptions for PAGE_ADDR1

Bits	Bit Name	Settings	Description	Reset	Access
[31:19]	Reserved		Reserved.	0x0	R
[18:10]	VALUE		Upper Address Bits of the Page Address.	0x0	R/W
[9:0]	Reserved		Reserved.	0x0	R

KEY REGISTER

Address: 0x40018020, Reset: 0x00000000, Name: KEY

When user code writes a key to access protected features, the key value must be written to this register.

Table 232. Bit descriptions for KEY

Bits	Bit Name	Settings	Description	Reset	Access
[31:0]	VALUE	0x676C7565	Key Register. Unlock protected features by writing the appropriate key value to this register. User key. Write this field with hexadecimal value of 0x676C7565 to enable certain registers to be modified or to allow certain commands to be executed. This key is used as a check to prevent	0x0	W

REGISTER DETAILS: FLASH CACHE CONTROLLER (FLCC)

Table 232. Bit descriptions for KEY (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
			accidental modification of settings or flash content. It is not a security component and is not intended to be confidential information.		

WRITE ABORT ADDRESS REGISTER

Address: 0x40018024, Reset: 0xXXXXXXXX, Name: WR_ABORT_ADDR

This register contains the address of a recently aborted write command. This address is only populated if the aborted write command was started. If the command is aborted early enough to have no effect on the flash IP, this address is not updated.

Table 233. Bit Descriptions for WR_ABORT_ADDR

Bits	Bit Name	Settings	Description	Reset	Access
[31:0]	VALUE		Hold Target Address. Holds the address targeted by an ongoing write command and retains its value after an abort event. User code can read this register to determine the flash locations affected by a write abort. The register value is not guaranteed to persist after a new flash command is requested. Therefore, read this value immediately following an aborted write.	0xXXXXXXXX	R

WRITE PROTECTION REGISTER

Address: 0x40018028, Reset: 0xFFFFFFFF, Name: WRPROT

A user key is required to modify this register. This register can be automatically configured during device startup, in which case the bootloader reads data from user space and loads that data into this register.

User code can affect nonvolatile write protection by writing to the appropriate location in the flash memory (see the Protection and Integrity section). By default, the relevant location in flash is 0x0019C but can be relocated by the Analog Devices bootloader.

Alternatively, user code can assert protection at run time for any unprotected blocks by directly writing this register. Blocks can have protection added but cannot have protection removed, and changes are lost on reset. This approach is suggested especially during user code development.

All write protection is cleared on a POR, but the Analog Devices bootloader reasserts write protection (as defined by the WRPROT word) in user space before enabling user access to the flash array. Removing write protection can only be performed by an erase page command of the most significant page in user space (provided that page is not currently protected) or by a mass erase command. Following a completed mass erase command, all protection of pages in user space is immediately cleared. The user can write to user space immediately following such an erase without a device reset required.

Table 234. Bit Descriptions for WRPROT

Bits	Bit Name	Settings	Description	Reset	Access
[31:0]	WORD		Clear Bits to Write Protect Related Groups of User Space Pages. After cleared, these bits can only be set again by resetting the device. Each bit of this 32-bit word represents a 32nd of the total available user space. For 128 kB parts consisting of 2 kB pages (64 pages), only 16 bits are usable with each bit represents the write protection state of a group of four pages. The least significant bit of this register corresponds to the least significant group of pages in user space.	0xFFFFFFFF	R/W0C

SIGNATURE REGISTER

Address: 0x4001802C, Reset: 0xXXXXXXXX, Name: SIGNATURE

Table 235. Bit Descriptions for SIGNATURE

Bits	Bit Name	Settings	Description	Reset	Access
[31:0]	VALUE		Provides Read Access to the Most Recently Generated Signature.	0xXXXXXXXX	R

USER CONFIGURATION REGISTER

Address: 0x40018030, Reset: 0x00000000, Name: UCFG

User key is required. Write to this register to enable user control of DMA and autoincrement features. When user code has finished accessing this register, write arbitrary data to the key register to reassert protection.

REGISTER DETAILS: FLASH CACHE CONTROLLER (FLCC)

Table 236. Bit Descriptions for UCFG

Bits	Bit Name	Settings	Description	Reset	Access
[31:2]	Reserved		Reserved.	0x0	R
1	AUTOINCEN		Automatic Address Increment for Keyhole Access. When this bit is set, KH_ADDR automatically increments by 0x8 during each write command or after each read command, enabling user code to write a series of sequential flash locations without having to manually set the flash address for each write. The KH_ADDR register is incremented, and can be observed by user code when STAT, Bit 5 is asserted during a write command or after a read command. When this bit is set, user code cannot directly modify KH_ADDR.	0x0	R/W
0	KHDMAEN		Keyhole DMA Enable. The flash controller interacts with the DMA controller when this bit is set. Prior to setting this bit, write the starting address to the KH_ADDR register. Then configure the DMA controller to write data to the KH_DATA1 register (address must be data word aligned), to always write pairs of 32-bit words (R_POWER = 1), and to write an integer number of data pairs (for an odd number of words, user code must write one word manually without the help of DMA). All DMA writes automatically increment the target address (similar to the behavior of UCFG, Bit 1). The DMA controller can only be used to write sequential addresses starting from the value of KH_ADDR. The flash controller automatically begins write operations each time the DMA controller provides a pair of words to write. Interaction with the DMA controller is designed to use burst writes, which can significantly reduce overall programming time.	0x0	R/W

IRQ ABORT ENABLE (LOWER BITS) REGISTER

Address: 0x4001803C, Reset: 0x00000000, Name: ABORT_EN_LO

Table 237. Bit Descriptions for ABORT_EN_LO

Bits	Bit Name	Settings	Description	Reset	Access
[31:0]	VALUE[31:0]		System IRQ Abort Enable. To allow a system interrupt to abort an ongoing flash command, write 1 to the bit in this register corresponding with the desired system IRQ number.	0x0	R/W

IRQ ABORT ENABLE (UPPER BITS) REGISTER

Address: 0x40018040, Reset: 0x00000000, Name: ABORT_EN_HI

Table 238. Bit Descriptions for ABORT_EN_HI

Bits	Bit Name	Settings	Description	Reset	Access
[31:0]	VALUE[63:32]		System IRQ Abort Enable. To allow a system interrupt to abort an ongoing flash command, write 1 to the bit in this register corresponding with the desired system IRQ number.	0x0	R/W

ECC CONFIGURATION REGISTER

Address: 0x40018044, Reset: 0x00000002, Name: ECC_CFG

Table 239. Bit Descriptions for ECC_CFG

Bits	Bit Name	Settings	Description	Reset	Access
[31:8]	PTR		ECC Start Page Pointer. Write Bits[31:8] of the start page address into Bits[31:8] of this register. This bit is a byte address for any page in user flash. The bottom bits of this address are ignored by the flash controller, forming a page address. When ECC is enabled and user code reads any address from within the page specified, ECC functions are performed. Reads from less significant pages bypass ECC entirely.	0x0	R/W
[7:2]	Reserved		Reserved.	0x0	R
1	INFOEN		Information Space ECC Enable Bit. ECC is enabled by default for information space. Clearing this bit disables ECC in information space. This bit is not key protected.	0x1	R/W
0	EN		ECC Enable. Set this bit to enable ECC on user space. ECC is enabled on all future flash reads in user space from any address between ECC_CFG, Bits[31:8] through the top of user space (inclusive). When cleared (or accessing addresses outside the enabled range), the flash controller returns the raw data in response to both ICode and DCode reads of user space. No error corrections are made or reported.	0x0	R/W

REGISTER DETAILS: FLASH CACHE CONTROLLER (FLCC)

ECC STATUS (ADDRESS) REGISTER

Address: 0x40018048, Reset: 0x00000000, Name: ECC_ADDR

This register is updated when ECC error or correction events occur. ECC error and correction events can generate interrupts if the appropriate bits of the IEN register are selected, which generates a bus fault. This register records the address of the first ECC error or correction event to generate an interrupt since reset or the last time the ECC status bits were cleared. If the status bits are cleared in the same cycle as a new ECC event (selected to generate an IRQ), a new address is recorded and the status bits remain set.

Errors have priority over corrections. Two or more corrupt bits results in an error. A correction results in proper data being returned after a single bit is corrected. If an error and a correction occur in the same cycle, this register reports the error address. When two of the same priority ECC events occur (both errors or both corrections), the ICode bus has priority over DCode. As such, if both ICode and DCode buses generate the same type of ECC event in the same cycle, the ICode address is stored in this register.

The register cannot be cleared except by reset. It always holds the address of the most recently reported ECC correction or error.

Table 240. Bit Descriptions for ECC_ADDR

Bits	Bit Name	Settings	Description	Reset	Access
[31:19]	Reserved		Reserved.	0x0	R
[18:0]	VALUE		Address for Which ECC Error Is Detected.	0x0	R

ANALOG DEVICES FLASH SECURITY REGISTER

Address: 0x40018050, Reset: 0x00000000, Name: ADI_POR_SEC

This register resets only after a POR or an external reset.

Table 241. Bit Descriptions for ADI_POR_SEC

Bits	Bit Name	Settings	Description	Reset	Access
[31:1]	Reserved		Reserved.	0x0	R
0	SECURE		Prevents Read or Write to User Space. Set this bit to prevent a read or write access to user space. This bit is sticky when set and requires a user key. When set, this bit cannot be cleared without resetting the device using a POR or external reset. This bit plays a direct role in user space security enforcement. When set, this bit prevents access to user space. DCode reads return bus faults with the data bus = 0. ICode reads return bus faults with the data bus = 0. APB writes are denied, and the flash content is unchanged. When set, the user can still perform mass erase and page erase operations. However, the WRPROT register still applies and only unprotected pages can be erased. Mass erase is not allowed if any pages are protected. See the Security Features section for more information.	0x0	R/W1S

SRAM

This section provides an overview of the SRAM functionality of the ADuCM355 processor. For details about the SRAM_INITSTAT, SRAM_CTL, and SRAMRET registers, refer to the Register Summary: Power Management Unit section.

This memory space contains the application instructions and constant data that must be executed in real time. It supports read and write access via the Cortex-M3 core and read and write DMA access by system peripherals. SRAM is divided into data SRAM of 32 kB and instruction SRAM of 32 kB. If instruction SRAM is not enabled, the associated 32 kB can be mapped as data SRAM, resulting in a 64 kB data SRAM.

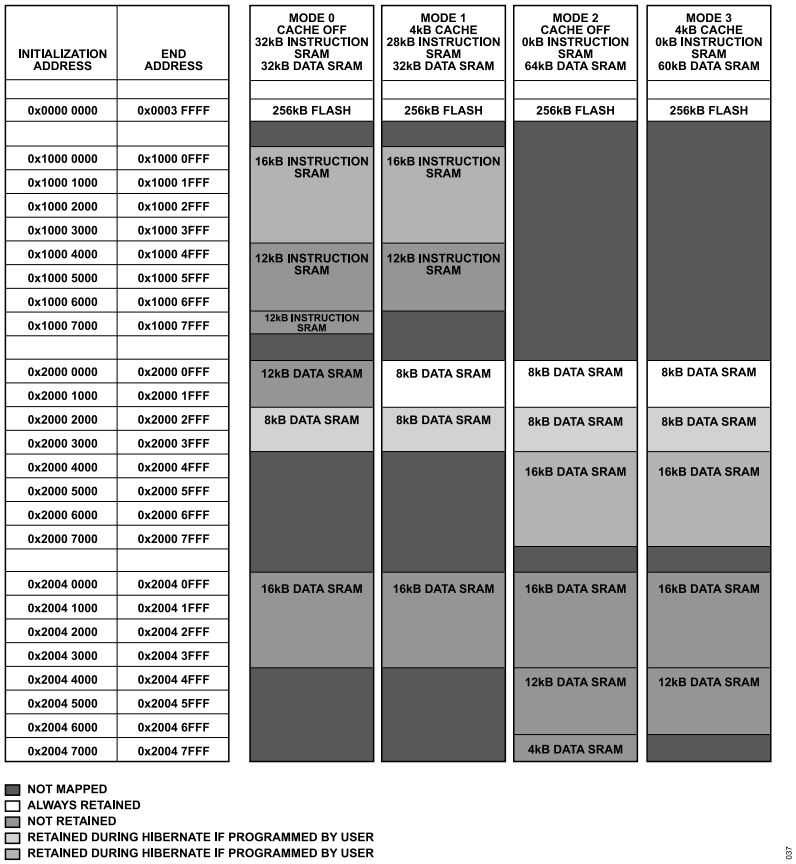

Figure 54. ADuCM355 SRAM Memory Details

SRAM FEATURES

The SRAM used by the ADuCM355 processor supports the following features:

- Low-power controller for data SRAM, instruction SRAM, and cache SRAM.
- Total available memory: 64 kB.
- Maximum retained memory in hibernate mode: 32 kB.
- The data SRAM is composed of 32 kB. There is an option to retain 8 kB or 16 kB in hibernate mode.
- The instruction SRAM is composed of 32 kB. There is an option to retain 16 kB in hibernate mode.
- If instruction SRAM is not enabled, the associated 32 kB can be mapped as data SRAM. In this case, there is the option to retain 8 kB, 16 kB, 24 kB, or 32 kB of data SRAM.
- When the cache controller is enabled, 4 kB of the instruction SRAM is reserved for cache data. Those 4 kB of cache data are not retained in hibernate mode.
- Parity bit error detection (optional) is available on all SRAM memories. Two parity bits are associated with each 32-bit word. Parity check can be configured to be enabled or disabled in different memory regions. Parity is on by default.
- Byte, half word, and word accesses are supported.

SRAM

INSTRUCTION VS. DATA SRAM

If SRAM_CTL, Bit 31 is asserted, 32 kB of SRAM is mapped at Start Address 0x10000000 as two 16 kB instruction SRAMs (see Mode 0 and Mode 1 in Figure 54). The 32 kB of data SRAM is mapped in two sections, the first starting at 0x20000000 and the second starting at 0x20040000. If the cache memory feature is used, only 28 kB are available for instruction SRAM (see Mode 1 in Figure 54).

If SRAM_CTL, Bit 31 is 0 and cache is disabled, the 64 kB of SRAM is mapped as data SRAM. The memory is arranged in two sections. The first (32 kB) is mapped at Start Address 0x20000000 and the second (32 kB) at 0x20040000 (see Mode 2 in Figure 54). If cache memory feature is used, the second section only maps 28 kB. Therefore, the total data SRAM available is 60 kB (see Mode 3 in Figure 54).

By default, at power-up and hardware reset, the 32 kB of SRAM is made available as instruction SRAM. If the user must use a total of 64 kB data SRAM, SRAM_CTL, Bit 31 must be programmed to 0 at the start of the user code. When the cache controller is enabled, SRAM Bank 5 is not accessible. A bus error (unmapped address) is generated if an access is attempted.

SRAM RETENTION IN HIBERNATE MODE

The amount of SRAM retained during the hibernate mode varies based on user configuration. The content of the first 8 kB (Bank 0) of data SRAM is mapped at 0x20000000 and is always retained. The SRAM mapped from 0x20040000 onwards (Bank 3, Bank 4, and Bank 5) cannot be retained in the hibernate mode. If SRAM_CTL, Bit 1 is enabled, 8 kB of data SRAM mapped from 0x20002000 to 0x20003FFF (Bank 1) is retained in the hibernate mode. If SRAM_CTL, Bit 31 = 1 and SRAM_CTL, Bit 2 is enabled, 16 kB of instruction SRAM mapped from 0x10000000 to 0x10003FFF (Bank 2) is retained. If SRAM_CTL, Bit 31 = 0, 16 kB of data SRAM mapped from 0x20004000 to 0x20007FFF is retained.

SRAM Programming Model, Stack

The SRAM start address is set to 0x20000000 and the stack pointer is set at 0x20002000. The stack is written from 0x20001FFF downward. The covered memory region is always retained. To reserve a given size for the stack area, the user can declare a data array of that desired size ending at Position 0x20001FFF so that the stack is not overwritten by the compiler when allocating new variables.

SRAM Parity

For robustness, parity check can be enabled on all or a user selected group of SRAM banks. Parity check can detect up to two errors per word. The parity check feature can be enabled by asserting SRAM_CTL, Bits[5:0] for each SRAM bank. The kernel enables these bits by default.

Parity is checked when data is read and when byte or half word data is written. Parity is not checked when a word write (32 bits) is performed. If a parity error is detected, a bus error is generated. Even if parity error is detected when writing a byte or half word, the write operation is completed and parity bits are updated according to the new data. The user must manage the parity error in the bus fault interrupt routine.

SRAM INITIALIZATION

If parity check is enabled, SRAM contents must be initialized to avoid false parity errors. A dedicated hardware feature can automatically initialize the selected SRAM banks. This process takes 1024 HCLK cycles to complete. This hardware is fully programmable by the user, and as such, initialization can be started automatically or manually.

Initialization overwrites the contents of the selected SRAM banks. Therefore, initialization must be performed before writing to those SRAM blocks. If, during the initialization sequence, a write or read access to a SRAM bank being initialized is detected, the access is pending until the initialization sequence is completed. SRAM banks that are not selected to be initialized can be accessed as usual during the initialization process of the rest of the banks.

The initialization for a particular SRAM bank can be monitored for its completion by polling the appropriate SRAM_INITSTAT, Bits[5:0] in the SRAM_INITSTAT read only register. Every time a particular SRAM bank is initialized, its associated SRAM_INITSTAT, Bits[5:0] are cleared and remain low until initialization is completed.

After power-up, SRAM Bank 0 (8 kB) is automatically initialized. This memory is always retained and contains the stack pointer and critical information. Bank 0 contents do not have to be overwritten in the future, because initialization has already been performed. Avoid initializing SRAM banks that are already initialized, because they may already contain user information.

Initialization of more SRAM banks, where parity is enabled, can be achieved at any time by writing to the SRAM_CTL register. Set the appropriate SRAM_CTL, Bits[5:0] for SRAM banks that need parity to be enabled to 1. Also, set SRAM_CTL, Bit 13 to 1. SRAM_CTL, Bit 13 is autocleared to 0 after being written and triggers the initialization sequence.

SRAM

After exiting hibernate mode, the contents of nonretained SRAM banks are lost. If these banks have parity enabled, initialization is required. There are two options to initialize the required SRAM banks after exiting hibernate mode as follows:

- Initialize by writing to SRAM_CTL, Bit 13 after exiting hibernate mode. The SRAM banks that are set to 1 by SRAM_CTL, Bits[5:0] are initialized.
- Automatic initialization after hibernate mode. There is no write required to SRAM_CTL after hibernate mode. To select automatic mode, set SRAM_CTL, Bit 14 prior to hibernation or initialization. The SRAMs selected for initialization in this register automatically initialize after exiting hibernate mode.

Initialization resets the contents of the selected memory banks. It is important that the user carefully select which memory banks are initialized so no user information is lost. The initialization sequence can be aborted at any time by writing to SRAM_CTL, Bit 15. This bit is self cleared after it has been written.

CACHE

INITIALIZATION IN CACHE AND INSTRUCTION SRAM

Enabling the cache provides a significant performance increase for applications executing from flash. Cache memory coexists with SRAM. When cache is enabled, part of the SRAM is allocated to the cache memory, and as such, cache memory cannot be used for other purposes. Instruction cache size is 4 kB. On wakeup from hibernation, instruction cache contents are not retained. Instruction cache is disabled on power-up.

When cache memory is used, parity can also be enabled on its associated SRAM bank (Bank 5). When SRAM Bank 5 is used as cache memory, initialization is not required. Initialization is only required when performing byte or half word accesses to SRAM with parity check enabled. When SRAM Bank 5 is used as cache memory, all the accesses are word accesses. Because initialization is not required for the cache memory, the cache feature is available immediately following hibernate mode. There is no initialization time penalty. To prevent undesired bus errors, ignore any initialization of SRAM Bank 5 when cache is enabled.

As with cache memory, the SRAM banks used as instruction memory do not require any previous initialization when parity check is enabled. Consequently, the instruction SRAM is available immediately following hibernate mode. If initialization is triggered on instruction SRAM, it is committed. If instruction SRAM was initialized and an access to it was received, the access is halted until initialization is completed.

PROGRAMMING GUIDELINES

The sequence to enable cache is as follows:

1. Read Bit 0 in the FLCC_STAT register to ensure that the cache is disabled. Poll until this bit is cleared.
2. Write user key to the FLCC key register.
3. Set Bit 0 in the FLCC setup register to enable the cache.

See Table 29 for register details for the SRAM block.

REGISTER SUMMARY: CACHE (FLCC)

Table 242. FLCC Register Summary

Address	Name	Description	Reset	Access
0x40018058	STAT	Cache status	0x00000000	R
0x4001805C	SETUP	Cache setup	0x00000000	R/W
0x40018060	KEY	Cache key	0x00000000	W

REGISTER DETAILS: CACHE (FLCC)

CACHE STATUS REGISTER

Address: 0x40018058, Reset: 0x00000000, Name: STAT

Table 243. Bit Descriptions for STAT

Bits	Bit Name	Settings	Description	Reset	Access
[31:1]	Reserved		Reserved.	0x0000000	R
0	ICEN		Instruction Cache Enable.	0x0	R
		0	Disabled. All AHB accesses take place via flash memory.		
		1	Enabled. Instruction cache enabled for AHB accesses.		

CACHE SETUP REGISTER

Address: 0x4001805C, Reset: 0x00000000, Name: SETUP

The cache user key is required to enable a write to this location. The key is cleared after a write to this register.

Table 244. Bit Descriptions for SETUP

Bits	Bit Name	Settings	Description	Reset	Access
[31:1]	Reserved		Reserved.	0x0000000	R
0	ICEN		Instruction Cache Enable.	0x0	R/W
		0	Disabled. All AHB accesses take place via flash memory.		
		1	Enabled for AHB accesses.		

CACHE KEY REGISTER

Address: 0x40018060, Reset: 0x00000000, Name: KEY

Table 245. Bit Descriptions for KEY

Bits	Bit Name	Settings	Description	Reset	Access
[31:0]	VALUE		Cache Key Register. Enter 0xF123F456 to set the user key. Returns 0x0 if read. The key is cleared automatically after writing to the setup register.	0x00000000	W

SILICON IDENTIFICATION

The digital and analog dice of the ADuCM355 contain a chip ID and hardware revision register that can be read by software to allow users to determine the current revision of the silicon. The automated test equipment (ATE) test program, kernel revisions, and unique chip ID number can be read from the read only locations detailed in Table 246.

Table 246. Kernel and ATE Test Program Revision Details

Address	Description	Access
0x4074C	ATE test program revision, 16-bit value	R
0x40760	Kernel revision number, 16-bit value	R
0x40770	Unique 16-byte serial number for each device, 16-byte value	R

REGISTER SUMMARY: SYSTEM (DIGITAL DIE)

Table 247. System Register Summary

Address	Name	Description	Reset	Access
0x40002020	ADIID	Analog Devices identification (digital die)	0x4144	R
0x40002024	CHIPID	Chip identifier (digital die)	0x0284	R
0x40002040	SWDEN	Serial wire debug enable	0x6E65	W

Table 248. AFE Control Register Summary

Address	Name	Description	Reset	Access
0x400C0400	ADIID	Analog Devices identification (analog die)	0x4144	R
0x400C0404	CHIPID	Chip identification (analog die)	0x5502	R
0x400C0428	DIE2DIESTA	16-bit scratch register to test interdie communications	0xAA55	R/W

REGISTER DETAILS: SYSTEM (DIGITAL DIE)

ANALOG DEVICES IDENTIFICATION (DIGITAL DIE) REGISTER

Address: 0x40002020, Reset: 0x4144, Name: ADIID

Analog Devices Cortex device identification.

Table 249. Bit Descriptions for ADIID

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	VALUE		Analog Devices Identification. Reads a fixed value of 0x4144 to indicate to debuggers that they are connected to an Analog Devices implemented Cortex-based device.	0x4144	R

CHIP IDENTIFIER (DIGITAL DIE) REGISTER

Address: 0x40002024, Reset: 0x0284, Name: CHIPID

Chip identification for the digital die.

Table 250. Bit Descriptions for CHIPID

Bits	Bit Name	Settings	Description	Reset	Access
[15:4]	PARTID		Part Identifier.	0x28	R
[3:0]	REV		Silicon Revision.	0x4	R

SERIAL WIRE DEBUG ENABLE REGISTER

Address: 0x40002040, Reset: 0x6E65, Name: SWDEN

This register is used to enable the SWD interface. This register is reset upon an internal power-on or an external pin reset. This register is not affected by a software reset.

Table 251. Bit Descriptions for SWDEN

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	VALUE		Enable SWD Interface. Writing this register with en (0x6E65) or EN (0x4E45) enables the SWD interface. Writing this register with rp (0x7072) or RP (0x5052) disables the SWD interface. Writes of any other value are ignored. This register cannot be modified when written with EN or RP. This register is reset by a POR or pin reset, but not software or WDT reset.	0x6E65	W

ANALOG DEVICES IDENTIFICATION (ANALOG DIE) REGISTER

Address: 0x400C0400, Reset: 0x4144, Name: ADIID

Analog Devices identifier.

Table 252. Bit Descriptions for ADIID

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	ADIID		Analog Devices Identifier. Always equals 0x4144.	0x4144	R

CHIP IDENTIFICATION (ANALOG DIE) REGISTER

Address: 0x400C0404, Reset: 0x5502, Name: CHIPID

Table 253. Bit Descriptions for CHIPID

Bits	Bit Name	Settings	Description	Reset	Access
[15:4]	PARTID		Part Identifier.	0x550	R
[3:0]	Revision		Silicon Revision Number.	0x2	R

16-BIT SCRATCH REGISTER TO TEST INTERDIE COMMUNICATIONS REGISTER

Address: 0x400C0428, Reset: 0xAA55, Name: DIE2DIESTA

Table 254. Bit Descriptions for DIE2DIESTA

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	STA		User Programmable Scratch Register. The user can use this register to prove interdie communications.	0xAA55	R/W

DIGITAL INPUTS AND OUTPUTS

DIGITAL INPUTS AND OUTPUTS FEATURES

The ADuCM355 features multiple bidirectional GPIO pins (GPIOx/PWMx). Most of the GPIO pins have multiple functions, configurable by user code. On power-up, these pins are configured as tristate. There are three 16-bit wide ports. However, not all bits on some ports are accessible. Ignore inaccessible bits. All GPIO port pins provide input interrupt functionality. A typical GPIO pin structure is shown in Figure 55.

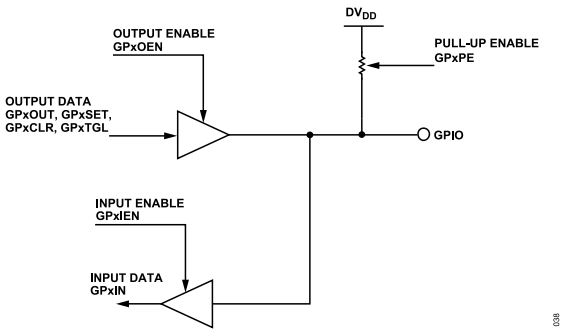

Figure 55. GPIO Structure for Pins with Pull-Up Resistors

DIGITAL INPUTS AND OUTPUTS OVERVIEW

The GPIOs are grouped into three ports, Port 0, Port 1, and Port 2. Each GPIO can be configured as an input, output, or fully open circuit. In input mode, the internal pull-up resistor or pull-down resistor can be enabled by software. All input and output pins are functional over the full supply range (DVDD = 2.8 V to 3.6 V (maximum)), and the GPIO low input voltage (V_{INL}) and GPIO high input voltage (V_{INH}) are specified as percentages of the supply as follows:

$$V_{INL} = 0.25 \times DVDD\ maximum \quad (21)$$

$$V_{INH} = 0.6 \times DVDD\ minimum \quad (22)$$

The absolute maximum input voltage is DVDD + 0.3 V. The typical leakage current of the GPIOs configured as input or open circuit is 10 nA per GPIO. When the ADuCM355 enters a power saving mode, the GPIO pins retain their states. In power saving mode, a driving peripheral cannot drive the pin. If the UART is driving the pin upon entry to deep sleep, it is isolated from the pin and power is gated. Its state and control are restored upon wakeup.

Some of the bits of Port 0, Port 1, and Port 2 are not bonded out of the package. The pin definitions in Table 255 indicate which are accessible. The inaccessible bits are still implemented. By default, these pins are tristate.

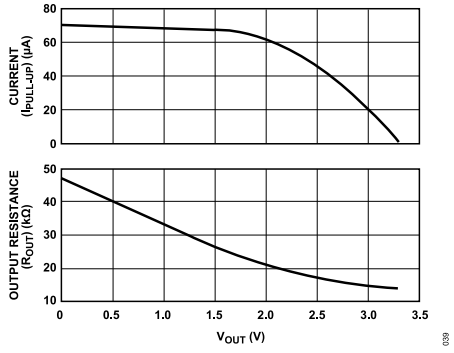

Figure 56. Typical Port 0, Port 1, and Port 2 Pull-Up Characteristics

DIGITAL INPUTS AND OUTPUTS OPERATION

Each digital input and output is configured, read, and written independent of the other bits.

General-Purpose Input Data (GPxIN)

The status of the GPIO pins can be read via the GPxIN registers when configured as inputs by the GPxIEN registers.

DIGITAL INPUTS AND OUTPUTS

General-Purpose Output Data (GPxOUT)

The values of the GPxOUT registers are output on the GPIO pins when configured as outputs by the GPxOEN registers.

Input/Output Data Out Enable (GPxOEN)

The GPxOEN registers enable the values of the GPxOUT registers to be output on the GPIO pins.

Input/Output Pull-Up Enable (GPxPE)

In input mode, the GPxPE registers enable and disable internal pull-up resistors. All Port 0, Port 1, and Port 2 pins have internal pull-up resistors. The pull-up resistors are implemented as metal-oxide semiconductor field effect transistors (MOSFETs), with typical performance shown in Figure 56.

GPIO Interrupt Enable (GPxIEN)

These registers enable the input pin interrupt sources for individual GPIO pins.

Bit Toggle Mode

Bit toggle mode toggles one or more GPIO data outputs without affecting other outputs within a port. Only the GPIO pins corresponding to 1 in the GPxTGL registers are toggled. The remaining GPIOs are unaffected.

INTERRUPTS

Each GPIO pin can be associated with an interrupt. Interrupts can be independently enabled for each GPIO pin and are always edge detecting. Only one interrupt is generated with each GPIO pin transition. The polarity of the detected edge can be positive (low to high) or negative (high to low). Each GPIO interrupt event can be mapped to one of two interrupts, Interrupt A or Interrupt B, allowing the system more flexibility in terms of how GPIO interrupts are grouped for servicing and how interrupt priorities are set. The interrupt status of each GPIO pin can be determined and cleared by accessing the GPxINT status registers. Set the appropriate bit in the GPxIEN registers to enable the full input path.

Interrupt Polarity

The polarity of the interrupt determines if the interrupt is accepted on the rising or the falling edge. Each GPIO pin has a corresponding interrupt register (GPxPOL) based on the port in which it is grouped. The interrupt registers configure the interrupt polarity of each pin. When set to 0, an interrupt event latches on a high to low transition on the corresponding pin. When set to 1, an interrupt event latches on a low to high transition on the corresponding pin.

Interrupt A Enable

Each GPIO port has a corresponding Interrupt Enable A register (GPxIENA) that is enabled or masked for each pin in the port. The bits in these registers determine if a latched edge event interrupts the core (Interrupt A) or is masked. In either case, the occurrence of the event is captured in the corresponding bit of the GPxINT status register. When set to 0, Interrupt A is not enabled (masked). No interrupts to the core are generated by this GPIO pin. When set to 1, Interrupt A is enabled. On a valid detected edge, an interrupt source to the core is generated.

Interrupt B Enable

Each GPIO port has a corresponding Interrupt Enable B (GPxIENB) register that is enabled or masked for each pin in the port. The bits in these registers determine if a latched edge event interrupts the core (Interrupt B) or is masked. In either case, the occurrence of the event is captured in the corresponding bit of the GPxINT status register. When set to 0, Interrupt B is not enabled (masked). No interrupts to the core are generated by this GPIO pin. When set to 1, Interrupt B is enabled. On a valid detected edge, an interrupt source to the core is generated.

Interrupt Status

Each GPIO port has an interrupt status register (GPxINT) that captures the interrupts occurring on its pins. These register bits indicate that the appropriately configured rising or falling edge has been detected on the corresponding GPIO pin.

When an event is detected, GPxINT remains set until cleared, even if the GPIO pin transitions back to a nonactive state. Out of reset, pull-up resistors combined with falling edge detect can result in the GPxINT status being cleared. However, this may not be the case if external circuits

DIGITAL INPUTS AND OUTPUTS

change the voltage level on the pin. Check the status of the GPxINT registers before enabling the GPxIENA and GPxIENB interrupts initially, as well as any time the GPIOx pins are configured.

Interrupt bits are cleared by writing 1 to the appropriate bit location in GPxINT. Writing 0 has no effect. If interrupts are enabled to the core (GPxIENA, GPxIENB), an interrupt GPxINT value of 1 results in an interrupt to the core. Clear this GPxINT bit during servicing of the interrupt. When GPxINT is read as 0, a rising or falling edge is not detected on the corresponding GPIO pin since this bit was last cleared. When read as 1, a rising or falling edge (GPxPOL selectable) is detected on the corresponding GPIO pin. This bit can be software cleared by writing 1 to the appropriate GPxINT bit.

The following is example code to enable BM/P1.1 as an input interrupt:

```
   pADI_GPIO1->PE = 0x2;                    // Enable internal pull-up resistors on P1.1
   pADI_GPIO1->IEN = 0x1;                   // Enable P1.1 input path
   pADI_GPIO1->IENA = 0x2;                 // Enable External Interrupt A on P1.11
   pADI_GPIO1->POL = 0x0;                     // Interrupt on falling edge
   NVIC_EnableIRQ(SYS_GPIO_INTA_IRQn);        // Enable GPIO_INTA interrupt source in NVIC
```

The following is example code for the GPIO pin interrupt handler routine:

```
void GPIO_A_Int_Handler()
{
   unsigned int uiIntSta = 0;
   uiIntSta = pADI_GPIO1->INT;
   if ((uiIntSta & 0x2) ==0x2)    // interrupt expected on P1.1
   {
       pADI_GPIO1->INT |= 0x2;
    }
}
```

The following is example code to set P0.3/SPI0_$\overline{CS}$ as an output. Write to GP0OUT, GP0SET, GP0CLR, and GP0TGL to set the level on P0.3/SPI0_$\overline{CS}$:

```
pADI_GPIO0->OEN |= 0x8;                 // Configure P0.3 as an output
```

DIGITAL DIE PORT MUX

This block provides control over the GPIO functionality of specified pins. Some pins have the ability to work as a GPIO or perform other specific functions. Only configuration of the P2.4 pin of Port 2 is permitted. Attempted writes to other port pins are not allowed. Any blank cells in Table 255 are not applicable.

Table 255. GPIO Multiplex Table

	Configuration Modes			
GPIO Pin	**00**	**01**	**10**	**11**
GP0 (GP0CON Controls These Bits)				
P0.0/SPI0_CLK	GPIO (GP0CON, Bits[1:0] = 0x0)	SPI0 serial clock (SCLK) (GP0CON, Bits[1:0] = 0x1)		
P0.1/SPI0_MOSI	GPIO (GP0CON, Bits[3:2] = 0x0)	SPI0 MOSI (GP0CON, Bits[3:2] = 0x1)		
P0.2/SPI0_MISO	GPIO (GP0CON, Bits[5:4] = 0x0)	SPI0 MISO (GP0CON, Bits[5:4] = 0x1)		
P0.3/SPI0_$\overline{CS}$	GPIO (GP0CON, Bits[7:6] = 0x0)	SPI0 chip select (GP0CON, Bits[7:6] = 0x1)		SPI1 ready (GP0CON, Bits[7:6] = 0x3)
P0.4/I2C_SCL	GPIO (GP0CON, Bits[9:8] = 0x0)	I2C I2C_SCL (GP0CON, Bits[9:8] = 0x1)		
P0.5/I2C_SDA	GPIO (GP0CON, Bits[11:10] = 0x0)	I2C I2C_SDA (GP0CON, Bits[11:10] = 0x1)		
P0.10/UART_SOUT	GPIO (GP0CON, Bits[21:20] = 0x0)	UART SOUT pin (GP0CON, Bits[21:20] = 0x1)		
P0.11/UART_SIN	GPIO (GP0CON, Bits[23:22] = 0x0)	UART SIN pin (GP0CON, Bits[23:22] = 0x1)		
GP1 (GP1CON Controls These Bits)				
P1.0/SYS_WAKE	GPIO (GP1CON, Bits[1:0] = 0x0)			

DIGITAL INPUTS AND OUTPUTS

Table 255. GPIO Multiplex Table (Continued)

	Configuration Modes			
GPIO Pin	**00**	**01**	**10**	**11**
BM/P1.1	GPIO boot (GP1CON, Bits[3:2] = 0x0)	GPIO (GP1CON, Bits[3:2] = 0x1)		
P1.2/SPI1_CLK	GPIO (GP1CON, Bits[5:4] = 0x0)	SPI1 SCLK (GP1CON, Bits[5:4] = 0x1)		
P1.3/SPI1_MOSI	GPIO (GP1CON, Bits[7:6] = 0x0)	SPI1 MOSI (GP1CON, Bits[7:6] = 0x1)		
P1.4/SPI1_MISO	GPIO (GP1CON, Bits[9:8] = 0x0)	SPI1 MISO (GP1CON, Bits[9:8] = 0x1)		
P1.5/SPI1_$\overline{CS}$	GPIO (GP1CON, Bits[11:10] = 0x0)	SPI1 chip select (GP1CON, Bits[11:10] = 0x1)		
P2.4	GPIO (GP2CON, Bits[9:8] = 0x0)			

AFE DIE DIGITAL PORT MUX

This block provides control over the three digital die GPIO pins. Two of these pins are bonded out as GPIO0/PWM0 and GPIO1/PWM1. The third pin, referred to as P2.2, is an internal pin bonded to the digital die P0.10/UART_SOUT pin and supports connecting the AFE die system clock to the digital die.

Table 256. AFE Die GPIO Mux Table

	Configuration Modes (GP2CON Controls These Bits)			
GPIO	**00**	**01**	**10**	**11**
GPIO0	Reserved	PWM0 (GP2CON, Bits[1:0] = 0x1)	GPIO (GP2CON, Bits[1:0] = 0x2)	Reserved
GPIO1	Reserved	PWM1 (GP2CON, Bits[3:2] = 0x1)	GPIO (GP2CON, Bits[3:2] = 0x2)	Reserved
P2.2 (Internal Only)	CLK_OUT (GP2CON, Bits[5:4] = 0x0)	Reserved	Reserved	Reserved

REGISTER SUMMARY: DIGITAL INPUTS AND OUTPUTS

Table 257. Digital Die GPIO Register Summary

Address	Name	Description	Reset	Access
0x40020000	GP0CON	GPIO Port 0 configuration	0x00000000	R/W
0x40020004	GP0OEN	GPIO Port 0 output enable	0x0000	R/W
0x40020008	GP0PE	GPIO Port 0 input/output pull-up enable	0x2	R/W
0x4002000C	GP0IEN	GPIO Port 0 input path enable	0x0000	R/W
0x40020010	GP0IN	GPIO Port 0 registered data input	0xXXXX	R
0x40020014	GP0OUT	GPIO Port 0 data output	0x0000	R/W
0x40020018	GP0SET	GPIO Port 0 data output set	0x0000	W
0x4002001C	GP0CLR	GPIO Port 0 data output clear	0x0000	W
0x40020020	GP0TGL	GPIO Port 0 pin toggle	0x0000	W
0x40020024	GP0POL	GPIO Port 0 interrupt polarity	0x0000	R/W
0x40020028	GP0IENA	GPIO Port 0 Interrupt A enable	0x0000	R/W
0x4002002C	GP0IENB	GPIO Port 0 Interrupt B enable	0x0000	R/W
0x40020030	GP0INT	GPIO Port 0 interrupt status	0x43C0	R/W
0x40020034	GP0DS	GPIO Port 0 drive strength select	0x0000	R/W
0x40020040	GP1CON	GPIO Port 1 configuration	0x00000000	R/W
0x40020044	GP1OEN	GPIO Port 1 output enable	0x0000	R/W
0x40020048	GP1PE	GPIO Port 1 input/output pull-up enable	0x0002	R/W
0x4002004C	GP1IEN	GPIO Port 1 input path enable	0x0002	R/W
0x40020050	GP1IN	GPIO Port 1 registered data input	0xXXXX	R
0x40020054	GP1OUT	GPIO Port 1 data output	0x0000	R/W
0x40020058	GP1SET	GPIO Port 1 data output set	0x0000	W
0x4002005C	GP1CLR	GPIO Port 1 data output clear	0x0000	W
0x40020060	GP1TGL	GPIO Port 1 pin toggle	0x0000	W
0x40020064	GP1POL	GPIO Port 1 interrupt polarity	0x0000	R/W
0x40020068	GP1IENA	GPIO Port 1 Interrupt A enable	0x0000	R/W
0x4002006C	GP1IENB	GPIO Port 1 Interrupt B enable	0x0000	R/W
0x40020070	GP1INT	GPIO Port 1 interrupt status	0x7800	R/W
0x40020074	GP1DS	GPIO Port 1 drive strength select	0x0000	R/W
0x40020080	GP2CON	GPIO Port 2 configuration	0x00000000	R/W
0x40020084	GP2OEN	GPIO Port 2 output enable	0x0000	R/W
0x40020088	GP2PE	GPIO Port 2 input/output pull-up enable	0x0000	R/W
0x4002008C	GP2IEN	GPIO Port 2 input path enable	0x0000	R/W
0x40020090	GP2IN	GPIO Port 2 registered data input	0x0XXX	R
0x40020094	GP2OUT	GPIO Port 2 data output	0x0000	R/W
0x40020098	GP2SET	GPIO Port 2 data output set	0x0000	W
0x4002009C	GP2CLR	GPIO Port 2 data output clear	0x0000	W
0x400200A0	GP2TGL	GPIO Port 2 pin toggle	0x0000	W
0x400200A4	GP2POL	GPIO Port 2 interrupt polarity	0x0000	R/W
0x400200A8	GP2IENA	GPIO Port 2 Interrupt A enable	0x0000	R/W
0x400200AC	GP2IENB	GPIO Port 2 Interrupt B enable	0x0000	R/W
0x400200B0	GP2INT	GPIO Port 2 interrupt status	0x4	R/W
0x400200B4	GP2DS	GPIO Port 2 drive strength select	0x0000	R/W

Table 258. AFE Die GPIO Register Summary

Address	Name	Description	Reset	Access
0x400C0080	CON	AFE GPIO port configuration	0x00	R/W
0x400C0084	OEN	AFE GPIO port output enable	0x0	R/W
0x400C0088	PE	AFE GPIO port output pull-up and pull-down enable	0x2	R/W
0x400C008C	IEN	AFE GPIO port input path enable	0x0	R/W

REGISTER SUMMARY: DIGITAL INPUTS AND OUTPUTS

Table 258. AFE Die GPIO Register Summary (Continued)

Address	Name	Description	Reset	Access
0x400C0090	IN	AFE GPIO port registered data input	0x0	R
0x400C0094	OUT	AFE GPIO port data output	0x0	R/W
0x400C0098	SET	AFE GPIO port data output set	0x0	W
0x400C009C	CLR	AFE GPIO port data output clear	0x0	W
0x400C00A0	TGL	AFE GPIO port pin toggle	0x0	W

REGISTER DETAILS: DIGITAL INPUTS AND OUTPUTS

Not all bits are accessible to the user on every port. Inaccessible bits are reserved. See Table 255 for more details on the accessible bits.

GPIO PORT CONFIGURATION REGISTERS

Address: 0x40020000, Reset: 0x00000000, Name: GP0CON

Address: 0x40020040, Reset: 0x00000000, Name: GP1CON

Address: 0x40020080, Reset: 0x00000000, Name: GP2CON

Table 259. Bit Descriptions for GP0CON, GP1CON, GP2CON

Bits	Bit Name	Settings	Description[1]	Access
[31:30]	CON15		Configuration Bits for Port x.15. See Table 255.	R/W
[29:28]	CON14		Configuration Bits for Port x.14. See Table 255.	R/W
[27:26]	CON13		Configuration Bits for Port x.13. See Table 255.	R/W
[25:24]	CON12		Configuration Bits for Port x.12. See Table 255.	R/W
[23:22]	CON11		Configuration Bits for Port x.11. See Table 255.	R/W
[21:20]	CON10		Configuration Bits for Port x.10. See Table 255.	R/W
[19:18]	CON9		Configuration Bits for Port x.9. See Table 255.	R/W
[17:16]	CON8		Configuration Bits for Port x.8. See Table 255.	R/W
[15:14]	CON7		Configuration Bits for Port x.7. See Table 255.	R/W
[13:12]	CON6		Configuration Bits for Port x.6. See Table 255.	R/W
[11:10]	CON5		Configuration Bits for Port x.5. See Table 255.	R/W
[9:8]	CON4		Configuration Bits for Port x.4. See Table 255.	R/W
[7:6]	CON3		Configuration Bits for Port x.3. See Table 255.	R/W
[5:4]	CON2		Configuration Bits for Port x.2. See Table 255.	R/W
[3:2]	CON1		Configuration Bits for Port x.1. See Table 255.	R/W
[1:0]	CON0		Configuration Bits for Port x.0. See Table 255.	R/W

[1] X is 0 for Port 0, 1 for Port 1, and 2 for Port 2.

GPIO PORT OUTPUT ENABLE REGISTERS

Address: 0x40020004, Reset: 0x0000, Name: GP0OEN

Address: 0x40020044, Reset: 0x0000, Name: GP1OEN

Address: 0x40020084, Reset: 0x0000, Name: GP2OEN

Table 260. Bit Descriptions for GP0OEN, GP1OEN, GP2OEN

Bits	Bit Name	Settings	Description	Access
[15:0]	OEN		Pin Output Drive Enable.	R/W
		0	Disable the output on the corresponding GPIO.	
		1	Enable the output on the corresponding GPIO.	

GPIO PORT INPUT/OUTPUT PULL-UP ENABLE REGISTERS

Address: 0x40020008, Reset: 0x2, Name: GP0PE

Address: 0x40020048, Reset: 0x0002, Name: GP1PE

Address: 0x40020088, Reset: 0x0000, Name: GP2PE

Table 261. Bit Descriptions for GP0PE, GP1PE, GP2PE

Bits	Bit Name	Settings	Description	Access
[15:0]	PUL		Pin Pull-Up Enable in Input and Output Mode.	R/W
		0	Disable the pull-up resistor on the corresponding GPIO.	
		1	Enable the pull-up resistor on the corresponding GPIO.	

REGISTER DETAILS: DIGITAL INPUTS AND OUTPUTS

GPIO PORT INPUT PATH ENABLE REGISTERS

Address: 0x4002000C, Reset: 0x0000, Name: GP0IEN

Address: 0x4002004C, Reset: 0x0002, Name: GP1IEN

Address: 0x4002008C, Reset: 0x0000, Name: GP2IEN

Table 262. Bit Descriptions for GP0IEN, GP1IEN, GP2IEN

Bits	Bit Name	Settings	Description	Access
[15:0]	IEN		Input Path Enable. Must be set for external interrupts and to read the pin value.	R/W
		0	Disable the input path on the corresponding GPIO.	
		1	Enable the input path on the corresponding GPIO.	

GPIO PORT REGISTERED DATA INPUT REGISTERS

Address: 0x40020010, Reset: 0xXXXX, Name: GP0IN

Address: 0x40020050, Reset: 0xXXXX, Name: GP1IN

Address: 0x40020090, Reset: 0x0XXX, Name: GP2IN

Table 263. Bit Descriptions for GP0IN, GP1IN, GP2IN

Bits	Bit Name	Settings	Description	Access
[15:0]	IN		Registered Data Input. Each bit reflects the state of the GPIO pin.	R

GPIO PORT DATA OUTPUT REGISTERS

Address: 0x40020014, Reset: 0x0000, Name: GP0OUT

Address: 0x40020054, Reset: 0x0000, Name: GP1OUT

Address: 0x40020094, Reset: 0x0000, Name: GP2OUT

Table 264. Bit Descriptions for GP0OUT, GP1OUT, GP2OUT

Bits	Bit Name	Settings	Description	Access
[15:0]	OUT		Data Output. Do not use the bit band alias addresses for this register.	R/W
		0	Cleared by user to drive the corresponding GPIO low.	
		1	Set by user code to drive the corresponding GPIO high.	

GPIO PORT DATA OUTPUT SET REGISTERS

Address: 0x40020018, Reset: 0x0000, Name: GP0SET

Address: 0x40020058, Reset: 0x0000, Name: GP1SET

Address: 0x40020098, Reset: 0x0000, Name: GP2SET

Table 265. Bit Descriptions for GP0SET, GP1SET, GP2SET

Bits	Bit Name	Settings	Description	Access
[15:0]	SET		Set the Output High. Do not use the bit band alias addresses for this register.	W
		0	Clearing this bit has no effect.	
		1	Set by user code to drive the corresponding GPIO high.	

REGISTER DETAILS: DIGITAL INPUTS AND OUTPUTS

GPIO PORT DATA OUTPUT CLEAR REGISTERS

Address: 0x4002001C, Reset: 0x0000, Name: GP0CLR

Address: 0x4002005C, Reset: 0x0000, Name: GP1CLR

Address: 0x4002009C, Reset: 0x0000, Name: GP2CLR

Table 266. Bit Descriptions for GP0CLR, GP1CLR, GP2CLR

Bits	Bit Name	Settings	Description	Access
[15:0]	CLR		Data Out Clear. Set the output low. Do not use the bit band alias addresses for this register.	W
		0	Clearing this bit has no effect.	
		1	Each bit is set to drive the corresponding GPIO pin low.	

GPIO PORT PIN TOGGLE REGISTERS

Address: 0x40020020, Reset: 0x0000, Name: GP0TGL

Address: 0x40020060, Reset: 0x0000, Name: GP1TGL

Address: 0x400200A0, Reset: 0x0000, Name: GP2TGL

Table 267. Bit Descriptions for GP0TGL, GP1TGL, GP2TGL

Bits	Bit Name	Settings	Description	Access
[15:0]	TGL		Toggle the Output of the Port Pin. Do not use the bit band alias addresses for this register.	W
		0	Clearing this bit has no effect.	
		1	Set by user code to invert the corresponding GPIO pin.	

GPIO PORT INTERRUPT POLARITY REGISTERS

Address: 0x40020024, Reset: 0x0000, Name: GP0POL

Address: 0x40020064, Reset: 0x0000, Name: GP1POL

Address: 0x400200A4, Reset: 0x0000, Name: GP2POL

Table 268. Bit Descriptions for GP0POL, GP1POL, GP2POL

Bits	Bit Name	Settings	Description	Access
[15:0]	POL		GPIO Pin Interrupt Polarity Control.	R/W
		0	Clearing this bit for pin interrupts triggered on a high to low transition.	
		1	Set this bit for pin interrupts triggered on a low to high transition.	

GPIO PORT INTERRUPT A ENABLE REGISTERS

Address: 0x40020028, Reset: 0x0000, Name: GP0IENA

Address: 0x40020068, Reset: 0x0000, Name: GP1IENA

Address: 0x400200A8, Reset: 0x0000, Name: GP2IENA

Table 269. Bit Descriptions for GP0IENA, GP1IENA, GP2IENA

Bits	Bit Name	Settings	Description	Access
[15:0]	INTAEN		GPIO Pin Interrupt A Enable Register.	R/W
		0	Clear this bit to disable the corresponding pin interrupt.	
		1	Set this bit to enable the corresponding pin interrupt.	

REGISTER DETAILS: DIGITAL INPUTS AND OUTPUTS

GPIO PORT INTERRUPT B ENABLE REGISTERS

Address: 0x4002002C, Reset: 0x0000, Name: GP0IENB

Address: 0x4002006C, Reset: 0x0000, Name: GP1IENB

Address: 0x400200AC, Reset: 0x0000, Name: GP2IENB

Table 270. Bit Descriptions for GP0IENB, GP1IENB, GP2IENB

Bits	Bit Name	Settings	Description	Access
[15:0]	INTBEN		GPIO Pin Interrupt B Enable Register.	R/W
		0	Clear this bit to disable the corresponding pin interrupt.	
		1	Set this bit to enable the corresponding pin interrupt.	

GPIO PORT INTERRUPT STATUS REGISTERS

Address: 0x40020030, Reset: 0x43C0, Name: GP0INT

Address: 0x40020070, Reset: 0x7800, Name: GP1INT

Address: 0x400200B0, Reset: 0x4, Name: GP2INT

Table 271. Bit Descriptions for GP0INT, GP1INT, GP2INT

Bits	Bit Name	Settings	Description	Access
[15:0]	INTSTATUS		GPIO Pin Interrupt Status Register.	R/W1C
		0	Indicates no interrupt on the corresponding pin.	
		1	When set, this bit indicates the corresponding pin interrupt event has been latched. To clear this bit and interrupt event, write 1 to the same bit. Writing 0 has no effect.	

GPIO PORT DRIVE STRENGTH SELECT REGISTERS

Address: 0x40020034, Reset: 0x0000, Name: GP0DS

Address: 0x40020074, Reset: 0x0000, Name: GP1DS

Address: 0x400200B4, Reset: 0x0000, Name: GP2DS

Table 272. Bit Descriptions for GP0DS, GP1DS, GP2DS

Bits	Bit Name	Settings	Description	Access
[15:0]	DS		Drive Strength Control. Do not use the bit band alias addresses for this register.	R/W
		0	Clear this bit for normal drive strength.	
		1	Set by user code for maximum drive strength capability on the corresponding GPIO pin.	

AFE GPIO PORT CONFIGURATION REGISTER

Address: 0x400C0080, Reset: 0x00, Name: CON

Table 273. Bit Descriptions for CON

Bits	Bit Name	Settings	Description	Reset	Access
[15:4]	Reserved		Reserved.	0x000	R/W
[3:2]	CON1		GPIO1/PWM1 Configuration Bits.	0x0	R/W
		00	PWM1.		
		10	GPIO.		
[1:0]	CON0		GPIO0/PWM0 Configuration Bits.	0x0	R/W
		00	PWM0.		
		01	GPIO.		

REGISTER DETAILS: DIGITAL INPUTS AND OUTPUTS

AFE GPIO PORT OUTPUT ENABLE REGISTER

Address: 0x400C0084, Reset: 0x0, Name: OEN

Table 274. Bit Descriptions for OEN

Bits	Bit Name	Settings	Description	Reset	Access
[15:3]	Reserved		Reserved.	0x0000	R/W
2	OEN1		Output Enable for AFE Die Clock to Digital Die. The AFE die Pad P2.2 is internally connected to the digital die internal Pad P1.10.	0	R/W
		0	Disconnects AFE die clock path from digital die P1.10 internal pad.		
		1	Enables path for AFE die clock to digital die.		
[1:0]	OEN0		Pin Output Drive Enable. Each bit is set to enable the output for that particular pin. These bits are cleared to disable the output for each pin.	0x0	R/W

AFE GPIO PORT OUTPUT PULL-UP AND PULL-DOWN ENABLE REGISTER

Address: 0x400C0088, Reset: 0x2, Name: PE

Table 275. Bit Descriptions for PE

Bits	Bit Name	Settings	Description	Reset	Access
[15:3]	Reserved		Reserved.	0x0000	R/W
[2:0]	PE		Pin Pull Enable. Each bit is set to enable the pull-up resistor and pull-down resistor for that particular pin. The bit is cleared to disable the pull-up resistor and pull-down resistor for each pin.	0x0	R/W

AFE GPIO PORT INPUT PATH ENABLE REGISTER

Address: 0x400C008C, Reset: 0x0, Name: IEN

Table 276. Bit Descriptions for IEN

Bits	Bit Name	Settings	Description	Reset	Access
[15:2]	Reserved		Reserved.	0x0000	R/W
[1:0]	IEN		Input Path Enable. Each bit is set to enable the input path and is cleared to disable the input path for the GPIO pin.	0x0	R/W

AFE GPIO PORT REGISTERED DATA INPUT

Address: 0x400C0090, Reset: 0x0, Name: IN

Table 277. Bit Descriptions for IN

Bits	Bit Name	Settings	Description	Reset	Access
[15:2]	Reserved		Reserved.	0x0000	R/W
[1:0]	IN		Registered Data Input. Each bit reflects the state of the GPIO pin if the corresponding input buffer is enabled. If the pin input buffer is disabled, the value shown is zero.	0x0	R

AFE GPIO PORT DATA OUTPUT REGISTER

Address: 0x400C0094, Reset: 0x0, Name: OUT

Table 278. Bit Descriptions for OUT

Bits	Bit Name	Settings	Description	Reset	Access
[15:2]	Reserved		Reserved.	0x0000	R/W
[1:0]	OUT		Data Out. Set by user code to drive the corresponding GPIO high. Cleared by user to drive the corresponding GPIO low.	0x0	R/W

REGISTER DETAILS: DIGITAL INPUTS AND OUTPUTS

AFE GPIO PORT DATA OUTPUT SET REGISTER

Address: 0x400C0098, Reset: 0x0, Name: SET

Table 279. Bit Descriptions for SET

Bits	Bit Name	Settings	Description	Reset	Access
[15:2]	Reserved		Reserved.	0x0000	R/W
[1:0]	SET		Set Output High for the AFE Die Port Pins (GPIOx/PWMx). Set by user code to drive the corresponding GPIO high. Clearing this bit has no effect.	0x0	W

AFE GPIO PORT DATA OUTPUT CLEAR REGISTER

Address: 0x400C009C, Reset: 0x0, Name: CLR

Table 280. Bit Descriptions for CLR

Bits	Bit Name	Settings	Description	Reset	Access
[15:2]	Reserved		Reserved.	0x0000	R/W
[1:0]	CLR		Set Output Low for the AFE Die Port Pins (GPIOx/PWMx). Each bit is set to drive the corresponding GPIO pin low. Clearing this bit has no effect.	0x0	W

AFE GPIO PORT PIN TOGGLE REGISTER

Address: 0x400C00A0, Reset: 0x0, Name: TGL

Table 281. Bit Descriptions for TGL

Bits	Bit Name	Settings	Description	Reset	Access
[15:2]	Reserved		Reserved.	0x0000	R/W
[1:0]	TGL		Toggle Output of the Port Pin. Each bit is set to invert the corresponding GPIO pin. Clearing this bit has no effect.	0x0	W

I²C SERIAL INTERFACE

I²C FEATURES

The I²C interface features an initiator or target mode with 2-byte transmit and receive FIFOs. The I²C interface supports 7-bit and 10-bit addressing modes, either as four 7-bit device addresses or a combination of one 10-bit address and two 7-bit addresses in the target with repeated starts in initiator and target modes. Other devices on the bus can enable clock stretching without causing any issues with the ADuCM355. Initiator arbitration, continuous read mode for the initiator or up to 512 bytes, fixed read, and internal and external loopback are also available.

Support for DMA in initiator and target modes is provided by the DMA controller, as well as software control on the target of the no acknowledge signal.

I²C OVERVIEW

The I²C data transfer uses a serial clock pin (I2C_SCL) and a serial data pin (I2C_SDA). These pins are configured in a wire-AND'ed gated format that allows arbitration in a multiinitiator system.

The transfer sequence of the I²C system is initiated by an initiator device, which generates a start condition while the bus is idle. The initiator transmits the target device address and the direction of the data transfer during the initial address transfer. If the initiator does not lose arbitration and the target acknowledges the initial address transfer, the data transfer is initiated. This transfer continues until the initiator issues a stop condition and the bus becomes idle. Figure 57 shows a typical I²C transfer.

An initiator device can be configured to generate the serial clock. The frequency is programmed by the user in the serial clock divisor register, DIV. The initiator channel can be set to operate in fast mode (400 kHz) or standard mode (100 kHz).

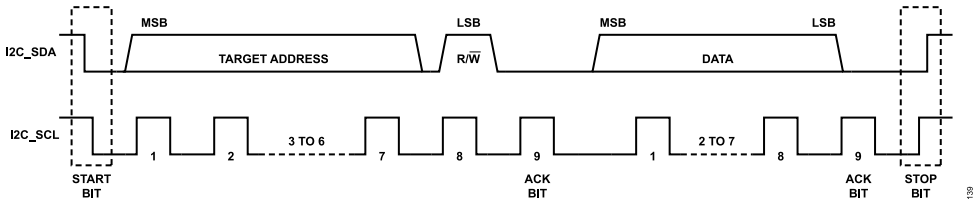

Figure 57. Typical I²C Transfer Sequence

The user programs the I²C bus peripheral address in the I²C bus system. This ID can be modified any time a transfer is not in progress. The user can set up to four target addresses that are recognized by the peripheral. The peripheral is implemented with a 2-byte FIFO for each transmit and receive shift register. The IRQ and status bits in the control registers are available to signal to the processor core when the FIFOs need to be serviced.

I²C OPERATION

I²C Startup

The following steps are required to run the I²C peripheral:

1. Enable PCLK to the I²C peripheral by setting CTL5, Bit 5 and CTL5, Bit 3 = 0. PCLK frequency is controlled via CTL1, Bits[13:8].
2. Configure the digital pins (P0.4/I2C_SCL and P0.5/I2C_SDA) for I²C operation via the GP0CON register.
3. Ensure the drive strength for the P0.4/I2C_SCL pin and P0.5/I2C_SDA pin is increased to ensure reliable I²C communications, as detailed in the following example code:

```
GP0DS = 0x30;  // Increase drive strength of I2C pins.
```

4. Configure the I²C registers as required for target or initiator operation.
5. Enable the I²C target or initiator interrupt source as required.

When using I²C, the user must disable the internal pull-up resistors on the I²C pins via the GP0POL register. The GPIO multiplexed configuration mode is I2C_SCL for P0.4/I2C_SCL and is I2C_SDA for P0.5/I2C_SDA.

I²C SERIAL INTERFACE

7-Bit Addressing

The ID0 register, ID1 register, ID2 register, and ID3 register contain the target device IDs. The ADuCM355 compares the four IDx registers to the address byte. To be correctly addressed, the seven MSBs of IDx registers must be identical to that of the seven MSBs of the first received address byte. The LSB of the IDx registers (R/W or the transfer directional bit) is ignored in the process of address recognition. The initiator addresses a device using the ADR1 register.

10-Bit Addressing

This feature is enabled by setting SCTL, Bit 1 for initiator and target mode. The 10-bit address of the target is stored in the ID0 register and ID1 register, where the ID0 register contains the first byte of the address, and the R/W bit and the upper five bits must be programmed to 11110, as shown in Figure 58. The ID1 register contains the remaining eight bits of the 10-bit address. The ID2 register and ID3 register can still be programmed with 7-bit addresses.

The initiator communicates to a 10-bit address target using the ADR1 register and ADR2 registers. The format is described in Figure 58. To perform a read from a target with a 10-bit address, the initiator must first send a 10-bit address with the R/W bit cleared. It must then generate a repeated start and send only the first byte of the address with the R/W bit set. A repeated start is generated by writing to the ADR1 register while the initiator is still busy.

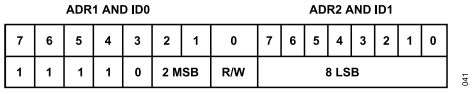

Figure 58. 10-Bit Address Format

A repeated start condition occurs when a second start condition is sent to a target without a stop condition being sent in between. This sequence allows the initiator to reverse the direction of the transfer by changing the R/W bit without having to give up control of the bus. An example of a transfer sequence is shown in Figure 59. This sequence is generally used in cases where the ability of the register to be read is established by the first data sent to the device.

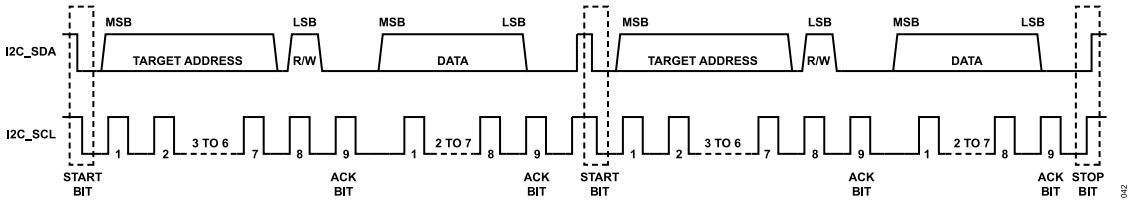

Figure 59. I²C Repeated Start Sequence

On the target side, an interrupt is generated (if enabled in the SCTL register) when a repeated start and a target address are received. This sequence is differentiated from receiving a start and target address by using the start and REPSTART status bits in the SSTAT MMR.

On the initiator side, the initiator generates a repeated start if the ADR1 register is written while the initiator is still busy with a transaction. After the I²C state machine has started to transmit the device address, it is safe to write to the ADR1 register.

For example, if a transaction involving a write, a repeated start, and then a read/write is required, write to the ADR1 register either after the state machine starts to transmit the device address or after the first MTXREQ interrupt is received. When the transmit FIFO empties, a repeated start is generated. Similarly, if a transaction involving a read, a repeated start, and then a read/write is required, write to the first initiator address byte register, ADR1, either after the state machine starts to transmit the device address or after the first MRXREQ interrupt is received. When the requested receive count is reached, a repeated start is generated.

I²C Clock Control

A PCLK clocks the I²C peripherals. CTL5, Bit 5 and CTL5, Bit 3 must be cleared to enable the clock to the I²C block. The frequency of PCLK is determined by CTL1, Bits[13:8]. The I²C initiator in the system generates the serial clock for a transfer. The initiator channel can be configured to operate in fast mode (400 kHz) or standard mode (100 kHz).

The bit rate is defined in the DIV MMR as follows:

$$f_{SCL} = f_{I2CCLK}/(Low + High + 3) \quad (23)$$

where:
f_{SCL} is the I²C baud rate.
f_{I2CCLK} is the PCLK frequency.
Low is the low period of the clock, DIV[7:0].
High is the high period of the clock, DIV[15:8].

High = *REQD_HIGH_TIME*/*PCLK_PERIOD* – 2

where:
REQD_HIGH_TIME is the required high time period.
PCLK_PERIOD is the PCLK period.

Low = *REQD_LOW_TIME*/*PCLK_PERIOD* – 1

where *REQD_LOW_TIME* is the required low time period.

For 100 kHz SCL operation with a low time of 5 µs, a high time of 5 µs, and a PCLK frequency of 26 MHz,

High = (5 µs/(1/26,000,000)) − 2 = 128 = 0x80 (24)

Low = (5 µs/(1/26,000,000)) − 1 = 129 = 0x81 (25)

f_{SCL} = 26,000,000/(128 + 129 + 3) = 100 kHz (26)

Resetting the I²C Block

Three steps are needed to reset the I²C block. Do not reset the I²C peripheral on two consecutive communication sequences.

In initiator mode, the steps are as follows:

1. Clear MCTL, Bit 0 to 0 and disable the I²C initiator.
2. Set SHCTL, Bit 0 to 1, which is a write only register. Writing to this bit resets the start and stop detection circuits of the I²C block and clears MSTAT, Bit 10.
3. Set MCTL, Bit 0 to 1 to reenable the I²C initiator.

In target mode, the steps are as follows:

1. Clear SCTL, Bit 0 to 0 and disable the I²C target.
2. Set SHCTL, Bit 0 to 1, which is a write only register. Writing to this bit resets the start and stop detection circuits of the I²C block.
3. Set SCTL, Bit 0 to 1 to reenable the I²C target.

I²C OPERATING MODES

Initiator Transfer Initiation

If the initiator enable bit (MCTL, Bit 0) is set, an initiator transfer sequence is initiated by writing a value to the ADR1 register. If there is valid data in the MTX register, it is the first byte transferred in the sequence after the address byte during a write sequence.

Target Transfer Initiation

If the target enable bit (SCTL, Bit 0) is set, a target transfer sequence is monitored for the device address in Register ID0, Register ID1, Register ID2, or Register ID3. If the device address is recognized, the device participates in the target transfer sequence.

Note that a target operation always starts with the assertion of one of three interrupt sources: a read request (MRXREQ, SRXREQ), a write request (MTXREQ, STXREQ), or a general call interrupt (GCINT). The software must always look for a stop interrupt to ensure that the transaction has completed correctly and to deassert the stop interrupt status bit.

Receive and Transmit Data FIFOs

The transmit data path consists of an initiator and target transmit FIFO (each two bytes deep), the MTX register and STX register, and a transmit shifter. The transmit status bits, MSTAT, Bits[1:0], and SSTAT, Bit 0 denote whether there is valid data in the transmit FIFO. Data from

I²C SERIAL INTERFACE

the transmit FIFO is loaded into the transmit shifter when a serial byte begins transmission. If the transmit FIFO is not full during an active transfer sequence, the transmit request bit (MSTAT, Bit 2 or SSTAT, Bit 2) asserts. Figure 60 shows the effect of not having the target transmit FIFO full at the start of a read request from an initiator. An extra transmit interrupt can be generated after the read bit. This extra transmit interrupt occurs if the transmit FIFO is not full.

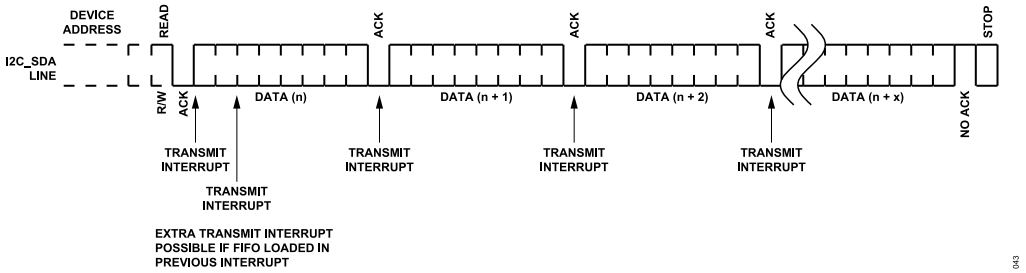

Figure 60. I²C Target Transmit Interrupt Details

In the target, if there is no valid data to transmit when the transmit shifter is loaded, the transmit underflow status bit asserts (MSTAT, Bit 12 or SSTAT, Bit 1). In target mode, the transmit FIFO must be loaded with a byte before the falling edge of I2C_SCL and before the acknowledge or no acknowledge is asserted. If the transmit FIFO is empty on the falling edge of I2C_SCL for an R/W bit, the target returns a no acknowledge because the target in this case controls the acknowledge or no acknowledge.

If the first byte is transmitted correctly in a target transmit sequence, but the transmit FIFO is empty for any subsequent bytes in the same transfer, the target returns the previous transmitted byte. This operation is due to the initiator having control of the acknowledge or no acknowledge during a target transfer sequence. The initiator generates a stop condition if there is no data in the transmit FIFO and the initiator is writing data.

The receive data path consists of an initiator and target receive FIFO, the MRX register, and the SRX register. Both are two bytes deep. The receive request interrupt bit (MSTAT, Bit 3 or SSTAT, Bit 3) indicates whether there is valid data in the receive FIFO. Data is loaded into the receive FIFO after each byte is received. If valid data in the receive FIFO is overwritten by the receive shifter, the receive overflow status bit is asserted (MSTAT, Bit 9 or SSTAT, Bit 4).

Automatic Clock Stretching

The ASTRETCH_SCL register controls automatic clock stretching. If automatic clock stretching is enabled, the I²C hardware holds the I2C_SCL pin low after the falling edge of I2C_SCL before an acknowledge or no acknowledge under the following conditions:

- The transmit FIFO is empty when a valid read request is active for the initiator or target.
- The receive FIFO is full when another byte is about to be received. If the receive FIFO has still not been read at the end of the timeout period, a no acknowledge is returned, and the initiator ends the sequence with a stop condition.

When enabling automatic clock stretching, enable the timeout feature to support recovery from incomplete data transfers. A separate status bit for initiator and target mode indicates if a stretch timeout has occurred. It is recommended that automatic clock stretching be enabled, especially in target mode.

If the transmit FIFO is empty on the falling edge of I2C_SCL for an R/W bit at the end of the timeout period, the target returns a no acknowledge after the timeout period. If the first byte is transmitted correctly in a target transmit sequence, but the transmit FIFO is empty for any subsequent bytes in the same transfer with clock stretch enabled, the target returns the previous transmitted byte at the end of the timeout period.

Initiator No Acknowledge

When receiving data, the initiator responds with a no acknowledge if its FIFO is full and an attempt is made to write another byte to the FIFO. This last byte received is not written to the FIFO and is lost.

No Acknowledge from the Target

If the target does not want to acknowledge a read access, not writing data into the target transmit FIFO results in a no acknowledge. If the target does not want to acknowledge an initiator write, assert the no acknowledge bit in the target control register, SCTL, Bit 7.

I²C SERIAL INTERFACE

Normally, the target acknowledges all bytes written into the receive FIFO. If the receive FIFO fills up, the target cannot write further bytes to it, and the target does not acknowledge subsequent bytes not written to the FIFO. The initiator must then stop the transaction.

The target does not acknowledge a matching device address if the R/W bit is set and the transmit FIFO is empty. Therefore, there is very little time for the microcontroller to respond to a target transmit request and the assertion of an acknowledge. It is recommended that SCTL, Bit 5 be asserted for this reason.

General Call

An I²C general call is for addressing every device on the I²C bus. A general call address is 0x00 or 0x01. The first byte, the address byte, is followed by a command byte.

If the address byte is 0x00, Byte 2 (the command byte) can be one of the following:

- 0x6. The I²C interface (initiator and target) is reset. The general call interrupt status asserts, and the general call ID bits, SSTAT, Bits[9:8], are 0x1. User code must take corrective action to reset the entire system or simply reenable the I²C interface.
- 0x4. The general call interrupt status bit is asserted, and the general call ID bits (SSTAT, Bits[9:8]) are 0x2.

If the address byte is 0x01, a hardware general call is issued. Byte 2 in this case is the hardware initiator address.

The general call interrupt status bit is set on any general call after the second byte is received, and user code must take corrective action to reprogram the device address.

If SCTL, Bit 2 is set to 1, the target always acknowledges the first byte of a general call. The target acknowledges the second byte of a general call if the second byte is 0x04 or 0x06, or if the second byte is a hardware general call and SCTL, Bit 3 is set to 1.

The ALT register contains the alternate device ID for a hardware general call sequence. If the hardware general call enable bit, the general call enable bit, and the target enable bit are all set (HGCEN, GCEN, and SLVEN in the SCTL register), the device recognizes a hardware general call. When a general call sequence is issued and the second byte of the sequence is identical to the ALT register, the hardware call sequence is recognized for the device.

I²C Reset Mode

The target state machine is reset when SCTL, Bit 0 is written to 0. The initiator state machine is reset when MCTL, Bit 0 is written to 0.

I²C Test Modes

The device can be placed in an internal loopback mode by setting MCTL, Bit 2. There are four FIFOs (initiator transmit and receive, and target transmit and receive). Therefore, the I²C peripheral can, in effect, be set up to communicate with itself. External loopback can be performed if the initiator is set up to address the target address.

I²C Low-Power Mode

If the initiator and target are both disabled (MCTL, Bit 0 = SCTL, Bit 0 = 0), the I²C section is off. To fully power down the I²C block, disable the clock to the I²C section of the chip by setting CTL5, Bit 3 = 1.

Power-Down Considerations

If the initiator or target is idle, they can be immediately disabled by clearing MCTL, Bit 0 or SCTL, Bit 0 in the initiator or target control registers, respectively. However, if the initiator or target is active, there are four possible events occurring and, therefore, four ways to power down the device, as follows:

- I²C is an initiator and is receiving data. The device is receiving data based on the count programmed in the MRXCNT register. It is in continuous read mode if the MRXCNT, Bit 8 is set. To stop the read transfer, clear this bit and assign the MRXCNT register with MCRXCNT, Bits[7:0] + 1, where MCRXCNT, Bits[7:0] give the current read count. The addition of + 1 signifies that there is some room for the completion. If the newly programmed value is less than the current count, the I²C initiator receives until the current count overflows and reaches the programmed count. When this overflow occurs, the transfer ends after receiving the next byte. When the transaction complete interrupt is received, the core disables the initiator by clearing MCTL, Bit 0.

I²C SERIAL INTERFACE

- I²C is an initiator and is transmitting data. The software flushes the transmit FIFO by setting STAT, Bit 9 and disables the transmit request by clearing MCTL, Bit 5. When the transmit request is disabled, the current transfer ends after transmitting the byte in progress. When the transaction complete interrupt is received, the software clears MCTL, Bit 0. Disabling the initiator before completion can cause the bus to stall indefinitely.
- I²C is a target and is receiving data. The software sets SCTL, Bit 7, giving a no acknowledge for the next communication, after which the external initiator must stop. On receiving the stop interrupt, the core disables the target by clearing SCTL, Bit 0.
- I²C is a target and is transmitting data. After the target transmits starts, it cannot no acknowledge any further transactions, because the acknowledge is driven only by the initiator. Therefore, the target transmit must wait until the external initiator issues a stop condition. After receiving the stop interrupt, the target can be disabled. However, if the target must be disabled immediately, such an action can only be performed at the cost of wrong data being transmitted (all 0xFFs) because the I2C_SDA line is not driven anymore and is pulled up during data phase. The bus does not stall in this case.

DMA Requests

Four DMA channels are provided to service the I²C initiator and target. The DMA enabled bits are provided in the target control register and in the initiator control register.

I²C Pins when ADuCM355 is Unpowered

When the ADuCM355 is not powered up, do not apply logic high signals to any digital pins. The maximum voltage that can be applied to a digital input pin at any time is DVDD + 0.3 V. If this limit is exceeded, the ESD protection diodes start to conduct to ground. If the ADuCM355 is unpowered but the I²C bus pins are at a logic high, the ADuCM355 pin protection structure pulls I2C_SCL and I2C_SDA toward ground, causing issues for devices that are powered on the bus.

It is recommended that all devices with an I²C bus (including the ADuCM355) are fully powered up before any communications is started on the bus.

REGISTER SUMMARY: I²C

Table 282. I²C Register Summary

Address	Name	Description	Reset	Access
0x40003000	MCTL	Initiator control	0x0000	R/W
0x40003004	MSTAT	Initiator status	0x6000	R
0x40003008	MRX	Initiator receive data	0x0000	R
0x4000300C	MTX	Initiator transmit data	0x0000	R/W
0x40003010	MRXCNT	Initiator receive data count	0x0000	R/W
0x40003014	MCRXCNT	Initiator current receive data count	0x0000	R
0x40003018	ADR1	First initiator address byte	0x0000	R/W
0x4000301C	ADR2	Second initiator address byte	0x0000	R/W
0x40003024	DIV	Serial clock period divisor	0x1F1F	R/W
0x40003028	SCTL	Target control	0x0000	R/W
0x4000302C	SSTAT	Target I²C status, error, and IRQ	0x0001	R
0x40003030	SRX	Target receive	0x0000	R
0x40003034	STX	Target transmit	0x0000	R/W
0x40003038	ALT	Hardware general call ID	0x0000	R/W
0x4000303C	ID0	First target address device ID	0x0000	R/W
0x40003040	ID1	Second target address device ID	0x0000	R/W
0x40003044	ID2	Third target address device ID	0x0000	R/W
0x40003048	ID3	Fourth target address device ID	0x0000	R/W
0x4000304C	FSTAT	Initiator and target FIFO status	0x0000	R/W
0x40003050	SHCTL	Initiator and target shared control	0x0000	W
0x40003058	ASTRETCH_SCL	Automatic stretch control for initiator and target mode	0x0000	R/W

REGISTER DETAILS: I²C

INITIATOR CONTROL REGISTER

Address: 0x40003000, Reset: 0x0000, Name: MCTL

Table 283. Bit Descriptions for MCTL

Bits	Bit Name	Settings	Description	Reset	Access
[15:12]	Reserved		Reserved.	0x0	R
11	MTXDMA		Enable Initiator Transmit DMA Request.	0x0	W
		0	Disable DMA mode.		
		1	Enable I²C initiator DMA transmit requests.		
10	MRXDMA		Enable Initiator Receive DMA Request.	0x0	W
		0	Disable DMA mode.		
		1	Enable I²C initiator DMA receive requests.		
9	Reserved		Reserved.	0x0	R/W
8	IENCMP		Transaction Completed (or Stop Detected) Interrupt Enable.	0x0	R/W
		0	Interrupt is not generated when a stop is detected.		
		1	Interrupt is generated when a stop is detected.		
7	IENACK		Acknowledge Not Received Interrupt Enable.	0x0	R/W
		0	Acknowledge not received interrupt disable.		
		1	Acknowledge not received interrupt enable.		
6	IENALOST		Arbitration Lost Interrupt Enable.	0x0	R/W
		0	Arbitration lost interrupt disable.		
		1	Arbitration lost interrupt enable.		
5	IENMTX		Transmit Request Interrupt Enable.	0x0	R/W
		0	Transmit request interrupt disable.		
		1	Transmit request interrupt enable.		
4	IENMRX		Receive Request Interrupt Enable.	0x0	R/W
		0	Receive request interrupt disable.		
		1	Receive request interrupt enable.		
3	Reserved		Reserved. Write 0 to this bit.	0x0	R/W
2	LOOPBACK		Internal Loopback Enable. It is also possible for the initiator to loop back a transfer to the target as long as the device address corresponds, otherwise known as external loopback.	0x0	R/W
		0	I2C_SCL and I2C_SDA out of the device are not muxed onto their corresponding inputs.		
		1	I2C_SCL and I2C_SDA out of the device are muxed onto their corresponding inputs.		
1	COMPETE		Start Back Off Disable. Setting this bit enables the device to compete for ownership even if another device is currently driving a start condition.	0x0	R/W
0	MASEN		Initiator Enable. The initiator must be disabled to gate the clock to the initiator when not in use and save power. Do not clear until a transaction has completed (see MSTAT, Bit 8).	0x0	R/W
		0	Initiator is disabled.		
		1	Initiator is enabled.		

INITIATOR STATUS REGISTER

Address: 0x40003004, Reset: 0x6000, Name: MSTAT

Table 284. Bit Descriptions for MSTAT

Bits	Bit Name	Settings	Description	Reset	Access
15	Reserved		Reserved.	0x0	R
14	SCLFILT		State of I2C_SCL Line. This bit is the output of the glitch filter on I2C_SCL. I2C_SCL is always pulled high when undriven.	0x1	R
13	SDAFILT		State of I2C_SDA Line. This bit is the output of the glitch filter on I2C_SDA. I2C_SDA is always pulled high when undriven.	0x1	R

REGISTER DETAILS: I²C

Table 284. Bit Descriptions for MSTAT (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
12	MTXUNDR		Initiator Transmit Underflow. Asserts when the I²C initiator ends the transaction due to the transmit FIFO being empty. This bit is asserted only when MCTL, Bit 5 is set.	0x0	RC
11	MSTOP		Stop Driven by this I²C Initiator. Asserts when this I²C initiator drives a stop condition on the I²C bus. This bit, when asserted, can indicate a transaction completion, transmit underflow, receive overflow, or a no acknowledge by the target. This bit is different from the TCOMP bit because this bit is not asserted when the stop condition occurs due to any other I²C initiator. No interrupt is generated for the assertion of this bit. However, if MCTL, Bit 8 is 1, every stop condition generates an interrupt and this bit can be read. When this bit is read, it clears status.	0x0	RC
10	LINEBUSY		Line is Busy. Asserts when a start is detected on the I²C bus. Deasserts when a stop is detected on the I²C bus.	0x0	R
9	MRXOVR		Initiator Receive FIFO Overflow. Asserts when a byte is written to the receive FIFO when the FIFO is already full. When the bit is read, it clears the status.	0x0	RC
8	TCOMP		Transaction Complete or Stop Detected. Transaction complete. This bit asserts when a stop condition is detected on the I²C bus. If MCTL, Bit 8 is 1, an interrupt is generated when this bit asserts. This bit only asserts if the initiator is enabled (MCTL, Bit 0 = 1). Use this bit to determine when it is safe to disable the initiator. This bit can also be used to wait for another initiator transaction to complete on the I²C bus when this initiator loses arbitration. When this bit is read, it clears status. This bit can drive an interrupt.	0x0	RC
7	NACKDATA		Acknowledge Not Received in Response to Data Write. This bit asserts when an acknowledge is not received in response to a data write transfer. If MCTL, Bit 7 is 1, an interrupt is generated when this bit asserts. This bit can drive an interrupt. This bit is cleared on a read of the MSTAT register.	0x0	RC
6	MBUSY		Initiator Busy. This bit indicates that the initiator state machine is servicing a transaction. It is cleared if the state machine is idle or another device has control of the I²C bus.	0x0	R
5	ALOST		Arbitration Lost. This bit asserts if the initiator loses arbitration. If MCLT, Bit 6 is 1, an interrupt is generated when this bit asserts. This bit is cleared on a read of the MSTAT register. This bit can drive an interrupt.	0x0	RC
4	NACKADDR		Acknowledge Not Received in Response to an Address. This bit asserts if an acknowledge is not received in response to an address. If MCTL, Bit 7 is 1, an interrupt is generated when this bit asserts. This bit is cleared on a read of the MSTAT register. This bit can drive an interrupt.	0x0	RC
3	MRXREQ		Initiator Receive Request. This bit asserts when there is data in the receive FIFO. If MCTL, Bit 4 is 1, an interrupt is generated when this bit asserts. This bit can drive an interrupt.	0x0	R
2	MTXREQ		Initiator Transmit Request. This bit asserts when the direction bit is 0 and the transmit FIFO is either empty or not full. If MCTL, Bit 5 is 1, an interrupt is generated when this bit asserts. This bit can drive an interrupt.	0x0	R
[1:0]	MTXF		Initiator Transmit FIFO Status. Shows the initiator transmit FIFO status.	0x0	R
		00	FIFO empty.		
		10	1 byte in FIFO.		
		11	FIFO full.		

INITIATOR RECEIVE DATA REGISTER

Address: 0x40003008, Reset: 0x0000, Name: MRX

Table 285. Bit Descriptions for MRX

Bits	Bit Name	Settings	Description	Reset	Access
[15:8]	Reserved		Reserved.	0x0	R
[7:0]	VALUE		Initiator Receive. Allows access to the receive data FIFO. The FIFO can hold two bytes.	0x0	R

INITIATOR TRANSMIT DATA REGISTER

Address: 0x4000300C, Reset: 0x0000, Name: MTX

Table 286. Bit Descriptions for MTX

Bits	Bit Name	Settings	Description	Reset	Access
[15:8]	Reserved		Reserved.	0x0	R

REGISTER DETAILS: I²C

Table 286. Bit Descriptions for MTX (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
[7:0]	VALUE		Initiator Transmit. For test and debug purposes, when read, this register returns the byte that is currently being transmitted by the initiator. A byte written to the transmit register can be read back later when that byte is being transmitted on the line. This register allows access to the transmit data FIFO. The FIFO can hold two bytes.	0x0	R/W

INITIATOR RECEIVE DATA COUNT REGISTER

Address: 0x40003010, Reset: 0x0000, Name: MRXCNT

Table 287. Bit Descriptions for MRXCNT

Bits	Bit Name	Settings	Description	Reset	Access
[15:9]	Reserved		Reserved.	0x0	R
8	EXTEND		Extended Read. Use this bit if more than 256 bytes are required on a read. For example, to receive 412 bytes, write 0x100 (this bit = 1). Wait for the first byte to be received, then check the register for every byte received after. When the count bits in this register return to 0, 256 bytes have been received. Then write 0x09C to this register.	0x0	R/W
[7:0]	COUNT		Receive Count. Program the number of bytes required minus 1 to these bits. If 1 byte is required, write 0. If more than 256 bytes are required, use the extend bit.	0x0	R/W

INITIATOR CURRENT RECEIVE DATA COUNT REGISTER

Address: 0x40003014, Reset: 0x0000, Name: MCRXCNT

Table 288. Bit Descriptions for MCRXCNT

Bits	Bit Name	Settings	Description	Reset	Access
[15:8]	Reserved		Reserved.	0x0	R
[7:0]	COUNT		Current Receive Count. This register gives the total number of bytes received. If 256 bytes are requested, this register reads 0 when the transaction has completed.	0x0	R

FIRST INITIATOR ADDRESS BYTE REGISTER

Address: 0x40003018, Reset: 0x0000, Name: ADR1

Table 289. Bit Descriptions for ADR1

Bits	Bit Name	Settings	Description	Reset	Access
[15:8]	Reserved		Reserved.	0x0	R
[7:0]	ADR1		Address Byte 0. If a 7-bit address is required, Bit 7 to Bit 1 of ADR1 are programmed with the address, and Bit 0 of ADR1 is programmed with the direction (0 = write, 1 = read). If a 10-bit address is required, Bit 7 to Bit 3 of ADR1 are programmed with 11110, Bit 2 to Bit 1 of ADR1 are programmed with the two MSBs of the address, and Bit 0 of ADR1 is programmed to 0.	0x0	R/W

SECOND INITIATOR ADDRESS BYTE REGISTER

Address: 0x4000301C, Reset: 0x0000, Name: ADR2

Table 290. Bit Descriptions for ADR2

Bits	Bit Name	Settings	Description	Reset	Access
[15:8]	Reserved		Reserved.	0x0	R
[7:0]	ADR2		Address Byte 1. This register is only required when addressing a target with a 10-bit address. Bit 7 to Bit 0 of ADR2 are programmed with the lower eight bits of the address.	0x0	R/W

REGISTER DETAILS: I²C

SERIAL CLOCK PERIOD DIVISOR REGISTER

Address: 0x40003024, Reset: 0x1F1F, Name: DIV

Table 291. Bit Descriptions for DIV

Bits	Bit Name	Settings	Description	Reset	Access
[15:8]	HIGH		Serial Clock High Time. This register controls the clock high time. The PCLK drives the timer. To derive the required high time, calculate: High = (REQD_HIGH_TIME/PCLK_PERIOD) – 2. For example, to generate a 400 kHz I2C_SCL with a low time of 1300 ns and a high time of 1200 ns, with a core clock frequency of 26 MHz, Low = 1300 ns/38 ns − 1 = 0x21 (33 decimal). High = 1200 ns/38 ns − 2 = 0x1D (29 decimal). This register is reset to 0x1F, which gives an I2C_SCL high time of 33 PCLK cycles.	0x1F	R/W
[7:0]	LOW		Serial Clock Low Time. This register controls the clock low time. The PCLK drives the timer. To derive the required low time, calculate: Low = (REQD_LOW_TIME/PCLK_PERIOD) – 1. This register is reset to 0x1F, which gives an I2C_SCL low time of 32 PCLK cycles.	0x1F	R/W

TARGET CONTROL REGISTER

Address: 0x40003028, Reset: 0x0000, Name: SCTL

Table 292. Bit Descriptions for SCTL

Bits	Bit Name	Settings	Description	Reset	Access
15	Reserved		Reserved.	0x0	R
14	STXDMA		Enable Target Transmit DMA Request.	0x0	R/W
		0	Disable DMA mode.		
		1	Enable I²C target DMA receive requests.		
13	SRXDMA		Enable target Receive DMA Request.	0x0	R/W
		0	Disable DMA mode.		
		1	Enable I²C target DMA receive requests.		
12	IENREPST		Repeated Start Interrupt Enable.	0x0	R/W
		0	Interrupt not generated when the SSTAT, Bit 13 asserts.		
		1	Generate interrupt when the SSTAT, Bit 13 asserts.		
11	Reserved		Reserved.	0x0	R/W
10	IENSTX		Target Transmit Request Interrupt Enable.	0x0	R/W
9	IENSRX		Target Receive Request Interrupt Enable.	0x0	R/W
8	IENSTOP		Stop Condition Detected Interrupt Enable.	0x0	R/W
7	NACK		No Acknowledge Next Communication. If this bit is set, the next communication is not acknowledged.	0x0	R/W
6	Reserved		Reserved. Write 0 to this bit.	0x0	R/W
5	EARLYTXR		Early Transmit Request Mode. Setting this bit enables a transmit request immediately after the positive edge of the direction bit I2C_SCL clock pulse.	0x0	R/W
4	GCSBCLR		General Call Status Bit Clear.	0x0	W
		0	Does not clear general call status and general call ID bits.		
		1	Clear general call status and general call ID bits. The general call status and general call ID bits are not reset by anything other than a write to this bit or a full reset.		
3	HGCEN		Hardware General Call Enable. When this bit and the general call enable bit are set, after receiving a general call, the device and a data byte check the contents of the ALT register against the receive shift register. If these registers match, the device has received a hardware general call. This call is used if a device requires urgent attention from an initiator device without knowing which initiator to which to turn. This is a to whom it may concern call. The device that requires attention embeds its own address into the message. The LSB of the ALT register must always be written to a 1, as per the I²C January 2000 specification.	0x0	R/W
2	GCEN		General Call Enable. This bit enables the I²C target to acknowledge an I²C general call, Address 0x00 (write).	0x0	R/W
1	ADR10EN		Enabled 10-Bit Addressing. If this bit is clear, the target can support four target addresses, programmed in the ID0 register to the ID3 register. When this bit is set, 10-bit addressing is enabled. One 10-bit address is supported by the target and is stored in ID0 and ID1, where ID0 contains the first byte of the address and	0x0	R/W

REGISTER DETAILS: I²C

Table 292. Bit Descriptions for SCTL (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
			the upper 5 bits must be programmed to 11110. ID3 and ID4 can be programmed with 7-bit addresses at the same time.		
0	SLVEN		Target Enable.	0x0	R/W
		0	Target disabled and target state machine flops are held in reset.		
		1	Target enabled.		

TARGET I²C STATUS, ERROR, AND IRQ REGISTER

Address: 0x4000302C, Reset: 0x0001, Name: SSTAT

Table 293. Bit Descriptions for SSTAT

Bits	Bit Name	Settings	Description	Reset	Access
15	Reserved		Reserved.	0x0	R
14	START		Start and Matching Address. This bit is asserted if a start is detected on the I2C_SCL and I2C_SDA and the device address matches, if a general call code (address = 00000000) is received and general call is enabled, if a high-speed code (address = 00001XXX) is received, or if a start byte (00000001) is received. It is cleared on receipt of either a stop or start condition.	0x0	R
13	REPSTART		Repeated Start and Matching Address. This bit is asserted if a start is already asserted and then a repeated start is detected. It is cleared when read or on receipt of a stop condition. This bit can drive an interrupt.	0x0	RC
[12:11]	IDMAT		Device ID Matched.	0x0	R
		00	Received address matched ID Register 0.		
		01	Received address matched ID Register 1.		
		10	Received address matched ID Register 2.		
		11	Received address matched ID Register 3.		
10	STOP		Stop After Start and Matching Address. This bit is set by hardware if the target device receives a stop condition after a previous start condition and a matching address. Cleared by a read of the status register. If SCTL, Bit 8 in the target control register is asserted, the target interrupt request asserts when this bit is set. This bit can drive an interrupt.	0x0	RC
[9:8]	GCID		General ID. This bit is cleared when SCTL, Bit 4 is set to 1. These status bits are not cleared by a general call reset.	0x0	R
		00	No general call.		
		01	General call reset and program address.		
		10	General call program address.		
		11	General call matching alternative ID.		
7	GCINT		General Call Interrupt. This bit always drives an interrupt. The bit is asserted if the target device receives a general call of any type. To clear, write 1 to the GCSBCLR in the target control register. If the call was a general call reset, all registers are at their default values. If the call was a hardware general call, the receive FIFO holds the second byte of the general call, which can be compared with the ALT register.	0x0	R
6	SBUSY		Target Busy. Set by hardware if the target device receives an I²C start condition. Cleared by hardware when the address does not match an ID register, the target device receives an I²C stop condition, or if a repeated start address does not match.	0x0	R
5	NOACK		Acknowledge Not Generated by the Target. When asserted, this bit indicates that the target responded to its device address with a no acknowledge. This bit is asserted if there was no data to transmit and the sequence was a target read or if the no acknowledge bit was set in the target control register and the device was addressed. This bit is cleared on a read of the SSTAT register.	0x0	RC
4	SRXOF		Target Receive FIFO Overflow. Asserts when a byte is written to the target receive FIFO and the FIFO is already full.	0x0	RC
3	SRXREQ		Target Receive Request. This bit asserts whenever the target receive FIFO is not empty. Read or flush the target receive FIFO to clear this bit. This bit asserts on the falling edge of the I2C_SCL clock pulse that clocks in the last data bit of a byte. This bit can drive an interrupt.	0x0	RC

REGISTER DETAILS: I²C

Table 293. Bit Descriptions for SSTAT (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
2	STXREQ		Target Transmit Request. If SCTL, Bit 5 = 0, this bit is set when the direction bit for a transfer is received high. As long as the transmit FIFO is not full, this bit remains asserted. Initially, this bit is asserted on the negative edge of the SCL pulse that clocks in the direction bit (if the device address matches). If SCTL, Bit 5 = 1, this bit is set when the direction bit for a transfer is received high. As long as the transmit FIFO is not full, this bit remains asserted. Initially, this bit is asserted after the positive edge of the I2C_SCL pulse that clocks in the direction bit (if the device address matches). This bit is cleared on a read of the SSTAT register.	0x0	RC
1	STXUR		Target Transmit FIFO Underflow. This bit is set if an initiator requests data from the device, and the transmit FIFO is empty for the rising edge of SCL.	0x0	RC
0	STXFSEREQ		Target Transmit FIFO Status or Early Request. If SCTL, Bit 5 = 0, this bit is asserted whenever the target transmit FIFO is empty. If SCTL, Bit 5 = 1, this bit is set when the direction bit for a transfer is received high. This bit asserts on the positive edge of the I2C_SCL clock pulse that clocks in the direction bit if the device address matches. This bit only asserts once for a transfer and is cleared when read if SCTL, Bit 5 is asserted.	0x1	R/W

TARGET RECEIVE REGISTER

Address: 0x40003030, Reset: 0x0000, Name: SRX

Table 294. Bit Descriptions for SRX

Bits	Bit Name	Settings	Description	Reset	Access
[15:8]	Reserved		Reserved.	0x0	R
[7:0]	SRX		Target Receive Register.	0x0	R

TARGET TRANSMIT REGISTER

Address: 0x40003034, Reset: 0x0000, Name: STX

Table 295. Bit Descriptions for STX

Bits	Bit Name	Settings	Description	Reset	Access
[15:8]	Reserved		Reserved.	0x0	R
[7:0]	ISTX		Target Transmit Register.	0x0	R/W

HARDWARE GENERAL CALL ID REGISTER

Address: 0x40003038, Reset: 0x0000, Name: ALT

Table 296. Bit Descriptions for ALT

Bits	Bit Name	Settings	Description	Reset	Access
[15:8]	Reserved		Reserved.	0x0	R
[7:0]	ALT		Target Alternative. This register is used in conjunction with SCTL, Bit 3 to match an initiator generating a hardware general call. This register is used when an initiator device cannot be programmed with the address of a target, and instead the target must recognize the address of the initiator.	0x0	R/W

FIRST TARGET ADDRESS DEVICE ID REGISTER

Address: 0x4000303C, Reset: 0x0000, Name: ID0

Table 297. Bit Descriptions for ID0

Bits	Bit Name	Settings	Description	Reset	Access
[15:8]	Reserved		Reserved.	0x0	R
[7:0]	ID0		Target Device ID 0. ID0, Bits[7:1] are programmed with the device ID. ID0, Bit 0 is don't care. See SCTL, Bit 1 to see how this register is programmed with a 10-bit address. Take care to avoid I²C reserved target addresses with values less than 0x10 and greater than 0xF6.	0x0	R/W

REGISTER DETAILS: I²C

SECOND TARGET ADDRESS DEVICE ID REGISTER

Address: 0x40003040, Reset: 0x0000, Name: ID1

Table 298. Bit Descriptions for ID1

Bits	Bit Name	Settings	Description	Reset	Access
[15:8]	Reserved		Reserved.	0x0	R
[7:0]	ID1		Target Device ID 1. ID1, Bits[7:1] are programmed with the device ID. ID1, Bit 0 is don't care. See SCTL, Bit 1 to see how this register is programmed with a 10-bit address. Take care to avoid I²C reserved target addresses with values less than 0x10 and greater than 0xF6.	0x0	R/W

THIRD TARGET ADDRESS DEVICE ID REGISTER

Address: 0x40003044, Reset: 0x0000, Name: ID2

Table 299. Bit Descriptions for ID2

Bits	Bit Name	Settings	Description	Reset	Access
[15:8]	Reserved		Reserved.	0x0	R
[7:0]	ID2		Target Device ID 2. ID2, Bits[7:1] are programmed with the device ID. ID2, Bit 0 is don't care. See SCTL, Bit 1 to see how this register is programmed with a 10-bit address. Take care to avoid I²C reserved target addresses with values less than 0x10 and greater than 0xF6.	0x0	R/W

FOURTH TARGET ADDRESS DEVICE ID REGISTER

Address: 0x40003048, Reset: 0x0000, Name: ID3

Table 300. Bit Descriptions for ID3

Bits	Bit Name	Settings	Description	Reset	Access
[15:8]	Reserved		Reserved.	0x0	R
[7:0]	ID3		Target Device ID 3. ID3, Bits[7:1] are programmed with the device ID. ID3, Bit 0 is don't care. See SCTL, Bit 1 to see how this register is programmed with a 10-bit address. Take care to avoid I²C reserved target addresses with values less than 0x10 and greater than 0xF6.	0x0	R/W

INITIATOR AND TARGET FIFO STATUS REGISTER

Address: 0x4000304C, Reset: 0x0000, Name: FSTAT

Table 301. Bit Descriptions for FSTAT

Bits	Bit Name	Settings	Description	Reset	Access
[15:10]	Reserved		Reserved.	0x0	R/W
9	MFLUSH		Flush the Initiator Transmit FIFO.	0x0	W
		0	No effect.		
		1	Flush the initiator transmit FIFO. The initiator transmit FIFO must be flushed if arbitration is lost or a target responds with a no acknowledge.		
8	SFLUSH		Flush the Target Transmit FIFO.	0x0	W
		0	No effect.		
		1	Flush the target transmit FIFO.		
[7:6]	MRXF		Initiator Receive FIFO Status. The status is a count of the number of bytes in a FIFO.	0x0	R
		00	FIFO empty.		
		01	1 byte in the FIFO.		
		10	2 bytes in the FIFO.		
		11	Reserved.		
[5:4]	MTXF		Initiator Transmit FIFO Status. The status is a count of the number of bytes in a FIFO.	0x0	R
		00	FIFO empty.		
		01	1 byte in the FIFO.		
		10	2 bytes in the FIFO.		
		11	Reserved.		

REGISTER DETAILS: I²C

Table 301. Bit Descriptions for FSTAT (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
[3:2]	SRXF		Target Receive FIFO Status. The status is a count of the number of bytes in a FIFO.	0x0	R
		00	FIFO empty.		
		01	1 byte in the FIFO.		
		10	2 bytes in the FIFO.		
		11	Reserved.		
[1:0]	STXF		Target Transmit FIFO Status. The status is a count of the number of bytes in a FIFO.	0x0	R
		00	FIFO empty.		
		01	1 byte in the FIFO.		
		10	2 bytes in the FIFO.		
		11	Reserved.		

INITIATOR AND TARGET SHARED CONTROL REGISTER

Address: 0x40003050, Reset: 0x0000, Name: SHCTL

Table 302. Bit Descriptions for SHCTL

Bits	Bit Name	Settings	Description	Reset	Access
[15:1]	Reserved		Reserved.	0x0000	R/W
0	RST		Reset LINEBUSY. Setting this bit resets the LINEBUSY status bit (Bit 10 in the MSTAT register).	0x0	W
		0	No effect.		
		1	Reset the I²C start and stop detection circuits.		

AUTOMATIC STRETCH CONTROL FOR INITIATOR AND TARGET MODE REGISTER

Address: 0x40003058, Reset: 0x0000, Name: ASTRETCH_SCL

Table 303. Bit Descriptions for ASTRETCH_SCL

Bits	Bit Name	Settings	Description	Reset	Access
[15:10]	Reserved		Reserved.	0x0	R
9	SLVTMO		Stretch Timeout Status Bit for Target.	0x0	R
		0	Cleared when this bit is read.		
		1	Set when target automatic stretch mode has timed out.		
8	MSTTMO		Stretch Timeout Status Bit for Initiator.	0x0	R
		0	Cleared when this bit is read.		
		1	Set when initiator automatic stretch mode has timed out.		
[7:4]	SLV		Automatic Stretch Mode Control for Target. These bits control automatic stretch mode for target operation. These bits allow the target to hold the I2C_SCL line low and gain more time to service an interrupt, load a FIFO, or read a FIFO. Use the timeout feature to avoid a bus lockup condition where the target indefinitely holds I2C_SCL low. As a target transmitter, I2C_SCL is automatically stretched from the negative edge of I2C_SCL (if the target transmit FIFO is empty) before sending an acknowledge or a no acknowledge for an address byte, or before sending data for a data byte. Stretching stops when the target transmit FIFO is no longer empty or a timeout occurs. As a target receiver, the I2C_SCL clock is automatically stretched from the negative edge of I2C_SCL before sending an acknowledge or a no acknowledge when the target receive FIFO is full. Stretching stops when the target receive FIFO is no longer in an overflow condition or a timeout occurs.	0x0	R/W
		0000	Automatic target clock stretching disabled.		
		0001 to 1110	Automatic target clock stretching enabled. The timeout period is defined as follows: $\frac{DIV[15:8]+(DIV[7:4]-1)}{UCLK/CTL1[13:8]-CTL1[13:8]} \times \left(2^{ASTRETCH\_SCL[7:4]}\right)$ Note that the I²C bus baud rate has no influence on the target stretch timeout period.		
		1111	Automatic target clock stretching enabled with indefinite timeout period.		
[3:0]	MST		Automatic Stretch Mode Control for Initiator. These bits control automatic stretch mode for initiator operation. These bits allow the initiator to hold the I2C_SCL line low and gain more time to service	0x0	R/W

REGISTER DETAILS: I²C

Table 303. Bit Descriptions for ASTRETCH_SCL (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
			an interrupt, load a FIFO, or read a FIFO. Use the timeout feature to avoid a bus lockup condition where the initiator indefinitely holds I2C_SCL low. As an initiator transmitter, I2C_SCL is automatically stretched from the negative edge of I2C_SCL (if the initiator transmit FIFO is empty) before sending an acknowledge or a no acknowledge for an address byte, or before sending data for a data byte. Stretching stops when the initiator transmit FIFO is no longer empty or a timeout occurs. As an initiator receiver, the I2C_SCL clock is automatically stretched from the negative edge of I2C_SCL before sending an acknowledge or a no acknowledge when the initiator receive FIFO is full. Stretching stops when the initiator receive FIFO is no longer in an overflow condition or a timeout occurs.		
		0000	Automatic initiator clock stretching disabled.		
		0001 to 1110	Automatic initiator clock stretching enabled. The timeout period is defined as follows: $\frac{DIV[15:8]+(DIV[7:4]-1)}{UCLK/CTL1[13:8]-CTL1[13:8]} \times \left(2^{ASTRETCH\_SCL[3:0]}\right)$		
		1111	Automatic initiator clock stretching enabled with indefinite timeout period.		

SERIAL PERIPHERAL INTERFACES

SPI FEATURES

The ADuCM355 integrates two complete hardware SPIs with the following standard features:

- Serial clock phase mode and serial clock polarity mode.
- LSB first transfer option.
- Loopback mode.
- Initiator or target mode.
- Flow control, the SPI for Channel 1 (SPI1) channel only.
- Support for 3-pin SPI initiator or target, single bidirectional data pin.
- Transfer and interrupt mode.
- Continuous transfer mode.
- Transmit and receive FIFO.
- Interrupt mode. Interrupt after one byte to eight bytes.
- Receive overflow mode and transmit under run mode.
- Open circuit data output mode.
- Full duplex communications supported (simultaneous transmit and receive).

SPI OVERVIEW

The ADuCM355 integrates two complete hardware SPIs. SPI is an industry-standard, synchronous serial interface that allows eight bits of data to be synchronously transmitted and simultaneously received (also known as full duplex). The two SPIs implemented on the ADuCM355 can operate to a maximum bit rate of 6.5 Mbps in both initiator and target modes.

Optional modes of operation include the following:

- Flow control. Supported by SPI1, which has an optional extra ready pin (P0.3/SPI0_$\overline{CS}$). Flow control helps slow target devices to interface with fast initiators. Another option available is to insert wait states during ready data, which is helpful in initiator mode when the user is looking to read bursts of data from a target and leave a timing gap between each burst. The gap or wait state is timer controlled.
- Fast mode.
- 3-pin mode. The SPI0_MOSI and SPI1_MOSI pins in this mode are bidirectional pins.

The SPI blocks have an additional DMA feature. Each SPI block has two DMA channels that interface with a microDMA controller of the Arm Cortex-M3 processor. One DMA channel is used for transmitting data, and the other is used for receiving data.

SPI OPERATION

In SPI operation, $\overline{CS}$ refers to the SPI0_$\overline{CS}$ pin and the SPI1_$\overline{CS}$ pin, SCLK refers to the SPI0_CLK pin and the SPI1_CLK pin, MOSI refers to the SPI0_MOSI pin and the SPI1_MOSI pin, and MISO refers to the SPI0_MISO pin and the SPI1_MISO pin.

The SPI port can be configured for initiator or target operation, and consists of four sets of pins: MISO, MOSI, SCLK, and $\overline{CS}$. The GPIOs used for SPI communication must be configured in SPI mode before enabling the SPI peripheral. Enable the internal pull-up resistors on the MISO and MOSI pins when communicating over SPI.

MISO Pin

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

MOSI Pin

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

SERIAL PERIPHERAL INTERFACES

SCLK Pin

The initiator SCLK synchronizes the data being transmitted and received through the MOSI SCLK period. Therefore, a byte is transmitted or received after eight SCLK periods. SCLK is configured as an output in initiator mode and as an input in target mode.

In initiator mode, the SPIx_CTL register controls the polarity and phase of the clock, and the bit rate is defined in the SPIx_DIV register as follows:

$$f_{SCLK} = \frac{PCLK}{2 \times (1 + SPIx\_DIV[5:0])} \quad (27)$$

where *PCLK* is the system clock divided by the factor set in CTL1, Bits[13:8].

By reducing the clock rate to the SPI blocks, it is possible to reduce the power consumption of the SPI block. The maximum data rate is 13 Mbps.

In target mode, the SPIx_CTL register must be configured with the phase and polarity of the expected input clock. The target accepts data from an external initiator up to 20 Mbps. In both initiator and target 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 initiator and target devices.

P0.3/SPI0_$\overline{CS}$ Pin and P1.5/SPI1_$\overline{CS}$ Pin

In SPI target mode, a transfer is initiated by the assertion of the chip select, 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 target mode, $\overline{CS}$ is always an input.

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

If an ADuCM355 initiator wants to communicate with multiple SPI targets, GPIOs can be connected to the chip select lines of the targets. Use the CSRISE and CSFALL bits (SPIx_STAT, Bit 13 and SPIx_STAT, Bit 14, respectively) to determine when to pull the GPIOs low or high.

SPI TRANSFER INITIATION

In initiator mode, the transfer and interrupt mode bit (SPIx_CTL, Bit 6) determines the manner in which an SPI serial transfer is initiated. If this bit is set, a serial transfer is initiated after a write to the transmit FIFO. If this bit is cleared, a serial transfer is initiated after a read of the receive FIFO. The read must be performed while the SPI interface is idle. A read performed during an active transfer does not initiate another transfer.

For any setting of SPIx_CTL, Bit 1 and SPIx_CTL, Bit 6, the SPI simultaneously receives and transmits data. Therefore, during data transmission, the SPI is also receiving data and filling up the receive FIFO. If the data is not read from the receive FIFO, the overflow interrupt occurs when the FIFO starts to overflow. If the user does not want to read the receive data or receive overflow interrupts, set SPIx_CTL, Bit 12 and the receive data is not saved to the receive FIFO. Similarly, to only receive data and not write data to the transmit FIFO, set SPIx_CTL, Bit 13 to avoid receiving underrun interrupts from the transmit FIFO.

Transmit Initiated Transfer

For transfers initiated by a write to the transmit FIFO, the SPI starts transmitting as soon as the first byte is written to the FIFO, irrespective of the configuration in SPIx_IEN, Bits[2:0]. The first byte is immediately read from the FIFO, written to the transmit shift register, and the transfer commences.

If the continuous transfer enable bit, SPIx_CTL, Bit 11, is set, the transfer continues until no valid data is available in the transmit FIFO. This completion is either the end of SPIx_CNT, Bits[13:0] number of bytes (if SPIx_CNT, Bits[13:0] > 0) or when no valid data is available in the transmit FIFO (if SPIx_CNT, Bits[13:0] = 0). Chip select remains asserted for the duration of the complete transfer. If SPIx_CNT, Bit 15 is cleared and SPIx_CNT, Bits[13:0] > 0, the transfer stops when all the bytes in SPIx_CNT, Bits[13:0] have been transferred. If SPIx_CNT, Bit 15 is set, a new frame starts after every SPIx_CNT, Bits[13:0] number of bytes. Multiples of bytes in SPIx_CNT, Bits[13:0] are always transferred in this case. If there is no data or space in the FIFO, the transfer stalls until it is available. Conversely, the transfer continues while there is valid data in the FIFO. If SPIx_CTL, Bit 11 is cleared, each transfer consists of a single 8-bit serial transfer. If valid data exists in the transmit FIFO, a new transfer is initiated after a stall period, and a chip select is deasserted.

SERIAL PERIPHERAL INTERFACES

Receive Initiated Transfer

Transfers initiated by a read of the receive FIFO depend on the number of bytes received in the FIFO. If SPIx_IEN, Bits[2:0] = 0b111 and a read to the receive FIFO occurs, the SPI initiates an 8-byte transfer. If continuous mode is set (SPIx_CTL, Bit 11 = 1), the eight bytes occur continuously with no deassertion of chip select between bytes. However, in continuous mode, if SPIx_CNT, Bits[13:0] > 0, the chip select asserts for the entire frame duration. The SPI introduces stall periods by not clocking SPI0_CLK or SPI1_CLK until FIFO space is available. If continuous mode is not set, the eight bytes occur with stall periods between transfers where the chip select output is deasserted.

If SPIx_IEN, Bits[2:0] = 0b110, a read of the receive FIFO initiates a 7-byte transfer. If SPIx_IEN, Bits[2:0] = 0b001, a read of the receive FIFO initiates a 2-byte transfer. If SPIx_IEN, Bits[2:0] = 0b000, a read of the receive FIFO initiates a 1-byte transfer. A read of the receive FIFO while the SPI is receiving data does not initiate another transfer after the present transfer is complete.

In continuous mode, if SPIx_CNT, Bits[13:0] > 0 and SPIx_CNT, Bit 15 = 1, a read of the receive FIFO at the end of an SPI frame always initiates a new SPI frame. To stop SPI transfers at any given frame, clear the SPIx_CNT, Bit 15 before reading the final set of receive bytes.

The SPI transfer protocol diagrams illustrate the data transfer protocol for the SPI and the effects of the CPHA and CPOL bits in the control register (SPIx_CTL) on that protocol. See Figure 61 and Figure 62.

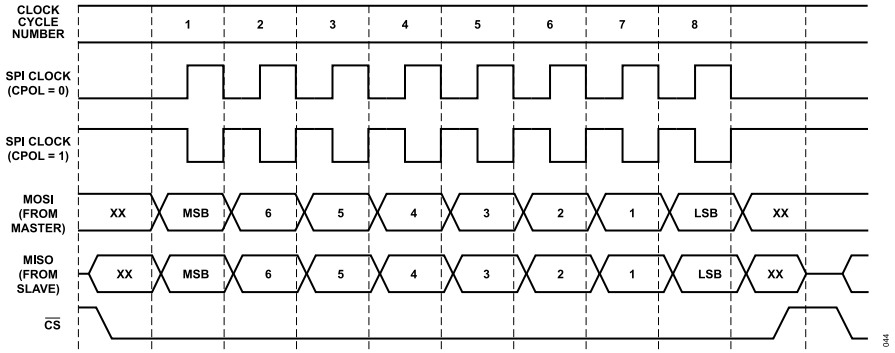

Figure 61. SPI Transfer Protocol, CPHA = 0

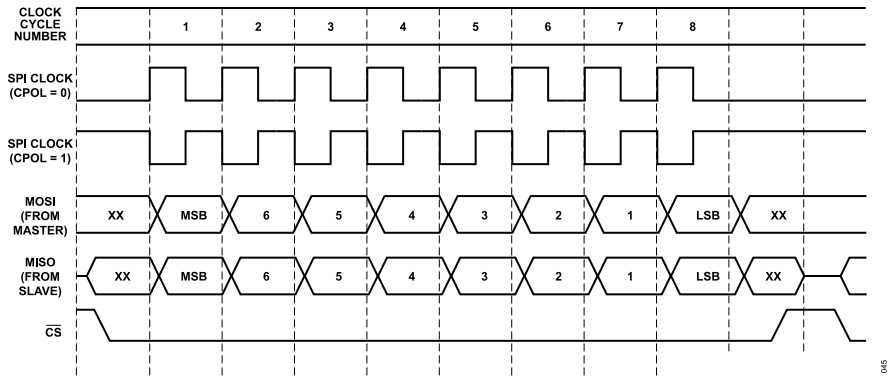

Figure 62. SPI Transfer Protocol, CPHA = 1

Transfers in Target Mode

In target mode, a transfer is initiated by the assertion of the chip select of the device. Though the initiator can support up to four chip select output lines, only one chip select input is used in target mode. The device as a target transmits and receives 8-bit data until the transfer is concluded by the deassertion of chip select. The SPI transfer protocol diagrams in Figure 61 and Figure 62 illustrate the data transfer protocol for the SPI, and the effects of SPIx_CTL, Bit 2 and SPIx_CTL, Bit 3 on that protocol. The chip select must not be tied to the ground.

SPI Data Underrun and Overflow

If the transmit zeros enable bit (SPIx_CTL, Bit 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 SPIx_CTL, Bit 7 is set to 1, 0s are transmitted when a transfer is initiated with no valid data in the

SERIAL PERIPHERAL INTERFACES

FIFO. If the receive overflow overwrite enable bit (SPIx_CTL, Bit 8) is set, and there is no space left in the FIFO, the valid data in the receive FIFO is overwritten by the new serial byte received. If SPIx_CTL, Bit 8 is cleared, and there is no space left in the FIFO, the new serial byte received is discarded. When SPIx_CTL, Bit 8 is set, the contents of the SPI receive FIFO are undefined, and its contents must be discarded by user code.

Full Duplex Operation

Simultaneous reads and writes are supported on the SPI. When implementing full duplex transfers in initiator mode, use the following procedure:

1. Initiate a transfer sequence via a transmit on the P0.1/SPI0_MOSI pin and the P1.3/SPI1_MOSI pin. Set SPIx_CTL, Bit 6 = 1. If interrupts are enabled, interrupts are triggered when a transmit interrupt occurs but not when a byte is received.
2. If using interrupts, the SPI transmit interrupt indicated by SPIx_STAT, Bit 5 or the transmit FIFO underrun interrupt (SPIx_STAT, Bit 4) is asserted approximately three to four SPI clock periods into the transfer of the first byte. If necessary, reload a byte into the transmit FIFO by writing to the SPIx_TX register.
3. The first byte received via the MISO pin does not update the receive FIFO status bits (SPIx_FIFO_STAT, Bits[11:8]) until 12 SPI clock periods after $\overline{CS}$ goes low. Therefore, two transmit interrupts can occur before the first receive byte is ready to be processed.
4. After the last transmit interrupt occurs, it may be necessary to read two more bytes. It is recommended that SPIx_FIFO_STAT, Bits[11:8] be polled outside of the SPI interrupt handler after the last transmit interrupt is handled.

SPI INTERRUPTS

There is one interrupt line per SPI and 11 sources of interrupts. SPIx_STAT, Bit 0 reflects the state of the interrupt line, and SPIx_STAT, Bits[15:12] and SPIx_STAT, Bits[7:1] reflect the state of the 11 sources. The SPI generates either transmit or receive interrupts. Both interrupts cannot be enabled at the same time. The appropriate interrupt is enabled using SPIx_CTL, Bit 6. If TIM = 1, the transmit IRQ is enabled. If TIM = 0, the receive IRQ is enabled. In addition, the SPI0 and SPI1 interrupt source must be enabled in the NVIC register as follows: ISER0, Bit 15 = SPI0, ISER0, Bit 16 = SPI1.

Transmit Interrupt

If SPIx_CTL, Bit 6 is set, the transmit FIFO status causes the interrupt. The SPIx_IEN, Bits[2:0] control when the interrupt occurs, as shown in Table 304.

Table 304. SPIx_IEN, Bits[2:0] IRQ Mode Bits

SPIx_IEN, Bits[2:0] Settings	Interrupt Condition
000	An interrupt occurs after each byte that is transmitted. The interrupt occurs when the byte is read from the FIFO and written to the shift register.
001	An interrupt occurs after every two bytes that are transmitted.
010	An interrupt occurs after every third byte that is transmitted.
011	An interrupt occurs after every fourth byte that is transmitted.
100	An interrupt occurs after every fifth byte that is transmitted.
101	An interrupt occurs after every sixth byte that is transmitted.
110	An interrupt occurs after every seventh byte that is transmitted.
111	An interrupt occurs after every eighth 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. The transmit interrupt is different from the receive interrupt, which depends on the number of bytes in the receive 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 read by reading SPIx_STAT, Bit 5. The interrupt is disabled if SPIx_CTL, Bit 13 is left high. A write to the control register, SPIx_CTL, resets the transmitted byte counter back to 0. For example, if SPIx_IEN, Bits[2:0] are set to 0x3 and SPIx_CTL is written to after three bytes are transmitted, the transmit interrupt does not occur until another four bytes are transmitted.

SERIAL PERIPHERAL INTERFACES

Receive Interrupt

If the TIM bit (SPIx_CTL, Bit 6) is cleared, the receive FIFO status causes the receive interrupt to be generated. SPIx_IEN, Bits[2:0] control when the interrupt occurs. The interrupt is cleared by a read of the SPIx_STAT register. The status of this interrupt can be read in SPIx_STAT, Bit 6.

Interrupts are only generated when data is written to the FIFO. For example, if SPIx_IEN, Bits[2:0] is set to 0b000, 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 is not regenerated. Another interrupt is not generated until another byte is received in the FIFO.

The interrupt depends on the number of valid bytes in FIFO and not on the number of bytes received. For example, when SPIx_IEN, Bits[2:0] is set to 0b001, an interrupt is generated after a byte is received if there are two or more bytes in the FIFO. The interrupt is not generated after every two bytes received. The receive interrupt is disabled if SPIx_CTL, Bit 12 is left high.

Underrun and Overflow Interrupts

SPIx_STAT, Bit 7 and SPIx_STAT, Bit 4 generate SPI interrupts. When a transfer starts with no data in the transmit FIFO, SPIx_STAT, Bit 4 is set to indicate an underrun condition, which causes an interrupt. The interrupt and status bits are cleared upon a read of the status register. This interrupt occurs irrespective of SPIx_IEN, Bits[2:0]. This interrupt is disabled if SPIx_CTL, Bit 13 is set.

When data is received and the receive FIFO is already full, SPIx_STAT, Bit 7 is set to 1, indicating an overflow condition, which causes an interrupt. The interrupt and status bit are cleared upon a read of the status register. This interrupt occurs irrespective of SPIx_CTL, Bits[2:0]. This interrupt is disabled if SPIx_CTL, Bit 12 is set.

When the SPI receive overflow bit (SPIx_STAT, Bit 7) is set to 1, the contents of the SPI receive FIFO are undetermined and must not be used. The user must flush the receive FIFO upon detecting this error condition. All interrupts are cleared either by a read of the status register or when SPIx_CTL, Bit 0 is cleared to 0. The receive and transmit interrupts are also cleared if the relevant flush bits are asserted. Otherwise, the interrupts remain active even if the SPI is reconfigured.

SPI WIRE-OR'ED MODE

To prevent contention when the SPI is used in a multiinitiator or multitarget system, the data output pins, MOSI and MISO, can be configured to behave as open circuit drivers. An external pull-up resistor is required when this feature is selected. The wire-OR bit (SPIx_CTL, Bit 4) controls the pad enable outputs for the data lines.

SPI CSERR CONDITION

The CSERR bit (SPIx_STAT, Bit 12) indicates if an erroneous deassertion of the $\overline{CS}$ signal has been detected before the completion of all eight SCLK cycles. This bit generates an interrupt and is available in all modes of operation: target, initiator, and during DMA transfers. If an interrupt generated by SPIx_STAT, Bit 12 occurs, SPIx_CTL, Bit 0 must be disabled and restarted to enable a clean recovery to ensure that subsequent transfers are error free.

SPI DMA

Two DMA channels are dedicated to transmit and receive. The SPI DMA channels must be configured in the microDMA controller of the Arm Cortex-M3 processor. It is possible to enable a DMA request on one or two channels at the same time by setting the DMA request bits for receive or transmit in the SPIx_DMA register. If only the DMA transmit request (SPIx_DMA, Bit 1) is enabled, the receive FIFO overflows during an SPI transfer, unless the received data is read by user code, in which case an overflow interrupt is generated. To avoid generating overflow interrupts, set the receive FIFO flush bit, or disable the SPI interrupt in the NVIC. If only the DMA receive request (SPIx_DMA, Bit 2) is enabled, the transmit FIFO is underrun. To avoid an underrun interrupt, the SPI interrupt must be disabled.

The SPI transmit (SPIx_STAT, Bit 5) and SPI receive (SPIx_STAT, Bit 6) interrupts are not generated when using DMA. The SPI transmit underflow (SPIx_STAT, Bit 4) and receive overflow (SPIx_STAT, Bit 7) interrupts are generated when using the microDMA controller. SPIx_IEN, Bits[2:0] are not used in transmit mode and must be set to 0b000 in receive mode.

SPIx_DMA, Bit 0 controls the start of a DMA transfer. DMA requests are only generated when SPIx_DMA, Bit 0 = 1. At the end of a DMA transfer, this bit must be cleared to prevent extra DMA requests to the microDMA controller. The data still present in the transmit FIFO is transmitted if in transmit mode. All DMA data transfers are 16-bit transfers. Program the DMA accordingly. For example, if 16 bytes of data are to be transferred over the SPI, program the DMA to perform eight half word (16-bit) transfers. If 17 bytes are to be transferred, nine half word transfers are required.

SERIAL PERIPHERAL INTERFACES

In DMA mode, the transmit and receive FIFOs are two bytes wide. Bits[7:0] are first accessed by the SPI, followed by Bits[15:8]. This is irrespective of count or SPIx_CTL, Bit 5 settings.

For example, if SPI_CNT, Bits[13:0] = 3, the order of transmission and reception is Byte 1, Byte 0 followed by Byte 3 (ignore), and Byte 2. SPI_CTL, Bit 5 does not affect the FIFO access order in DMA mode. SPI_CTL, Bit 5 only affects how each byte is transferred over SPI.

DMA Initiator Transmit Configuration

The DMA SPI transmit channel must be configured. Configure the NVIC to enable DMA transmit initiator interrupts.

All DMA transfers are 16-bit transfers. When all data present in the DMA buffer is transmitted, the DMA generates an interrupt. User code must disable the DMA request. Data is still in the transmit FIFO because the DMA request is generated each time there is free space in the transmit FIFO to keep the FIFO full. User code can check how many bytes are still present in the FIFO in the SPIx_FIFO_STAT register. The SPI block must be configured as follows:

```
SPI_DIV = SPI_SERIAL_FREQ;  //configures serial clock frequency.
SPI_CTL = 0x1043;          //enables SPI in initiator mode and transmit mode, receive FIFO
//flush enabled.
SPI_CNT.VALUE = NUM_BYTES_TO transfer; //sets the number of bytes to transfer. SPI_DMA = 0x1;
//(optional) enables FIFO to accept 16-bit
//core data writes.
SPI_TX = 0xXXXX;      //(optional) up to four 16-bit core writes can be performed
//to preload FIFO.
SPI_DMA = 0x3;      //enable DMA mode, enable transmit DMA request.
```

DMA Initiator Receive Configuration

The SPIx_CNT register is available in DMA receive initiator mode only. This register sets the number of receive bytes required by the SPI initiator or the number of clocks that the initiator must generate. When the required number of bytes are received, no more transfers are initiated. To initiate a DMA initiator receive transfer, complete a dummy read by user code. Add this dummy read to the SPIx_CNT number.

The counter counting the bytes as they are received is reset either when SPI is disabled in SPIx_CTL, Bit 0 or when the SPIx_CNT register is modified by user code.

Performing SPI DMA Initiator Receive

The DMA SPI receive channel must be configured. The NVIC must be configured to enable DMA receive initiator interrupts. The SPI block must be configured as follows:

```
SPI_DIV = SPI_SERIAL_FREQ; //configures serial clock frequency.
SPI_CTL = 0x2003;    //enable SPI in initiator mode and
//receive mode, 1 byte transfer.
SPI_DMA = 0x5;        //enable DMA mode, enable receive DMA request. SPI_CNT.VALUE = XXX;
//number of bytes to be received.
A = SPI_RX;    //dummy read.
```

The DMA transfer stops when the appropriate number of clock cycles have been generated. All DMA data transfers are 16-bit transfers. Program the DMA accordingly. For example, if 16 bytes of data are to be received over the SPI, program the DMA to perform eight 16-bit transfers. If 17 bytes are to be received, nine 16-bit transfers are required. The additional bytes are padded for the final DMA transfer. Data errors occur if the DMA transfers are programmed as byte wide transfers.

The DMA transfer stops when the appropriate number of bytes have been transferred. The DMA buffer must be of the same size as SPIx_CNT to generate a DMA interrupt when the transfer is complete. SPIx_CNT must always be ≥2.

SPI AND POWER-DOWN MODES

In initiator mode, before entering power-down mode, disable the SPI block in SPIx_CTL, Bit 0. In target mode, in either mode of operation (interrupt driven or DMA), the chip select line level must be checked via the GPIO registers using SPIx_STAT, Bit 11 to ensure that the SPI is

SERIAL PERIPHERAL INTERFACES

not communicating and that the SPI block is disabled while the chip select line is high. At power-up, the SPI block can be reenabled. While being powered down, the following fields are retained:

- All bit fields of SPIx_CTL register, except SPIx_CTL, Bit 0 is reset to 0 on power-up, which allows a clean start of the design at wakeup.
- SPIx_IEN, Bits[2:0] bit field.
- SPIx_DIV, Bits[5:0] bit field.
- SPIx_RD_CTL, Bit 8.
- SPIx_FLOW_CTL, Bit 4.

All other bit fields are not retained. As such, they are all reset on power-up. On exiting the power-down mode, the software reprograms all nonretained registers as required. Then, the SPI block must be reenabled by setting the SPIx_CTL, Bit 0.

REGISTER SUMMARY: SPI0/SPI1

Table 305. SPI Register Summary

Address	Name	Description	Reset	Access
0x40004000	SPI0_STAT	Status	0x0800	R
0x40004004	SPI0_RX	Receive	0x0000	R
0x40004008	SPI0_TX	Transmit	0x0000	W
0x4000400C	SPI0_DIV	Baud rate selection	0x0000	R/W
0x40004010	SPI0_CTL	Configuration	0x0000	R/W
0x40004014	SPI0_IEN	Interrupt configuration	0x0000	R/W
0x40004018	SPI0_CNT	Transfer byte count	0x0000	R/W
0x4000401C	SPI0_DMA	DMA enable	0x0000	R/W
0x40004020	SPI0_FIFO_STAT	FIFO status	0x0000	R
0x40004024	SPI0_RD_CTL	Read control	0x0000	R/W
0x40004028	SPI0_FLOW_CTL	Flow control	0x0000	R/W
0x4000402C	SPI0_WAIT_TMR	Wait timer for flow control	0x0000	R/W
0x40004034	SPI0_CS_OVERRIDE	Chip select override	0x0000	R/W
0x40024000	SPI1_STAT	Status	0x0800	R
0x40024004	SPI1_RX	Receive	0x0000	R
0x40024008	SPI1_TX	Transmit	0x0000	W
0x4002400C	SPI1_DIV	Baud rate selection	0x0000	R/W
0x40024010	SPI1_CTL	Configuration	0x0000	R/W
0x40024014	SPI1_IEN	Interrupt configuration	0x0000	R/W
0x40024018	SPI1_CNT	Transfer byte count	0x0000	R/W
0x4002401C	SPI1_DMA	DMA enable	0x0000	R/W
0x40024020	SPI1_FIFO_STAT	FIFO status	0x0000	R
0x40024024	SPI1_RD_CTL	Read control	0x0000	R/W
0x40024028	SPI1_FLOW_CTL	Flow control	0x0000	R/W
0x4002402C	SPI1_WAIT_TMR	Wait timer for flow control	0x0000	R/W
0x40024034	SPI1_CS_OVERRIDE	Chip select override	0x0000	R/W

REGISTER DETAILS: SPI0/SPI1

STATUS REGISTERS

Address: 0x40004000, Reset: 0x0800, Name: SPI0_STAT

Address: 0x40024000, Reset: 0x0800, Name: SPI1_STAT

Table 306. Bit Descriptions for SPI0_STAT, SPI1_STAT

Bits	Bit Name	Settings	Description	Reset	Access
15	RDY		Detected an Edge on Ready Indicator for Flow Control. This bit indicates that there was an active edge on the P0.3 line depending on the flow control mode. If SPIx_FLOW_CTL, Bits[1:0] = 0b10, this bit is set whenever an active edge is detected on the P0.3 signal. If SPIx_FLOW_CTL, Bits[1:0] = 0b11, this bit is set if an active edge is detected on the MISO signal.	0x0	R/W1C
		0	If SPIx_FLOW_CTL, Bits[1:0] = 0b00 or 0b01. The active edge (rising or falling) is determined by SPI_FLOW_CTL, Bit 4.		
		1	When SPIx_IEN, Bit 11 is set to 1, this bit causes an interrupt, which is only cleared by writing 1 to this bit.		
14	CSFALL		Detected a Falling Edge on Chip Select in Target Continuous Mode. This bit causes an interrupt when SPIx_IEN, Bit 8 is set to 1, which can be used to identify the start of an SPI data frame.	0x0	R/W1C
		0	Cleared to 0 when a 1 is written to this bit.		
		1	Set when there is a falling edge on chip select line. Used to identify the start of an SPI data frame.		
13	CSRISE		Detected a Rising Edge on Chip Select in Target Continuous Mode. This bit causes an interrupt when SPIx_IEN, Bit 8 is set to 1.	0x0	R/W1C
		0	Cleared to 0 when a 1 is written to this bit or when SPIx_CTL, Bit 0 is cleared to 0.		
		1	Set when there is a rising edge on the chip select line. This bit can be used to identify the end of an SPI data frame.		
12	CSERR		Detected a Chip Select Error Condition.	0x0	R/W1C
		0	Cleared to 0 when a 1 is written to this bit.		
		1	Set when the chip select line is deasserted abruptly, even before the full byte of data is transmitted completely. This bit causes an interrupt.		
11	CS		Chip Select Status. This bit reflects the actual chip select status as seen by the SPI module. This bit uses SCLK to PCLK synchronization. As such, there is a slight delay when chip select changes state.	0x1	R
		0	Chip select line is low.		
		1	Chip select line is high.		
[10:8]	Reserved		Reserved.	0x0	R
7	RXOVR		SPI Receive FIFO overflow.	0x0	R/W1C
		0	Cleared to 0 when a 1 is written to this bit or when SPIx_CTL, Bit 0 is cleared to 0.		
		1	Set when the receive FIFO is already full when new data is loaded to the FIFO. This bit generates an interrupt if SPIx_IEN, Bit 10 = 1, except when SPIx_CTL, Bit 12 is set.		
6	RXIRQ		SPI Receive IRQ. Not available in DMA mode.	0x0	R/W1C
		1	Cleared to 0 when a 1 is written to this bit or when SPIx_CTL, Bit 0 is cleared to 0.		
		0	Set when a receive interrupt occurs. This bit is set when TIM in SPIx_CTL, Bit 6 is cleared and the required number of bytes is received.		
5	TXIRQ		SPI Transmit IRQ. Status bit. This bit is not available in DMA mode.	0x0	R/W1C
		0	Cleared to 0 when a 1 is written to this bit or when SPIx_CTL, Bit 0 is cleared to 0.		
		1	Set when a transmit interrupt occurs. This bit is set when SPIx_CTL, Bit 6 is set and the required number of bytes is transmitted.		
4	TXUNDR		SPI Transmit FIFO Underflow.	0x0	R/W1C
		0	Cleared to 0 when a 1 is written to this bit or when SPIx_CTL, Bit 0 is cleared to 0.		
		1	Set when a transmit is initiated without any valid data in the transmit FIFO. This bit generates an interrupt when SPIx_IEN, Bit 9 = 1 except when SPIx_CTL, Bit 13 is set.		
3	TXDONE		SPI Transmit Done in Read Command Mode.	0x0	R/W1C
		0	Cleared to 0 when 1 is written to this bit or when SPIx_CTL, Bit 0 is cleared to 0.		
		1	Set when the entire transmit is completed in a read command. This bit generates an interrupt if SPIx_IEN, Bit 12 is set, and is only valid if SPIx_RD_CTL, Bit 0 is set.		
2	TXEMPTY		SPI Transmit FIFO Empty Interrupt.	0x0	R/W1C

REGISTER DETAILS: SPI0/SPI1

Table 306. Bit Descriptions for SPI0_STAT, SPI1_STAT (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		0	Cleared to 0 when 1 is written to this bit or when SPIx_CTL, Bit 0 is cleared to 0.		
		1	Set when the transmit FIFO is empty. This bit generates an interrupt if SPIx_IEN, Bit 14 is set to 1 except when SPIx_CTL, Bit 13 is set.		
1	XFRDONE		SPI Transfer Completion. This bit indicates the status of SPI transfer completion in initiator mode.	0x0	R/W1C
		0	Cleared to 0 when 1 is written to this bit.		
		1	Set when the transfer of SPIx_CNT, Bits[13:0] number of bytes has finished. In target mode or if SPIx_CNT, Bits[13:0] = 0, this bit is invalid. If SPIx_IENx, Bit 13 is set, this bit generates an interrupt. It uses the state of the initiator state machine to determine the completion of a SPI transfer. Therefore, a chip select override does not affect this bit.		
0	IRQ		SPI Interrupt Status.	0x0	R
		0	Cleared to 0 when all SPI interrupt sources are cleared.		
		1	Set when an SPI based interrupt occurs.		

RECEIVE REGISTERS

Address: 0x40004004, Reset: 0x0000, Name: SPI0_RX

Address: 0x40024004, Reset: 0x0000, Name: SPI1_RX

Table 307. Bit Descriptions for SPI0_RX, SPI1_RX

Bits	Bit Name	Settings	Description	Reset	Access
[15:8]	BYTE2		8-Bit Receive Buffer. These 8 bits are used only in DMA mode, where all FIFO accesses occur as half word accesses. They return 0 if DMA is disabled.	0x0	R
[7:0]	BYTE1		8-Bit Receive Buffer.	0x0	R

TRANSMIT REGISTERS

Address: 0x40004008, Reset: 0x0000, Name: SPI0_TX

Address: 0x40024008, Reset: 0x0000, Name: SPI1_TX

Table 308. Bit Descriptions for SPI0_TX, SPI1_TX

Bits	Bit Name	Settings	Description	Reset	Access
[15:8]	BYTE2		8-Bit Transmit Buffer. These 8 bits are used only in DMA mode, where all FIFO accesses occur as half word accesses. They return 0 if DMA is disabled.	0x0	W
[7:0]	BYTE1		8-Bit Transmit Buffer.	0x0	W

BAUD RATE SELECTION REGISTERS

Address: 0x4000400C, Reset: 0x0000, Name: SPI0_DIV

Address: 0x4002400C, Reset: 0x0000, Name: SPI1_DIV

Table 309. Bit Descriptions for SPI0_DIV, SPI1_DIV

Bits	Bit Name	Settings	Description	Reset	Access
[15:6]	Reserved		Reserved.	0x0	R
[5:0]	VALUE		SPI Clock Divider. The clock divider value is the factor used to divide UCLK to generate the serial clock.	0x0	R/W

CONFIGURATION REGISTERS

Address: 0x40004010, Reset: 0x0000, Name: SPI0_CTL

Address: 0x40024010, Reset: 0x0000, Name: SPI1_CTL

Table 310. Bit Descriptions for SPI0_CTL, SPI1_CTL

Bits	Bit Name	Settings	Description	Reset	Access
15	Reserved		Reserved.	0x0	R

REGISTER DETAILS: SPI0/SPI1

Table 310. Bit Descriptions for SPI0_CTL, SPI1_CTL (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
14	CSRST		Reset Mode for Chip Select Error Bit.	0x0	R/W
		0	The bit counter continues from where it stopped. The SPI can receive the remaining bits when the chip select is asserted and user code must ignore the SPIx_STAT, Bit 12 interrupt.		
		1	The bit counter is reset after a chip select error condition and the user code is expected to clear SPIx_CTL, Bit 0. Set this bit for a recovery after a chip select error.		
13	TFLUSH		SPI Transmit FIFO Flush Enable.	0x0	R/W
		0	Disable transmit FIFO flushing.		
		1	Flush the transmit 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 transmit FIFO are ignored while this bit is set.		
12	RFLUSH		SPI Receive FIFO Flush Enable.	0x0	R/W
		0	Disable receive FIFO flushing.		
		1	Flush the receive 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 set and the TIM bit = 0, a read of the receive FIFO initiates a transfer.		
11	CON		Continuous Transfer Enable.	0x0	R/W
		0	Disable continuous transfer. Each transfer consists of a single 8-bit serial transfer. If valid data exists in the SPIx_TX register, a new transfer is initiated after a stall period of one serial clock cycle.		
		1	Enable continuous transfer. In initiator mode, the transfer continues until no valid data is available in the transmit FIFO. Chip select is asserted and remains asserted for the duration of each 8-bit serial transfer until FIFO is empty.		
10	LOOPBACK		Loopback Enable.	0x0	R/W
		0	Normal mode.		
		1	Connect MISO to MOSI and test software.		
9	OEN		Target MISO Output Enable.	0x0	R/W
		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		SPI Receive Overflow Overwrite Enable.	0x0	R/W
		0	The new serial byte received is discarded.		
		1	The valid data in the receive register is overwritten by the new serial byte received.		
7	ZEN		Transmit Zeros Enable.	0x0	R/W
		0	Transmit the last transmitted value when there is no valid data in the transmit FIFO.		
		1	Transmit 0x00 when there is no valid data in the transmit FIFO.		
6	TIM		SPI Transfer and Interrupt Mode.	0x0	R/W
		0	Initiate transfer with a read of the SPIx_RX register. An interrupt only occurs when the receive FIFO is full.		
		1	Initiate transfer with a write to the SPIx_TX register. An interrupt only occurs when the transmit FIFO is empty.		
5	LSB		LSB First Transfer Enable.	0x0	R/W
		0	MSB transmitted first.		
		1	LSB transmitted first.		
4	WOM		SPI Wire-OR'ed Mode.	0x0	R/W
		0	Normal output levels.		
		1	Enables open circuit data output enable. External pull-up resistors required on data out pins.		
3	CPOL		Serial Clock Polarity.	0x0	R/W
		0	Serial clock idles low.		
		1	Serial clock idles high.		
2	CPHA		Serial Clock Phase Mode.	0x0	R/W
		0	Serial clock pulses at the end of each serial bit transfer.		
		1	Serial clock pulses at the beginning of each serial bit transfer.		
1	MASEN		Initiator Mode Enable.	0x0	R/W

REGISTER DETAILS: SPI0/SPI1

Table 310. Bit Descriptions for SPI0_CTL, SPI1_CTL (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		0	Enable target mode.		
		1	Enable initiator mode.		
0	SPIEN		SPI Enable.	0x0	R/W
		0	Disable the SPI.		
		1	Enable the SPI.		

INTERRUPT CONFIGURATION REGISTERS

Address: 0x40004014, Reset: 0x0000, Name: SPI0_IEN

Address: 0x40024014, Reset: 0x0000, Name: SPI1_IEN

Table 311. Bit Descriptions for SPI0_IEN, SPI1_IEN

Bits	Bit Name	Settings	Description	Reset	Access
15	Reserved		Reserved.	0x0	R
14	TXEMPTY		Transmit FIFO Empty Interrupt Enable. This bit enables the SPIx_STAT, Bit 2 interrupt whenever the transmit FIFO is emptied.	0x0	R/W
		0	TXEMPTY interrupt is disabled.		
		1	TXEMPTY interrupt is enabled.		
13	XFRDONE		SPI Transfer Completion Interrupt Enable. This bit enables the SPIx_STAT, Bit 1 interrupt.	0x0	R/W
		0	XFRDONE interrupt is disabled.		
		1	XFRDONE interrupt is enabled.		
12	TXDONE		SPI Transmit Done Interrupt Enable. This bit enables the SPIx_STAT, Bit 3 interrupt in read command mode. This interrupt can be used to signal the change of SPI transfer direction in read command mode.	0x0	R/W
		0	TXDONE interrupt is disabled.		
		1	TXDONE interrupt is enabled.		
11	RDY		Ready Signal Edge Interrupt Enable. This bit enables the SPIx_STAT, Bit 15 interrupt whenever an active edge occurs on P0.3 signals. If SPIx_FLOW_CTL, Bits[1:0] = 0b10, this bit is set whenever an active edge is detected on the P0.3 signal. If SPIx_FLOW_CTL, Bits[1:0] = 0b11, this bit is set if an active edge is detected on the MISO signal. If SPIx_FLOW_CTL, Bits[1:0] = 0b00 or 0b01, this bit is always 0. The active edge (rising or falling) is determined by SPIx_FLOW_CTL, Bit 4.	0x0	R/W
		0	Ready signal edge interrupt is disabled.		
		1	Ready signal edge interrupt is enabled.		
10	RXOVR		Receive Overflow Interrupt Enable.	0x0	R/W
		0	Receive overflow interrupt is disabled.		
		1	Receive overflow interrupt is enabled.		
9	TXUNDR		Transmit Underflow Interrupt Enable.	0x0	R/W
		0	Transmit underflow interrupt is disabled.		
		1	Transmit underflow interrupt is enabled.		
8	CS		Enable Interrupt on Every Chip Select Edge in Target Continuous Mode.	0x0	R/W
		0	No interrupt is generated and the status bits are not asserted.		
		1	If the SPI module is configured as a target in continuous mode, any edge on chip select generates an interrupt and the corresponding status bits (SPIx_STAT, Bit 14 and SPIx_STAT, Bit 13) are asserted. This bit has no effect if the SPI is not in continuous mode or if it is an initiator.		
[7:3]	Reserved		Reserved.	0x0	R
[2:0]	IRQMODE		SPI IRQ Mode Bits. These bits configure when the transmit or receive interrupts occur in a transfer. For DMA receive transfer, these bits are 0b000.	0x0	R/W
		000	Transmit interrupt occurs when 1 byte has been transferred. Receive interrupt occurs when 1 or more bytes have been received into the FIFO.		
		001	Transmit interrupt occurs when 2 bytes have been transferred. Receive interrupt occurs when 2 or more bytes have been received into the FIFO.		

REGISTER DETAILS: SPI0/SPI1

Table 311. Bit Descriptions for SPI0_IEN, SPI1_IEN (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		010	Transmit interrupt occurs when 3 bytes have been transferred. Receive interrupt occurs when 3 or more bytes have been received into the FIFO.		
		011	Transmit interrupt occurs when 4 bytes have been transferred. Receive interrupt occurs when 4 or more bytes have been received into the FIFO.		
		100	Transmit interrupt occurs when 5 bytes have been transferred. Receive interrupt occurs when 5 or more bytes have been received into the FIFO.		
		101	Transmit interrupt occurs when 6 bytes have been transferred. Receive interrupt occurs when 6 or more bytes have been received into the FIFO.		
		110	Transmit interrupt occurs when 7 bytes have been transferred. Receive interrupt occurs when 7 or more bytes have been received into the FIFO.		
		111	Transmit interrupt occurs when 8 bytes have been transferred. Receive interrupt occurs when the FIFO is full.		

TRANSFER BYTE COUNT REGISTERS

Address: 0x40004018, Reset: 0x0000, Name: SPI0_CNT

Address: 0x40024018, Reset: 0x0000, Name: SPI1_CNT

This register is only used in initiator mode.

Table 312. Bit Descriptions for SPI0_CNT, SPI1_CNT

Bits	Bit Name	Settings	Description	Reset	Access
15	FRAMECONT		Continue Frame. Use this bit in conjunction with the SPIx_CTL, Bit 11 and SPIx_CNT, Bits[13:0] fields. This bit is used to control SPI data framing.	0x0	R/W
		0	When writing to this bit, if this bit is cleared, the SPI initiator transfers only one frame of SPIx_CNT, Bits[13:0]. When reading this bit, if the value bits > 0, stop SPI transfers after the specified number of bytes.		
		1	When writing to this bit, the SPI initiator transfers data in frames of SPIx_CNT, Bits[13:0] bytes per frame. If SPIx_CNT, Bits[13:0] = 0, this field has no effect because the SPI initiator continues with transfers as long as the transmit or receive FIFO is ready. If SPIx_CTL, Bit 11 = 0, this field has no effect because all SPI frames are a single byte wide, irrespective of other control fields. When reading this bit, continue SPI transfers as long as the transmit or receive FIFO is ready.		
14	Reserved		Reserved.	0x0	R
[13:0]	VALUE		Transfer Byte Count. This field specifies the number of bytes to be transferred. It is used in both receive and transmit transfer types. This value ensures that an initiator mode transfer terminates at the proper time and that 16-bit SPIx_DMA transfers are byte padded or discarded as required to match odd transfer counts. Reset by clearing SPIx_CTL, Bit 0 to 0.	0x0	R/W

DMA ENABLE REGISTERS

Address: 0x4000401C, Reset: 0x0000, Name: SPI0_DMA

Address: 0x4002401C, Reset: 0x0000, Name: SPI1_DMA

Table 313. Bit Descriptions for SPI0_DMA, SPI1_DMA

Bits	Bit Name	Settings	Description	Reset	Access
[15:3]	Reserved		Reserved.	0x0	R
2	RXEN		Enable Receive DMA Request.	0x0	R/W
		0	Disable receive DMA interrupt.		
		1	Enable receive DMA interrupt.		
1	TXEN		Enable Transmit DMA Request.	0x0	R/W
		0	Disable transmit DMA interrupt.		
		1	Enable transmit DMA interrupt.		
0	EN		Enable DMA for Data Transfer. Set by user code to start a DMA transfer. Cleared by user code at the end of DMA transfer. This bit must be cleared to prevent extra DMA requests to the µDMA controller.	0x0	R/W

REGISTER DETAILS: SPI0/SPI1

FIFO STATUS REGISTERS

Address: 0x40004020, Reset: 0x0000, Name: SPI0_FIFO_STAT

Address: 0x40024020, Reset: 0x0000, Name: SPI1_FIFO_STAT

Table 314. Bit Descriptions for SPI0_FIFO_STAT, SPI1_FIFO_STAT

Bits	Bit Name	Settings	Description	Reset	Access
[15:12]	Reserved		Reserved.	0x0	R
[11:8]	RX		SPI Receive FIFO Status. This field specifies the number of bytes in the receive FIFO when the DMA is disabled. In DMA mode, these bits refer to the number of half words in the receive FIFO.	0x0	R
		0000	Receive FIFO empty.		
		0001	1 valid byte or half word in the receive FIFO.		
		0010	2 valid bytes or half words in the receive FIFO.		
		0011	3 valid bytes or half words in the receive FIFO.		
		0100	4 valid bytes or half words in the receive FIFO.		
		0101	5 valid bytes or half words in the receive FIFO.		
		0110	6 valid bytes or half words in the receive FIFO.		
		0111	7 valid bytes or half words in the receive FIFO.		
		1000	8 valid bytes or half words in the receive FIFO. Receive FIFO full.		
[7:4]	Reserved		Reserved.	0x0	R
[3:0]	TX		SPI Transmit FIFO Status. This field specifies the number of bytes in the transmit FIFO when DMA is disabled. In DMA mode, these bits refer to the number of half words in the transmit FIFO.	0x0	R
		0000	Transmit FIFO empty.		
		0001	1 valid byte or half word in the transmit FIFO.		
		0010	2 valid bytes or half words in the transmit FIFO.		
		0011	3 valid bytes or half words in the transmit FIFO.		
		0100	4 valid bytes or half words in the transmit FIFO.		
		0101	5 valid bytes or half words in the transmit FIFO.		
		0110	6 valid bytes or half words in the transmit FIFO.		
		0111	7 valid bytes or half words in the transmit FIFO.		
		1000	8 valid bytes or half words in the transmit FIFO. Transmit FIFO full.		

READ CONTROL REGISTERS

Address: 0x40004024, Reset: 0x0000, Name: SPI0_RD_CTL

Address: 0x40024024, Reset: 0x0000, Name: SPI1_RD_CTL

This register is only used in initiator mode.

Table 315. Bit Descriptions for SPI0_RD_CTL, SPI1_RD_CTL

Bits	Bit Name	Settings	Description	Reset	Access
[15:9]	Reserved		Reserved.	0x0	R
8	THREEPIN		3-Pin SPI Mode. This field specifies if the SPI interface has a bidirectional data pin (3-pin interface) or dedicated unidirectional data pins for transmit and receive (4-pin interface). This bit is only valid in read command mode and when SPIx_FLOW_CTL, Bits[1:0] = 0b01. If 3-pin mode is selected, the MOSI pin is driven by the initiator during the transmit phase. After a wait time of SPI_WAIT_TMR SCLK cycles, the target is expected to drive the same MOSI pin. Program SPI_FLOW_CTL, Bits[1:0] to 01 to introduce wait states for allowing turnaround time. Otherwise, the target only has a turnaround time of a half SCLK period (between sampling and driving edges of SCLK). If this mode is used, set the OVERLAP bit = 0.	0x0	R/W
		0	SPI is a 4-pin interface.		
		1	SPI is a 3-pin interface.		
[7:6]	Reserved		Reserved.	0x0	R
[5:2]	TXBYTES		Transmit Byte Count Minus 1 for Read Command. This field specifies the number of bytes to be transmitted minus 1 before reading data from a target. This field can accept values from 0 to 15,	0x0	R/W

REGISTER DETAILS: SPI0/SPI1

Table 315. Bit Descriptions for SPI0_RD_CTL, SPI1_RD_CTL (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
			corresponding to 1 to 16 transmit bytes. Accepted bytes include all the bytes that must be sent out to the target, such as command and address (if required). The design does not differentiate between the command and address, but transmits the specified number of bytes from the transmit FIFO. If there is a latency between the command transmission and data reception, account for the number of transmit bytes (mostly 0s) to be padded for that delay.		
1	OVERLAP		Transmit and Receive Overlap Mode. This bit specifies if the start of transmit and receive overlap. In most of the targets, the read data starts only after the initiator completes the transmission of command and address. This transfer is nonoverlapping. In some targets, there may be status bytes sent out while the command is being received, called overlapping mode. Therefore, start receiving the bytes from the beginning of the chip select frame. In case of overlapping mode, SPIx_CNT, Bits[13:0] refer to the total number of bytes to be received. Account for the extra status bytes (in addition to the actual read bytes) while programming SPIx_CNT, Bits[13:0].	0x0	R/W
		0	Transmit receive overlap is disabled.		
		1	Transmit receive overlap is enabled.		
0	CMDEN		Read Command Enable. SPI read command mode where a command and address are transmitted and read data is expected in the same chip select frame. If this bit is cleared, all other fields of the SPIx_RD_CTL register, SPIx_FLOW_CTL register, and SPIx_WAIT_TMR register have no effect.	0x0	R/W
		0	Read command mode is disabled.		
		1	Read command mode is enabled.		

FLOW CONTROL REGISTERS

Address: 0x40004028, Reset: 0x0000, Name: SPI0_FLOW_CTL

Address: 0x40024028, Reset: 0x0000, Name: SPI1_FLOW_CTL

This register is only used in initiator mode.

Table 316. Bit Descriptions for SPI0_FLOW_CTL, SPI1_FLOW_CTL

Bits	Bit Name	Settings	Description	Reset	Access
[15:12]	Reserved		Reserved.	0x0	R
[11:8]	RDBURSTSZ		Read Data Burst Size Minus 1. This field specifies the number of bytes to be received minus 1 in a single burst from a target before waiting for flow control. This bit is not valid if SPIx_FLOW_CTL, Bits[1:0] = 0b00. For all other values of SPIx_FLOW_CTL, Bits[1:0], this field is valid. This field can take values from 0 to 15, implying a read burst of 1 byte to 16 bytes. This mode is useful for reading fixed width conversion results periodically.	0x0	R/W
[7:5]	Reserved		Reserved.	0x0	R
4	RDYPOL		Polarity of P0.3 Line. This field specifies the polarity of the P0.3 pin, which indicates that the read data of the target is ready. If SPIx_FLOW_CTL, Bits[1:0] = 0b10, this field refers to the polarity of the P0.3 pin. Otherwise, if SPIx_FLOW_CTL, Bits[1:0] = 0b11, this field refers to the polarity of the MISO line. For all other values of SPIx_FLOW_CTL, Bits[1:0], this bit is ignored.	0x0	R/W
		0	Polarity is active high. SPI initiator waits until P0.3 becomes high.		
		1	Polarity is active low. SPI initiator waits until P0.3 becomes low.		
[3:2]	Reserved		Reserved.	0x0	R
[1:0]	MODE		Flow Control Mode. Flow control configuration for data reads. When the P0.3 signal is used for flow control, P0.3 can be any control signal that is tied to this P0.3 input of the SPI module.	0x0	R/W
		00	Flow control is disabled.		
		01	Flow control is based on timer SPIx_WAIT_TMR.		
		10	Flow control is based on P0.3 signal.		
		11	Flow control is based on MISO pin.		

REGISTER DETAILS: SPI0/SPI1

WAIT TIMER FOR FLOW CONTROL REGISTERS

Address: 0x4000402C, Reset: 0x0000, Name: SPI0_WAIT_TMR

Address: 0x4002402C, Reset: 0x0000, Name: SPI1_WAIT_TMR

This register is only used in initiator mode.

Table 317. Bit Descriptions for SPI0_WAIT_TMR, SPI1_WAIT_TMR

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	VALUE		Wait Timer for Flow Control. This field specifies the number of SCLK cycles to wait before continuing the SPI read. This field can take values of 0 to 65,535. This field is only valid if SPIx_FLOW_CTL, Bits[1:0] = 0b01b. For all other values of SPIx_FLOW_CTL, Bits[1:0], this field is ignored. A value of 0 implies a wait time of 1 SCLK cycle.	0x0000	R/W

CHIP SELECT OVERRIDE REGISTERS

Address: 0x40004034, Reset: 0x0000, Name: SPI0_CS_OVERRIDE

Address: 0x40024034, Reset: 0x0000, Name: SPI1_CS_OVERRIDE

This register is only used in initiator mode.

Table 318. Bit Descriptions for CS_OVERRIDE

Bits	Bit Name	Settings	Description	Reset	Access
[15:2]	Reserved		Reserved.	0x0	R
[1:0]	CTL		Chip Select Override Control. This bit overrides the chip select output from the initiator state machine. This bit may be needed for special SPI transfers. Do not use for normal SPI transfers.	0x0	R/W
		00	Chip select is not forced.		
		01	Chip select is forced to drive 1.		
		10	Chip select is forced to drive 0.		
		11	Chip select is not forced.		

UART SERIAL INTERFACE

UART OVERVIEW

The UART peripheral is a full duplex UART, compatible with the industry-standard 16450 UART or 16550 UART. The UART is responsible for converting data between serial and parallel formats. The serial communication follows an asynchronous protocol, supporting various word lengths, stop bits, and parity generation options. This UART also contains interrupt handling hardware. The UART features a fractional divider that facilitates high accuracy baud rate generation.

Interrupts can be generated from several unique events, such as full or empty data buffer, transfer error detection, and break detection. Modem signals are supported by the UART block, but these signals are not brought to external pins. The modem signals are tied off internally.

UART FEATURES

The ADuCM355 features an industry-standard 16450 UART or 16550 UART peripheral with support for DMA.

UART OPERATION

Serial Communications

An asynchronous serial communication protocol is followed with these options:

- 5 data bits to 8 data bits.
- 1, 2, or 1.5 stop bits.
- None, even, or odd parity.

All data-words require a start bit and at least one stop bit, which creates a range from 7 bits to 12 bits for each word. Transmit operation is initiated by writing to the transmit holding register (COMTX). After a synchronization delay, the data is moved to the internal transmit shift register, where it is shifted out at a baud (bit) rate equal to the following equation with start, stop, and parity bits appended as required:

$$Baud\ Rate = PCLK(M + (N/2048) \times 2^{OSR+2} \times UART\_COMDIV) \quad (28)$$

where:
PCLK is the divided root clock as configured via CNT1, Bits[13:8].
M = 1 to 3 (COMFBR, Bits[12:11]).
N = 0 to 2047 (COMFBR, Bits[10:0]).
UART_COMDIV = 1 to 65,536.

All data-words begin with a low going start bit. The transfer of COMTX to the transmit shift register causes the transmit register empty status flag to be set. The receive operation uses the same data format as the transmit configuration except for the number of stop bits, which is always one. After detection of the start bit, the received word is shifted into the internal receive shift register. After the appropriate number of bits (including stop bits) is received, the data and any status are updated, and the receive shift register is transferred to the receive buffer register (COMRX). The receive buffer register full status flag (COMIIR, Bits[3:1] = 0b010) is updated upon the transfer of the received word to this buffer and the appropriate synchronization delay.

A sampling clock equal to 2^{OSR+2} times the baud rate is used to sample the data as close to the midpoint of the bit as possible. A receive filter is also present that removes spurious pulses of less than two times the sampling clock period. Data is transmitted and received LSB first.

Programmed Input and Output Mode

In programmed input and output mode, the software is responsible for moving data to and from the UART. This movement is typically accomplished by interrupt service routines that respond to the transmit and receive interrupts by either reading or writing data as appropriate. This mode puts certain constraints on the software itself in that the software must respond within a certain time to prevent overflow errors from occurring in the receive channel.

Polling the status flag is processor intensive and not typically performed unless the system can tolerate the overhead. Interrupts can be disabled using the COMIEN register.

Do not write to COMTX when it is not empty or read COMRX when it is not full, because such actions produce incorrect results. When COMTX is not empty, COMTX is overwritten by the new word, and the previous word is never transmitted. When COMRX is not full, the previously received word is read again. Avoid these errors by correctly using either interrupts or status register polling. These errors are not detected in hardware.

UART SERIAL INTERFACE

Interrupts

The UART peripheral has one interrupt output to the interrupt controller for both receive and transmit interrupts. The COMIIR register must be read by the software to determine the cause of the interrupt. In DMA mode, the break interrupt is not available. When receiving in input or output mode, the interrupt is generated for the following cases:

- COMRX is full.
- Receive overflow error.
- Receive parity error.
- Receive framing error.
- Receive FIFO timeout if FIFO (16550 UART) is enabled.
- Break interrupt (UART input pin UART_SIN held low).
- COMTX empty.

Buffer Requirements

This UART is double buffered, meaning it has a hold register and a shift register.

FIFO Mode (16550 UART)

The 16-byte deep transmit FIFO and receive FIFO are implemented. Therefore, the UART is compatible with the industry-standard 16550 UART. By default, these FIFOs are disabled. To enable them, set COMFCR, Bit 0. When enabled, the internal FIFOs allow 16 bytes to be stored in both the receive and transmit modes of operation, and 3 bits of error data per byte in the receive FIFO.

The interrupt and DMA trigger for the number of bytes received into the receive FIFO is programmed via COMFCR, Bits[7:6]. The DMA requests are programmed by the COMFCR, Bit 3. If this bit is set, the FIFO must also be enabled by setting COMFCR, Bit 0 to 1. If the remaining bytes in a packet are less than the interrupt trigger number, a timeout interrupt occurs. This timeout is indicated by COMIIR, Bits[3:1] = 0b110. This timeout period is equal to the period of four consecutive characters where a single character time is one start bit, n data bits, one parity bit, and one stop bit, where n depends on the word length selected by COMLCR, Bits[1:0].

DMA Mode

In DMA mode, user code does not move data to and from the UART. DMA request signals entering the external DMA block indicate that the UART is ready to transmit or receive data. These DMA request signals can be disabled in the COMIEN register.

Automatic Baud Rate Detection

The automatic baud detection (ABD) block is used to match the baud rates of two UART devices automatically. The receiver must be enabled to detect the mode before a common baud rate is configured. The COMACR, Bit 0 enables the receiver to work in ABD mode. A 20-bit counter logic counts the number of cycles between the programmed rising or falling edge and another rising or falling edge. An interrupt is generated after the expected edges are reached. The counter can overflow and generate a timeout interrupt, such as when there is a continuous break condition or no expected edges.

For example, if the data byte being received is 0x0D (0b00001101, resulting in a carriage return) in 8-bit mode without a parity bit, LSB first, each bit reads as DATA0 = 1, DATA1 = 0, DATA2 = 1, DATA3 = 1, and DATA4 = DATA5 = DATA6 = DATA7 = 0.

There are three falling edges and three rising edges. The COMACR, Bits[6:4] can be written to 1 decimal (second edge), and COMACR, Bits[11:8] can be written to 5 decimal (sixth edge) to count between the first rising edge and the second rising edge. See Figure 63 for more details.

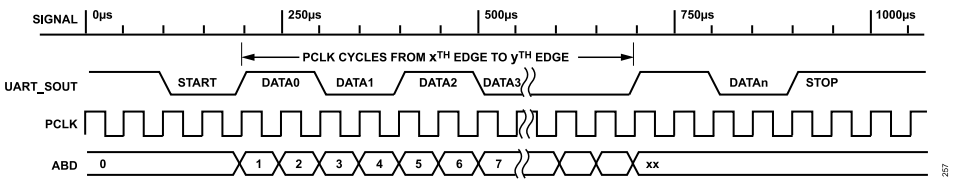

Figure 63. Autobaud Rate Example

UART SERIAL INTERFACE

Similarly, for 0x7F (0b01111111, the ASCII DEL value delete key), COMACR, Bits[6:4] = 1 and COMACR, Bits[11:8] = 3 to count between the first rising edge and the second rising edge. Automatic baud rate must be disabled to clear the internal counter and reenabled for another sequence (if required). Based on the UART baud rate configuration, the ABD result can be calculated as follows:

CNT, Bits[19:0] = *CountedBits* × 2*OSR* + 2 × *COMDIV* × (*COMFBR, Bits[12:11]* + *COMFBR, Bits[10:0]*/2048)

where:
CountedBits is the effective number of bits between an active starting edge and ending edge. It is determined by the application code on the selected edges and character used for ABD.
COMDIV is calculated as follows:

If CNT, Bits[19:0] < 8 × CountedBits, then OSR = 0, COMDIV = 1, and COMFBR, Bits[10:0] = 512 × CNT, Bits[19:0]/CountedBits − 2048.

If CNT, Bits[19:0] < 16 × CountedBits, then OSR = 1, COMDIV = 1, and COMFBR, Bits[10:0] = 256 × CNT, Bits[19:0]/CountedBits − 2048.

If CNT, Bits[19:0] < 32 × CountedBits, then OSR = 2, COMDIV = 1, and COMFBR, Bits[10:0] = 128 × CNT, Bits[19:0]/CountedBits − 2048.

If CNT, Bits[19:0] ≥ 32 × CountedBits, then OSR = 3.

If CNT, Bits[19:0] is exactly divided by 32 × CountedBits, then COMDIV = (CNT, Bits[19:0]/32)/CountedBits.

Otherwise, COMDIV = $2\log_2$((CNT[19:0]/32)/CountedBits) and COMFBR, Bits[10:0] = (((64 × CNT, Bits[19:0])/COMDIV)/ CountedBits) − 2048.

To reduce truncation error, the DIVM field (COMFBR, Bits[12:11]) is set to 1. The DIV field (COMDIV, Bits[15:0]) is set to the nearest power of 2. COMASRH, Bits[7:0] and COMASRL, Bits[15:4] make up CNT, Bits[19:0].

Clock Gating

The clock driving the UART logic is automatically gated off when idle, and not accessed. This automatic clock gating cannot be disabled by COMCTL, Bit 1.

UART and Power-Down Modes

Complete ongoing UART transfers before powering down the chip into hibernate mode. Alternatively, disable the UART by clearing the COMDIV register to 0x0000 before placing the device into hibernation. If hibernate mode is selected while a UART transfer is on, the transfer does not continue on a return from hibernation. All the intermediate data, states, and status logic in the UART are cleared. However, the transmit pad (UART_SOUT) remains active in the hibernate mode while transmitting. After hibernation, the UART can be enabled by setting the COMDIV register, if previously cleared. If DMA mode is needed, COMIEN, Bits[5:4] must be configured.

Table 319 details registers retained through hibernate mode. All other registers and internal logic are cleared by a hardware default value.

Table 319. Registers Retained Through Hibernate Mode

Register	Affected Bits
COMIEN	ELSI, ERBFI
COMLCR	BRK, SP, EPS, PEN, WLS
COMFCR	RFTRIG, FDMAMD, FIFOEN
COMFBR	FBEN, DIVM, DIVN
COMDIV	DIV
COMLCR2	OSR
COMCTL	RXINV, FORCECLKON
COMRSC	DISTX, DISRX, OENSP, OENP

Recommendations for UART Receive Wake-Up from Hibernate Mode

If the UART receive input is used to wake the ADuCM355 from hibernate mode, keep in mind that the UART block, along with the rest of the ADuCM355 chip, requires 10 µs settling time after the first falling edge of the UART wake-up byte. This delay means that the first UART receive byte may not be read correctly by the ADuCM355, especially with UART baud rates ≥ 57,600.

For reliable exit of hibernate mode via the UART receive pin, perform one of the two following actions:

UART SERIAL INTERFACE

- Ensure that the wake-up character received by the ADuCM355 is a break byte. On receiving a break byte, the ADuCM355 UART sets the break indicator status flag in the COMLSR, Bit 4 register. This flag generates an interrupt.
- Pull the ADuCM355 low for a period ≤1 µs. This setting triggers the UART wake-up. After ≥10 µs, normal UART communications can proceed.

REGISTER SUMMARY: UART

Table 320. UART Register Summary

Address	Name	Description	Reset	Access
0x40005000	COMTX	Transmit holding	0x0000	W
0x40005000	COMRX	Receive buffer	0x0000	R
0x40005004	COMIEN	Interrupt enabler	0x0000	R/W
0x40005008	COMIIR	Interrupt identification	0x0001	RC
0x4000500C	COMLCR	Line control	0x0000	R/W
0x40005010	COMMCR	Modem control	0x0000	R/W
0x40005014	COMLSR	Line status	0x0060	RC
0x40005018	COMMSR	Modem status	0x0000	RC
0x4000501C	COMSCR	Scratch buffer	0x0000	R/W
0x40005020	COMFCR	FIFO control	0x0000	R/W
0x40005024	COMFBR	Fractional baud rate	0x0000	R/W
0x40005028	COMDIV	Baud rate divider	0x0000	R/W
0x4000502C	COMLCR2	Second line control	0x0002	R/W
0x40005030	COMCTL	UART control	0x0100	R/W
0x40005034	COMRFC	Receive FIFO count	0x0000	R
0x40005038	COMTFC	Transmit FIFO count	0x0000	R
0x4000503C	COMRSC	RS485 half-duplex control	0x0000	R/W
0x40005040	COMACR	Autobaud control	0x0000	R/W
0x40005044	COMASRL	Autobaud status (low)	0x0000	R
0x40005048	COMASRH	Autobaud status (high)	0x0000	R

REGISTER DETAILS: UART

TRANSMIT HOLDING REGISTER

Address: 0x40005000, Reset: 0x0000, Name: COMTX

COMRX and COMTX share the same address although they are implemented as different registers. If these registers are written to, the user accesses the transmit holding register (COMTX). If these registers are read from, the user accesses the receive buffer register (COMRX).

Table 321. Bit Descriptions for COMTX

Bits	Bit Name	Settings	Description	Reset	Access
[15:8]	Reserved		Reserved.	0x0	R
[7:0]	THR		Transmit Holding Register. This is an 8-bit register to which the user can write the data to be sent. If COMIEN, Bit 1 is set, an interrupt generates when COMTX is empty. If user code sets COMIEN, Bit 1 when COMTX is already empty, an interrupt generates immediately.	0x0	W

RECEIVE BUFFER REGISTER

Address: 0x40005000, Reset: 0x0000, Name: COMRX

Table 322. Bit Descriptions for COMRX

Bits	Bit Name	Settings	Description	Reset	Access
[15:8]	Reserved		Reserved.	0x0	R
[7:0]	RBR		Receive Buffer Register. This is an 8-bit register from which the user can read received data. If COMIEN, Bit 0 is set, an interrupt generates when this register is fully loaded with the received data via the serial input port. If user code sets the COMIEN, Bit 0 while COMRX is full, an interrupt generates immediately.	0x0	R

INTERRUPT ENABLE REGISTER

Address: 0x40005004, Reset: 0x0000, Name: COMIEN

COMIEN is the interrupt enable register that configures which interrupt source generates the interrupt. Only the lowest four bits in this register enable interrupts. Bit 4 and Bit 5 enable UART DMA signals. The UART DMA channel and interrupt must be configured in the DMA block.

Table 323. Bit Descriptions for COMIEN

Bits	Bit Name	Settings	Description	Reset	Access
[15:6]	Reserved		Reserved.	0x0	R
5	EDMAR		DMA Requests in Receive Mode.	0x0	R/W
		0	DMA requests disabled.		
		1	DMA requests enabled.		
4	EDMAT		DMA Requests in Transmit Mode.	0x0	R/W
		0	DMA requests are disabled.		
		1	DMA requests are enabled.		
3	EDSSI		Modem Status Interrupt. Interrupt is generated when COMMSR, Bits[3:0] is set.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
2	ELSI		Receive Status Interrupt.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
1	ETBEI		Transmit Buffer Empty Interrupt.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		
0	ERBFI		Receive Buffer Full Interrupt.	0x0	R/W
		0	Interrupt disabled.		
		1	Interrupt enabled.		

REGISTER DETAILS: UART

INTERRUPT IDENTIFICATION REGISTER

Address: 0x40005008, Reset: 0x0001, Name: COMIIR

Table 324. Bit Descriptions for COMIIR

Bits	Bit Name	Settings	Description	Reset	Access
[15:8]	Reserved		Reserved.	0x0	R
[7:6]	FEND		FIFO Enabled.	0x0	R
		00	FIFO not enabled, 16450 UART mode.		
		11	FIFO enabled, 16550 UART mode.		
[5:4]	Reserved		Reserved.	0x0	R
[3:1]	STA		Interrupt Status. When NIRQ is active low, this bit indicates an interrupt and the following decoding for this bit is used.	0x0	RC
		000	Modem status interrupt. Read COMMSR to clear.		
		001	Transmit buffer empty interrupt. Write to COMTX or read COMIIR to clear.		
		010	Receive buffer full interrupt. Read COMRX to clear.		
		110	Receive FIFO timed out. Read COMRX to clear.		
		011	Receive line status interrupt. Read COMLSR to clear.		
0	NIRQ		Interrupt flag.	0x1	RC

LINE CONTROL REGISTER

Address: 0x4000500C, Reset: 0x0000, Name: COMLCR

Table 325. Bit Descriptions for COMLCR

Bits	Bit Name	Settings	Description	Reset	Access
[15:7]	Reserved		Reserved.	0x0	R
6	BRK		Set Break.	0x0	R/W
		1	Force UART_SOUT pin to 0.		
		0	Normal UART_SOUT pin operation.		
5	SP		Stick Parity. Used to force parity to defined values. When set, the parity is based on the following bit settings: When EPS = 1 and PEN = 1, the parity is forced to 0. When EPS = 0 and PEN = 1, the parity is forced to 1. When EPS = X and PEN = 0, no parity is transmitted.	0x0	R/W
		0	Parity is not forced based on EPS and PEN.		
		1	Parity forced based on EPS and PEN.		
4	EPS		Parity Select. This bit only has meaning if parity is enabled (PEN set).	0x0	R/W
		0	Odd parity is transmitted and checked.		
		1	Even parity is transmitted and checked.		
3	PEN		Parity Enable. This bit is used to enable parity to be transmitted and checked. The value transmitted and the value checked are based on the settings of EPS and SP.	0x0	R/W
		0	Parity is not transmitted or checked.		
		1	Parity is transmitted and checked.		
2	STOP		Stop Bit. Used to control the number of stop bits transmitted. In all cases, only the first stop bit is evaluated when data is received.	0x0	R/W
		0	Send one stop bit regardless of the word length in the WLS bit.		
		1	Send a number of stop bits based on the word length. Transmit 1.5 stop bits if the word length is 5 bits (WLS = 00), or 2 stop bits if the word length is 6 (WLS = 01), 7 (WLS = 10), or 8 bits (WLS = 11).		
[1:0]	WLS		Word Length Select. Selects the number of bits per transmission.	0x0	R/W
		00	5 bits.		
		01	6 bits.		
		10	7 bits.		
		11	8 bits.		

REGISTER DETAILS: UART

MODEM CONTROL REGISTER

Address: 0x40005010, Reset: 0x0000, Name: COMMCR

Table 326. Bit Descriptions for COMMCR

Bits	Bit Name	Settings	Description	Reset	Access
[15:5]	Reserved		Reserved.	0x0	R
4	LOOPBACK		Loopback Mode. In loopback mode, the UART_SOUT is forced high. The modem signals are also directly connected to the status inputs (RTS in COMMCR to CTS in COMMSR, DTR in COMMCR to DSR in COMMSR, OUT1 in COMMCR to RI in COMMSR, and OUT2 in COMMCR to DCD in COMMSR).	0x0	R/W
		0	Normal operation, loopback disabled.		
		1	Loop back enabled.		
3	OUT2		Output 2.	0x0	R/W
		0	Force OUT2 to a Logic 1.		
		1	Force OUT2 to a Logic 0.		
2	OUT1		Output 1.	0x0	R/W
		0	Force OUT1 to a Logic 1.		
		1	Force OUT1 to a Logic 0.		
1	RTS		Request to Send.	0x0	R/W
		0	Force RTS to a Logic 1.		
		1	Force RTS to a Logic 0.		
0	DTR		Data Terminal Ready.	0x0	R/W
		0	Force DTR to a Logic 1.		
		1	Force DTR to a Logic 0.		

LINE STATUS REGISTER

Address: 0x40005014, Reset: 0x0060, Name: COMLSR

Table 327. Bit Descriptions for COMLSR

Bits	Bit Name	Settings	Description	Reset	Access
[15:6]	Reserved		Reserved.	0x0	R
7	FIFOERR		Data Byte(s) in Receive FIFO have Parity Error, Frame Error, or Break Indication. Only used in 16550 UART mode. This bit is cleared if there are no more errors in the receive FIFO.	0x0	RC
6	TEMT		COMTX and Shift Register Empty Status.	0x1	R
		0	COMTX has been written to and contains data to be transmitted. Take care not to overwrite its value.		
		1	COMTX and the transmit shift register are empty and it is safe to write new data to COMTX. Data has been transmitted.		
5	THRE		COMTX Empty. This bit is cleared when COMRX is read.	0x1	R
		0	COMTX has been written to and contains data to be transmitted. Take care not to overwrite its value.		
		1	COMTX is empty and it is safe to write new data to COMTX. The previous data may not have been transmitted yet and can still be present in the shift register.		
4	BI		Break Indicator. If set, this bit self clears after COMLSR is read.	0x0	RC
		0	UART_SIN was not detected to be longer than the maximum word length.		
		1	UART_SIN was held low for more than the maximum word length.		
3	FE		Framing Error. If set, this bit self clears after COMLSR is read.	0x0	RC
		0	No invalid stop bit was detected.		
		1	An invalid stop bit was detected on a received word.		
2	PE		Parity Error. If set, this bit self clears after COMLSR is read.	0x0	RC
		0	No parity error was detected.		
		1	A parity error occurred on a received word.		
1	OE		Overrun Error. If set, this bit self clears after COMLSR is read.	0x0	RC
		0	Receive data has not been overwritten.		

REGISTER DETAILS: UART

Table 327. Bit Descriptions for COMLSR (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		1	Receive data was overwritten by new data before COMRX was read.		
0	DR		Data Ready. This bit is cleared only by reading COMRX. If set, this bit does not self clear.	0x0	RC
		0	COMRX does not contain new receive data.		
		1	COMRX contains receive data that needs to be read.		

MODEM STATUS REGISTER

Address: 0x40005018, Reset: 0x0000, Name: COMMSR

Table 328. Bit Descriptions for COMMSR

Bits	Bit Name	Settings	Description	Reset	Access
[15:4]	Reserved		Reserved.	0x0	R
3	DDCD		Delta DCD. If set, this bit self clears after COMMSR is read.	0x0	R
		0	DCD has not changed state since COMMSR was last read.		
		1	DCD changed state since COMMSR was last read.		
2	TERI		Trailing Edge Ring Indicator. If set, this bit self clears after COMMSR is read.	0x0	R
		0	RI has not changed from 0 to 1 since COMMSR was last read.		
		1	RI changed from 0 to 1 since COMMSR was last read.		
1	DDSR		Delta DSR. If set, this bit self clears after COMMSR is read.	0x0	R
		0	DSR has not changed state since COMMSR was last read.		
		1	DSR changed state since COMMSR was last read.		
0	DCTS		Delta CTS. If set, this bit self clears after COMMSR is read.	0x0	R
		0	CTS has not changed state since COMMSR was last read.		
		1	CTS changed state since COMMSR was last read.		

SCRATCH BUFFER REGISTER

Address: 0x4000501C, Reset: 0x0000, Name: COMSCR

Table 329. Bit Descriptions for COMSCR

Bits	Bit Name	Settings	Description	Reset	Access
[15:8]	Reserved		Reserved.	0x0	R
[7:0]	SCR		Scratch. The scratch register is an 8-bit register used to store intermediate results. The value contained in the scratch register does not affect UART functionality or performance. Only eight bits of this register are implemented. Bits[15:8] are read only and always return 0x00 when read. SCR is writable with any value from 0 to 255. A read returns the last value written.	0x0	R/W

FIFO CONTROL REGISTER

Address: 0x40005020, Reset: 0x0000, Name: COMFCR

Table 330. Bit Descriptions for COMFCR

Bits	Bit Name	Settings	Description	Reset	Access
[15:8]	Reserved		Reserved.	0x0	R
[7:6]	RFTRIG		Receive FIFO Trigger Level.	0x0	R/W
		00	1 byte to trigger receive interrupt.		
		01	4 bytes to trigger receive interrupt.		
		10	8 bytes to trigger receive interrupt.		
		11	14 bytes to trigger receive interrupt.		
[5:4]	Reserved		Reserved.	0x0	R
3	FDMAMD		FIFO DMA Mode.	0x0	R/W
		0	Receive DMA request is asserted when there is data in RBR in the COMRX register or the receive FIFO, and deasserts when RBR or the receive FIFO is empty. A transmit DMA request is asserted when THR in		

REGISTER DETAILS: UART

Table 330. Bit Descriptions for COMFCR (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
			the COMTX register or the transmit FIFO is empty, and deasserts whenever data is written to the COMTX register.		
		1	Receive DMA request is asserted when the receive FIFO trigger level or timeout is reached, and deasserts when the receive FIFO is empty. The transmit DMA request is asserted when the transmit FIFO is empty and deasserts when the transmit FIFO is completely full.		
2	TFCLR		Clear Transmit FIFO.	0x0	W
		0	No effect.		
		1	Clear transmit FIFO.		
1	RFCLR		Clear Receive FIFO.	0x0	W
		0	No effect.		
		1	Clear receive FIFO.		
0	FIFOEN		FIFO Enabled to Work in 16550 UART Mode.	0x0	R/W

FRACTIONAL BAUD RATE REGISTER

Address: 0x40005024, Reset: 0x0000, Name: COMFBR

Table 331. Bit Descriptions for COMFBR

Bits	Bit Name	Settings	Description	Reset	Access
15	FBEN		Fractional Baud Rate Generator Enable. The generating of fractional baud rate and the final baud rate of UART operation are calculated using the following: *Baud Rate* = ((*UCLK*)/(2 × (*M* + *N*/2048)) 16 × *COMDIV*)	0x0	R/W
[14:13]	Reserved		Reserved.	0x0	R
[12:11]	DIVM		Fractional Baud Rate M Divide Bits, 1 to 3. This bit must not be 0.	0x0	R/W
[10:0]	DIVN		Fractional Baud Rate N Divide Bits, 0 to 2047.	0x0	R/W

BAUD RATE DIVIDER REGISTER

Address: 0x40005028, Reset: 0x0000, Name: COMDIV

Table 332. Bit Descriptions for COMDIV

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	DIV		Baud Rate Divider. Ensure that the COMDIV register is not 0. The range of allowed DIV values is from 1 to 65,535.	0x1	R/W

SECOND LINE CONTROL REGISTER

Address: 0x4000502C, Reset: 0x0002, Name: COMLCR2

Table 333. Bit Descriptions for COMLCR2

Bits	Bit Name	Settings	Description	Reset	Access
[15:2]	Reserved		Reserved.	0x0	R
[1:0]	OSR		Over Sample Rate.	0x2	R/W
		00	Over sample by 4.		
		01	Over sample by 8.		
		10	Over sample by 16.		
		11	Over sample by 32.		

UART CONTROL REGISTER

Address: 0x40005030, Reset: 0x0100, Name: COMCTL

Table 334. Bit Descriptions for COMCTL

Bits	Bit Name	Settings	Description	Reset	Access
[15:8]	REV		UART Revision ID.	0x1	R

REGISTER DETAILS: UART

Table 334. Bit Descriptions for COMCTL (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
[7:5]	Reserved		Reserved.	0x0	R
4	RXINV		Invert Receiver Line.	0x0	R/W
		0	Do not invert receiver line (idling high).		
		1	Invert receiver line (idling low).		
[3:2]	Reserved		Reserved.	0x0	R
1	FORCECLKON		Force PCLK to UART On All the Time.	0x0	R/W
		0	PCLK to UART automatically clock gated.		
		1	PCLK to UART always on.		
0	Reserved		Reserved.	0x0	R

RECEIVE FIFO COUNT REGISTER

Address: 0x40005034, Reset: 0x0000, Name: COMRFC

Table 335. Bit Descriptions for COMRFC

Bits	Bit Name	Settings	Description	Reset	Access
[15:5]	Reserved		Reserved.	0x0	R
[4:0]	RFC		Current Receive FIFO Data Bytes.	0x0	R

TRANSMIT FIFO COUNT REGISTER

Address: 0x40005038, Reset: 0x0000, Name: COMTFC

Table 336. Bit Descriptions for COMTFC

Bits	Bit Name	Settings	Description	Reset	Access
[15:5]	Reserved		Reserved.	0x0	R
[4:0]	TFC		Current Transmit FIFO Data Bytes.	0x0	R

RS485 HALF-DUPLEX CONTROL REGISTER

Address: 0x4000503C, Reset: 0x0000, Name: COMRSC

Table 337. Bit Descriptions for COMRSC

Bits	Bit Name	Settings	Description	Reset	Access
[15:4]	Reserved		Reserved.	0x0	R
3	DISTX		Disable Transmit when Receiving.	0x0	R/W
2	DISRX		Disable Receive when Transmitting.	0x0	R/W
1	OENSP		UART_SOUT Deassert Before Full Stop Bits.	0x0	R/W
		0	UART_SOUT deasserts at same time as full stop bits.		
		1	UART_SOUT deasserts half bit earlier than full stop bits.		
0	OENP		UART_SOUT Polarity.	0x0	R/W
		0	High active.		
		1	Low active.		

AUTOBAUD CONTROL REGISTER

Address: 0x40005040, Reset: 0x0000, Name: COMACR

Table 338. Bit Descriptions for COMACR

Bits	Bit Name	Reserved	Description	Reset	Access
[15:12]	Reserved		Reserved.	0x0	R
[11:8]	EEC		Ending Edge Count.	0x0	R/W
		0000	First edge.		
		0001	Second edge.		

REGISTER DETAILS: UART

Table 338. Bit Descriptions for COMACR (Continued)

Bits	Bit Name	Reserved	Description	Reset	Access
		0010	Third edge.		
		0011	Fourth edge.		
		0100	Fifth edge.		
		0101	Sixth edge.		
		0110	Seventh edge.		
		0111	Eighth edge.		
		1000	Ninth edge.		
7	Reserved		Reserved.	0x0	R
[6:4]	SEC		Starting Edge Count.	0x0	R/W
		000	First edge. Always the falling edge of UART start bit.		
		001	Second edge.		
		010	Third edge.		
		011	Fourth edge.		
		100	Fifth edge.		
		101	Sixth edge.		
		110	Seventh edge.		
		111	Eighth edge.		
3	Reserved		Reserved.	0x0	R
2	TOIEN		Enable Time Out Interrupt.	0x0	R/W
1	DNIEN		Enable Done Interrupt.	0x0	R/W
0	ABE		Autobaud Enable.	0x0	R/W

AUTOBAUD STATUS (LOW) REGISTER

Address: 0x40005044, Reset: 0x0000, Name: COMASRL

Table 339. Bit Descriptions for COMASRL

Bits	Bit Name	Settings	Description	Reset	Access
[15:4]	CNT[11:0]		Autobaud Counter Value.	0x0	R
3	NEETO		Timed Out Due to No Valid Ending Edge Found.	0x0	RC
2	NSETO		Timed Out Due to No Valid Start Edge Found.	0x0	RC
1	BRKTO		Timed Out Due to Long Time Break Condition.	0x0	RC
0	DONE		Autobaud Operation Completed.	0x0	RC

AUTOBAUD STATUS (HIGH) REGISTER

Address: 0x40005048, Reset: 0x0000, Name: COMASRH

Table 340. Bit Descriptions for COMASRH

Bits	Bit Name	Settings	Description	Reset	Access
[15:8]	Reserved		Reserved.	0x0	R
[7:0]	CNT[19:12]		Autobaud Counter Value.	0x0	R

DIGITAL DIE GENERAL-PURPOSE TIMERS

DIGITAL DIE GENERAL-PURPOSE TIMERS FEATURES

The ADuCM355 digital die integrates three identical general-purpose, 16-bit count up or count down timers: Timer 0, Timer 1, and Timer 2. These timers can be clocked from the 32 kHz internal low frequency oscillator, the PCLK, or the internal 26 MHz high frequency oscillator. Clock sources can be scaled down using a prescaler to 1, 4, 16, or 256. Free running mode and periodic mode are available. The timers have a capture events feature, with the capability to capture 32 different events on each timer. See Figure 64 for an overview of the general-purpose timers.

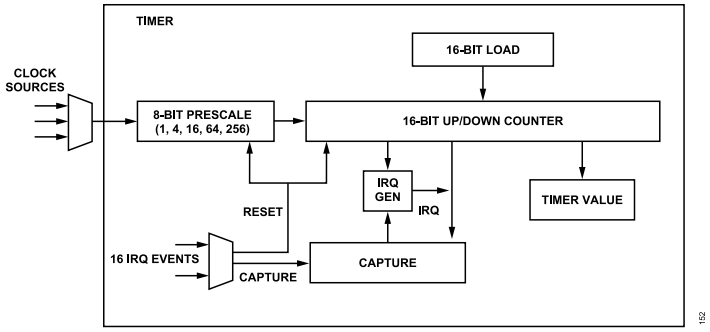

Figure 64. General-Purpose Timers Block Diagram

GENERAL-PURPOSE TIMERS OVERVIEW

The timers can either be in free running mode or periodic mode. In free running mode, the counter decrements from full scale to zero scale or increments from zero scale to full scale and then restarts. In periodic mode, the counter decrements or increments from the value in the load register (GPTx_LOAD, where x is 0 for Timer 0, 1 for Timer 1, and 2 for Timer 2) until zero scale or full scale is reached. The counter then restarts at the value stored in the load register.

The value of a counter can be read at any time by accessing its value register, GPTx_ACURCNT or GPTx_CURCNT. GPTx_ACURCNT assumes the timer and the CPU are synchronized (using the same clock source). Do not use GPTx_ACURCNT if the timer is using a different clock source, such as the low frequency oscillator. In this case, use GPTx_CURCNT. GPTx_CURCNT returns a synchronized timer value but has a slightly delayed result owing to the synchronization period required.

The CON0 register selects the timer mode, configures the clock source, selects count up or count down, starts the counter, and controls the event capture function.

An interrupt signal is generated each time the value of the counter reaches 0 when counting down or each time the counter value reaches the maximum value when counting up. Clear an IRQ by writing 1 to the time clear interrupt register of that particular timer (GPTx_CLRINT).

In addition, Timer 0, Timer 1, and Timer 2 have a capture register that is triggered by a selected IRQ source initial assertion. When triggered, the current timer value is copied to the GPTx_CAPTURE register, and the timer continues to run. This feature determines the assertion of an event with increased accuracy.

GENERAL-PURPOSE TIMER OPERATIONS

Free Running Mode

In free running mode, the timer is started by setting the enable bit (GPTx_CTL, Bit 4) to 1 and the mode bit (GPTx_CTL, Bit 3) to 0. The timer increments from zero scale or full scale to full scale or zero scale if counting up or down. Full scale is 216, which is 1 or 0xFFFF. Upon reaching full scale or zero scale, a timeout interrupt occurs and GPTx_STAT, Bit 0 is set. To clear the timer interrupt, user code must write 1 to GPTx_CLRINT, Bit 0. If GPTx_CTL, Bit 7 is set, the timer keeps counting and reloads when GPTx_CLRINT, Bit 0 is set to 1.

Periodic Mode

In periodic mode, the initial GPTx_LOAD value must be loaded before starting the timer by setting the enable bit (GPTx_CTL, Bit 4) to 1. The timer value either increments from the value in GPTx_LOAD to full scale or decrements from the value in GPTx_LOAD to zero scale, depending on the GPTx_CTL, Bit 2 settings (count up or down). Upon reaching full scale or zero scale, the timer generates an interrupt. GPTx_LOAD is reloaded into GPTx_CURCNT, and the timer continues counting up or down. The timer must be disabled prior to changing the GPTx_CTL or GPTx_LOAD register. If the GPTx_LOAD register is changed while the timer is being loaded, undefined results can occur. By default, the

DIGITAL DIE GENERAL-PURPOSE TIMERS

counter is reloaded automatically when generating the interrupt signal. If GPTx_CTL, Bit 7 is set to 1, the counter is also reloaded when user code writes GPTx_CLRINT, which allows user changes to GPTx_LOAD to take effect immediately instead of waiting until the next timeout.

The timer interval is calculated as follows:

If the timer is set to count down,

$$Interval = (GPTx\_LOAD \times Prescaler)/Clock\ Source \quad (29)$$

For example, if GPTx_LOAD = 0x100, prescaler = 4, and clock source = high frequency oscillator, the interval is 39.38 µs (where high frequency oscillator = 26 MHz).

If the timer is set to count up,

$$Interval = ((Full\ Scale - GPTx\_LOAD) \times Prescaler)/Clock\ Source \quad (30)$$

Asynchronous Clock Source

Timers are started by setting the enable bit (GPTx_CTL, Bit 4) to 1 in the control register of the corresponding timer. However, when the timer clock source is the low frequency oscillator, the following precautions must be taken.

- Do not write to GPTx_CTL if GPTx_STAT, Bit 6 is set. GPTx_STAT must be read prior to configuring GPTx_CTL. When GPTx_STAT, Bit 6 is cleared, the register can be modified, ensuring that synchronizing the timer control between the processor and the timer clock domains is complete. GPTx_STAT, Bit 6 is the timer busy status bit.
- After clearing the interrupt in GPTx_CLRINT, ensure that the register write has completed before returning from the interrupt handler. Use the data synchronization barrier (DSB) instruction if necessary and check that GPTx_STAT, Bit 7 = 0, as follows:

```
__asm void asmDSB()
{
nop
DSB
BX LR
}
```

- The value of a counter can be read at any time by accessing its value register (GPTx_CURCNT). In an asynchronous configuration, GPTx_CURCNT must always be read twice. If the two readings are different, this register must be read a third time to determine the correct value.

GPTx_STAT must be read prior to writing to any timer register after setting or clearing the enable bit. When GPTx_STAT, Bit 7 is cleared, registers can be modified, which ensures that the timer has completed synchronization between the processor and the timer clock domains. The typical synchronization time is two timer clock periods.

The GPTx_CTL register enables the counter, selects the mode, selects the prescale value, and controls the event capture function.

Capture Event Function

The general-purpose timers can capture several interrupt events. These events are shown in Table 341. Any one of the events associated with a general-purpose timer can cause a capture of the 16-bit GPTx_CURCNT register into the 16-bit GPTx_CAPTURE register. GPTx_CTL has a 5-bit field that can select which event to capture.

When the selected interrupt event occurs, the GPTx_CURCNT register is copied into the GPTx_CAPTURE register. When GPTx_STAT, Bit 1 is set, it indicates that a capture event is pending. The bit is cleared by writing 1 to GPTx_CLRINT, Bit 1. The GPTx_CAPTURE register also holds its value and cannot be overwritten until a 1 is written to GPTx_CLRINT, Bit 1.

Table 341. Capture Event Function

Event Select Range Bits, CON0, Bits[12:8]	Timer 0 Capture Source	Timer 1 Capture Source	Timer 2 Capture Source
0000	WUT	UART	SYS_WAKE
0001	SYS_WAKE	SPI0	Reserved
0010	Reserved	Reserved	Reserved
0011	Reserved	SPI1	Reserved

DIGITAL DIE GENERAL-PURPOSE TIMERS

Table 341. Capture Event Function (Continued)

Event Select Range Bits, CON0, Bits[12:8]	Timer 0 Capture Source	Timer 1 Capture Source	Timer 2 Capture Source
0100	Reserved	I^2C target	DMA error
0101	Reserved	I^2C initiator	WUT
0110	DVDD_REG	Reserved	General-Purpose Timer 0
0111	Reserved	Reserved	General-Purpose Timer 1
1000	Reserved	Reserved	Reserved
1001	GPIO Interrupt A	Reserved	Reserved
1010	GPIO Interrupt B	Reserved	Reserved
1011	General-Purpose Timer 1	Reserved	Reserved
1100	General-Purpose Timer 1	SYS_WAKE	Reserved
1101	Flash controller	Reserved	Reserved
1110	Reserved	General-Purpose Timer 0	Reserved
1111	Reserved	General-Purpose Timer 2	Reserved

General-Purpose Timers Power Gating

To limit power consumption, the general-purpose timers are powered off with the clock input disabled. To enable the clock input to each of the general-purpose timer blocks, clear the appropriate bit in the CTL5 register. This register must be cleared before writing to any of the timer control registers.

REGISTER SUMMARY: GENERAL-PURPOSE TIMERS

Table 342. Timer Register Summary

Address	Name	Description	Reset	Access
0x40000000	GPT0_LOAD	16-bit synchronous load value	0x0000	R/W
0x40000004	GPT0_CURCNT	16-bit timer synchronous value	0x0000	R
0x40000008	GPT0_CTL	Control	0x000A	R/W
0x4000000C	GPT0_CLRINT	Clear interrupt	0x0000	W
0x40000010	GPT0_CAPTURE	Capture	0x0000	R
0x40000014	GPT0_ALOAD	16-bit asynchronous load value	0x0000	R/W
0x40000018	GPT0_ACURCNT	16-bit timer asynchronous value	0x0000	R
0x4000001C	GPT0_STAT	Status	0x0000	R
0x40000400	GPT1_LOAD	16-bit synchronous load value	0x0000	R/W
0x40000404	GPT1_CURCNT	16-bit timer synchronous value	0x0000	R
0x40000408	GPT1_CTL	Control	0x000A	R/W
0x4000040C	GPT1_CLRINT	Clear interrupt	0x0000	W
0x40000410	GPT1_CAPTURE	Capture	0x0000	R
0x40000414	GPT1_ALOAD	16-bit asynchronous load value	0x0000	R/W
0x40000418	GPT1_ACURCNT	16-bit timer asynchronous value	0x0000	R
0x4000041C	GPT1_STAT	Status	0x0000	R
0x40000800	GPT2_LOAD	16-bit synchronous load value	0x0000	R/W
0x40000804	GPT2_CURCNT	16-bit timer synchronous value	0x0000	R
0x40000808	GPT2_CTL	Control	0x000A	R/W
0x4000080C	GPT2_CLRINT	Clear interrupt	0x0000	W
0x40000810	GPT2_CAPTURE	Capture	0x0000	R
0x40000814	GPT2_ALOAD	16-bit asynchronous load value	0x0000	R/W
0x40000818	GPT2_ACURCNT	16-bit timer asynchronous value	0x0000	R
0x4000081C	GPT2_STAT	Status	0x0000	R

REGISTER DETAILS: GENERAL-PURPOSE TIMERS

16-BIT SYNCHRONOUS LOAD VALUE REGISTERS

Address: 0x40000000, Reset: 0x0000, Name: GPT0_LOAD

Address: 0x40000400, Reset: 0x0000, Name: GPT1_LOAD

Address: 0x40000800, Reset: 0x0000, Name: GPT2_LOAD

Table 343. Bit Descriptions for GPT0_LOAD, GPT1_LOAD, GPT2_LOAD

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	LOAD		Load Value. The up or down counter is periodically loaded with this value if periodic mode is selected (GPTx_CTL, Bit 3 = 1). This bit writes during up or down counter timeout events that are delayed until the event passes.	0x0000	R/W

16-BIT TIMER SYNCHRONOUS VALUE REGISTERS

Address: 0x40000004, Reset: 0x0000, Name: GPT0_CURCNT

Address: 0x40000404, Reset: 0x0000, Name: GPT1_CURCNT

Address: 0x40000804, Reset: 0x0000, Name: GPT2_CURCNT

Table 344. Bit Descriptions for GPT0_CURCNT, GPT1_CURCNT, GPT2_CURCNT

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	VALUE		Current Count. Reflects the current up or down counter value. Value delayed by two PCLK cycles due to clock synchronizers.	0x0000	R

CONTROL REGISTERS

Address: 0x40000008, Reset: 0x000A, Name: GPT0_CTL

Address: 0x40000408, Reset: 0x000A, Name: GPT1_CTL

Address: 0x40000808, Reset: 0x000A, Name: GPT2_CTL

Table 345. Bit Descriptions for GPT0_CTL, GPT1_CTL, GPT2_CTL

Bits	Bit Name	Settings	Description	Reset	Access
15	SYNCBYP		Synchronization Bypass. Used to bypass the synchronization logic within the block. Use only when both the general-purpose timer and the CPU are clocked from the same source.	0x0	R/W
14	Reserved		Reserved.	0x0	R
13	EVTEN		Event Select. Used to enable and to disable the capture of events. This bit is used in conjunction with the event select range. When a selected event occurs, the current value of the up or down counter is captured in GPTx_CAPTURE.	0x0	R/W
		0	Events are not captured.		
		1	Events are captured.		
[12:8]	EVTRANGE		Event Select Range. Timer event select range (0 to 31).	0x0	R/W
7	RLD		Reload Control. This bit is only used for periodic mode. This bit allows the user to select whether the up or down counter is reset only on a timeout event or also when CLRINT, Bit 0 is set.	0x0	R/W
		0	Up or down counter is only reset on a timeout event.		
		1	Resets the up down counter when CLRINT, Bit 0 is set.		
[6:5]	CLK		Clock Select. Used to select a timer clock from the four available clock sources.	0x0	R/W
		00	PCLK.		
		01	High frequency oscillator. 26 MHz high frequency oscillator.		
		10	Low frequency oscillator. 32 kHz low frequency oscillator.		
		11	Reserved.		
4	EN		Timer Enable. Used to enable and to disable the timer. Clearing this bit resets the timer, including the GPTx_CURCNT register.	0x0	R/W
		0	Timer is disabled. Default.		
		1	Timer is enabled.		

REGISTER DETAILS: GENERAL-PURPOSE TIMERS

Table 345. Bit Descriptions for GPT0_CTL, GPT1_CTL, GPT2_CTL (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
3	MODE		Timer Mode. This bit controls whether the timer runs in periodic or free running mode. In periodic mode, the up or down counter starts at the defined load value (GPTx_LOAD). In free running mode, the up or down counter starts at 0x0000 or 0xFFFF depending on whether the timer is counting up or down.	0x1	R/W
		0	Timer runs in free running mode.		
		1	Timer runs in periodic mode. Default.		
2	UP		Count Up. Used to control whether the timer increments (counts up) or decrements (counts down) the up or down counter.	0x0	R/W
		0	Timer is set to count down. Default.		
		1	Timer is set to count up.		
[1:0]	PRE		Prescaler. Controls the prescaler division factor, applied to the selected clock of the timer. If Clock Source 0 (PCLK) or Clock Source 1 (HCLK) is selected, a prescaler value of 0 means divide by 4. Otherwise, it means divide by 1.	0x2	R/W
		00	Source clock/1 or source clock/4. Divide by 1 if SYNCBYP = 1, or divide by 4 if SYNCBYP = 0.		
		01	Source clock/16.		
		10	Source clock/64.		
		11	Source clock/256.		

CLEAR INTERRUPT REGISTERS

Address: 0x4000000C, Reset: 0x0000, Name: GPT0_CLRINT

Address: 0x4000040C, Reset: 0x0000, Name: GPT1_CLRINT

Address: 0x4000080C, Reset: 0x0000, Name: GPT2_CLRINT

Table 346. Bit Descriptions for GPT0_CLRINT, GPT1_CLRINT, GPT2_CLRINT

Bits	Bit Name	Settings	Description	Reset	Access
[15:2]	Reserved		Reserved.	0x000	R
1	EVTCAPT		Clear Captured Event Interrupt. This bit is used to clear a capture event interrupt.	0x0	W1C
		0	No effect.		
		1	Clear the capture event interrupt.		
0	TIMEOUT		Clear Timeout Interrupt. This bit is used to clear a timeout interrupt.	0x0	W1C
		0	No effect.		
		1	Clears the timeout interrupt.		

CAPTURE REGISTERS

Address: 0x40000010, Reset: 0x0000, Name: GPT0_CAPTURE

Address: 0x40000410, Reset: 0x0000, Name: GPT1_CAPTURE

Address: 0x40000810, Reset: 0x0000, Name: GPT2_CAPTURE

Table 347. Bit Descriptions for GPT0_CAPTURE, GPT1_CAPTURE, GPT2_CAPTURE

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	VALUE		16-Bit Captured Value. GPTx_CAPTURE holds its value until GPTx_CLRINT, Bit 1 is set by user code. GPTx_CAPTURE is not overwritten even if another event occurs without writing to GPTx_CLRINT, Bit 1.	0x0000	R

REGISTER DETAILS: GENERAL-PURPOSE TIMERS

16-BIT ASYNCHRONOUS LOAD VALUE REGISTERS

Address: 0x40000014, Reset: 0x0000, Name: GPT0_ALOAD

Address: 0x40000414, Reset: 0x0000, Name: GPT1_ALOAD

Address: 0x40000814, Reset: 0x0000, Name: GPT2_ALOAD

Table 348. Bit Descriptions for GPT0_ALOAD, GPT1_ALOAD, GPT2_ALOAD

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	VALUE		16-Bit Captured Value. GPTx_CAPTURE holds its value until GPTx_CLRINT, Bit 1 is set by user code. GPTx_CAPTURE is not overwritten even if another event occurs without writing to GPTx_CLRINT, Bit 1. Only use when a synchronous clock source is selected (GPTx_CTL, Bits[6:5] = 00 or if the high frequency oscillator is clocking both the timer and CPU directly).	0x0000	R

16-BIT TIMER ASYNCHRONOUS VALUE REGISTERS

Address: 0x40000018, Reset: 0x0000, Name: GPT0_ACURCNT

Address: 0x40000418, Reset: 0x0000, Name: GPT1_ACURCNT

Address: 0x40000818, Reset: 0x0000, Name: GPT2_ACURCNT

Table 349. Bit Descriptions for GPT0_ACURCNT, GPT1_ACURCNT, GPT2_ACURCNT

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	VALUE		Counter Value. Reflects the current up or down counter value. Reading GPTx_ACURCNT takes advantage of having the timer run on PCLK by bypassing clock synchronization logic that is otherwise required. Only use when a synchronous clock source is selected (GPTx_CTL, Bits[6:5] = 00 or if the high frequency oscillator is clocking both the timer and CPU directly).	0x0000	R

STATUS REGISTERS

Address: 0x4000001C, Reset: 0x0000, Name: GPT0_STAT

Address: 0x4000041C, Reset: 0x0000, Name: GPT1_STAT

Address: 0x4000081C, Reset: 0x0000, Name: GPT2_STAT

Table 350. Bit Descriptions for GPT0_STAT, GPT1_STAT, GPT2_STAT

Bits	Bit Name	Settings	Description	Reset	Access
[15:8]	Reserved		Reserved.	0x0	R
7	PDOK		GPTx_CLRINT Synchronization. This bit is set automatically when the user sets GPTx_CLRINT, Bit 0 = 1. This bit is cleared automatically when the clear interrupt request has crossed clock domains and taken effect in the timer clock domain.	0x0	R
		0	The interrupt is cleared in the timer clock domain.		
		1	GPTx_CLRINT, Bit 0 is being updated in the timer clock domain.		
6	BUSY		Timer Busy. This bit informs the user that a write to GPTx_CTL is still crossing into the timer clock domain. Check this bit after writing GPTx_CTL and suppress further writes until this bit is cleared.	0x0	R
		0	Timer ready to receive commands to GPTx_CTL.		
		1	Timer not ready to receive commands to GPTx_CTL.		
[5:2]	Reserved		Reserved.	0x0	R
1	CAPTURE		Capture Event Pending.	0x0	R
		0	No capture event is pending.		
		1	A capture event is pending.		
0	TIMEOUT		Timeout Event Occurred. This bit is set automatically when the value of the counter reaches zero while counting down or reaches full scale when counting up. This bit is cleared when GPTx_CLRINT, Bit 0 is set by the user.	0x0	R
		0	No timeout event has occurred.		
		1	A timeout event has occurred.		

ANALOG DIE GENERAL-PURPOSE TIMERS

ANALOG DIE GENERAL-PURPOSE TIMERS FEATURES

The ADuCM355 analog die integrates two identical general-purpose, 16-bit count up or count down timers: AGPT0 and AGPT1. The timers can be clocked from the 32 kHz internal low frequency oscillator on the analog die, the analog die PCLK, internal 16 MHz or 32 MHz high frequency oscillator, or the external clock input. Clock sources can be scaled down using a prescaler to 1, 4, 16, or 256. Free running mode and periodic modes are available. The timers have a PWM output feature. Configure the INTEN register to enable AFE general-purpose timer interrupts.

AFE PWM

The AFE die in the ADuCM355 has a dedicated PWM output feature. The high period for the PWM is the difference between the values in the timer load register and either the PWMMAT0 register or PWMMATCH register. The low period for the PWM is the difference between either the PWMMAT0 register or PWMMATCH register and the overflow value 0xFFFF. Increasing and reducing the load value increases and reduces the frequency of the PWM output.

The following example sets up the PWM for a duty cycle of 2 ms (high) period and 1.56 ms (low) period:

```
void PWM1Init(void)
{
   AfeDioCfgPin(pADI_AGPIO2, PIN1, 1);
   /*1MHz GPT clock, count up, periodic */
   AfeGptCfg(pADI_AGPT1, TCTL_CLK_PCLK, TCTL_PRE_DIV16, BITM_TMR_CTL_UP|BITM_TMR_CTL_MODE);
  /*count 2000, 2ms period:  64000 - 62000, assuming 16MHz clock and prescaler of 16 */
   AfeGptLd(pADI_AGPT1,62000);
   /*Low period is 65535-64000*/
   AfePwmCfg(pADI_AGPT1, PWM_IDLEHI, PWM_MATCH_MODE);
   AfePwmMatch(pADI_AGPT1, 64000);
   /*start timer, output PWM*/
   pADI_AGPT1->CON1 |= BITM_TMR_CTL_EN;
}
```

REGISTER SUMMARY: ANALOG DIE GENERAL-PURPOSE TIMERS

Table 351. AGPT0 Register Summary

Address	Name	Description	Reset	Access
0x400C0D00	LD0	16-bit load value	0x0000	R/W
0x400C0D04	VAL0	16-bit timer value	0x0000	R
0x400C0D08	CON0	Control	0x000A	R/W
0x400C0D0C	CLRI0	Clear interrupt	0x0000	W
0x400C0D14	ALD0	16-bit load value, asynchronous	0x0000	R/W
0x400C0D18	AVAL0	16-bit timer value, asynchronous	0x0000	R
0x400C0D1C	STA0	Status	0x0000	R
0x400C0D20	PWMCON0	PWM control	0x0000	R/W
0x400C0D24	PWMMAT0	PWM match value	0x0000	R/W
0x400C0D28	INTEN	Interrupt enable	0x0000	R/W

Table 352. AGPT1 Register Summary

Address	Name	Description	Reset	Access
0x400C0E00	LOAD	16-bit load value	0x0000	R/W
0x400C0E04	CURCNT	16-bit timer value	0x0000	R
0x400C0E08	CTL	Control	0x000A	R/W
0x400C0E0C	CLRINT	Clear interrupt	0x0000	W
0x400C0E14	ALOAD	16-bit load value, asynchronous	0x0000	R/W
0x400C0E18	ACURCNT	16-bit timer value, asynchronous	0x0000	R
0x400C0E1C	STAT	Status	0x0000	R
0x400C0E20	PWMCTL	PWM control	0x0000	R/W
0x400C0E24	PWMMATCH	PWM match value	0x0000	R/W

REGISTER DETAILS: ANALOG DIE GENERAL-PURPOSE TIMERS

16-BIT LOAD VALUE REGISTER

Address: 0x400C0D00, Reset: 0x0000, Name: LD0

Table 353. Bit Descriptions for LD0

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	LOAD		Load Value. The up or down counter is periodically loaded with this value if periodic mode is selected (CON0, Bit 3 = 1). LD0 writes during up or down counter timeout events are delayed until the event has passed.	0x0	R/W

16-BIT TIMER VALUE REGISTER

Address: 0x400C0D04, Reset: 0x0000, Name: VAL0

Table 354. Bit Descriptions for VAL0

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	VAL		Current Count. Reflects the current up or down counter value. Value delayed two PCLK cycles due to clock synchronizers.	0x0	R

CONTROL REGISTER

Address: 0x400C0D08, Reset: 0x000A, Name: CON0

Table 355. Bit Descriptions for CON0

Bits	Bit Name	Settings	Description	Reset	Access
15	SYNCBYP		Synchronization Bypass. Used to bypass the synchronization logic within the block. Use only with synchronous clocks. This bit field also changes the CON0, Bits[1:0] maximum prescaler count from 3 to 0.	0x0	R/W
14	RSTEN		Counter and Prescale Reset Enable. Used to enable and disable the reset feature. Used in conjunction with the EVTEN and EVENT select range. When a selected event occurs, the 16-bit counter and 8-bit prescale are reset. This reset is required in PWM demodulation mode.	0x0	R/W
13	EVTEN		Event Select. Used to enable and disable the capture of events. Used in conjunction with the event select range. When a selected event occurs, the current value of the up or down counter is captured in GPTx_CAPTURE.	0x0	R/W
		0	Events are not captured.		
		1	Events are captured.		
[12:8]	EVENT		Event Select Range. Timer event select range (0 to 31).	0x0	R/W
7	RLD		Reload Control. This bit is only used for periodic mode. This bit allows the user to select whether the up or down counter is reset only on a timeout event or when CLRI0, Bit 0 is set.	0x0	R/W
		1	Resets the up or down counter when the clear timeout interrupt bit is set.		
		0	Up or down counter is only reset on a timeout event.		
[6:5]	CLK		Clock Select. Used to select a timer clock from the four available clock sources.	0x0	R/W
		00	AFE PCLK.		
		01	AFE high-power oscillator.		
		10	AFE low frequency oscillator.		
		11	External clock.		
4	ENABLE		Timer Enable. Used to enable and disable the timer. Clearing this bit resets the timer, including the VAL0 register.	0x0	R/W
		0	Timer is disabled. Default.		
		1	Timer is enabled.		
3	MOD		Timer Mode. This bit is used to control whether the timer runs in periodic or free running mode. In periodic mode, the up or down counter starts at the defined load value. In free running mode, the up or down counter starts at 0x0000 or 0xFFFF, depending on whether the timer is counting up or down.	0x1	R/W
		1	Timer runs in periodic mode. Default.		
		0	Timer runs in free running mode.		
2	UP		Count Up. Used to control whether the timer increments (counts up) or decrements (counts down) the up or down counter.	0x0	R/W

REGISTER DETAILS: ANALOG DIE GENERAL-PURPOSE TIMERS

Table 355. Bit Descriptions for CON0 (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		1	Timer is set to count up.		
		0	Timer is set to count down. Default.		
[1:0]	PRE		Prescaler. Controls the prescaler division factor applied to the selected clock of the timer.	0x2	R/W
		00	Source clock/1 or source clock/4. When CON0, Bit 15 is set, source is source clock/1. When cleared, the source is source clock/4.		
		01	Source clock/16.		
		10	Source clock/64.		
		11	Source clock/256.		

CLEAR INTERRUPT REGISTER

Address: 0x400C0D0C, Reset: 0x0000, Name: CLRI0

Table 356. Bit Descriptions for CLRI0

Bits	Bit Name	Settings	Description	Reset	Access
[15:2]	Reserved		Reserved.	0x0	R
1	CAP		Clear Captured Event Interrupt. This bit is used to clear a capture event interrupt.	0x0	W1C
		1	Clear the capture event interrupt.		
		0	No effect.		
0	TMOUT		Clear Timeout Interrupt. This bit is used to clear a timeout interrupt.	0x0	W1C
		1	Clears the timeout interrupt.		
		0	No effect.		

16-BIT LOAD VALUE, ASYNCHRONOUS REGISTER

Address: 0x400C0D14, Reset: 0x0000, Name: ALD0

Only use when a synchronous clock source is selected (CON0, Bits[6:5] = 00).

Table 357. Bit Descriptions for ALD0

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	ALOAD		Load Value, Asynchronous. The up or down counter is periodically loaded with this value if periodic mode is selected (CON0, Bit 3 = 1). Writing ALOAD takes advantage of having the timer run on PCLK by bypassing clock synchronization logic that is otherwise required.	0x0	R/W

16-BIT TIMER VALUE, ASYNCHRONOUS REGISTER

Address: 0x400C0D18, Reset: 0x0000, Name: AVAL0

Only use when a synchronous clock source is selected (CON0, Bits[6:5] = 00).

Table 358. Bit Descriptions for AVAL0

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	AVAL		Counter Value. Reflects the current up or down counter value. Reading AVAL takes advantage of having the timer run on PCLK by bypassing clock synchronization logic that is otherwise required.	0x0	R

STATUS REGISTER

Address: 0x400C0D1C, Reset: 0x0000, Name: STA0

Table 359. Bit Descriptions for STA0

Bits	Bit Name	Settings	Description	Reset	Access
[15:9]	Reserved		Reserved.	0x0	R
8	RSTCNT		Counter Reset Occurring. Indicates that the counter is currently being reset due to an event detection. CON0, Bit 14 must be set.	0x0	R

REGISTER DETAILS: ANALOG DIE GENERAL-PURPOSE TIMERS

Table 359. Bit Descriptions for STA0 (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
7	PDOK		Clear Interrupt Register Synchronization. This bit is set automatically when the user sets CLRI0, Bit 0 = 1. It is cleared automatically when the clear interrupt request has crossed clock domains and taken effect in the timer clock domain.	0x0	R
		1	The interrupt bit is being updated in the timer clock domain.		
		0	The interrupt is cleared in the timer clock domain.		
6	BUSY		Timer Busy. This bit informs the user that a write to CON0 is still crossing into the timer clock domain. Check this bit after writing CON0 and suppress further writes until this bit is cleared.	0x0	R
		0	Timer ready to receive commands to control register.		
		1	Timer not ready to receive commands to control register.		
[5:2]	Reserved		Reserved.	0x0	R
1	CAP		Capture Event Pending. A capture of the current timer value has occurred.	0x0	R
		0	No capture event is pending.		
		1	A capture event is pending.		
0	TMOUT		Timeout Event Occurred. This bit is set automatically when the value of the counter reaches zero while counting down or reaches full scale when counting up. This bit is cleared when CLRI0, Bit 0 is set by the user.	0x0	R
		0	No timeout event has occurred.		
		1	A timeout event has occurred.		

PWM CONTROL REGISTER

Address: 0x400C0D20, Reset: 0x0000, Name: PWMCON0

Table 360. Bit Descriptions for PWMCON0

Bits	Bit Name	Settings	Description	Reset	Access
[15:2]	Reserved		Reserved.	0x0	R
1	IDLE		PWM Idle State. This bit is used to set the PWM idle state.	0x0	R/W
0	MATCHEN		PWM Match Enabled. This bit is used to control PWM operational mode.	0x0	R/W
		0	PWM in toggle mode.		
		1	PWM in match mode.		

PWM MATCH VALUE REGISTER

Address: 0x400C0D24, Reset: 0x0000, Name: PWMMAT0

Table 361. Bit Descriptions for PWMMAT0

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	MATCHVAL		PWM Match Value. The value is used when the PWM is operating in match mode. The PWM output is asserted when the up or down counter is equal to this match value. PWM output is deasserted again when a timeout event occurs. If the match value is never reached, or occurs simultaneous to a timeout event, the PWM output remains idle.	0x0	R/W

INTERRUPT ENABLE REGISTER

Address: 0x400C0D28, Reset: 0x0000, Name: INTEN

Table 362. Bit Descriptions for INTEN

Bits	Bit Name	Settings	Description	Reset	Access
[15:1]	Reserved		Reserved.	0x0	R
0	INTEN		Interrupt Enable. This value is used when the PWM is operating in match mode. The PWM output is asserted when the up or down counter is equal to this match value. PWM output is deasserted again when a timeout event occurs. If the match value is never reached, or occurs simultaneous to a timeout event, the PWM output remains idle.	0x0	R/W

REGISTER DETAILS: ANALOG DIE GENERAL-PURPOSE TIMERS

16-BIT LOAD VALUE REGISTER

Address: 0x400C0E00, Reset: 0x0000, Name: LOAD

Table 363. Bit Descriptions for LOAD

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	VALUE		Load Value. The up or down counter is periodically loaded with this value if periodic mode is selected (CTL, Bit 3 = 1). Writes from this bit during up or down counter timeout events are delayed until the event has passed.	0x0	R/W

16-BIT TIMER VALUE REGISTER

Address: 0x400C0E04, Reset: 0x0000, Name: CURCNT

Table 364. Bit Descriptions for CURCNT

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	VALUE		Current Count. Reflects the current up or down counter value. Value delayed two PCLK cycles due to clock synchronizers.	0x0	R

CONTROL REGISTER

Address: 0x400C0E08, Reset: 0x000A, Name: CTL

Table 365. Bit Descriptions for CTL

Bits	Bit Name	Settings	Description	Reset	Access
15	SYNCBYP		Synchronization Bypass. Used to bypass the synchronization logic within the block. Use only with synchronous clocks. This bit field also changes the PRE bit maximum prescaler count from 3 to 0.	0x0	R/W
14	RSTEN		Counter and Prescale Reset Enable. Used to enable and disable the reset feature. Used in conjunction with CTL, Bit 13 (EVTEN) and CTL, Bits[12:8] (EVTRANGE). When a selected event occurs, the 16-bit counter and 8-bit prescale are reset. This reset is required in PWM demodulation mode.	0x0	R/W
13	EVTEN		Event Select. Used to enable and disable the capture of events. Used in conjunction with the CTL, Bits[12:8] (EVTRANGE) bits. When a selected event occurs, the current value of the up or down counter is captured in GPTx_CAPTURE.	0x0	R/W
		0	Events are not captured.		
		1	Events are captured.		
[12:8]	EVTRANGE		Event Select Range. Timer event select range (0 to 31).	0x0	R/W
7	RLD		Reload Control. This bit is only used for periodic mode. This bit allows the user to select whether the up or down counter is reset only on a timeout event or also when CLRINT, Bit 0 is set.	0x0	R/W
		1	Resets the up or down counter when the clear timeout interrupt bit is set.		
		0	Up or down counter is only reset on a timeout event.		
[6:5]	CLK		Clock Select. Used to select a timer clock from the four available clock sources.	0x0	R/W
		00	AFE PCLK.		
		01	AFE high-power oscillator.		
		10	AFE low frequency oscillator.		
		11	External clock.		
4	EN		Timer Enable. Used to enable and disable the timer. Clearing this bit resets the timer, including the CURCNT register.	0x0	R/W
		0	Timer is disabled. Default.		
		1	Timer is enabled.		
3	MODE		Timer Mode. This bit is used to control whether the timer runs in periodic or free running mode. In periodic mode, the up or down counter starts at the defined ALOAD, Bits[15:0]. In free running mode, the up or down counter starts at 0x0000 or 0xFFFF, depending on whether the timer is counting up or down.	0x1	R/W
		1	Timer runs in periodic mode. Default.		
		0	Timer runs in free running mode.		
2	UP		Count Up. Used to control whether the timer increments (counts up) or decrements (counts down) the up or down counter.	0x0	R/W

REGISTER DETAILS: ANALOG DIE GENERAL-PURPOSE TIMERS

Table 365. Bit Descriptions for CTL (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		1	Timer is set to count up.		
		0	Timer is set to count down. Default.		
[1:0]	PRE		Prescaler. Controls the prescaler division factor applied to the selected clock of the timer.	0x2	R/W
		00	Source clock/1 or source clock/4. When CTL, Bit 15 is set, source clock/1. When cleared, source clock/4.		
		01	Source clock/16.		
		10	Source clock/64.		
		11	Source clock/256.		

CLEAR INTERRUPT REGISTER

Address: 0x400C0E0C, Reset: 0x0000, Name: CLRINT

Table 366. Bit Descriptions for CLRINT

Bits	Bit Name	Settings	Description	Reset	Access
[15:2]	Reserved		Reserved.	0x0	R
1	EVTCAPT		Clear Captured Event Interrupt. This bit is used to clear a capture event interrupt.	0x0	W1C
		1	Clear the capture event interrupt.		
		0	No effect.		
0	TIMEOUT		Clear Timeout Interrupt. This bit is used to clear a timeout interrupt.	0x0	W1C
		1	Clears the timeout interrupt.		
		0	No effect.		

16-BIT LOAD VALUE, ASYNCHRONOUS REGISTER

Address: 0x400C0E14, Reset: 0x0000, Name: ALOAD

Only use when a synchronous clock source is selected (CTL, Bits[6:5] = 00).

Table 367. Bit Descriptions for ALOAD

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	VALUE		Load Value, Asynchronous. The up or down counter is periodically loaded with this value if periodic mode is selected (CTL, Bit 5 = 1). Writing this register takes advantage of having the timer run on PCLK by bypassing clock synchronization logic that is otherwise required.	0x0	R/W

16-BIT TIMER VALUE, ASYNCHRONOUS REGISTER

Address: 0x400C0E18, Reset: 0x0000, Name: ACURCNT

Only use when a synchronous clock source is selected (CTL, Bits[6:5] = 00).

Table 368. Bit Descriptions for ACURCNT

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	VALUE		Counter Value. Reflects the current up or down counter value. Reading this register takes advantage of having the timer run on PCLK by bypassing clock synchronization logic that is otherwise required.	0x0	R

STATUS REGISTER

Address: 0x400C0E1C, Reset: 0x0000, Name: STAT

Table 369. Bit Descriptions for STAT

Bits	Bit Name	Settings	Description	Reset	Access
[15:9]	Reserved		Reserved.	0x0	R
8	CNTRST		Counter Reset Occurring. Indicates that the counter is currently being reset due to an event detection. CTL, Bit 14 must be set.	0x0	R

REGISTER DETAILS: ANALOG DIE GENERAL-PURPOSE TIMERS

Table 369. Bit Descriptions for STAT (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
7	PDOK		Clear Interrupt Register Synchronization. This bit is set automatically when the user sets CLRINT, Bit 0 = 1. This bit is cleared automatically when the clear interrupt request has crossed clock domains and taken effect in the timer clock domain.	0x0	R
		1	The clear timeout interrupt bit is being updated in the timer clock domain.		
		0	The interrupt is cleared in the timer clock domain.		
6	BUSY		Timer Busy. This bit informs the user that a write to CTL is still crossing into the timer clock domain. Check this bit after writing CTL and suppress further writes until this bit is cleared.	0x0	R
		0	Timer ready to receive commands to control register.		
		1	Timer not ready to receive commands to control register.		
[5:1]	Reserved		Reserved.	0x0	R
0	TIMEOUT		Timeout Event Occurred. This bit is set automatically when the value of the counter reaches zero while counting down or reaches full scale when counting up. This bit is cleared when CLRINT, Bit 0 is set by the user.	0x0	R
		0	No timeout event has occurred.		
		1	A timeout event has occurred.		

PWM CONTROL REGISTER

Address: 0x400C0E20, Reset: 0x0000, Name: PWMCTL

Table 370. Bit Descriptions for PWMCTL

Bits	Bit Name	Settings	Description	Reset	Access
[15:2]	Reserved		Reserved.	0x0	R
1	IDLESTATE		PWM Idle State. Used to set the PWM idle state.	0x0	R/W
		0	PWM idles low.		
		1	PWM idles high.		
0	MATCH		PWM Match Enabled. Used to control PWM operational mode.	0x0	R/W
		0	PWM in toggle mode.		
		1	PWM in match mode.		

PWM MATCH VALUE REGISTER

Address: 0x400C0E24, Reset: 0x0000, Name: PWMMATCH

Table 371. Bit Descriptions for PWMMATCH

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	VALUE		PWM Match Value. The value is used when the PWM is operating in match mode. The PWM output is asserted when the up or down counter is equal to this match value. PWM output is deasserted again when a timeout event occurs. If the match value is never reached, or occurs simultaneous to a timeout event, the PWM output remains idle.	0x0	R/W

AFE WATCHDOG TIMER

WATCHDOG TIMER FEATURES AND BLOCK DIAGRAM

To satisfy the IEC 61508 standard requirement for separating the watchdog timer from all processor clock sources, the analog die watchdog timer is used to recover from an invalid software state. After the watchdog timer is enabled by user code, it requires periodic servicing to prevent it from forcing a reset of the processor. The required maximum watchdog timer reset period is 16 sec in gas sensor applications.

The watchdog timer is a windowed timer with an upper and lower refresh period. The watchdog timer triggers a reset if the following are true:

- The timer is not refreshed before the largest interval. The largest interval is set by the user via the WDTLD register.
- The timer is refreshed before the shortest interval. The shortest interval is set by the user via the WDTMINLD register.
- The timer is refreshed by writing 0xCCCC to the WDTCLRI register.
- A watchdog timer reset is forced by writing a value other than 0xCCCC to the WDTCLRI register.

The watchdog timer is clocked by the low frequency oscillator (see the Clock Divider Configuration Register section). It is clocked at all times, except during reset, while in debug mode, and when it is selectively disabled while in hibernate mode.

The watchdog timer is a 16-bit countdown timer with a programmable prescaler. The prescaler source is selectable and can be scaled by a factor of 1, 16, 256, or 4096. The analog die watchdog timer control registers ensure a maximum timeout period of 16383.5 sec when WDTCON, Bit 8 = 1. WDTCON, Bit 8 = 1 results in the 32.768 kHz clock being divided by 2. When WDTCON, Bit 8 = 0, the maximum timeout is 8191.75 sec. The default timeout period is 32 sec.

WATCHDOG TIMER OPERATION

After any reset, the watchdog timer is initialized with an initial configuration. This initial configuration can be modified by user code. However, setting the watchdog timer enable register write protects the watchdog timer configuration or enable register so that the watchdog timer keeps running. Only a reset clears the write protection and allows reconfiguration of the timer. When the watchdog timer is not enabled, the watchdog timer can be reconfigured at any time.

When the watchdog timer decrements to 0, a reset is generated. This reset can be prevented by writing 0xCCCC to the WDTCLRI register. Writing 0xCCCC to WDTCLRI causes the watchdog timer to reload with the watchdog timer configuration in free running mode. In this case, the watchdog timer immediately begins a new timeout period and starts to count again.

The reset output of the watchdog timer works solely from the 32 kHz clock and does not require the system clock to be active. Therefore, the reset output can work with all the power-down modes, including hibernate mode.

In MCU software debugging mode, disable the WDT first before software debugging. These specific values written to specific registers eliminate the possibility of rogue code refreshing the watchdog timer.

WINDOWED WATCHDOG FEATURE

The windowed watchdog feature provides extra robustness to the watchdog timer for safety critical applications. With the window feature enabled, the watchdog timer reset fires if user code refreshes the watchdog timer either too quickly or too slowly. When the window feature is enabled (WDTCON, Bit 9 = 1), the watchdog timer must be refreshed before the counter value reaches 0. The WDT must also be refreshed after the counter has passed the value written to the WDTMINLD register.

The following are example instructions to set up the windowed watchdog feature:

```
pADI_AFEWDT->WDTLD = 0x800;  //16 second timeout period
pADI_AFEWDT->WDTMINLD = 0x600;       // Min window is 4s after start
pADI_AFEWDT->WDTCON = 0x248;          // Enable Windowed feature
```

AFE WATCHDOG TIMER

INTERRUPT MODE

If a watchdog reset occurs while debugging via the SWD port, communications between the debugger and the ADuCM355 can be lost. As such, while debugging, the user can optionally configure the watchdog timer to generate an interrupt instead of a reset. Enable this feature only during code development and debug. Enable the watchdog timer to generate a reset in full user applications. Setting WDTCON, Bit 0 to 1 or 0 has no effect in interrupt mode. The following are example instructions to set up the watchdog timer in interrupt mode:

```
pADI_AFEWDT->WDTLD = 0x200;  //4second timeout period
pADI_AFEWDT->WDTCON = 0x44A;   // WDT IRQ, Window On, periodic, Clock div256,
NVIC_EnableIRQ(AFE_Watchdog_IRQn);  // Enable the NVIC interrupt for the AFE Watchdog timer
```

REGISTER SUMMARY: AFE WATCHDOG TIMER

Table 372. AFEWDT Register Summary

Address	Name	Description	Reset	Access
0x400C0900	WDTLD	Watchdog timer load value	0x1000	R/W
0x400C0904	WDTVALS	Current count value	0x1000	R
0x400C0908	WDTCON	Watchdog timer control	0x00C9	R/W
0x400C090C	WDTCLRI	Refresh watchdog	0x0000	W
0x400C0918	WDTSTA	Timer status	0x0000	R
0x400C091C	WDTMINLD	Minimum load value	0x0800	R/W

REGISTER DETAILS: AFE WATCHDOG TIMER

WATCHDOG TIMER LOAD VALUE REGISTER

Address: 0x400C0900, Reset: 0x1000, Name: WDTLD

Table 373. Bit Descriptions for WDTLD

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	LOAD		Watchdog Timer Load Value. This user programmable value is the value that the counter starts from before counting down to 0.	0x1000	R/W

CURRENT COUNT VALUE REGISTER

Address: 0x400C0904, Reset: 0x1000, Name: WDTVALS

When read, returns the value of the counter.

Table 374. Bit Descriptions for WDTVALS

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	CCOUNT		Current Watchdog Timer Count Value. Read only register.	0x1000	R

WATCHDOG TIMER CONTROL REGISTER

Address: 0x400C0908, Reset: 0x00C9, Name: WDTCON

Table 375. Bit Descriptions for WDTCON

Bits	Bit Name	Settings	Description	Reset	Access
[15:11]	Reserved		Reserved.	0x0	R/W
10	WDTIRQEN		Watchdog Timer Interrupt Enable.	0x0	R/W
		0	Disable. Cleared by user to generate a reset when the counter times out or if a refresh occurs within WDTMINLD.		
		1	Enable. Debug feature. An interrupt occurs instead of a reset if the counter times out or if a refresh occurs within WDTMINLD.		
9	MINLOAD_EN		Timer Window Control. When enabled, if the user refreshes the timer before the counter reaches the value in the WDTMINLD register, a reset or IRQ occurs.	0x1	R/W
		0	Disable. Disable window feature. Watchdog is refreshed if user code writes to WDTCLRI before the counter reaches 0.		
		1	Enable. Enable window feature. Watchdog is only refreshed if user code writes to WDTCLRI before the counter reaches 0, but after the counter has passed the minimum load value set in WDT.		
8	CLKDIV2		Clock Source.	0x0	R/W
		0	Analog die 32.768 kHz oscillator.		
		1	Analog die 32.768 kHz oscillator divided by 2.		
7	Reserved		Reserved.	0x1	R
6	MDE		Timer Mode Select.	0x1	R/W
		0	Free running mode. In free running mode, timer wraps around at 0x1000.		
		1	Periodic mode. Default. In this mode, the counter counts from the WDTLD value down to 0.		
5	EN		Timer Enable.	0x1	R/W
		0	Cleared by user to disable timer.		
		1	Set by user to enable timer. Default.		
4	Reserved		Reserved.	0x0	R
[3:2]	PRE		Prescaler.	0x2	R/W
		00	Source clock/1.		
		01	Source clock/16.		
		10	Source clock/256. Default.		
		11	Source clock/4096.		
1	IRQ		WDT Interrupt Enable.	0x0	R/W
		0	Watchdog timer timeout creates a reset.		
		1	Watchdog timer timeout creates an interrupt instead of a reset.		

REGISTER DETAILS: AFE WATCHDOG TIMER

Table 375. Bit Descriptions for WDTCON (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
0	PDSTOP		Power Down Stop Enable.	0x1	R/W
		0	Continue counting when in hibernate mode. The watchdog timer continues its count down while in hibernate mode.		
		1	Stop counter when in hibernate mode. When hibernate mode is entered, the watchdog counter suspends its countdown. As hibernate mode is exited, the countdown resumes from its current count value (the count is not reset).		

REFRESH WATCHDOG REGISTER

Address: 0x400C090C, Reset: 0x0000, Name: WDTCLRI

Table 376. Bit Descriptions for WDTCLRI

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	CLRWDG		Refresh Register. User writes 0xCCCC to reset, reload, or restart the counter or clear the IRQ. A write of any other value causes a watchdog reset or IRQ. Write only, reads 0.	0x0	W

TIMER STATUS REGISTER

Address: 0x400C0918, Reset: 0x0000, Name: WDTSTA

Table 377. Bit Descriptions for WDTSTA

Bits	Bit Name	Settings	Description	Reset	Access
[15:7]	Reserved		Reserved.	0x0	R/W
6	TMINLD		WDTMINLD Register Write Status.	0x0	R
		0	The WDTMINLD value is fully updated. Arm and AFE watchdog clock domains and the WDTMINLD values match.		
		1	WDTMINLD value update in progress. Arm peripheral bus and the WDTMINLD value are being synchronized to the 32 kHz clock domain.		
5	Reserved		Reserved.	0x0	R
4	LOCK		Lock Status.	0x0	R
		0	Timer operation not locked.		
		1	Timer enabled and locked. Set automatically in hardware when WDTCON, Bit 5 has been set by user code.		
3	CON		WDTCON Write Status.	0x0	R
		0	WDTCON is up to date. Arm peripheral bus and watchdog timer clock domains match WDTCON values.		
		1	WDTCON register synchronizing. The Arm peripheral bus and watchdog timer clock domains are in the process of synchronizing with the WDTCON values.		
2	TLD		WDTVALS Write Status.	0x0	R
		0	Arm and AFE watchdog clock WDTLD register values match.		
		1	Synchronize in progress. Arm peripheral bus WDTLD value is being synchronized to the 32 kHz clock domain.		
1	CLRI		WDTCLRI Write Status.	0x0	R
		0	Arm peripheral bus WDTCLRI write synchronization not complete.		
		1	Arm peripheral bus WDTCLRI write is being synchronized to WDT clock domain. WDT restarts (if 0xCCCC was written) when latter synchronization is complete.		
0	IRQ		WDT Interrupt. Set to 1 when watchdog timer interrupt occurs.	0x0	R
		0	Watchdog timer interrupt not pending.		
		1	Watchdog timer interrupt pending. Cleared by writing to WDTCLRI register.		

REGISTER DETAILS: AFE WATCHDOG TIMER

MINIMUM LOAD VALUE REGISTER

Address: 0x400C091C, Reset: 0x0800, Name: WDTMINLD

Watchdog timer minimum timeout period. Lower window limit.

Table 378. Bit Descriptions for WDTMINLD

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	MIN_LOAD		WDT Minimum Load Value. If software writes to WDTCLRI before the counter reaches the MIN_LOAD value, a WDT reset or IRQ occurs.	0x800	R/W

DIGITAL DIE WAKE-UP TIMER

OVERVIEW

The digital die WUT is the highest priority interrupt on the ADuCM355, as described in Table 41. The WUT is also one of four interrupt sources that can wake the digital die from hibernate mode. The WUT is clocked by the low frequency oscillator. The accuracy specifications of the low frequency oscillator deem this timer unsuitable for a real-time clock setup, although some register names indicate support for a true real-time clock. Because the WUT is clocked by the 32 kHz oscillator, take care when performing CPU reads or writes to the wake-up timer registers. Synchronization status bits are provided to indicate when a read or write access is fully complete in the 32 kHz domain.

FEATURES

Key digital die WUT features include the following:

- 32 kHz input clock can be further divided by a prescale factor of 2^0 to 2^{15}.
- The timer count register is nominally a 32-bit value configured by the 16-bit registers, CNT1 and CNT0, for total Bits[31:0]. This value can be expanded to 47 bits if CNT2 (the fractional divide register) is used, enabling CNT1, CNT0, and CNT2 for Bits[46:0]. When initializing or reenabling the WUT count, or when changing the prescale division ratio, the prescaler is automatically zeroed so that the WUT count value is positioned on exact, coincident boundaries of both the start of the prescale sequence and the modulo 60 count roll over.
- The WUT block can generate interrupts from multiple sources, unmasked by programming the CR0 register. The source of the interrupt is reflected in the SR0 register. The timer interrupt sources include two optionally enabled, independent alarm features (one at absolute time and the other at modulo 60 periodic time) that cause a processor interrupt when the timer count equals the alarm values.
- The WUT can take and preserve a snapshot of its elapsed real time count when prompted to do so by the CPU, allowing the CPU to associate a time stamp with an incoming data packet. The WUT preserves the snapshot for readback by the CPU. The snapshot is persistent and is only overwritten when the CPU issues a request to capture a new value.

REGULAR AND PERIODIC MODULO 60 INTERRUPTS

To enable periodic interrupts, the modulo 60 feature of the timer can be used. The modulo block divides the timer counter by 60 if the remainder modulus equals the value in CR0, Bits[10:5] or a modulo alarm or interrupt occurs.

To enable periodic interrupts, perform the following steps:

1. Select the timer prescale value by writing to CR1, Bits[8:5]. This setting configures the base clock frequency for the timer.
2. Poll the WSYNCCR0 bit and wait for it to be set in the SR0 register as the MMR write occurs in the slower RTC domain.
3. Select the number of counts after 0 for the interrupt to occur. Write a value between 0 and 59 to CR0, Bits[10:5].
4. Enable the MOD60ALMINTEN interrupt source. Set CR0, Bit 11 = 1.
5. Enable modulo 60 alarms. Set CR0, Bit 4 = 1.
6. Set the global enable bit for the timer by setting CR0, Bit 0 = 1 to start the timer.

TIMER MATCHING ALARM VALUE INTERRUPTS

The WUT generates an interrupt when the timer counter value matches the value set by the user in the alarm registers, CNT1 and CNT0, for Bits[31:0], or the alarm registers, CNT1, CNT0, and CNT2, for Bits[46:0]. To program this interrupt, perform the following steps:

1. Reset the CNTx registers to 0.
2. Configure the prescaler to divide the WUT base clock in the CR1 register.
3. Poll the WSYNCCR0 bit and wait for the bit to be set in the SR0 register as the MMR write occurs in the slower WUT domain.
4. Program the ALM0 register, ALM1 register, RTC register, and ALM2 register with the intended alarm time.
5. Enable the interrupt for alarm by setting CR0, Bit 2.
6. Set the ALMEN and CNTEN bits in the CR0 register.
7. Wait for the WUT alarm interrupt, which is triggered when the CNTx register value matches the ALMx register value.

WUT FUNCTIONAL DESCRIPTION

A high level block diagram of the WUT is shown in Figure 65. All functionality for counting, alarm, snapshot, and wake-up interrupts is located in a dedicated 32 kHz, timed, always on WUT power domain. The APB interface with the CPU (which comprises queuing and dispatch logic for posted register writes) and interrupts to the Cortex NVIC are located in a PCLK and FCLK timed section of the main power gated core domain.

DIGITAL DIE WAKE-UP TIMER

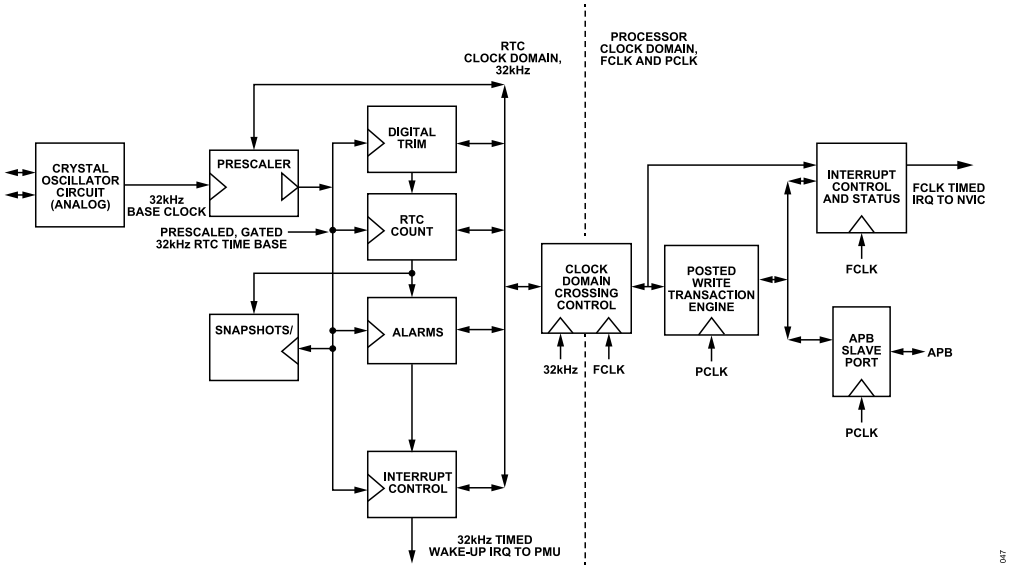

Figure 65. Digital Die WUT Block Diagram

WUT OPERATING MODES

Initial WUT Power-Up

The WUT operates in a dedicated voltage domain that is continuously powered under normal conditions. However, when a battery is attached for the first time or replaced, a POR occurs. This POR resets all WUT registers. Upon detecting a WUT failure, the CPU reprograms the count registers of the timers and clears the fail flag in the timer control register. The CPU can optionally program the alarm registers of the timer to generate an interrupt when the alarm and count values match.

Persistent Sticky Wake-Up Events

When the device is in a power-down mode, there are no losses of timer alarm events. If the alarm is enabled, the resulting interrupt is asserted by the WUT so that the NVIC subsequently sees an RCLK timed version of the interrupt when power is restored to the processor. To facilitate this interrupt, the timer sends a 32 kHz timed version of the same interrupt to the wake-up controller in the PMU, which causes the digital core to be repowered. When the CPU is woken up, it can inspect both the PMU and WUT to understand the cause of the interrupt event for the wakeup.

WUT Capacity to Accommodate Posted Writes by CPU

If a posted write by the CPU to a 32 kHz sourced MMR in the WUT is pending dispatch in the WUT clock domain due to a queue of other similar register writes to the 32 kHz domain, a second or subsequent write by the CPU to the same WUT register cannot be stacked up to overwrite the pending transaction. Any such attempts are rejected by the WUT. These result in SR0, Bit 4 interrupt events in the WUT (see the Status 0 Register section).

Snapshot of the Timer Counter

The CPU can instruct the timer to take a snapshot of its elapsed time count by writing a software key of 0x7627 to the GWY register. This causes the combined three snapshot registers (SNAP0, SNAP1, and SNAP2) to update to the current value of the three count registers (CNT0, CNT1, and CNT2) and to maintain this snapshot until subsequently told by the CPU to overwrite it.

WUT RECOMMENDATIONS: CLOCK AND POWER

Stopping the PCLK

Before entering any mode that causes the PCLK to stop, the CPU must first wait until there is confirmation from the WUT that no previously posted writes have yet to complete. The CPU can check this by reading both the SR0 and SR2 registers.

DIGITAL DIE WAKE-UP TIMER

Ensuring No Communication Across WUT Power Boundary When Powering Down

When the CPU has advance knowledge about a power-down, it must either check to confirm that there are no posted writes in the WUT awaiting execution or cancel all queued and executing posted writes in the WUT. Cancellation is achieved by writing a cancellation key of 0xA2C5 to the GWY register, which takes immediate effect. These actions maintain the integrity of the always on half of the WUT.

Do not post any further register writes to the WUT until power is lost by the core to ensure that no communication between the CPU and the WUT occurs. Such communication makes the WUT liable to corruption when the WUT power domains isolation barrier is subsequently activated.

REGISTER SUMMARY: DIGITAL DIE WAKE-UP TIMER

Table 379. WUT Register Summary

Address	Name	Description	Reset	Access
0x40001400	CR0	Control 0	0x03C4	R/W
0x40001404	SR0	Status 0	0x7F88	R/W
0x40001408	SR1	Status 1	0x0078	R
0x4000140C	CNT0	Count 0	0x0000	R/W
0x40001410	CNT1	Count 1	0x0000	R/W
0x40001414	ALM0	Alarm 0	0xFFFF	R/W
0x40001418	ALM1	Alarm 1	0xFFFF	R/W
0x40001420	GWY	Gateway	0x0000	W
0x40001428	CR1	Control 1	0x01E0	R/W
0x4000142C	SR2	Status 2	0xC000	R/W
0x40001430	SNAP0	Snapshot 0	0x0000	R
0x40001434	SNAP1	Snapshot 1	0x0000	R
0x40001438	SNAP2	Snapshot 2	0x0000	R
0x4000143C	MOD	Modulo	0x0040	R
0x40001440	CNT2	Count 2	0x0000	R
0x40001444	ALM2	Alarm 2	0x0000	R/W
0x40001488	SR6	Status 6	0x7900	R

REGISTER DETAILS: DIGITAL DIE WAKE-UP TIMER

CONTROL 0 REGISTER

Address: 0x40001400, Reset: 0x03C4, Name: CR0

CR0 is the primary of two control registers for the WUT, the other being CR1. All mainstream WUT operations are enabled and disabled by the CPU using CR0. The granularity of the WUT control is expanded by the CR1 register.

Table 380. Bit Descriptions for CR0

Bits	Bit Name	Settings	Description	Reset	Access
15	WPNDINTEN		Enable WPENDINT Sourced Interrupts to the CPU. This field is an enable for WUT interrupts to the CPU, based on the WPENDINT sticky interrupt in the SR0 register.	0x0	R/W
		0	Disable WPENDINT sourced interrupts to the CPU.		
		1	Enable WPENDINT sourced interrupts to the CPU.		
14	WSYNCINTEN		Enable WSYNCINT Sourced Interrupts to the CPU. WSYNCINTEN is an enable for WUT interrupts to the CPU based on the WSYNCINT sticky interrupt source field of the SR0 MMR. WSYNCINTEN is activated whenever the effects of a posted write to a 32 kHz sourced MMR or MMR bit field become visible to the CPU.	0x0	R/W
		0	Disable WSYNCINT sourced interrupts to the CPU.		
		1	Enable WSYNCINT sourced interrupts to the CPU.		
13	WPNDERRINTEN		Enable WPNDERRINT Sourced Interrupts to the CPU when a WUT Register Write Pending Error Occurs. Write pending errors can be avoided by the CPU by checking the pending status of a register in SR1 before undertaking a write to that register. If a WPNDERRINT error occurs, and this bit is set to 1, the WUT interrupts the CPU.	0x0	R/W
		0	Disable interrupts if write pending errors occur in the WUT.		
		1	Enable interrupts for write pending errors in the WUT.		
12	ISOINTEN		Enable ISOINT Sourced Interrupts to the CPU. This bit enables interrupts to the CPU based on the ISOINT sticky interrupt source in the SR0 status register. When power loss is imminent to all power domains on the device apart from the WUT, the WUT activates its isolation barrier so that the WUT can continue to operate independently of the core. When power is subsequently restored to the rest of the device, the WUT activates the ISOINT interrupt source to act as a sticky record of the power loss event just finishing. This activation occurs as the WUT lowers its isolation barrier when the core regains power. If enabled by this bit, the WUT interrupts the CPU based on ISOINT. The CPU can then inspect the ISOINT field of SR0 to determine if the CPU has recovered from a total loss of power.	0x0	R/W
		0	Disable ISOINT sourced interrupts to the CPU.		
		1	Enable ISOINT sourced interrupts to the CPU.		
11	MOD60ALMINTEN		Enable Periodic MOD60ALMINT Sourced Interrupts to the CPU. This bit allows the CPU to enable a periodic, repeating interrupt from the timer at a displacement time in WUT time units given by the MOD60ALM bit beyond a Modulo 60 boundary.	0x0	R/W
		0	Disable periodic interrupts due to Modulo 60 WUT elapsed time.		
		1	Enable periodic interrupts due to Modulo 60 WUT elapsed time.		
[10:5]	MOD60ALM		Periodic Modulo 60 Alarm Time in Prescaled WUT Time Units Beyond a Modulo 60 Boundary. This bit allows the CPU to position a periodic alarm interrupt from the WUT at any integer number of prescaled WUT time units from a Modulo 60 boundary (roll over event) of the value in CNT1 and CNT0. Values of 0 to 59 are allowed for MOD60ALM. If a greater value is configured, this value is treated as zero prescaled WUT time units. Boundaries are defined when the CPU writes a new pair of values to the CNT1 and CNT0 registers, the CPU enables the WUT from a disabled state using CR0, Bit 0, or when the WUT is enabled by CR0, Bit 0. For example, a value of 30 results in the Modulo 60 periodic interrupt from the WUT to be issued to the CPU at 30 time units past a Modulo 60 boundary.	0x1E	R/W
4	MOD60ALMEN		Enable WUT Modulo 60 Counting of Time Past a Modulo 60 Boundary. Enables the detection of the counter passing a value of 60, whereas MOD60ALMINTEN enables the generation of a resultant interrupt.	0x0	R/W
		0	Disable determination of Modulo 60 WUT elapsed time.		
		1	Enable determination of Modulo 60 WUT elapsed time.		
3	Reserved		Reserved. Clear to 0.	0x0	R/W

REGISTER DETAILS: DIGITAL DIE WAKE-UP TIMER

Table 380. Bit Descriptions for CR0 (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
2	ALMINTEN		Enable ALMINT Sourced Alarm Interrupts to the CPU. ALMINTEN gives the CPU extra control over whether an alarm event (alarm count matches the WUT count) triggers an interrupt. Under normal conditions, ALMINTEN is set active, most notably when the detection of an alarm condition is enabled by ALMEN. If alarm interrupts enabled but the alarm itself is disabled, no interrupts occur.	0x1	R/W
		0	Disable alarm interrupts.		
		1	Enable an interrupt if the WUT alarm and count values match.		
1	ALMEN		Enable the WUT Alarm Absolute Operation. This bit must be set active for the alarm logic to function and for any alarm event to be detected. Such an event is defined as a match between the values of the WUT count and alarm registers, namely CNT1, CNT0, CNT2, ALM1, ALM0, and ALM2. Count and alarm values and match conditions are defined on a 47-bit basis, although the constituent registers are individually 16 bits wide. When enabled by ALMEN, the detection of an alarm event is held in the sticky interrupt source bit field, ALMINT, of the status register, SR0.	0x0	R/W
		0	Disable detection of alarm events.		
		1	Enable detection of alarm events.		
0	CNTEN		Global Enable for the WUT. CNTEN enables counting of elapsed real time and acts as an initiator enable for the WUT. If the WUT is enabled by activating CNTEN, this event causes a realignment of the prescaler and the Modulo 60 counter used by the WUT to generate MOD60ALMINT sourced interrupts in SR0.	0x0	R/W
		0	Disable the WUT.		
		1	Enable the WUT.		

STATUS 0 REGISTER

Address: 0x40001404, Reset: 0x7F88, Name: SR0

Information on WUT operation is made available to the CPU via three status registers: SR0, SR1, and SR2. These registers include all flags related to CPU interrupt sources and error conditions within the WUT.

Table 381. Bit Descriptions for SR0

Bits	Bit Name	Settings	Description	Reset	Access
15	Reserved		Reserved.	0x0	R
14	ISOENB		Visibility Status of 32 kHz Sourced Registers, Taking Account of Power Domain Isolation. This bit indicates whether 32 kHz sourced MMRs in the always on half of the WUT are visible to the CPU. During normal powered operation, ISOENB is high, confirming that all registers are visible. If ISOENB is low when the device is emerging from hibernation, 32 kHz sourced information is not yet available to the CPU.	0x1	R
		0	32 kHz sourced MMRs in the always on half of the WUT are not yet visible to the CPU due to isolation.		
		1	32 kHz sourced MMRs in the always on half of the WUT are visible to the CPU.		
13	Reserved		Reserved.	0x1	R
12	WSYNCALM1		Synchronization Status of Posted Writes to the ALM1 Register. WSYNCALM1 indicates if the effects of a posted write to ALM1 are visible to the CPU.	0x1	R
		0	Results of a posted write are not yet visible to the CPU.		
		1	Results of a posted write are visible to the CPU.		
11	WSYNCALM0		Synchronization Status of Posted Writes to the ALM0 Register. WSYNCALM0 indicates if the effects of a posted write to ALM0 are visible to the CPU.	0x1	R
		0	Results of a posted write are not yet visible to the CPU.		
		1	Results of a posted write are visible to the CPU.		
10	WSYNCCNT1		Synchronization Status of Posted Writes to the CNT1 Register. WSYNCCNT1 indicates if the effects of a posted write to CNT1 are visible to the CPU.	0x1	R
		0	Results of a posted write are not yet visible to the CPU.		
		1	Results of a posted write are visible to the CPU.		

REGISTER DETAILS: DIGITAL DIE WAKE-UP TIMER

Table 381. Bit Descriptions for SR0 (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
9	WSYNCCNT0		Synchronization Status of Posted Writes to the CNT0 Register. WSYNCCNT0 indicates if the effects of a posted write to CNT0 are visible to the CPU.	0x1	R
		0	Results of a posted write are not yet visible to the CPU.		
		1	Results of a posted write are visible to the CPU.		
8	WSYNCSR0		Synchronization Status of Posted Clearances to Interrupt Sources in the SR0 Register. WSYNCSR0 indicates if the effects of a posted write to SR0 are visible to the CPU. No posting or associated masking is needed for clearances of the WPNDERRINT, WSYNCINT, and WPENDINT bits in SR0, because these fields are sourced in the clock domain of the core. Their clearance is immediate.	0x1	R
		0	Results of a posted write are not yet visible to the CPU.		
		1	Results of a posted write are visible to the CPU.		
7	WSYNCCR0		Synchronization Status of Posted Writes to the CR0 Register. WSYNCCR0 indicates if the effects of a posted write to CR0 are visible to the CPU.	0x1	R
		0	Results of a posted write are not yet visible by the CPU.		
		1	Results of a posted write are visible by the CPU.		
6	WPENDINT		Write Pending Interrupt. WPENDINT is a sticky interrupt source that is activated whenever there is room in the CPU to post a new write transaction to a 32 kHz sourced MMR or MMR bit field in the RTC. To enable a WPENDINT interrupt, set WPENDINTEN to 1 in the CR0 register. Cleared by writing a value of one to this bit.	0x0	R/W1C
		0	There has been no change in the pending status of any posted write transaction in the WUT since WPENDINT was last cleared.		
		1	A posted write transaction has been dispatched since WPENDINT was last cleared, freeing up a slot for a new posted write by the CPU to the same MMR.		
5	WSYNCINT		Write Synchronization Interrupt. WSYNCINT is a sticky interrupt source that is activated whenever a posted write transaction to a 32 kHz sourced MMR or MMR bit field completes and whose effects are then visible to the CPU. By checking the synchronization status of posted write bits of SR0 (Bits[12:7]) and SR2 (Bits[15:14]), the CPU can identify which posted write transaction has just completed and caused the WSYNCINT interrupt source to stick (or restick if already active). Cleared by writing 1 to this bit.	0x0	R/W1C
		0	Since the CPU last cleared WSYNCINT, there has been no occurrence of the effects of a posted write transaction to a 32 kHz sourced MMR or MMR bit field.		
		1	Since the CPU last cleared WSYNCINT, the effects of a posted write transaction to a 32 kHz sourced MMR or MMR bit field has become newly visible to the clock domain of the CPU.		
4	WPNDERRINT		Write Pending Error Interrupt Source. This bit is a sticky interrupt source that indicates that an error has occurred because the CPU attempted to write to a WUT register while a previous write to the same register was pending execution. If multiple write pending errors occur, WPNDERRINT sticks at the first occurrence. Cleared by writing 1 to this bit.	0x0	R/W1C
		0	No posted write has been rejected by the WUT since WPNDERRINT was last cleared by the CPU.		
		1	A posted write has been rejected by the WUT due to a previously posted write to the same MMR, which is still awaiting execution. Such a rejection has occurred since the CPU last cleared WPNDERRINT.		
3	ISOINT		WUT Power Domain Isolation Interrupt Source. ISOINT is a sticky interrupt source that indicates whether the WUT has had to activate its power domain isolation barrier due to a power loss in the core. When the core regains power, the CPU can read ISOINT to inform itself of such a power event. Cleared by writing a value of one to this bit.	0x1	R/W1C
		0	The always on WUT power domain has not activated its isolation from the core since the ISOINT interrupt source was last cleared by the CPU.		
		1	The always on WUT power domain has activated and subsequently deactivated its isolation from the core due to a power event. This event occurred since ISOINT was last cleared by the CPU. Clear bit by writing a 1.		
2	MOD60ALMINT		Modulo 60 WUT Alarm Interrupt Source. MOD60ALMINT is a sticky flag that is the source of an optionally enabled interrupt to the CPU. This interrupt is activated once every 60 increments of CNT1 and CNT0, at a displacement of MOD60ALM increments past a modulo 60 boundary. Cleared by writing 1 to this bit.	0x0	R/W1C

REGISTER DETAILS: DIGITAL DIE WAKE-UP TIMER

Table 381. Bit Descriptions for SR0 (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		0	A MOD60ALMINT interrupt event has not occurred since this bit was last cleared by the CPU.		
		1	A MOD60ALMINT interrupt event has occurred since this bit was last cleared by the CPU.		
1	ALMINT		Alarm Interrupt Source. ALMINT is a sticky flag that is the source of an optionally enabled interrupt to the CPU. ALMINT indicates that an alarm event has occurred due to a match between the WUT count and alarm register values. A match is defined as the value in CNT1, CNT0, and CNT2, equating to the alarm time given by ALM1, ALM0, and ALM2. The detection of such an event is enabled by ALMEN in CR0. ALMINT is cleared by writing 1 to this bit.	0x0	R/W1C
		0	An ALMINT interrupt event has not occurred since this bit was last cleared by the CPU.		
		1	An ALMINT interrupt event has occurred since this bit was last cleared by the CPU.		
0	Reserved		Reserved.	0x0	R

STATUS 1 REGISTER

Address: 0x40001408, Reset: 0x0078, Name: SR1

Table 382. Bit Descriptions for SR1

Bits	Bit Name	Settings	Description	Reset	Access
[15:13]	Reserved		Reserved.	0x0	R
12	WPNDALM1		Pending Status of Posted Writes to the ALM1 Register. WPNDALM1 indicates if a posted register write to ALM1 is currently pending.	0x0	R
		0	The WUT can accept a new posted write to the ALM1 MMR.		
		1	A previously posted write to ALM1 is still awaiting execution. No new posting to this MMR can be accepted.		
11	WPNDALM0		Pending Status of Posted Writes to the ALM0 Register. WPNDALM0 indicates if a posted register write to ALM0 is currently pending.	0x0	R
		0	The WUT can accept a new posted write to the ALM0 MMR.		
		1	A previously posted write to ALM0 is still awaiting execution. No new posting to this MMR can be accepted.		
10	WPNDCNT1		Pending Status of Posted Writes to the CNT1 Register. WPNDCNT1 indicates if a posted register write to CNT1 is currently pending.	0x0	R
		0	The WUT can accept a new posted write to the CNT1 MMR.		
		1	A previously posted write to CNT1 is still awaiting execution. No new posting to this MMR can be accepted.		
9	WPNDCNT0		Pending Status of Posted Writes to the CNT0 Register. WPNDCNT0 indicates if a posted register write to CNT0 is currently pending.	0x0	R
		0	The WUT can accept a new posted write to the CNT0 MMR.		
		1	A previously posted write to CNT0 is still awaiting execution. No new posting to this MMR can be accepted.		
8	WPNDSR0		Pending Status of Posted Clearances of Interrupt Sources in the SR0 Register. WPNDSR0 indicates if posted clearances of interrupt sources in SR0 are currently pending. Clearances can always be accepted and accumulated into the same pending transaction for SR0. As such, accumulation is nondestructive of previously posted clearances. This is in contrast to other MMRs, where posted write data (rather than posted clearances) cannot be aggregated into one transaction.	0x0	R
		0	The WUT can accept new posted clearances of interrupt sources in SR0 located in the 32 kHz domain.		
		1	A previously posted clearance of interrupt sources in SR0 and maintained in the 32 kHz domain is still awaiting execution. Additional clearances can still be aggregated into the existing, pending transaction.		
7	WPNDCR0		Pending Status of Posted Writes to CR0.	0x0	R
		0	The WUT can accept a new posted write to CR0.		
		1	A previously posted write to CR0 is still awaiting execution. No new posting to this MMR can be accepted.		
[6:0]	Reserved		Reserved.	0x78	R

REGISTER DETAILS: DIGITAL DIE WAKE-UP TIMER

COUNT 0 REGISTER

Address: 0x4000140C, Reset: 0x0000, Name: CNT0

CNT0 contains the lower 16 bits of the WUT counter, which maintains a real-time count in elapsed prescaled WUT time units. The instantaneous value of CNT0 can be read back by the CPU. The CPU can also redefine the value in this register. In this case, the WUT continues its real-time count from the redefined value.

Any write to CNT0 pends until a corresponding write to CNT1 is carried out by the CPU, so that the combined 32-bit count redefinition can be executed as a single transaction. CNT0 and CNT1 can be written in any order, but paired twin writes must be carried out by the CPU to have any effect on the WUT count. A paired write to CNT0 and CNT1 in any order zeroes the prescaler in the WUT and thus causes a redefinition of elapsed time by the CPU to align exactly with newly created modulo 1 and modulo 60 boundaries. Such a redefinition also causes the WUT to create a trim boundary and initiate a new trim interval. When the WUT count is redefined by the CPU, no coincident trim adjustment of the count is carried out.

The WUT supports immediate redefinition of CNT0 and CNT1 while CR0, Bit 0 is active. Alternatively, the CPU can disable the WUT by first making CR0, Bit 0 inactive while redefining these registers.

Table 383. Bit Descriptions for CNT0

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	VALUE		Lower 16 Prescaled Nonfractional Bits of the WUT Real-Time Count.	0x0	R/W

COUNT 1 REGISTER

Address: 0x40001410, Reset: 0x0000, Name: CNT1

CNT1 contains the upper 16 bits of the WUT counter, which maintains a real-time count in elapsed prescaled WUT time units. Any write to CNT1 pends until a corresponding write to CNT0 is carried out by the CPU, so that the combined 32-bit count redefinition can be executed as a single transaction.

Table 384. Bit Descriptions for CNT1

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	VALUE		Upper 16 Prescaled Nonfractional Bits of the WUT Real-Time Count.	0x0	R/W

ALARM 0 REGISTER

Address: 0x40001414, Reset: 0xFFFF, Name: ALM0

ALM0 contains the lower 16 bits of the prescaled nonfractional WUT alarm target time value, where the overall alarm is defined as ALM1, ALM0, and ALM2.

Any write to ALM0 pends until corresponding writes to ALM1 and ALM2 are carried out by the CPU, so that the combined 47-bit alarm redefinition can be executed as a single transaction. ALM0, ALM1, and ALM2 can be written in any order, but coordinated, triple writes must be carried out by the CPU to have any effect on the WUT alarm. ALM0 can be written to regardless of whether the ALMEN or CNTEN bits are active in the CR0 register.

Table 385. Bit Descriptions for ALM0

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	VALUE		Lower 16 Prescaled Nonfractional Bits of the WUT Alarm Target Time. The alarm register has a different reset value to the WUT count to avoid spurious alarms.	0xFFFF	R/W

ALARM 1 REGISTER

Address: 0x40001418, Reset: 0xFFFF, Name: ALM1

ALM1 contains the upper 16 bits of the prescaled nonfractional WUT alarm target time value, where the overall alarm is defined as ALM1, ALM0, and ALM2.

Any write to ALM1 pends until corresponding writes to ALM0 and ALM2 are carried out by the CPU, so that the combined 47-bit alarm redefinition can be executed as a single transaction. ALM0, ALM1, and ALM2 can be written in any order, but coordinated, triple writes must

REGISTER DETAILS: DIGITAL DIE WAKE-UP TIMER

be carried out by the CPU to have any effect on the WUT alarm. ALM1 can be written to regardless of whether the ALMEN or CNTEN bits are active in the CR0 register.

Table 386. Bit Descriptions for ALM1

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	VALUE		Upper 16 Prescaled Nonfractional Bits of the WUT Alarm Target Time. The alarm register has a different reset value to the WUT count to avoid spurious alarms.	0xFFFF	R/W

GATEWAY REGISTER

Address: 0x40001420, Reset: 0x0000, Name: GWY

GWY is a gateway MMR address through which the CPU can order actions to be taken within the WUT. The CPU does this by writing specific keys to GWY. GWY reads back as all zeros. The WUT supports the following independent, software keyed commands:

- Cancel all posted and executing write transactions in the WUT with immediate effect.
- Capture a sticky snapshot of the CNT1, CNT0, and CNT2 MMRs into SNAP1, SNAP0, and SNAP2.

Table 387. Bit Descriptions for GWY

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	SWKEY		Software Keyed Command Issued by the CPU. This register is the write target for activating software keyed commands issued by the CPU to the WUT. Supported keys are FLUSH_RTC and SNAPSHOT_RTC. FLUSH_RTC is a software key of Value 0xA2C5 delivered via a register write to GWY. FLUSH_RTC causes the WUT to flush all posted write transactions and to immediately stop any transaction that it is currently executing. The CPU only uses this key when power loss to the core is imminent and the CPU wants to cleanly and quickly terminate communication activity across the power domain boundary before the always on half of the WUT activates its isolation barrier. SNAPSHOT_RTC is a key of Value 0x7627 delivered via a register write to GWY. SNAPSHOT_RTC causes the WUT to take a sticky snapshot of the value of CNT1, CNT0, and CNT2 and store it in SNAP1, SNAP0, and SNAP2. Snapshots are sticky until a new one is requested.	0x0	W

CONTROL 1 REGISTER

Address: 0x40001428, Reset: 0x01E0, Name: CR1

This register is a secondary control register that expands the level of WUT control, which is provided by CR0.

CR1 allows additional sticky interrupt sources in SR2 to be enabled. These sources fan into the WUT peripheral interrupt to optionally tell the CPU when the WUT count has changed, when a prescaled (modulo 1) gated clock event has occurred to advance the WUT count, when a trim boundary has occurred, when the 32-bit WUT count in CNT1 or CNT0 has rolled over, and when the modulo 60 version of the WUT count has rolled over.

Table 388. Bit Descriptions for CR1

Bits	Bit Name	Settings	Description	Reset	Access
[15:9]	Reserved		Reserved.	0x0	R
[8:5]	PRESCALE2EXP		Prescale Power of Two Division Factor for the WUT Base Clock. PRESCALE2EXP defines the power of two by which the WUT base clock (nominally 32 kHz) is prescaled before being used to count time by advancing the contents of the CNT1 and CNT0 MMRs. In contrast, RTC1 can prescale the clock by any power of two in the interval Bits[15:0] to use as its time base.	0xF	R/W
		0000	Prescale the WUT base clock by 2^0 = 1.		
		0001	Prescale the WUT base clock by 2^1 = 2.		
		0010	Prescale the WUT base clock by 2^2 = 4.		
		0011	Prescale the WUT base clock by 2^3 = 8.		
		0100	Prescale the WUT base clock by 2^4 = 16.		
		0101	Prescale the WUT base clock by 2^5 = 32.		
		0110	Prescale the WUT base clock by 2^6 = 64.		
		0111	Prescale the WUT base clock by 2^7 = 128.		

REGISTER DETAILS: DIGITAL DIE WAKE-UP TIMER

Table 388. Bit Descriptions for CR1 (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		1000	Prescale the WUT base clock by 2^8 = 256.		
		1001	Prescale the WUT base clock by 2^9 = 512.		
		1010	Prescale the WUT base clock by 2^{10} = 1024.		
		1011	Prescale the WUT base clock by 2^{11} = 2048.		
		1100	Prescale the WUT base clock by 2^{12} = 4096.		
		1101	Prescale the WUT base clock by 2^{13} = 8192.		
		1110	Prescale the WUT base clock by 2^{14} = 16384.		
		1111	Prescale the WUT base clock by 2^{15} = 32768.		
4	CNTMOD60ROLLINTEN		Enable for the WUT Modulo 60 Count Roll Over Interrupt Source, CNTMOD60ROLLINT in SR2.	0x0	R/W
		0	Disable CNTMOD60ROLLINT interrupt.		
		1	Enable CNTMOD60ROLLINT interrupt.		
3	CNTROLLINTEN		Enable CNTROLLINT in SR2 for the WUT Count Roll Over Interrupt Source.	0x0	R/W
		0	Disable CNTROLLINT as an indicator of the WUT peripheral interrupt.		
		1	Enable CNTROLLINT as an indicator of the WUT peripheral interrupt.		
2	Reserved		Reserved.	0x0	R/W
1	PSINTEN		Enable for the Prescaled Modulo 1 Interrupt Source PSINT in SR2.	0x0	R/W
		0	Disable PSINT in SR2 as an indicator of the WUT peripheral interrupt.		
		1	Enable PSINT in SR2 as an indicator of the WUT peripheral interrupt.		
0	CNTINTEN		Enable for the WUT Count Interrupt Source, CNTINT in SR2.	0x0	R/W
		0	Disable CNTINT in SR2 as an indicator of the WUT peripheral interrupt.		
		1	Enable CNTINT in SR2 as an indicator of the WUT peripheral interrupt.		

STATUS 2 REGISTER

Address: 0x4000142C, Reset: 0xC000, Name: SR2

SR2 is a status register that further complements the status information provided by SR0 and SR1.

SR2 contains sticky interrupt sources optionally enabled via the CR1 register. These interrupt sources alert the CPU when the WUT count has changed, when the prescaled gated clock that controls the advancement of the WUT count has activated, when a trim boundary has occurred, when the 32-bit WUT count has rolled over, and when the modulo 60 version of the WUT count has rolled over.

All interrupt sources in SR2 are sticky, active high level signals. Each one can individually be cleared by writing a value of 0b1 to its bit position in SR2.

Table 389. Bit Descriptions for SR2

Bits	Bit Name	Settings	Description	Reset	Access
15	WSYNCALM2MIR		Synchronization Status of Posted Writes to ALM2. This bit indicates if the effects of a posted write to ALM2 are visible to the CPU.	0x1	R
		0	Results of a posted write to CR1 are not yet visible to the CPU.		
		1	Results of a posted write to CR1 are visible to the CPU.		
14	WSYNCCR1MIR		Synchronization Status of Posted Writes to CR1. This bit indicates if the effects of a posted write to CR1 are visible to the CPU. If this bit is low, a posted write to CR1 is currently queued up or in the process of being executed, but the results of this transaction are not yet visible to the CPU. When this bit goes high and thereby activates the SR0, Bit 5 sticky interrupt source, the effects of a write to CR1 are then available to the processor.	0x1	R
		0	Results of a posted write to CR1 are not yet visible to the CPU.		
		1	Results of a posted write to CR1 are visible to the CPU.		
13	WPNDALM2MIR		Pending Status of Posted Writes to ALM2. This bit indicates if a posted register write to ALM2 is currently pending and awaiting execution, meaning that no further write to the same MMR can be accepted at this time.	0x0	R
		0	Results of a posted write to CR1 are not yet visible by the CPU.		

REGISTER DETAILS: DIGITAL DIE WAKE-UP TIMER

Table 389. Bit Descriptions for SR2 (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		1	Results of a posted write to CR1 are visible by the CPU.		
12	WPNDCR1MIR		Pending Status of Posted Writes to CR1. This bit indicates if a posted register write to CR1 is currently pending and awaiting execution, meaning that no further write to the same MMR can be accepted at this time.	0x0	R
		0	The WUT can accept a new posted write to CR1.		
		1	A previously posted write to CR1 is still awaiting execution. No new posting to this MMR can be accepted.		
[11:7]	Reserved		Reserved.	0x0	R
6	CNTMOD60ROLL		WUT Count Modulo 60 Roll Over. When the CPU reacts to an interrupt due to the CR1, Bit 4 interrupt source, or if the CPU sees CR1, Bit 4 stickily set, the CPU can use this bit to confirm if the value of CNTMOD60 in the MOD register is still reflective of the roll over or has moved on since the actual occurrence. The latter is the case if the CPU was tardy in its response to a WUT interrupt caused by CR1, Bit 4 or if CR1, Bit 4 was not enabled as a contributory interrupt source for the RTC.	0x0	R
		0	The modulo 60 value of the WUT real-time count in CNTMOD60 has not risen due to a roll over.		
		1	The modulo 60 value of the WUT real-time count currently in CNTMOD60 has rolled over from a value at or within trimming distance of its maximum to a value at or within trimming distance of its minimum.		
5	CNTROLL		WUT Count Roll Over. CNTROLL indicates whether the current value of the WUT real-time count given by CNT1, CNT0, and CNT2 is due to a rollover from its maximum possible value to its minimum possible value when incremented at the most recent prescaled time unit. When the CPU reacts to an interrupt due to the CNTROLLINT interrupt source or sees that CNTROLLINT is stickily set, the CPU can use this bit to confirm if the values in CNT1, CNT0, and CNT2 are still reflective of the roll over or have moved on since the actual occurrence. The latter is the case if the CPU was tardy in its response to a WUT interrupt caused by CNTROLLINT or if CNTROLLINT was not enabled as a contributory interrupt source for the RTC.	0x0	R
		0	The WUT real-time count in CNT1, CNT0, and CNT2 has not risen due to a roll over.		
		1	The WUT real-time count currently in CNT1, CNT0, and CNT2 has rolled over from a value at or within trimming distance of its maximum to a value at or within trimming distance of its minimum.		
4	CNTMOD60ROLLINT		WUT Modulo 60 Count Roll Over Interrupt Source. This bit sticks active high when the modulo 60 equivalent of the CNT1 and CNT0 count value rolls over from 59 to zero. Such a roll over event happens every 60 prescaled increments of the WUT count, or fewer if positive trimming is active.	0x0	R/W1C
		0	The modulo 60 value of CNT1 and CNT0 in CNTMOD60 has not rolled over since this bit was last cleared.		
		1	The modulo 60 value of CNT1 and CNT0 in CNTMOD60 has rolled over since this bit was last cleared.		
3	CNTROLLINT		WUT Count Roll Over Interrupt Source. This bit sticks active high when the CNT1 and CNT0 count value rolls over from $2^{32} - 1$ to zero or is trimmed such that the trim increment causes the WUT to pass through (potentially spanning) these maximum and minimum values.	0x0	R/W1C
		0	CNT1 and CNT0 have not rolled over since this bit was last cleared.		
		1	CNT1 and CNT0 have rolled over since this bit was last cleared.		
2	Reserved		Reserved.	0x0	R/W1C
1	PSINT		WUT Prescaled Modulo 1 Boundary Interrupt Source. This bit sticks active high whenever the gated clock that defines the prescaled WUT time unit and the advancement of the WUT count is activated. For PSINT to cause an interrupt from the RTC, the corresponding enable bit for the interrupt fan in the PSINTEN bit in CR1 must be active high. This interrupt source is cleared by writing 1. Full interrupt capability is available for RTC1.	0x0	R/W1C
		0	The prescaled gated clock for the WUT count in CNT1, CNT0, and CNT2 has not activated since this bit was last cleared.		

REGISTER DETAILS: DIGITAL DIE WAKE-UP TIMER

Table 389. Bit Descriptions for SR2 (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
		1	The prescaled gated clock for the WUT count in CNT1, CNT0, and CNT2 has activated since this bit was last cleared.		
0	CNTINT		WUT Count Interrupt Source. CNTINT sticks active high whenever the value of CNT1 or CN0 changes. Such an event is not the same as the occurrence of a prescaled WUT time unit (as denoted by PSINT), because the WUT count can either be redefined or trimmed, which may or may not lead to value changes. This interrupt source is cleared by writing one to its bit position in SR2.	0x0	R/W1C
		0	The value of CNT1 and CNT0 has not changed since this bit was last cleared.		
		1	The value of CNT1 and CNT0 has changed since this bit was last cleared.		

SNAPSHOT 0 REGISTER

Address: 0x40001430, Reset: 0x0000, Name: SNAP0

SNAP0 is a sticky snapshot of the value of CNT0. It is updated at the same time as its counterparts, SNAP1 and SNAP2, thereby overwriting any previous values of SNAP1, SNAP0, and SNAP2. This updating and overwriting occurs when the CPU writes a snapshot request key of 0x7627 to GWY.

Table 390. Bit Descriptions for SNAP0

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	VALUE		Contains a Sticky Snapshot of CNT0. This channel takes a sticky snapshot of the 47-bit WUT count in CNT1, CNT0, and CNT2 and stores it in SNAP1, SNAP0, and SNAP2, respectively.	0x0	R

SNAPSHOT 1 REGISTER

Address: 0x40001434, Reset: 0x0000, Name: SNAP1

SNAP1 is a sticky snapshot of the value of CNT1. It is updated at the same time as its counterparts, SNAP0 and SNAP2, thereby overwriting any previous value of SNAP1, SNAP0, and SNAP2. This updating and overwriting occurs when the CPU writes a snapshot request key of 0x7627 to the GWY register.

Table 391. Bit Descriptions for SNAP1

Bits	Bit Name	Settings	Description	Reset	Access
[15:0]	VALUE		Contains a Sticky Snapshot of CNT1. This channel takes a sticky snapshot of the 47-bit WUT count in CNT1, CNT0, and CNT2 and stores it in SNAP1, SNAP0, and SNAP2, respectively.	0x0	R

SNAPSHOT 2 REGISTER

Address: 0x40001438, Reset: 0x0000, Name: SNAP2

SNAP2 is a sticky snapshot of the value of CNT2. It is updated as the same time as its counterparts, SNAP0 and SNAP1, thereby overwriting any previous values of SNAP1, SNAP0, and SNAP2 whenever the CPU writes a snapshot request key of 0x7627 to the GWY register.

Table 392. Bit Descriptions for SNAP2

Bits	Bit Name	Settings	Description	Reset	Access
15	Reserved		Reserved.	0x0	R
[14:0]	VALUE		Contains a Sticky Snapshot of CNT2. This channel takes a sticky snapshot of the 47-bit WUT count in CNT1, CNT0, and CNT2 and stores it in SNAP1, SNAP0, and SNAP2, respectively.	0x0	R

MODULO REGISTER

Address: 0x4000143C, Reset: 0x0040, Name: MOD

MOD is a read only register that makes available CNTMOD60, the modulo 60 equivalent of the CNT1 and CNT0 count values. This modulo 60 value is equal to the displacement in prescaled WUT time units past the most recent modulo 60 roll over event. A roll over is a synonym for a modulo 60 boundary.

The WUT realigns itself to create coincident modulo 60 and modulo 1 boundaries whenever either of the following events occurs:

REGISTER DETAILS: DIGITAL DIE WAKE-UP TIMER

- CPU writes a new pair of values to the CNT1 and CNT0 registers to redefine the elapsed time units count while the WUT is enabled and this posted twin write is subsequently executed.
- CPU enables the WUT from a disabled state using the CR0, Bit 0.

Other read only fields accessible via this register are the magnitude of the most recent increment to the WUT count (INCR) and confirmation as to whether this increment coincided with a trim boundary. The same increment is applied by the WUT to both its absolute (32-bit) count and the modulo 60 equivalent.

To facilitate debug and to clarify the relationship between CNTMOD60 and CNTx, the upper bits of MOD are padded with the LSBs of CNT0, allowing CNTMOD60 and part of the main WUT count to be read out at the same time and their alignment with each other to be understood.

Table 393. Bit Descriptions for MOD

Bits	Bit Name	Settings	Description	Reset	Access
[15:11]	CNT0_4TOZERO		Mirror of CNT0, Bits[4:0]. These bits are a mirror of CNT0, Bits[4:0], made available for simultaneous readback along with the CNTMOD60 bit field of MOD. Having this mirror available at the same time allows the relationship between the modulo 60 and absolute versions of the WUT count to be better understood and debugged.	0x0	R
10	Reserved		Reserved.	0x0	R
[9:6]	INCR		Most Recent Increment Value Added to the WUT Count in CNT1 and CNT0. INCR is the read only value by which the WUT count has most recently been incremented. Under normal circumstances, when the WUT is enabled, this value is one.	0x1	R
[5:0]	CNTMOD60		Modulo 60 Value of the WUT Count. CNTMOD60 is the modulo 60 value of the prescaled WUT count in CNT1 and CNT0. CNTMOD60 counts from 0 to 59 and then rolls over to 0 again. It advances and is trimmed in tandem with the main WUT count in CNT1, CNT0, and CNT2. CNTMOD60 is zeroed during a normal roll over from a value of 59 when advancing at a prescaled time unit, when the CPU writes a new pair of values to the CNT1 and CNT0 registers to redefine the elapsed time count while the WUT is enabled and this posted twin write is executed, when the CPU enables the WUT from a disabled state using the CNTEN bit of CR0, or while the WUT is enabled via CNTEN. The degree of prescaling in the WUT is changed by PRESCALE2EXP in CR1.	0x0	R

COUNT 2 REGISTER

Address: 0x40001440, Reset: 0x0000, Name: CNT2

CNT2 contains the fractional part of the WUT count, where the count is denominated in prescaled time units and is given by CNT1, CNT0, and CNT2. The overall resolution of the real-time count, including the fractional bits in CNT2, is one 32 kHz clock period.

CNT2 makes available to the CPU a read only view of the internal sequence count in the WUT prescaler as it steps its way (in units of one 32 kHz period) from all zeros to all ones across the number of prescale sequence bits given by the PRESCALE2EXP field of CR1. By completely traversing such a sequence, the prescaler effectively advances the main nonfractional part of the WUT count in CNT1 and CNT0 by one prescaled time unit. The prescale sequence count thus equates to the fractional part of the main WUT count.

Table 394. Bit Descriptions for CNT2

Bits	Bit Name	Settings	Description	Reset	Access
15	Reserved		Reserved.	0x0	R
[14:0]	VALUE		Fractional Bits of the WUT Real-Time Count. CNT2 contains the fractional part of the WUT count, where the count is denominated in prescaled time units and is given by CNT1, CNT0, and CNT2. The overall resolution of the real-time count, including the fractional bits in CNT2, is therefore one 32 kHz clock period. CNT2 is zeroed during the CPU writes a new pair of values to the CNT1 and CNT0 registers to redefine the elapsed time units count while the WUT is enabled and this posted twin write is executed, when the CPU enables the WUT from a disabled state using the CNTEN bit of CR0, or when the WUT is enabled. The degree of prescaling in the WUT is changed by PRESCALE2EXP in CR1.	0x0	R

REGISTER DETAILS: DIGITAL DIE WAKE-UP TIMER

ALARM 2 REGISTER

Address: 0x40001444, Reset: 0x0000, Name: ALM2

ALM2 specifies the fractional nonprescaled bits of the WUT alarm target time value, down to an individual 32 kHz clock cycle, where the overall alarm is defined as ALM1, ALM0, and ALM2.

The number of valid bits that can legitimately have values of one optionally written to them in ALM2 equals the number of prescale bits specified by the PRESCALE2EXP bit field of the CR1 MMR. If a bit position in ALM2 is set to one, such that this is incompatible with PRESCALE2EXP, the whole value of ALM2 is treated by the WUT as if it were zero.

Any write to ALM2 pends until corresponding writes to ALM0 and ALM1 are carried out by the CPU, so that the combined 47-bit alarm redefinition can be executed as a single transaction. ALM0, ALM1, and ALM2 can be written in any order, but coordinated, triple writes must be carried out by the CPU to have any effect on the WUT alarm. ALM2 can be written to regardless of whether ALMEN or CNTEN is active in the CR0 register.

In contrast, on RTC1, full support for the fractional alarm time is present, such that an alarm can be specified down to an individual 32 kHz clock cycle using ALM2.

Table 395. Bit Descriptions for ALM2

Bits	Bit Name	Settings	Description	Reset	Access
15	Reserved		Reserved.	0x0	R
[14:0]	VALUE		Fractional Nonprescaled Bits of the WUT Alarm Target Time. Note that any value written to ALM2 must be in keeping with the number of prescale bits specified by the PRESCALE2EXP field of the CR1 MMR. If a value in ALM2 cannot be reached by the degree of prescaling configured by PRESCALE2EXP, the whole value of ALM2 is treated as if it were zero.	0x0	R/W

STATUS 6 REGISTER

Address: 0x40001488, Reset: 0x7900, Name: SR6

SR6 is a status register that provides the unread status of snapshots of input capture channels, SNAP0, SNAP1, and SNAP2.

Table 396. Bit Descriptions for SR6

Bits	Bit Name	Settings	Description	Reset	Access
[15:11]	Reserved		Reserved.	0xF	R
[10:9]	RTCFRZCNTPTR		Pointer for the Triple Read Sequence of the CNTx MMRs. This bit indicates the sequence number for the next read in triple read sequences of the CNTx registers. CNTx allows a coherent triple 16-bit read of the 47-bit WUT count contained in CNT2, CNT1, and CNT0. CNTx is always read in sequences of three reads (although these can be spread out in time, or have other APB accesses interspersed), so that the first read in the sequence returns the current value of CNT0. Simultaneously with this first read, a snapshot is taken of the values of CNT2 and CNT1 so that in the second and third reads in the sequence of CNTx, the snapshot values of CNT1 and CNT2 are returned, respectively. In this way, a triple read of CNTx gives an overall 47 bits of the WUT count that belong together and are coherent with each other, even though the actual continuing value of CNT2, CNT1, and CNT0 keeps advancing while the WUT counts real time. This bit both indicates the sequence number in the triple read and acts a read data select for the value returned when CNTx is read. Normally, this bit keeps advancing by one, starting from 0b00, and wrapping from 0b10 to 0b00 with every read. However, to clear the CNTx pointer and reinitialize the count sequence for reads of CNTx, the CPU can write a software key (value of 0x9376) to the GWY register.	0x0	R
		00	The next read of CNTx causes the read data for CNTx to be the current value of CNT0 and takes a coherent snapshot of the current values of CNT2 and CNT1 for return during the subsequent two reads of CNTx. The value that can be read in SNAP0, SNAP1, and SNAP2 is due to a software initiated snapshot.		
		01	The next read of CNTx is the second in a triple read sequence and returns the snapshot of CNT1 that was taken when the first read of CNTx in the sequence occurred.		
		10	The next read of CNTx is the third in a triple read sequence and returns the snapshot of CNT2 that was taken when the first read of CNTx in the sequence occurred.		

REGISTER DETAILS: DIGITAL DIE WAKE-UP TIMER

Table 396. Bit Descriptions for SR6 (Continued)

Bits	Bit Name	Settings	Description	Reset	Access
[8:0]	Reserved		Reserved.	0x100	R

CYCLIC REDUNDANCY CHECK

The CRC accelerator on the digital die computes the CRC for a block of memory locations on the digital die only. The exact memory location can be in the SRAM, flash, or any combination of memory mapped registers. The CRC accelerator generates a checksum that can be compared to an expected signature. The final CRC comparison is the responsibility of the MCU. CRC function is not supported for the AFE die blocks.

CRC FEATURES

The CRC, used by the ADuCM355 MCU, supports the following features:

- Generation of a CRC signature for a block of data.
- Programmable polynomial length of up to 32 bits.
- Operates on 32 bits of data at a time.
- MSB first and LSB first CRC implementations.
- Various data mirroring capabilities.
- Initial seed to be programmed by user.
- DMA controller (using software DMA) can be used for data transfer to offload workload from the MCU.

CRC FUNCTIONAL DESCRIPTION

The following sections detail the functionality of the CRC. Control for address decrementation and incrementation options for computing the CRC on a block of memory is in the DMA controller, and details on these options can be found in the DMA Controller section.

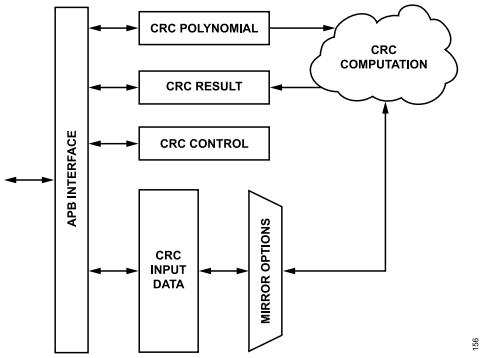

Figure 66. CRC Block Diagram

CRC Architectural Concepts

The CRC accelerator works on 32-bit data words, which are either fed to the block through the DMA channel dedicated to the CRC accelerator or directly by the MCU. The CRC accelerator guarantees immediate availability of the CRC output.

CRC Operating Modes

The accelerator calculates CRC on the data stream it receives, 32 bits at a time. The CRC is then written into the block using either the DMA engine or via the MCU directly.

The CRC works on 32-bit data words. For data words less than 32 bits in size, the MCU must pack the data into 32-bit data units. Data mirroring on the input data can be performed at bit, byte, or word level (only for 32-bit data) by setting CTL, Bits[4:2].

When operating, the CRC algorithm runs on the incoming data stream written to the IPDATA register. For every new word of data received, the CRC is computed and the result register is updated with the calculated CRC. The CRC accelerator guarantees the immediate availability of the CRC result up to the current data in the result register.

The CRC engine uses the current result for generating the next result when a new data word is received. The result register can be programmed with an initial seed. The bit width of the seed value for an x-bit polynomial must be x. The seed must be justified in the result register.

CYCLIC REDUNDANCY CHECK

Polynomial

The CRC accelerator supports the calculation of the CRC using any length polynomial. The polynomial must be written to the polynomial register. For MSB first implementation, omit the highest power while programming the CRC polynomial register and left justify the polynomial. For LSB first implementation, right justify the polynomial and omit the LSB. The result register contains x-bit MSBs as a checksum for an x-bit CRC polynomial.

The following examples illustrate the CRC polynomial.

16-Bit Polynomial Programming for MSB First Calculation

Polynomial: CRC-16-CCITT

$x^{16} + x^{12} + x^5 + 1$ = (1) 0001 0000 0010 0001 = 0x1021

where the largest exponent (x^{16} term) is implied. Therefore, the polynomial is 0001 0000 0010 0001.

When left justified in the polynomial register, the register format is detailed in Table 397.

Table 397. 16-Bit Polynomial Programming Register Format, MSB First Calculation

Register	Bit(s)	Value
CRC Polynomial Register (POLY)	[31:24]	0001 0000
	[23:16]	0010 0001
	[15:8]	0x08B0
	[7:0]	0x08B0
CRC Result Register (Result)	[31:24]	CRC
	[23:16]	Result
	[15:8]	0x08B0
	[7:0]	0x08B0
Initial Seed Programmed in CRC Result Register (Result)	[31:24]	CRC
	[23:16]	Seed
	[15:8]	0x08B0
	[7:0]	0x08B0

16-Bit Polynomial Programming for LSB First Calculation

Polynomial: CRC-16-CCITT

$x^{16} + x^{12} + x^5 + 1$ = 1000 0100 0000 1000 (1) = 0x8408

where the smallest exponent (x^0 term) is implied. Therefore, the polynomial is 1000 0100 0000 1000.

When right justified in the polynomial register, the register format is detailed in Table 398.

Table 398. 16-Bit Polynomial Programming Register Format, LSB First Calculation

Register	Bit(s)	Value
CRC Polynomial Register (POLY)	[31:24]	0x08B0
	[23:16]	0x08B0
	[15:8]	1000 0100
	[7:0]	0000 1000
CRC Result Register (Result)	[31:24]	0x08B0
	[23:16]	0x08B0
	[15:8]	CRC
	[7:0]	Result
Initial Seed Programmed in CRC Result Register (Result)	[31:24]	0x08B0
	[23:16]	0x08B0

CYCLIC REDUNDANCY CHECK

Table 398. 16-Bit Polynomial Programming Register Format, LSB First Calculation (Continued)

Register	Bit(s)	Value
	[15:8]	CRC
	[7:0]	Seed

8-Bit Polynomial Programming for MSB First Calculation

Polynomial: CRC-8-ATM

$x^8 + x^2 + x + 1$ = (1) 0000 0111 = 0x07

where the largest exponent (x^8 term) is implied. Therefore, the polynomial is 0000 0111.

When left justified in the polynomial register, the register format is detailed in Table 399.

Table 399. 8-Bit Polynomial Programming Register Format, MSB First Calculation

Register	Bit(s)	Value
CRC Polynomial Register (POLY)	[31:24]	0000 0111
	[23:16]	0x08B0
	[15:8]	0x08B0
	[7:0]	0x08B0
CRC Result Register (Result)	[31:24]	CRC result
	[23:16]	0x08B0
	[15:8]	0x08B0
	[7:0]	0x08B0
Initial Seed Programmed in CRC Result Register (Result)	[31:24]	CRC seed
	[23:16]	0x08B0
	[15:8]	0x08B0
	[7:0]	0x08B0

8-Bit Polynomial Programming for LSB First Calculation

Polynomial: CRC-8-ATM

$x^8 + x^2 + x + 1$ = 1000 0011 (1) = 0x83

where the smallest exponent (x^0 term) is implied. Therefore, the polynomial is 1000 0011.

When right justified in the polynomial register, the register format is detailed in Table 400.

Table 400. 8-Bit Polynomial Programming Register Format, LSB First Calculation

Register	Bit(s)	Value
CRC Polynomial Register (POLY)	[31:24]	0x08B0
	[23:16]	0x08B0
	[15:8]	0x08B0
	[7:0]	1000 0011
CRC Result Register (Result)	[31:24]	0x08B0
	[23:16]	0x08B0
	[15:8]	0x08B0
	[7:0]	CRC result
Initial Seed Programmed in CRC Result Register (Result)	[31:24]	0x08B0
	[23:16]	0x08B0
	[15:8]	0x08B0

CYCLIC REDUNDANCY CHECK

Table 400. 8-Bit Polynomial Programming Register Format, LSB First Calculation (Continued)

Register	Bit(s)	Value
	[7:0]	CRC seed

The CRC engine uses the following 32-bit CRC polynomial as the default (as per the IEEE 802.3 standard):

$g(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$

This value is programmed for MSB first calculation by default as shown in Table 401.

Table 401. Default CRC Polynomial Register (POLY) Value

Bit(s)	Value
[31:24]	0x04
[23:16]	0xC1
[15:8]	0x1D
[7:0]	0xB7

Reset and Hibernate Modes

The CRC configuration bits are retained, except for the block enable bit (CTL, Bit 0). The block must be enabled again after exiting hibernate mode. The CRC polynomial and CRC result registers are retained after exiting hibernate mode. See Table 402 for details on the CRC registers after a reset and in hibernate mode.

Table 402. Reset and Hibernate by Register

Register	Reset	Hibernate
CTL	0x10000000	Apart from EN, all other bits retained
POLY	0x04C11DB7	Retained
IPDATA	0x0	Not retained
Result	0x0	Retained

CRC DATA TRANSFER

The data stream can be written to the block using the DMA controller or by using the MCU directly.

CRC INTERRUPTS AND EXCEPTIONS

The DMA channel generates an interrupt upon completion of data transfer to the CRC block.

CRC PROGRAMMING MODEL

The CRC block is provided to calculate the CRC signature over a block of data in the background while the core performs other tasks. The CRC block supports two modes of CRC calculation: core access and DMA access.

Core Access Steps

To access the core, take the following steps:

1. Program the POLY register with the required polynomial justified, as detailed in the Polynomial section.
2. Program the result register with the initial seed. The seed must be justified and written to the result register, as detailed in the Polynomial section.
3. Enable accelerator function by writing to the CTL register. Note that the following steps require a single write to the CTL register:
   a. Set the EN bit high.
   b. Modify the W16SWP, BYTMIRR, and BITMIRR bits in the CTL register, which configure the application with different mirroring options. For more information, see the Mirroring Options section.
   c. Set or reset the LSBFIRST bit to indicate whether to use LSB first or MSB first in CRC calculation.

CYCLIC REDUNDANCY CHECK

The core can start sending data to the CRC block by writing into the IPDATA register. The CRC accelerator continues to calculate the CRC as long as data is written to the IPDATA register. It is the responsibility of the application to count the number of words written to the CRC block. After all the words are written, the application can read the result register.

4. Read the result register. This register contains the x-bit result in x MSB bits for MSB first and in x LSB bits for LSB first CRC calculation.
5. Calculate CRC on the next data block. To calculate the CRC on the next block of data, repeat Step 1 to Step 4.
6. Disable the CRC accelerator block by clearing the EN bit in CTL to ensure that the block is in a low-power state.

CYCLIC REDUNDANCY CHECK

DMA Access Steps

The CRC accelerator block supports software DMA. To access the DMA, take the following steps:

1. Program the POLY register with the required polynomial left justified, as shown in the Polynomial section.
2. Program the result register with an initial seed value. The seed must be justified and written to the result register, as detailed in the Polynomial section.
3. Enable accelerator function by writing to the CTL register. Note that the following steps require a single write to the CTL register:
   a. Set the EN bit high.
   b. Modify the W16SWP, BYTMIRR, and BITMIRR bits in the CTL register, which configure the application with different mirroring options. For more information, see the Mirroring Options section.
   c. Set or reset the LSBFIRST bit to indicate LSB first or MSB first CRC calculation.

   The DMA can start sending CRC data by writing to the IPDATA register. The CRC accelerator block continues to calculate the CRC as long as the data is written to IPDATA.
4. Set up the DMA channels using the required parameters: DST_END_PTR is the IPDATA register address, data size is the word, destination no increment is the channel used. For more information about programming the DMA, see the DMA Controller section. A DMA_DONE interrupt signal of the DMA channel indicates the completion of data transfer to the CRC block.
5. Repeat Step 1 to Step 4 until all the data has been sent to the accelerator block.
6. Read the result register. This register contains the x-bit result in x MSB bits for MSB first and in x LSB bits for LSB first CRC calculations.
7. Calculate CRC on the next data block. To calculate the CRC on the next block of data, repeat Step 1 to Step 5.
8. Disable the CRC accelerator block by clearing the EN bit in CTL to ensure that the block is in a low-power state.

Mirroring Options

The W16SWP, BITMIRR, and BYTMIRR bits in CTL determine the sequence of the bits in which the CRC is calculated. Table 403 details all of the mirroring options used within the CRC block for a 32-bit polynomial.

DIN, Bits[31:0] is the data being written to the IPDATA register, and CIN, Bits[31:0] is the data after the mirroring of the data. The serial engine calculates CIN, Bits[31:0] starting with the MSB bit and ending with LSB bit in sequence (CIN, Bit 31 to CIN, Bit 0 in descending order).

Table 403. Mirroring Options for 32-Bit Input Data with 32-Bit Polynomial

W16SWP	BYTMIRR	BITMIRR	Input Data DIN, Bits[31:0]	CRC Input Data (CIN, Bits[31:0])
0	0	0	DIN, Bits[31:0]	CIN, Bits[31:0] = DIN, Bits[31:0]
0	0	1	DIN, Bits[31:0]	CIN, Bits[31:0] = DIN, Bits[31:24]; DIN, Bits[23:16]; DIN, Bits[15:8]; DIN, Bits[7:0]
0	1	0	DIN, Bits[31:0]	CIN, Bits[31:0] = DIN, Bits[31:24]; DIN, Bits[23:16]; DIN, Bits[15:8]; DIN, Bits[7:0]
0	1	1	DIN, Bits[31:0]	CIN, Bits[31:0] = DIN, Bits[31:24]; DIN, Bits[23:16]; DIN, Bits[15:8]; DIN, Bits[7:0]
1	0	0	DIN, Bits[31:0]	CIN, Bits[31:0] = DIN, Bits[31:16]; DIN, Bits[15:0]
1	0	1	DIN, Bits[31:0]	CIN, Bits[31:0] = DIN, Bits[31:24]; DIN, Bits[23:16]; DIN, Bits[15:8]; DIN, Bits[7:0]
1	1	0	DIN, Bits[31:0]	CIN, Bits[31:0] = DIN, Bits[31:24]; DIN, Bits[23:16]; DIN, Bits[15:8]; DIN, Bits[7:0]
1	1	1	DIN, Bits[31:0]	CIN, Bits[31:0] = DIN, Bits[31:24]; DIN, Bits[23:16]; DIN, Bits[15:8]; DIN, Bits[7:0]

REGISTER SUMMARY: CRC

Table 404. CRC Register Summary

Address	Name	Description	Reset	Access
0x40040000	CTL	CRC control register	0x10000000	R/W
0x40040004	IPDATA	Input data word register	0x00000000	W
0x40040008	RESULT	CRC result register	0x00000000	R/W
0x4004000C	POLY	Programmable CRC polynomial	0x04C11DB7	R/W
0x40040010 to 0x40040017	IPBITSN	Input data bits	0x00 for 8 bytes	W
0x40040010	IPBYTE	Input data byte	0x00	W

REGISTER DETAILS: CRC

CRC CONTROL REGISTER

Address: 0x40040000, Reset: 0x10000000, Name: CTL

Table 405. Bit Descriptions for CTL

Bits	Bit Name	Settings	Description	Reset	Access
[31:28]	REVID		Revision ID.	0x1	R
[27:5]	Reserved		Reserved.	0x0	R
4	W16SWP		Word 16 Swap. This bit swaps 16-bit half words within a 32-bit word.	0x0	R/W
		0	Word 16 swap disabled.		
		1	Word 16 swap enabled.		
3	BYTMIRR		Byte Mirroring. This bit swaps 8-bit bytes within each 16-bit half word.	0x0	R/W
		0	Byte mirroring is disabled.		
		1	Byte Mirroring is enabled.		
2	BITMIRR		Bit Mirroring. This bit swaps bits within each byte.	0x0	R/W
		0	Bit mirroring is disabled.		
		1	Bit mirroring is enabled.		
1	LSBFIRST		LSB First Calculation Order.	0x0	R/W
		0	MSB first CRC calculation.		
		1	LSB first CRC calculation.		
0	EN		CRC Peripheral Enable.	0x0	R/W
		0	CRC peripheral is disabled.		
		1	CRC peripheral is enabled.		

INPUT DATA WORD REGISTER

Address: 0x40040004, Reset: 0x00000000, Name: IPDATA

Table 406. Bit Descriptions for IPDATA

Bits	Bit Name	Settings	Description	Reset	Access
[31:0]	VALUE		Data Input.	0x0	W

CRC RESULT REGISTER

Address: 0x40040008, Reset: 0x00000000, Name: RESULT

Table 407. Bit Descriptions for RESULT

Bits	Bit Name	Settings	Description	Reset	Access
[31:0]	VALUE		CRC Reset.	0x0	R/W

PROGRAMMABLE CRC POLYNOMIAL REGISTER

Address: 0x4004000C, Reset: 0x04C11DB7, Name: POLY

Table 408. Bit Descriptions for POLY

Bits	Bit Name	Settings	Description	Reset	Access
[31:0]	VALUE		CRC Reduction Polynomial.	0x4C11DB7	R/W

INPUT DATA BITS REGISTER

Address: 0x40040010 to 0x40040017 (Increments of 0x01), Reset: 0x00, Name: IPBITSN

Table 409. Bit Descriptions for IPBITSN

Bits	Bit Name	Settings	Description	Reset	Access
[7:0]	DATA_BITS		Input Data Bits. These fields are used to calculate CRC data byte from 1 bit to 7 bits of input data. Computing CRC on x bits of input data can be achieved by writing the byte to Bit x of this register.	0x0	W

REGISTER DETAILS: CRC

INPUT DATA BYTE REGISTER

Address: 0x40040010, Reset: 0x00, Name: IPBYTE

Table 410. Bit Descriptions for IPBYTE

Bits	Bit Name	Settings	Description	Reset	Access
[7:0]	DATA_BYTE		Input Data Byte. Writing data to this field calculates CRC on a byte of data.	0x0	W

HARDWARE DESIGN CONSIDERATIONS

TYPICAL SYSTEM CONFIGURATION

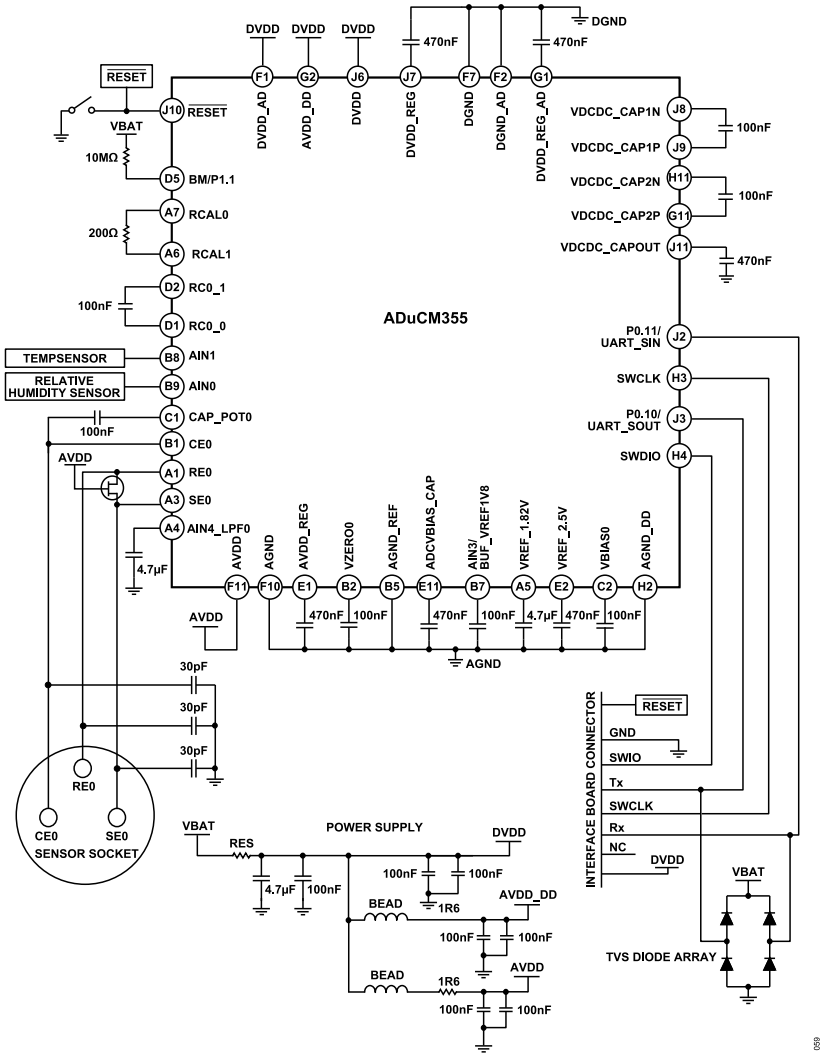

Figure 67. ADuCM355 Typical System Configuration

SERIAL WIRE DEBUG INTERFACE

The SWD interface provides a debug port for pin limited packages. The SWD replaces the 5-pin JTAG port with the SWCLK pin and a single bidirectional data pin (SWDIO), providing all the normal JTAG debug and test functionality. SWDIO and SWCLK are overlaid on the TMS and TCK pins, respectively, on the Arm 20-pin JTAG interface (see Figure 68).

HARDWARE DESIGN CONSIDERATIONS

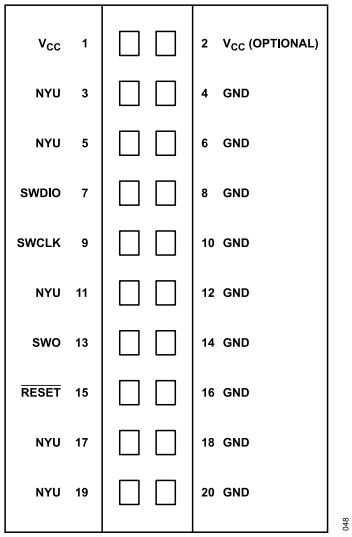

Figure 68. SWD 20-Pin Connector Pinout

Table 411. SWD Connections

Signal	Connection
SWDIO	Data input/output pin. Use a 100 kΩ pull-up resistor to V_{CC} from SWDIO.
SWO	No connect.
SWCLK	Clock pin. Use a 100 kΩ pull-up resistor to V_{CC} from SWCLK.
V_{CC}	Positive supply voltage. Power supply for JTAG interface drivers.
GND	Digital ground.
$\overline{\text{RESET}}$	No connect.
NYU	No connect, not used.

I^2C refers to a communications protocol originally developed by Philips Semiconductors (now NXP Semiconductors).