



### FEATURES

#### Analog input/output

- Dual (24-bit) ADCs
- Single-ended and differential inputs
- Programmable ADC output rate (4 Hz to 8 kHz)
- Programmable digital filters
- Built-in system calibration

#### Low power operation mode

- Primary (24-bit) ADC channel
  - 2 differential pairs or 4 single-ended channels
  - PGA (1 to 512) input stage
  - Selectable input range:  $\pm 2.34$  mV to  $\pm 1.2$  V
  - 30 nV rms noise
- Auxiliary (24-bit) ADC: 4 differential pairs or 7 single-ended channels

#### On-chip precision reference ( $\pm 10$ ppm/ $^{\circ}$ C)

- Programmable sensor excitation current sources
  - 200  $\mu$ A to 2 mA current source range
- Single 14-bit voltage output DAC

#### Microcontroller

- ARM7TDMI core, 16-/32-bit RISC architecture
- JTAG port supports code download and debug
- Multiple clocking options

#### Memory

- 32 kB (16 kB  $\times$  16) Flash/EE memory, including 2 kB kernel
- 4 kB (1 kB  $\times$  32) SRAM

#### Tools

- In-circuit download, JTAG based debug
- Low cost, QuickStart™ development system

#### Communications interfaces

##### SPI interface (5 Mbps)

- 4-byte receive and transmit FIFOs

##### UART serial I/O and I<sup>2</sup>C (master/slave)

#### On-chip peripherals

- 4 $\times$  general-purpose (capture) timers including
  - Wake-up timer
  - Watchdog timer

#### Vectored interrupt controller for FIQ and IRQ

- 8 priority levels for each interrupt type
- Interrupt on edge or level external pin inputs

#### 16-bit, 6-channel PWM

#### General-purpose inputs/outputs

- Up to 14 GPIO pins that are fully 3.3 V compliant

#### Power

- AVDD/DVDD specified for 2.5 V ( $\pm 5\%$ )
- Active mode: 2.74 mA (@ 640 kHz, ADC0 active)
- 10 mA (@ 10.24 MHz, both ADCs active)

#### Packages and temperature range

- Fully specified for  $-40^{\circ}$ C to  $+125^{\circ}$ C operation
- 32-lead LFCSP (5 mm  $\times$  5 mm)
- 48-lead LFCSP and LQFP

#### Derivatives

- 32-lead LFCSP ([ADuC7061](#))
- 48-lead LQFP and 48-lead LFCSP ([ADuC7060](#))

### APPLICATIONS

#### Industrial automation and process control

Intelligent, precision sensing systems, 4 mA to 20 mA loop-based smart sensors

### GENERAL DESCRIPTION

The [ADuC7060/ADuC7061](#) series are fully integrated, 8 kSPS, 24-bit data acquisition systems incorporating high performance multichannel sigma-delta ( $\Sigma$ - $\Delta$ ) analog-to-digital converters (ADCs), 16-bit/ 32-bit ARM7TDMI® MCU, and Flash/EE memory on a single chip.

The ADCs consist of a primary ADC with two differential pairs or four single-ended channels and an auxiliary ADC with up to seven channels. The ADCs operate in single-ended or differential input mode. A single-channel buffered voltage output DAC is available on chip. The DAC output range is programmable to one of four voltage ranges.

The devices operate from an on-chip oscillator and a PLL generating an internal high frequency clock up to 10.24 MHz. The microcontroller core is an ARM7TDMI, 16-bit/32-bit RISC machine offering up to 10 MIPS peak performance; 4 kB of SRAM and 32 kB of nonvolatile Flash/EE memory are provided on chip. The ARM7TDMI core views all memory and registers as a single linear array.

The [ADuC7060/ADuC7061](#) contains four timers. Timer1 is a wake-up timer with the ability to bring the part out of power saving mode. Timer2 is configurable as a watchdog timer. A 16-bit PWM with six output channels is also provided. The [ADuC7060/ADuC7061](#) contains an advanced interrupt controller. The vectored interrupt controller (VIC) allows every interrupt to be assigned a priority level. It also supports nested interrupts to a maximum level of eight per IRQ and FIQ. When IRQ and FIQ interrupt sources are combined, a total of 16 nested interrupt levels is supported. On-chip factory firmware supports in-circuit serial download via the UART serial interface ports and nonintrusive emulation via the JTAG interface. The parts operate from 2.375 V to 2.625 V over an industrial temperature range of  $-40^{\circ}$ C to  $+125^{\circ}$ C.

Rev. F

[Document Feedback](#)

Information furnished by Analog Devices is believed to be accurate and reliable. However, no responsibility is assumed by Analog Devices for its use, nor for any infringements of patents or other rights of third parties that may result from its use. Specifications subject to change without notice. No license is granted by implication or otherwise under any patent or patent rights of Analog Devices. Trademarks and registered trademarks are the property of their respective owners.

## TABLE OF CONTENTS

|  |    |  |     |
|--|----|--|-----|
| Features .....                                     | 1  | MMR Interface .....  | 57  |
| Applications .....                                 | 1  | Using the DAC .....  | 58  |
| General Description .....                          | 1  | Nonvolatile Flash/EE Memory .....                                  | 59  |
| Revision History .....                             | 3  | Flash/EE Memory Reliability .....                                  | 59  |
| Functional Block Diagram .....                     | 5  | Programming .....  | 59  |
| Specifications .....                               | 6  | Processor Reference Peripherals .....                              | 60  |
| Electrical Specifications .....                    | 6  | Interrupt System .....   | 60  |
| Timing Specifications .....                        | 11 | IRQ .....  | 60  |
| Absolute Maximum Ratings .....                     | 15 | Fast Interrupt Request (FIQ) .....                                 | 61  |
| ESD Caution .....                                  | 15 | Programmed Interrupts .....  | 62  |
| Pin Configurations and Function Descriptions ..... | 16 | Vectored Interrupt Controller (VIC) .....                          | 62  |
| Terminology .....                                  | 21 | VIC MMRs .....   | 62  |
| Overview of the ARM7TDMI Core .....                | 22 | Timers .....   | 67  |
| Thumb Mode (T) .....                               | 22 | HR:MIN:SEC: 1/128 Format .....                                     | 67  |
| Multiplier (M) .....                               | 22 | Timer0 .....   | 68  |
| EmbeddedICE (I) .....                              | 22 | Timer1 or Wake-Up Timer .....                                      | 70  |
| ARM Registers .....                                | 22 | Timer2 or Watchdog Timer .....                                     | 72  |
| Interrupt Latency .....                            | 23 | Timer3 .....   | 74  |
| Memory Organization .....                          | 23 | Pulse-Width Modulator .....  | 76  |
| Flash/EE Control Interface .....                   | 24 | Pulse-Width Modulator General Overview .....                       | 76  |
| Memory Mapped Registers .....                      | 28 | UART Serial Interface .....  | 81  |
| Complete MMR Listing .....                         | 29 | Baud Rate Generation .....   | 81  |
| Reset .....  | 34 | UART Register Definitions .....                                    | 81  |
| Oscillator, PLL, and Power Control .....           | 35 | I <sup>2</sup> C .....   | 87  |
| Clocking System .....                              | 35 | Configuring External Pins for I <sup>2</sup> C Functionality ..... | 87  |
| Power Control System .....                         | 35 | Serial Clock Generation .....                                      | 88  |
| ADC Circuit Information .....                      | 39 | I <sup>2</sup> C Bus Addresses .....                               | 88  |
| Reference Sources .....                            | 40 | I <sup>2</sup> C Registers .....                                   | 88  |
| Diagnostic Current Sources .....                   | 40 | Serial Peripheral Interface .....                                  | 97  |
| Sinc3 Filter .....                                 | 41 | MISO (Master In, Slave Out) Pin .....                              | 97  |
| ADC Chopping .....                                 | 41 | MOSI (Master Out, Slave In) Pin .....                              | 97  |
| Programmable Gain Amplifier .....                  | 41 | SCLK (Serial Clock I/O) Pin .....                                  | 97  |
| Excitation Sources .....                           | 41 | Slave Select (P0.0/SS) Input Pin .....                             | 97  |
| ADC Low Power Mode .....                           | 41 | Configuring External Pins for SPI Functionality .....              | 97  |
| ADC Comparator and Accumulator .....               | 42 | SPI Registers .....  | 98  |
| Temperature Sensor .....                           | 42 | General-Purpose I/O .....  | 102 |
| ADC MMR Interface .....                            | 42 | GPxCON Registers .....   | 102 |
| Example Application Circuits .....                 | 55 | GPxDAT Registers .....   | 103 |
| DAC Peripherals .....                              | 57 | GPxSET Registers .....   | 103 |
| DAC .....  | 57 | GPxCLR Registers .....   | 103 |

|                                      |     |
|--------------------------------------|-----|
| GPxPAR Registers .....               | 103 |
| Hardware Design Considerations ..... | 105 |
| Power Supplies .....                 | 105 |

## REVISION HISTORY

### 2/2017—Rev. E to Rev. F

|  |            |
|--|------------|
| Changed CP-32-4 to CP-32-11 .....            | Throughout |
| Changed CP-48-3 to CP-48-5 .....             | Throughout |
| Changes to Table 5 .....                     | 13         |
| Changes to Table 6 .....                     | 14         |
| Changes to Serial Peripheral Interface ..... | 97         |
| Changes to Figure 30 and Figure 31 .....     | 106        |
| Updated Outline Dimensions .....             | 106        |
| Changes to Ordering Guide .....              | 107        |

### 10/2014—Rev. D to Rev. E

|  |     |
|--|-----|
| Changes to Table 1 .....   | 6   |
| Changed FEESIGN to FEESIG (Throughout) .....                             | 25  |
| Changed FEEHIDE to FEEHID (Throughout) .....                             | 26  |
| Changes to Table 36 .....  | 40  |
| Changes to Sinc3 Filter Section .....                                    | 41  |
| Changes to Table 43 .....  | 46  |
| Changes to ADC Filter Register Section .....                             | 48  |
| Changes to Table 94 .....  | 85  |
| Changes to I <sup>2</sup> C Section .....                                | 87  |
| Changes to Table 97 .....  | 89  |
| Changed Register I2CMSTA, Bit 7 from I2CMNA to I2CMND;<br>Table 98 ..... | 90  |
| Changes to Table 104 .....   | 93  |
| Changes to Table 105 .....   | 94  |
| Updated Figure 31, Outline Dimensions .....                              | 106 |

### 4/2012—Rev. C to Rev. D

|  |     |
|--|-----|
| Changes to Table 1 .....   | 6   |
| Changes to Table 7 .....   | 14  |
| Changes to Table 16 .....  | 25  |
| Change to Command Sequence for Executing a Mass Erase<br>Section .....   | 26  |
| Changes to Table 19 .....  | 29  |
| Changes to Power and Clock Control Registers Section .....   | 35  |
| Changes to Figure 20 .....   | 55  |
| Changes to Bit 5 in Table 63 .....   | 57  |
| Changes to Timers Section; Added Hr:Min:Sec: 1/128 Format<br>Section and Table 79, Renumbered Sequentially ..... | 67  |
| Changes to Timer1 or Wake-Up Timer Section .....   | 70  |
| Changes to Timer2 Load Register Section and Timer2 Value<br>Register Section .....                               | 71  |
| Added Table 108 .....  | 98  |
| Updated Outline Dimensions .....   | 105 |

### 5/2011—Rev. B to Rev. C

|                          |   |
|--------------------------|---|
| Change to Figure 1 ..... | 4 |
| Changes to Table 1 ..... | 6 |

|                          |     |
|--------------------------|-----|
| Outline Dimensions ..... | 106 |
| Ordering Guide .....     | 107 |

|  |     |
|--|-----|
| Add Temporary Protection Section and Keyed Permanent<br>Protection Section .....   | 25  |
| Added Permanent Protection Section and Sequence to Write the<br>Software Protection Key and Set Permanent Protection Section ..... | 26  |
| Changes to Power Control System Section .....  | 35  |
| Changes to Bit 9:6, Table 43 .....   | 45  |
| Changes to Primary Channel ADC Data Register Section and<br>Table 49 .....   | 50  |
| Changes to IRQEN Section and IRQCLR Section .....  | 59  |
| Changes to Timer1 or Wake-Up Timer Section .....   | 69  |
| Changes to Table 108 .....   | 101 |

### 2/2010—Rev. A to Rev. B

|   |     |
|---|-----|
| Changes to Features Section .....                                       | 1   |
| Changes to Table 1 .....  | 4   |
| Changes to Digital I/O Voltage to DGND Parameter .....                  | 14  |
| Changes to Pin 19, Pin 20, and Pin 45 Descriptions (Table 8) ..         | 16  |
| Changes to Pin 13, Pin 14, and Pin 29 Descriptions (Table 9) ..         | 18  |
| Changes to Bit 8 in Table 14 .....                                      | 23  |
| Changes to Table 20 .....   | 28  |
| Changes to Power Control System Section .....                           | 34  |
| Added Table 32 .....  | 35  |
| Changes to Endnote 2 and Endnote 3 of Table 34 .....                    | 36  |
| Changes to Table 42 .....   | 42  |
| Changes to Bit 12 and Bits[3:0] in Table 43 .....                       | 44  |
| Changes to Bit 12 in Table 44 .....                                     | 45  |
| Changes to Endnote 2 in Table 45 .....                                  | 47  |
| Changes to Bit 5 in Table 63 .....                                      | 55  |
| Changes to Serial Downloading (In-Circuit Programming)<br>Section ..... | 57  |
| Changes to Priority Registers Section .....                             | 61  |
| Changes to GPxPAR Registers Section .....                               | 101 |

### 6/2009—Rev. 0 to Rev. A

|   |           |
|---|-----------|
| Added ADuC7061 .....                                      | Universal |
| Added New Package CP-32-4 .....                           | Universal |
| Changes to Features Section .....                         | 1         |
| Changes to General Description Section .....              | 1         |
| Changes to Figure 1 .....                                 | 4         |
| Changes to Table 1 .....                                  | 7         |
| Deleted Endnote to Table 2 .....                          | 10        |
| Changes to Endnotes, Table 3 and Table 4 .....            | 11        |
| Changes to Endnotes, Table 5 .....                        | 12        |
| Changes to Endnotes, Table 6 .....                        | 13        |
| Changes to Figure 7 and Table 8 .....                     | 15        |
| Added Figure 8 and Table 9, Renumbered Sequentially ..... | 18        |
| Changes to Flash EE/Control Interface Section .....       | 23        |
| Change to Code 0x04 Description, Table 15 .....           | 24        |
| Change to Bit 31 Description, Table 16 .....              | 25        |

|  |    |   |     |
|--|----|---|-----|
| Changes to Table 17.....   | 27 | Changes to Figure 23.....   | 65  |
| Changes to Table 19 T0CLRI and Table 20.....   | 28 | Changes Table 78.....   | 66  |
| Changes to Endnote, Table 21.....  | 29 | Changes to Figure 24 and Table 79.....  | 68  |
| Change to SPITX Default Value, Table 25.....   | 30 | Changes to Timer2 Interface Section and Figure 25.....                                  | 69  |
| Changes to External Clock Selection Section.....   | 33 | Changes to Timer3 Capture Register Section.....   | 71  |
| Changes to ADC Circuit Information Section.....  | 36 | Change to Bits[16:12] Description, Table 81.....  | 72  |
| Change to Column Heading Table 35.....   | 37 | Changes Pulse-Width Modulator General Overview Section,<br>Table 82, and Figure 26..... | 73  |
| Change to Bit 6 Description, Table 39.....   | 40 | Changes to Table 84 Column Headings.....  | 75  |
| Change to Bit 12 Description, Table 43.....  | 44 | Changes to Table 92.....  | 82  |
| Changes to Primary Channel ADC Data Register Section<br>and Auxiliary Channel ADC Data Register Section..... | 48 | Changes to Bit 1, Table 102.....  | 90  |
| Change to Table 59 and Figure 17.....  | 51 | Changes to Bit 11 Description, Table 105.....   | 95  |
| Changes to Using the DAC Section.....  | 55 | Changes to SPIMDE Bit Description, Table 106.....                                       | 97  |
| Changes to Nonvolatile Flash/EE Memory Section and<br>Programming Section.....                               | 56 | Updated Outline Dimensions.....   | 103 |
| Changes to Vectored Interrupt Controller (VIC) Section.....  | 59 | Changes to Ordering Guide.....  | 104 |
| Changes to Priority Registers Section.....   | 60 |   |     |
| Change to Table 73.....  | 61 |   |     |

**4/2009—Revision 0: Initial Version**

# FUNCTIONAL BLOCK DIAGRAM

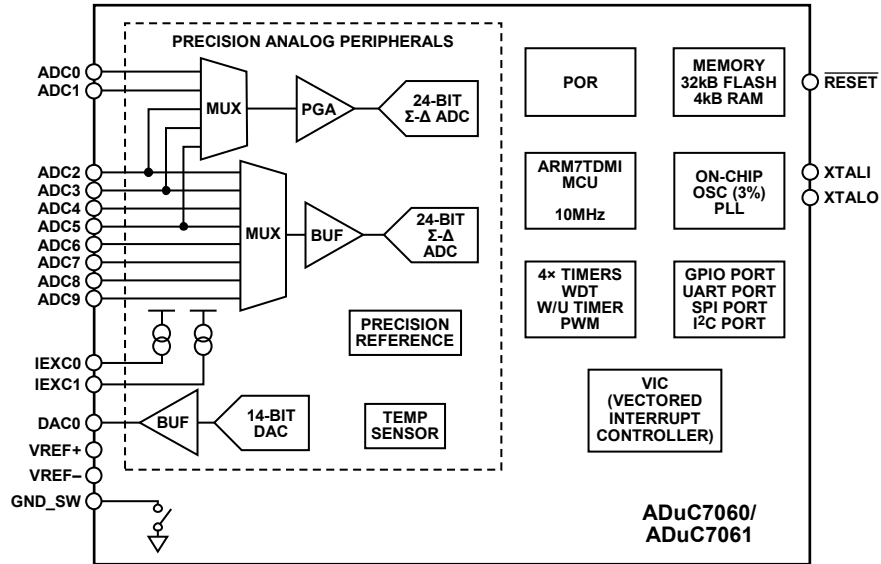


Figure 1.

07079-001

## SPECIFICATIONS

## ELECTRICAL SPECIFICATIONS

$V_{DD} = 2.5\text{ V} \pm 5\%$ ,  $V_{REF+} = 1.2\text{ V}$ ,  $V_{REF-} = \text{GND}$ ,  $f_{CORE} = 10.24\text{ MHz}$  driven from an external 32.768 kHz watch crystal or on-chip oscillator, all specifications  $T_A = -40^\circ\text{C}$  to  $+125^\circ\text{C}$ , unless otherwise noted. Output noise specifications can be found in Table 36 (primary ADC) and Table 38 (ADC auxiliary channel).

Table 1. ADuC7060/ADuC7061 Specifications

| Parameter   | Test Conditions/Comments  | Min  | Typ          | Max  | Unit                  |
|---|---|------|--------------|------|-----------------------|
| ADC SPECIFICATIONS  |   |      |              |      |                       |
| For all ADC specifications, assume normal operating mode unless specifically stated otherwise |   |      |              |      |                       |
| Conversion Rate <sup>1</sup>  | Chop off, ADC normal operating mode   | 50   |              | 8000 | Hz                    |
|   | Chop on, ADC normal operating mode  | 4    |              | 2600 | Hz                    |
|   | Chop on, ADC low power mode   | 1    |              | 650  | Hz                    |
| Main Channel  |   |      |              |      |                       |
| No Missing Codes <sup>1</sup>   | Chop off ( $f_{ADC} \leq 1\text{ kHz}$ )  | 24   |              |      | Bits                  |
|   | Chop on ( $f_{ADC} \leq 666\text{ Hz}$ )  | 24   |              |      | Bits                  |
| Integral Nonlinearity <sup>1,2</sup>  | Gain = 4  |      | $\pm 15$     |      | ppm of FSR            |
| Offset Error <sup>3,4</sup>   | Chop off, offset error is in the order of the noise for the programmed gain and update rate following calibration | -27  | $\pm 8$      | +27  | $\mu\text{V}$         |
| Offset Error <sup>1,3,4</sup>   | Chop on   | -2.7 | $\pm 0.5$    | +2.7 | $\mu\text{V}$         |
| Offset Error Drift vs. Temperature <sup>5</sup>   | Chop off (with gain $\leq 64$ )   |      | 650/PGA_GAIN |      | nV/ $^\circ\text{C}$  |
|   | Chop on (with gain $\leq 64$ )  |      | 10           |      | nV/ $^\circ\text{C}$  |
| Full-Scale Error <sup>1,6,7,8</sup>   | Normal mode   | -1   | $\pm 0.5$    | +1   | mV                    |
| Full-Scale Error <sup>6,8</sup>   | Low power mode  | -2   | $\pm 1.0$    | +2   | mV                    |
| Gain Drift vs. Temperature <sup>9</sup>   |   |      | 5            |      | ppm/ $^\circ\text{C}$ |
| PGA Gain Mismatch Error   |   |      | $\pm 0.1$    |      | %                     |
| Power Supply Rejection <sup>1</sup>   | Chop on, ADC = 1 V (gain = 1)   |      | 65           |      | dB                    |
|   | Chop on, ADC = 7.8 mV (gain = 128)  | 84.7 | 113          |      | dB                    |
|   | Chop off, ADC = 1 V (gain = 1)  | 56   | 65           |      | dB                    |
| Auxiliary Channel   |   |      |              |      |                       |
| No Missing Codes <sup>1</sup>   | Chop off ( $f_{ADC} \leq 1\text{ kHz}$ )  | 24   |              |      | Bits                  |
|   | Chop on ( $f_{ADC} \leq 666\text{ Hz}$ )  | 24   |              |      | Bits                  |
| Integral Nonlinearity <sup>1</sup>  |   |      | $\pm 15$     |      | ppm of FSR            |
| Offset Error <sup>4</sup>   | Chop off  | -120 | $\pm 30$     | +100 | $\mu\text{V}$         |
| Offset Error <sup>1,4</sup>   | Chop on   | -1.5 | $\pm 0.5$    | +3.2 | $\mu\text{V}$         |
| Offset Error Drift vs. Temperature <sup>5</sup>   | Chop off  |      | 200          |      | nV/ $^\circ\text{C}$  |
|   | Chop on   |      | 10           |      | nV/ $^\circ\text{C}$  |
| Full-Scale Error <sup>1,6,7,8</sup>   | Normal mode   | -1   | $\pm 0.5$    | +1   | mV                    |
| Full-Scale Error <sup>1,6,8</sup>   | Low power mode  | -2   | $\pm 1.0$    | +2   | mV                    |
| Gain Drift vs. Temperature <sup>9</sup>   |   |      | 3            |      | ppm/ $^\circ\text{C}$ |
| Power Supply Rejection <sup>1</sup>   | Chop on, ADC = 1 V  | 55   | 65           |      | dB                    |
|   | Chop off, ADC = 1 V   | 53   | 65           |      | dB                    |

| Parameter   | Test Conditions/Comments  | Min  | Typ      | Max             | Unit                  |
|---|---|------|----------|-----------------|-----------------------|
| ADC SPECIFICATIONS: ANALOG INPUT                                  |   |      |          |                 |                       |
| Internal $V_{REF} = 1.2\text{ V}$                                 |   |      |          |                 |                       |
| Main Channel  |   |      |          |                 |                       |
| Absolute Input Voltage Range                                      | Applies to both VIN+ and VIN-   | 0.1  |          | $V_{DD} - 0.7$  | V                     |
| Input Voltage Range   | Gain = 1 <sup>1</sup>   | 0    |          | 1.2             | V                     |
| (Differential Voltage Between AIN+ and AIN-)                      | Gain = 2 <sup>1</sup>   | 0    |          | 600             | mV                    |
|   | Gain = 4 <sup>1</sup>   | 0    |          | 300             | mV                    |
|   | Gain = 8 <sup>1</sup>   | 0    |          | 150             | mV                    |
|   | Gain = 16 <sup>1</sup>  | 0    |          | 75              | mV                    |
|   | Gain = 32 <sup>1</sup>  | 0    |          | 37.5            | mV                    |
|   | Gain = 64 <sup>1</sup>  | 0    |          | 18.75           | mV                    |
|   | Gain = 128 <sup>1</sup>   | 0    |          | 9.375           | mV                    |
| Common Mode Voltage, $V_{CM}$ <sup>10</sup>                       | $V_{CM} = (AIN(+) + AIN(-))/2$ ,<br>gain = 4 to 128                           | 0.5  |          |                 | V                     |
| Input Leakage Current <sup>1</sup>                                | ADC0 and ADC1   |      | 10       | 18 <sup>1</sup> | nA                    |
|   | ADC2, ADC3, ADC4, and ADC5  |      | 15       | 30 <sup>1</sup> | nA                    |
|   | ADC6, ADC7, ADC8, and ADC9,<br>VREF+, VREF-                                   |      | 15       | 25 <sup>1</sup> | nA                    |
|   | Measurements are taken when the ADC is not operating                          |      |          |                 |                       |
| Common-Mode Rejection DC <sup>1</sup><br>On ADC Input             | ADC = 7.8 mV  | 113  |          |                 | dB                    |
|   | ADC = 1 V <sup>1</sup>  |      | 95       |                 | dB                    |
| Common-Mode Rejection<br>50 Hz/60 Hz <sup>1</sup>                 | 50 Hz/60 Hz $\pm 1\text{ Hz}$ , 16.6 Hz and<br>50 Hz update rate, chop on     | 95   |          |                 | dB                    |
|   | ADC = 7.8 mV, range $\pm 20\text{ mV}$<br>ADC = 1 V, range $\pm 1.2\text{ V}$ | 90   |          |                 | dB                    |
| Normal-Mode Rejection<br>50 Hz/60 Hz <sup>1</sup><br>On ADC Input | 50 Hz/60 Hz $\pm 1\text{ Hz}$ , 16.6 Hz $f_{ADC}$ ,<br>chop on                | 75   |          |                 | dB                    |
|   | 50 Hz/60 Hz $\pm 1\text{ Hz}$ , 16.6 Hz $f_{ADC}$ ,<br>chop off               | 67   |          |                 | dB                    |
| Auxiliary Channel   |   |      |          |                 |                       |
| Absolute Input Voltage<br>Range <sup>1</sup>                      | Buffer enabled  | 0.1  |          | $AVDD - 0.1$    | V                     |
|   | Buffer disabled   | AGND |          | $AVDD$          | V                     |
| Input Voltage Range   | Range-based reference source  | 0    |          | 1.2             | V                     |
| Common-Mode Rejection DC <sup>1</sup><br>On ADC Input             | ADC = 1 V <sup>1</sup>  |      | 87       |                 | dB                    |
|   | 50 Hz/60 Hz $\pm 1\text{ Hz}$ , 16.6 Hz and<br>50 Hz update rate, chop on     |      |          |                 |                       |
| Common-Mode Rejection<br>50 Hz/60 Hz <sup>1</sup>                 | ADC = 1 V, range $\pm 1.2\text{ V}$   | 90   |          |                 | dB                    |
|   | 50 Hz/60 Hz $\pm 1\text{ Hz}$ , 16.6 Hz $f_{ADC}$ ,<br>chop on                | 75   |          |                 | dB                    |
| Normal-Mode Rejection<br>50 Hz/60 Hz <sup>1</sup><br>On ADC Input | 50 Hz/60 Hz $\pm 1\text{ Hz}$ , 16.6 Hz $f_{ADC}$ ,<br>chop on                | 75   |          |                 | dB                    |
|   | 50 Hz/60 Hz $\pm 1\text{ Hz}$ , 16.6 Hz $f_{ADC}$ ,<br>chop off               | 67   |          |                 | dB                    |
| VOLTAGE REFERENCE   |   |      |          |                 |                       |
| ADC Precision Reference   |   |      |          |                 |                       |
| Internal $V_{REF}$  |   |      | 1.2      |                 | V                     |
| Initial Accuracy  | Measured at $T_A = 25^\circ\text{C}$  | -0.1 |          | +0.1            | %                     |
| Reference Temperature<br>Coefficient (Tempco) <sup>1, 11</sup>    |   | -20  | $\pm 10$ | +20             | ppm/ $^\circ\text{C}$ |
| Power Supply Rejection <sup>1</sup>                               |   |      | 70       |                 | dB                    |
| External Reference Input Range <sup>12</sup>                      |   | 0.1  |          | $AVDD$          | V                     |
| $V_{REF}$ Divide-by-2 Initial Error <sup>1</sup>                  |   |      | 0.1      |                 | %                     |

| Parameter                                  | Test Conditions/Comments  | Min | Typ       | Max          | Unit                        |
|--|---|-----|-----------|--------------|-----------------------------|
| <b>DAC CHANNEL SPECIFICATIONS</b>          |   |     |           |              |                             |
| Voltage Range                              | $R_L = 5\text{ k}\Omega$ , $C_L = 100\text{ pF}$  | 0   |           | $V_{REF}$    | V                           |
|  |   | 0   |           | $AVDD - 0.2$ | V                           |
| <b>DAC 12-BIT MODE</b>                     |   |     |           |              |                             |
| DC Specifications <sup>13</sup>            |   | 12  |           |              | Bits                        |
| Resolution                                 |   |     | $\pm 2$   |              | LSB                         |
| Relative Accuracy                          | Guaranteed monotonic  |     | $\pm 0.2$ | $\pm 1$      | LSB                         |
| Differential Nonlinearity                  | 1.2 V internal reference  |     | $\pm 2$   | $\pm 15$     | mV                          |
| Offset Error                               | $V_{REF}$ range (reference = 1.2 V)   |     |           | $\pm 1$      | %                           |
| Gain Error                                 | $AVDD$ range  |     |           | $\pm 1$      | %                           |
| Gain Error Mismatch                        |   |     |           | 0.1          | % of full scale on DAC      |
| <b>DAC 16-BIT MODE<sup>1</sup></b>         |   |     |           |              |                             |
| DC Specifications <sup>14</sup>            | Only monotonic to 14 bits   | 14  |           |              | Bits                        |
| Resolution                                 |   |     | $\pm 3$   |              | LSB                         |
| Relative Accuracy                          | For 14-bit resolution   |     | $\pm 0.5$ | $\pm 1$      | LSB                         |
| Differential Nonlinearity                  | Guaranteed monotonic (14 bits)  |     | $\pm 2$   | $\pm 15$     | mV                          |
| Offset Error                               | 1.2 V internal reference  |     | $\pm 1$   |              | %                           |
| Gain Error                                 | $V_{REF}$ range (reference = 1.2 V)   |     | $\pm 1$   |              | %                           |
| Gain Error Mismatch                        | $AVDD$ range  |     | 0.1       |              | % of full scale on DAC      |
| <b>DAC AC CHARACTERISTICS</b>              |   |     |           |              |                             |
| Voltage Output Settling Time               |   |     | 10        |              | $\mu\text{s}$               |
| Digital-to-Analog Glitch Energy            | 1 LSB change at major carry (where maximum number of bits simultaneously change in the DAC0DAT register)    |     | $\pm 20$  |              | nV-sec                      |
| <b>TEMPERATURE SENSOR<sup>1, 15</sup></b>  |   |     |           |              |                             |
| Accuracy                                   | After user calibration<br>MCU in power-down or standby mode   |     | $\pm 4$   |              | $^{\circ}\text{C}$          |
| Voltage Output at $0^{\circ}\text{C}$      | Typical value   |     | 96        |              | mV                          |
| Voltage Tempco                             | Typical value   |     | 0.28      |              | mV/ $^{\circ}\text{C}$      |
| Thermal Impedance                          | 48-lead LFCSP   |     | 27        |              | $^{\circ}\text{C}/\text{W}$ |
|  | 48-lead LQFP  |     | 55        |              | $^{\circ}\text{C}/\text{W}$ |
|  | 32-lead LFCSP   |     | 30        |              | $^{\circ}\text{C}/\text{W}$ |
| <b>GROUND SWITCH<sup>1</sup></b>           |   |     |           |              |                             |
| $R_{ON}$                                   |   |     | 10        |              | $\Omega$                    |
| Input Leakage                              |   |     | 15        |              | nA                          |
| <b>POWER-ON RESET (POR)</b>                |   |     |           |              |                             |
| POR Trip Level <sup>1</sup>                | Refers to voltage at DVDD pin   |     | 2.0       |              | V                           |
|  | Power-on level  |     | 2.25      |              | V                           |
|  | Power-down level  |     |           | 128          | ms                          |
| $\overline{\text{RESET}}$ Timeout from POR | Maximum supply ramp between 1.8 V and 2.25 V; after POR trip, DVDD must reach 2.25 V within this time limit |     |           |              |                             |



| Parameter                                  | Test Conditions/Comments  | Min          | Typ    | Max          | Unit   |
|--|---|--------------|--------|--------------|--------|
| <b>EXCITATION CURRENT SOURCES</b>          |   |              |        |              |        |
| Output Current                             | Available from each current source  | 200          | 1000   |              | μA     |
| Initial Tolerance at 25°C                  |   |              | ±5     |              | %      |
| Drift <sup>1</sup>                         |   |              | 0.06   |              | %/°C   |
| Initial Current Matching at 25°C           | Matching between both current sources   |              | ±0.5   |              | %      |
| Drift Matching <sup>1</sup>                |   |              | 20     |              | ppm/°C |
| Line Regulation (AVDD) <sup>1</sup>        | AVDD = 2.5 V ± 5%   |              | 0.2    |              | %/V    |
| Output Compliance <sup>1</sup>             |   | AGND – 30 mV |        | AVDD – 0.7 V | V      |
| <b>WATCHDOG TIMER (WDT)</b>                |   |              |        |              |        |
| Timeout Period <sup>1</sup>                | 32.768 kHz clock, 256 prescale  | 0.008        |        | 512          | sec    |
| Timeout Step Size                          |   |              | 7.8    |              | ms     |
| <b>FLASH/EE MEMORY<sup>1</sup></b>         |   |              |        |              |        |
| Endurance <sup>16</sup>                    |   | 10,000       |        |              | Cycles |
| Data Retention <sup>17</sup>               |   | 20           |        |              | Years  |
| <b>DIGITAL INPUTS</b>                      |   |              |        |              |        |
| Input Leakage Current                      | All digital inputs except NTRST<br>Input (high) = DVDD  |              | ±1     | ±10          | μA     |
| Input Pull-Up Current                      | Input (low) = 0 V   | 10           | 20     | 80           | μA     |
| Input Capacitance                          |   |              | 10     |              | pF     |
| Input Leakage Current                      | NTRST only: input (low) = 0 V   |              | ±1     | ±10          | μA     |
| Input Pull-Down Current                    | NTRST only: input (high) = DVDD   | 30           | 55     | 100          | μA     |
| <b>LOGIC INPUTS<sup>1</sup></b>            |   |              |        |              |        |
| Input Low Voltage (VINL)                   | All logic inputs  |              |        | 0.4          | V      |
| Input High Voltage (VINH)                  |   | 2.0          |        |              | V      |
| <b>LOGIC OUTPUTS<sup>1</sup></b>           |   |              |        |              |        |
| Output Low Voltage (VOL)                   | All logic outputs except XTALO<br>I <sub>SOURCE</sub> = 1.6 mA                                      |              |        | 0.6          | V      |
| Output High Voltage (VOH)                  | I <sub>SOURCE</sub> = 1.6 mA  | 2.0          |        |              | V      |
| <b>CRYSTAL OSCILLATOR<sup>1</sup></b>      |   |              |        |              |        |
| Logic Inputs, XTALI Only                   |   |              |        | 0.8          | V      |
| Input Low Voltage (VINL)                   |   | 1.7          |        |              | V      |
| Input High Voltage (VINH)                  |   |              |        |              | V      |
| XTALI Capacitance                          |   |              | 12     |              | pF     |
| XTALO Capacitance                          |   |              | 12     |              | pF     |
| <b>ON-CHIP OSCILLATORS</b>                 |   |              |        |              |        |
| Oscillator                                 |   |              | 32,768 |              | kHz    |
| Accuracy                                   |   | -3           |        | +3           | %      |
| <b>MCU CLOCK RATE</b>                      |   |              |        |              |        |
|  | Eight programmable core clock selections within this range: binary divisions 1, 2, 4, 8 ... 64, 128 | 0.08         | 1.28   | 10.24        | MHz    |
| Using an External Clock to P2.0/EXTCLK Pin |   | 0.08         |        | 10.24        | MHz    |
| <b>MCU START-UP TIME</b>                   |   |              |        |              |        |
| At Power-On                                | Includes kernel power-on execution time   |              | 134    |              | ms     |
| After Reset Event                          | Includes kernel power-on execution time   |              | 5      |              | ms     |
| <b>From MCU Power-Down</b>                 |   |              |        |              |        |
| PLL On                                     |   |              |        |              |        |
| Wake-Up from Interrupt                     | CD = 0  |              | 4.8    |              | μs     |
| PLL Off                                    |   |              |        |              |        |
| Wake-Up from Interrupt                     | CD = 0  |              | 66     |              | μs     |
| Internal PLL Lock Time                     |   |              | 1      |              | ms     |

| Parameter                                | Test Conditions/Comments  | Min   | Typ      | Max   | Unit    |
|--|---|-------|----------|-------|---------|
| <b>POWER REQUIREMENTS</b>                |   |       |          |       |         |
| Power Supply Voltages                    |   |       |          |       |         |
| DVDD ( $\pm 5\%$ )                       |   | 2.375 | 2.5      | 2.625 | V       |
| AVDD ( $\pm 5\%$ )                       |   | 2.375 | 2.5      | 2.625 | V       |
| Power Consumption                        |   |       |          |       |         |
| $I_{DD}$ (MCU Normal Mode) <sup>18</sup> | MCU clock rate = 10.24 MHz,<br>ADC0 on  |       | 6        | 10    | mA      |
|  | MCU clock rate = 640 kHz,<br>ADC0 on, G = 4, ADC1/DAC off, SPI<br>on; POWCON1 = 0x4 |       |          |       |         |
|  | Full temperature range  |       |          | 3.1   | mA      |
|  | Reduced temperature range<br>–40°C to +85°C <sup>1</sup>                            |       |          | 2.74  | mA      |
| $I_{DD}$ (MCU Powered Down) <sup>1</sup> | Full temperature range  |       | 55       | 350   | $\mu$ A |
|  | Reduced temperature range<br>–40°C to +85°C   |       | 55       | 120   | $\mu$ A |
| $I_{DD}$ (Primary ADC)                   | PGA enabled, normal mode/low<br>power mode; current is<br>dependent on gain setting |       | 0.6/0.3  |       | mA      |
|  | ADC0 on, G = 1, normal mode   |       | 0.03     |       | mA      |
|  | ADC0 on, G = 4, normal mode   |       | 0.44     |       | mA      |
|  | ADC0 on, G = >128, normal mode  |       | 0.63     |       | mA      |
| $I_{DD}$ (Auxiliary ADC)                 | Normal mode/low power mode  |       | 0.35/0.1 |       | mA      |
| $I_{DD}$ (DAC)                           | DAC0CON = 0x10  |       | 0.33     |       | mA      |
| PWM                                      |   |       | 0.34     |       | mA      |

<sup>1</sup> These numbers are not production tested but are guaranteed by design and/or characterization data at production release.

<sup>2</sup> Valid for primary ADC gain setting of PGA = 4 to 64.

<sup>3</sup> Tested at gain range = 4 after initial offset calibration.

<sup>4</sup> Measured with an internal short. A system zero-scale calibration removes this error. ADC factory calibration done with chop off.

<sup>5</sup> Measured with an internal short.

<sup>6</sup> These numbers do not include internal reference temperature drift.

<sup>7</sup> Factory calibrated at gain = 1.

<sup>8</sup> System calibration at a specific gain range removes the error at this gain range. ADC factory calibration done with chop off.

<sup>9</sup> Measured using an external reference.

<sup>10</sup> Ensure common mode voltage is set so VIN\*Gain setting, which is the PGA output voltage, is between 0.1V and VDD – 0.7V. 900 mV is an optimum value for the common mode voltage across all gains.

<sup>11</sup> Measured using the box method.

<sup>12</sup> References up to AVDD are accommodated by setting ADC0CON Bit 12.

<sup>13</sup> Reference DAC linearity is calculated using a reduced code range of 171 to 4095.

<sup>14</sup> Reference DAC linearity is calculated using a reduced code range of 2731 to 65,535.

<sup>15</sup> Die temperature.

<sup>16</sup> Endurance is qualified to 10,000 cycles as per JEDEC Std. 22 Method A117 and measured at –40°C, +25°C, and +125°C. Typical endurance at 25°C is 170,000 cycles.

<sup>17</sup> Retention lifetime equivalent at junction temperature ( $T_j$ ) = 85°C as per JEDEC Std. 22 Method A117. Retention lifetime derates with junction temperature.

<sup>18</sup> Typical additional supply current consumed during Flash/EE memory program and erase cycles is 7 mA and 5 mA, respectively.

**TIMING SPECIFICATIONS**

***I<sup>2</sup>C* Timing**

**Table 2. I<sup>2</sup>C® Timing in Standard Mode (100 kHz)**

| Parameter        | Description  | Slave |      | Unit |
|------------------|--|-------|------|------|
|                  |  | Min   | Max  |      |
| t <sub>L</sub>   | SCLOCK low pulse width                                       | 4.7   |      | μs   |
| t <sub>H</sub>   | SCLOCK high pulse width                                      | 4.0   |      | ns   |
| t <sub>SHD</sub> | Start condition hold time                                    | 4.0   |      | μs   |
| t <sub>DSU</sub> | Data setup time  | 250   |      | ns   |
| t <sub>DHD</sub> | Data hold time   | 0     | 3.45 | μs   |
| t <sub>RSU</sub> | Setup time for repeated start                                | 4.7   |      | μs   |
| t <sub>PSU</sub> | Stop condition setup time                                    | 4.0   |      | μs   |
| t <sub>BUF</sub> | Bus-free time between a stop condition and a start condition | 4.7   |      | μs   |
| t <sub>R</sub>   | Rise time for both CLOCK and SDATA                           |       | 1    | μs   |
| t <sub>F</sub>   | Fall time for both CLOCK and SDATA                           |       | 300  | ns   |

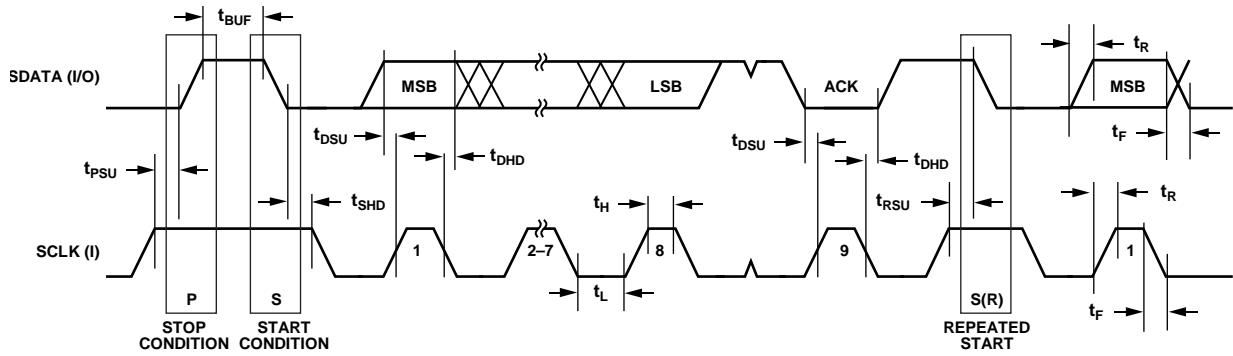


Figure 2. I<sup>2</sup>C Compatible Interface Timing

SPI Timing

Table 3. SPI Master Mode Timing (Phase Mode = 1)

| Parameter        | Description   | Min                   | Typ                              | Max | Unit |
|------------------|---|-----------------------|----------------------------------|-----|------|
| t <sub>SL</sub>  | SCLOCK low pulse width                                |                       | (SPIDIV + 1) × t <sub>HCLK</sub> |     | ns   |
| t <sub>SH</sub>  | SCLOCK high pulse width                               |                       | (SPIDIV + 1) × t <sub>HCLK</sub> |     | ns   |
| t <sub>DAV</sub> | Data output valid after SCLOCK edge                   |                       |                                  | 25  | ns   |
| t <sub>DSU</sub> | Data input setup time before SCLOCK edge <sup>1</sup> | 1 × t <sub>UCLK</sub> |                                  |     | ns   |
| t <sub>DHD</sub> | Data input hold time after SCLOCK edge <sup>1</sup>   | 2 × t <sub>UCLK</sub> |                                  |     | ns   |
| t <sub>DF</sub>  | Data output fall time                                 |                       | 30                               | 40  | ns   |
| t <sub>DR</sub>  | Data output rise time                                 |                       | 30                               | 40  | ns   |
| t <sub>SR</sub>  | SCLOCK rise time                                      |                       | 30                               | 40  | ns   |
| t <sub>SF</sub>  | SCLOCK fall time                                      |                       | 30                               | 40  | ns   |

<sup>1</sup> t<sub>UCLK</sub> = 97.6 ns. It corresponds to the 10.24 MHz internal clock from the PLL.

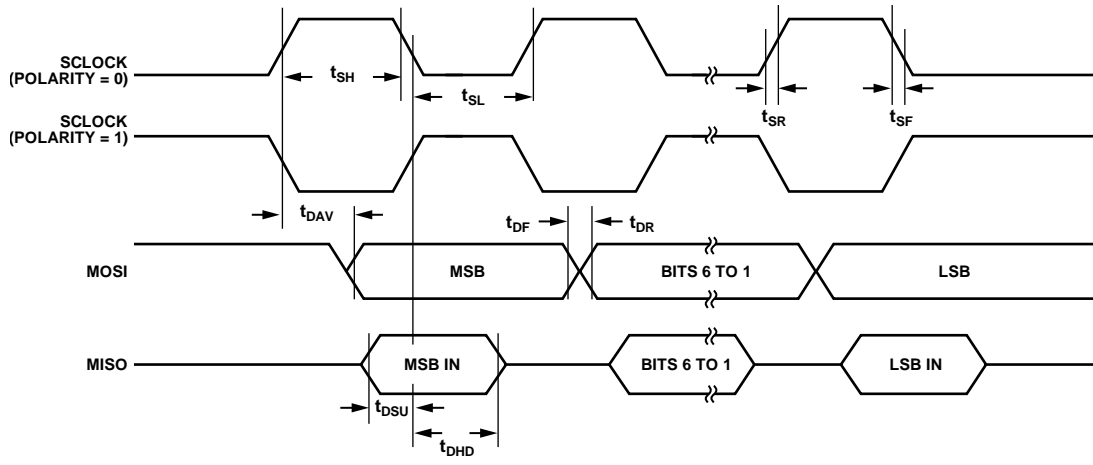


Figure 3. SPI Master Mode Timing (Phase Mode = 1)

Table 4. SPI Master Mode Timing (Phase Mode = 0)

| Parameter         | Description   | Min                   | Typ                              | Max | Unit |
|-------------------|---|-----------------------|----------------------------------|-----|------|
| t <sub>SL</sub>   | SCLOCK low pulse width                                |                       | (SPIDIV + 1) × t <sub>HCLK</sub> |     | ns   |
| t <sub>SH</sub>   | SCLOCK high pulse width                               |                       | (SPIDIV + 1) × t <sub>HCLK</sub> |     | ns   |
| t <sub>DAV</sub>  | Data output valid after SCLOCK edge                   |                       |                                  | 25  | ns   |
| t <sub>DOSU</sub> | Data output setup before SCLOCK edge                  |                       |                                  | 90  | ns   |
| t <sub>DSU</sub>  | Data input setup time before SCLOCK edge <sup>1</sup> | 1 × t <sub>UCLK</sub> |                                  |     | ns   |
| t <sub>DHD</sub>  | Data input hold time after SCLOCK edge <sup>1</sup>   | 2 × t <sub>UCLK</sub> |                                  |     | ns   |
| t <sub>DF</sub>   | Data output fall time                                 |                       | 30                               | 40  | ns   |
| t <sub>DR</sub>   | Data output rise time                                 |                       | 30                               | 40  | ns   |
| t <sub>SR</sub>   | SCLOCK rise time                                      |                       | 30                               | 40  | ns   |
| t <sub>SF</sub>   | SCLOCK fall time                                      |                       | 30                               | 40  | ns   |

<sup>1</sup> t<sub>UCLK</sub> = 97.6 ns. It corresponds to the 10.24 MHz internal clock from the PLL.

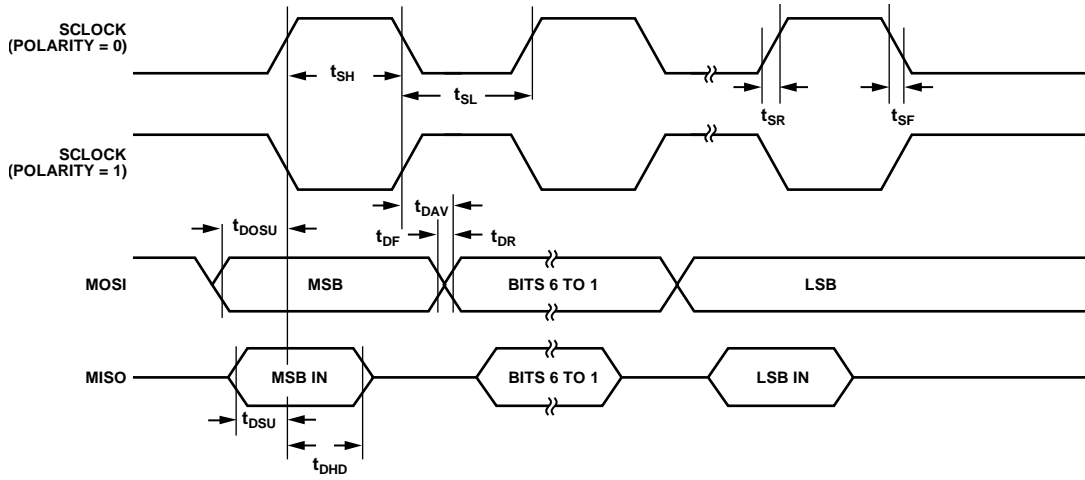


Figure 4. SPI Master Mode Timing (Phase Mode = 0)

07079-031

Table 5. SPI Slave Mode Timing (Phase Mode = 1)

| Parameter           | Description   | Min   | Typ | Max | Unit |
|---------------------|---|---|-----|-----|------|
| $t_{\overline{CS}}$ | $\overline{CS}$ to SCLOCK edge <sup>1</sup>           | $(2 \times t_{HCLK}) + (2 \times t_{UCLK})$ |     |     | ns   |
| $t_{SL}$            | SCLOCK low pulse width                                | $2 \times t_{UCLK}$                         |     |     | ns   |
| $t_{SH}$            | SCLOCK high pulse width                               | $2 \times t_{UCLK}$                         |     |     | ns   |
| $t_{DAV}$           | Data output valid after SCLOCK edge                   |   |     | 40  | ns   |
| $t_{DSU}$           | Data input setup time before SCLOCK edge <sup>1</sup> | $1 \times t_{UCLK}$                         |     |     | ns   |
| $t_{DHD}$           | Data input hold time after SCLOCK edge <sup>1</sup>   | $2 \times t_{UCLK}$                         |     |     | ns   |
| $t_{DF}$            | Data output fall time                                 |   | 30  | 40  | ns   |
| $t_{DR}$            | Data output rise time                                 |   | 30  | 40  | ns   |
| $t_{SR}$            | SCLOCK rise time                                      | 1   |     |     | ns   |
| $t_{SF}$            | SCLOCK fall time                                      | 1   |     |     | ns   |
| $t_{SFS}$           | $\overline{CS}$ high after SCLOCK edge                | 0   |     |     | ns   |

<sup>1</sup>  $t_{UCLK} = 97.6$  ns. It corresponds to the 10.24 MHz internal clock from the PLL.

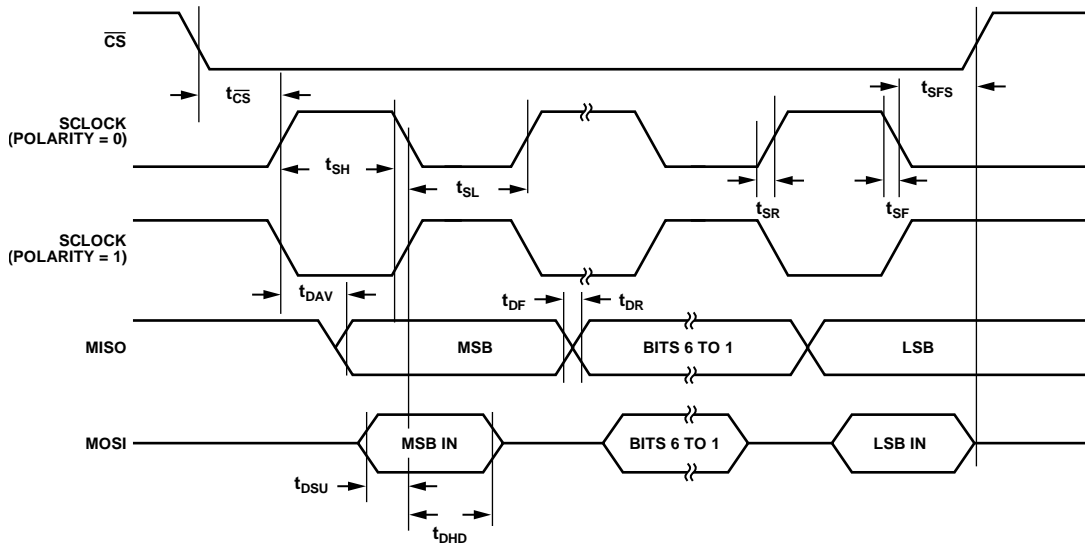


Figure 5. SPI Slave Mode Timing (Phase Mode = 1)

07079-032

Table 6. SPI Slave Mode Timing (Phase Mode = 0)

| Parameter           | Description   | Min   | Typ | Max | Unit |
|---------------------|---|---|-----|-----|------|
| $t_{\overline{CS}}$ | $\overline{CS}$ to SCLOCK edge <sup>1</sup>           | $(2 \times t_{HCLK}) + (2 \times t_{UCLK})$ |     |     | ns   |
| $t_{SL}$            | SCLOCK low pulse width                                | $2 \times t_{UCLK}$                         |     |     | ns   |
| $t_{SH}$            | SCLOCK high pulse width                               | $2 \times t_{UCLK}$                         |     |     | ns   |
| $t_{DAV}$           | Data output valid after SCLOCK edge                   |   |     | 40  | ns   |
| $t_{DSU}$           | Data input setup time before SCLOCK edge <sup>1</sup> | $1 \times t_{UCLK}$                         |     |     | ns   |
| $t_{DHD}$           | Data input hold time after SCLOCK edge <sup>1</sup>   | $2 \times t_{UCLK}$                         |     |     | ns   |
| $t_{DF}$            | Data output fall time                                 |   | 30  | 40  | ns   |
| $t_{DR}$            | Data output rise time                                 |   | 30  | 40  | ns   |
| $t_{SR}$            | SCLOCK rise time                                      | 1   |     |     | ns   |
| $t_{SF}$            | SCLOCK fall time                                      | 1   |     |     | ns   |
| $t_{DOCS}$          | Data output valid after $\overline{CS}$ edge          |   |     | 10  | ns   |
| $t_{SFS}$           | $\overline{CS}$ high after SCLOCK edge                | 0   |     |     | ns   |

<sup>1</sup>  $t_{UCLK} = 97.6$  ns. It corresponds to the 10.24 MHz internal clock from the PLL.

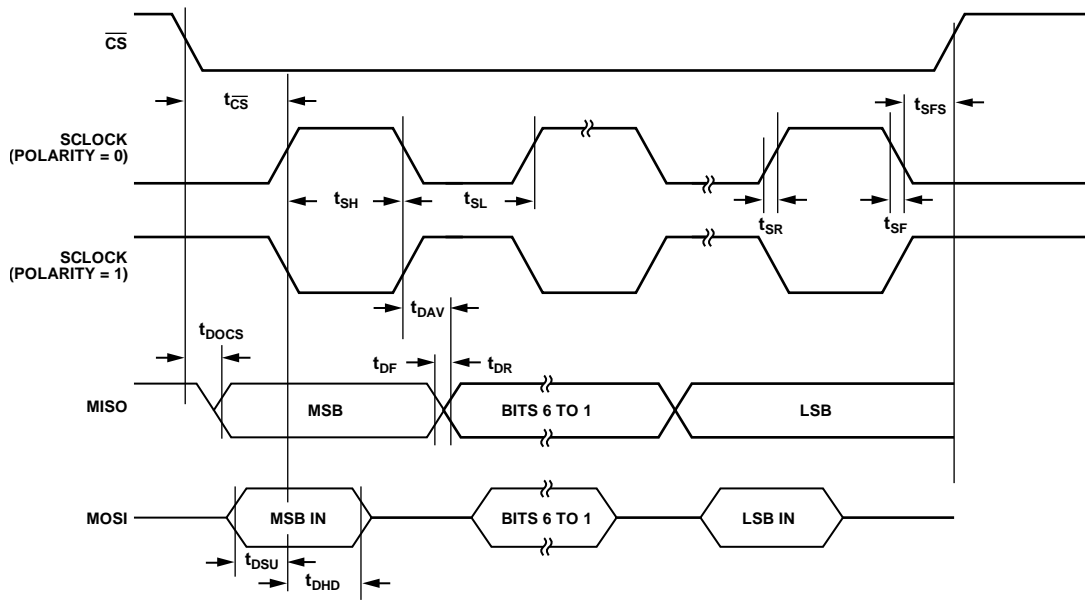


Figure 6. SPI Slave Mode Timing (Phase Mode = 0)

07075-033

## ABSOLUTE MAXIMUM RATINGS

$T_A = -40^{\circ}\text{C}$  to  $+125^{\circ}\text{C}$ , unless otherwise noted.

Table 7.

| Parameter                     | Rating                 |
|-------------------------------|------------------------|
| AGND to DGND to AVDD to DVDD  | -0.3 V to +0.3 V       |
| Digital I/O Voltage to DGND   | -0.3 V to +3.6 V       |
| VREF $\pm$ to AGND            | -0.3 V to AVDD + 0.3 V |
| ADC Inputs to AGND            | -0.3 V to AVDD + 0.3 V |
| ESD (Human Body Model) Rating |                        |
| All Pins                      | $\pm 2$ kV             |
| Storage Temperature           | 125°C                  |
| Junction Temperature          |                        |
| Transient                     | 150°C                  |
| Continuous                    | 130°C                  |
| Lead Temperature              |                        |
| Soldering Reflow (15 sec)     | 260°C                  |

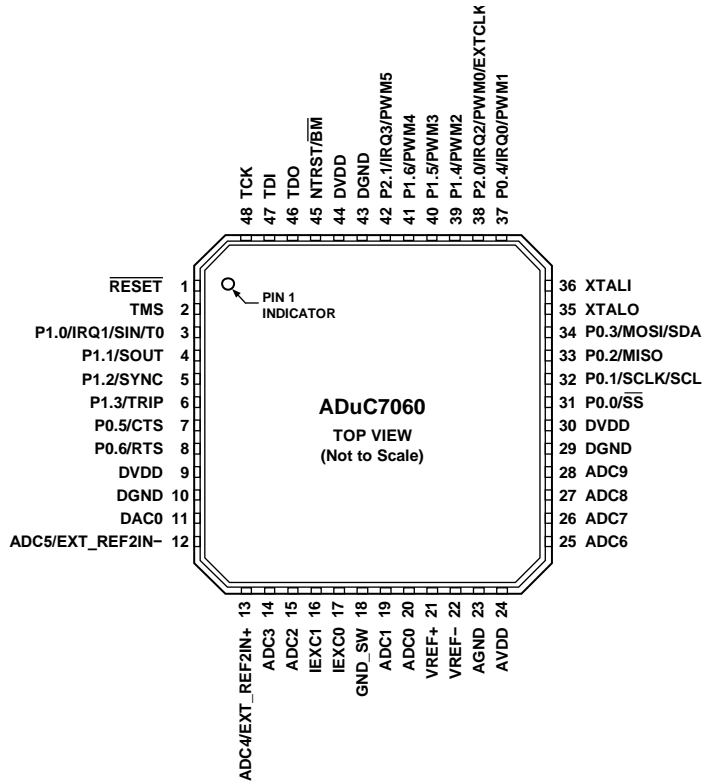
Stresses at or above those listed under Absolute Maximum Ratings may cause permanent damage to the product. This is a stress rating only; functional operation of the product at these or any other conditions above those indicated in the operational section of this specification is not implied. Operation beyond the maximum operating conditions for extended periods may affect product reliability.

### ESD CAUTION



**ESD (electrostatic discharge) sensitive device.** Charged devices and circuit boards can discharge without detection. Although this product features patented or proprietary protection circuitry, damage may occur on devices subjected to high energy ESD. Therefore, proper ESD precautions should be taken to avoid performance degradation or loss of functionality.

PIN CONFIGURATIONS AND FUNCTION DESCRIPTIONS



NOTES  
 1. THE LFCSP\_VQ ONLY HAS AN EXPOSED PADDLE THAT MUST BE LEFT UNCONNECTED. THIS DOES NOT APPLY TO THE LQFP.

07079-002

Figure 7. 48-Lead LQFP and 48-Lead LFCSP\_VQ Pin Configuration

Table 8. ADuC7060 Pin Function Descriptions

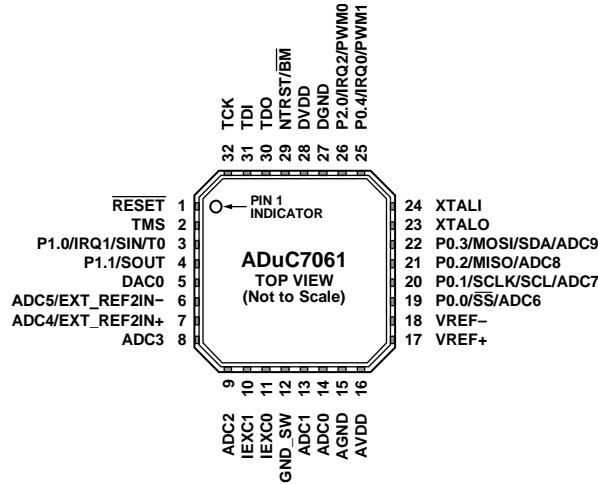
| Pin No. | Mnemonic         | Type <sup>1</sup> | Description   |
|---------|------------------|-------------------|---|
| 0       | EP               |                   | Exposed Paddle. The LFCSP_VQ only has an exposed paddle that must be left unconnected. This does not apply to the LQFP.   |
| 1       | RESET            | I                 | Reset. Input pin, active low. An external 1 kΩ pull-up resistor is recommended with this pin.   |
| 2       | TMS              | I                 | JTAG Test Mode Select. Input pin used for debug and download. An external pull-up resistor (~100 kΩ) should be added to this pin.   |
| 3       | P1.0/IRQ1/SIN/T0 | I/O               | General-Purpose Input and General Purpose Output P1.0/External Interrupt Request 1/Serial Input/Timer0 Input. This is a multifunction input/output pin offering four functions. |
| 4       | P1.1/SOUT        | I/O               | General-Purpose Input and General-Purpose Output P1.1/Serial Output. This is a dual function input/output pin.  |
| 5       | P1.2/SYNC        | I/O               | General-Purpose Input and General-Purpose Output P1.2/PWM External Sync Input. This is a dual function input/output pin.  |
| 6       | P1.3/TRIP        | I/O               | General-Purpose Input and General-Purpose Output P1.3/PWM External Trip Input. This is a dual function input/output pin.  |
| 7       | P0.5/CTS         | I/O               | General-Purpose Input and General-Purpose Output P0.5/Clear-to-Send Signal in UART Mode.  |
| 8       | P0.6/RTS         | I/O               | General-Purpose Input and General-Purpose Output P0.6/Request-to-Send Signal in UART Mode.  |
| 9       | DVDD             | S                 | Digital Supply Pin.   |
| 10      | DGND             | S                 | Digital Ground.   |
| 11      | DAC0             | O                 | DAC Output. Analog output pin.  |



| Pin No. | Mnemonic              | Type <sup>1</sup> | Description   |
|---------|-----------------------|-------------------|---|
| 12      | ADC5/EXT_REF2IN-      | I                 | Single-Ended or Differential Analog Input 5/External Reference Negative Input. This is a dual function analog input pin. ADC5 serves as the analog input for the auxiliary ADC. EXT_REF2IN- serves as the external reference negative input by ADC for the auxiliary channel. |
| 13      | ADC4/EXT_REF2IN+      | I                 | Multifunction Analog Input Pin. This pin can be used for the single-ended or differential Analog Input 4, which is the analog input for the auxiliary ADC, or it can be used for the external reference positive input for the auxiliary channel.                             |
| 14      | ADC3                  | I                 | Single-Ended or Differential Analog Input 3. Analog input for the primary and auxiliary ADCs.   |
| 15      | ADC2                  | I                 | Single-Ended or Differential Analog Input 2. Analog input for the primary and auxiliary ADCs.   |
| 16      | IEXC1                 | O                 | Programmable Current Source. Analog output pin.   |
| 17      | IEXC0                 | O                 | Programmable Current Source. Analog output pin.   |
| 18      | GND_SW                | I                 | Switch to Internal Analog Ground Reference. When this input pin is not used, connect it directly to the AGND system ground.   |
| 19      | ADC1                  | I                 | Single-Ended or Differential Analog Input 1. Analog input for the primary ADC. Negative differential input for primary ADC.   |
| 20      | ADC0                  | I                 | Single-Ended or Differential Analog Input 0. Analog input for the primary ADC. Positive differential input for primary ADC.   |
| 21      | VREF+                 | I                 | External Reference Positive Input for the Primary Channel. Analog input pin.  |
| 22      | VREF-                 | I                 | External Reference Negative Input for the Primary Channel. Analog input pin.  |
| 23      | AGND                  | S                 | Analog Ground.  |
| 24      | AVDD                  | S                 | Analog Supply Pin.  |
| 25      | ADC6                  | I                 | Analog Input 6 for Auxiliary ADC. Single-ended or differential Analog Input 6.  |
| 26      | ADC7                  | I                 | Analog Input 7 for Auxiliary ADC. Single-ended or differential Analog Input 7.  |
| 27      | ADC8                  | I                 | Analog Input 8 for Auxiliary ADC. Single-ended or differential Analog Input 8.  |
| 28      | ADC9                  | I                 | Analog Input 9 for Auxiliary ADC. Single-ended or differential Analog Input 9.  |
| 29      | DGND                  | S                 | Digital Ground.   |
| 30      | DVDD                  | S                 | Digital Supply Pin.   |
| 31      | P0.0/ $\overline{SS}$ | I/O               | General-Purpose Input and General-Purpose Output P0.0/SPI Slave Select Pin (Active Low). This is a dual function input/output pin.  |
| 32      | P0.1/SCLK/SCL         | I/O               | General-Purpose Input and General-Purpose Output P0.1/SPI Clock Pin/I <sup>2</sup> C Clock Pin. This is a triple function input/output pin.   |
| 33      | P0.2/MISO             | I/O               | General-Purpose Input and General-Purpose Output P0.2/SPI Master Input Slave Output. This is a dual function input/output pin.  |
| 34      | P0.3/MOSI/SDA         | I/O               | General-Purpose Input and General-Purpose Output P0.3/SPI Master Output Slave Input/I <sup>2</sup> C Data Pin. This is a triple function input/output pin.  |
| 35      | XTALO                 | O                 | External Crystal Oscillator Output Pin.   |
| 36      | XTALI                 | I                 | External Crystal Oscillator Input Pin.  |
| 37      | P0.4/IRQ0/PWM1        | I/O               | General-Purpose Input and General-Purpose Output P0.4/External Interrupt Request 0/PWM1 Output. This is a triple function input/output pin.   |
| 38      | P2.0/IRQ2/PWM0/EXTCLK | I/O               | General-Purpose Input and General-Purpose Output P2.0/External Interrupt Request 2/PWM0 Output/External Clock Input. This is a multifunction input/output pin.  |
| 39      | P1.4/PWM2             | I/O               | General-Purpose Input and General-Purpose Output P1.4/PWM2 Output. This is a dual function input/output pin.  |
| 40      | P1.5/PWM3             | I/O               | General-Purpose Input and General-Purpose Output P1.5/PWM3 Output. This is a dual function input/output pin.  |
| 41      | P1.6/PWM4             | I/O               | General-Purpose Input and General-Purpose Output P1.6/PWM4 Output. This is a dual function input/output pin.  |
| 42      | P2.1/IRQ3/PWM5        | I/O               | General-Purpose Input and General-Purpose Output P2.1/External Interrupt Request 3/PWM5 Output. This is a triple function input/output pin.   |

| Pin No. | Mnemonic               | Type <sup>1</sup> | Description  |
|---------|------------------------|-------------------|--|
| 43      | DGND                   | S                 | Digital Ground.  |
| 44      | DVDD                   | S                 | Digital Supply Pin.  |
| 45      | NTRST/ $\overline{BM}$ | I                 | JTAG Reset/Boot Mode. Input pin used for debug and download only and boot mode ( $\overline{BM}$ ). The ADuC7060 enters serial download mode if $\overline{BM}$ is low at reset and executes code if $\overline{BM}$ is pulled high at reset through a 13 k $\Omega$ resistor. |
| 46      | TDO                    | O                 | JTAG Data Out. Output pin used for debug and download only.  |
| 47      | TDI                    | I                 | JTAG Data In. Input pin used for debug and download only. Add an external pull-up resistor (~100 k $\Omega$ ) to this pin.   |
| 48      | TCK                    | I                 | JTAG Clock Pin. Input pin used for debug and download only. Add an external pull-up resistor (~100 k $\Omega$ ) to this pin.   |

<sup>1</sup> I = input, O = output, I/O = input/output, and S = supply.



NOTES  
 1. THE 32-LEAD LFCSP\_VQ HAS AN EXPOSED PADDLE. THIS EXPOSED PADDLE MUST BE LEFT UNCONNECTED.

07079-903

Figure 8. 32-Lead LFCSP Pin Configuration

Table 9. ADuC7061 Pin Function Descriptions

| Pin No. | Mnemonic           | Type <sup>1</sup> | Description   |
|---------|--------------------|-------------------|---|
| 0       | EP                 |                   | Exposed Paddle. The 32-lead LFCSP_VQ has an exposed paddle that must be left unconnected.   |
| 1       | RESET              | I                 | Reset Pin. Input pin, active low. An external 1 kΩ pull-up resistor is recommended with this pin.   |
| 2       | TMS                | I                 | JTAG Test Mode Select. Input pin used for debug and download. An external pull-up resistor (~100 kΩ) should be added to this pin.   |
| 3       | P1.0/IRQ1/SIN/T0   | I/O               | General-Purpose Input and General-Purpose Output P1.0/External Interrupt Request 1/Serial Input/Timer0 Input. This is a multifunction input/output pin offering four functions.   |
| 4       | P1.1/SOUT          | I/O               | General-Purpose Input and General-Purpose Output P1.1/Serial Output. This is a dual function input/output pin.  |
| 5       | DAC0               | O                 | DAC Output. Analog output pin.  |
| 6       | ADC5/EXT_REF2IN-   | I                 | Single-Ended or Differential Analog Input 5/External Reference Negative Input. This is a dual function analog input pin. The ADC5 serves as the analog input for the auxiliary ADC. The EXT_REF2IN- serves as the external reference negative input by ADC for the auxiliary channel. |
| 7       | ADC4/EXT_REF2IN+   | I                 | Multifunction Analog Input Pin. This pin can be used for the single-ended or differential Analog Input 4, which is the analog input for the auxiliary ADC, or it can be used for the external reference positive input for the auxiliary channel.                                     |
| 8       | ADC3               | I                 | Single-Ended or Differential Analog Input 3. Analog input for primary and auxiliary ADCs.   |
| 9       | ADC2               | I                 | Single-Ended or Differential Analog Input 2. Analog input for primary and auxiliary ADCs.   |
| 10      | IEXC1              | O                 | Programmable Current Source. Analog output pin.   |
| 11      | IEXC0              | O                 | Programmable Current Source. Analog output pin.   |
| 12      | GND_SW             | I                 | Switch to Internal Analog Ground Reference. When this input pin is not used, connect it directly to the AGND system ground.   |
| 13      | ADC1               | I                 | Single-Ended or Differential Analog Input 1. Analog input for the primary ADC. Negative differential input for primary ADC.   |
| 14      | ADC0               | I                 | Single-Ended or Differential Analog Input 0. Analog input for the primary ADC. Positive differential input for primary ADC.   |
| 15      | AGND               | S                 | Analog Ground.  |
| 16      | AVDD               | S                 | Analog Supply Pin.  |
| 17      | VREF+              | I                 | External Reference Positive Input for the Primary Channel. Analog input pin.  |
| 18      | VREF-              | I                 | External Reference Negative Input for the Primary Channel. Analog input pin.  |
| 19      | P0.0/SS/ADC6       | I/O               | General-Purpose Input and General-Purpose Output P0.0/SPI Slave Select (Active Low)/Input to Auxiliary ADC6. This is a multifunction input/output pin. Single-ended or differential Analog Input 6. Analog input for the auxiliary ADC.   |
| 20      | P0.1/SCLK/SCL/ADC7 | I/O               | General-Purpose Input and General-Purpose Output P0.1/SPI Clock/I <sup>2</sup> C Clock/Input to Auxiliary ADC7. This is a multifunction input/output pin. Single-ended or differential Analog Input 7. Analog input for the auxiliary ADC.  |

| Pin No. | Mnemonic                      | Type <sup>1</sup> | Description   |
|---------|-------------------------------|-------------------|---|
| 21      | P0.2/MISO/ADC8                | I/O               | General-Purpose Input and General-Purpose Output P0.2/SPI Master Input Slave Output/Auxiliary ADC8 Input. This is a triple function input/output pin. Single-ended or differential Analog Input 8. Analog input for the auxiliary ADC.  |
| 22      | P0.3/MOSI/SDA/ADC9            | I/O               | General-Purpose Input and General-Purpose Output P0.3/SPI Master Output Slave Input/I <sup>2</sup> C Data Pin/Auxiliary ADC9 Input. This is a multifunction input/output pin. Single-ended or differential Analog Input 9. Analog input for the auxiliary ADC.                                      |
| 23      | XTALO                         | O                 | External Crystal Oscillator Output Pin.   |
| 24      | XTALI                         | I                 | External Crystal Oscillator Input Pin.  |
| 25      | P0.4/IRQ0/PWM1                | I/O               | General-Purpose Input and General-Purpose Output P0.4/External Interrupt Request 0/PWM1 Output. This is a triple function input/output pin.   |
| 26      | P2.0/IRQ2/PWM0                | I/O               | General-Purpose Input and General-Purpose Output P2.0/External Interrupt Request 2/PWM0 Output. This is a triple function input/output pin.   |
| 27      | DGND                          | S                 | Digital Ground.   |
| 28      | DVDD                          | S                 | Digital Supply Pin.   |
| 29      | NTRST/ $\overline{\text{BM}}$ | I                 | JTAG Reset/Boot Mode. Input pin used for debug and download only and boot mode ( $\overline{\text{BM}}$ ). The ADuC7061 enters serial download mode if $\overline{\text{BM}}$ is low at reset and executes code if $\overline{\text{BM}}$ is pulled high at reset through a 13 k $\Omega$ resistor. |
| 30      | TDO                           | O                 | JTAG Data Out. Output pin used for debug and download only.   |
| 31      | TDI                           | I                 | JTAG Data In. Input pin used for debug and download only. Add an external pull-up resistor (~100 k $\Omega$ ) to this pin.  |
| 32      | TCK                           | I                 | JTAG Clock. Input pin used for debug and download only. Add an external pull-up resistor (~100 k $\Omega$ ) to this pin.  |

<sup>1</sup> I = input, O = output, I/O = input/output, and S = supply.

## TERMINOLOGY

### Conversion Rate

The conversion rate specifies the rate at which an output result is available from the ADC, when the ADC has settled.

The sigma-delta ( $\Sigma$ - $\Delta$ ) conversion techniques used on this part mean that whereas the ADC front-end signal is oversampled at a relatively high sample rate, a subsequent digital filter is used to decimate the output, giving a valid 24-bit data conversion result at output rates from 1 Hz to 8 kHz.

Note that, when software switches from one input to another (on the same ADC), the digital filter must first be cleared and then allowed to average a new result. Depending on the configuration of the ADC and the type of filter, this can take multiple conversion cycles.

### Integral Nonlinearity (INL)

INL is the maximum deviation of any code from a straight line passing through the endpoints of the transfer function. The endpoints of the transfer function are zero scale, a point  $\frac{1}{2}$  LSB below the first code transition, and full scale, a point  $\frac{1}{2}$  LSB above the last code transition (111 . . . 110 to 111 . . . 111). The error is expressed as a percentage of full scale.

### No Missing Codes

No missing codes is a measure of the differential nonlinearity of the ADC. The error is expressed in bits and specifies the number of codes (ADC results) as  $2N$  bits, where  $N$  is no missing codes guaranteed to occur through the full ADC input range.

### Offset Error

Offset error is the deviation of the first code transition ADC input voltage from the ideal first code transition.

### Offset Error Drift

Offset error drift is the variation in absolute offset error with respect to temperature. This error is expressed as least significant bits per degree Celsius.

### Gain Error

Gain error is a measure of the span error of the ADC. It is a measure of the difference between the measured and the ideal span between any two points in the transfer function.

### Output Noise

The output noise is specified as the standard deviation (or  $1 \times$  Sigma) of the distribution of the ADC output codes collected when the ADC input voltage is at a dc voltage. It is expressed as micro root mean square. The output, or root mean square (rms) noise, can be used to calculate the effective resolution of the ADC as defined by the following equation:

$$\text{Effective Resolution} = \log_2(\text{Full-Scale Range}/\text{rms Noise}) \text{ bits}$$

The peak-to-peak noise is defined as the deviation of codes that fall within  $6.6 \times$  Sigma of the distribution of ADC output codes collected when the ADC input voltage is at dc. The peak-to-peak noise is, therefore, calculated as

$$6.6 \times \text{rms Noise}$$

The peak-to-peak noise can be used to calculate the ADC (noise free code) resolution for which there is no code flicker within a 6.6-Sigma limit as defined by the following equation:

$$\text{Noise Free Code Resolution} = \log_2 \left( \frac{\text{Full-Scale Range}}{\text{Peak-to-Peak Noise}} \right) \text{ bits}$$

### Data Sheet Acronyms

|      |                             |
|------|-----------------------------|
| ADC  | analog-to-digital converter |
| ARM  | advanced RISC machine       |
| JTAG | joint test action group     |
| LSB  | least significant byte/bit  |
| LVF  | low voltage flag            |
| MCU  | microcontroller             |
| MMR  | memory mapped register      |
| MSB  | most significant byte/bit   |
| PID  | protected identifier        |
| POR  | power-on reset              |
| PSM  | power supply monitor        |
| rms  | root mean square            |

## OVERVIEW OF THE ARM7TDMI CORE

The ARM7<sup>®</sup> core is a 32-bit, reduced instruction set computer (RISC), developed by ARM<sup>®</sup> Ltd. The ARM7TDMI is a von Neumann-based architecture, meaning that it uses a single 32-bit bus for instruction and data. The length of the data can be 8, 16, or 32 bits, and the length of the instruction word is either 16 bits or 32 bits, depending on the mode in which the core is operating.

The ARM7TDMI is an ARM7 core with four additional features, as listed in Table 10.

**Table 10. ARM7TDMI Features**

| Feature | Description   |
|---------|---|
| T       | Support for the Thumb <sup>®</sup> (16-bit) instruction set                       |
| D       | Support for debug   |
| M       | Enhanced multiplier   |
| I       | Includes the EmbeddedICE <sup>®</sup> module to support embedded system debugging |

### THUMB MODE (T)

An ARM instruction is 32 bits long. The ARM7TDMI processor supports a second instruction set compressed into 16 bits, the Thumb instruction set. Faster code execution from 16-bit memory and greater code density is achieved by using the Thumb instruction set, making the ARM7TDMI core particularly suited for embedded applications.

However, the Thumb mode has three limitations.

- Relative to ARM, the Thumb code usually requires more instructions to perform the same task. Therefore, ARM code is best for maximizing the performance of time-critical code in most applications.
- The Thumb instruction set does not include some instructions that are needed for exception handling, so ARM code can be required for exception handling.
- When an interrupt occurs, the core vectors to the interrupt location in memory and executes the code present at that address. The first command is required to be in ARM code.

### MULTIPLIER (M)

The ARM7TDMI instruction set includes an enhanced multiplier, with four extra instructions to perform 32-bit by 32-bit multiplication with a 64-bit result, and 32-bit by 32-bit multiplication-accumulation (MAC) with a 64-bit result.

### EmbeddedICE (I)

The EmbeddedICE module provides integrated on-chip debug support for the ARM7TDMI. The EmbeddedICE module contains the breakpoint and watchpoint registers that allow nonintrusive user code debugging. These registers are controlled through the JTAG test port. When a breakpoint or watchpoint is encountered, the processor halts and enters the debug state. When in a debug state, the processor registers can be interrogated, as can the Flash/EE, SRAM, and memory mapped registers.

### ARM7 Exceptions

The ARM7 supports five types of exceptions, with a privileged processing mode associated with each type. The five types of exceptions are as follows:

Type 1: normal interrupt or IRQ. This is provided to service general-purpose interrupt handling of internal and external events. Note that the ADuC7060/ADuC7061 supports eight configurable priority levels for all IRQ sources.

Type 2: fast interrupt or FIQ. This is provided to service data transfer or a communication channel with low latency. FIQ has priority over IRQ. Note that the ADuC7060/ADuC7061 supports eight configurable priority levels for all FIQ sources.

Type 3: memory abort (prefetch and data).

Type 4: attempted execution of an undefined instruction.

Type 5: software interrupts (SWI) instruction that can be used to make a call to an operating system.

Typically, the programmer defines interrupts as IRQ, but for higher priority interrupts, the programmer can define interrupts as the FIQ type.

The priority of these exceptions and vector addresses are listed in Table 11.

**Table 11. Exception Priorities and Vector Addresses**

| Priority | Exception                          | Address |
|----------|------------------------------------|---------|
| 1        | Hardware reset                     | 0x00    |
| 2        | Memory abort (data)                | 0x10    |
| 3        | FIQ                                | 0x1C    |
| 4        | IRQ                                | 0x18    |
| 5        | Memory abort (prefetch)            | 0x0C    |
| 6        | Software interrupt <sup>1</sup>    | 0x08    |
| 6        | Undefined instruction <sup>1</sup> | 0x04    |

<sup>1</sup> A software interrupt and an undefined instruction exception have the same priority and are mutually exclusive.

The exceptions listed in Table 11 are located from 0x00 to 0x1C, with a reserved location at 0x14.

### ARM REGISTERS

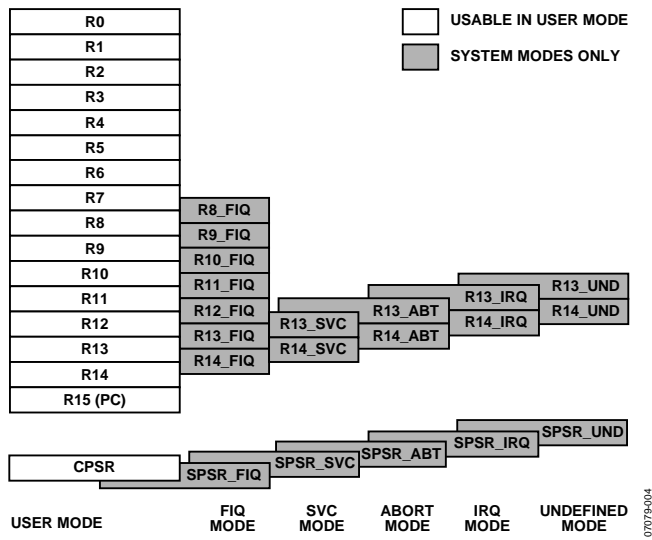
The ARM7TDMI has 16 standard registers. R0 to R12 are for data manipulation, R13 is the stack pointer, R14 is the link register, and R15 is the program counter that indicates the instruction currently being executed. The link register contains the address from which the user has branched (when using the branch and link command) or the command during which an exception occurred.

The stack pointer contains the current location of the stack. Generally, on an ARM7TDMI, the stack starts at the top of the available RAM area and descends using the area as required. A separate stack is defined for each of the exceptions. The size of each stack is user configurable and is dependent on the target application. When programming using high level languages,

such as C, it is necessary to ensure that the stack does not overflow. This is dependent on the performance of the compiler that is used.

When an exception occurs, some of the standard registers are replaced with registers specific to the exception mode. All exception modes have replacement banked registers for the stack pointer (R13) and the link register (R14) as represented in Figure 9. The FIQ mode has more registers (R8 to R12) supporting faster interrupt processing. With the increased number of noncritical registers, the interrupt can be processed without the need to save or restore these registers, thereby reducing the response time of the interrupt handling process.

More information relative to the programmer’s model and the ARM7TDMI core architecture can be found in ARM7TDMI technical and ARM architecture manuals available directly from ARM Ltd.



**INTERRUPT LATENCY**

The worst-case latency for an FIQ consists of the longest time that the request can take to pass through the synchronizer, plus the time for the longest instruction to complete (the longest instruction is an LDM) that loads all the registers including the PC, plus the time for the data abort entry, plus the time for FIQ entry. At the end of this time, the ARM7TDMI is executing the instruction at 0x1C (FIQ interrupt vector address). The maximum total time is 50 processor cycles, or just over 4.88 μs in a system using a continuous 10.24 MHz processor clock. The maximum IRQ latency calculation is similar but must allow for the FIQ having higher priority, which can delay entry into the IRQ handling routine for an arbitrary length of time. This time can be reduced to 42 cycles if the LDM command is not used; some compilers have an option to compile without using this command. Another option is to run the part in Thumb mode where this time is reduced to 22 cycles.

The minimum latency for FIQ or IRQ interrupts is five cycles. This consists of the shortest time that the request can take through the synchronizer plus the time to enter the exception mode.

Note that the ARM7TDMI initially (first instruction) runs in ARM (32-bit) mode when an exception occurs. The user can immediately switch from ARM mode to Thumb mode if required, for example, when executing interrupt service routines.

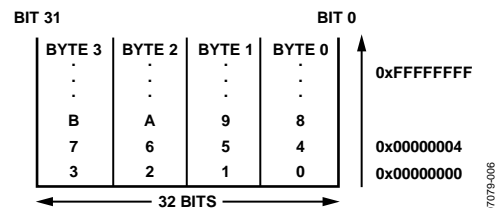
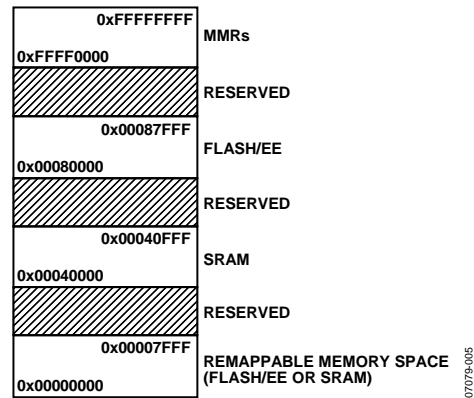
**MEMORY ORGANIZATION**

The ARM7, a von Neumann architecture MCU core, sees memory as a linear array of 232-byte locations. As shown in Figure 10, the ADuC7060/ADuC7061 maps this into four distinct user areas: a memory area that can be remapped, an SRAM area, a Flash/EE area, and a memory mapped register (MMR) area.

The first 30 kB of this memory space is used as an area into which the on-chip Flash/EE or SRAM can be remapped. Any access, either reading or writing, to an area not defined in the memory map results in a data abort exception.

**Memory Format**

The ADuC7060/ADuC7061 memory organization is configured in little endian format: the least significant byte is located in the lowest byte address and the most significant byte in the highest byte address (see Figure 11).



**SRAM**

The ADuC7060/ADuC7061 features 4 kB of SRAM, organized as 1024 × 32 bits, that is, 1024 words located at 0x40000. The RAM space can be used as data memory as well as volatile program space.

ARM code can run directly from SRAM at full clock speed given that the SRAM array is configured as a 32-bit wide memory array. SRAM is read/writable in 8-, 16-, and 32-bit segments.

## Remap

The ARM exception vectors are all situated at the bottom of the memory array, from Address 0x00000000 to Address 0x00000020.

By default, after a reset, the Flash/EE memory is logically mapped to Address 0x00000000. It is possible to logically remap the SRAM to Address 0x00000000 by setting Bit 0 of the remap MMR located at 0xFFFF0220. To revert Flash/EE to 0x00000000, Bit 0 of remap is cleared.

It is sometimes desirable to remap RAM to 0x00000000 to optimize the interrupt latency of the ADuC7060/ADuC7061 because code can run in full 32-bit ARM mode and at maximum core speed. Note that, when an exception occurs, the core defaults to ARM mode.

## Remap Operation

When a reset occurs on the ADuC7060/ADuC7061, execution starts automatically in the factory programmed internal configuration code. This so-called kernel is hidden and cannot be accessed by user code. If the ADuC7060/ADuC7061 is in normal mode, it executes the power-on configuration routine of the kernel and then jumps to the reset vector, Address 0x00000000, to execute the user's reset exception routine. Because the Flash/EE is mirrored at the bottom of the memory array at reset, the reset routine must always be written in Flash/EE.

The remap command must be executed from the absolute Flash/EE address and not from the mirrored, remapped segment of memory, because this may be replaced by SRAM. If a remap operation is executed while operating code from the mirrored location, pre-fetch/data aborts can occur, or the user can observe abnormal program operation. Any kind of reset logically remaps the Flash/EE memory to the bottom of the memory array.

## Remap Register

Name: Remap  
 Address: 0xFFFF0220  
 Default value: 0x0000  
 Access: Read and write  
 Function: This 8-bit register allows user code to remap either RAM or Flash/EE space into the bottom of the ARM memory space starting at Address 0x00000000.

**Table 12. Remap MMR Bit Designations**

| Bit | Description   |
|-----|---|
| 7:1 | Reserved. These bits are reserved and should be written as 0 by user code.  |
| 0   | Remap bit.<br>Set by user to remap the SRAM to 0x00000000.<br>Cleared automatically after reset to remap the Flash/EE memory to 0x00000000. |

## FLASH/EE CONTROL INTERFACE

Serial and JTAG programming use the Flash/EE control interface, which includes the eight MMRs outlined in this section. Note that the flash page size is 512 bytes.

## FEESTA Register

FEESTA is a read-only register that reflects the status of the flash control interface as described in Table 13.

Name: FEESTA  
 Address: 0xFFFF0E00  
 Default value: 0x0020  
 Access: Read

**Table 13. FEESTA MMR Bit Designations**

| Bit  | Description  |
|------|--|
| 15:6 | Reserved.  |
| 5    | Reserved.  |
| 4    | Reserved.  |
| 3    | Flash interrupt status bit. Set automatically when an interrupt occurs, that is, when a command is complete and the Flash/EE interrupt enable bit in the FEESTA register is set. Cleared when reading the FEESTA register. |
| 2    | Flash/EE controller busy. Set automatically when the controller is busy. Cleared automatically when the controller is not busy.  |
| 1    | Command fail. Set automatically when a command completes unsuccessfully. Cleared automatically when reading the FEESTA register.   |
| 0    | Command pass. Set by the MicroConverter® when a command completes successfully. Cleared automatically when reading the FEESTA register.  |

## FEEMOD Register

FEEMOD sets the operating mode of the flash control interface.

Table 14 lists FEEMOD MMR bit designations.

Name: FEEMOD  
 Address: 0xFFFF0E04  
 Default value: 0x0000  
 Access: Read and write



Table 14. FEEMOD MMR Bit Designations

| Bit  | Description  |
|------|--|
| 15:9 | Reserved.  |
| 8    | Reserved. Always set this bit to 1.  |
| 7:5  | Reserved. Always set these bits to 0 except when writing keys.   |
| 4    | Flash/EE interrupt enable.<br>Set by user to enable the Flash/EE interrupt. The interrupt occurs when a command is complete.<br>Cleared by user to disable the Flash/EE interrupt. |
| 3    | Erase/write command protection.<br>Set by user to enable the erase and write commands.<br>Cleared to protect the Flash/EE against the erase/write command.                         |
| 2:0  | Reserved. Always set these bits to 0.  |

**FEECON Register**

FEECON is an 8-bit command register. The commands are described in Table 15.

|                |                |
|----------------|----------------|
| Name:          | FEECON         |
| Address:       | 0xFFFFF0E08    |
| Default value: | 0x07           |
| Access:        | Read and write |

Table 15. Command Codes in FEECON

| Code              | Command       | Description  |
|-------------------|---------------|--|
| 0x00 <sup>1</sup> | Null          | Idle state.  |
| 0x01 <sup>1</sup> | Single read   | Load FEEDAT with the 16-bit data. Indexed by FEEADR.   |
| 0x02 <sup>1</sup> | Single write  | Write FEEDAT at the address pointed to by FEEADR. This operation takes 50 $\mu$ s.   |
| 0x03 <sup>1</sup> | Erase/write   | Erase the page indexed by FEEADR and write FEEDAT at the location pointed to by FEEADR. This operation takes approximately 24 ms.  |
| 0x04 <sup>1</sup> | Single verify | Compare the contents of the location pointed to by FEEADR to the data in FEEDAT. The result of the comparison is returned in FEESTA Bit 0 and Bit 1.   |
| 0x05 <sup>1</sup> | Single erase  | Erase the page indexed by FEEADR.  |
| 0x06 <sup>1</sup> | Mass erase    | Erase 30 kB of user space. The 2 kB of kernel are protected. To prevent accidental execution, a command sequence is required to execute this instruction. See the Command Sequence for Executing a Mass Erase section. |
| 0x07              | Reserved      | Reserved.  |
| 0x08              | Reserved      | Reserved.  |
| 0x09              | Reserved      | Reserved.  |
| 0x0A              | Reserved      | Reserved.  |
| 0x0B              | Signature     | This command results in a 24-bit LFSR-based signature being generated and loaded into the FEESIG MMR. This operation takes 16,389 clock cycles.  |
| 0x0C              | Protect       | This command can run only once. The value of FEEPRO is saved and is removed only with a mass erase (0x06) or the key.  |
| 0x0D              | Reserved      | Reserved.  |
| 0x0E              | Reserved      | Reserved.  |
| 0x0F              | Ping          | No operation; interrupt generated.   |

<sup>1</sup> The FEECON register always reads 0x07 immediately after execution of any of these commands.

**FEEDAT Register**

FEEDAT is a 16-bit data register. This register holds the data value for flash read and write commands.

Name: FEEDAT  
 Address: 0xFFFFF0E0C  
 Default value: 0XXXXX  
 Access: Read and write

**FEEADR Register**

FEEADR is a 16-bit address register used for accessing individual pages of the 32 kB flash block. The valid address range for a user is: 0x0000 to 0x77FF. This represents the 30 kB flash user memory space. A read or write access outside this boundary causes a data abort exception to occur.

Name: FEEADR  
 Address: 0xFFFFF0E10  
 Default value: 0x0000  
 Access: Read and write

**FEESIG Register**

The FEESIG register is a 24-bit MMR. This register is updated with the 24-bit signature value after the signature command is executed. This value is the result of the linear feedback shift register (LFSR) operation initiated by the signature command.

Name: FEESIG  
 Address: 0xFFFFF0E18  
 Default value: 0FFFFFFF  
 Access: Read

**FEEPRO Register**

The FEEPRO MMR provides protection following a subsequent reset of the MMR. It requires a software key (see Table 16).

Name: FEEPRO  
 Address: 0xFFFFF0E1C  
 Default value: 0x00000000  
 Access: Read and write

**FEEHID Register**

The FEEHID MMR provides immediate protection. It does not require any software key. Note that the protection settings in FEEHID are cleared by a reset (see Table 16).

Name: FEEHID  
 Address: 0xFFFFF0E20  
 Default value: 0xFFFFFFFF  
 Access: Read and write

**Table 16. FEEPRO and FEEHID MMR Bit Designations**

| Bit  | Description  |
|------|--|
| 31   | Read protection.<br>Cleared by user to protect all code. No JTAG read accesses for protected pages if this bit is cleared.<br>Set by the user to allow reading the code via JTAG.  |
| 30   | Protection for Page 59 (0x00087600 to 0x000877FF).<br>Set by the user to allow writing to Page 59. Cleared to protect Page 59.   |
| 29   | Protection for Page 58 (0x00087400 to 0x000875FF).<br>Set by the user to allow writing to Page 58. Cleared to protect Page 58.   |
| 28:0 | Write protection for Page 57 to Page 0. Each bit represents two pages. Each page is 512 bytes in size.<br>Bit 0 is protection for Page 0 and Page 1 (0x00080000 to 0x000803FF). Set by the user to allow writing Page 0 and Page 1. Cleared to protect Page 0 and Page 1.<br>Bit 1 is protection for Page 2 and Page 3 (0x00080400 to 0x000807FF). Set by the user to allow writing Page 2 and Page 3. Cleared to protect Page 2 and Page 3.<br>...<br>...<br>Bit 27 is protection for page 54 and page 55 (0x86C00 to 0x86FFF). Set by the user to allow writing to Page 54 and Page 55. Cleared to protect Page 54 and Page 55.<br>Bit 28 is protection for page 56 and page 57 (0x87000 to 0x873FF). Set by the user to allow writing to Page 56 and Page 57. Cleared to protect Page 56 and Page 57. |

**Temporary Protection**

Temporary protection can be set and removed by writing directly into the FEEHID MMR. This register is volatile and, therefore, protection is only in place for as long as the part remains powered on. The protection setting is not reloaded after a power cycle.

**Keyed Permanent Protection**

Keyed permanent protection can be set via FEEPRO to lock the protection configuration. The software key used at the start of the required FEEPRO write sequence is saved one time only and must be used for any subsequent access of the FEEHID or FEEPRO MMRs. A mass erase sets the software protection key back to 0xFFFF but also erases the entire user code space.

**Permanent Protection**

Permanent protection can be set via FEEPRO, similar to how keyed permanent protection is set, with the only difference being that the software key used is 0xDEADDEAD. When the FEEPRO write sequence is saved, only a mass erase sets the software protection key back to 0xFFFFFFFF. This also erases the entire user code space.

**Sequence to Write the Software Protection Key and Set Permanent Protection**

1. Write in FEEPRO corresponding to the pages to be protected.
2. Write the new (user-defined) 32-bit software protection key in FEEADR (Bits[31:16]) and FEEDAT (Bits[15:0]).
3. Write 10 in FEEMOD (Bits[6:5]) and set FEEMOD (Bit 3).
4. Run the protect command (Code 0x0C) in FEECON.

To remove or modify the protection, the same sequence can be used with a modified value of FEEPRO.

The previous sequence for writing the key and setting permanent protection is illustrated in the following example, this protects writing Page 4 and Page 5 of the Flash/EE:

```
Int a = FEESTA;           // Ensure FEESTA
is cleared
FEEPRO = 0xFFFFFFFFB;    // Protect Page 4
and Page 5
FEEADR = 0x66BB;         // 32-bit key
value (Bits[31:16])
FEEDAT = 0xAA55;         // 32-bit key
value (Bits[15:0])
FEEMOD = 0x0048          // Lock security
sequence
FEECON = 0x0C;           // Write key
command
while (FEESTA & 0x04){}  // Wait for
command to finish
```

**Command Sequence for Executing a Mass Erase**

```
FEEDAT = 0x3CFF;
FEEADR = 0xFFC3;
FEEMOD = FEEMOD|0x8;     //Erase key enable
FEECON = 0x06;           //Mass erase command
```

**MEMORY MAPPED REGISTERS**

The memory mapped register (MMR) space is mapped into the upper two pages of the memory array and is accessed by indirect addressing through the ARM7 banked registers.

The MMR space provides an interface between the CPU and all on-chip peripherals. All registers, except the core registers, reside in the MMR area. All shaded locations shown in Figure 12 are unoccupied or reserved locations and should not be accessed by user software. Figure 12 shows the full MMR memory map.

The access time for reading from or writing to an MMR depends on the advanced microcontroller bus architecture (AMBA) bus used to access the peripheral. The processor has two AMBA buses: the advanced high performance bus (AHB) used for system modules and the advanced peripheral bus (APB) used for a lower performance peripheral. Access to the AHB is one cycle, and access to the APB is two cycles. All peripherals on the ADuC7060/ADuC7061 are on the APB except for the Flash/EE memory, the GPIOs, and the PWM.

|            |                                |
|------------|--------------------------------|
| 0xFFFFFFFF |                                |
| 0xFFFF0FC0 |                                |
| 0xFFFF0F80 | PWM                            |
| 0xFFFF0E24 | FLASH CONTROL INTERFACE        |
| 0xFFFF0E00 |                                |
| 0xFFFF0D50 |                                |
| 0xFFFF0D00 | GPIO                           |
| 0xFFFF0A14 |                                |
| 0xFFFF0A00 | SPI                            |
| 0xFFFF0948 |                                |
| 0xFFFF0900 | I <sup>2</sup> C               |
| 0xFFFF0730 |                                |
| 0xFFFF0700 | UART                           |
| 0xFFFF0620 |                                |
| 0xFFFF0600 | DAC                            |
| 0xFFFF0570 |                                |
| 0xFFFF0500 | ADC                            |
| 0xFFFF0490 |                                |
| 0xFFFF048C | BAND GAP REFERENCE             |
| 0xFFFF0470 |                                |
| 0xFFFF0450 | SPI/I <sup>2</sup> C SELECTION |
| 0xFFFF0420 |                                |
| 0xFFFF0404 | PLL AND OSCILLATOR CONTROL     |
| 0xFFFF0394 |                                |
| 0xFFFF0380 | GENERAL-PURPOSE TIMER          |
| 0xFFFF0370 |                                |
| 0xFFFF0360 | WATCHDOG TIMER                 |
| 0xFFFF0350 |                                |
| 0xFFFF0340 | WAKE-UP TIMER                  |
| 0xFFFF0334 |                                |
| 0xFFFF0320 | GENERAL-PURPOSE TIMER          |
| 0xFFFF0238 |                                |
| 0xFFFF0220 | REMAP AND SYSTEM CONTROL       |
| 0xFFFF0140 |                                |
| 0xFFFF0000 | INTERRUPT CONTROLLER           |

07079-007

Figure 12. Memory Mapped Registers

**COMPLETE MMR LISTING**

In the following MMR tables, addresses are listed in hexadecimal code. Access types include R for read, W for write, and R/W for read and write.

**Table 17. IRQ Address Base = 0xFFFF0000**

| Address | Name    | Bytes | Access Type | Default Value | Description  |
|---------|---------|-------|-------------|---------------|--|
| 0x0000  | IRQSTA  | 4     | R           | 0x00000000    | Active IRQ source status.  |
| 0x0004  | IRQSIG  | 4     | R           |               | Current state of all IRQ sources (enabled and disabled).   |
| 0x0008  | IRQEN   | 4     | R/W         | 0x00000000    | Enabled IRQ sources.   |
| 0x000C  | IRQCLR  | 4     | W           | 0x00000000    | MMR to disable IRQ sources.  |
| 0x0010  | SWICFG  | 4     | W           | 0x00000000    | Software interrupt configuration MMR.  |
| 0x0014  | IRQBASE | 4     | R/W         | 0x00000000    | Base address of all vectors. Points to the start of the 64-byte memory block, which can contain up to 32 pointers to separate subroutine handlers. |
| 0x001C  | IRQVEC  | 4     | R           | 0x00000000    | This register contains the subroutine address for the currently active IRQ source.   |
| 0x0020  | IRQP0   | 4     | R/W         | 0x00000000    | Contains the interrupt priority setting for Interrupt Source 1 to Interrupt Source 7. An interrupt can have a priority setting of 0 to 7.          |
| 0x0024  | IRQP1   | 4     | R/W         | 0x00000000    | Contains the interrupt priority setting for Interrupt Source 8 to Interrupt Source 15.   |
| 0x0028  | IRQP2   | 4     | R/W         | 0x00000000    | Contains the interrupt priority setting for Interrupt Source 16 to Interrupt Source 19.  |
| 0x0030  | IRQCONN | 4     | R/W         | 0x00000000    | Used to enable IRQ and FIQ interrupt nesting.  |
| 0x0034  | IRQCONE | 4     | R/W         | 0x00000000    | Configures the external interrupt sources as rising edge, falling edge, or level triggered.  |
| 0x0038  | IRQCLRE | 4     | R/W         | 0x00000000    | Used to clear an edge-level-triggered interrupt source.  |
| 0x003C  | IRQSTAN | 4     | R/W         | 0x00000000    | This register indicates the priority level of an interrupt that has just caused an interrupt exception.  |
| 0x0100  | FIQSTA  | 4     | R           | 0x00000000    | Active FIQ source status.  |
| 0x0104  | FIQSIG  | 4     | R           |               | Current state of all FIQ sources (enabled and disabled).   |
| 0x0108  | FIQEN   | 4     | R/W         | 0x00000000    | Enabled FIQ sources.   |
| 0x010C  | FIQCLR  | 4     | W           | 0x00000000    | MMR to disable FIQ sources.  |
| 0x011C  | FIQVEC  | 4     | R           | 0x00000000    | This register contains the subroutine address for the currently active FIQ source.   |
| 0x013C  | FIQSTAN | 4     | R/W         | 0x00000000    | Indicates the priority level of an FIQ that has just caused an FIQ exception.  |

**Table 18. System Control Address Base = 0xFFFF0200**

| Address | Name               | Bytes | Access Type | Default Value | Description  |
|---------|--------------------|-------|-------------|---------------|--|
| 0x0220  | REMAP <sup>1</sup> | 1     | R/W         | 0x00          | Remap control register. See the Remap Operation section. |
| 0x0230  | RSTSTA             | 1     | R/W         | 0x01          | RSTSTA status MMR. See the Reset section.                |
| 0x0234  | RSTCLR             | 1     | W           | 0x00          | Register for clearing the RSTSTA register.               |

<sup>1</sup>Updated by the kernel.

Table 19. Timer Address Base = 0xFFFF0300

| Address | Name   | Bytes | Access Type | Default Value | Description                      |
|---------|--------|-------|-------------|---------------|----------------------------------|
| 0x0320  | T0LD   | 4     | R/W         | 0x00000000    | Timer0 load register.            |
| 0x0324  | T0VAL  | 4     | R           | 0xFFFFFFFF    | Timer0 value register.           |
| 0x0328  | T0CON  | 4     | R/W         | 0x01000000    | Timer0 control MMR.              |
| 0x032C  | T0CLR1 | 1     | W           | N/A           | Timer0 interrupt clear register. |
| 0x0330  | T0CAP  | 4     | R           | 0x00000000    | Timer0 capture register.         |
| 0x0340  | T1LD   | 4     | R/W         | 0x00000000    | Timer1 load register.            |
| 0x0344  | T1VAL  | 4     | R           | 0xFFFFFFFF    | Timer1 value register.           |
| 0x0348  | T1CON  | 2     | R/W         | 0x0000        | Timer1 control MMR.              |
| 0x034C  | T1CLR1 | 1     | W           | N/A           | Timer1 interrupt clear register. |
| 0x0360  | T2LD   | 2     | R/W         | 0x3BF8        | Timer2 load register.            |
| 0x0364  | T2VAL  | 2     | R           | 0x3BF8        | Timer2 value register.           |
| 0x0368  | T2CON  | 2     | R/W         | 0x0000        | Timer2 control MMR.              |
| 0x036C  | T2CLR1 | 1     | W           | N/A           | Timer2 interrupt clear register. |
| 0x0380  | T3LD   | 2     | R/W         | 0x0000        | Timer3 load register.            |
| 0x0384  | T3VAL  | 2     | R           | 0xFFFF        | Timer3 value register.           |
| 0x0388  | T3CON  | 4     | R/W         | 0x00000000    | Timer3 control MMR.              |
| 0x038C  | T3CLR1 | 1     | W           | N/A           | Timer3 interrupt clear register. |
| 0x0390  | T3CAP  | 2     | R           | 0x0000        | Timer3 capture register.         |

Table 20. PLL Base Address = 0xFFFF0400

| Address | Name    | Bytes | Access Type | Default Value | Description  |
|---------|---------|-------|-------------|---------------|--|
| 0x0404  | POWKEY1 | 2     | W           | 0xFFFF        | POWCON0 prewrite key.  |
| 0x0408  | POWCON0 | 1     | R/W         | 0x7B          | Power control and core speed control register.   |
| 0x040C  | POWKEY2 | 2     | W           | 0xFFFF        | POWCON0 postwrite key.   |
| 0x0410  | PLLKEY1 | 2     | W           | 0xFFFF        | PLLCON prewrite key.   |
| 0x0414  | PLLCON  | 1     | R/W         | 0x00          | PLL clock source selection MMR.  |
| 0x0418  | PLLKEY2 | 2     | W           | 0xFFFF        | PLLCON postwrite key.  |
| 0x0434  | POWKEY3 | 2     | W           | 0xFFFF        | POWCON1 prewrite key.  |
| 0x0438  | POWCON1 | 2     | R/W         | 0x124         | Power control register.  |
| 0x043C  | POWKEY4 | 2     | W           | 0xFFFF        | POWCON1 postwrite key.   |
| 0x0464  | GP0KEY1 | 2     | W           | 0xFFFF        | GP0CON1 prewrite key.  |
| 0x0468  | GP0CON1 | 1     | R/W         | 0x00          | Configures P0.0, P0.1, P0.2, and P0.3 as analog inputs or digital I/Os. Also enables SPI or I <sup>2</sup> C mode. |
| 0x046C  | GP0KEY2 | 2     | W           | 0xFFFF        | GP0CON1 postwrite key.   |

Table 21. ADC Address Base = 0xFFFF0500

| Address | Name                | Bytes | Access Type | Default Value                             | Description   |
|---------|---------------------|-------|-------------|---|---|
| 0x0500  | ADCSTA              | 2     | R           | 0x0000                                    | ADC status MMR.   |
| 0x0504  | ADCMSKI             | 2     | R/W         | 0x0000                                    | ADC interrupt source enable MMR.  |
| 0x0508  | ADCMDDE             | 1     | R/W         | 0x03                                      | ADC mode register.  |
| 0x050C  | ADC0CON             | 2     | R/W         | 0x8000                                    | Primary ADC control MMR.  |
| 0x0510  | ADC1CON             | 2     | R/W         | 0x0000                                    | Auxiliary ADC control MMR.  |
| 0x0514  | ADCFLT              | 2     | R/W         | 0x0007                                    | ADC filter control MMR.   |
| 0x0518  | ADCCFG              | 1     | R/W         | 0x00                                      | ADC configuration MMR.  |
| 0x051C  | ADC0DAT             | 4     | R           | 0x00000000                                | Primary ADC result MMR.   |
| 0x0520  | ADC1DAT             | 4     | R           | 0x00000000                                | Auxiliary ADC result MMR  |
| 0x0524  | ADC0OF <sup>1</sup> | 2     | R/W         | 0x0000, part specific, factory programmed | Primary ADC offset calibration setting.   |
| 0x0528  | ADC1OF <sup>1</sup> | 2     | R/W         | 0x0000, part specific, factory programmed | Auxiliary ADC offset MMR.   |
| 0x052C  | ADC0GN <sup>1</sup> | 2     | R/W         | 0x5555                                    | Primary ADC offset MMR.   |
| 0x0530  | ADC1GN <sup>1</sup> | 2     | R/W         | 0x5555                                    | Auxiliary ADC offset MMR. See the ADC operation mode configuration bit (ADCLPMCFCG[1:0]) in Table 42. |
| 0x0534  | ADC0RCR             | 2     | R/W         | 0x0001                                    | Primary ADC result counter/reload MMR.  |
| 0x0538  | ADC0RCV             | 2     | R           | 0x0000                                    | Primary ADC result counter MMR.   |
| 0x053C  | ADC0TH              | 2     | R/W         | 0x0000                                    | Primary ADC 16-bit comparator threshold MMR.  |
| 0x0540  | ADC0THC             | 2     | R/W         | 0x0001                                    | Primary ADC 16-bit comparator threshold counter limit.  |
| 0x0544  | ADC0THV             | 2     | R           | 0x0000                                    | ADC0 8-bit threshold exceeded counter register  |
| 0x0548  | ADC0ACC             | 4     | R           | 0x00000000                                | Primary ADC accumulator.  |
| 0x054C  | ADC0ATH             | 4     | R/W         | 0x00000000                                | Primary ADC 32-bit comparator threshold MMR.  |
| 0x0570  | IEXCON              | 1     | R/W         | 0x00                                      | Excitation current sources control register.  |

<sup>1</sup> Updated by the kernel to part specific calibration value.

Table 22. DAC Control Address Base = 0xFFFF0600

| Address | Name    | Bytes | Access Type | Default Value | Description               |
|---------|---------|-------|-------------|---------------|---------------------------|
| 0x0600  | DAC0CON | 2     | R/W         | 0x0200        | DAC control register.     |
| 0x0604  | DAC0DAT | 4     | R/W         | 0x00000000    | DAC output data register. |

Table 23. UART Base Address = 0xFFFF0700

| Address | Name    | Bytes | Access Type | Default Value | Description  |
|---------|---------|-------|-------------|---------------|--|
| 0x0700  | COMTX   | 1     | W           | N/A           | UART transmit register.                            |
| 0x0700  | COMRX   | 1     | R           | 0x00          | UART receive register.                             |
| 0x0700  | COMDIV0 | 1     | R/W         | 0x00          | UART Standard Baud Rate Generator Divisor Value 0. |
| 0x0704  | COMIEN0 | 1     | R/W         | 0x00          | UART Interrupt Enable MMR 0.                       |
| 0x0704  | COMDIV1 | 1     | R/W         | 0x00          | UART Standard Baud Rate Generator Divisor Value 1. |
| 0x0708  | COMIID0 | 1     | R           | 0x01          | UART Interrupt Identification 0.                   |
| 0x070C  | COMCON0 | 1     | R/W         | 0x00          | UART Control Register 0.                           |
| 0x0710  | COMCON1 | 1     | R/W         | 0x00          | UART Control Register 1.                           |
| 0x0714  | COMSTA0 | 1     | R           | 0x60          | UART Status Register 0.                            |
| 0x0718  | COMSTA1 | 1     | R           | 0x00          | UART Status Register 1.                            |
| 0x072C  | COMDIV2 | 2     | R/W         | 0x0000        | UART fractional divider MMR.                       |

Table 24. I<sup>2</sup>C Base Address = 0xFFFF0900

| Address | Name     | Bytes | Access Type | Default Value | Description   |
|---------|----------|-------|-------------|---------------|---|
| 0x0900  | I2CMCON  | 2     | R/W         | 0x0000        | I <sup>2</sup> C master control register.   |
| 0x0904  | I2CMSTA  | 2     | R           | 0x0000        | I <sup>2</sup> C master status register.  |
| 0x0908  | I2CMRX   | 1     | R           | 0x00          | I <sup>2</sup> C master receive register.   |
| 0x090C  | I2CMTX   | 1     | W           | 0x00          | I <sup>2</sup> C master transmit register.  |
| 0x0910  | I2CMCNT0 | 2     | R/W         | 0x0000        | I <sup>2</sup> C master read count register. Write the number of required bytes into this register prior to reading from a slave device.            |
| 0x0914  | I2CMCNT1 | 1     | R           | 0x00          | I <sup>2</sup> C master current read count register. This register contains the number of bytes already received during a read from slave sequence. |
| 0x0918  | I2CADR0  | 1     | R/W         | 0x00          | Address byte register. Write the required slave address here prior to communications.   |
| 0x091C  | I2CADR1  | 1     | R/W         | 0x00          | Address byte register. Write the required slave address here prior to communications. Only used in 10-bit mode.                                     |
| 0x0924  | I2CDIV   | 2     | R/W         | 0x1F1F        | I <sup>2</sup> C clock control register. Used to configure the SCLK frequency.  |
| 0x0928  | I2CSCON  | 2     | R/W         | 0x0000        | I <sup>2</sup> C slave control register.  |
| 0x092C  | I2CSSTA  | 2     | R/W         | 0x0000        | I <sup>2</sup> C slave status register.   |
| 0x0930  | I2CSRX   | 1     | R           | 0x00          | I <sup>2</sup> C slave receive register.  |
| 0x0934  | I2CSTX   | 1     | W           | 0x00          | I <sup>2</sup> C slave transmit register.   |
| 0x0938  | I2CALT   | 1     | R/W         | 0x00          | I <sup>2</sup> C hardware general call recognition register.  |
| 0x093C  | I2CID0   | 1     | R/W         | 0x00          | I <sup>2</sup> C Slave ID0 register. Slave bus ID register.   |
| 0x0940  | I2CID1   | 1     | R/W         | 0x00          | I <sup>2</sup> C Slave ID1 register. Slave bus ID register.   |
| 0x0944  | I2CID2   | 1     | R/W         | 0x00          | I <sup>2</sup> C Slave ID2 register. Slave bus ID register.   |
| 0x0948  | I2CID3   | 1     | R/W         | 0x00          | I <sup>2</sup> C Slave ID3 register. Slave bus ID register.   |
| 0x094C  | I2CFSTA  | 2     | R/W         | 0x0000        | I <sup>2</sup> C FIFO status register. Used in both master and slave modes.   |

Table 25. SPI Base Address = 0xFFFF0A00

| Address | Name   | Bytes | Access Type | Default Value | Description               |
|---------|--------|-------|-------------|---------------|---------------------------|
| 0x0A00  | SPISTA | 4     | R           | 0x00000000    | SPI status MMR.           |
| 0x0A04  | SPIRX  | 1     | R           | 0x00          | SPI receive MMR.          |
| 0x0A08  | SPITX  | 1     | W           | 0x00          | SPI transmit MMR.         |
| 0x0A0C  | SPIDIV | 1     | W           | 0x1B          | SPI baud rate select MMR. |
| 0x0A10  | SPICON | 2     | R/W         | 0x0000        | SPI control MMR.          |

Table 26. GPIO Base Address = 0xFFFF0D00

| Address | Name    | Bytes | Access Type | Default Value | Description                      |
|---------|---------|-------|-------------|---------------|----------------------------------|
| 0x0D00  | GP0CON0 | 4     | R/W         | 0x00000000    | GPIO Port 0 control MMR.         |
| 0x0D04  | GP1CON  | 4     | R/W         | 0x00000000    | GPIO Port 1 control MMR.         |
| 0x0D08  | GP2CON  | 4     | R/W         | 0x00000000    | GPIO Port 2 control MMR.         |
| 0x0D20  | GP0DAT  | 4     | R/W         | 0x000000XX    | GPIO Port 0 data control MMR.    |
| 0x0D24  | GP0SET  | 4     | W           | 0x000000XX    | GPIO Port 0 data set MMR.        |
| 0x0D28  | GP0CLR  | 4     | W           | 0x000000XX    | GPIO Port 0 data clear MMR.      |
| 0x0D2C  | GP0PAR  | 4     | R/W         | 0x00000000    | GPIO Port 0 pull-up disable MMR. |
| 0x0D30  | GP1DAT  | 4     | R/W         | 0x000000XX    | GPIO Port 1 data control MMR.    |
| 0x0D34  | GP1SET  | 4     | W           | 0x000000XX    | GPIO Port 1 data set MMR.        |
| 0x0D38  | GP1CLR  | 4     | W           | 0x000000XX    | GPIO Port 1 data clear MMR.      |
| 0x0D3C  | GP1PAR  | 4     | R/W         | 0x00000000    | GPIO Port 1 pull-up disable MMR. |
| 0x0D40  | GP2DAT  | 4     | R/W         | 0x000000XX    | GPIO Port 2 data control MMR.    |
| 0x0D44  | GP2SET  | 4     | W           | 0x000000XX    | GPIO Port 2 data set MMR.        |
| 0x0D48  | GP2CLR  | 4     | W           | 0x000000XX    | GPIO Port 2 data clear MMR.      |
| 0x0D4C  | GP2PAR  | 4     | R/W         | 0x00000000    | GPIO Port 2 pull-up disable MMR. |



Table 27. Flash/EE Base Address = 0xFFFF0E00

| Address | Name   | Bytes | Access Type | Default Value | Description              |
|---------|--------|-------|-------------|---------------|--------------------------|
| 0x0E00  | FEESTA | 2     | R           | 0x20          | Flash/EE status MMR.     |
| 0x0E04  | FEEMOD | 2     | R/W         | 0x0000        | Flash/EE control MMR.    |
| 0x0E08  | FEECON | 1     | R/W         | 0x07          | Flash/EE control MMR.    |
| 0x0E0C  | FEEDAT | 2     | R/W         | 0xFFFF        | Flash/EE data MMR.       |
| 0x0E10  | FEEADR | 2     | R/W         | 0x0000        | Flash/EE address MMR.    |
| 0x0E18  | FEESIG | 3     | R           | 0xFFFFFFFF    | Flash/EE LFSR MMR.       |
| 0x0E1C  | FEEPRO | 4     | R/W         | 0x00000000    | Flash/EE protection MMR. |
| 0x0E20  | FEEHID | 4     | R/W         | 0xFFFFFFFF    | Flash/EE protection MMR. |

Table 28. PWM Base Address = 0xFFFF0F80

| Address | Name     | Bytes | Access Type | Default Value | Description   |
|---------|----------|-------|-------------|---------------|---|
| 0x0F80  | PWMCON   | 2     | R/W         | 0x0012        | PWM control register. See the Pulse-Width Modulator section for full details.                   |
| 0x0F84  | PWM0COM0 | 2     | R/W         | 0x0000        | Compare Register 0 for PWM Output 0 and PWM Output 1.   |
| 0x0F88  | PWM0COM1 | 2     | R/W         | 0x0000        | Compare Register 1 for PWM Output 0 and PWM Output 1.   |
| 0x0F8C  | PWM0COM2 | 2     | R/W         | 0x0000        | Compare Register 2 for PWM Output 0 and PWM Output 1.   |
| 0x0F90  | PWM0LEN  | 2     | R/W         | 0x0000        | Frequency control for PWM Output 0 and PWM Output 1.  |
| 0x0F94  | PWM1COM0 | 2     | R/W         | 0x0000        | Compare Register 0 for PWM Output 2 and PWM Output 3.   |
| 0x0F98  | PWM1COM1 | 2     | R/W         | 0x0000        | Compare Register 1 for PWM Output 2 and PWM Output 3.   |
| 0x0F9C  | PWM1COM2 | 2     | R/W         | 0x0000        | Compare Register 2 for PWM Output 2 and PWM Output 3.   |
| 0x0FA0  | PWM1LEN  | 2     | R/W         | 0x0000        | Frequency control for PWM Output 2 and PWM Output 3.  |
| 0x0FA4  | PWM2COM0 | 2     | R/W         | 0x0000        | Compare Register 0 for PWM Output 4 and PWM Output 5.   |
| 0x0FA8  | PWM2COM1 | 2     | R/W         | 0x0000        | Compare Register 1 for PWM Output 4 and PWM Output 5.   |
| 0x0FAC  | PWM2COM2 | 2     | R/W         | 0x0000        | Compare Register 2 for PWM Output 4 and PWM Output 5.   |
| 0x0FB0  | PWM2LEN  | 2     | R/W         | 0x0000        | Frequency control for PWM Output 4 and PWM Output 5.  |
| 0x0FB8  | PWMCLR1  | 2     | W           | 0x0000        | PWM interrupt clear register. Writing any value to this register clears a PWM interrupt source. |

**RESET**

There are four kinds of resets: external reset, power-on reset, watchdog reset, and software reset. The RSTSTA register indicates the source of the last reset and can be written by user code to initiate a software reset event.

The bits in this register can be cleared to 0 by writing to the RSTCLR MMR at 0xFFFF0234. The bit designations in RSTCLR mirror those of RSTSTA. These registers can be used during a reset exception service routine to identify the source of the reset. The implications of all four kinds of reset events are tabulated in Table 30.

**RSTSTA Register**

Name: RSTSTA  
 Address: 0xFFFF0230  
 Default value: Depends on type of reset  
 Access: Read and write  
 Function: This 8-bit register indicates the source of the last reset event and can be written by user code to initiate a software reset.

**RSTCLR Register**

Name: RSTCLR  
 Address: 0xFFFF0234  
 Access: Write only  
 Function: This 8-bit write only register clears the corresponding bit in RSTSTA.

**Table 29. RSTSTA/RSTCLR MMR Bit Designations**

| Bit | Description  |
|-----|--|
| 7:4 | Not used. These bits are not used and always read as 0.  |
| 3   | External reset.<br>Automatically set to 1 when an external reset occurs.<br>This bit is cleared by setting the corresponding bit in RSTCLR.                        |
| 2   | Software reset.<br>This bit is set to 1 by user code to generate a software reset.<br>This bit is cleared by setting the corresponding bit in RSTCLR. <sup>1</sup> |
| 1   | Watchdog timeout.<br>Automatically set to 1 when a watchdog timeout occurs.<br>Cleared by setting the corresponding bit in RSTCLR.                                 |
| 0   | Power-on reset.<br>Automatically set when a power-on reset occurs.<br>Cleared by setting the corresponding bit in RSTCLR.  |

<sup>1</sup> If the software reset bit in RSTSTA is set, any write to RSTCLR that does not clear this bit generates a software reset.

**Table 30. Device Reset Implications**

| RESET        | Reset External Pins to Default State | Kernel Executed | Reset All External MMRs (Excluding RSTSTA) | Peripherals Reset | Watchdog Timer Reset | RAM Valid | RSTSTA (Status After Reset Event) |
|--------------|--------------------------------------|-----------------|--|-------------------|----------------------|-----------|-----------------------------------|
| POR          | Yes                                  | Yes             | Yes  | Yes               | Yes                  | Yes/No    | RSTSTA[0] = 1                     |
| Watchdog     | Yes                                  | Yes             | Yes  | Yes               | No                   | Yes       | RSTSTA[1] = 1                     |
| Software     | Yes                                  | Yes             | Yes  | Yes               | No                   | Yes       | RSTSTA[2] = 1                     |
| External Pin | Yes                                  | Yes             | Yes  | Yes               | No                   | Yes       | RSTSTA[3] = 1                     |

## OSCILLATOR, PLL, AND POWER CONTROL CLOCKING SYSTEM

The ADuC7060/ADuC7061 integrates a 32.768 kHz  $\pm 3\%$  oscillator, a clock divider, and a PLL. The PLL locks onto a multiple of the internal oscillator or an external 32.768 kHz crystal to provide a stable 10.24 MHz clock (UCLK) for the system. To allow power saving, the core can operate at this frequency or at binary submultiples of it. The actual core operating frequency,  $UCLK/2^{CD}$ , is referred to as HCLK. The default core clock is the PLL clock divided by 8 ( $CD = 3$ ) or 1.28 MHz.

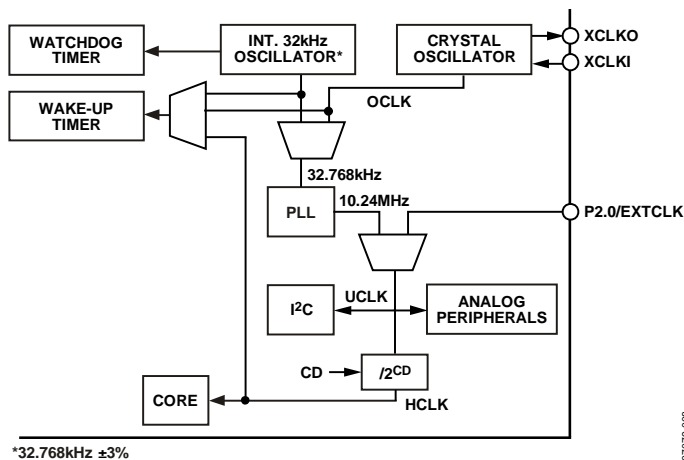


Figure 13. Clocking System

### External Crystal Selection

To switch to an external crystal, users must follow this procedure:

1. Enable the Timer1 interrupt and configure it for a timeout period of  $>120 \mu s$ .
2. Follow the write sequence to the PLLCON register, setting the OSEL bits to [10] and clearing the EXTCLK bit.
3. Force the part into nap mode by following the correct write sequence to the POWCON register.
4. When the part is interrupted from nap mode by the Timer1 interrupt source, the clock source has switched to the external crystal.

Example source code

```
T1LD = 0x80; // 32,768 clock ticks
T1CON = 0xC0; // Periodic mode, enable
           // timer, 32,768 Hz clock/1
IRQEN |= 0x10; // Enable Timer1 interrupt
           // source
PLLKEY1 = 0xAA; // Switch to external crystal
PLLCON = 0x2;
PLLKEY2 = 0x55;

POWKEY1 = 0x1; // Enter nap mode
POWCON0 = 0x73;
POWKEY2 = 0xF4;
```

In case of crystal loss, the watchdog timer should be used. During initialization, a test on the RSTSTA can determine if the reset came from the watchdog timer.

### External Clock Selection

To switch to an external clock on P2.0, configure P2.0 in Mode 0. The external clock can be up to 20.48 MHz, provided that the tolerance is 1%. The external clock is divided by 2 internally on the part.

Example source code

```
T1LD = 0x80;
T1CON = 0xC0;
IRQEN |= 0x10; // Enable Timer1 interrupt
PLLKEY1 = 0xAA; // Switch to external clock
PLLCON = 0x4;
PLLKEY2 = 0x55;

POWKEY1 = 0x1; // Enter NAP mode
POWCON0 = 0x73;
POWKEY2 = 0xF4;
```

The selection of the clock source is in the PLLCON register. By default, the part uses the internal oscillator feeding the PLL.

### POWER CONTROL SYSTEM

The core clock frequency is changed by writing to the POWCON0 register. This is a key protected register; therefore, Register POWKEY1 and Register POWKEY2 must be written to immediately before and after configuring the POWCON0 register. The following is a simple example showing how to configure the core clock for 10.24 MHz:

```
POWKEY1 = 0x1;
POWCON0 = 0x78; //Set core to max CPU
           //speed of 10.24 MHz
POWKEY2 = 0xF4;
```

A choice of operating modes is available on the ADuC7060/ADuC7061. Table 33 describes what part is powered on in the different modes and indicates the power-up time.

Table 34 gives some typical values for the total current consumption (analog + digital supply currents) in the different modes, depending on the clock divider bits. The ADC is turned off. Note that these values also include the current consumption of the regulator and other parts on the test board where these values are measured.

By writing to POWCON1, it is possible to further reduce power consumption in active mode by powering down the UART, PWM or I<sup>2</sup>C/SPI blocks. To access POWCON1, POWKEY3 must be set to 0x76 in the instruction immediately before accessing POWCON1 and POWKEY4 must be set to 0xB1 in the instruction immediately after.

For example, the following code enables the SPI/I<sup>2</sup>C blocks but, powers down the PWM and UART blocks.

```
POWKEY3 =0x76;
POWCON1 =0x4;           //0x100 PWM; 0x20
Uart; 0x4 SPI/I2C
POWKEY4 =0xB1;
```

## Power and Clock Control Registers

### POWKEY1 Register

Name: POWKEY1  
 Address: 0xFFFF0404  
 Default value: 0XXXXX  
 Access: Write  
 Function: When writing to POWCON0, the value of 0x01 must be written to this register in the instruction immediately before writing to POWCON0.

### POWCON0 Register

Name: POWCON0  
 Address: 0xFFFF0408  
 Default value: 0x7B  
 Access: Read and write  
 Function: This register controls the clock divide bits controlling the CPU clock (HCLK).

**Table 31. POWCON0 MMR Bit Designations**

| Bit | Name     | Description   |
|-----|----------|---|
| 7   | Reserved | This bit must always be set to 0.   |
| 6   | XPD      | XTAL power-down.<br>Cleared by user to power down the external crystal circuitry.<br>Set by user to enable the external crystal circuitry.  |
| 5   | PLLPD    | PLL power-down. Timer peripherals power down if driven from the PLL output clock. Timers driven from an active clock source remain in normal power mode.<br>This bit is cleared to 0 to power down the PLL. The PLL cannot be powered down if either the core or peripherals are enabled; Bit 3, Bit 4, and Bit 5 must be cleared simultaneously.<br>Set by default, and set by hardware on a wake-up event.  |
| 4   | PPD      | Peripherals power-down. The peripherals that are powered down by this bit are as follows:<br>SRAM, Flash/EE memory and GPIO interfaces, and SPI/I <sup>2</sup> C and UART serial ports.<br>Cleared to power down the peripherals. The peripherals cannot be powered down if the core is enabled; Bit 3 and Bit 4 must be cleared simultaneously.<br>Set by default and/or by hardware on a wake-up event. Wake-up timer (Timer1) can remain active. |
| 3   | COREPD   | Core power-down. If user code powers down the MCU, include a dummy MCU cycle after the power-down command is written to POWCON0.<br>Cleared to power down the ARM core.<br>Set by default and set by hardware on a wake-up event.   |
| 2:0 | CD[2:0]  | Core clock depends on CD setting:<br>[000] = 10.24 MHz<br>[001] = 5.12 MHz<br>[010] = 2.56 MHz<br>[011] = 1.28 MHz [default value]<br>[100] = 640 kHz<br>[101] = 320 kHz<br>[110] = 160 kHz<br>[111] = 80 kHz   |

**POWKEY2 Register**

|                |   |
|----------------|---|
| Name:          | POWKEY2   |
| Address:       | 0xFFFF040C  |
| Default value: | 0XXXXX  |
| Access:        | Write   |
| Function:      | When writing to POWCON0, the value of 0xF4 must be written to this register in the instruction immediately before writing to POWCON0. |

**POWKEY3 Register**

|                |   |
|----------------|---|
| Name:          | POWKEY3   |
| Address:       | 0xFFFF0434  |
| Default value: | 0XXXXX  |
| Access:        | Write   |
| Function:      | When writing to POWCON1, the value of 0x76 must be written to this register in the instruction immediately before writing to POWCON1. |

**POWCON1 Register**

|                |   |
|----------------|---|
| Name:          | POWCON1   |
| Address:       | 0xFFFF0438  |
| Default value: | 0x124   |
| Access:        | Read and write  |
| Function:      | This register controls the clock signal to the PWM, UART and I2C/SPI blocks.<br><br>By disabling the clock to these blocks, power consumption is reduced. |

**POWKEY4 Register**

|                |  |
|----------------|--|
| Name:          | POWKEY4  |
| Address:       | 0xFFFF043C   |
| Default value: | 0XXXXX   |
| Access:        | Write  |
| Function:      | When writing to POWCON1, the value of 0xB1 must be written to this register in the instruction immediately after writing to POWCON1. |

**Table 32. POWCON1 MMR Bit Designations**

| Bit  | Name      | Description   |
|------|-----------|---|
| 15:9 | Reserved  | This bit must always be set to 0.   |
| 8    | PWMOFF    | PWM power-down bit.<br>Set by user to 1 to enable the PWM block. This bit is set by default.<br>Cleared by user to 0 to power down the PWM block.               |
| 7:6  | Reserved  | Reserved bits. Always clear these bits to 0.  |
| 5    | UARTOFF   | UART power-down bit.<br>Set by user to 1 to enable the UART block. This bit is set by default.<br>Cleared by user to 0 to power down the UART block.            |
| 4:3  | Reserved  | Reserved bits. Always clear these bits to 0.  |
| 2    | I2CSPIOFF | I2C/SPI power-down bit.<br>Set by user to 1 to enable the I2C/SPI blocks. This bit is set by default.<br>Cleared by user to 0 to power down the I2C/SPI blocks. |
| 1:0  | Reserved  | Reserved Bits. Always clear these bits to 0.  |

**Table 33. ADuC7060/ADuC7061 Power Saving Modes**

| POWCON0[6:3] | Mode   | Core | Peripherals | PLL | XTAL/T1/T2 | IRQ0 to IRQ3 | Start-Up/Power-On Time                       |
|--------------|--------|------|-------------|-----|------------|--------------|--|
| 1111         | Active | Yes  | Yes         | Yes | Yes        | Yes          | 130 ms at CD = 0                             |
| 1110         | Pause  |      | Yes         | Yes | Yes        | Yes          | 4.8 $\mu$ s at CD = 0; 660 $\mu$ s at CD = 7 |
| 1100         | Nap    |      |             | Yes | Yes        | Yes          | 4.8 $\mu$ s at CD = 0; 660 $\mu$ s at CD = 7 |
| 1000         | Sleep  |      |             |     | Yes        | Yes          | 66 $\mu$ s at CD = 0; 900 $\mu$ s at CD = 7  |
| 0000         | Stop   |      |             |     |            | Yes          | 66 $\mu$ s at CD = 0; 900 $\mu$ s at CD = 7  |

**Table 34. Typical Current Consumption at 25°C in mA<sup>1</sup>**

| POWCON0[6:3] | Mode                | CD = 0 | CD = 1 | CD = 2 | CD = 3 | CD = 4 | CD = 5 | CD = 6 | CD = 7 |
|--------------|---------------------|--------|--------|--------|--------|--------|--------|--------|--------|
| 1111         | Active <sup>2</sup> | 5.22   | 4.04   | 2.69   | 2.01   | 1.67   | 1.51   | 1.42   | 1.38   |
| 1110         | Pause <sup>3</sup>  | 2.6    | 1.95   | 1.6    | 1.49   | 1.4    | 1.33   | 1.31   | 1.3    |
| 1100         | Nap <sup>3</sup>    | 1.33   | 1.29   | 1.29   | 1.29   | 1.29   | 1.29   | 1.29   | 1.29   |
| 1000         | Sleep <sup>3</sup>  | 0.085  | 0.085  | 0.085  | 0.085  | 0.085  | 0.085  | 0.085  | 0.085  |
| 0000         | Stop <sup>3</sup>   | 0.055  | 0.055  | 0.055  | 0.055  | 0.055  | 0.055  | 0.055  | 0.055  |

<sup>1</sup> All values listed in Table 34 have been taken with both ADCs turned off.

<sup>2</sup> In active mode, GP0PAR bit 7 = 1.

<sup>3</sup> The values for pause, nap, sleep, and stop modes are measured with the NTRST pin low. To minimize I<sub>DD</sub> due to nTRST in all modes, set GP0PAR Bit 7 = 1. This disables the internal pull-down on the nTRST pin and means there is no ground path for the external pull-up resistor through the nTRST pin. By default, GP0PAR Bit 7 = 0, therefore, setting this bit in user code will not affect the BMOperation.

Name: PLLKEY1  
 Address: 0xFFFF0410  
 Default value: 0XXXXX  
 Access: Write  
 Function: When writing to the PLLCON register, the value of 0xAA must be written to this register in the instruction immediately before writing to PLLCON.

**Table 35. PLLCON MMR Bit Designations**

| Bit | Name     | Description   |
|-----|----------|---|
| 7:3 | Reserved | These bits must always be set to 0.   |
| 2   | EXTCLK   | Set this bit to 1 to select external clock input from P2.0.<br>Clear this bit to disable the external clock.  |
| 1:0 | OSEL     | Oscillator selection bits.<br>[00] = internal 32,768 Hz oscillator.<br>[01] = internal 32,768 Hz oscillator.<br>[10] = external crystal.<br>[11] = internal 32,768 Hz oscillator. |

Name: PLLCON  
 Address: 0xFFFF0414  
 Default value: 0x00  
 Access: Read and write  
 Function: This register selects the clock input to the PLL.

Name: PLLKEY2  
 Address: 0xFFFF0418  
 Default value: 0XXXXX  
 Access: Write  
 Function: When writing to PLLCON, the value of 0x55 must be written to this register in the instruction immediately after writing to PLLCON.

## ADC CIRCUIT INFORMATION

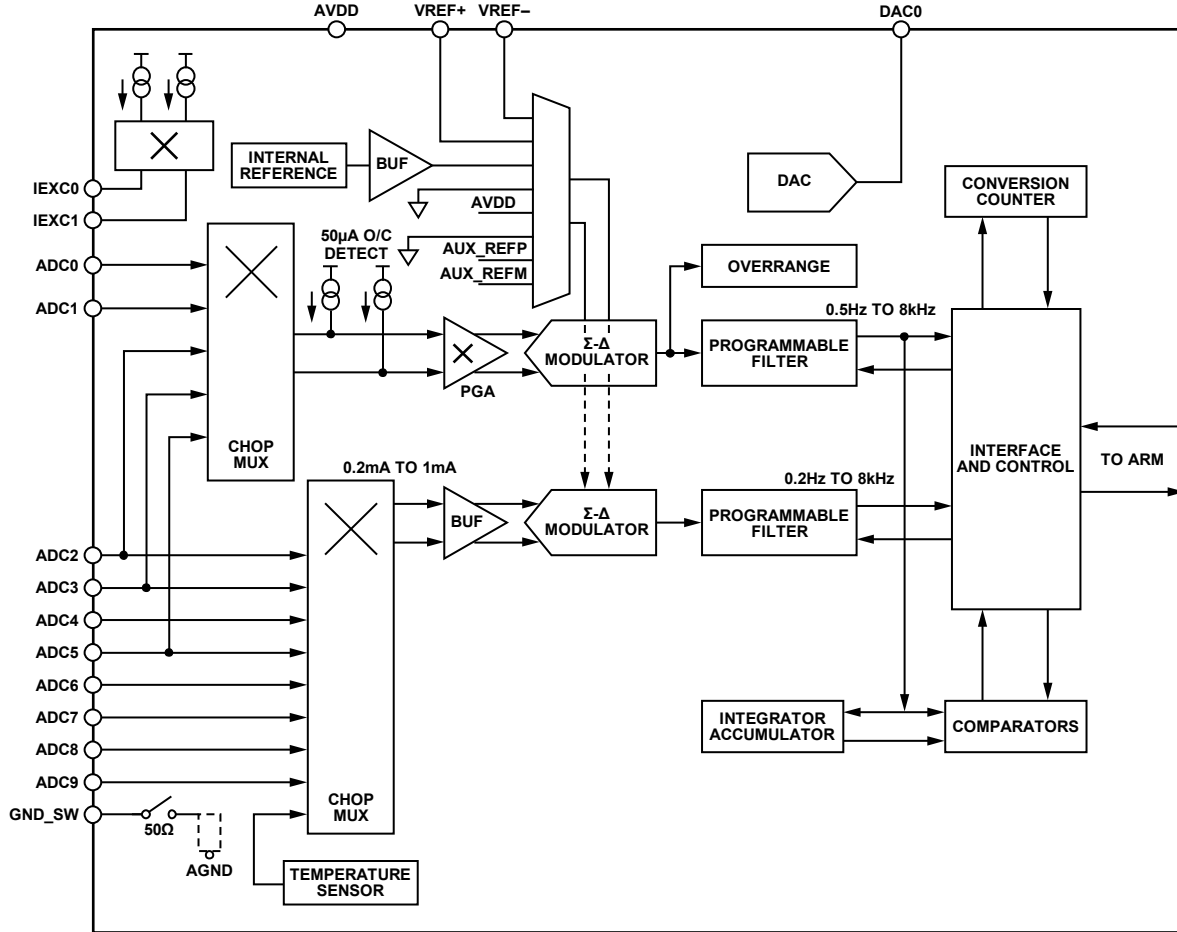


Figure 14. Analog Block Diagram

07079-009

The ADuC7060/ADuC7061 incorporates two independent multichannel  $\Sigma$ - $\Delta$  ADCs. The primary ADC is a 24-bit, 4-channel ADC. The auxiliary ADC is a 24-bit  $\Sigma$ - $\Delta$  ADC, with up to seven single-ended input channels.

The primary ADC input has a mux and a programmable gain amplifier on its input stage. The mux on the primary channel can be configured as two fully differential input channels or as four single-ended input channels.

The auxiliary ADC incorporates a buffer on its input stage. Digital filtering is present on both ADCs, which allows

measurement of a wide dynamic range and low frequency signals such as those in pressure sensor, temperature sensor, weigh scale, or strain gage type applications.

The ADuC7060/ADuC7061 auxiliary ADC can be configured as four fully differential input channels or as seven single-ended input channels.

Because of internal buffering, the internal channels can convert signals directly from sensors without the need for external signal conditioning.

Table 36. Primary ADC—Typical Output RMS Noise in Normal Mode ( $\mu\text{V}$ )<sup>1</sup>

| ADC Register Status | Data Update Rate | Selectable Input Range (PGA Settings) |                               |                               |                               |                               |                                 |                                  |                                   |                                  |                                  |
|---------------------|------------------|---------------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|---------------------------------|----------------------------------|-----------------------------------|----------------------------------|----------------------------------|
|                     |                  | $\pm 1.2\text{ V}$ (PGA = 1)          | $\pm 600\text{ mV}$ (PGA = 2) | $\pm 300\text{ mV}$ (PGA = 4) | $\pm 150\text{ mV}$ (PGA = 8) | $\pm 75\text{ mV}$ (PGA = 16) | $\pm 37.5\text{ mV}$ (PGA = 32) | $\pm 18.75\text{ mV}$ (PGA = 64) | $\pm 9.375\text{ mV}$ (PGA = 128) | $\pm 4.68\text{ mV}$ (PGA = 256) | $\pm 2.34\text{ mV}$ (PGA = 512) |
| Chop On             | 4 Hz             | 0.62 $\mu\text{V}$                    | 0.648 $\mu\text{V}$           | 0.175 $\mu\text{V}$           | 0.109 $\mu\text{V}$           | 0.077 $\mu\text{V}$           | 0.041 $\mu\text{V}$             | 0.032 $\mu\text{V}$              | 0.0338 $\mu\text{V}$              | 0.032 $\mu\text{V}$              | 0.033 $\mu\text{V}$              |
| Chop Off            | 50 Hz            | 1.97 $\mu\text{V}$                    | 1.89 $\mu\text{V}$            | 0.570 $\mu\text{V}$           | 0.38 $\mu\text{V}$            | 0.27 $\mu\text{V}$            | 0.147 $\mu\text{V}$             | 0.123 $\mu\text{V}$              | 0.12 $\mu\text{V}$                | 0.098 $\mu\text{V}$              | 0.098 $\mu\text{V}$              |
| Chop Off            | 1 kHz            | 8.54 $\mu\text{V}$                    | 8.4 $\mu\text{V}$             | 2.55 $\mu\text{V}$            | 1.6 $\mu\text{V}$             | 1.17 $\mu\text{V}$            | 0.658 $\mu\text{V}$             | 0.53 $\mu\text{V}$               | 0.55 $\mu\text{V}$                | 0.56 $\mu\text{V}$               | 0.52 $\mu\text{V}$               |
| Chop Off            | 8 kHz            | 54.97 $\mu\text{V}$                   | 55.54 $\mu\text{V}$           | 14.30 $\mu\text{V}$           | 7.88 $\mu\text{V}$            | 4.59 $\mu\text{V}$            | 2.5 $\mu\text{V}$               | 1.71 $\mu\text{V}$               | 1.75 $\mu\text{V}$                | 0.915 $\mu\text{V}$              | 0.909 $\mu\text{V}$              |

<sup>1</sup> The input voltage range is centered around the common-mode voltage and should meet the input voltage range specified in the Electrical Specifications section.

Table 37. Primary ADC—Typical Output RMS Effective Number of Bits in Normal Mode (Peak-to-Peak Bits in Parentheses)

| ADC Register Status | Data Update Rate | Input Voltage Noise (mV)     |                               |                               |                               |                               |                                 |                                  |                                   |                                  |                                  |
|---------------------|------------------|------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|---------------------------------|----------------------------------|-----------------------------------|----------------------------------|----------------------------------|
|                     |                  | $\pm 1.2\text{ V}$ (PGA = 1) | $\pm 600\text{ mV}$ (PGA = 2) | $\pm 300\text{ mV}$ (PGA = 4) | $\pm 150\text{ mV}$ (PGA = 8) | $\pm 75\text{ mV}$ (PGA = 16) | $\pm 37.5\text{ mV}$ (PGA = 32) | $\pm 18.75\text{ mV}$ (PGA = 64) | $\pm 9.375\text{ mV}$ (PGA = 128) | $\pm 4.68\text{ mV}$ (PGA = 256) | $\pm 2.34\text{ mV}$ (PGA = 512) |
| Chop On             | 4 Hz             | 21.9 (19.1 p-p)              | 20.8 (18.1 p-p)               | 21.7 (19.0 p-p)               | 21.4 (18.7 p-p)               | 20.9 (18.2 p-p)               | 20.8 (18.1 p-p)                 | 20.2 (17.4 p-p)                  | 19.1 (16.4 p-p)                   | 18.2 (15.4 p-p)                  | 17.1 (14.4 p-p)                  |
| Chop Off            | 50 Hz            | 20.2 (17.5 p-p)              | 19.3 (16.6 p-p)               | 20.0 (17.3 p-p)               | 19.6 (16.9 p-p)               | 19.1 (16.4 p-p)               | 19.0 (16.2 p-p)                 | 18.2 (15.5 p-p)                  | 17.3 (14.6 p-p)                   | 16.6 (13.8 p-p)                  | 15.5 (12.8 p-p)                  |
| Chop Off            | 1 kHz            | 18.1 (15.3 p-p)              | 17.1 (14.4 p-p)               | 17.8 (15.1 p-p)               | 17.5 (14.8 p-p)               | 17.0 (14.2 p-p)               | 16.8 (14.1 p-p)                 | 16.1 (13.4 p-p)                  | 15.1 (12.3 p-p)                   | 14.0 (11.3 p-p)                  | 13.1 (10.4 p-p)                  |
| Chop Off            | 8 kHz            | 15.4 (12.7 p-p)              | 14.4 (11.7 p-p)               | 15.4 (12.6 p-p)               | 15.2 (12.5 p-p)               | 15.0 (12.3 p-p)               | 14.9 (12.2 p-p)                 | 14.4 (11.7 p-p)                  | 13.4 (10.7 p-p)                   | 13.3 (10.6 p-p)                  | 12.3 (9.6 p-p)                   |

Table 38. Auxiliary ADC—Typical Output RMS Noise

| ADC Register | Data Update Rate | RMS Value           |
|--------------|------------------|---------------------|
| Chop On      | 4 Hz             | 0.633 $\mu\text{V}$ |
| Chop On      | 10 Hz            | 0.810 $\mu\text{V}$ |
| Chop Off     | 1 kHz            | 7.4 $\mu\text{V}$   |
| Chop Off     | 8 kHz            | 54.18 $\mu\text{V}$ |

**REFERENCE SOURCES**

Both the primary and auxiliary ADCs have the option of using the internal reference voltage or one of two external differential reference sources. The first external reference is applied to the VREF+/VREF- pins. The second external reference is applied to the ADC4/EXT\_REF2IN+ and ADC5/EXT\_REF2IN- pins. By default, each ADC uses the internal 1.2 V reference source.

For details on how to configure the external reference source for the primary ADC, see the description of the ADC0REF[1:0] bits in the ADC0 control register, ADC0CON.

For details on how to configure the external reference source for the auxiliary ADC, see the description of the ADC1REF[2:0] bits in the ADC1 control register, ADC1CON.

If an external reference source of greater than 1.35 V is needed for ADC0, the HIGHEXTREF0 bit must be set in ADC0CON.

Similarly, if an external reference source of greater than 1.35 V is used for ADC1, the HIGHEXTREF1 bit must be set in ADC1CON.

**DIAGNOSTIC CURRENT SOURCES**

To detect a connection failure to an external sensor, the ADuC7060/ADuC7061 incorporates a 50  $\mu\text{A}$  constant current source on the selected analog input channels to both the primary and auxiliary ADCs.

The diagnostic current sources for the primary ADC analog inputs are controlled by the ADC0DIAG[1:0] bits in the ADC0CON register.

Similarly, the diagnostic current sources for the auxiliary ADC analog inputs are controlled by the ADC1DIAG[1:0] bits in the ADC1CON register.

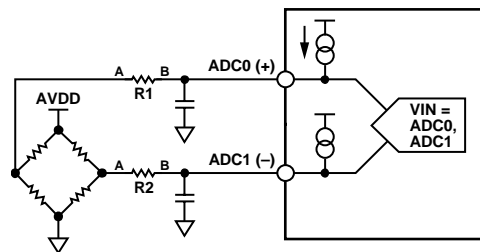


Figure 15. Example Circuit Using Diagnostic Current Sources



Table 39. Example Scenarios for Using Diagnostic Current Sources

| Diagnostic Test   |  | Normal Result   | Fault Result   | Detected Measurement for Fault                                     |
|-------------------|--|---|--|--|
| Register Setting  | Description  |   |  |  |
| ADC0DIAG[1:0] = 0 | Convert ADC0/ADC1 as normal with diagnostic currents disabled.   | Expected differential result across ADC0/ADC1.  | Short circuit.   | Primary ADC reading $\approx 0$ V regardless of PGA setting.       |
| ADC0DIAG[1:0] = 1 | Enable a 50 $\mu$ A diagnostic current source on ADC0 by setting ADC0DIAG[1:0] = 1. Convert ADC0 and ADC1.               | Main ADC changes by $\Delta V = +50 \mu\text{A} \times R1$ . For example, $\sim 100$ mV for $R1 = 2$ k $\Omega$ . | Short circuit between ADC0 and ADC1.<br>Short circuit between R1_a and R1_b. | Primary ADC reading $\approx 0$ V regardless of PGA setting.       |
|                   | Convert ADC0 in single-ended mode with diagnostic currents disabled.   | Expected voltage on ADC0.   | ADC0 open circuit or R1 open circuit.  | Primary ADC reading = +full scale, even on the lowest PGA setting. |
| ADC0DIAG[1:0] = 3 | Enable a 50 $\mu$ A diagnostic current source on both ADC0 and ADC1 by setting ADC0DIAG[1:0] = 3. Convert ADC0 and ADC1. | Primary ADC changes by $\Delta V = 50 \mu\text{A} \times (R1 - R2)$ , that is, $\sim 10$ mV for 10% tolerance.    | R1 does not match R2.  | Primary ADC reading $> 10$ mV.                                     |

### SINC3 FILTER

The number entered into Bits[6:0] of the ADCFLT register sets the decimation factor of the sinc3 filter. See Table 46 and Table 47 for further details on the decimation factor values.

The range of operation of the sinc3 filter (SF) word depends on whether the chop function is enabled. With chopping disabled, the minimum SF word allowed is 0 and the maximum is 127, giving an ADC throughput range of 50 Hz to 8 kHz.

For details on how to calculate the ADC sampling frequency based on the value programmed to the SF[6:0] bits in the ADCFLT register, refer to Table 46. When changing conversion speeds, put ADC into idle mode before restarting.

### ADC CHOPPING

The ADCs on the ADuC7060/ADuC7061 implements a chopping scheme whereby the ADC repeatedly reverses its inputs. Therefore, the decimated digital output values from the sinc3 filter have a positive and negative offset term associated with them. This results in the ADC including a final summing stage that sums and averages each value from the filter with previous filter output values. This new value is then sent to the ADC data MMR. This chopping scheme results in excellent dc offset and offset drift specifications and is extremely beneficial in applications where drift and noise rejection are required.

### PROGRAMMABLE GAIN AMPLIFIER

The primary ADC incorporates an on-chip programmable gain amplifier (PGA). The PGA can be programmed through 10 different settings giving a range of 1 to 512. The gain is controlled by the ADC0PGA[3:0] bits in the ADC0CON MMR.

### EXCITATION SOURCES

The ADuC7060/ADuC7061 contains two matched software configurable current sources. These excitation currents are sourced from AVDD. They are individually configurable to give

a current range of 200  $\mu$ A to 1 mA. The current step sizes are 200  $\mu$ A. These current sources can be used to excite an external resistive bridge or RTD sensors. The IEXCON MMR controls the excitation current sources. Bit 6 of IEXCON must be set to enable Excitation Current Source 0. Similarly, Bit 7 must be set to enable Excitation Current Source 1. The output current of each current source is controlled by the IOUT[3:0] bits of this register.

It is also possible to configure the excitation current sources to output current to a single output pin, either IEXC0 or IEXC1, by using the IEXC0\_DIR and IEXC1\_DIR bits of IEXCON. This allows up to 2 mA to output current on a single excitation pin.

### ADC LOW POWER MODE

The ADuC7060/ADuC7061 allows the primary and auxiliary ADCs to be placed in low power operating mode. When configured for this mode, the ADC throughput time is reduced, but the power consumption of the primary ADC is reduced by a factor of about 4; the auxiliary ADC power consumption is reduced by a factor of roughly 3. The maximum ADC conversion rate in low power mode is 2 kHz. The operating mode of the ADCs is controlled by the ADCMDE register. This register configures the part for either normal mode (default), low power mode, or low power plus mode. Low power plus mode is the same as low power mode except that the PGA is disabled. To place the ADCs into low power mode, the following steps must be completed:

- ADCMDE[4:3]—Setting these bits enables normal mode, low power mode, or low power plus mode.
- ADCMDE[5]—Setting this bit configures the part for low power mode.
- ADCMDE[7]—Clearing this bit further reduces power consumption by reducing the frequency of the ADC clock.

## ADC COMPARATOR AND ACCUMULATOR

Every primary ADC result can be compared to a preset threshold level (ADC0TH) as configured via ADCCFG[4:3]. An MCU interrupt is generated if the absolute (sign independent) value of the ADC result is greater than the preprogrammed comparator threshold level. An extended function of this comparator function allows user code to configure a threshold counter (ADC0THV) to monitor the number of primary ADC results that have occurred above or below the preset threshold level. Again, an ADC interrupt is generated when the threshold counter reaches a preset value (ADC0RCR).

Finally, a 32-bit accumulator (ADC0ACC) function can be configured (ADCCFG[6:5]) allowing the primary ADC to add (or subtract) multiple primary ADC sample results. User code can read the accumulated value directly (ADC0ACC) without any further software processing.

## TEMPERATURE SENSOR

The ADuC7060/ADuC7061 provides a voltage output from an on-chip band gap reference proportional to absolute temperature. This voltage output can also be routed through the front-end auxiliary ADC multiplexer (effectively, an additional ADC channel input), facilitating an internal temperature sensor channel that measures die temperature.

The internal temperature sensor is not designed for use as an absolute ambient temperature calculator. It is intended for use as an approximate indicator of the temperature of the ADuC7060/ADuC7061 die.

The typical temperature coefficient is 0.28 mV/°C.

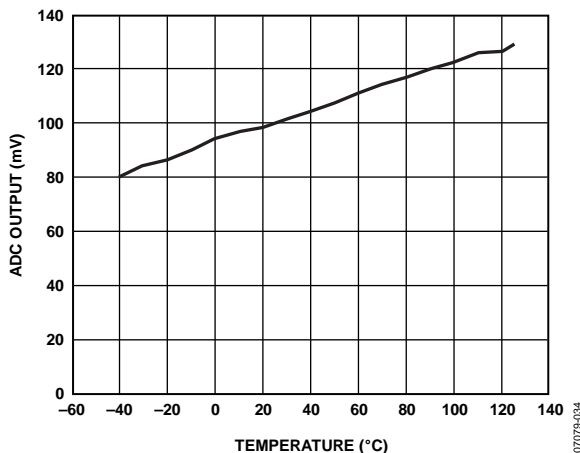


Figure 16. ADC Output vs. Temperature

## ADC MMR INTERFACE

The ADCs are controlled and configured through a number of MMRs that are described in detail in the following sections.

In response to an ADC interrupt, user code should interrogate the ADCSTA MMR to determine the source of the interrupt. Each ADC interrupt source can be individually masked via the ADCMSKI MMR described in Table 41.

All primary ADC result ready bits are cleared by a read of the ADC0DAT MMR. If the primary channel ADC is not enabled, all ADC result ready bits are cleared by a read of the ADC1DAT MMR. To ensure that primary ADC and auxiliary ADC conversion data are synchronous, user code should first read the ADC1DAT MMR and then the ADC0DAT MMR. New ADC conversion results are not written to the ADCxDAT MMRs unless the respective ADC result ready bits are first cleared. The only exception to this rule is the data conversion result updates when the ARM core is powered down. In this mode, ADCxDAT registers always contain the most recent ADC conversion result even though the ready bits are not cleared.

### ADC Status Register

Name: ADCSTA

Address: 0xFFFF0500

Default value: 0x0000

Access: Read only

Function: This read-only register holds general status information related to the mode of operation or current status of the ADuC7060/ADuC7061 ADCs.

Table 40. ADCSTA MMR Bit Designations

| Bit  | Name      | Description  |
|------|-----------|--|
| 15   | ADCCALSTA | ADC calibration status.<br>This bit is set automatically in hardware to indicate that an ADC calibration cycle has been completed.<br>This bit is cleared after ADCMDE is written to.  |
| 14   |           | Not used.<br>This bit is reserved for future functionality.  |
| 13   | ADC1CERR  | Auxiliary ADC conversion error.<br>This bit is set automatically in hardware to indicate that an auxiliary ADC conversion overrange or underrange has occurred. The conversion result is clamped to negative full scale (underrange error) or positive full scale (overrange error) in this case.<br>This bit is cleared when a valid (in-range) voltage conversion result is written to the ADC1DAT register.   |
| 12   | ADC0CERR  | Primary ADC conversion error.<br>This bit is set automatically in hardware to indicate that a primary ADC conversion overrange or underrange has occurred. The conversion result is clamped to negative full scale (underrange error) or positive full scale (overrange error) in this case.<br>This bit is cleared when a valid (in-range) conversion result is written to the ADC0DAT register.  |
| 11:7 |           | Not used. These bits are reserved for future functionality and should not be monitored by user code.   |
| 6    | ADC0ATHEX | ADC0 accumulator comparator threshold exceeded.<br>This bit is set when the ADC0 accumulator value in ADC0ACC exceeds the threshold value programmed in the ADC0 comparator threshold register, ADC0ATH.<br>This bit is cleared when the value in ADC0ACC does not exceed the value in ADC0ATH.  |
| 5    |           | Not used. This bit is reserved for future functionality and should not be monitored by user code.  |
| 4    | ADC0THEX  | Primary channel ADC comparator threshold. This bit is valid only if the primary channel ADC comparator is enabled via the ADCCFG MMR.<br>This bit is set by hardware if the absolute value of the primary ADC conversion result exceeds the value written in the ADC0TH MMR. If the ADC threshold counter is used (ADC0RCR), this bit is set only when the specified number of primary ADC conversions equals the value in the ADC0THV MMR.<br>Otherwise, this bit is cleared. |
| 3    | ADC0OVR   | Primary channel ADC overrange bit. If the overrange detect function is enabled via the ADCCFG MMR, this bit is set by hardware if the primary ADC input is grossly (>30% approximate) overrange. This bit is updated every 125 $\mu$ s. After it is set, this bit can be cleared only by software when ADCCFG[2] is cleared to disable the function, or the ADC gain is changed via the ADC0CON MMR.   |
| 2    |           | Not used. This bit is reserved for future functionality and should not be monitored by user code.  |
| 1    | ADC1RDY   | Auxiliary ADC result ready bit.<br>If the auxiliary channel ADC is enabled, this bit is set by hardware as soon as a valid conversion result is written in the ADC1DAT MMR. It is also set at the end of a calibration sequence.<br>This bit is cleared by reading ADC1DAT followed by reading ADC0DAT. ADC0DAT must be read to clear this bit, even if the primary ADC is not enabled.  |
| 0    | ADC0RDY   | Primary ADC result ready bit.<br>If the primary channel ADC is enabled, this bit is set by hardware as soon as a valid conversion result is written in the ADC0DAT MMR. It is also set at the end of a calibration sequence.<br>This bit is cleared by reading ADC0DAT.  |

**ADC Interrupt Mask Register**

Name: ADCMSKI

Address: 0xFFFF0504

Default value: 0x0000

Access: Read and write

Function: This register allows the ADC interrupt sources to be enabled individually. The bit positions in this register are the same as the lower eight bits in the ADCSTA MMR. If a bit is set by user code to 1, the respective interrupt is enabled. By default, all bits are 0, meaning all ADC interrupt sources are disabled.

**Table 41. ADCMSKI MMR Bit Designations**

| Bit | Name            | Description  |
|-----|-----------------|--|
| 7   |                 | Not used. This bit is reserved for future functionality and should not be monitored by user code.  |
| 6   | ADC0ATHEX_INTEN | ADC0 accumulator comparator threshold exceeded interrupt enable bit.<br>When set to 1, this bit enables an interrupt when the ADC0ATHEX bit in the ADCSTA register is set.<br>When this bit is cleared, this interrupt source is disabled.   |
| 5   |                 | Not used. This bit is reserved for future functionality and should not be monitored by user code.  |
| 4   | ADC0THEX_INTEN  | Primary channel ADC comparator threshold exceeded interrupt enable bit.<br>When set to 1, this bit enables an interrupt when the ADC0THEX bit in the ADCSTA register is set.<br>When this bit is cleared, this interrupt source is disabled. |
| 3   | ADC0OVR_INTEN   | When set to 1, this bit enables an interrupt when the ADC0OVR bit in the ADCSTA register is set.<br>When this bit is cleared, this interrupt source is disabled.   |
| 2   |                 | Not used. This bit is reserved for future functionality and should not be monitored by user code.  |
| 1   | ADC1RDY_INTEN   | Auxiliary ADC result ready bit.<br>When set to 1, this bit enables an interrupt when the ADC1RDY bit in the ADCSTA register is set.<br>When this bit is cleared, this interrupt source is disabled.  |
| 0   | ADC0RDY_INTEN   | Primary ADC result ready bit.<br>When set to 1, this bit enables an interrupt when the ADC0RDY bit in the ADCSTA register is set.<br>When this bit is cleared, this interrupt source is disabled.  |

**ADC Mode Register**

Name: ADCMDE

Address: 0xFFFF0508

Default value: 0x03

Access: Read and write

Function: The ADC mode MMR is an 8-bit register that configures the mode of operation of the ADC subsystem.

**Table 42. ADCMDE MMR Bit Designations**

| Bit | Name      | Description   |
|-----|-----------|---|
| 7   | ADCCLKSEL | Set this bit to 1 to enable ADCCLK = 512 kHz. This bit should be set for normal ADC operation.<br>Clear this bit to enable ADCCLK = 131 kHz. This bit should be cleared for low power ADC operation.              |
| 6   |           | Not used. This bit is reserved for future functionality and should not be monitored by user code.   |
| 5   | ADCLPMEN  | Enable low power mode. This bit has no effect if ADCMDE[4:3] = 00 (ADC is in normal mode).<br>This bit must be set to 1 in low power mode.<br>Clearing this bit in low power mode results in erratic ADC results. |

| Bit | Name           | Description   |
|-----|----------------|---|
| 4:3 | ADCLPMCFG[1:0] | <p>ADC power mode configuration.</p> <p>[00] = ADC normal mode. If enabled, the ADC operates with normal current consumption yielding optimum electrical performance.</p> <p>[01] = ADC low power mode.</p> <p>[10] = ADC normal mode, same as [00].</p> <p>[11] = ADC low power plus mode (low power mode and PGA off).</p>  |
| 2:0 | ADCMD[2:0]     | <p>ADC operation mode configuration.</p> <p>[000] = ADC power-down mode. All ADC circuits and the input amplifier are powered down.</p> <p>[001] = ADC continuous conversion mode. In this mode, any enabled ADC continuously converts at a frequency equal to <math>f_{ADC}</math>. ADCxRDY must be cleared to enable new data to be written to ADC0DAT/ADC1DAT.</p> <p>[010] = ADC single conversion mode. In this mode, any enabled ADC performs a single conversion. The ADC enters idle mode when the single shot conversion is complete. A single conversion takes two to three ADC clock cycles, depending on the chop mode.</p> <p>[011] = ADC idle mode. In this mode, the ADC is fully powered on but is held in reset. The part enters this mode after calibration.</p> <p>[100] = ADC self-offset calibration. In this mode, an offset calibration is performed on any enabled ADC using an internally generated 0 V. The calibration is carried out at the user-programmed ADC settings; therefore, as with a normal single ADC conversion, it takes two to three ADC conversion cycles before a fully settled calibration result is ready. The calibration result is automatically written to the ADCxOF MMR of the respective ADC. The ADC returns to idle mode, and the calibration and conversion ready status bits are set at the end of an offset calibration cycle.</p> <p>Note: Always use ADC0 for single-ended self-calibration cycles on the primary ADC. Always use ADC0/ADC1 when self-calibrating for a differential input to the primary ADC.</p> <p>[101] = ADC self-gain calibration. In this mode, a gain calibration against an internal reference voltage is performed on all enabled ADCs. A gain calibration is a two-stage process and takes twice the time of an offset calibration. The calibration result is automatically written to the ADCxGN MMR of the respective ADC. The ADC returns to idle mode and the calibration and conversion ready status bits are set at the end of a gain calibration cycle. An ADC self-gain calibration should only be carried out on the primary channel ADC.</p> <p>Note that self-gain calibration works only when the gain = 1; do not use it when the gain &gt; 1.</p> <p>[110] = ADC system zero-scale calibration. In this mode, a zero-scale calibration is performed on enabled ADC channels against an external zero-scale voltage driven at the ADC input pins. To do this, short the channel externally.</p> <p>[111] = ADC system full-scale calibration. In this mode, a full-scale calibration is performed on enabled ADC channels against an external full-scale voltage driven at the ADC input pins. The ADCxGN register is updated after a full-scale calibration sequence.</p> |

### Primary ADC Control Register

Name: ADC0CON

Address: 0xFFFF050C

Default value: 0x8000

Access: Read and write

Function: The primary channel ADC control MMR is a 16-bit register. If the primary ADC is reconfigured via ADC0CON, the auxiliary ADC is also reset.

Table 43. ADC0CON MMR Bit Designations

| Bit   | Name          | Description  |
|-------|---------------|--|
| 15    | ADC0EN        | Primary channel ADC enable.<br>This bit is set to 1 by user code to enable the primary ADC.<br>Clearing this bit to 0 powers down the primary ADC and resets the respective ADC ready bit in the ADCSTA MMR to 0.  |
| 14:13 | ADCODIAG[1:0] | Diagnostic current source enable bits.<br>[00] = current sources off.<br>[01] = enables a 50 $\mu$ A current source on the selected positive input (for example, ADC0).<br>[10] = enables a 50 $\mu$ A current source on the selected negative input (for example, ADC1).<br>[11] = enables a 50 $\mu$ A current source on both selected inputs (for example, ADC0 and ADC1).  |
| 12    | HIGHEXTREF0   | This bit must be set high if the external reference for ADC0 exceeds 1.35 V. This results in the reference source being divided by 2.<br>Clear this bit when using the internal reference or an external reference of less than 1.35 V.  |
| 11    | AMP_CM        | This bit is set to 1 by user to set the PGA output common-mode voltage to AVDD/2.<br>This bit is cleared to 0 by user code to set the PGA output common-mode voltage to the PGA input common-mode voltage level.   |
| 10    | ADC0CODE      | Primary channel ADC output coding.<br>This bit is set to 1 by user code to configure primary ADC output coding as unipolar.<br>This bit is cleared to 0 by user code to configure primary ADC output coding as twos complement.  |
| 9:6   | ADC0CH[3:0]   | Primary channel ADC input select. Note that single-ended channels are selected with respect to ADC5. Bias ADC5 to a minimum level of 0.1 V.<br>[0000] = ADC0/ADC1 (differential mode).<br>[0001] = ADC0/ADC5 (single-ended mode).<br>[0010] = ADC1/ADC5 (single-ended mode).<br>[0011] = VREF+, VREF-. Note: This is the reference selected by the ADC0REF bits.<br>[0100] = Not used. This bit combination is reserved for future functionality and should not be written.<br>[0101] = ADC2/ADC3 (differential mode).<br>[0110] = ADC2/ADC5 (single-ended mode).<br>[0111] = ADC3/ADC5 (single-ended mode).<br>[1000] = internal short to ADC1.<br>[1001] = internal short to ADC1. |
| 5:4   | ADC0REF[1:0]  | Primary channel ADC reference select.<br>[00] = internal reference selected. In ADC low power mode, the voltage reference selection is controlled by ADCMDE[5].<br>[01] = external reference inputs (VREF+, VREF-) selected. Set the HIGHEXTREF0 bit if the reference voltage exceeds 1.3 V.<br>[10] = auxiliary external reference inputs (ADC4/EXT_REF2IN+, ADC5/EXT_REF2IN-) selected. Set the HIGHEXTREF0 bit if the reference voltage exceeds 1.3 V.<br>[11] = (AVDD, AGND) divide-by-two selected.   |
| 3:0   | ADC0PGA[3:0]. | Primary channel ADC gain select. Note, nominal primary ADC full-scale input voltage = (VREF/gain).<br>[0000] = ADC0 gain of 1. Buffer of negative input is bypassed.<br>[0001] = ADC0 gain of 2.<br>[0010] = ADC0 gain of 4 (default value). Enables the in-amp.<br>[0011] = ADC0 gain of 8.<br>[0100] = ADC0 gain of 16.<br>[0101] = ADC0 gain of 32.<br>[0110] = ADC0 gain of 64 (maximum PGA gain setting).<br>[0111] = ADC0 gain of 128 (extra gain implemented digitally).<br>[1000] = ADC0 gain of 256.<br>[1001] = ADC0 gain of 512.<br>[1XXX] = ADC0 gain is undefined.  |

**Auxiliary ADC Control Register**

Name: ADC1CON  
 Address: 0xFFFF0510  
 Default value: 0x0000  
 Access: Read and write  
 Function: The auxiliary ADC control MMR is a 16-bit register.

**Table 44. ADC1CON MMR Bit Designations**

| Bit   | Name          | Description  |
|-------|---------------|--|
| 15    | ADC1EN        | Auxiliary channel ADC enable.<br>This bit is set to 1 by user code to enable the auxiliary ADC.<br>Clearing this bit to 0 powers down the auxiliary ADC.   |
| 14:13 | ADC1DIAG[1:0] | Diagnostic current source enable bits. This is the same current source as that used on ADC0DIAG[1:0]. The ADCs cannot enable the diagnostic current sources at the same time.<br>[00]= current sources off.<br>[01] = enables a 50 $\mu$ A current source on selected positive input (for example, ADC2).<br>[10] = enables a 50 $\mu$ A current source on selected negative input (for example, ADC3).<br>[11] = enables a 50 $\mu$ A current source on both selected inputs (for example, ADC2 and ADC3).  |
| 12    | HIGHEXTREF1   | This bit must be set high if the external reference for ADC1 exceeds 1.35 V. This results in the reference source being divided by 2.<br>Clear this bit when using the internal reference or an external reference of less than 1.35 V.  |
| 11    | ADC1CODE      | Auxiliary channel ADC output coding.<br>This bit is set to 1 by user code to configure auxiliary ADC output coding as unipolar.<br>This bit is cleared to 0 by user code to configure auxiliary ADC output coding as twos complement.  |
| 10:7  | ADC1CH[3:0]   | Auxiliary channel ADC input select. Note: Single-ended channels are selected with respect to ADC5. Bias ADC5 to a minimum level of 0.1 V.<br>[0000] = ADC2/ADC3 (differential mode).<br>[0001] = ADC4/ADC5 (differential mode).<br>[0010] = ADC6/ADC7 (differential mode).<br>[0011] = ADC8/ADC9 (differential mode).<br>[0100] = ADC2/ADC5 (single-ended mode).<br>[0101] = ADC3/ADC5 (single-ended mode).<br>[0110] = ADC4/ADC5 (single-ended mode).<br>[0111] = ADC6/ADC5 (single-ended mode).<br>[1000] = ADC7/ADC5 (single-ended mode).<br>[1001] = ADC8/ADC5 (single-ended mode).<br>[1010] = ADC9/ADC5 (single-ended mode).<br>[1011] = internal temperature sensor+/internal temperature sensor-.<br>[1100] = VREF+, VREF-. Note: This is the reference selected by the ADC1REF bits.<br>[1101] = DAC_OUT/AGND.<br>[1110] = undefined.<br>[1111] = internal short to ADC3. |

| Bit | Name            | Description  |
|-----|-----------------|--|
| 6:4 | ADC1REF[2:0]    | Auxiliary channel ADC reference select.<br>[000] = internal reference selected. In ADC low power mode, the voltage reference selection is controlled by ADCMODE[5].<br>[001] = external reference inputs (VREF+, VREF-) selected. Set the HIGHEXTREF1 bit if reference voltage exceeds 1.3 V.<br>[010] = auxiliary external reference inputs (ADC4/EXT_REF2IN+, ADC5/EXT_REF2IN-) selected. Set the HIGHEXTREF1 bit if reference voltage exceeds 1.35 V.<br>[011] = (AVDD, AGND) divide-by-2 selected. If this configuration is selected, the HIGHEXTREF1 bit is set automatically.<br>[100] = (AVDD, ADC3). ADC3 can be used as the negative input terminal for the reference source.<br>[101] to [111] = reserved. |
| 3:2 | BUF_BYPASS[1:0] | Buffer bypass.<br>[00] = full buffer on. Both positive and negative buffer inputs active.<br>[01] = negative buffer is bypassed, positive buffer is on.<br>[10] = negative buffer is on, positive buffer is bypassed.<br>[11] = full buffer bypass. Both positive and negative buffer inputs are off.  |
| 1:0 |                 | Digital gain. Select for auxiliary ADC inputs.<br>[00] = ADC1 gain = 1.<br>[01] = ADC1 gain = 2.<br>[10] = ADC1 gain = 4.<br>[11] = ADC1 gain = 8.   |

**ADC Filter Register**

Name: ADCFLT

Address: 0xFFFF0514

Default value: 0x0007

Access: Read and write

Function: The ADC filter MMR is a 16-bit register that controls the speed and resolution of both the on-chip ADCs. Note that, if ADCFLT is modified, the primary and auxiliary ADCs are reset. When changing conversions speeds, put the ADC into idle mode before restarting.

**Table 45. ADCFLT MMR Bit Designations**

| Bit  | Name    | Description   |
|------|---------|---|
| 15   | CHOPEN  | Chop enable. Set by user to enable system chopping of all active ADCs. When this bit is set, the ADC has very low offset errors and drift, but the ADC output rate is reduced by a factor of 3 if AF = 0 (see sinc3 decimation factor, Bits[6:0] in this table). If AF > 0, then the ADC output update rate is the same with chop on or off. When chop is enabled, the settling time is two output periods.                                   |
| 14   | RAVG2   | Running average-by-2 enable bit.<br>Set by user to enable a running-average-by-2 function, reducing ADC noise. This function is automatically enabled when chopping is active. It is an optional feature when chopping is inactive, and if enabled (when chopping is inactive), does not reduce the ADC output rate but does increase the settling time by one conversion period.<br>Cleared by user to disable the running average function. |
| 13:8 | AF[5:0] | Averaging factor (AF). The values written to these bits are used to implement a programmable first-order sinc3 post filter. The averaging factor can further reduce ADC noise at the expense of output rate as described in Bits[6:0] (sinc3 decimation factor) in this table.  |



| Bit | Name    | Description   |
|-----|---------|---|
| 7   | NOTCH2  | Sinc3 modify. Set by user to modify the standard sinc3 frequency response to increase the filter stop-band rejection by approximately 5 dB. This is achieved by inserting a second notch (NOTCH2) at<br>$f_{NOTCH2} = 1.333 \times f_{NOTCH}$ where $f_{NOTCH}$ is the location of the first notch in the response.   |
| 6:0 | SF[6:0] | Sinc3 decimation factor (SF). <sup>1</sup> The value (SF) written in these bits controls the oversampling (decimation factor) of the sinc3 filter. The output rate from the sinc3 filter is given by<br>$f_{ADC} = (512,000 / ([SF + 1] \times 64)) \text{ Hz}^2$ when the chop bit (Bit 15, chop enable) = 0 and the averaging factor (AF) = 0. This is valid for all SF values $\leq 125$ .<br>For SF = 126, $f_{ADC}$ is forced to 60 Hz.<br>For SF = 127, $f_{ADC}$ is forced to 50 Hz.<br>For information on calculating the $f_{ADC}$ for SF (other than 126 and 127) and AF values, refer to Table 46. |

<sup>1</sup> Due to limitations on the digital filter internal data path, there are some limitations on the combinations of the sinc3 decimation factor (SF) and averaging factor (AF) that can be used to generate a required ADC output rate. This restriction limits the minimum ADC update in normal power mode to 4 Hz or 1 Hz in lower power mode.

<sup>2</sup> In low power mode, the ADC is driven directly by the low power oscillator (131 kHz) and not 512 kHz. All  $f_{ADC}$  calculations should be divided by 4 (approximately).

**Table 46. ADC Conversion Rates and Settling Times**

| Chop Enabled | Averaging Factor | Running Average | $f_{ADC}$ Normal Mode                                    | $f_{ADC}$ Low Power Mode                                 | $t_{SETTLING}^1$    |
|--------------|------------------|-----------------|--|--|---------------------|
| No           | No               | No              | $\frac{512,000}{[SF + 1] \times 64}$                     | $\frac{131,072}{[SF + 1] \times 64}$                     | $\frac{3}{f_{ADC}}$ |
| No           | No               | Yes             | $\frac{512,000}{[SF + 1] \times 64}$                     | $\frac{131,072}{[SF + 1] \times 64}$                     | $\frac{4}{f_{ADC}}$ |
| No           | Yes              | No              | $\frac{512,000}{[SF + 1] \times 64 \times [3 + AF]}$     | $\frac{131,072}{[SF + 1] \times 64 \times [3 + AF]}$     | $\frac{1}{f_{ADC}}$ |
| No           | Yes              | Yes             | $\frac{512,000}{[SF + 1] \times 64 \times [3 + AF]}$     | $\frac{131,072}{[SF + 1] \times 64 \times [3 + AF]}$     | $\frac{2}{f_{ADC}}$ |
| Yes          | N/A              | N/A             | $\frac{512,000}{[SF + 1] \times 64 \times [3 + AF] + 3}$ | $\frac{131,072}{[SF + 1] \times 64 \times [3 + AF] + 3}$ | $\frac{2}{f_{ADC}}$ |

<sup>1</sup> An additional time of approximately 60  $\mu$ s per ADC is required before the first ADC is available.

**Table 47. Allowable Combinations of SF and AF**

| SF        | AF Range |        |         |
|-----------|----------|--------|---------|
|           | 0        | 1 to 7 | 8 to 63 |
| 0 to 31   | Yes      | Yes    | Yes     |
| 32 to 63  | Yes      | Yes    | No      |
| 64 to 127 | Yes      | No     | No      |

**ADC Configuration Register**

Name: ADCCFG

Address: 0xFFFF0518

Default value: 0x00

Access: Read and write

Function: The 8-bit ADC configuration MMR controls extended functionality related to the on-chip ADCs.

**Table 48. ADCCFG MMR Bit Designations**

| Bit | Name           | Description   |
|-----|----------------|---|
| 7   | GND_SW_EN      | Analog ground switch enable.<br>This bit is set to 1 by user software to connect the external GND_SW pin to an internal analog ground reference point. This bit can be used to connect and disconnect external circuits and components to ground under program control and thereby minimize dc current consumption when the external circuit or component is not being used. This bit is used in conjunction with ADCCFG[1] to select a 20 kΩ resistor to ground.<br>When this bit is cleared, the analog ground switch is disconnected from the external pin.  |
| 6:5 | ADC0ACCEN[1:0] | Primary channel (32-bit) accumulator enable.<br>[00] = accumulator disabled and reset to 0. The accumulator must be disabled for a full ADC conversion (ADCSTA[0] set twice) before the accumulator can be re-enabled to ensure that the accumulator is reset.<br>[01] = accumulator active. Positive current values are added to the accumulator total; the accumulator can overflow if allowed to run for >65,535 conversions. Negative current values are subtracted from the accumulator total; the accumulator is clamped to a minimum value of 0.<br>[10] = accumulator active. Same as [01] except that there is no clamp. Positive current values are added to the accumulator total; the accumulator can overflow if allowed to run for >65,535 conversions. The absolute values of negative current are subtracted from the accumulator total; the accumulator in this mode continues to accumulate negatively, below 0.<br>[11] = accumulator and comparator active. This causes an ADC0 interrupt if ADCMSKI[6] is set. |
| 4:3 | ADC0CMPEN[1:0] | Primary ADC comparator enable bits.<br>[00] = comparator disabled.<br>[01] = comparator active. Interrupt asserted if absolute value of ADC0 conversion result $    \geq \text{ADC0TH}$ .<br>[10] = comparator count mode active. Interrupt asserted if absolute value of ADC0 conversion result $    \geq \text{ADC0TH}$ for the number of ADC0THC conversions. A conversion value $    < \text{ADC0TH}$ resets the threshold counter value (ADC0THV) to 0.<br>[11] = comparator count mode active, interrupt asserted if absolute value of ADC0 conversion result $    \geq \text{ADC0TH}$ for the number of ADC0THC conversions. A conversion value $    < \text{ADC0TH}$ decrements the threshold counter value (ADC0THV) toward 0.   |
| 2   | ADC0OREN       | ADC0 overrange enable.<br>Set by the user to enable a coarse comparator on the primary channel ADC. If the reading is grossly (>30% approximate) overrange for the active gain setting, the overrange bit in the ADCSTA MMR is set. The ADC reading must be outside this range for greater than 125 μs for the flag to be set.<br>Do not use this feature in ADC low power mode.  |
| 1   | GND_SW_RES_EN  | Set to 1 to enable a 20 kΩ resistor in series with the ground switch.<br>Clear this bit to disable this resistor.   |
| 0   | ADCR_CEN       | ADC result counter enable.<br>Set by user to enable the result count mode. ADC interrupts occur if ADC0RCR = ADC0RCV.<br>Cleared to disable the result counter. ADC interrupts occur after every conversion.  |

**Primary Channel ADC Data Register**

|                |  |
|----------------|--|
| Name:          | ADC0DAT  |
| Address:       | 0xFFFF051C   |
| Default value: | 0x00000000   |
| Access:        | Read only  |
| Function:      | This ADC data MMR holds the 24-bit conversion result from the primary ADC. The ADC does not update this MMR if the ADC0 conversion result ready bit (ADCSTA[0]) is set. A read of this MMR by the MCU clears all asserted ready flags (ADCSTA[1:0]). |

**Table 49. ADC0DAT MMR Bit Designations**

| Bit  | Description                    |
|------|--------------------------------|
| 23:0 | ADC0 24-bit conversion result. |

**Auxiliary Channel ADC Data Register**

|                |   |
|----------------|---|
| Name:          | ADC1DAT   |
| Address:       | 0xFFFF0520  |
| Default value: | 0x00000000  |
| Access:        | Read only   |
| Function:      | This ADC data MMR holds the 24-bit conversion result from the auxiliary ADC. The ADC does not update this MMR if the ADC0 conversion result ready bit (ADCSTA[1]) is set. |

**Table 50. ADC1DAT MMR Bit Designations**

| Bit  | Description                    |
|------|--------------------------------|
| 23:0 | ADC1 24-bit conversion result. |

**Primary Channel ADC Offset Calibration Register**

|                |   |
|----------------|---|
| Name:          | ADC0OF  |
| Address:       | 0xFFFF0524  |
| Default value: | Part specific, factory programmed   |
| Access:        | Read and write  |
| Function:      | This ADC offset MMR holds a 16-bit offset calibration coefficient for the primary ADC. The register is configured at power-on with a factory default value. However, this register automatically overwrites if an offset calibration of the primary ADC is initiated by the user via bits in the ADCMDE MMR. User code can write to this calibration register only if the ADC is in idle mode. An ADC must be enabled and in idle mode before being written to any offset or gain register. The ADC must be in idle mode for at least 23 $\mu$ s. |

**Table 51. ADC0OF MMR Bit Designations**

| Bit  | Description                           |
|------|---------------------------------------|
| 15:0 | ADC0 16-bit offset calibration value. |

**Auxiliary Channel ADC Offset Calibration Register**

|                |   |
|----------------|---|
| Name:          | ADC1OF  |
| Address:       | 0xFFFF0528  |
| Default value: | Part specific, factory programmed   |
| Access:        | Read and write  |
| Function:      | This offset MMR holds a 16-bit offset calibration coefficient for the auxiliary channel. The register is configured at power-on with a factory default value. However, this register is automatically overwritten if an offset calibration of the auxiliary channel is initiated by the user via bits in the ADCMDE MMR. User code can write to this calibration register only if the ADC is in idle mode. An ADC must be enabled and in idle mode before being written to any offset or gain register. The ADC must be in idle mode for at least 23 $\mu$ s. |

**Table 52. ADC1OF MMR Bit Designations**

| Bit  | Description                           |
|------|---------------------------------------|
| 15:0 | ADC1 16-bit offset calibration value. |

**Primary Channel ADC Gain Calibration Register**

|                |  |
|----------------|--|
| Name:          | ADC0GN   |
| Address:       | 0xFFFF052C   |
| Default value: | Part specific, factory programmed  |
| Access:        | Read and write   |
| Function:      | This gain MMR holds a 16-bit gain calibration coefficient for scaling the primary ADC conversion result. The register is configured at power-on with a factory default value. However, this register is automatically overwritten if a gain calibration of the primary ADC is initiated by the user via bits in the ADCMDE MMR. User code can write to this calibration register only if the ADC is in idle mode. An ADC must be enabled and in idle mode before being written to any offset or gain register. The ADC must be in idle mode for at least 23 $\mu$ s. |

**Table 53. ADC0GN MMR Bit Designations**

| Bits | Description                         |
|------|-------------------------------------|
| 15:0 | ADC0 16-bit calibration gain value. |

**Auxiliary Channel Gain Calibration Register**

|                |   |
|----------------|---|
| Name:          | ADC1GN  |
| Address:       | 0xFFFF0530  |
| Default value: | Part specific, factory programmed   |
| Access:        | Read and write  |
| Function:      | This gain MMR holds a 16-bit gain calibration coefficient for scaling an auxiliary channel conversion result. The register is configured at power-on with a factory default value. However, this register is automatically overwritten if a gain calibration of the auxiliary channel is initiated by the user via bits in the ADCMDE MMR. User code can write to this calibration register only if the ADC is in idle mode. An ADC must be enabled and in idle mode before being written to any offset or gain register. The ADC must be in idle mode for at least 23 $\mu$ s. |

**Table 54. ADC1GN MMR Bit Designations**

| Bits | Description                         |
|------|-------------------------------------|
| 15:0 | ADC1 16-bit gain calibration value. |

**Primary Channel ADC Result Counter Limit Register**

|                |   |
|----------------|---|
| Name:          | ADC0RCR   |
| Address:       | 0xFFFF0534  |
| Default value: | 0x0001  |
| Access:        | Read and write  |
| Function:      | This 16-bit MMR sets the number of conversions required before an ADC interrupt is generated. By default, this register is set to 0x01. The ADC counter function must be enabled via the ADC result counter enable bit in the ADCCFG MMR. |

**Table 55. ADC0RCR MMR Bit Designations**

| Bits | Description                                |
|------|--|
| 15:0 | ADC0 result counter limit/reload register. |

**Primary Channel ADC Result Counter Register**

|                |   |
|----------------|---|
| Name:          | ADC0RCV   |
| Address:       | 0xFFFF0538  |
| Default value: | 0x0000  |
| Access:        | Read only   |
| Function:      | This 16-bit, read-only MMR holds the current number of primary ADC conversion results. It is used in conjunction with ADC0RCR to mask primary channel ADC interrupts, generating a lower interrupt rate. When ADC0RCV = ADC0RCR, the value in ADC0RCV resets to 0 and recommences counting. It can also be used in conjunction with the accumulator (ADC0ACC) to allow an average calculation to be taken. The result counter is enabled via ADCCFG[0]. This MMR is also reset to 0 when the primary ADC is reconfigured, that is, when the ADC0CON or ADCMDE is written. |

**Table 56. ADC0RCV MMR Bit Designations**

| Bits | Description                   |
|------|-------------------------------|
| 15:0 | ADC0 result counter register. |

**Primary Channel ADC Threshold Register**

Name: ADC0TH  
 Address: 0xFFFF053C  
 Default value: 0x0000  
 Access: Read and write  
 Function: This 16-bit MMR sets the threshold against which the absolute value of the primary ADC conversion result is compared. In unipolar mode, ADC0TH[15:0] are compared, and in twos complement mode, ADC0TH[14:0] are compared.

**Table 57. ADC0TH MMR Bit Designations**

| Bit  | Description                                |
|------|--|
| 15:0 | ADC0 16-bit comparator threshold register. |

**Primary Channel ADC Threshold Counter Limit Register**

Name: ADC0THC  
 Address: 0xFFFF0540  
 Default value: 0x0001  
 Access: Read and write  
 Function: This 8-bit MMR determines how many cumulative (values below the threshold decrement or reset the count to 0) primary ADC conversion result readings above ADC0TH must occur before the primary ADC comparator threshold bit is set in the ADCSTA MMR, generating an ADC interrupt. The primary ADC comparator threshold bit is asserted as soon as ADC0THV = ADC0RCR.

**Table 58. ADC0THC MMR Bit Designations**

| Bit  | Description                                  |
|------|--|
| 15:8 | Reserved.                                    |
| 7:0  | ADC0 8-bit threshold counter limit register. |

**Primary Channel ADC Threshold Counter Register**

Name: ADC0THV  
 Address: 0xFFFF0544  
 Default value: 0x0000  
 Access: Read only  
 Function: This 8-bit MMR is incremented every time the absolute value of a primary ADC conversion result  $|\text{Result}| \geq \text{ADC0TH}$ . This register is decremented or reset to 0 every time the absolute value of a primary ADC conversion result  $|\text{Result}| < \text{ADC0TH}$ . The configuration of this function is enabled via the primary channel ADC comparator bits in the ADCCFG MMR.

**Table 59. ADC0THV MMR Bit Designations**

| Bit | Description                                     |
|-----|---|
| 7:0 | ADC0 8-bit threshold exceeded counter register. |

**Primary Channel ADC Accumulator Register**

Name: ADC0ACC  
 Address: 0xFFFF0548  
 Default value: 0x00000000  
 Access: Read only  
 Function: This 32-bit MMR holds the primary ADC accumulator value. The primary ADC ready bit in the ADCSTA MMR should be used to determine when it is safe to read this MMR. The MMR value is reset to 0 by disabling the accumulator in the ADCCFG MMR or by reconfiguring the primary channel ADC.

**Table 60. ADC0ACC MMR Bit Designations**

| Bit  | Description                       |
|------|-----------------------------------|
| 31:0 | ADC0 32-bit accumulator register. |

## Primary Channel ADC Comparator Threshold Register

Name: ADC0ATH

Address: 0xFFFF054C

Default value: 0x00000000

Access: Read and write

Function: This 32-bit MMR holds the threshold value for the accumulator comparator of the primary channel. When the accumulator value in ADC0ACC exceeds the value in ADC0ATH, the ADC0ATHEX bit in ADCSTA is set. This causes an interrupt if the corresponding bit in ADCMSKI is also enabled.

Table 61. ADC0ATH MMR Bit Designations

| Bit  | Description   |
|------|---|
| 31:0 | ADC0 32-bit comparator threshold register of the accumulator. |

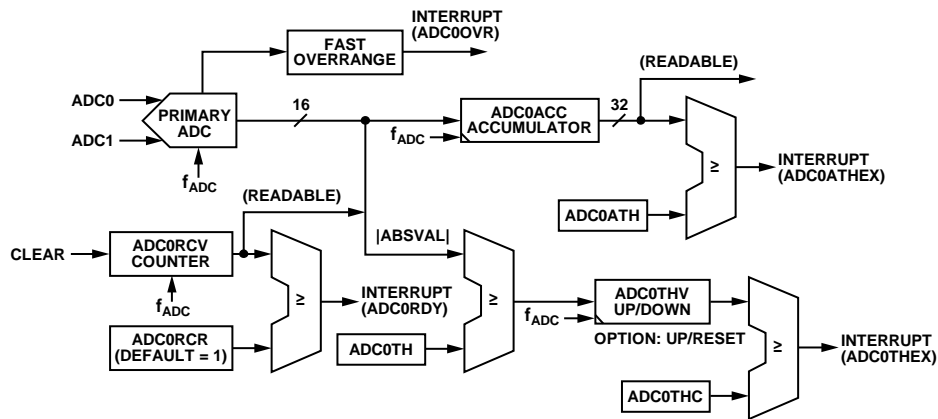


Figure 17. Primary ADC Accumulator/Comparator/Counter Block Diagram

07079-011

**Excitation Current Sources Control Register**

Name: IEXCON

Address: 0xFFFF0570

Default value: 0x00

Access: Read and write

Function: This 8-bit MMR controls the two excitation current sources, IEXC0 and IEXC1.

**Table 62. IEXCON MMR Bit Designations**

| Bit | Name      | Description  |
|-----|-----------|--|
| 7   | IEXC1_EN  | Enable bit for IEXC1 current source.<br>Set this bit to 1 to enable Excitation Current Source 1.<br>Clear this bit to disable Excitation Current Source 1.   |
| 6   | IEXC0_EN  | Enable bit for IEXC0 current source.<br>Set this bit to 1 to enable Excitation Current Source 0.<br>Clear this bit to disable Excitation Current Source 0.   |
| 5   | IEXC1_DIR | Set this bit to 1 to direct Excitation Current Source 1 to the IEXC0 pin.<br>Set this bit to 0 to direct Excitation Current Source 1 to the IEXC1 pin.   |
| 4   | IEXC0_DIR | Set this bit to 1 to direct Excitation Current Source 0 to the IEXC1 pin.<br>Set this bit to 0 to direct Excitation Current Source 0 to the IEXC0 pin.   |
| 3:1 | IOUT[3:1] | These bits control the excitation current level for each source.<br>IOUT[3:1] = 000, excitation current = 0 $\mu$ A + (IOUT[0] $\times$ 10 $\mu$ A).<br>IOUT[3:1] = 001, excitation current = 200 $\mu$ A + (IOUT[0] $\times$ 10 $\mu$ A).<br>IOUT[3:1] = 010, excitation current = 400 $\mu$ A + (IOUT[0] $\times$ 10 $\mu$ A).<br>IOUT[3:1] = 011, excitation current = 600 $\mu$ A + (IOUT[0] $\times$ 10 $\mu$ A).<br>IOUT[3:1] = 100, excitation current = 800 $\mu$ A + (IOUT[0] $\times$ 10 $\mu$ A).<br>IOUT[3:1] = 101, excitation current = 1 mA + (IOUT[0] $\times$ 10 $\mu$ A).<br>All other values are undefined. |
| 0   | IOUT[0]   | Set this bit to 1 to enable 10 $\mu$ A diagnostic current source.<br>Clear this bit to 0 to disable 10 $\mu$ A diagnostic current source.  |

**EXAMPLE APPLICATION CIRCUITS**

Figure 18 shows a simple bridge sensor interface to the [ADuC7060/ADuC7061](#), including the RC filters on the analog input channels. Notice that the sense lines from the bridge (connecting to the reference inputs) are wired separately from the excitation lines (going to DVDD/AVDD and ground). This results in a total of six wires going to the bridge. This 6-wire connection scheme is a feature of most off-the-shelf bridge transducers (such as load cells) that helps to minimize errors that would otherwise result from wire impedances.

In Figure 19, the AD592 is an external temperature sensor used to measure the thermocouple cold junction, and its output is connected to the auxiliary channel. The ADR280 is an external 1.2 V reference part—alternatively, the internal reference can be used. Here, the thermocouple is connected to the primary ADC as a differential input to ADC0/ADC1. Note the resistor between VREF+ and ADC1 to bias the ADC inputs above 100 mV.

Figure 20 shows a simple 4-wire RTD interface circuit. As with the bridge transducer implementation in Figure 18, if a power supply and a serial connection to the outside world are added, Figure 20 represents a complete system.

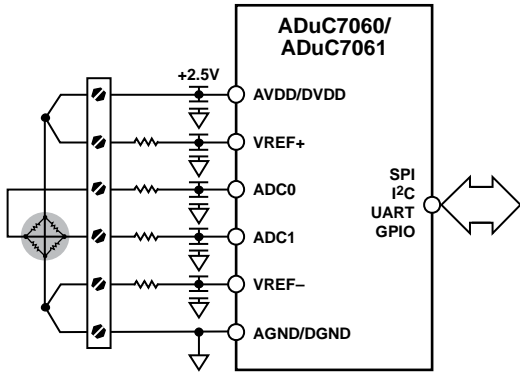


Figure 18. Bridge Interface Circuit

07079-012

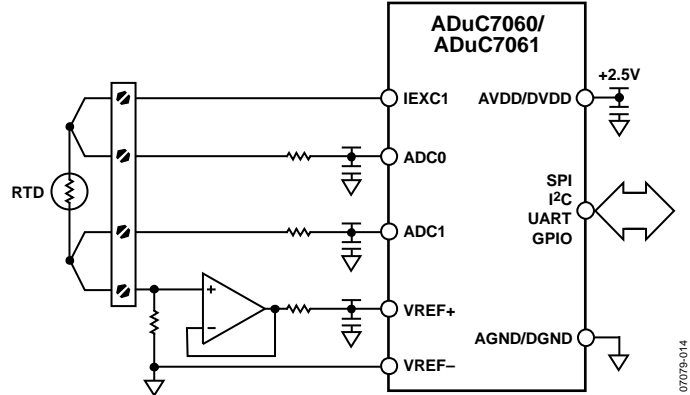


Figure 20. Example of an RTD Interface Circuit

07079-014

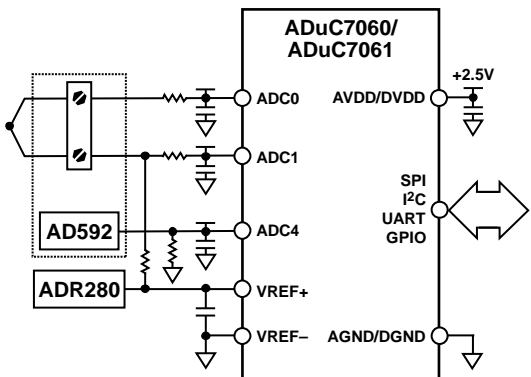


Figure 19. Example of a Thermocouple Interface Circuit

07079-013



## DAC PERIPHERALS

### DAC

The ADuC7060/ADuC7061 incorporates a voltage output DAC on chip. In normal mode, the DAC resolution is 12-bits. In interpolation, the DAC resolution is 16 bits with 14 effective bits. The DAC has a rail-to-rail voltage output buffer capable of driving 5 k $\Omega$ /100 pF.

The DAC has four selectable ranges.

- 0 V to V<sub>REF</sub> (internal band gap 1.2 V reference)
- VREF– to VREF+
- ADC5/EXT\_REF2IN– to ADC4/EXT\_REF2IN+
- 0 V to AVDD

The maximum signal range is 0 V to AVDD.

### Op Amp Mode

As an option, the DAC can be disabled and its output buffer used as an op amp.

### MMR INTERFACE

The DAC is configurable through a control register and a data register.

#### DAC0CON Register

|                |                |
|----------------|----------------|
| Name:          | DAC0CON        |
| Address:       | 0xFFFF0600     |
| Default value: | 0x0200         |
| Access:        | Read and write |

**Table 63. DAC0CON MMR Bit Designations**

| Bit   | Name           | Description   |
|-------|----------------|---|
| 15:10 |                | Reserved.   |
| 9     | DACPD          | Set to 1 to power down DAC output (DAC output is tristated).<br>Clear this bit to enable the DAC.   |
| 8     | DACBUFLP       | Set to 1 to place the DAC output buffer in low power mode. See the Normal DAC Mode and Op Amp Mode sections for further details on electrical specifications.<br>Clear this bit to enable the DAC buffer.   |
| 7     | OPAMP          | Set to 1 to place the DAC output buffer in op amp mode.<br>Clear this bit to enable the DAC output buffer for normal DAC operation.   |
| 6     | DACBUFBYPASS   | Set to 1 to bypass the output buffer and send the DAC output directly to the output pin.<br>Clear this bit to buffer the DAC output.  |
| 5     | DACCLK         | Cleared to 0 to update the DAC on the negative edge of HCLK.<br>Set to 1 to update the DAC on the negative edge of Timer0. This mode is ideally suited for waveform generation where the next value in the waveform is written to DAC0DAT at regular intervals of Timer0. |
| 4     | DACCLR         | Set to 1 for normal DAC operation.<br>Set to 0 to clear the DAC output and to set DAC0DAT to 0. Writing to this bit has an immediate effect on the DAC output.  |
| 3     | DACMODE        | Set to 1 to enable the DAC in 16-bit interpolation mode.<br>Set to 0 to enable the DAC in normal 12-bit mode.   |
| 2     | Rate           | Used with interpolation mode.<br>Set to 1 to configure the interpolation clock as UCLK/16.<br>Set to 0 to configure the interpolation clock as UCLK/32.   |
| 1:0   | DAC range bits | [11] = 0 V to AVDD range.<br>[10] = ADC5/EXT_REF2IN– to ADC4/EXT_REF2IN+.<br>[01] = VREF– to VREF+.<br>[00] = 0 V to VREF (1.2 V) range. Internal reference source.   |

**DAC0DAT Register**

|                |  |
|----------------|--|
| Name:          | DAC0DAT  |
| Address:       | 0xFFFF0604                                     |
| Default value: | 0x00000000                                     |
| Access:        | Read and write                                 |
| Function:      | This 32-bit MMR contains the DAC output value. |

**Table 64. DAC0DAT MMR Bit Designations**

| Bit   | Description                                 |
|-------|---|
| 31:28 | Reserved.                                   |
| 27:16 | 12-bit data for DAC0.                       |
| 15:12 | Extra four bits used in interpolation mode. |
| 11:0  | Reserved.                                   |

**USING THE DAC**

The on-chip DAC architecture consists of a resistor string DAC followed by an output buffer amplifier.

The reference source for the DAC is user selectable in software. It can be AVDD, VREF±, or ADCx/EXT\_REF2IN±.

- In 0-to-AVDD mode, the DAC output transfer function spans from 0 V to the voltage at the AVDD pin.
- In VREF± and ADCx/EXT\_REF2IN± modes, the DAC output transfer function spans from negative input voltage to the voltage positive input pin. Note that these voltages must never go below 0 V or above AVDD.
- In 0-to-VREF mode, the DAC output transfer function spans from 0 V to the internal 1.2 V reference, VREF.

The DAC can be configured in three different user modes: normal mode, DAC interpolation mode, and op amp mode.

**Normal DAC Mode**

In this mode of operation, the DAC is configured as a 12-bit voltage output DAC. By default, the DAC buffer is enabled, but the output buffer can be disabled. If the DAC output buffer is disabled, the DAC is capable of driving a capacitive load of only 20 pF. The DAC buffer is disabled by setting the DACBUFYPASS bit in DAC0CON.

The DAC output buffer amplifier features a true, rail-to-rail output stage implementation. This means that when unloaded, each output is capable of swinging to within less than 5 mV of both AVDD and ground. Moreover, the linearity specification of the DAC (when driving a 5 kΩ resistive load to ground) is guaranteed through the full transfer function except for Code 0 to Code 100 and, in 0-to-AVDD mode only, Code 3995 to

Code 4095. Linearity degradation near ground and AVDD is caused by saturation of the output amplifier, and a general representation of its effects (neglecting offset and gain error) is illustrated in Figure 21. The dotted line in Figure 21 indicates the ideal transfer function, and the solid line represents what the transfer function may look like with endpoint nonlinearities due to saturation of the output amplifier. Note that Figure 21 represents a transfer function in 0-to-AVDD mode only. In 0-to-VREF or, VREF±, and ADCx/EXT\_REF2IN± modes (with VREF < AVDD or ADCx/EXT\_REF2IN± < AVDD), the lower nonlinearity is similar. However, the upper portion of the transfer function follows the ideal line all the way to the end (VREF in this case, not AVDD), showing no signs of endpoint linearity errors.

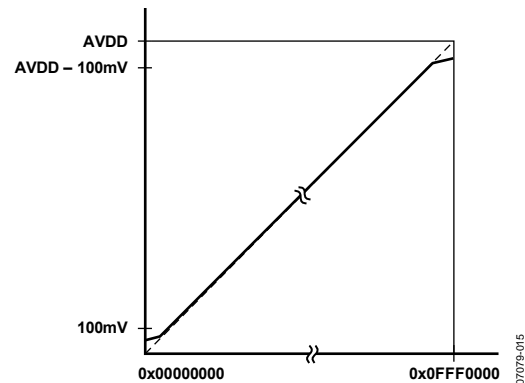


Figure 21. Endpoint Nonlinearities Due to Amplifier Saturation

The endpoint nonlinearities conceptually illustrated in Figure 21 worsen as a function of output loading. Most of the ADuC7060/ADuC7061 data sheet specifications in normal mode assume a 5 kΩ resistive load to ground at the DAC output. As the output is forced to source or sink more current, the nonlinear regions at the top or bottom (respectively) of Figure 21 become larger. With larger current demands, this can significantly limit output voltage swing.

**DAC Interpolation Mode**

In interpolation mode, a higher DAC output resolution of 16 bits is achieved with a longer update rate than normal mode. The update rate is controlled by the interpolation clock rate selected in the DAC0CON register. In this mode, an external RC filter is required to create a constant voltage.

**Op Amp Mode**

In op amp mode, the DAC output buffer is used as an op amp with the DAC itself disabled.

ADC6 is the positive input to the op amp, ADC7 is the negative input, and ADC8 is the output. In this mode, the DAC should be powered down by setting Bit 9 of DAC0CON.

## NONVOLATILE FLASH/EE MEMORY

The ADuC7060/ADuC7061 incorporates Flash/EE memory technology on chip to provide the user with nonvolatile, in-circuit reprogrammable memory space.

Like EEPROM, flash memory can be programmed in-system at a byte level, although it must first be erased. The erase is performed in page blocks. As a result, flash memory is often and, more correctly, referred to as Flash/EE memory.

Overall, Flash/EE memory represents a step closer to the ideal memory device that includes nonvolatility, in-circuit programmability, high density, and low cost. Incorporated in the ADuC7060/ADuC7061, Flash/EE memory technology allows the user to update program code space in-circuit, without the need to replace one time programmable (OTP) devices at remote operating nodes.

The ADuC7060/ADuC7061 contains a 32 kB array of Flash/EE memory. The lower 30 kB are available to the user and the upper 2 kB contain permanently embedded firmware, allowing in-circuit serial download. These 2 kB of embedded firmware also contain a power-on configuration routine that downloads factory-calibrated coefficients to the various calibrated peripherals (such as ADC, temperature sensor, and band gap references). This 2 kB embedded firmware is hidden from user code.

### FLASH/EE MEMORY RELIABILITY

The Flash/EE memory arrays on the parts are fully qualified for two key Flash/EE memory characteristics: Flash/EE memory cycling endurance and Flash/EE memory data retention.

Endurance quantifies the ability of the Flash/EE memory to be cycled through many program, read, and erase cycles. A single endurance cycle is composed of four independent, sequential events, defined as

- Initial page erase sequence
- Read/verify sequence for a single Flash/EE
- Byte program sequence memory
- Second read/verify sequence endurance cycle

In reliability qualification, every half word (16-bit wide) location of the three pages (top, middle, and bottom) in the Flash/EE memory is cycled 10,000 times from 0x0000 to 0xFFFF. The Flash/EE memory endurance qualification is carried out in accordance with JEDEC *Retention Lifetime Specification A117* over the industrial temperature range of  $-40^{\circ}\text{C}$  to  $+125^{\circ}\text{C}$ . The results allow the specification of a minimum endurance figure over a supply temperature of 10,000 cycles.

Retention quantifies the ability of the Flash/EE memory to retain its programmed data over time. Again, the parts are qualified in accordance with the formal *JEDEC Retention Lifetime Specification A117* at a specific junction temperature ( $T_j = 85^{\circ}\text{C}$ ). As part of this qualification procedure, the Flash/EE memory is cycled to its specified endurance limit, described previously, before data retention is characterized. This means that the Flash/EE memory is guaranteed to retain its data for its fully specified retention lifetime every time that the Flash/EE memory is reprogrammed. Also note that retention lifetime, based on activation energy of 0.6 eV, derates with  $T_j$ , as shown in Figure 22.

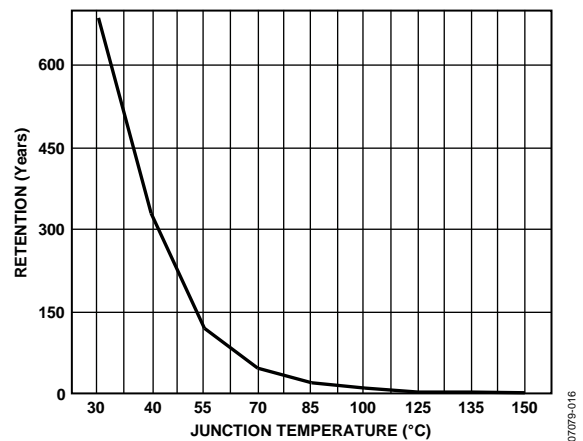


Figure 22. Flash/EE Memory Data Retention

### PROGRAMMING

The 30 kB of Flash/EE memory can be programmed in-circuit, using the serial download mode or the provided JTAG mode.

#### Serial Downloading (In-Circuit Programming)

The ADuC7060/ADuC7061 facilitates code download via the standard UART serial port. The parts enter serial download mode after a reset or power cycle if the NTRST/BM pin is pulled low through an external 1 k $\Omega$  resistor. When in serial download mode, the user can download code to the full 30 kB of Flash/EE memory while the device is in-circuit in its target application hardware. An executable PC serial download is provided as part of the development system for serial downloading via the UART.

When the ADuC7060/ADuC7061 enters download mode, the user should be aware that the internal watchdog is enabled with a time-out period of 2 minutes. If the flash erase/write sequence is not completed in this period, a reset occurs.

#### JTAG Access

The JTAG protocol uses the on-chip JTAG interface to facilitate code download and debug.

## PROCESSOR REFERENCE PERIPHERALS

### INTERRUPT SYSTEM

There are 15 interrupt sources on the [ADuC7060/ADuC7061](#) that are controlled by the interrupt controller. All interrupts are generated from the on-chip peripherals, except for the software interrupt (SWI), which is programmable by the user. The ARM7TDMI CPU core recognizes interrupts as one of two types only: a normal interrupt request (IRQ) or a fast interrupt request (FIQ). All the interrupts can be masked separately.

The control and configuration of the interrupt system are managed through a number of interrupt related registers. The bits in each IRQ and FIQ register represent the same interrupt source, as described in Table 65.

Each [ADuC7060/ADuC7061](#) contains a vectored interrupt controller (VIC) that supports nested interrupts up to eight levels. The VIC also allows the programmer to assign priority levels to all interrupt sources. Interrupt nesting needs to be enabled by setting the ENIRQN bit in the IRQCONN register. A number of extra MMRs are used when the full vectored interrupt controller is enabled.

Immediately save IRQSTA/FIQSTA upon entering the interrupt service routine (ISR) to ensure that all valid interrupt sources are serviced.

**Table 65. IRQ/FIQ MMR Bit Designations**

| Bit | Description                     | Comments                                     |
|-----|---------------------------------|--|
| 0   | All interrupts OR'ed (FIQ only) | This bit is set if any FIQ is active         |
| 1   | Software interrupt              | User programmable interrupt source           |
| 2   | Undefined                       | This bit is not used                         |
| 3   | Timer0                          | General-Purpose Timer0                       |
| 4   | Timer1 or wake-up timer         | General-Purpose Timer1 or wake-up timer      |
| 5   | Timer2 or watchdog timer        | General-Purpose Timer2 or watchdog timer     |
| 6   | Timer3 or STI timer             | General-Purpose Timer3                       |
| 7   | Undefined                       | This bit is not used                         |
| 8   | Undefined                       | This bit is not used                         |
| 9   | Undefined                       | This bit is not used                         |
| 10  | ADC                             | ADC interrupt source bit                     |
| 11  | UART                            | UART interrupt source bit                    |
| 12  | SPI                             | SPI interrupt source bit                     |
| 13  | XIRQ0 (GPIO IRQ0)               | External Interrupt 0                         |
| 14  | XIRQ1 (GPIO IRQ1)               | External Interrupt 1                         |
| 15  | I <sup>2</sup> C master IRQ     | I <sup>2</sup> C master interrupt source bit |
| 16  | I <sup>2</sup> C slave IRQ      | I <sup>2</sup> C slave interrupt source bit  |
| 17  | PWM                             | PWM trip interrupt source bit                |
| 18  | XIRQ2 (GPIO IRQ2)               | External Interrupt 2                         |
| 19  | XIRQ3 (GPIO IRQ3)               | External Interrupt 3                         |

### IRQ

The IRQ is the exception signal to enter the IRQ mode of the processor. It services general-purpose interrupt handling of internal and external events.

All 32 bits are logically OR'ed to create a single IRQ signal to the ARM7TDMI core. The four 32-bit registers dedicated to IRQ are described in the following sections.

### IRQSIG

IRQSIG reflects the status of the different IRQ sources. If a peripheral generates an IRQ signal, the corresponding bit in the IRQSIG is set; otherwise, it is cleared. The IRQSIG bits clear when the interrupt in the particular peripheral is cleared. All IRQ sources can be masked in the IRQEN MMR. IRQSIG is read only.

#### IRQSIG Register

|                |            |
|----------------|------------|
| Name:          | IRQSIG     |
| Address:       | 0xFFFF0004 |
| Default value: | Undefined  |
| Access:        | Read only  |

### IRQEN

IRQEN provides the value of the current enable mask. When a bit is set to 1, the corresponding source request is enabled to create an IRQ exception. The IRQEN register cannot be used to disable an interrupt. Clear to 0 has no effect.

#### IRQEN Register

|                |                |
|----------------|----------------|
| Name:          | IRQEN          |
| Address:       | 0xFFFF0008     |
| Default value: | 0x00000000     |
| Access:        | Read and write |

### IRQCLR

IRQCLR is a write-only register that allows the IRQEN register to clear to mask an interrupt source. Each bit that is set to 1 clears the corresponding bit in the IRQEN register without affecting the remaining bits. The pair of registers, IRQEN and IRQCLR, allows independent manipulation of the enable mask without requiring an atomic read-modify-write. Clear to 0 has no effect.

**IRQCLR Register**

|                |            |
|----------------|------------|
| Name:          | IRQCLR     |
| Address:       | 0xFFFF000C |
| Default value: | 0x00000000 |
| Access:        | Write only |

**IRQSTA**

IRQSTA is a read-only register that provides the current enabled IRQ source status (effectively a logic AND of the IRQSIG and IRQEN bits). When set to 1, that source generates an active IRQ request to the ARM7TDMI core. There is no priority encoder or interrupt vector generation. This function is implemented in software in a common interrupt handler routine.

**IRQSTA Register**

|                |            |
|----------------|------------|
| Name:          | IRQSTA     |
| Address:       | 0xFFFF0000 |
| Default value: | 0x00000000 |
| Access:        | Read only  |

**FAST INTERRUPT REQUEST (FIQ)**

The fast interrupt request (FIQ) is the exception signal to enter the FIQ mode of the processor. It is provided to service data transfer or communication channel tasks with low latency. The FIQ interface is identical to the IRQ interface and provides the second level interrupt (highest priority). Four 32-bit registers are dedicated to FIQ: FIQSIG, FIQEN, FIQCLR, and FIQSTA.

Bit 31 to Bit 1 of FIQSTA are logically OR'ed to create the FIQ signal to the core and to Bit 0 of both the FIQ and IRQ registers (FIQ source).

The logic for FIQEN and FIQCLR does not allow an interrupt source to be enabled in both IRQ and FIQ masks. A bit set to 1 in FIQEN clears, as a side effect, the same bit in IRQEN.

Likewise, a bit set to 1 in IRQEN clears, as a side effect, the same bit in FIQEN. An interrupt source can be disabled in both IRQEN and FIQEN masks.

**FIQSIG**

FIQSIG reflects the status of the different FIQ sources. If a peripheral generates an FIQ signal, the corresponding bit in the FIQSIG is set; otherwise, it is cleared. The FIQSIG bits are cleared when the interrupt in the particular peripheral is cleared. All FIQ sources can be masked in the FIQEN MMR. FIQSIG is read only.

**FIQSIG Register**

|                |            |
|----------------|------------|
| Name:          | FIQSIG     |
| Address:       | 0xFFFF0104 |
| Default value: | Undefined  |
| Access:        | Read only  |

**FIQEN**

FIQEN provides the value of the current enable mask. When a bit is set to 1, the corresponding source request is enabled to create an FIQ exception. When a bit is set to 0, the corresponding source request is disabled or masked, which does not create an FIQ exception. The FIQEN register cannot be used to disable an interrupt.

**FIQEN Register**

|                |                |
|----------------|----------------|
| Name:          | FIQEN          |
| Address:       | 0xFFFF0108     |
| Default value: | 0x00000000     |
| Access:        | Read and write |

**FIQCLR**

FIQCLR is a write-only register that allows the FIQEN register to clear in order to mask an interrupt source. Each bit that is set to 1 clears the corresponding bit in the FIQEN register without affecting the remaining bits. The pair of registers, FIQEN and FIQCLR, allows independent manipulation of the enable mask without requiring an atomic read-modify-write.

**FIQCLR Register**

|                |            |
|----------------|------------|
| Name:          | FIQCLR     |
| Address:       | 0xFFFF010C |
| Default value: | 0x00000000 |
| Access:        | Write only |

**FIQSTA**

FIQSTA is a read-only register that provides the current enabled FIQ source status (effectively a logic AND of the FIQSIG and FIQEN bits). When set to 1, that source generates an active FIQ request to the ARM7TDMI core. There is no priority encoder or interrupt vector generation. This function is implemented in software in a common interrupt handler routine.

**FIQSTA Register**

Name: FIQSTA  
 Address: 0xFFFF0100  
 Default value: 0x00000000  
 Access: Read only

**PROGRAMMED INTERRUPTS**

Because the programmed interrupts are not maskable, they are controlled by another register (SWICFG) that writes into both IRQSTA and IRQSIG registers and/or the FIQSTA and FIQSIG registers at the same time.

**SWICFG**

SWICFG is a 32-bit register dedicated to software interrupt, described in Table 66. This MMR allows control of a programmed source interrupt.

**SWICFG Register**

Name: SWICFG  
 Address: 0xFFFF0010  
 Default value: 0x00000000  
 Access: Write only

**Table 66. SWICFG MMR Bit Designations**

| Bit  | Description   |
|------|---|
| 31:3 | Reserved.   |
| 2    | Programmed interrupt FIQ. Setting/clearing this bit corresponds to setting/clearing Bit 1 of FIQSTA and FIQSIG. |
| 1    | Programmed interrupt IRQ. Setting/clearing this bit corresponds to setting/clearing Bit 1 of IRQSTA and IRQSIG. |
| 0    | Reserved.   |

Any interrupt signal must be active for at least the minimum interrupt latency time to be detected by the interrupt controller and to be detected by the user in the IRQSTA/FIQSTA register.

**VECTORED INTERRUPT CONTROLLER (VIC)**

Each ADuC7060/ADuC7061 incorporates an enhanced interrupt control system or vectored interrupt controller. The vectored interrupt controller for IRQ interrupt sources is enabled by setting Bit 0 of the IRQCONN register. Similarly, Bit 1 of IRQCONN enables the vectored interrupt controller for the FIQ interrupt sources. The vectored interrupt controller provides the following enhancements to the standard IRQ/FIQ interrupts:

- Vectored interrupts—allows a user to define separate interrupt service routine addresses for every interrupt source. This is achieved by using the IRQBASE and IRQVEC registers.
- IRQ/FIQ interrupts—can be nested up to eight levels depending on the priority settings. An FIQ still has a higher priority than an IRQ. Therefore, if the VIC is enabled for both the FIQ and IRQ and prioritization is maximized, it is possible to have 16 separate interrupt levels.
- Programmable interrupt priorities—using the IRQP0 to IRQP2 registers, an interrupt source can be assigned an interrupt priority level value from 0 to 7.

**VIC MMRS**

**IRQBASE**

The vector base register, IRQBASE, is used to point to the start address of memory used to store 32 pointer addresses. These pointer addresses are the addresses of the individual interrupt service routines.

**IRQBASE Register**

Name: IRQBASE  
 Address: 0xFFFF0014  
 Default value: 0x00000000  
 Access: Read and write

**Table 67. IRQBASE MMR Bit Designations**

| Bit   | Access    | Initial Value | Description          |
|-------|-----------|---------------|----------------------|
| 31:16 | Read only | Reserved      | Always read as 0.    |
| 15:0  | R/W       | 0             | Vector base address. |

**IRQVEC**

The IRQ interrupt vector register, IRQVEC, points to a memory address containing a pointer to the interrupt service routine of the currently active IRQ. This register should be read only when an IRQ occurs and IRQ interrupt nesting has been enabled by setting Bit 0 of the IRQCONN register.

**IRQVEC Register**

Name: IRQVEC  
 Address: 0xFFFF001C  
 Default value: 0x00000000  
 Access: Read only

Table 68. IRQVEC MMR Bit Designations

| Bit   | Access    | Initial Value | Description  |
|-------|-----------|---------------|--|
| 31:23 | Read only | 0             | Always read as 0.  |
| 22:7  | Read only | 0             | IRQBASE register value.  |
| 6:2   | Read only | 0             | Highest priority IRQ source. This is a value between 0 to 19 representing the possible interrupt sources. For example, if the highest currently active IRQ is Timer1, then these bits are [01000]. |
| 1:0   | Reserved  | 0             | Reserved bits.   |

### Priority Registers

The interrupt priority registers, IRQP0, IRQP1, and IRQP2, allow each interrupt source to have its priority level configured for a level between 0 and 7. Level 0 is the highest priority level.

#### IRQP0 Register

Name: IRQP0  
 Address: 0xFFFF0020  
 Default value: 0x00000000  
 Access: Read and write

Table 69. IRQP0 MMR Bit Designations

| Bit   | Name     | Description  |
|-------|----------|--|
| 31:27 | Reserved | Reserved bits.   |
| 26:24 | T3PI     | A priority level of 0 to 7 can be set for Timer3.                        |
| 23    | Reserved | Reserved bit.  |
| 22:20 | T2PI     | A priority level of 0 to 7 can be set for Timer2.                        |
| 19    | Reserved | Reserved bit.  |
| 18:16 | T1PI     | A priority level of 0 to 7 can be set for Timer1.                        |
| 15    | Reserved | Reserved bit.  |
| 14:12 | T0PI     | A priority level of 0 to 7 can be set for Timer0.                        |
| 11:7  | Reserved | Reserved bits.   |
| 6:4   | SWINTP   | A priority level of 0 to 7 can be set for the software interrupt source. |
| 3:0   | Reserved | Interrupt 0 cannot be prioritized.                                       |

#### IRQP1 Register

Name: IRQP1  
 Address: 0xFFFF0024  
 Default value: 0x00000000  
 Access: Read and write

Table 70. IRQP1 MMR Bit Designations

| Bit   | Name     | Description   |
|-------|----------|---|
| 31    | Reserved | Reserved bit.   |
| 30:28 | I2CMPI   | A priority level of 0 to 7 can be set for I <sup>2</sup> C master.  |
| 27    | Reserved | Reserved bit.   |
| 26:24 | IRQ1PI   | A priority level of 0 to 7 can be set for IRQ1.                     |
| 23    | Reserved | Reserved bit.   |
| 22:20 | IRQ0PI   | A priority level of 0 to 7 can be set for IRQ0.                     |
| 19    | Reserved | Reserved bit.   |
| 18:16 | SPIMPI   | A priority level of 0 to 7 can be set for SPI master.               |
| 15    | Reserved | Reserved bit.   |
| 14:12 | UARTPI   | A priority level of 0 to 7 can be set for UART.                     |
| 11    | Reserved | Reserved bit.   |
| 10:8  | ADCPI    | A priority level of 0 to 7 can be set for the ADC interrupt source. |
| 7:0   | Reserved | Reserved bits.  |

#### IRQP2 Register

Name: IRQP2  
 Address: 0xFFFF0028  
 Default value: 0x00000000  
 Access: Read and write

Table 71. IRQP2 MMR Bit Designations

| Bit   | Name     | Description   |
|-------|----------|---|
| 31:15 | Reserved | Reserved bit.   |
| 14:12 | IRQ3PI   | A priority level of 0 to 7 can be set for IRQ3.                   |
| 11    | Reserved | Reserved bit.   |
| 10:8  | IRQ2PI   | A priority level of 0 to 7 can be set for IRQ2.                   |
| 7     | Reserved | Reserved bit.   |
| 6:4   | SPISPI   | A priority level of 0 to 7 can be set for SPI slave.              |
| 3     | Reserved | Reserved bit.   |
| 2:0   | I2CSPI   | A priority level of 0 to 7 can be set for I <sup>2</sup> C slave. |

**IRQCONN**

The IRQCONN register is the IRQ and FIQ control register. It contains two active bits: the first to enable nesting and prioritization of IRQ interrupts, and the other to enable nesting and prioritization of FIQ interrupts.

If these bits are cleared, FIQs and IRQs can still be used, but it is not possible to nest IRQs or FIQs. Neither is it possible to set an interrupt source priority level. In this default state, an FIQ does have a higher priority than an IRQ.

**IRQCONN Register**

Name: IRQCONN

Address: 0xFFFF0030

Default value: 0x00000000

Access: Read and write

**Table 72. IRQCONN MMR Bit Designations**

| Bit  | Name     | Description   |
|------|----------|---|
| 31:2 | Reserved | These bits are reserved and should not be written to.   |
| 1    | ENFIQN   | Setting this bit to 1 enables nesting of FIQ interrupts. Clearing this bit means no nesting or prioritization of FIQs is allowed. |
| 0    | ENIRQN   | Setting this bit to 1 enables nesting of IRQ interrupts. Clearing this bit means no nesting or prioritization of IRQs is allowed. |

**IRQSTAN**

If IRQCONN[0] is asserted and IRQVEC is read, then one of these bits is asserted. The bit that asserts depends on the priority of the IRQ. If the IRQ is of Priority 0, then Bit 0 asserts; Priority 1, then Bit 1 asserts; and so forth. When a bit is set in this register, all interrupts of that priority and lower are blocked.

To clear a bit in this register, all bits of a higher priority must be cleared first. It is possible to clear only one bit at a time. For example, if this register is set to 0x09, writing 0xFF changes the register to 0x08, and writing 0xFF a second time changes the register to 0x00.

**IRQSTAN Register**

Name: IRQSTAN

Address: 0xFFFF003C

Default value: 0x00000000

Access: Read and write

**Table 73. IRQSTAN MMR Bit Designations**

| Bit  | Name     | Description   |
|------|----------|---|
| 31:8 | Reserved | These bits are reserved and should not be written to.   |
| 7:0  |          | Setting this bit to 1 enables nesting of FIQ interrupts. Clearing this bit means no nesting or prioritization of FIQs is allowed. |

**FIQVEC**

The FIQ interrupt vector register, FIQVEC, points to a memory address containing a pointer to the interrupt service routine of the currently active FIQ. This register should be read only when an FIQ occurs and FIQ interrupt nesting has been enabled by setting Bit 1 of the IRQCONN register.

**FIQVEC Register**

Name: FIQVEC

Address: 0xFFFF011C

Default value: 0x00000000

Access: Read only

**Table 74. FIQVEC MMR Bit Designations**

| Bit   | Access    | Initial Value | Description   |
|-------|-----------|---------------|---|
| 31:23 | Read only | 0             | Always read as 0.   |
| 22:7  | Read only | 0             | IRQBASE register value.   |
| 6:2   |           | 0             | Highest priority FIQ source. This is a value between 0 to 19 that represents the possible interrupt sources. For example, if the highest currently active FIQ is Timer1, then these bits are [01000]. |
| 1:0   | Reserved  | 0             | Reserved bits.  |



**FIQSTAN**

If IRQCONN[1] is asserted and FIQVEC is read, then one of these bits asserts. The bit that asserts depends on the priority of the FIQ. If the FIQ is of Priority 0, Bit 0 asserts; Priority 1, Bit 1 asserts; and so forth.

When a bit is set in this register, all interrupts of that priority and lower are blocked.

To clear a bit in this register, all bits of a higher priority must be cleared first. It is possible to clear only one bit as a time. For example, if this register is set to 0x09, writing 0xFF changes the register to 0x08, and writing 0xFF a second time changes the register to 0x00.

**FIQSTAN Register**

Name: FIQSTAN  
 Address: 0xFFFF013C  
 Default value: 0x00000000  
 Access: Read and write

**External Interrupts (IRQ0 to IRQ3)**

The ADuC7060/ADuC7061 provides up to four external interrupt sources. These external interrupts can be individually configured as level triggered or rising/falling edge triggered.

To enable the external interrupt source, the appropriate bit must first be set in the FIQEN or IRQEN register. To select the required edge or level to trigger on, the IRQCONE register must be appropriately configured.

To properly clear an edge based external IRQ interrupt, set the appropriate bit in the IRQCLRE register.

**IRQCONE Register**

Name: IRQCONE  
 Address: 0xFFFF0034  
 Default value: 0x00000000  
 Access: Read and write

**Table 75. FIQSTAN MMR Bit Designations**

| Bit  | Name     | Description   |
|------|----------|---|
| 31:8 | Reserved | These bits are reserved and should not be written to.   |
| 7:0  |          | Setting this bit to 1 enables nesting of FIQ interrupts. Clearing this bit means no nesting or prioritization of FIQs is allowed. |

**Table 76. IRQCONE MMR Bit Designations**

| Bit  | Name         | Description  |
|------|--------------|--|
| 31:8 | Reserved     | These bits are reserved and should not be written to.  |
| 7:6  | IRQ3SRC[1:0] | [11] = External IRQ3 triggers on falling edge.<br>[10] = External IRQ3 triggers on rising edge.<br>[01] = External IRQ3 triggers on low level.<br>[00] = External IRQ3 triggers on high level. |
| 5:4  | IRQ2SRC[1:0] | [11] = External IRQ2 triggers on falling edge.<br>[10] = External IRQ2 triggers on rising edge.<br>[01] = External IRQ2 triggers on low level.<br>[00] = External IRQ2 triggers on high level. |
| 3:2  | IRQ1SRC[1:0] | [11] = External IRQ1 triggers on falling edge.<br>[10] = External IRQ1 triggers on rising edge.<br>[01] = External IRQ1 triggers on low level.<br>[00] = External IRQ1 triggers on high level. |
| 1:0  | IRQ0SRC[1:0] | [11] = External IRQ0 triggers on falling edge.<br>[10] = External IRQ0 triggers on rising edge.<br>[01] = External IRQ0 triggers on low level.<br>[00] = External IRQ0 triggers on high level. |

**IRQCLRE Register**

Name: IRQCLRE  
 Address: 0xFFFF0038  
 Default value: 0x00000000  
 Access: Read and write

**Table 77. IRQCLRE MMR Bit Designations**

| Bit   | Name     | Description  |
|-------|----------|--|
| 31:20 | Reserved | These bits are reserved and should not be written to.  |
| 19    | IRQ3CLRI | A 1 must be written to this bit in the IRQ3 interrupt service routine to clear an edge triggered IRQ3 interrupt. |
| 18    | IRQ2CLRI | A 1 must be written to this bit in the IRQ2 interrupt service routine to clear an edge triggered IRQ2 interrupt. |
| 17:15 | Reserved | These bits are reserved and should not be written to.  |
| 14    | IRQ1CLRI | A 1 must be written to this bit in the IRQ1 interrupt service routine to clear an edge triggered IRQ1 interrupt. |
| 13    | IRQ0CLRI | A 1 must be written to this bit in the IRQ0 interrupt service routine to clear an edge triggered IRQ0 interrupt. |
| 12:0  | Reserved | These bits are reserved and should not be written to.  |

## TIMERS

The ADuC7060/ADuC7061 features four general-purpose timer/counters.

- Timer0
- Timer1 or wake-up timer
- Timer2 or watchdog timer
- Timer3

The four timers in their normal mode of operation can be either free running or periodic.

In free running mode, the counter decrements/increments from the maximum or minimum value until zero/full scale and starts again at the maximum or minimum value.

In periodic mode, the counter decrements/increments from the value in the load register (TxLD MMR) until zero/full scale and starts again at the value stored in the load register. Note that the TxLD MMR should be configured before the TxCON MMR.

The timer interval is calculated as follows:

If the timer is set to count down then

$$\text{Interval} = (\text{TxLD} \times \text{Prescaler}) / \text{Source Clock}$$

If the timer is set to count up then

$$\text{Interval} = ((\text{Full Scale} - \text{TxLD}) \times \text{Prescaler}) / \text{Source Clock}$$

The value of a counter can be read at any time by accessing its value register (TxVAL). Timers are started by writing in the control register of the corresponding timer (TxCON).

In normal mode, an IRQ is generated each time that the value of the counter reaches zero (if counting down) or full scale (if counting up). An IRQ can be cleared by writing any value to the clear register of the particular timer (TxCLR).

**Table 78. Timer Event Capture**

| Bit | Description              |
|-----|--------------------------|
| 0   | Reserved                 |
| 1   | Timer0                   |
| 2   | Timer1 or wake-up timer  |
| 3   | Timer2 or watchdog timer |
| 4   | Timer3                   |
| 5   | Reserved                 |
| 6   | Reserved                 |
| 7   | Reserved                 |
| 8   | ADC                      |
| 9   | UART                     |
| 10  | SPI                      |
| 11  | XIRQ0                    |
| 12  | XIRQ1                    |
| 13  | I <sup>2</sup> C master  |
| 14  | I <sup>2</sup> C slave   |
| 15  | PWM                      |
| 16  | XIRQ2 (GPIO IRQ2)        |
| 17  | XIRQ3 (GPIO IRQ3)        |

### HR:MIN:SEC: 1/128 FORMAT

To use the timer in Hr : Min : Sec : hundreds format the 32768 kHz clock and prescaler of 256 should be selected. The hundreds field does not represent milliseconds but 1/128 of a second (256/32768). The bits representing the Hour, minute and second are not consecutive in the register. This arrangement applies to TxLd and TxVAL when using the Hr : Min : Sec : hundreds format as set in TxCON[5:4]. See Table 79 for more details.

**Table 79. Hr:Min:Sec: hundreds format**

| Bit   | Value               | Description     |
|-------|---------------------|-----------------|
| 31:24 | 0 to 23 or 0 to 255 | Hours           |
| 23:22 | 0                   | Reserved        |
| 21:16 | 0 to 59             | Minutes         |
| 15:14 | 0                   | Reserved        |
| 13:8  | 0 to 59             | Seconds         |
| 7     | 0                   | Reserved        |
| 6:0   | 0 to 127            | 1/128 of second |

**TIMER0**

Timer0 is a 32-bit, general-purpose timer, count down or count up, with a programmable prescaler. The prescaler source can be the low power 32.768 kHz oscillator, the core clock, or from one of two external GPIOs. This source can be scaled by a factor of 1, 16, 256, or 32,768. This gives a minimum resolution of 97.66 ns with a prescaler of 1 (ignoring the external GPIOs).

The counter can be formatted as a standard 32-bit value or as hours:minutes:seconds:hundredths.

Timer0 has a capture register (T0CAP) that is triggered by a selected IRQ source initial assertion. When triggered, the current timer value is copied to T0CAP, and the timer continues to run. Use this feature to determine the assertion of an event with increased accuracy. Note that only peripherals that have their IRQ source enabled can be used with the timer capture feature.

The Timer0 interface consists of five MMRS: T0LD, T0VAL, T0CAP, T0CLRI, and T0CON.

- T0LD, T0VAL, and T0CAP are 32-bit registers and hold 32-bit, unsigned integers of which T0VAL and T0CAP are read only.
- T0CLRI is an 8-bit register and writing any value to this register clears the Timer0 interrupt.
- T0CON is the configuration MMR, which is described in Table 80.

Timer0 features a postscaler that allows the user to count between 1 and 256 the number of Timer0 timeouts. To activate the postscaler, the user sets Bit 18 and writes the desired number to count into Bits[24:31] of T0CON. When that number of timeouts is reached, Timer0 can generate an interrupt if T0CON[18] is set.

Note that, if the part is in a low power mode and Timer0 is clocked from the GPIO or low power oscillator source, Timer0 continues to operate.

Timer0 reloads the value from T0LD when Timer0 overflows.

**Timer0 Load Registers**

|                |  |
|----------------|--|
| Name:          | T0LD   |
| Address:       | 0xFFFF0320   |
| Default value: | 0x00000000   |
| Access:        | Read and write   |
| Function:      | T0LD is a 32-bit register that holds the 32-bit value that is loaded into the counter. |

**Timer0 Clear Register**

|           |   |
|-----------|---|
| Name:     | T0CLRI  |
| Address:  | 0xFFFF032C  |
| Access:   | Write only  |
| Function: | This 8-bit, write-only MMR is written (with any value) by user code to clear the interrupt. |

**Timer0 Value Register**

|                |  |
|----------------|--|
| Name:          | T0VAL  |
| Address:       | 0xFFFF0324   |
| Default value: | 0xFFFFFFFF   |
| Access:        | Read only  |
| Function:      | T0VAL is a 32-bit register that holds the current value of Timer0. |

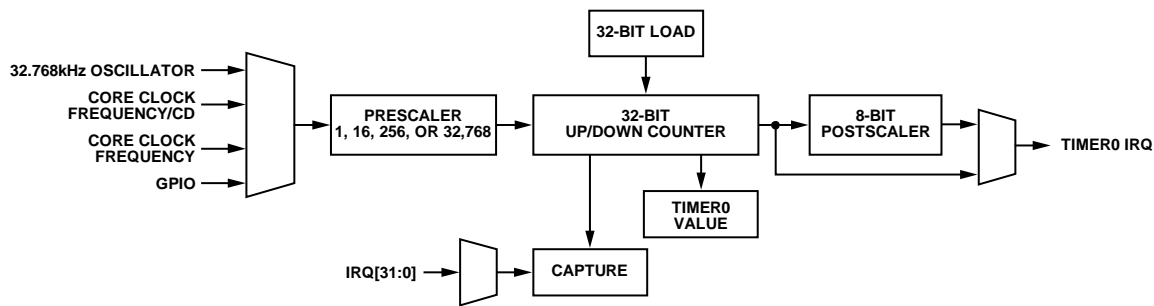


Figure 23. Timer0 Block Diagram

07079-017

**Timer0 Capture Register**

Name: T0CAP  
 Address: 0xFFFFF0330  
 Default value: 0x00000000  
 Access: Read only  
 Function: This 32-bit register holds the 32-bit value captured by an enabled IRQ event.

**Timer0 Control Register**

Name: T0CON  
 Address: 0xFFFFF0328  
 Default value: 0x01000000  
 Access: Read and write  
 Function: This 32-bit MMR configures the mode of operation of Timer0.

**Table 80. T0CON MMR Bit Designations**

| Bit   | Name     | Description   |
|-------|----------|---|
| 31:24 | T0PVAL   | 8-bit postscaler.<br>By writing to these eight bits, a value is written to the postscaler. Writing 0 is interpreted as a 1.<br>By reading these eight bits, the current value of the counter is read. |
| 23    | T0PEN    | Timer0 enable postscaler.<br>Set to enable the Timer0 postscaler. If enabled, interrupts are generated after T0CON[31:24] periods as defined by T0LD.<br>Cleared to disable the Timer0 postscaler.    |
| 22:20 |          | Reserved. These bits are reserved and should be written as 0 by user code.  |
| 19    | T0PCF    | Postscaler compare flag; read only. Set if the number of Timer0 overflows is equal to the number written to the postscaler.   |
| 18    | T0SRCI   | Timer0 interrupt source.<br>Set to select interrupt generation from the postscaler counter.<br>Cleared to select interrupt generation directly from Timer0.   |
| 17    | T0CAPEN  | Event enable bit.<br>Set by user to enable time capture of an event.<br>Cleared by user to disable time capture of an event.  |
| 16:12 | T0CAPSEL | Event Select Bits[17:0]. The events are described in Table 78.  |
| 11    |          | Reserved bit.   |
| 10:9  | T0CLKSEL | Clock select.<br>[00] = 32.768 kHz.<br>[01] = 10.24 MHz/CD.<br>[10] = 10.24 MHz.<br>[11] = P1.0.  |
| 8     | T0DIR    | Count up.<br>Set by user for Timer0 to count up.<br>Cleared by user for Timer0 to count down (default).   |
| 7     | T0EN     | Timer0 enable bit.<br>Set by user to enable Timer0.<br>Cleared by user to disable Timer0 (default).   |
| 6     | T0MOD    | Timer0 mode.<br>Set by user to operate in periodic mode.<br>Cleared by user to operate in free running mode (default).  |

| Bit | Name     | Description  |
|-----|----------|--|
| 5:4 | T0FORMAT | Format.<br>[00] = binary (default).<br>[01] = reserved.<br>[10] = hours:minutes:seconds:hundredths (23 hours to 0 hours).<br>[11] = hours:minutes:seconds:hundredths (255 hours to 0 hours). |
| 3:0 | T0SCALE  | Prescaler.<br>[0000] = source clock/1 (default).<br>[0100] = source clock/16.<br>[1000] = source clock/256.<br>[1111] = source clock/32,768. Note that all other values are undefined.       |

### TIMER1 OR WAKE-UP TIMER

Timer1 is a 32-bit wake-up timer, count down or count up, with a programmable prescaler. The prescaler is clocked directly from one of four clock sources, namely, the core clock (which is the default selection), external 32.768 kHz watch crystal, or the 32.768 kHz oscillator. The selected clock source can be scaled by a factor of 1, 16, 256, or 32,768. The wake-up timer continues to run when the core clock is disabled. This gives a minimum resolution of 97.66 ns when operating at CD zero, the core is operating at 10.24 MHz, and with a prescaler of 1 (ignoring the external GPIOs).

The counter can be formatted as a plain 32-bit value or as hours:minutes:seconds:hundredths.

Timer1 reloads the value from T1LD either when Timer1 overflows or immediately when T1LD is written.

The Timer1 interface consists of four MMRS.

- T1LD and T1VAL are 32-bit registers and hold 32-bit, unsigned integers. T1VAL is read only.
- T1CLRI is an 8-bit register. Writing any value to this register clears the Timer1 interrupt.
- T1CON is the configuration MMR, described in Table 81.

### Timer1 Load Registers

|                |  |
|----------------|--|
| Name:          | T1LD   |
| Address:       | 0xFFFF0340   |
| Default value: | 0x00000000   |
| Access:        | Read and write   |
| Function:      | T1LD is a 32-bit register that holds the 32-bit value that is loaded into the counter. |

### Timer1 Clear Register

|           |   |
|-----------|---|
| Name:     | T1CLRI  |
| Address:  | 0xFFFF034C  |
| Access:   | Write only  |
| Function: | This 8-bit, write-only MMR is written (with any value) by user code to clear the interrupt. |

### Timer1 Value Register

|                |  |
|----------------|--|
| Name:          | T1VAL  |
| Address:       | 0xFFFF0344   |
| Default value: | 0xFFFFFFFF   |
| Access:        | Read only  |
| Function:      | T1VAL is a 32-bit register that holds the current value of Timer1. |

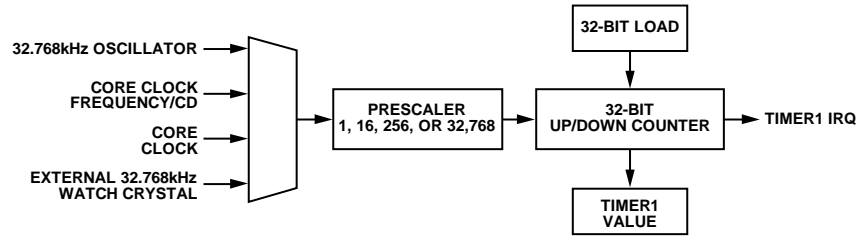


Figure 24. Timer1 Block Diagram

**Timer1 Control Register**

Name: TICON

Address: 0xFFFF0348

Default value: 0x0000

Access: Read and write

Function: This 16-bit MMR configures the mode of operation of Timer1.

**Table 81. TICON MMR Bit Designations**

| Bit   | Name     | Description   |
|-------|----------|---|
| 15:11 |          | Reserved.   |
| 10: 9 | T1CLKSEL | Clock source select.<br>[00] = 32.768 kHz oscillator.<br>[01] = 10.24 MHz/CD.<br>[10] = XTALI.<br>[11] = 10.24 MHz.   |
| 8     | T1DIR    | Count up.<br>Set by user for Timer1 to count up.<br>Cleared by user for Timer1 to count down (default).   |
| 7     | T1EN     | Timer1 enable bit.<br>Set by user to enable Timer1.<br>Cleared by user to disable Timer1 (default).   |
| 6     | T1MOD    | Timer1 mode.<br>Set by user to operate in periodic mode.<br>Cleared by user to operate in free running mode (default).  |
| 5:4   | T1FORMAT | Format.<br>[00] = binary (default).<br>[01] = reserved.<br>[10] = hours:minutes:seconds:hundredths (23 hours to 0 hours). This is only valid with a 32 kHz clock.<br>[11] = hours:minutes:seconds:hundredths (255 hours to 0 hours). This is only valid with a 32 kHz clock.  |
| 3:0   | T1SCALE  | Prescaler.<br>[0000] = source clock/1 (default).<br>[0100] = source clock/16.<br>[1000] = source clock/256. This setting should be used in conjunction with Timer1 in the format hours:minutes:seconds:hundredths. See Format 10 and Format 11 listed with Bits[5:4] in this table (Table 81).<br>[1111] = source clock/32,768. |

**TIMER2 OR WATCHDOG TIMER**

Timer2 has two modes of operation, normal mode and watchdog mode. The watchdog timer is used to recover from an illegal software state. When enabled, it requires periodic servicing to prevent it from forcing a reset of the processor.

Timer2 reloads the value from T2LD either when Timer2 overflows or immediately when T2CLRI is written.

**Normal Mode**

Timer2 in normal mode is identical to Timer0 in the 16-bit mode of operation, except for the clock source. The clock source is the low power, 32.768 kHz oscillator scalable by a factor of 1, 16, or 256.

**Watchdog Mode**

Watchdog mode is entered by setting T2CON[Bit 5]. Timer2 decrements from the timeout value present in the T2LD register until zero. The maximum timeout is 512 seconds, using a maximum prescaler/256 and full scale in T2LD.

User software should not configure a timeout period of less than 30 ms. This is to avoid any conflict with Flash/EE memory page erase cycles that require 20 ms to complete a single page erase cycle and kernel execution.

If T2VAL reaches 0, a reset or an interrupt occurs, depending on T2CON[1]. To avoid a reset or an interrupt event, any value must be written to T2CLRI before T2VAL reaches zero. This reloads the counter with T2LD and begins a new timeout period.

When watchdog mode is entered, T2LD and T2CON are write protected. These two registers cannot be modified until a power-on reset event resets the watchdog timer. After any other reset event, the watchdog timer continues to count. To avoid an infinite loop of watchdog resets, configure the watchdog timer in the initial lines of user code. User software should configure a minimum timeout period of 30 ms only.

Timer2 halts automatically during JTAG debug access and only recommences counting after JTAG relinquishes control of the ARM7 core. By default, Timer2 continues to count during power-down. To disable this, set Bit 0 in T2CON. It is recommended that the default value be used, that is, that the watchdog timer continues to count during power-down.

**Timer2 Interface**

The Timer2 interface consists of four MMRs.

- T2CON is the configuration MMR, described in (Table 82).
- T2LD and T2VAL are 16-bit registers (Bit 0 to Bit 15) and hold 16-bit, unsigned integers. T2VAL is read only.
- T2CLRI is an 8-bit register. Writing any value to this register clears the Timer2 interrupt in normal mode or resets a new timeout period in watchdog mode.

**Timer2 Load Register**

Name: T2LD  
 Address: 0xFFFF0360  
 Default value: 0x3BF8  
 Access: Read and write  
 Function: This 16-bit MMR holds the Timer2 reload value.

**Timer2 Clear Register**

Name: T2CLRI  
 Address: 0xFFFF036C  
 Access: Write only  
 Function: This 8-bit, write-only MMR is written (with any value) by user code to refresh (reload) Timer2 in watchdog mode to prevent a watchdog timer reset event.

**Timer2 Value Register**

Name: T2VAL  
 Address: 0xFFFF0364  
 Default value: 0x3BF8  
 Access: Read only  
 Function: This 16-bit, read-only MMR holds the current Timer2 count value.

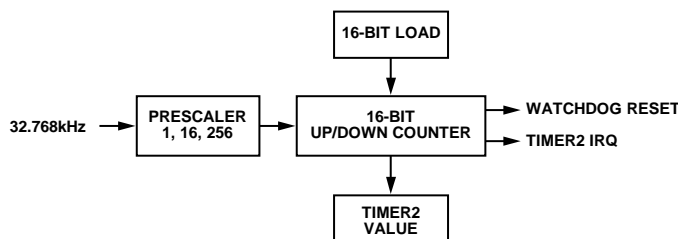


Figure 25. Timer2 Block Diagram

07075-018



**Timer2 Control Register**

Name: T2CON

Address: 0xFFFF0368

Default value: 0x0000

Access: Read and write

Function: This 16-bit MMR configures the mode of operation of Timer2, as described in detail in Table 82.

**Table 82. T2CON MMR Bit Designations**

| Bit  | Name     | Description   |
|------|----------|---|
| 15:9 |          | Reserved. These bits are reserved and should be written as 0 by user code.  |
| 8    | T2DIR    | Count up/count down enable.<br>Set by user code to configure Timer2 to count up.<br>Cleared by user code to configure Timer2 to count down.   |
| 7    | T2EN     | Timer2 enable.<br>Set by user code to enable Timer2.<br>Cleared by user code to disable Timer2.   |
| 6    | T2MOD    | Timer2 operating mode.<br>Set by user code to configure Timer2 to operate in periodic mode.<br>Cleared by user to configure Timer2 to operate in free running mode.   |
| 5    | WDOGMDEN | Watchdog timer mode enable.<br>Set by user code to enable watchdog mode.<br>Cleared by user code to disable watchdog mode.  |
| 4    |          | Reserved. This bit is reserved and should be written as 0 by user code.   |
| 3:2  | T2SCALE  | Timer2 clock (32.768 kHz) prescaler.<br>00 = 32.768 kHz (default).<br>01 = source clock/16.<br>10 = source clock/256.<br>11 = reserved.   |
| 1    | WDOGENI  | Watchdog timer IRQ enable.<br>Set by user code to produce an IRQ instead of a reset when the watchdog reaches 0.<br>Cleared by user code to disable the IRQ option.   |
| 0    | T2PDOFF  | Stop Timer2 when power-down is enabled.<br>Set by user code to stop Timer2 when the peripherals are powered down using Bit 4 in the POWCON0 MMR.<br>Cleared by user code to enable Timer2 when the peripherals are powered down using Bit 4 in the POWCON0 MMR. |

**TIMER3**

Timer3 is a general-purpose, 16-bit, count up/count down timer with a programmable prescaler. Timer3 can be clocked from the core clock or the low power 32.768 kHz oscillator with a prescaler of 1, 16, 256, or 32,768.

Timer3 has a capture register (T3CAP) that can be triggered by a selected IRQ source initial assertion. Once triggered, the current timer value is copied to T3CAP, and the timer continues to run. This feature can be used to determine the assertion of an event with increased accuracy.

The Timer3 interface consists of five MMRs.

- T3LD, T3VAL, and T3CAP are 16-bit registers and hold 16-bit, unsigned integers. T3VAL and T3CAP are read only.
- T3CLRI is an 8-bit register. Writing any value to this register clears the interrupt.
- T3CON is the configuration MMR, described in Table 83.

**Timer3 Load Registers**

|                |  |
|----------------|--|
| Name:          | T3LD   |
| Address:       | 0xFFFF0380   |
| Default value: | 0x0000   |
| Access:        | Read and write   |
| Function:      | T3LD is a 16-bit register that holds the 16-bit value that is loaded into the counter. |

**Timer3 Clear Register**

|           |   |
|-----------|---|
| Name:     | T3CLRI  |
| Address:  | 0xFFFF038C  |
| Access:   | Write only  |
| Function: | This 8-bit, write-only MMR is written (with any value) by user code to clear the interrupt. |

**Timer3 Value Register**

|                |  |
|----------------|--|
| Name:          | T3VAL  |
| Address:       | 0xFFFF0384   |
| Default value: | 0xFFFF   |
| Access:        | Read only  |
| Function:      | T3VAL is a 16-bit register that holds the current value of Timer3. |

**Time3 Capture Register**

|                |   |
|----------------|---|
| Name:          | T3CAP   |
| Address:       | 0xFFFF0390  |
| Default value: | 0x0000  |
| Access:        | Read only   |
| Function:      | This is a 16-bit register that holds the 16-bit value captured by an enabled IRQ event. |

**Timer3 Control Register**

|                |   |
|----------------|---|
| Name:          | T3CON   |
| Address:       | 0xFFFF0388  |
| Default value: | 0x00000000  |
| Access:        | Read and write  |
| Function:      | This 32-bit MMR configures the mode of operation of Timer3. |

Table 83. T3CON MMR Bit Designations

| Bit   | Name     | Description  |
|-------|----------|--|
| 31:18 |          | Reserved.  |
| 17    | T3CAPEN  | Event enable bit.<br>Set by user to enable time capture of an event.<br>Cleared by user to disable time capture of an event.                 |
| 16:12 | T3CAPSEL | Event select range, 0 to 17. The events are described in Table 78.   |
| 11    |          | Reserved.  |
| 10:9  | T3CLKSEL | Clock select.<br>[00] = 32.768 kHz oscillator.<br>[01] = 10.24 MHz/CD.<br>[10] = 10.24 MHz.<br>[11] = reserved.                              |
| 8     | T3DIR    | Count up.<br>Set by user for Timer3 to count up.<br>Cleared by user for Timer3 to count down (default).                                      |
| 7     | T3EN     | Timer3 enable bit.<br>Set by user to enable Timer3.<br>Cleared by user to disable Timer3 (default).  |
| 6     | T3MOD    | Timer3 mode.<br>Set by user to operate in periodic mode.<br>Cleared by user to operate in free running mode (default mode).                  |
| 5:4   |          | Reserved.  |
| 3:0   | T3SCALE  | Prescaler.<br>[0000] = source clock/1 (default).<br>[0100] = source clock/16.<br>[1000] = source clock/256.<br>[1111] = source clock/32,768. |

# PULSE-WIDTH MODULATOR

## PULSE-WIDTH MODULATOR GENERAL OVERVIEW

Each ADuC7060/ADuC7061 integrates a 6-channel pulse-width modulator (PWM) interface. The PWM outputs can be configured to drive an H-bridge or can be used as standard PWM outputs. On power-up, the PWM outputs default to H-bridge mode. This ensures that the motor is turned off by default. In standard PWM mode, the outputs are arranged as three pairs of PWM pins. Users have control over the period of each pair of outputs and over the duty cycle of each individual output.

Table 84. PWM MMRs

| MMR Name | Description   |
|----------|---|
| PWMCON   | PWM control.  |
| PWM0COM0 | Compare Register 0 for PWM Output 0 and PWM Output 1. |
| PWM0COM1 | Compare Register 1 for PWM Output 0 and PWM Output 1. |
| PWM0COM2 | Compare Register 2 for PWM Output 0 and PWM Output 1. |
| PWM0LEN  | Frequency control for PWM Output 0 and PWM Output 1.  |
| PWM1COM0 | Compare Register 0 for PWM Output 2 and PWM Output 3. |
| PWM1COM1 | Compare Register 1 for PWM Output 2 and PWM Output 3. |
| PWM1COM2 | Compare Register 2 for PWM Output 2 and PWM Output 3. |
| PWM1LEN  | Frequency control for PWM Output 2 and PWM Output 3.  |
| PWM2COM0 | Compare Register 0 for PWM Output 4 and PWM Output 5. |
| PWM2COM1 | Compare Register 1 for PWM Output 4 and PWM Output 5. |
| PWM2COM2 | Compare Register 2 for PWM Output 4 and PWM Output 5. |
| PWM2LEN  | Frequency control for PWM Output 4 and PWM Output 5.  |
| PWMCLRI  | PWM interrupt clear.                                  |

In all modes, the PWMxCOMx MMRs control the point at which the PWM outputs change state. An example of the first pair of PWM outputs (PWM0 and PWM1) is shown in Figure 26.

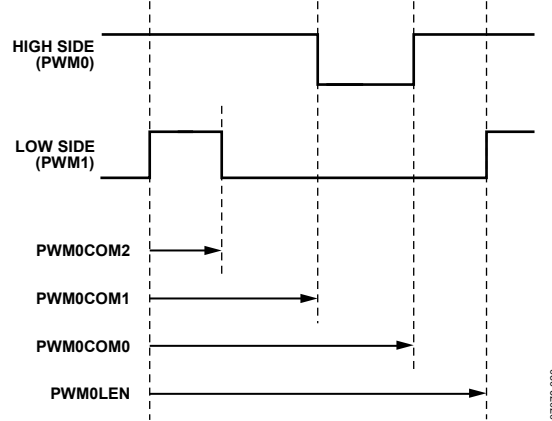


Figure 26. PWM Timing

The PWM clock is selectable via PWMCON with one of the following values: UCLK divided by 2, 4, 8, 16, 32, 64, 128, or 256. The length of a PWM period is defined by PWMxLEN.

The PWM waveforms are set by the count value of the 16-bit timer and the compare registers contents, as shown with the PWM0 and PWM1 waveforms in Figure 26.

The low-side waveform, PWM1, goes high when the timer count reaches PWM0LEN, and it goes low when the timer count reaches the value held in PWM0COM2 or when the high-side waveform (PWM0) goes low.

The high-side waveform, PWM0, goes high when the timer count reaches the value held in PWM0COM0, and it goes low when the timer count reaches the value held in PWM0COM1.

### PWMCON Control Register

Name: PWMCON

Address: 0xFFFF0F80

Default value: 0x0012

Access: Read and write

Function: This is a 16-bit MMR that configures the PWM outputs.

Table 85. PWMCON MMR Bit Designations

| Bit | Name       | Description   |
|-----|------------|---|
| 15  | Reserved   | This bit is reserved. Do not write to this bit.   |
| 14  | Sync       | Enables PWM synchronization.<br>Set to 1 by user so that all PWM counters are reset on the next clock edge after the detection of a high-to-low transition on the P1.2/SYNC pin.<br>Cleared by user to ignore transitions on the P1.2/SYNC pin.                     |
| 13  | PWM5INV    | Set to 1 by user to invert PWM5.<br>Cleared by user to use PWM5 in normal mode.   |
| 12  | PWM3INV    | Set to 1 by user to invert PWM3.<br>Cleared by user to use PWM3 in normal mode.   |
| 11  | PWM1INV    | Set to 1 by user to invert PWM1.<br>Cleared by user to use PWM1 in normal mode.   |
| 10  | PWMTRIP    | Set to 1 by user to enable PWM trip interrupt. When the PWM trip input (Pin P1.3/TRIP) is low, the PWMEN bit is cleared and an interrupt is generated.<br>Cleared by user to disable the PWMTRIP interrupt.   |
| 9   | ENA        | If HOFF = 0 and HMODE = 1. Note that, if not in H-bridge mode, this bit has no effect.<br>Set to 1 by user to enable PWM outputs.<br>Cleared by user to disable PWM outputs.<br>If HOFF = 1 and HMODE = 1, see Table 86.  |
| 8:6 | PWMCP[2:0] | PWM clock prescaler bits. Sets the UCLK divider.<br>[000] = UCLK/2.<br>[001] = UCLK/4.<br>[010] = UCLK/8.<br>[011] = UCLK/16.<br>[100] = UCLK/32.<br>[101] = UCLK/64.<br>[110] = UCLK/128.<br>[111] = UCLK/256.   |
| 5   | POINV      | Set to 1 by user to invert all PWM outputs.<br>Cleared by user to use PWM outputs as normal.  |
| 4   | HOFF       | High side off.<br>Set to 1 by user to force PWM0 and PWM2 outputs high. This also forces PWM1 and PWM3 low.<br>Cleared by user to use the PWM outputs as normal.  |
| 3   | LCOMP      | Load compare registers.<br>Set to 1 by user to load the internal compare registers with the values in PWMxCOMx on the next transition of the PWM timer from 0x00 to 0x01.<br>Cleared by user to use the values previously stored in the internal compare registers. |
| 2   | DIR        | Direction control.<br>Set to 1 by user to enable PWM0 and PWM1 as the output signals while PWM2 and PWM3 are held low.<br>Cleared by user to enable PWM2 and PWM3 as the output signals while PWM0 and PWM1 are held low.   |
| 1   | HMODE      | Enables H-bridge mode. <sup>1</sup><br>Set to 1 by user to enable H-bridge mode and Bit 1 to Bit 5 of PWMCON.<br>Cleared by user to operate the PWMs in standard mode.  |
| 0   | PWMEN      | Set to 1 by user to enable all PWM outputs.<br>Cleared by user to disable all PWM outputs.  |

<sup>1</sup> In H-bridge mode, HMODE = 1. See Table 86 to determine the PWM outputs.

On power-up, PWMCON defaults to 0x0012 (HOFF = 1 and HMODE = 1). All GPIO pins associated with the PWM are configured in PWM mode by default (see Table 86). Clear the PWM trip interrupt by writing any value to the PWMCLRI

MMR. Note that when using the PWM trip interrupt, clear the PWM interrupt before exiting the ISR. This prevents generation of multiple interrupts.

**Table 86. PWM Output Selection**

| PWMCON MMR <sup>1</sup> |      |       |     | PWM Outputs <sup>2</sup> |      |      |      |
|-------------------------|------|-------|-----|--------------------------|------|------|------|
| ENA                     | HOFF | POINV | DIR | PWM0                     | PWM1 | PWM2 | PWM3 |
| 0                       | 0    | X     | X   | 1                        | 1    | 1    | 1    |
| X                       | 1    | X     | X   | 1                        | 0    | 1    | 0    |
| 1                       | 0    | 0     | 0   | 0                        | 0    | HS1  | LS1  |
| 1                       | 0    | 0     | 1   | HS1                      | LS1  | 0    | 0    |
| 1                       | 0    | 1     | 0   | HS1                      | LS1  | 1    | 1    |
| 1                       | 0    | 1     | 1   | 1                        | 1    | HS1  | LS1  |

<sup>1</sup> X is don't care.

<sup>2</sup> HS = high side, LS = low side.

**Table 87. Compare Registers**

| Name     | Address    | Default Value | Access |
|----------|------------|---------------|--------|
| PWM0COM0 | 0xFFFF0F84 | 0x0000        | R/W    |
| PWM0COM1 | 0xFFFF0F88 | 0x0000        | R/W    |
| PWM0COM2 | 0xFFFF0F8C | 0x0000        | R/W    |
| PWM1COM0 | 0xFFFF0F94 | 0x0000        | R/W    |
| PWM1COM1 | 0xFFFF0F98 | 0x0000        | R/W    |
| PWM1COM2 | 0xFFFF0F9C | 0x0000        | R/W    |
| PWM2COM0 | 0xFFFF0FA4 | 0x0000        | R/W    |
| PWM2COM1 | 0xFFFF0FA8 | 0x0000        | R/W    |
| PWM2COM2 | 0xFFFF0FAC | 0x0000        | R/W    |

**PWM0COM0 Compare Register**

Name: PWM0COM0  
Address: 0xFFFF0F84  
Default value: 0x0000  
Access: Read and write  
Function: PWM0 output pin goes high when the PWM timer reaches the count value stored in this register.

**PWM0COM1 Compare Register**

Name: PWM0COM1  
Address: 0xFFFF0F88  
Default value: 0x0000  
Access: Read and write  
Function: PWM0 output pin goes low when the PWM timer reaches the count value stored in this register.

**PWM0COM2 Compare Register**

Name: PWM0COM2  
Address: 0xFFFF0F8C  
Default value: 0x0000  
Access: Read and write  
Function: PWM1 output pin goes low when the PWM timer reaches the count value stored in this register.

**PWM0LEN Register**

Name: PWM0LEN  
Address: 0xFFFF0F90  
Default value: 0x0000  
Access: Read and write  
Function: PWM1 output pin goes high when the PWM timer reaches the value stored in this register.

**PWM1COM0 Compare Register**

Name: PWM1COM0  
Address: 0xFFFF0F94  
Default value: 0x0000  
Access: Read and write  
Function: PWM2 output pin goes high when the PWM timer reaches the count value stored in this register.

**PWM1COM1 Compare Register**

Name: PWM1COM1  
Address: 0xFFFF0F98  
Default value: 0x0000  
Access: Read and write  
Function: PWM2 output pin goes low when the PWM timer reaches the count value stored in this register.

**PWM1COM2 Compare Register**

Name: PWM1COM2  
Address: 0xFFFF0F9C  
Default value: 0x0000  
Access: Read and write  
Function: PWM3 output pin goes low when the PWM timer reaches the count value stored in this register.

**PWM1LEN Register**

Name: PWM1LEN  
Address: 0xFFFF0FA0  
Default value: 0x0000  
Access: Read and write  
Function: PWM3 output pin goes high when the PWM timer reaches the value stored in this register.

**PWM2COM0 Compare Register**

Name: PWM2COM0  
Address: 0xFFFF0FA4  
Default value: 0x0000  
Access: Read and write  
Function: PWM4 output pin goes high when the PWM timer reaches the count value stored in this register.

**PWM2COM1 Compare Register**

Name: PWM2COM1  
Address: 0xFFFF0FA8  
Default value: 0x0000  
Access: Read and write  
Function: PWM4 output pin goes low when the PWM timer reaches the count value stored in this register.

**PWM2COM2 Compare Register**

Name: PWM2COM2  
Address: 0xFFFF0FAC  
Default value: 0x0000  
Access: Read and write  
Function: PWM5 output pin goes low when the PWM timer reaches the count value stored in this register.

**PWM2LEN Register**

Name: PWM2LEN  
Address: 0xFFFF0FB0  
Default value: 0x0000  
Access: Read and write  
Function: PWM5 output pin goes high when the PWM timer reaches the value stored in this register.

**PWMCLRI Register**

Name: PWMCLRI  
Address: 0xFFFF0FB8  
Default value: 0x0000  
Access: Write only  
Function: Write any value to this register to clear a PWM interrupt source. This register must be written to before exiting a PWM interrupt service routine; otherwise, multiple interrupts occur.



## UART SERIAL INTERFACE

Each ADuC7060/ADuC7061 features a 16450-compatible UART. The UART is a full-duplex, universal, asynchronous receiver/transmitter. A UART performs serial-to-parallel conversion on data characters received from a peripheral device and parallel-to-serial conversion on data characters received from the ARM7TDMI. The UART features a fractional divider that facilitates high accuracy baud rate generation and a network addressable mode. The UART functionality is available on the P1.0/IRQ1/SIN/T0 and P1.1/SOUT pins of the ADuC7060/ADuC7061.

The serial communication adopts an asynchronous protocol that supports various word lengths, stop bits, and parity generation options selectable in the configuration register.

### BAUD RATE GENERATION

The ADuC7060/ADuC7061 features two methods of generating the UART baud rate: normal 450 UART baud rate generation and ADuC7060/ADuC7061 fractional divider.

#### Normal 450 UART Baud Rate Generation

The baud rate is a divided version of the core clock using the value in COMDIV0 and COMDIV1 MMRs (16-bit value, divisor latch (DL)). The standard baud rate generator formula is

$$\text{Baud Rate} = \frac{10.24 \text{ MHz}}{16 \times 2 \times DL} \quad (1)$$

Table 88 lists common baud rate values.

**Table 88. Baud Rate Using the Standard Baud Rate Generator**

| Baud Rate | DL   | Actual Baud Rate | % Error |
|-----------|------|------------------|---------|
| 9600      | 0x21 | 9696             | 1.01%   |
| 19,200    | 0x11 | 18,824           | 1.96%   |
| 115,200   | 0x3  | 106,667          | 7.41%   |

#### ADuC7060/ADuC7061 Fractional Divider

The fractional divider combined with the normal baud rate generator allows the generation of accurate high speed baud rates.

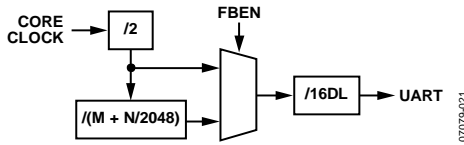


Figure 27. Fractional Divider Baud Rate Generation

Calculation of the baud rate using a fractional divider is as follows:

$$\text{Baud Rate} = \frac{10.24 \text{ MHz}}{16 \times DL \times 2 \times \left(M + \frac{N}{2048}\right)} \quad (2)$$

$$M + \frac{N}{2048} = \frac{10.24 \text{ MHz}}{\text{Baud Rate} \times 16 \times DL \times 2}$$

Table 89 lists common baud rate values.

**Table 89. Baud Rate Using the Fractional Baud Rate Generator**

| Baud Rate | DL   | M | N   | Actual Baud Rate | % Error |
|-----------|------|---|-----|------------------|---------|
| 9600      | 0x21 | 1 | 21  | 9598.55          | 0.015%  |
| 19,200    | 0x10 | 1 | 85  | 19,203           | 0.015%  |
| 115,200   | 0x2  | 1 | 796 | 115,218          | 0.015%  |

### UART REGISTER DEFINITIONS

The UART interface consists of the following 11 registers:

- COMTX: 8-bit transmit register
- COMRX: 8-bit receive register
- COMDIV0: divisor latch (low byte)
- COMDIV1: divisor latch (high byte)
- COMCON0: line control register
- COMCON1: line control register
- COMSTA0: line status register
- COMSTA1: line status register
- COMIEN0: interrupt enable register
- COMIID0: interrupt identification register
- COMDIV2: 16-bit fractional baud divide register

COMTX, COMRX, and COMDIV0 share the same address location. COMTX and COMRX can be accessed when Bit 7 in the COMCON0 register is cleared. COMDIV0 or COMDIV1 can be accessed when Bit 7 of COMCON0 or COMCON1, respectively, is set.

**UART Transmit Register**

Write to this 8-bit register (COMTX) to transmit data using the UART.

**COMTX Register**

Name: COMTX  
Address: 0xFFFF0700  
Access: Write only

**UART Receive Register**

This 8-bit register (COMRX) is read to receive data transmitted using the UART.

**COMRX Register**

Name: COMRX  
Address: 0xFFFF0700  
Default value: 0x00  
Access: Read only

**UART Divisor Latch Register 0**

This 8-bit register (COMDIV0) contains the least significant byte of the divisor latch that controls the baud rate at which the UART operates.

**COMDIV0 Register**

Name: COMDIV0  
Address: 0xFFFF0700  
Default value: 0x00  
Access: Read and write

**UART Divisor Latch Register 1**

This 8-bit register contains the most significant byte of the divisor latch that controls the baud rate at which the UART operates.

**COMDIV1 Register**

Name: COMDIV1  
Address: 0xFFFF0704  
Default value: 0x00  
Access: Read and write

**UART Control Register 0**

This 8-bit register (COMCON0) controls the operation of the UART in conjunction with COMCON1.

**COMCON0 Register**

Name: COMCON0  
Address: 0xFFFF070C  
Default value: 0x00  
Access: Read and write

Table 90. COMCON0 MMR Bit Designations

| Bit | Name | Description  |
|-----|------|--|
| 7   | DLAB | Divisor latch access.<br>Set by user to enable access to the COMDIV0 and COMDIV1 registers.<br>Cleared by user to disable access to COMDIV0 and COMDIV1 and enable access to COMRX, COMTX, and COMIEN0.  |
| 6   | BRK  | Set break.<br>Set by user to force transmit to 0.<br>Cleared to operate in normal mode.  |
| 5   | SP   | Stick parity. Set by user to force parity to defined values.<br>1 if EPS = 1 and PEN = 1.<br>0 if EPS = 0 and PEN = 1.   |
| 4   | EPS  | Even parity select bit.<br>Set for even parity.<br>Cleared for odd parity.   |
| 3   | PEN  | Parity enable bit.<br>Set by user to transmit and check the parity bit.<br>Cleared by user for no parity transmission or checking.   |
| 2   | Stop | Stop bit.<br>Set by user to transmit 1.5 stop bits if the word length is 5 bits, or 2 stop bits if the word length is 6 bits, 7 bits, or 8 bits. The receiver checks the first stop bit only, regardless of the number of stop bits selected.<br>Cleared by user to generate one stop bit in the transmitted data. |
| 1:0 | WLS  | Word length select.<br>[00] = 5 bits.<br>[01] = 6 bits.<br>[10] = 7 bits.<br>[11] = 8 bits.  |

**UART Control Register 1**

This 8-bit register controls the operation of the UART in conjunction with COMCON0.

**COMCON1 Register**

Name: COMCON1

Address: 0xFFFF0710

Default value: 0x00

Access: Read and write

**Table 91. COMCON1 MMR Bit Designations**

| Bit | Name     | Description  |
|-----|----------|--|
| 7:5 |          | Reserved bits. Not used.   |
| 4   | LOOPBACK | Loopback. Set by user to enable loopback mode. In loopback mode, the transmit pin is forced high.                  |
| 3:2 |          | Reserved bits. Not used.   |
| 1   | RTS      | Request to send.<br>Set by user to force the RTS output to 0.<br>Cleared by user to force the RTS output to 1.     |
| 0   | DTR      | Data terminal ready.<br>Set by user to force the DTR output to 0.<br>Cleared by user to force the DTR output to 1. |

**UART Status Register 0****COMSTA0 Register**

Name: COMSTA0

Address: 0xFFFF0714

Default value: 0x60

Access: Read only

Function: This 8-bit read-only register reflects the current status on the UART.

**Table 92. COMSTA0 MMR Bit Designations**

| Bit | Name | Description   |
|-----|------|---|
| 7   |      | Reserved.   |
| 6   | TEMT | COMTX and shift register empty status bit.<br>Set automatically if COMTX and the shift register are empty. This bit indicates that the data has been transmitted, that is, no more data is present in the shift register.<br>Cleared automatically when writing to COMTX. |
| 5   | THRE | COMTX empty status bit.<br>Set automatically if COMTX is empty.<br>COMTX can be written as soon as this bit is set; the previous data might not have been transmitted yet and can still be present in the shift register.<br>Cleared automatically when writing to COMTX. |
| 4   | BI   | Break indicator.<br>Set when P1.0/IRQ1/SIN/T0 pin is held low for more than the maximum word length.<br>Cleared automatically.  |
| 3   | FE   | Framing error.<br>Set when the stop bit is invalid.<br>Cleared automatically.   |
| 2   | PE   | Parity error.<br>Set when a parity error occurs.<br>Cleared automatically.  |
| 1   | OE   | Overrun error.<br>Set automatically if data is overwritten before being read.<br>Cleared automatically.   |
| 0   | DR   | Data ready.<br>Set automatically when COMRX is full.<br>Cleared by reading COMRX.   |

**UART Status Register 1****COMSTA1 Register**

Name: COMSTA1

Address: 0xFFFF0718

Default value: 0x00

Access: Read only

Function: COMSTA1 is a modem status register.

**Table 93. COMSTA1 MMR Bit Designations**

| Bit | Name | Description  |
|-----|------|--|
| 7:5 |      | Reserved. Not used.  |
| 4   | CTS  | Clear to send.   |
| 3:1 |      | Reserved. Not used.  |
| 0   | DCTS | Delta CTS.<br>Set automatically if CTS changed state since COMSTA1 was last read.<br>Cleared automatically by reading COMSTA1. |

**UART Interrupt Enable Register 0****COMIEN0 Register**

Name: COMIEN0

Address: 0xFFFF0704

Default value: 0x00

Access: Read and write

Function: This 8-bit register enables and disables the individual UART interrupt sources.

**Table 94. COMIEN0 MMR Bit Designations**

| Bit | Name  | Description  |
|-----|-------|--|
| 7:4 |       | Reserved. Not used.  |
| 3   | EDSSI | Modem status interrupt enable bit.<br>Set by user to enable generation of an interrupt if COMSTA1[4] or COMSTA1[0] are set.<br>Cleared by user.                                    |
| 2   | ELSI  | Receive status interrupt enable bit.<br>Set by user to enable generation of an interrupt if any of the COMSTA0[3:1] register bits are set.<br>Cleared by user.                     |
| 1   | ETBEI | Enable transmit buffer empty interrupt.<br>Set by user to enable an interrupt when the buffer is empty during a transmission; that is, when COMSTA0[5] is set.<br>Cleared by user. |
| 0   | ERBFI | Enable receive buffer full interrupt.<br>Set by user to enable an interrupt when the buffer is full during a reception.<br>Cleared by user.  |

**UART Interrupt Identification Register 0****COMIID0 Register**

Name: COMIID0

Address: 0xFFFF0708

Default value: 0x01

Access: Read only

Function: This 8-bit register reflects the source of the UART interrupt.

Table 95. COMIID0 MMR Bit Designations

| Status Bits[2:1] | Bit 0 | Priority | Definition                      | Clearing Operation                  |
|------------------|-------|----------|---------------------------------|-------------------------------------|
| 00               | 1     |          | No interrupt                    |                                     |
| 11               | 0     | 1        | Receive line status interrupt   | Read COMSTA0                        |
| 10               | 0     | 2        | Receive buffer full interrupt   | Read COMRX                          |
| 01               | 0     | 3        | Transmit buffer empty interrupt | Write data to COMTX or read COMIID0 |
| 00               | 0     | 4        | Modem status interrupt          | Read COMSTA1 register               |

Table 96. COMDIV2 MMR Bit Designations

| Bit   | Name      | Description   |
|-------|-----------|---|
| 15    | FBEN      | Fractional baud rate generator enable bit. Set by user to enable the fractional baud rate generator. Cleared by user to generate the baud rate using the standard 450 UART baud rate generator. |
| 14:13 |           | Reserved.   |
| 12:11 | FBM[1:0]  | M. If FBM = 0, M = 4. See Equation 2 for the calculation of the baud rate using a fractional divider and Table 88 for common baud rate values.  |
| 10:0  | FBN[10:0] | N. See Equation 2 for the calculation of the baud rate using a fractional divider and Table 88 for common baud rate values.   |

### UART Fractional Divider Register

This 16-bit register (COMDIV2) controls the operation of the fractional divider for the [ADuC7060/ADuC7061](#).

#### COMDIV2 Register

Name: COMDIV2  
 Address: 0xFFFF072C  
 Default value: 0x0000  
 Access: Read and write

## I<sup>2</sup>C

Each ADuC7060/ADuC7061 incorporates an I<sup>2</sup>C peripheral that can be configured as a fully I<sup>2</sup>C-compatible I<sup>2</sup>C bus master device or as a fully I<sup>2</sup>C bus-compatible slave device. The two pins used for data transfer, SDA and SCL, are configured in a wire-ANDed format that allows arbitration in a multimaster system. These pins require external pull-up resistors. Typical pull-up resistor values are between 4.7 k $\Omega$  and 10 k $\Omega$ .

Users program the I<sup>2</sup>C bus peripheral (addressed in the I<sup>2</sup>C bus system). This ID can be modified any time that a transfer is not in progress. The user can configure the interface to respond to four slave addresses.

The transfer sequence of an I<sup>2</sup>C system consists of a master device initiating a transfer by generating a start condition while the bus is idle. The master transmits the slave device address and the direction of the data transfer (read or write) during the initial address transfer. If the master does not lose arbitration and the slave acknowledges, the data transfer is initiated. This continues until the master issues a stop condition and the bus becomes idle.

The I<sup>2</sup>C peripheral can be configured only as a master or a slave at any given time. The same I<sup>2</sup>C channel cannot simultaneously support master and slave modes.

The I<sup>2</sup>C interface on the ADuC7060/ADuC7061 includes the following features:

- Support for repeated start conditions. In master mode, the ADuC7060/ADuC7061 can be programmed to generate a repeated start. In slave mode, the ADuC7060/ADuC7061 recognizes repeated start conditions.
- In master and slave modes, the part recognizes both 7-bit and 10-bit bus addresses.

- In I<sup>2</sup>C master mode, the ADuC7060/ADuC7061 supports continuous reads from a single slave up to 512 bytes in a single transfer sequence.
- Clock stretching can be enabled by other devices on the bus without causing any issues with the ADuC7060/ADuC7061. However, the ADuC7060/ADuC7061 cannot enable clock stretching.
- In slave mode, the ADuC7060/ADuC7061 can be programmed to return a no acknowledge (NACK). This allows the validation of checksum bytes at the end of I<sup>2</sup>C transfers.
- Bus arbitration in master mode is supported.
- Internal and external loopback modes are supported for I<sup>2</sup>C hardware testing.
- The transmit and receive circuits in both master and slave modes contain 2-byte FIFOs. Status bits are available to the user to control these FIFOs.

### CONFIGURING EXTERNAL PINS FOR I<sup>2</sup>C FUNCTIONALITY

The I<sup>2</sup>C functions of the P0.1/SCLK/SCL and P0.3/MOSI/SDA pins of the ADuC7060/ADuC7061 device are P0.1 and P0.3. The function of P0.1 is the I<sup>2</sup>C clock signal (SCL) and the function of P0.3 is the I<sup>2</sup>C data signal (SDA). To configure P0.1 and P0.3 for I<sup>2</sup>C mode, Bit 4 and Bit 12 of the GP0CON0 register must be set to 1. Bit 1 of the GP0CON1 register must also be set to 1 to enable I<sup>2</sup>C mode.

Note that, to write to GP0CON1, the GP0KEY1 register must be set to 0x7 immediately before writing to GP0CON1. Also, the GP0KEY2 register must be set to 0x13 immediately after writing to GP0CON1. The following code example shows this in detail:

```
GP0CON0 = BIT4 + BIT12;    // Select SPI/I2C alternative function for P0.1 and P0.3
GP0KEY1 = 0x7;            // Write to GP0KEY1
GP0CON1 = BIT1;          // Select I2C functionality for P0.1 and P0.3
GP0KEY2 = 0x13;          // Write to GP0KEY2
```

## SERIAL CLOCK GENERATION

The I<sup>2</sup>C master in the system generates the serial clock for a transfer. The master channel can be configured to operate in fast mode (400 kHz) or standard mode (100 kHz).

The bit rate is defined in the I2CDIV MMR as follows:

$$f_{SERIAL\ CLOCK} = \frac{f_{UCLK}}{(2 + DIVH) + (2 + DIVL)}$$

where:

$f_{UCLK}$  is the clock before the clock divider.

$DIVH$  is the high period of the clock.

$DIVL$  is the low period of the clock.

Thus, for 100 kHz operation

$$DIVH = DIVL = 0x33$$

and for 400 kHz

$$DIVH = 0x0A, DIVL = 0x0F$$

The I2CDIV register corresponds to DIVH:DIVL.

## I<sup>2</sup>C BUS ADDRESSES

### Slave Mode

In slave mode, the I2CID0, I2CID1, I2CID2, and I2CID3 registers contain the device IDs. The device compares the four I2CIDx registers to the address byte received from the bus master. To be correctly addressed, the 7 MSBs of any ID register must be identical to the 7 MSBs of the first received address byte. The least significant bit of the ID registers (the transfer direction bit) is ignored in the process of address recognition.

The ADuC7060/ADuC7061 also supports 10-bit addressing mode. When Bit 1 of I2CSCON (ADR10EN bit) is set to 1, then one 10-bit address is supported in slave mode and is stored in the I2CID0 and I2CID1 registers. The 10-bit address is derived as follows:

I2CID0[0] is the read/write bit and is not part of the I<sup>2</sup>C address.

I2CID0[7:1] = Address Bits[6:0].

I2CID1[2:0] = Address Bits[9:7].

I2CID1[7:3] must be set to 11110b.

### Master Mode

In master mode, the I2CADR0 register is programmed with the I<sup>2</sup>C address of the device.

In 7-bit address mode, I2CADR0[7:1] are set to the device address. I2CADR0[0] is the read/write bit.

In 10-bit address mode, the 10-bit address is created as follows:

I2CADR0[7:3] must be set to 11110b.

I2CADR0[2:1] = Address Bits[9:8].

I2CADR1[7:0] = Address Bits[7:0].

I2CADR0[0] is the read/write bit.

## I<sup>2</sup>C REGISTERS

The I<sup>2</sup>C peripheral interface consists overall of 19 MMRs. Nine of these are master related only, nine are slave related only, and one MMR is common to both master and slave modes.

### I<sup>2</sup>C Master Registers

#### I<sup>2</sup>C Master Control, I2CMCON Register

Name: I2CMCON

Address: 0xFFFF0900

Default value: 0x0000

Access: Read and write

Function: This 16-bit MMR configures the I<sup>2</sup>C peripheral in master mode.



Table 97. I2CMCON MMR Bit Designations

| Bit  | Name       | Description  |
|------|------------|--|
| 15:9 |            | Reserved. These bits are reserved and should not be written to.  |
| 8    | I2CMCENI   | I <sup>2</sup> C transmission complete interrupt enable bit.<br>Set this bit to enable an interrupt on detecting a stop condition on the I <sup>2</sup> C bus.<br>Clear this interrupt source.   |
| 7    | I2CNACKENI | I <sup>2</sup> C no acknowledge (NACK) received interrupt enable bit.<br>Set this bit to enable interrupts when the I <sup>2</sup> C master receives a no acknowledge.<br>Clear this interrupt source.   |
| 6    | I2CALENI   | I <sup>2</sup> C arbitration lost interrupt enable bit.<br>Set this bit to enable interrupts when the I <sup>2</sup> C master did not gain control of the I <sup>2</sup> C bus.<br>Clear this interrupt source.  |
| 5    | I2CMTENI   | I <sup>2</sup> C transmit interrupt enable bit.<br>Set this bit to enable interrupts when the I <sup>2</sup> C master has transmitted a byte.<br>Clear this interrupt source.  |
| 4    | I2CMRENI   | I <sup>2</sup> C receive interrupt enable bit.<br>Set this bit to enable interrupts when the I <sup>2</sup> C master receives data.<br>Cleared by user to disable interrupts when the I <sup>2</sup> C master is receiving data.                             |
| 3    |            | Reserved. A value of 0 should be written to this bit.  |
| 2    | I2CILEN    | I <sup>2</sup> C internal loopback enable.<br>Set this bit to enable loopback test mode. In this mode, the SCL and SDA signals are connected internally to their respective input signals.<br>Cleared by user to disable loopback mode.                      |
| 1    | I2CBD      | I <sup>2</sup> C master backoff disable bit.<br>Set this bit to allow the device to compete for control of the bus even if another device is currently driving a start condition.<br>Clear this bit to back off until the I <sup>2</sup> C bus becomes free. |
| 0    | I2CMEN     | I <sup>2</sup> C master enable bit.<br>Set by user to enable the I <sup>2</sup> C master mode.<br>Cleared to disable the I <sup>2</sup> C master mode.   |

**I<sup>2</sup>C Master Status, I2CMSTA, Register**

Name: I2CMSTA

Address: 0xFFFF0904

Default value: 0x0000

Access: Read only

Function: This 16-bit MMR is the I<sup>2</sup>C status register in master mode.**Table 98. I2CMSTA MMR Bit Designations**

| Bit   | Name      | Description   |
|-------|-----------|---|
| 15:11 |           | Reserved. These bits are reserved.  |
| 10    | I2CBBUSY  | I <sup>2</sup> C bus busy status bit.<br>This bit is set to 1 when a start condition is detected on the I <sup>2</sup> C bus.<br>This bit is cleared when a stop condition is detected on the bus.  |
| 9     | I2CMRxFO  | Master receive FIFO overflow.<br>This bit is set to 1 when a byte is written to the receive FIFO when it is already full.<br>This bit is cleared in all other conditions.   |
| 8     | I2CMTC    | I <sup>2</sup> C transmission complete status bit.<br>This bit is set to 1 when a transmission is complete between the master and the slave with which it was communicating. If the I2CMCENI bit in I2CMCON is set, an interrupt is generated when this bit is set.<br>Clear this interrupt source.                   |
| 7     | I2CMND    | I <sup>2</sup> C master no acknowledge data bit<br>This bit is set to 1 when a no acknowledge condition is received by the master in response to a data write transfer. If the I2CNACKENI bit in I2CMCON is set, an interrupt is generated when this bit is set.<br>This bit is cleared in all other conditions.      |
| 6     | I2CMBUSY  | I <sup>2</sup> C master busy status bit.<br>Set to 1 when the master is busy processing a transaction.<br>Cleared if the master is ready or if another master device has control of the bus.  |
| 5     | I2CAL     | I <sup>2</sup> C arbitration lost status bit.<br>This bit is set to 1 when the I <sup>2</sup> C master does not gain control of the I <sup>2</sup> C bus. If the I2CALENI bit in I2CMCON is set, an interrupt is generated when this bit is set.<br>This bit is cleared in all other conditions.                      |
| 4     | I2CMNA    | I <sup>2</sup> C master no acknowledge address bit.<br>This bit is set to 1 when a no acknowledge condition is received by the master in response to an address. If the I2CNACKENI bit in I2CMCON is set, an interrupt is generated when this bit is set.<br>This bit is cleared in all other conditions.             |
| 3     | I2CMRXQ   | I <sup>2</sup> C master receive request bit.<br>This bit is set to 1 when data enters the receive FIFO. If the I2CMRENI in I2CMCON is set, an interrupt is generated.<br>This bit is cleared in all other conditions.   |
| 2     | I2CMTXQ   | I <sup>2</sup> C master transmit request bit.<br>This bit goes high if the transmit FIFO is empty or contains only one byte and the master has transmitted an address + write. If the I2CMTENI bit in I2CMCON is set, an interrupt is generated when this bit is set.<br>This bit is cleared in all other conditions. |
| 1:0   | I2CMTFSTA | I <sup>2</sup> C master transmit FIFO status bits.<br>[00] = I <sup>2</sup> C master transmit FIFO empty.<br>[01] = 1 byte in master transmit FIFO.<br>[10] = 1 byte in master transmit FIFO.<br>[11] = I <sup>2</sup> C master transmit FIFO full.   |

**I<sup>2</sup>C Master Receive, I2CMRX, Register**

Name: I2CMRX  
 Address: 0xFFFF0908  
 Default value: 0x00  
 Access: Read only  
 Function: This 8-bit MMR is the I<sup>2</sup>C master receive register.

**I<sup>2</sup>C Master Transmit, I2CMTX, Register**

Name: I2CMTX  
 Address: 0xFFFF090C  
 Default value: 0x00  
 Access: Write only  
 Function: This 8-bit MMR is the I<sup>2</sup>C master transmit register.

**I<sup>2</sup>C Master Read Count, I2CMCNT0, Register**

Name: I2CMCNT0  
 Address: 0xFFFF0910  
 Default value: 0x0000  
 Access: Read and write  
 Function: This 16-bit MMR holds the required number of bytes when the master begins a read sequence from a slave device.

**Table 99. I2CMCNT0 MMR Bit Designations**

| Bit  | Name     | Description   |
|------|----------|---|
| 15:9 |          | Reserved.   |
| 8    | I2CRECNT | Set this bit if more than 256 bytes are required from the slave.<br>Clear this bit when reading 256 bytes or fewer.                               |
| 7:0  | I2CRCNT  | These eight bits hold the number of bytes required during a slave read sequence, minus 1. If only a single byte is required, set these bits to 0. |

**I<sup>2</sup>C Master Current Read Count, I2CMCNT1, Register**

Name: I2CMCNT1  
 Address: 0xFFFF0914  
 Default value: 0x00  
 Access: Read only  
 Function: This 8-bit MMR holds the number of bytes received so far during a read sequence with a slave device.

**I<sup>2</sup>C Address 0, I2CADR0, Register**

Name: I2CADR0  
 Address: 0xFFFF0918  
 Default value: 0x00  
 Access: Read and write  
 Function: This 8-bit MMR holds the 7-bit slave address and the read/write bit when the master begins communicating with a slave.

**Table 100. I2CADR0 MMR in 7-Bit Address Mode**

| Bit | Name   | Description   |
|-----|--------|---|
| 7:1 | I2CADR | These bits contain the 7-bit address of the required slave device.  |
| 0   | R/W    | Bit 0 is the read/write bit.<br>When this bit = 1, a read sequence is requested.<br>When this bit = 0, a write sequence is requested. |

**Table 101. I2CADR0 MMR in 10-Bit Address Mode**

| Bit | Name    | Description  |
|-----|---------|--|
| 7:3 |         | These bits must be set to [11110b] in 10-bit address mode.   |
| 2:1 | I2CMADR | These bits contain ADDR[9:8] in 10-bit addressing mode.  |
| 0   | R/W     | Read/write bit.<br>When this bit = 1, a read sequence is requested.<br>When this bit = 0, a write sequence is requested. |

**I<sup>2</sup>C Address 1, I2CADR1, Register**

Name: I2CADR1  
 Address: 0xFFFF091C  
 Default value: 0x00  
 Access: Read and write  
 Function: This 8-bit MMR is used in 10-bit addressing mode only. This register contains the least significant byte of the address.

**Table 102. I2CADR1 MMR in 10-Bit Address Mode**

| Bit | Name    | Description   |
|-----|---------|---|
| 7:0 | I2CLADR | These bits contain ADDR[7:0] in 10-bit addressing mode. |

**I<sup>2</sup>C Master Clock Control, I2CDIV, Register**

Name: I2CDIV  
 Address: 0xFFFF0924  
 Default value: 0x1F1F  
 Access: Read and write  
 Function: This MMR controls the frequency of the I<sup>2</sup>C clock generated by the master on to the SCL pin. For further details, see the Serial Clock Generation section.

**Table 103. I2CDIV MMR Bit Designations**

| Bit  | Name | Description  |
|------|------|--|
| 15:8 | DIVH | These bits control the duration of the high period of SCL. |
| 7:0  | DIVL | These bits control the duration of the low period of SCL.  |

**I<sup>2</sup>C Slave Registers****I<sup>2</sup>C Slave Control, I2CSCON, Register**

Name: I2CSCON  
 Address: 0xFFFF0928  
 Default value: 0x0000  
 Access: Read and write  
 Function: This 16-bit MMR configures the I<sup>2</sup>C peripheral in slave mode.

Table 104. I2CSCON MMR Bit Designations

| Bit   | Name      | Description   |
|-------|-----------|---|
| 15:11 |           | Reserved bits.  |
| 10    | I2CSTXENI | Slave transmit interrupt enable bit.<br>Set this bit to enable an interrupt after a slave transmits a byte.<br>Clear this interrupt source.   |
| 9     | I2CSRXENI | Slave receive interrupt enable bit.<br>Set this bit to enable an interrupt after the slave receives data.<br>Clear this interrupt source.   |
| 8     | I2CSSENI  | I <sup>2</sup> C stop condition detected interrupt enable bit.<br>Set this bit to enable an interrupt on detecting a stop condition on the I <sup>2</sup> C bus.<br>Clear this interrupt source.  |
| 7     | I2CNACKEN | I <sup>2</sup> C no acknowledge enable bit.<br>Set this bit to no acknowledge the next byte in the transmission sequence.<br>Clear this bit to let the hardware control the acknowledge/no acknowledge sequence.  |
| 6     |           | Reserved. A value of 0 should be written to this bit.   |
| 5     | I2CSETEN  | I <sup>2</sup> C early transmit interrupt enable bit.<br>Setting this bit enables a transmit request interrupt just after the positive edge of SCL during the read bit transmission.<br>Clear this bit to enable a transmit request interrupt just after the negative edge of SCL during the read bit transmission.   |
| 4     | I2CGCCLR  | I <sup>2</sup> C general call status and ID clear bit.<br>Writing a 1 to this bit clears the general call status and ID bits in the I2CSSSTA register.<br>Clear this bit at all other times.  |
| 3     | I2CHGCEN  | Hardware general call enable. When this bit and Bit 2 are set, and having received a general call (Address 0x00) and a data byte, the device checks the contents of the I2CALT against the receive register. If the contents match, the device has received a hardware general call. This is used if a device needs urgent attention from a master device without knowing which master it needs to turn to. This is a "to whom it may concern" call. The <a href="#">ADuC7060/ADuC7061</a> watches for these addresses. The device that requires attention embeds its own address into the message. All masters listen, and the one that can handle the device contacts its slave and acts appropriately. The LSB of the I2CALT register should always be written to 1, as per the I <sup>2</sup> C January 2000 bus specification. |
| 2     | I2CGCEN   | General call enable bit. Set this bit to enable the slave device to acknowledge an I <sup>2</sup> C general call, Address 0x00 (write). The device then recognizes a data bit. If it receives a 0x06 (reset and write programmable part of the slave address by hardware) as the data byte, the I <sup>2</sup> C interface resets as per the I <sup>2</sup> C January 2000 bus specification. This command can be used to reset an entire I <sup>2</sup> C system. If it receives a 0x04 (write programmable part of the slave address by hardware) as the data byte, the general call interrupt status bit sets on any general call. The user must take corrective action by reprogramming the device address.   |
| 1     | ADR10EN   | I <sup>2</sup> C 10-bit address mode.<br>Set to 1 to enable 10-bit address mode.<br>Clear to 0 to enable normal address mode.   |
| 0     | I2CSEN    | I <sup>2</sup> C slave enable bit.<br>Set by user to enable I <sup>2</sup> C slave mode.<br>Clear to disable I <sup>2</sup> C slave mode.   |

**I<sup>2</sup>C Slave Status, I2CSSTA, Register**

Name: I2CSSTA

Address: 0xFFFF092C

Default value: 0x0000

Access: Read and write

Function: This 16-bit MMR is the I<sup>2</sup>C status register in slave mode.**Table 105. I2CSSTA MMR Bit Designations**

| Bit   | Name         | Description  |
|-------|--------------|--|
| 15    |              | Reserved bit.  |
| 14    | I2CSTA       | This bit is set to 1 if a start condition followed by a matching address is detected, a start byte (0x01) is received, or general calls are enabled and a general call code of 0x00 is received.<br>This bit is cleared on receiving a stop condition  |
| 13    | I2CREPS      | This bit is set to 1 if a repeated start condition is detected.<br>This bit is cleared on receiving a stop condition. A read of the I2CSSTA register also clears this bit.   |
| 12:11 | I2CID[1:0]   | I <sup>2</sup> C address matching register. These bits indicate which I2CIDx register matches the received address.<br>[00] = received address matches I2CID0.<br>[01] = received address matches I2CID1.<br>[10] = received address matches I2CID2.<br>[11] = received address matches I2CID3.  |
| 10    | I2CSS        | I <sup>2</sup> C stop condition after start detected bit.<br>This bit is set to 1 when a stop condition is detected after a previous start and matching address. When the I2CSSENI bit in I2CSCON is set, an interrupt is generated.<br>This bit is cleared by reading this register.  |
| 9:8   | I2CGCID[1:0] | I <sup>2</sup> C general call ID bits.<br>[00] = no general call received.<br>[01] = general call reset and program address.<br>[10] = general program address.<br>[11] = general call matching alternative ID.<br>Note that these bits are not cleared by a general call reset command.<br>Clear these bits by writing a 1 to the I2CGCLR bit in I2CSCON.   |
| 7     | I2CGC        | I <sup>2</sup> C general call status bit.<br>This bit is set to 1 if the slave receives a general call command of any type. If the command received was a reset command, then all registers return to their default states. If the command received was a hardware general call, the receive FIFO holds the second byte of the command, and this can be compared with the I2CALT register.<br>Clear this bit by writing a 1 to the I2CGCLR bit in I2CSCON. |
| 6     | I2CSBUSY     | I <sup>2</sup> C slave busy status bit.<br>Set to 1 when the slave receives a start condition.<br>Cleared by hardware if the received address does not match any of the I2CIDx registers, the slave device receives a stop condition, or a repeated start address does not match any of the I2CIDx registers.  |
| 5     | I2CSNA       | I <sup>2</sup> C slave no acknowledge data bit.<br>This bit is set to 1 when the slave responds to a bus address with a no acknowledge. This bit is asserted under the following conditions: if a no acknowledge was returned because there was no data in the transmit FIFO or if the I2CNACKEN bit was set in the I2CSCON register.<br>This bit is cleared in all other conditions.  |
| 4     | I2CSRxFO     | Slave receive FIFO overflow.<br>This bit is set to 1 when a byte is written to the receive FIFO when it is already full.<br>This bit is cleared in all other conditions.   |
| 3     | I2CSRXQ      | I <sup>2</sup> C slave receive request bit.<br>This bit is set to 1 when the receive FIFO of the slave is not empty. This bit causes an interrupt to occur if the I2CSRXENI bit in I2CSCON is set.<br>The receive FIFO must be read or flushed to clear this bit.  |

| Bit | Name     | Description   |
|-----|----------|---|
| 2   | I2CSTXQ  | I <sup>2</sup> C slave transmit request bit.<br>This bit is set to 1 when the slave receives a matching address followed by a read. If the I2CSETEN bit in I2CSCON is =0, this bit goes high just after the negative edge of SCL during the read bit transmission. If the I2CSETEN bit in I2CSCON is =1, this bit goes high just after the positive edge of SCL during the read bit transmission. This bit causes an interrupt to occur if the I2CSTXENI bit in I2CSCON is set.<br>This bit is cleared in all other conditions. |
| 1   | I2CSTFE  | I <sup>2</sup> C slave FIFO underflow status bit.<br>This bit goes high if the transmit FIFO is empty when a master requests data from the slave. This bit is asserted at the rising edge of SCL during the read bit.<br>This bit is cleared in all other conditions.   |
| 0   | I2CETSTA | I <sup>2</sup> C slave early transmit FIFO status bit.<br>If the I2CSETEN bit in I2CSCON is =0, this bit goes high if the slave transmit FIFO is empty. If the I2CSETEN bit in I2CSCON = 1, this bit goes high just after the positive edge of SCL during the write bit transmission. This bit asserts once only for a transfer.<br>This bit is cleared after being read.   |

**I<sup>2</sup>C Slave Receive, I2CSRX, Register**

Name: I2CSRX  
Address: 0xFFFF0930  
Default value: 0x00  
Access: Read only  
Function: This 8-bit MMR is the I<sup>2</sup>C slave receive register.

**I<sup>2</sup>C Slave Transmit, I2CSTX, Register**

Name: I2CSTX  
Address: 0xFFFF0934  
Default value: 0x00  
Access: Write only  
Function: This 8-bit MMR is the I<sup>2</sup>C slave transmit register.

**I<sup>2</sup>C Hardware General Call Recognition, I2CALT, Register**

Name: I2CALT  
Address: 0xFFFF0938  
Default value: 0x00  
Access: Read and write  
Function: This 8-bit MMR is used with hardware general calls when the I2CSCON Bit 3 is set to 1. This register is used in cases where a master is unable to generate an address for a slave and, instead, the slave must generate the address for the master.

**I<sup>2</sup>C Slave Device ID, I2CIDx, Registers**

Name: I2CIDx  
Addresses: 0xFFFF093C = I2CID0  
0xFFFF0940 = I2CID1  
0xFFFF0944 = I2CID2  
0xFFFF0948 = I2CID3  
Default value: 0x00  
Access: Read and write  
Function: These 8-bit MMRs are programmed with the I<sup>2</sup>C bus IDs of the slave. See the I2C Bus Addresses section for further details.

**I<sup>2</sup>C Common Registers****I<sup>2</sup>C FIFO Status, I2CFSTA, Register**

Name: I2CFSTA

Address: 0xFFFF094C

Default value: 0x0000

Access: Read and write

Function: This 16-bit MMR contains the status of the receive/transmit FIFOs in both master and slave modes.

**Table 106. I2CFSTA MMR Bit Designations**

| Bit   | Name      | Description   |
|-------|-----------|---|
| 15:10 |           | Reserved bits.  |
| 9     | I2CFMTX   | Set this bit to 1 to flush the master transmit FIFO.  |
| 8     | I2CFSTX   | Set this bit to 1 to flush the slave transmit FIFO.   |
| 7:6   | I2CMRXSTA | I <sup>2</sup> C master receive FIFO status bits.<br>[00] = FIFO empty.<br>[01] = byte written to FIFO.<br>[10] = one byte in FIFO.<br>[11] = FIFO full.  |
| 5:4   | I2CMTXSTA | I <sup>2</sup> C master transmit FIFO status bits.<br>[00] = FIFO empty.<br>[01] = byte written to FIFO.<br>[10] = one byte in FIFO.<br>[11] = FIFO full. |
| 3:2   | I2CSRSTA  | I <sup>2</sup> C slave receive FIFO status bits.<br>[00] = FIFO empty<br>[01] = byte written to FIFO<br>[10] = one byte in FIFO<br>[11] = FIFO full       |
| 1:0   | I2CSTXSTA | I <sup>2</sup> C slave transmit FIFO status bits.<br>[00] = FIFO empty.<br>[01] = byte written to FIFO.<br>[10] = one byte in FIFO.<br>[11] = FIFO full.  |



## SERIAL PERIPHERAL INTERFACE

The ADuC7060/ADuC7061 integrates a complete hardware serial peripheral interface (SPI) on chip. SPI is an industry standard, synchronous serial interface that allows eight bits of data to be synchronously transmitted and simultaneously received, that is, full duplex up to a maximum bit rate of 2.56 Mbps.

The SPI port can be configured for master or slave operation and typically consists of four pins: MISO, MOSI, SCLK, and  $\overline{SS}$ .

### MISO (MASTER IN, SLAVE OUT) PIN

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

### MOSI (MASTER OUT, SLAVE IN) PIN

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

### SCLK (SERIAL CLOCK I/O) PIN

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

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

$$f_{SERIAL\ CLOCK} = \frac{f_{UCLK}}{2 \times (1 + SPIDIV)}$$

The maximum speed of the SPI clock is independent of the clock divider bits.

```
GP0CON0 = BIT0 + BIT4 + BIT8 + BIT12; //Select SPI/I2C alternative function for P0[0...3]
GP0KEY1 = 0x7; //Write to GP0KEY1
GP0CON1 &=~ BIT1; //Select SPI functionality for P0.0 to P0.3
GP0KEY2 = 0x13; //Write to GP0KEY2
```

In slave mode, the SPICON register must be configured with the phase and polarity of the expected input clock. The slave accepts data from an external master up to 5.12 Mbps.

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

### SLAVE SELECT (P0.0/ $\overline{SS}$ ) INPUT PIN

In SPI slave mode, a transfer is initiated by the assertion of  $\overline{SS}$  on the P0.0/ $\overline{SS}$  pin, 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  $\overline{SS}$ . In slave mode,  $\overline{SS}$  is always an input.

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

### CONFIGURING EXTERNAL PINS FOR SPI FUNCTIONALITY

The SPI pins of the ADuC7060/ADuC7061 device are represented by the P0[0:3] function of the following pins:

- P0.0/ $\overline{SS}$  is the slave chip select pin. In slave mode, this pin is an input and must be driven low by the master. In master mode, this pin is an output and goes low at the beginning of a transfer and high at the end of a transfer.
- P0.1/SCLK/SCL is the SCLK pin.
- P0.2/MISO is the master in, slave out (MISO) pin.
- P0.3/MOSI/SDA is the master out, slave in (MOSI) pin.

To configure P0.0 to P0.3 for SPI mode, Bit 0, Bit 4, Bit 8, and Bit 12 of the GP0CON0 register must be set to 1. Bit 1 of the GP0CON1 must be set to 1. Note that to write to GP0CON1, the GP0KEY1 register must be set to 0x7 immediately before writing to GP0CON1. Also, the GP0KEY2 register must be set to 0x13 immediately after writing to GP0CON1. The following code example shows this in detail:

**SPI REGISTERS**

The following MMR registers control the SPI interface: SPISTA, SPIRX, SPITX, SPIDIV, and SPICON.

**SPI Status Register****SPISTA Register**

Name: SPISTA

Address: 0xFFFF0A00

Default value: 0x00000000

Access: Read only

Function: This 32-bit MMR contains the status of the SPI interface in both master and slave modes.

**Table 107. SPISTA MMR Bit Designations**

| Bit   | Name           | Description  |
|-------|----------------|--|
| 15:12 |                | Reserved bits.   |
| 11    | SPIREX         | SPI receive FIFO excess bytes present. This bit is set when there are more bytes in the receive FIFO than indicated in the SPIMDE bits in SPICON.<br>This bit is cleared when the number of bytes in the FIFO is equal to or less than the number in SPIMDE. |
| 10:8  | SPIRXFSTA[2:0] | SPI receive FIFO status bits.<br>[000] = receive FIFO is empty.<br>[001] = 1 valid byte in the FIFO.<br>[010] = 2 valid bytes in the FIFO.<br>[011] = 3 valid bytes in the FIFO.<br>[100] = 4 valid bytes in the FIFO.                                       |
| 7     | SPIFOF         | SPI receive FIFO overflow status bit.<br>Set when the receive FIFO was already full when new data was loaded to the FIFO. This bit generates an interrupt except when SPIRFLH is set in SPICON.<br>Cleared when the SPISTA register is read.                 |
| 6     | SPIRXIRQ       | SPI receive IRQ status bit.<br>Set when a receive interrupt occurs. This bit is set when SPITMDE in SPICON is cleared and the required number of bytes has been received.<br>Cleared when the SPISTA register is read.                                       |
| 5     | SPITXIRQ       | SPI transmit IRQ status bit.<br>Set when a transmit interrupt occurs. This bit is set when SPITMDE in SPICON is set and the required number of bytes has been transmitted.<br>Cleared when the SPISTA register is read.                                      |
| 4     | SPITXUF        | SPI transmit FIFO underflow.<br>This bit is set when a transmit is initiated without any valid data in the transmit FIFO. This bit generates an interrupt except when SPITFLH is set in SPICON.<br>Cleared when the SPISTA register is read.                 |
| 3:1   | SPITXFSTA[2:0] | SPI transmit FIFO status bits.<br>[000] = transmit FIFO is empty.<br>[001] = 1 valid bytes in the FIFO.<br>[010] = 2 valid bytes in the FIFO.<br>[011] = 3 valid bytes in the FIFO.<br>[100] = 4 valid bytes in the FIFO.                                    |
| 0     | SPIISTA        | SPI interrupt status bit.<br>Set to 1 when an SPI based interrupt occurs.<br>Cleared after reading SPISTA.   |

**SPI Receive Register****SPIRX Register**

Name: SPIRX  
 Address: 0xFFFF0A04  
 Default value: 0x00  
 Access: Read only  
 Function: This 8-bit MMR is the SPI receive register.

**SPI Transmit Register****SPITX Register**

Name: SPITX  
 Address: 0xFFFF0A08  
 Default value: 0x00  
 Access: Write only  
 Function: This 8-bit MMR is the SPI transmit register.

**SPI Baud Rate Selection Register****SPIDIV Register**

Name: SPIDIV  
 Address: 0xFFFF0A0C  
 Default value: 0x1B  
 Access: Write only  
 Function: This 8-bit MMR is the SPI baud rate selection register.

**Table 108. SPIDIV MMR Bit Designations**

| Bit | Description  |
|-----|--|
| 7:6 | Reserved.  |
| 5:0 | SPI Baud rate setting:<br>$f_{\text{SERIAL CLOCK}} = \frac{f_{\text{UCLK}}}{2 \times (1 + \text{SPIDIV})}$ |

**SPI Control Register****SPICON Register**

Name: SPICON  
 Address: 0xFFFF0A10  
 Default value: 0x0000  
 Access: Read and write  
 Function: This 16-bit MMR configures the SPI peripheral in both master and slave modes.

Table 109. SPICON MMR Bit Designations

| Bit   | Name    | Description   |
|-------|---------|---|
| 15:14 | SPIMDE  | SPI IRQ mode bits. These bits are configured when transmit/receive interrupts occur in a transfer.<br>[00] = transmit interrupt occurs when 1 byte has been transferred. Receive interrupt occurs when one or more bytes have been received into the FIFO.<br>[01] = transmit interrupt occurs when 2 bytes have been transferred. Receive interrupt occurs when two or more bytes have been received into the FIFO.<br>[10] = transmit interrupt occurs when 3 bytes have been transferred. Receive interrupt occurs when three or more bytes have been received into the FIFO.<br>[11] = transmit interrupt occurs when 4 bytes have been transferred. Receive interrupt occurs when the receive FIFO is full or 4 bytes are present. |
| 13    | SPITFLH | SPI transmit FIFO flush enable bit.<br>Set this bit to flush the transmit FIFO. This bit does not clear itself and should be toggled if a single flush is required. If this bit is left high, then either the last transmitted value or 0x00 is transmitted, depending on the SPIZEN bit. Any writes to the transmit FIFO are ignored while this bit is set.<br>Clear this bit to disable transmit FIFO flushing.   |
| 12    | SPIRFLH | SPI receive FIFO flush enable bit.<br>Set this bit to flush the receive FIFO. This bit does not clear itself and should 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 SPITMDE = 0, a read of the receive FIFO initiates a transfer.<br>Clear this bit to disable receive FIFO flushing.  |
| 11    | SPICONT | Continuous transfer enable.<br>Set by user to enable continuous transfer. In master mode, the transfer continues until no valid data is available in the transmit register. $\overline{SS}$ is asserted and remains asserted for the duration of each 8-bit serial transfer until the transmit register is empty.<br>Cleared by user to disable continuous transfer. Each transfer consists of a single 8-bit serial transfer. If valid data exists in the SPITX register, then a new transfer is initiated after a stall period of one serial clock cycle.   |
| 10    | SPILP   | Loopback enable bit.<br>Set by user to connect MISO to MOSI and test software.<br>Cleared by user to be in normal mode.   |
| 9     | SPIOEN  | Slave MISO output enable bit.<br>Set this bit for MISO to operate as normal.<br>Clear this bit to disable the output driver on the MISO pin. The MISO pin is open drain when this bit is cleared.   |
| 8     | SPIROW  | SPIRX overflow overwrite enable.<br>Set by user, the valid data in the receive register is overwritten by the new serial byte received.<br>Cleared by user, the new serial byte received is discarded.  |
| 7     | SPIZEN  | SPI transmit zeros when transmit FIFO is empty.<br>Set this bit to transmit 0x00 when there is no valid data in the transmit FIFO.<br>Clear this bit to transmit the last transmitted value when there is no valid data in the transmit FIFO.   |
| 6     | SPITMDE | SPI transfer and interrupt mode.<br>Set by user to initiate transfer with a write to the SPITX register. Interrupt occurs only when the transmit FIFO is empty.<br>Cleared by user to initiate transfer with a read of the SPI register. Interrupt occurs only when the receive FIFO is full.   |
| 5     | SPILF   | LSB first transfer enable bit.<br>Set by user, the LSB is transmitted first.<br>Cleared by user, the MSB is transmitted first.  |
| 4     | SPIWOM  | SPI wired or mode enable bit.<br>Set to 1 to enable the open-drain data output enable. External pull-ups are required on data out pins.<br>Clear for normal output levels.  |
| 3     | SPICPO  | Serial clock polarity mode bit.<br>Set by user, the serial clock idles high.<br>Cleared by user, the serial clock idles low.  |
| 2     | SPICPH  | Serial clock phase mode bit.<br>Set by user, the serial clock pulses at the beginning of each serial bit transfer.<br>Cleared by user, the serial clock pulses at the end of each serial bit transfer.  |

| Bit | Name  | Description  |
|-----|-------|--|
| 1   | SPIEN | Master mode enable bit.<br>Set by user to enable master mode.<br>Cleared by user to enable slave mode. |
| 0   | SPIEN | SPI enable bit.<br>Set by user to enable the SPI.<br>Cleared by user to disable the SPI.               |

## GENERAL-PURPOSE I/O

The ADuC7060/ADuC7061 features up to 16 general-purpose bidirectional input/output (GPIO) pins. In general, many of the GPIO pins have multiple functions that are configurable by user code. By default, the GPIO pins are configured in GPIO mode. All GPIO pins have an internal pull-up resistor with a drive capability of 1.6 mA.

All I/O pins are 3.3 V tolerant, meaning that the GPIOs support an input voltage of 3.3 V.

When the ADuC7060/ADuC7061 enters power-saving mode, the GPIO pins retain their state.

The GPIO pins are grouped into three port buses.

Table 110 lists all the GPIO pins and their alternative functions. A GPIO pin alternative function can be selected by writing to the correct bits of the GPxCON register.

**Table 110. GPIO Multifunction Pin Descriptions**

| Port | Pin Mnemonic          | Configuration via GPxCON Including GP0CON0 |                                    |
|------|-----------------------|--|------------------------------------|
|      |                       | 00   | 01                                 |
| 0    | P0.0/SS               | GPIO                                       | SS (SPI slave select).             |
|      | P0.1/SCLK/SCL         | GPIO                                       | SCLK/SCL (serial clock/SPI clock). |
|      | P0.2/MISO             | GPIO                                       | MISO (SPI—master in/slave out).    |
|      | P0.3/MOSI/SDA         | GPIO                                       | MOSI (SPI—master out/slave in).    |
|      | P0.4/IRQ0/PWM1        | GPIO/IRQ0                                  | PWM1 (PWM Output 1).               |
|      | P0.5/CTS              | GPIO                                       | CTS. UART clear to send pin.       |
|      | P0.6/RTS              | GPIO                                       | RTS. UART request to send pin.     |
| 1    | P1.0/IRQ1/SIN/T0      | GPIO/IRQ1                                  | SIN (serial input).                |
|      | P1.1/SOUT             | GPIO                                       | SOUT (serial output).              |
|      | P1.2/SYNC             | GPIO                                       | PWM sync (PWM sync input pin).     |
|      | P1.3/TRIP             | GPIO                                       | PWM trip (PWM trip input pin).     |
|      | P1.4/PWM2             | GPIO                                       | PWM2 (PWM Output 2).               |
|      | P1.5/PWM3             | GPIO                                       | PWM3 (PWM Output 3).               |
|      | P1.6/PWM4             | GPIO                                       | PWM4 (PWM Output 4).               |
| 2    | P2.0/IRQ2/PWM0/EXTCLK | GPIO/IRQ2/EXTCLK                           | PWM0 (PWM Output 0).               |
|      | P2.1/IRQ3/PWM5        | GPIO/IRQ3                                  | PWM5 (PWM Output 5).               |

## GPxCON REGISTERS

GPxCON are the Port x (where x is 0, 1, or 2) control registers, which select the function of each pin of Port x as described in Table 112.

**Table 111. GPxCON Registers**

| Name    | Address    | Default Value | Access |
|---------|------------|---------------|--------|
| GP0CON0 | 0xFFFF0D00 | 0x00000000    | R/W    |
| GP1CON  | 0xFFFF0D04 | 0x00000000    | R/W    |
| GP2CON  | 0xFFFF0D08 | 0x00000000    | R/W    |

Table 112. GPxCON MMR Bit Designations

| Bit   | Description   |
|-------|---|
| 31:30 | Reserved.   |
| 29:28 | Reserved.   |
| 27:26 | Reserved.   |
| 25:24 | Selects the function of the P0.6/RTS and P1.6/PWM pins.   |
| 23:22 | Reserved.   |
| 21:20 | Selects the function of the P0.5/CTS and P1.5/PWM3 pins.  |
| 19:18 | Reserved.   |
| 17:16 | Selects the function of the P0.4/IRQ0/PWM1 and P1.4/PWM2 pins.                                    |
| 15:14 | Reserved.   |
| 13:12 | Selects the function of the P0.3/MOSI/SDA and P1.3/TRIP pins.                                     |
| 11:10 | Reserved.   |
| 9:8   | Selects the function of the P0.2/MISO and P1.2/SYNC pins.   |
| 7:6   | Reserved.   |
| 5:4   | Selects the function of the P0.1/SCLK/SCL, P1.1/SOUT, and P2.1/IRQ3/PWM5 pins.                    |
| 3:2   | Reserved.   |
| 1:0   | Selects the function of the P0.0/ $\overline{SS}$ , P1.0/IRQ1/SIN/T0, P2.0/IRQ2/PWM0/EXTCLK pins. |

### GPxDAT REGISTERS

GPxDAT are Port x configuration and data registers. They configure the direction of the GPIO pins of Port x, set the output value for the pins that are configured as output, and store the input value of the pins that are configured as input.

Table 113. GPxDAT Registers

| Name   | Address    | Default Value | Access |
|--------|------------|---------------|--------|
| GP0DAT | 0xFFFF0D20 | 0x000000XX    | R/W    |
| GP1DAT | 0xFFFF0D30 | 0x000000XX    | R/W    |
| GP2DAT | 0xFFFF0D40 | 0x000000XX    | R/W    |

Table 114. GPxDAT MMR Bit Designations

| Bit   | Description   |
|-------|---|
| 31:24 | Direction of the data.<br>Set to 1 by user to configure the GPIO pin as an output.<br>Cleared to 0 by user to configure the GPIO pin as an input. |
| 23:16 | Port x data output.   |
| 15:8  | Reflect the state of Port x pins at reset (read only).  |
| 7:0   | Port x data input (read only).  |

### GPxSET REGISTERS

GPxSET are data set Port x registers.

Table 115. GPxSET Registers

| Name   | Address    | Default Value | Access |
|--------|------------|---------------|--------|
| GP0SET | 0xFFFF0D24 | 0x000000XX    | W      |
| GP1SET | 0xFFFF0D34 | 0x000000XX    | W      |
| GP2SET | 0xFFFF0D44 | 0x000000XX    | W      |

Table 116. GPxSET MMR Bit Designations

| Bit   | Description   |
|-------|---|
| 31:24 | Reserved.   |
| 23:16 | Data Port x set bit.<br>Set to 1 by user to set bit on Port x; also sets the corresponding bit in the GPxDAT MMR.<br>Cleared to 0 by user; does not affect the data output. |
| 15:0  | Reserved.   |

### GPxCLR REGISTERS

GPxCLR are data clear Port x registers.

Table 117. GPxCLR Registers

| Name   | Address    | Default Value | Access |
|--------|------------|---------------|--------|
| GP0CLR | 0xFFFF0D28 | 0x000000XX    | W      |
| GP1CLR | 0xFFFF0D38 | 0x000000XX    | W      |
| GP2CLR | 0xFFFF0D48 | 0x000000XX    | W      |

Table 118. GPxCLR MMR Bit Designations

| Bit   | Description   |
|-------|---|
| 31:24 | Reserved.   |
| 23:16 | Data Port x clear bit.<br>Set to 1 by user to clear the bit on Port x; also clears the corresponding bit in the GPxDAT MMR.<br>Cleared to 0 by user; does not affect the data output. |
| 15:0  | Reserved.   |

### GPxPAR REGISTERS

The GPxPAR registers program the parameters for Port 0, Port 1, and Port 2. Note that the GPxDAT MMR must always be written after changing the GPxPAR MMR. Note that it is not possible to disable the internal pull-up resistor on P0.2.

Table 119. GPxPAR Registers

| Name   | Address    | Default Value | Access |
|--------|------------|---------------|--------|
| GP0PAR | 0xFFFF0D2C | 0x00000000    | R/W    |
| GP1PAR | 0xFFFF0D3C | 0x00000000    | R/W    |
| GP2PAR | 0xFFFF0D4C | 0x00000000    | R/W    |

**Table 120. GPxPAR MMR Bit Designations**

| Bit   | Name      | Description  |
|-------|-----------|--|
| 31:15 |           | Reserved.  |
| 23:16 | GPL[7:0]  | General I/O port pin functionality lock registers.<br>GPL[7:0] = 0, normal operation.<br>GPL[7:0] = 1, for each GPIO pin, if this bit is set, writing to the corresponding bit in GPxCON or GPxDAT register bit has no effect. |
| 15:8  | GPDS[7:0] | Drive strength configuration. This bit is configurable.<br>GPDS[x] = 0, maximum source current is 2 mA.<br>GPDS[x] = 1, maximum source current is 4 mA.  |
| 7:0   | GPPD[7:0] | Pull-Up Disable Port x[7:0].<br>GPPD[x] = 0, pull-up resistor is active.<br>GPPD[x] = 1, pull-up resistor is disabled.   |

### GP0CON1 Control Registers

The GP0CON1 write values are as follows: GP0KEY1 = 0x7, GP0CON1 = user value, and GP0KEY2 = 0x13.

Name: GP0CON1  
 Address: 0xFFFF0468  
 Default value: 0x00  
 Access: Read and write  
 Function: This register controls the P0.0, P0.1, P0.2, and P0.3 functionality of the multifunction GPIO pins.

**Table 121. GP0CON1 Write Sequence**

| Name    | Value      |
|---------|------------|
| GP0KEY1 | 0x7        |
| GP0CON1 | User value |
| GP0KEY2 | 0x13       |

**Table 122. GP0CON1 MMR Bit Designations**

| Bit | Name          | Description   |
|-----|---------------|---|
| 7:2 | Reserved      | These bits must always be set to 0.   |
| 1   | SPII2CS<br>EL | This bit configures the P0.0 to P0.3 functions in I <sup>2</sup> C or SPI mode. Note that Bit 0 of GP0CON1 must be set to 0 for this bit to work.<br>To select the P0.0, P0.1, P0.2, and P0.3 functions in SPI mode, clear this bit to 0.<br>To select the P0.0, P0.1, P0.2, and P0.3 functions in I <sup>2</sup> C mode, set this bit to 1.<br>This bit is cleared by default. |
| 0   | ADCSEL        | This bit configures the P0.0 to P0.3 functions as GPIO pins or as ADC input pins.<br>To enable P0.0, P0.1, P0.2 and P0.3 functions as ADC inputs, set this bit to 1.<br>To enable P0.0, P0.1, P0.2, and P0.3 functions as digital I/O, clear this bit to 0.<br>This bit is cleared by default.  |

Name: GP0KEY1  
 Address: 0xFFFF0464  
 Default value: 0xFFFF  
 Access: Write only  
 Function: When writing to GP0CON1, the value of 0x07 must be written to this register in the instruction immediately before writing to GP0CON1.

Name: GP0KEY2  
 Address: 0xFFFF046C  
 Default value: 0xFFFF  
 Access: Write only  
 Function: When writing to GP0CON1, the value of 0x13 must be written to this register in the instruction immediately after writing to GP0CON1.



## HARDWARE DESIGN CONSIDERATIONS

### POWER SUPPLIES

The ADuC7060/ADuC7061 operational power supply voltage range is 2.375 V to 2.625 V. Separate analog and digital power supply pins (AVDD and DVDD, respectively) allow AVDD to be kept relatively free of noisy digital signals often present on the system DVDD line. In this mode, the part can also operate with split supplies; that is, it can use different voltage levels for each supply. For example, the system can be designed to operate with a DVDD voltage level of 2.6 V, whereas the AVDD level can be at 2.5 V or vice versa. A typical split supply configuration is shown in Figure 28.

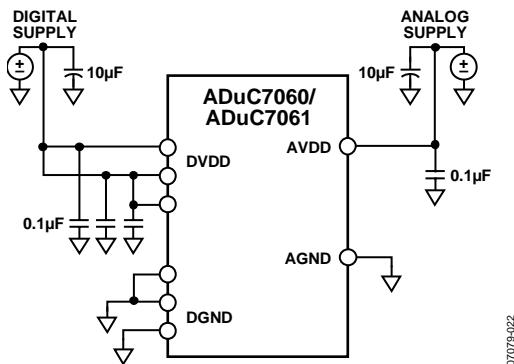


Figure 28. External Dual Supply Connections

As an alternative to providing two separate power supplies, the user can reduce noise on AVDD by placing a small series resistor and/or ferrite bead between AVDD and DVDD, and then decoupling AVDD separately to ground. An example of this configuration is shown in Figure 29. With this configuration, other analog circuitry (such as op amps, voltage reference, and others) can be powered from the AVDD supply line as well.

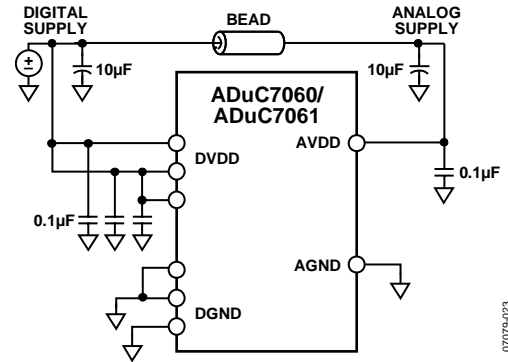


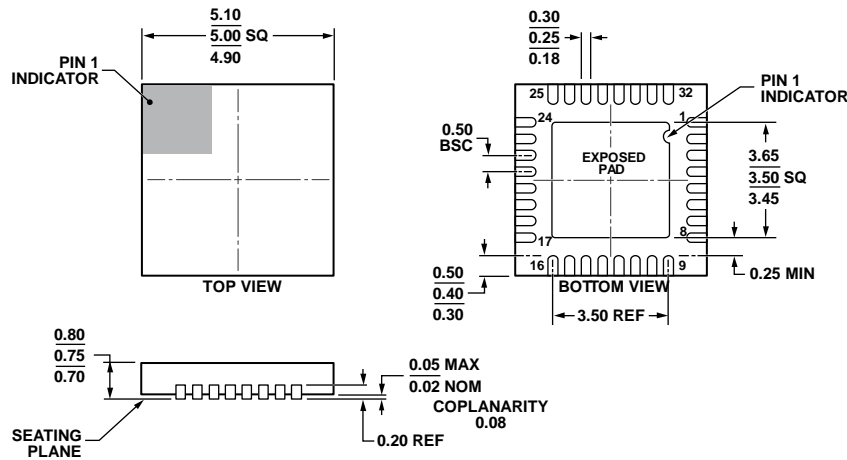
Figure 29. External Single Supply Connections

Notice that in both Figure 28 and Figure 29, a large value (10 µF) reservoir capacitor sits on DVDD, and a separate 10 µF capacitor sits on AVDD. In addition, local, small value (0.1 µF) capacitors are located at each AVDD and DVDD pin of the chip. As per standard design practice, be sure to include all of these capacitors and ensure that the smaller capacitors are close to the AVDD pin with trace lengths as short as possible. Connect the ground terminal of each of these capacitors directly to the underlying ground plane.

Note that the analog and digital ground pins on the ADuC7060/ADuC7061 must be referenced to the same system ground reference point at all times.

Finally, note that, when the DVDD supply reaches 1.8 V, it must ramp to 2.25 V in less than 128 ms. This is a requirement of the internal power-on reset circuitry.

OUTLINE DIMENSIONS

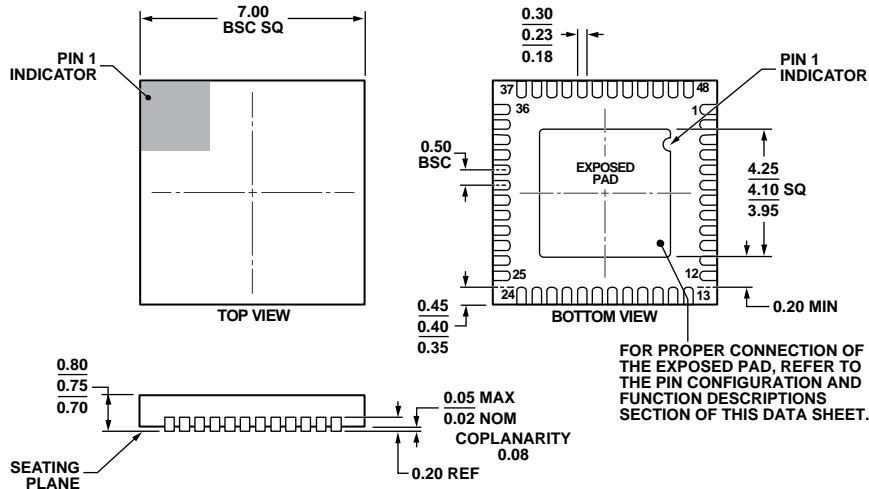


COMPLIANT TO JEDEC STANDARDS MO-220-WHHD.

Figure 30. 32-Lead Lead Frame Chip Scale Package [LFCSP]  
5 mm x 5 mm Body and 0.75 mm Package Height  
(CP-32-11)

Dimensions shown in millimeters

04-02-2012-A



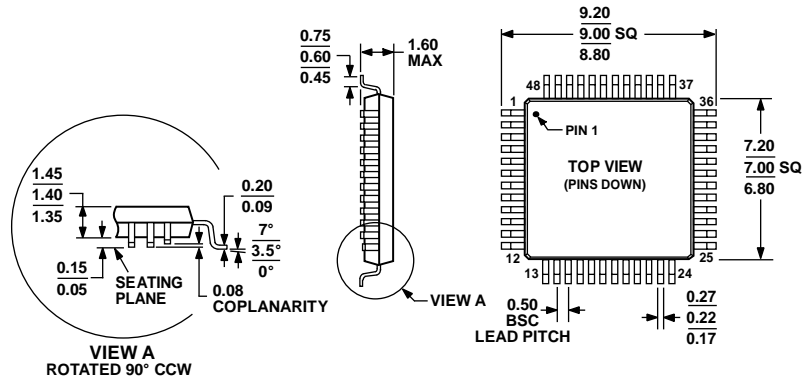
FOR PROPER CONNECTION OF THE EXPOSED PAD, REFER TO THE PIN CONFIGURATION AND FUNCTION DESCRIPTIONS SECTION OF THIS DATA SHEET.

COMPLIANT TO JEDEC STANDARDS MO-220-WKGD.

Figure 31. 48-Lead Lead Frame Chip Scale Package [LFCSP]  
7 mm x 7 mm Body and 0.75 mm Package Height  
(CP-48-5)

Dimensions shown in millimeters

08-16-2010-B



COMPLIANT TO JEDEC STANDARDS MS-026-BBC

Figure 32. 48-Lead Low Profile Quad Flat Package [LQFP] (ST-48)  
Dimensions shown in millimeters

051706-A

**ORDERING GUIDE**

| Model <sup>1</sup> | Temperature Range | Package Description                           | Package Option | Ordering Quantity |
|--------------------|-------------------|---|----------------|-------------------|
| ADuC7060BCPZ32     | -40°C to +125°C   | 48-Lead Lead Frame Chip Scale Package [LFCSP] | CP-48-5        |                   |
| ADuC7060BCPZ32-RL  | -40°C to +125°C   | 48-Lead Lead Frame Chip Scale Package [LFCSP] | CP-48-5        | 2,500             |
| ADuC7060BSTZ32     | -40°C to +125°C   | 48-Lead Low Profile Quad Flat Package [LQFP]  | ST-48          |                   |
| ADuC7060BSTZ32-RL  | -40°C to +125°C   | 48-Lead Low Profile Quad Flat Package [LQFP]  | ST-48          | 2,000             |
| ADuC7061BCPZ32     | -40°C to +125°C   | 32-Lead Lead Frame Chip Scale Package [LFCSP] | CP-32-11       |                   |
| ADuC7061BCPZ32-RL  | -40°C to +125°C   | 32-Lead Lead Frame Chip Scale Package [LFCSP] | CP-32-11       | 5,000             |
| EVAL-ADuC7060QSPZ  |                   | ADuC7060 Quick Start Plus Development System  |                |                   |
| EVAL-ADuC7061MKZ   |                   | ADuC7061 Quick Start Evaluation System        |                |                   |

<sup>1</sup> Z = RoHS Compliant Part.