



#### FEATURES

##### Analog I/O

- Multichannel, 12-bit, 1 MSPS ADC
- Up to 12 ADC channels
- Fully differential and single-ended modes
- 0 V to  $V_{REF}$  analog input range
- 12-bit voltage output DACs
- 4 DAC outputs available
- On-chip voltage reference
- On-chip temperature sensor
- Voltage comparator

##### Microcontroller

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

##### Clocking options

- Trimmed on-chip oscillator ( $\pm 3\%$ )
- External watch crystal
- External clock source up to 44 MHz
- 41.78 MHz PLL with programmable divider

##### Memory

- 62 kB Flash/EE memory, 8 kB SRAM
- In-circuit download, JTAG-based debug
- Software-triggered in-circuit reprogrammability

##### Vectored interrupt controller for FIQ and IRQ

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

##### On-chip peripherals

- 2 $\times$  fully I<sup>2</sup>C-compatible channels
- SPI (20 Mbps in master mode, 10 Mbps in slave mode)
- With 4-byte FIFO on input and output stages
- Up to 20 GPIO pins—Digital only GPIOs are 5 V tolerant
- 3 $\times$  general-purpose timers
- Watchdog timer (WDT)
- Programmable logic array (PLA)
- 16 PLA elements
- 16-bit, 5-channel PWM

##### Power

- Specified for 3 V operation
- Active mode: 11 mA at 5 MHz, 28 mA at 41.78 MHz

##### Packages and temperature range

- 32-lead 5 mm  $\times$  5 mm LFCSP
- 40-lead LFCSP
- 36-Lead WLCSP

##### Fully specified for $-40^{\circ}\text{C}$ to $+125^{\circ}\text{C}$ operation

##### Tools

- Low cost QuickStart development system
- Full third-party support

#### APPLICATIONS

- Optical networking
- Industrial control and automation systems
- Smart sensors, precision instrumentation
- Base station systems

#### GENERAL DESCRIPTION

The ADuC7023 is a fully integrated, 1 MSPS, 12-bit data acquisition system, incorporating high performance multichannel ADCs, 16-bit/32-bit MCUs, and Flash/EE memory on a single chip.

The ADC consists of up to 12 single-ended inputs. An additional four inputs are available but are multiplexed with the four DAC output pins. The ADC can operate in single-ended or differential input modes. The ADC input voltage is 0 V to  $V_{REF}$ . A low drift band gap reference, temperature sensor, and voltage comparator complete the ADC peripheral set.

The DAC output range is programmable to one of two voltage ranges. The DAC outputs have an enhanced feature of being able to retain their output voltage during a watchdog or software reset sequence.

The devices operate from an on-chip oscillator and a PLL, generating an internal high frequency clock of 41.78 MHz. This clock is routed through a programmable clock divider from which the MCU core clock operating frequency is generated. The microcontroller core is an ARM7TDMI<sup>®</sup>, 16-bit/32-bit RISC machine that offers up to 41 MIPS peak performance. Eight kilobytes of SRAM and 62 kilobytes of nonvolatile Flash/EE memory are provided on chip. The ARM7TDMI core views all memory and registers as a single linear array.

The ADuC7023 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 are supported.

On-chip factory firmware supports in-circuit download via the I<sup>2</sup>C serial interface port, and nonintrusive emulation is supported via the JTAG interface. These features are incorporated into a low cost QuickStart<sup>™</sup> development system supporting this MicroConverter<sup>®</sup> family. The part contains a 16-bit PWM with five output signals.

For communication purposes, the part contains 2  $\times$  I<sup>2</sup>C channels that can be individually configured for master or slave mode. An SPI interface supporting both master and slave modes is also provided.

The parts operate from 2.7 V to 3.6 V and are specified over an industrial temperature range of  $-40^{\circ}\text{C}$  to  $+125^{\circ}\text{C}$ . The ADuC7023 is available in either a 32-lead or 40-lead LFCSP package. A 36-ball wafer level CSP package (WLCSP) is also available.

## TABLE OF CONTENTS

|  |    |  |    |
|--|----|--|----|
| Features .....                                     | 1  | Security .....   | 39 |
| Applications.....                                  | 1  | Flash/EE Control Interface .....                                   | 39 |
| General Description .....                          | 1  | Execution Time from SRAM and Flash/EE.....                         | 42 |
| Revision History .....                             | 3  | Reset and Remap .....  | 42 |
| Functional Block Diagram .....                     | 5  | Other Analog Peripherals.....                                      | 45 |
| Specifications.....                                | 6  | DAC.....   | 45 |
| Timing Specifications .....                        | 9  | Power Supply Monitor .....   | 47 |
| Absolute Maximum Ratings.....                      | 14 | Comparator .....   | 47 |
| ESD Caution.....                                   | 14 | Oscillator and PLL—Power Control.....                              | 49 |
| Pin Configurations and Function Descriptions ..... | 15 | Digital Peripherals.....   | 52 |
| Typical Performance Characteristics .....          | 19 | General-Purpose Input/Output.....                                  | 52 |
| Terminology .....                                  | 20 | Serial Peripheral Interface.....                                   | 55 |
| ADC Specifications .....                           | 20 | I <sup>2</sup> C.....  | 60 |
| DAC Specifications.....                            | 20 | Configuring External Pins for I <sup>2</sup> C Functionality ..... | 60 |
| Overview of the ARM7TDMI Core .....                | 21 | Serial Clock Generation .....                                      | 60 |
| Thumb Mode (T).....                                | 21 | I <sup>2</sup> C Bus Addresses.....                                | 60 |
| Long Multiply (M).....                             | 21 | I <sup>2</sup> C Registers .....                                   | 61 |
| EmbeddedICE (I) .....                              | 21 | Programmable Logic Array (PLA).....                                | 68 |
| Exceptions .....                                   | 21 | Pulse-Width Modulator.....   | 72 |
| ARM Registers .....                                | 21 | Pulse-Width Modulator General Overview .....                       | 72 |
| Interrupt Latency.....                             | 22 | Processor Reference Peripherals.....                               | 77 |
| Memory Organization .....                          | 23 | Interrupt System.....  | 77 |
| Memory Access.....                                 | 23 | IRQ .....  | 77 |
| Flash/EE Memory.....                               | 23 | Fast Interrupt Request (FIQ) .....                                 | 78 |
| SRAM.....  | 23 | Vectored Interrupt Controller (VIC).....                           | 79 |
| Memory Mapped Registers .....                      | 23 | Timers.....  | 84 |
| ADC Circuit Overview .....                         | 30 | Hardware Design Considerations.....                                | 89 |
| Transfer Function .....                            | 30 | Power Supplies .....   | 89 |
| Typical Operation.....                             | 31 | Grounding and Board Layout Recommendations.....                    | 90 |
| MMR Interface.....                                 | 31 | Clock Oscillator .....   | 90 |
| Converter Operation.....                           | 34 | Power-On Reset Operation.....                                      | 91 |
| Driving the Analog Inputs .....                    | 35 | Typical System Configuration .....                                 | 92 |
| Calibration.....                                   | 35 | Development Tools.....   | 93 |
| Temperature Sensor .....                           | 35 | PC-Based Tools.....  | 93 |
| Band Gap Reference.....                            | 37 | In-Circuit I <sup>2</sup> C Downloader .....                       | 93 |
| Nonvolatile Flash/EE Memory .....                  | 38 | Outline Dimensions .....   | 94 |
| Programming .....                                  | 38 | Ordering Guide .....   | 96 |

**REVISION HISTORY****1/15—Rev. F to Rev. G**

|  |    |
|--|----|
| Changes to Table 53 .....                          | 51 |
| Changes to I <sup>2</sup> C Section .....          | 60 |
| Changes to Table 65 .....                          | 61 |
| Changes to Table 72 .....                          | 64 |
| Changes to I2CREPS Bit Description, Table 73 ..... | 66 |

**5/14—Rev. E to Rev. F**

|  |            |
|--|------------|
| Change CONV <sub>START</sub> Pin to CONV <sub>START</sub> Pin..... | Throughout |
| Change to Layout, Power Requirements Parameter, Table 1.....       | 7          |
| Change to Table 8 .....  | 13         |
| Changes to Figure 7 and Table 9 .....                              | 14         |
| Change to Table 21 .....   | 28         |
| Change to Figure 23 .....  | 30         |
| Change to JTAG Access Section .....                                | 37         |
| Changes to Table 36 .....  | 42         |
| Changes to Table 55 .....  | 51         |
| Changes to I <sup>2</sup> C Bus Addresses Section .....            | 59         |
| Change to Table 84.....  | 71         |
| Added PWM2LEN Register Section.....                                | 75         |

**7/13—Rev. D to Rev. E**

|                                |    |
|--------------------------------|----|
| Changes to Ordering Guide..... | 95 |
|--------------------------------|----|

**7/13—Rev. C to Rev. D**

|  |    |
|--|----|
| Added WLCSP (Throughout) .....   | 1  |
| Changes to Features Section .....  | 1  |
| Added Shared Analog/Digital Inputs to AGND Rating of -0.3 V to AV <sub>DD</sub> + 0.3 V, Endnote 1, and Endnote 2; Table 8 ..... | 13 |
| Added Figure 9; Renumbered Sequentially; Added WLCSP Pin Numbers to Table 9 .....  | 14 |
| Changes to Pin P1.7/PWM3/SDA1/PLAI[6] and Pin P1.6/PWM2/SCL1/PLAI[5] Descriptions; Table 9.....                                  | 16 |
| Changes to ADC Circuit Overview Section, Transfer Function Section, and Figure 20 Caption .....                                  | 29 |
| Changes to Typical Operation Section, ADCCON Register Section, and ADCON[13] Description in Table 24.....                        | 30 |
| Changes to Bits[4:3] Value 10 Description; Table 24.....   | 31 |
| Changes to Converter Operation Section and Deleted Pseudo Differential Mode Section .....  | 33 |
| Changes to Figure 27 and Figure 28 Caption .....   | 34 |
| Changes to Table 30 and Following Text.....  | 36 |
| Changes to JTAG Access Section .....   | 37 |
| Changes to References to ADC and the DACs Section .....  | 45 |
| Changes to General-Purpose Input/Output Section .....  | 51 |
| Changes to SPIDIV Register Section .....   | 56 |
| Changes to Bits[1:0] Value 01 Description; Table 66.....   | 61 |
| Changes to T0CLRI Register Section .....   | 84 |
| Changes to Figure 53 .....   | 90 |
| Updated Outline Dimensions.....  | 93 |
| Changes to Ordering Guide.....   | 95 |

**5/12—Rev. B to Rev. C**

|  |    |
|--|----|
| Changed SDATA to SDA and SCLK to SCL, Table 2; SDATA to SDA and SCLK to SCL, Table 3; and SDATA to SDA and SCLK to SCL, Figure 2 ..... | 8  |
| Changes to Figure 7, Figure 8, and Table 9.....  | 14 |
| Changed SCLK to SCL, Table 17.....   | 25 |
| Changed SCLK to SCL, Table 18.....   | 26 |
| Changes to Bit 6, Table 24 and 4 to 0, Description Column, Table 25.....   | 30 |
| Changed Reference in REFCON Register Section from Table 22 to Table 30 .....   | 35 |
| Added Note 1 to Table 53.....  | 49 |
| Changes to Note 1, Table 55.....   | 50 |
| Changed SPICLK (Serial Clock I/O) Pin Section to SCLK (Serial Clock I/O) Pin Section.....  | 53 |
| Changed SPICLK to SCLK in Serial Peripheral Interface Section and in SCLK (Serial Clock I/O) Pin Section .....                         | 53 |
| Changes to Table 79 .....  | 68 |
| Changes to Timers Section .....  | 82 |
| Added Hours, Minutes, Seconds, and 1/128 Format Section and Table 101.....   | 82 |
| Changes to T0LD Register Section and T1LD Register Section.....  | 83 |
| Changes to T2LD Register Section.....  | 85 |
| Updated Outline Dimensions.....  | 92 |
| Changes to Ordering Guide.....   | 93 |

**7/10—Rev. A to Rev. B**

|  |    |
|--|----|
| Changes to Temperature Sensor Parameter in Table 1 .....                     | 6  |
| Change to Table 10 and changes to Table 11 .....                             | 23 |
| Changes to Table 12 and Table 13.....  | 24 |
| Changes to Table 16 and Table 17.....  | 25 |
| Changes to Table 18 .....  | 26 |
| Change to Table 21 and changes to Table 22 .....                             | 27 |
| Changes to Table 24 .....  | 29 |
| Changes to ADCGN Register and ADCOF Register Sections .....                  | 32 |
| Changes to Temperature Sensor Section .....                                  | 34 |
| Changes to Table 29 .....  | 35 |
| Change to REMAP Register and RSTCLR Register Sections .....                  | 41 |
| Change to RSTKEY1 Register and RSTKEY2 Register Sections.....                | 42 |
| Changes to Oscillator and PLL—Power Control Section.....                     | 48 |
| Changes to General-Purpose Input/Output Section.....                         | 51 |
| Changes to Serial Peripheral Interface Section .....                         | 53 |
| Changes to Table 75 .....  | 67 |
| Changes to Table 83 and Pulse-Width Modulator General Overview Section ..... | 70 |
| Changes to Table 84 .....  | 71 |
| Change to Table 85.....  | 72 |
| Change to FIQSTAN Register Section .....                                     | 81 |
| Change to T2CLRI Register Section.....                                       | 85 |

**6/10—Rev. 0 to Rev. A**

Changes to Temperature Sensor Parameter in Table 1 ..... 6  
Changes to Table 24..... 29  
Changes to Temperature Sensor Section ..... 34  
Changes to DACBKEY0 Register Section and to Table 43 ..... 47  
Changes to Ordering Guide ..... 93

**1/10—Revision 0: Initial Version**

FUNCTIONAL BLOCK DIAGRAM

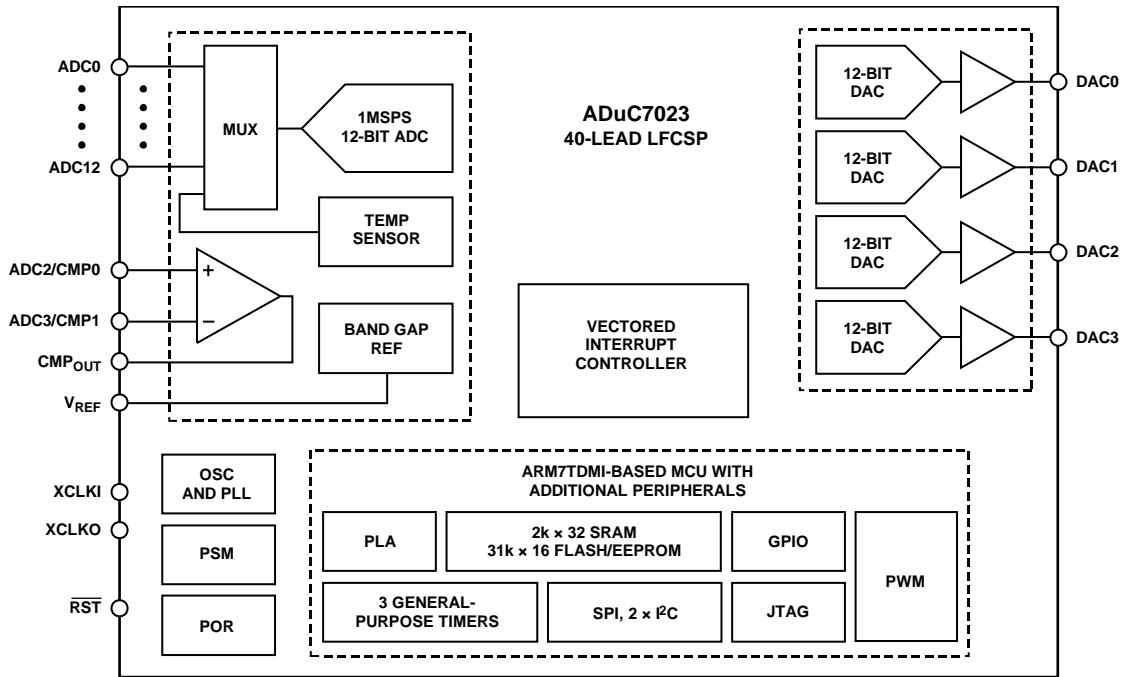


Figure 1.

08675-001

## SPECIFICATIONS

$AV_{DD} = IOV_{DD} = 2.7\text{ V to }3.6\text{ V}$ ,  $V_{REF} = 2.5\text{ V}$  internal reference,  $f_{CORE} = 41.78\text{ MHz}$ ,  $T_A = -40^\circ\text{C to }+125^\circ\text{C}$ , unless otherwise noted.

Table 1.

| Parameter                                 | Min   | Typ            | Max                      | Unit                  | Test Conditions/Comments  |
|---|-------|----------------|--------------------------|-----------------------|---|
| <b>ADC CHANNEL SPECIFICATIONS</b>         |       |                |                          |                       |   |
| ADC Power-Up Time                         |       | 5              |                          | $\mu\text{s}$         | Eight acquisition clocks and $f_{ADC}/2$<br><br>2.5 V internal reference<br>1.0 V external reference<br>2.5 V internal reference<br>1.0 V external reference<br>ADC input is a dc voltage |
| DC Accuracy <sup>1, 2</sup>               | 12    |                |                          | Bits                  |   |
| Resolution                                |       |                |                          | LSB                   |   |
| Integral Nonlinearity                     |       | $\pm 0.6$      | $\pm 1.5$                | LSB                   |   |
| Differential Nonlinearity <sup>3, 4</sup> |       | $\pm 0.5$      | $+1/-0.9$                | LSB                   |   |
|   |       | $+0.7/-0.6$    |                          | LSB                   |   |
| DC Code Distribution                      |       | 1              |                          | LSB                   |   |
| <b>ENDPOINT ERRORS<sup>5</sup></b>        |       |                |                          |                       |   |
| Offset Error                              |       | $\pm 1$        | $\pm 2$                  | LSB                   |   |
| Offset Error Match                        |       | $\pm 1$        |                          | LSB                   |   |
| Gain Error                                |       | $\pm 2$        |                          | LSB                   |   |
| Gain Error Match                          |       | $\pm 1$        |                          | LSB                   |   |
| <b>DYNAMIC PERFORMANCE</b>                |       |                |                          |                       |   |
| Signal-to-Noise Ratio (SNR)               |       | 69             |                          | dB                    | $f_{IN} = 10\text{ kHz}$ sine wave, $f_{SAMPLE} = 1\text{ MSPS}$<br>Includes distortion and noise components  |
| Total Harmonic Distortion (THD)           |       | -78            |                          | dB                    |   |
| Peak Harmonic or Spurious Noise           |       | -75            |                          | dB                    | Measured on adjacent channels   |
| Channel-to-Channel Crosstalk              |       | -80            |                          | dB                    |   |
| <b>ANALOG INPUT</b>                       |       |                |                          |                       |   |
| Input Voltage Ranges                      |       |                |                          |                       | During ADC acquisition  |
| Differential Mode                         |       |                | $V_{CM} \pm V_{REF}/2^6$ | V                     |   |
| Single-Ended Mode                         |       |                | 0 to $V_{REF}$           | V                     |   |
| Leakage Current                           |       | $\pm 1$        | $\pm 6$                  | $\mu\text{A}$         |   |
| Input Capacitance                         |       | 20             |                          | pF                    |   |
|   |       |                |                          |                       |   |
| <b>ON-CHIP VOLTAGE REFERENCE</b>          |       |                |                          |                       |   |
| Output Voltage                            |       | 2.5            |                          | V                     | 0.47 $\mu\text{F}$ from $V_{REF}$ to AGND<br><br>$T_A = 25^\circ\text{C}$   |
| Accuracy                                  |       |                | $\pm 4$                  | mV                    |   |
| Reference Temperature Coefficient         |       | $\pm 15$       |                          | ppm/ $^\circ\text{C}$ | $T_A = 25^\circ\text{C}$  |
| Power Supply Rejection Ratio              |       | 75             |                          | dB                    |   |
| Output Impedance                          |       | 51             |                          | $\Omega$              |   |
| Internal $V_{REF}$ Power-On Time          |       | 1              |                          | ms                    |   |
| <b>EXTERNAL REFERENCE INPUT</b>           |       |                |                          |                       |   |
| Input Voltage Range                       | 0.625 |                | $AV_{DD}$                | V                     |   |
| <b>DAC CHANNEL SPECIFICATIONS</b>         |       |                |                          |                       |   |
| DC Accuracy <sup>7</sup>                  |       |                |                          |                       | $R_L = 5\text{ k}\Omega$ , $C_L = 100\text{ pF}$  |
| Resolution                                |       | 12             |                          | Bits                  |   |
| Relative Accuracy                         |       | $\pm 2$        |                          | LSB                   | Guaranteed monotonic<br>2.5 V internal reference  |
| Differential Nonlinearity                 |       |                | $\pm 1$                  | LSB                   |   |
| Offset Error                              |       |                | $\pm 15$                 | mV                    |   |
| Gain Error <sup>8</sup>                   |       |                | $\pm 1$                  | %                     |   |
| Gain Error Mismatch                       |       | 0.1            |                          | %                     |   |
| DC Accuracy <sup>9</sup>                  |       |                |                          |                       | $R_L = 1\text{ k}\Omega$ , $C_L = 100\text{ pF}$  |
| Resolution                                |       | 12             |                          | Bits                  |   |
| Relative Accuracy                         |       | $\pm 2.5$      |                          | LSB                   | Guaranteed monotonic<br>2.5 V internal reference  |
| Differential Nonlinearity                 |       | $\pm 1$        |                          | LSB                   |   |
| Offset Error                              |       | $\pm 15$       |                          | mV                    |   |
| Gain Error <sup>10</sup>                  |       | $\pm 1$        |                          | %                     |   |
| Gain Error Mismatch                       |       | 0.1            |                          | %                     |   |
| <b>ANALOG OUTPUTS</b>                     |       |                |                          |                       |   |
| Output Voltage Range 1                    |       | 0 to 2.5       |                          | V                     | $V_{REF}$ range: AGND to $AV_{DD}$  |
| Output Voltage Range 2                    |       | 0 to $AV_{DD}$ |                          | V                     |   |
| Output Impedance                          |       | 2              |                          | $\Omega$              |   |

| Parameter   | Min    | Typ   | Max                    | Unit   | Test Conditions/Comments   |
|---|--------|-------|------------------------|--------|--|
| <b>DAC IN OP AMP MODE</b>   |        |       |                        |        |  |
| DAC Output Buffer in Op Amp Mode  |        |       |                        |        |  |
| Input Offset Voltage  |        | ±0.25 |                        | mV     |  |
| Input Offset Voltage Drift  |        | 8     |                        | μV/°C  |  |
| Input Offset Current  |        | 0.3   |                        | nA     |  |
| Input Bias Current  |        | 0.4   |                        | nA     |  |
| Gain  |        | 80    |                        | dB     | 5 kΩ load  |
| Unity-Gain Frequency  |        | 5     |                        | MHz    | R <sub>L</sub> = 5 kΩ, C <sub>L</sub> = 100 pF   |
| CMRR  |        | 80    |                        | dB     |  |
| Settling Time   |        | 10    |                        | μs     | R <sub>L</sub> = 5 kΩ, C <sub>L</sub> = 100 pF   |
| Output Slew Rate  |        | 1.5   |                        | V/μs   | R <sub>L</sub> = 5 kΩ, C <sub>L</sub> = 100 pF   |
| PSRR  |        | 75    |                        | dB     |  |
| <b>DAC AC CHARACTERISTICS</b>   |        |       |                        |        |  |
| Voltage Output Settling Time  |        | 10    |                        | μs     |  |
| Digital-to-Analog Glitch Energy   |        | ±20   |                        | nV-sec | 1 LSB change at major carry (where maximum number of bits simultaneously change in the DACxDAT register)   |
| <b>COMPARATOR</b>   |        |       |                        |        |  |
| Input Offset Voltage  |        | ±10   |                        | mV     |  |
| Input Bias Current  |        | 1     |                        | μA     |  |
| Input Voltage Range   | AGND   |       | AV <sub>DD</sub> - 1.2 | V      |  |
| Input Capacitance   |        | 7     |                        | pF     |  |
| Hysteresis <sup>4,6</sup>   | 2      |       | 15                     | mV     | Hysteresis can be turned on or off via the CMPHYST bit in the CMPCON register                              |
| Response Time   |        | 3     |                        | μs     | 100 mV overdrive and configured with CMPRES = 11   |
| <b>TEMPERATURE SENSOR</b>   |        |       |                        |        |  |
| Voltage Output at 25°C  |        | 1.369 |                        | V      | Indicates die temperature  |
| Voltage TC  |        | 4.42  |                        | mV/°C  |  |
| Accuracy with No Calibration  |        | ±3    |                        | °C     |  |
| Accuracy with One Point Calibration<br>Using Contents of TEMPREF Register |        | ±1.5  |                        | °C     |  |
| θ <sub>JA</sub> Thermal Impedance   |        |       |                        |        |  |
| 40-Lead LFCSP   |        | 26    |                        | °C/W   |  |
| 32-Lead LFCSP   |        | 32.5  |                        | °C/W   |  |
| <b>POWER SUPPLY MONITOR (PSM)</b>   |        |       |                        |        |  |
| IOV <sub>DD</sub> Trip Point Selection                                    |        | 2.79  |                        | V      | One trip point   |
| Power Supply Trip Point Accuracy  |        | ±2    |                        | %      | Of the selected nominal trip point voltage   |
| <b>POWER-ON RESET</b>   |        |       |                        |        |  |
|   |        | 2.41  |                        | V      |  |
| <b>WATCHDOG TIMER (WDT)</b>   |        |       |                        |        |  |
| Timeout Period  | 0      |       | 512                    | sec    |  |
| <b>FLASH/EE MEMORY</b>  |        |       |                        |        |  |
| Endurance <sup>11</sup>   | 10,000 |       |                        | Cycles |  |
| Data Retention <sup>12</sup>  | 20     |       |                        | Years  | T <sub>J</sub> = 85°C  |
| <b>DIGITAL INPUTS</b>   |        |       |                        |        |  |
| Logic 1 Input Current   |        | ±0.2  | ±1                     | μA     | All digital inputs excluding XCLKI and XCLKO<br>V <sub>IH</sub> = V <sub>DD</sub> or V <sub>IH</sub> = 5 V |
| Logic 0 Input Current   |        | -40   | -60                    | μA     | V <sub>IL</sub> = 0 V; except TDI  |
|   |        | -80   | -120                   | μA     | V <sub>IL</sub> = 0 V; TDI   |
| Input Capacitance   |        | 10    |                        | pF     |  |
| <b>LOGIC INPUTS<sup>4</sup></b>   |        |       |                        |        |  |
| V <sub>INL</sub> , Input Low Voltage                                      |        |       | 0.8                    | V      | All logic inputs excluding XCLKI   |
| V <sub>INH</sub> , Input High Voltage                                     | 2.0    |       |                        | V      |  |
| <b>LOGIC OUTPUTS</b>  |        |       |                        |        |  |
| V <sub>OH</sub> , Output High Voltage                                     | 2.4    |       |                        | V      | All digital outputs excluding XCLKO<br>I <sub>SOURCE</sub> = 1.6 mA  |
| V <sub>OL</sub> , Output Low Voltage <sup>13</sup>                        |        |       | 0.4                    | V      | I <sub>SINK</sub> = 1.6 mA   |
| <b>CRYSTAL INPUTS XCLKI AND XCLKO</b>                                     |        |       |                        |        |  |
| Logic Inputs, XCLKI Only  |        |       |                        |        |  |
| V <sub>INL</sub> , Input Low Voltage                                      |        | 1.1   |                        | V      |  |
| V <sub>INH</sub> , Input High Voltage                                     |        | 1.7   |                        | V      |  |
| XCLKI Input Capacitance   |        | 20    |                        | pF     |  |
| XCLKO Output Capacitance  |        | 20    |                        | pF     |  |

| Parameter  | Min  | Typ    | Max   | Unit     | Test Conditions/Comments               |
|--|------|--------|-------|----------|--|
| INTERNAL OSCILLATOR                                    |      | 32.768 | ±3    | kHz<br>% |  |
| MCU CLOCK RATE   |      | 326    |       | kHz      | CD = 7                                 |
| From 32 kHz Internal Oscillator                        |      | 41.78  |       | MHz      | CD = 0                                 |
| From 32 kHz External Crystal                           | 0.05 |        | 44    | MHz      | T <sub>A</sub> = 85°C                  |
| Using an External Clock                                | 0.05 |        | 41.78 | MHz      | T <sub>A</sub> = 125°C                 |
| START-UP TIME  |      |        |       |          | Core clock = 41.78 MHz                 |
| At Power-On  |      | 66     |       | ms       |  |
| From Pause/Nap Mode                                    |      | 24     |       | ns       | CD = 0                                 |
|  |      | 3.07   |       | µs       | CD = 7                                 |
| From Sleep Mode  |      | 1.58   |       | ms       |  |
| From Stop Mode   |      | 1.7    |       | ms       |  |
| PROGRAMMABLE LOGIC ARRAY (PLA)                         |      |        |       |          |  |
| Pin Propagation Delay                                  |      | 12     |       | ns       | From input pin to output pin           |
| Element Propagation Delay                              |      | 2.5    |       | ns       |  |
| POWER REQUIREMENTS <sup>14, 15</sup>                   |      |        |       |          |  |
| Power Supply Voltage Range                             | 2.7  |        | 3.6   | V        |  |
| AV <sub>DD</sub> to AGND and IOV <sub>DD</sub> to DGND |      |        |       |          |  |
| Analog Power Supply Currents                           |      | 200    |       | µA       | ADC in idle mode                       |
| AV <sub>DD</sub> Current                               |      |        |       |          |  |
| Digital Power Supply Current                           |      | 8.5    | 10    | mA       | Code executing from Flash/EE           |
| IOV <sub>DD</sub> Current in Normal Mode               |      | 11     | 15    | mA       | CD = 7                                 |
|  |      | 28     | 35    | mA       | CD = 3                                 |
|  |      | 14     | 20    | mA       | CD = 0 (41.78 MHz clock)               |
| IOV <sub>DD</sub> Current in Pause Mode                |      | 14     | 20    | mA       | CD = 0 (41.78 MHz clock)               |
| IOV <sub>DD</sub> Current in Sleep Mode                |      | 230    | 650   | µA       | T <sub>A</sub> = 125°C                 |
| Additional Power Supply Currents                       |      |        |       |          |  |
| ADC  |      | 1.4    |       | mA       | At 1 MSPS                              |
|  |      | 0.7    |       | mA       | At 62.5 kSPS                           |
| DAC  |      | 400    |       | µA       | Per DAC                                |
| ESD TESTS  |      |        |       |          | 2.5 V reference, T <sub>A</sub> = 25°C |
| HBM Passed   |      |        | 3     | kV       |  |
| FICDM Passed   |      |        | 1.0   | kV       |  |

<sup>1</sup> All ADC channel specifications are guaranteed during normal microcontroller core operation.

<sup>2</sup> Apply to all ADC input channels.

<sup>3</sup> Measured using the factory-set default values in the ADC offset register (ADCOF) and gain coefficient register (ADCGN).

<sup>4</sup> Not production tested but supported by design and/or characterization data on production release.

<sup>5</sup> Measured using the factory-set default values in ADCOF and ADCGN with an external AD845 op amp as an input buffer stage as shown in Figure 28. Based on external ADC system components, the user may need to execute a system calibration to remove external endpoint errors and achieve these specifications (see the Calibration section).

<sup>6</sup> The input signal can be centered on any dc common-mode voltage (V<sub>CM</sub>) as long as this value is within the ADC voltage input range specified.

<sup>7</sup> DAC linearity is calculated using a reduced code range of 100 to 3995.

<sup>8</sup> DAC gain error is calculated using a reduced code range of 100 to internal 2.5 V V<sub>REF</sub>.

<sup>9</sup> DAC linearity is calculated using a reduced code range of 100 to 3995.

<sup>10</sup> DAC gain error is calculated using a reduced code range of 100 to internal 2.5 V V<sub>REF</sub>.

<sup>11</sup> Endurance is qualified as per JEDEC Standard 22 Method A117 and measured at -40°C, +25°C, +85°C, and +125°C.

<sup>12</sup> Retention lifetime equivalent at junction temperature (T<sub>J</sub>) = 85°C as per JEDEC Standard 22 Method A117. Retention lifetime derates with junction temperature.

<sup>13</sup> Test carried out with a maximum of eight I/Os set to a low output level.

<sup>14</sup> Power supply current consumption is measured in normal, pause, and sleep modes under the following conditions: normal mode with 3.6 V supply, pause mode with 3.6 V supply, and sleep mode with 3.6 V supply.

<sup>15</sup> IOV<sub>DD</sub> power supply current decreases typically by 2 mA during a Flash/EE erase cycle.



**TIMING SPECIFICATIONS**

**Table 2. I<sup>2</sup>C Timing in Fast Mode (400 kHz)**

| Parameter        | Description  | Slave |     | Master | Unit |
|------------------|--|-------|-----|--------|------|
|                  |  | Min   | Max | Typ    |      |
| t <sub>L</sub>   | SCL low pulse width  | 200   |     | 1360   | ns   |
| t <sub>H</sub>   | SCL high pulse width   | 100   |     | 1140   | ns   |
| t <sub>SHD</sub> | Start condition hold time                                    | 300   |     |        | ns   |
| t <sub>DSU</sub> | Data setup time  | 100   |     | 740    | ns   |
| t <sub>DHD</sub> | Data hold time   | 0     |     | 400    | ns   |
| t <sub>RSU</sub> | Setup time for repeated start                                | 100   |     |        | ns   |
| t <sub>PSU</sub> | Stop condition setup time                                    | 100   |     | 800    | ns   |
| t <sub>BUF</sub> | Bus-free time between a stop condition and a start condition | 1.3   |     |        | μs   |
| t <sub>R</sub>   | Rise time for both SCL and SDA                               |       | 300 | 200    | ns   |
| t <sub>F</sub>   | Fall time for both SCL and SDA                               |       | 300 |        | ns   |

**Table 3. I<sup>2</sup>C Timing in Standard Mode (100 kHz)**

| Parameter        | Description  | Slave |      | Unit |
|------------------|--|-------|------|------|
|                  |  | Min   | Max  |      |
| t <sub>L</sub>   | SCL low pulse width  | 4.7   |      | μs   |
| t <sub>H</sub>   | SCL 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 SCL and SDA                               |       | 1    | μs   |
| t <sub>F</sub>   | Fall time for both SCL and SDA                               |       | 300  | ns   |

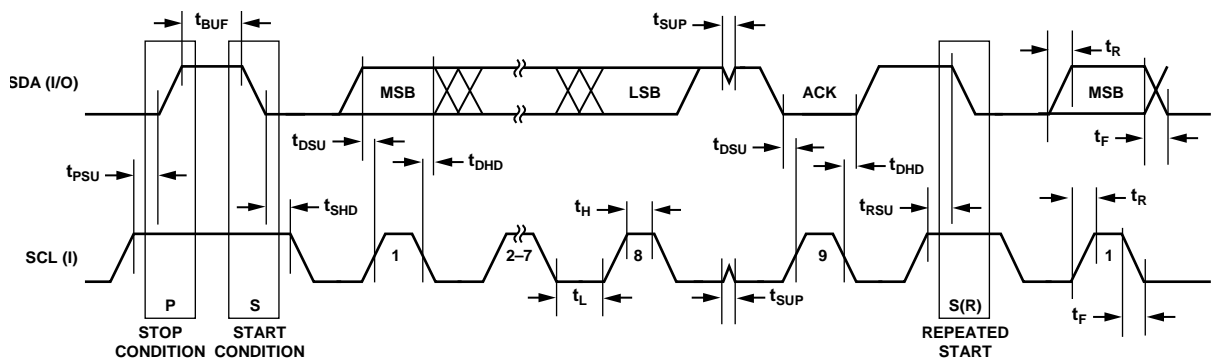


Figure 2. I<sup>2</sup>C-Compatible Interface Timing

Table 4. SPI Master Mode Timing (Phase Mode = 1)

| Parameter | Description   | Min                 | Typ                            | Max  | Unit |
|-----------|---|---------------------|--------------------------------|------|------|
| $t_{SL}$  | SCLK low pulse width <sup>1</sup>                   |                     | $(SPIDIV + 1) \times t_{UCLK}$ |      | ns   |
| $t_{SH}$  | SCLK high pulse width <sup>1</sup>                  |                     | $(SPIDIV + 1) \times t_{UCLK}$ |      | ns   |
| $t_{DAV}$ | Data output valid after SCLK edge                   |                     |                                | 25   | ns   |
| $t_{DSU}$ | Data input setup time before SCLK edge <sup>1</sup> | $1 \times t_{UCLK}$ |                                |      | ns   |
| $t_{DHD}$ | Data input hold time after SCLK edge <sup>1</sup>   | $2 \times t_{UCLK}$ |                                |      | ns   |
| $t_{DF}$  | Data output fall time                               |                     | 5                              | 12.5 | ns   |
| $t_{DR}$  | Data output rise time                               |                     | 5                              | 12.5 | ns   |
| $t_{SR}$  | SCLK rise time                                      |                     | 5                              | 12.5 | ns   |
| $t_{SF}$  | SCLK fall time                                      |                     | 5                              | 12.5 | ns   |

<sup>1</sup>  $t_{UCLK} = 23.9$  ns. It corresponds to the 41.78 MHz internal clock from the PLL before the clock divider.

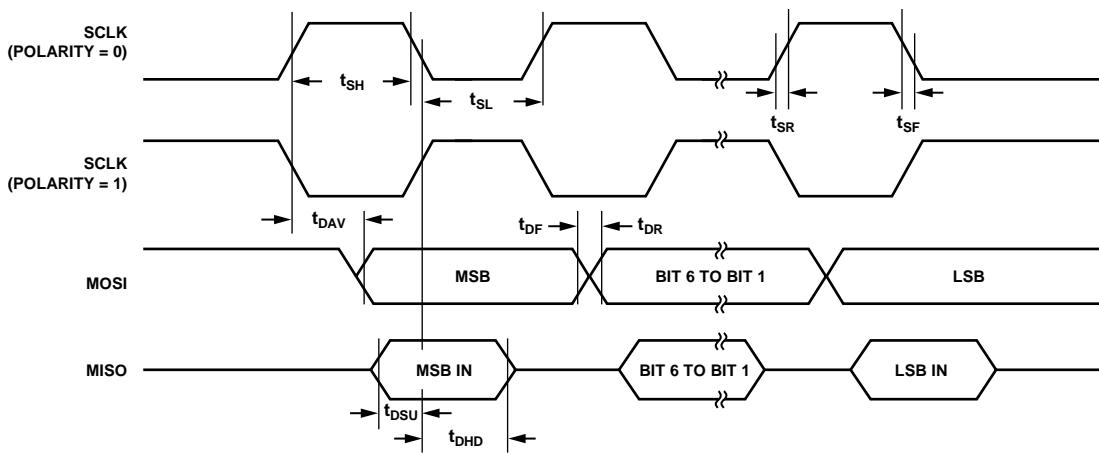


Figure 3. SPI Master Mode Timing (Phase Mode = 1)

081675-003

Table 5. SPI Master Mode Timing (Phase Mode = 0)

| Parameter  | Description   | Min                 | Typ                            | Max  | Unit |
|------------|---|---------------------|--------------------------------|------|------|
| $t_{SL}$   | SCLK low pulse width <sup>1</sup>                   |                     | $(SPIDIV + 1) \times t_{UCLK}$ |      | ns   |
| $t_{SH}$   | SCLK high pulse width <sup>1</sup>                  |                     | $(SPIDIV + 1) \times t_{UCLK}$ |      | ns   |
| $t_{DAV}$  | Data output valid after SCLK edge                   |                     |                                | 25   | ns   |
| $t_{DOSU}$ | Data output setup before SCLK edge                  |                     |                                | 75   | ns   |
| $t_{DSU}$  | Data input setup time before SCLK edge <sup>1</sup> | $1 \times t_{UCLK}$ |                                |      | ns   |
| $t_{DHD}$  | Data input hold time after SCLK edge <sup>1</sup>   | $2 \times t_{UCLK}$ |                                |      | ns   |
| $t_{DF}$   | Data output fall time                               |                     | 5                              | 12.5 | ns   |
| $t_{DR}$   | Data output rise time                               |                     | 5                              | 12.5 | ns   |
| $t_{SR}$   | SCLK rise time                                      |                     | 5                              | 12.5 | ns   |
| $t_{SF}$   | SCLK fall time                                      |                     | 5                              | 12.5 | ns   |

<sup>1</sup>  $t_{UCLK} = 23.9$  ns. It corresponds to the 41.78 MHz internal clock from the PLL before the clock divider.

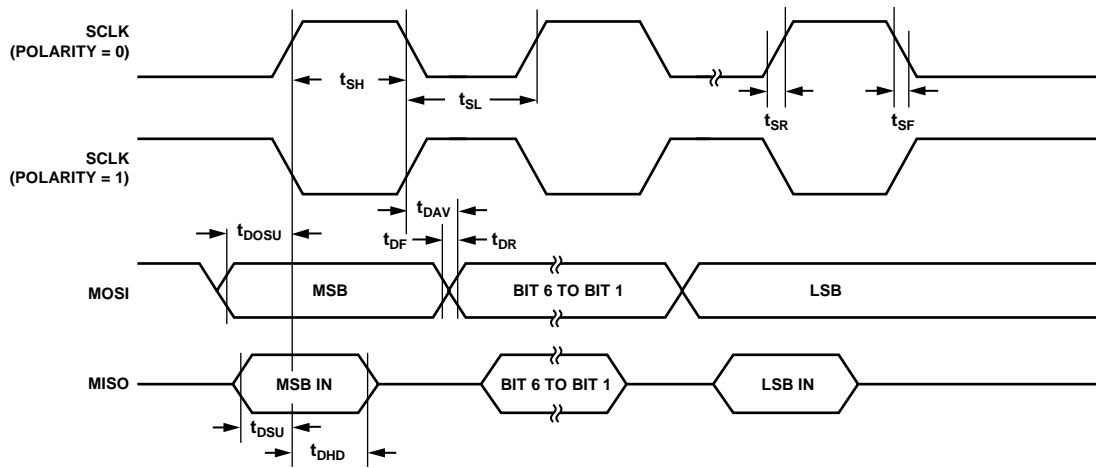


Figure 4. SPI Master Mode Timing (Phase Mode = 0)

08675-004

Table 6. SPI Slave Mode Timing (Phase Mode = 1)

| Parameter           | Description   | Min                 | Typ                            | Max  | Unit |
|---------------------|---|---------------------|--------------------------------|------|------|
| $t_{\overline{SS}}$ | $\overline{SS}$ to SCLK edge                        | 200                 |                                |      | ns   |
| $t_{SL}$            | SCLK low pulse width <sup>1</sup>                   |                     | $(SPIDIV + 1) \times t_{UCLK}$ |      | ns   |
| $t_{SH}$            | SCLK high pulse width <sup>1</sup>                  |                     | $(SPIDIV + 1) \times t_{UCLK}$ |      | ns   |
| $t_{DAV}$           | Data output valid after SCLK edge                   |                     |                                | 25   | ns   |
| $t_{DSU}$           | Data input setup time before SCLK edge <sup>1</sup> | $1 \times t_{UCLK}$ |                                |      | ns   |
| $t_{DHD}$           | Data input hold time after SCLK edge <sup>1</sup>   | $2 \times t_{UCLK}$ |                                |      | ns   |
| $t_{DF}$            | Data output fall time                               |                     | 5                              | 12.5 | ns   |
| $t_{DR}$            | Data output rise time                               |                     | 5                              | 12.5 | ns   |
| $t_{SR}$            | SCLK rise time                                      |                     | 5                              | 12.5 | ns   |
| $t_{SF}$            | SCLK fall time                                      |                     | 5                              | 12.5 | ns   |
| $t_{SFS}$           | $\overline{SS}$ high after SCLK edge                | 0                   |                                |      | ns   |

<sup>1</sup>  $t_{UCLK} = 23.9$  ns. It corresponds to the 41.78 MHz internal clock from the PLL before the clock divider.

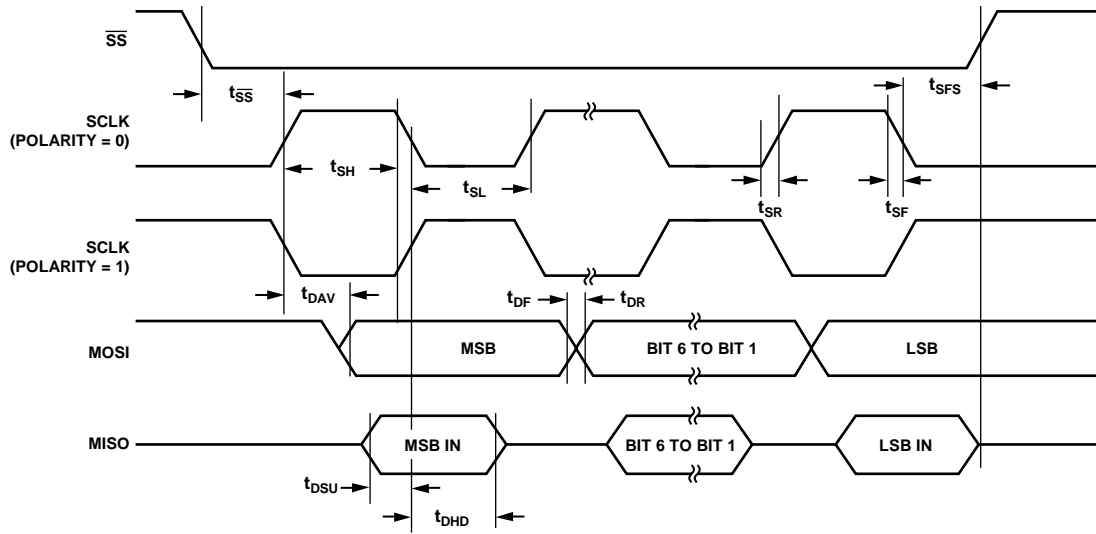


Figure 5. SPI Slave Mode Timing (Phase Mode = 1)

08675-005

Table 7. SPI Slave Mode Timing (Phase Mode = 0)

| Parameter           | Description   | Min                 | Typ                            | Max  | Unit |
|---------------------|---|---------------------|--------------------------------|------|------|
| $t_{\overline{SS}}$ | $\overline{SS}$ to SCLK edge                        | 200                 |                                |      | ns   |
| $t_{SL}$            | SCLK low pulse width <sup>1</sup>                   |                     | $(SPIDIV + 1) \times t_{UCLK}$ |      | ns   |
| $t_{SH}$            | SCLK high pulse width <sup>1</sup>                  |                     | $(SPIDIV + 1) \times t_{UCLK}$ |      | ns   |
| $t_{DAV}$           | Data output valid after SCLK edge                   |                     |                                | 25   | ns   |
| $t_{DSU}$           | Data input setup time before SCLK edge <sup>1</sup> | $1 \times t_{UCLK}$ |                                |      | ns   |
| $t_{DHD}$           | Data input hold time after SCLK edge <sup>1</sup>   | $2 \times t_{UCLK}$ |                                |      | ns   |
| $t_{DF}$            | Data output fall time                               |                     | 5                              | 12.5 | ns   |
| $t_{DR}$            | Data output rise time                               |                     | 5                              | 12.5 | ns   |
| $t_{SR}$            | SCLK rise time                                      |                     | 5                              | 12.5 | ns   |
| $t_{SF}$            | SCLK fall time                                      |                     | 5                              | 12.5 | ns   |
| $t_{DOCS}$          | Data output valid after $\overline{SS}$ edge        |                     |                                | 25   | ns   |
| $t_{SFS}$           | $\overline{SS}$ high after SCLK edge                | 0                   |                                |      | ns   |

<sup>1</sup>  $t_{UCLK} = 23.9$  ns. It corresponds to the 41.78 MHz internal clock from the PLL before the clock divider.

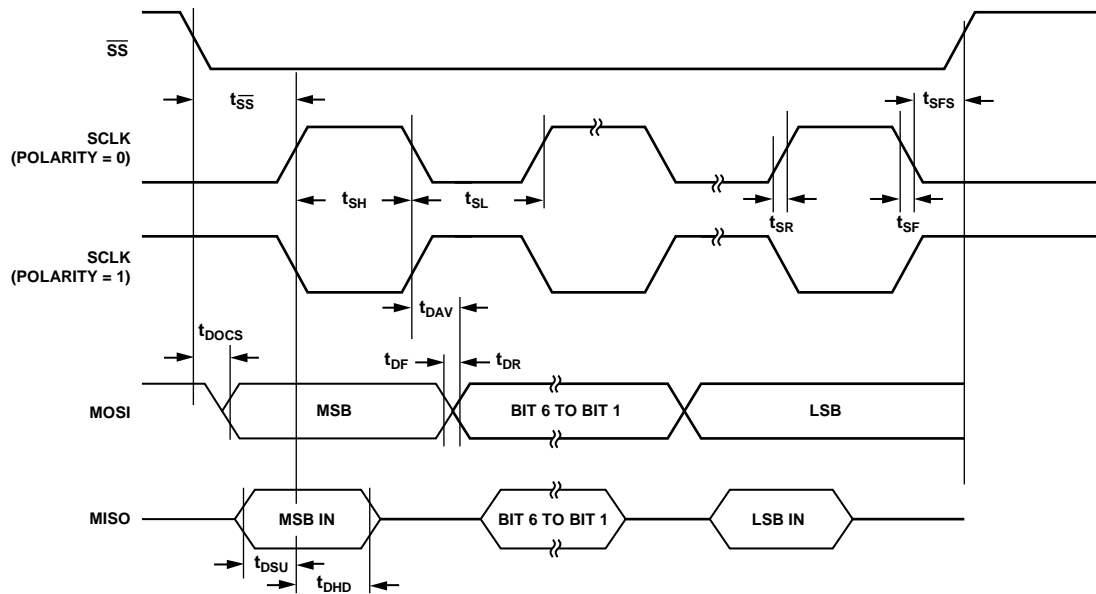


Figure 6. SPI Slave Mode Timing (Phase Mode = 0)

08675-006

## ABSOLUTE MAXIMUM RATINGS

AGND = GND<sub>REF</sub>, T<sub>A</sub> = 25°C, unless otherwise noted.

Table 8.

| Parameter   | Rating                              |
|---|-------------------------------------|
| AV <sub>DD</sub> to IOV <sub>DD</sub>               | −0.3 V to +0.3 V                    |
| AGND to DGND  | −0.3 V to +0.3 V                    |
| IOV <sub>DD</sub> to DGND, AV <sub>DD</sub> to AGND | −0.3 V to +6 V                      |
| Digital Input Voltage to DGND <sup>1</sup>          | −0.3 V to +5.3 V                    |
| Digital Output Voltage to DGND <sup>1</sup>         | −0.3 V to IOV <sub>DD</sub> + 0.3 V |
| Shared Analog/Digital Inputs to AGND <sup>2</sup>   | −0.3 V to AV <sub>DD</sub> + 0.3 V  |
| V <sub>REF</sub> to AGND                            | −0.3 V to AV <sub>DD</sub> + 0.3 V  |
| Analog Inputs to AGND                               | −0.3 V to AV <sub>DD</sub> + 0.3 V  |
| Analog Outputs to AGND                              | −0.3 V to AV <sub>DD</sub> + 0.3 V  |
| Operating Temperature Range, Industrial             | −40°C to +125°C                     |
| Storage Temperature Range                           | −65°C to +150°C                     |
| Junction Temperature                                | 150°C                               |
| θ <sub>JA</sub> Thermal Impedance                   |                                     |
| 40-Lead LFCSP                                       | 26°C/W                              |
| 32-Lead LFCSP                                       | 32.5°C/W                            |
| 36-Lead WLCSP                                       | 50°C/W                              |
| Peak Solder Reflow Temperature                      |                                     |
| SnPb Assemblies (10 sec to 30 sec)                  | 240°C                               |
| RoHS Compliant Assemblies<br>(20 sec to 40 sec)     | 260°C                               |

<sup>1</sup> These limits apply to the P0.0, P0.1, P0.2, P0.3, P0.4, P0.5, P0.6, P0.7, P1.0, P1.1, P1.6, and P1.7 pins.

<sup>2</sup> These limits apply to the P1.2, P1.3, P1.4, P1.5, P2.0, P2.2, P2.3, and P2.4 pins.

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.

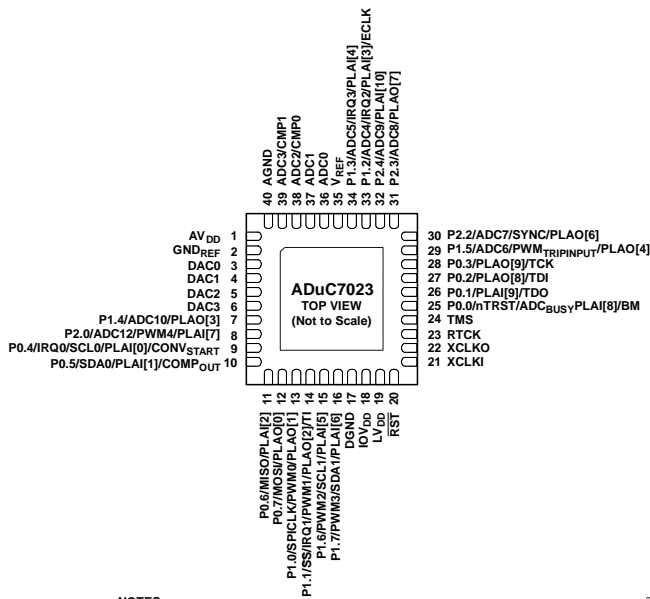
Only one absolute maximum rating can be applied at any one time.

### ESD CAUTION



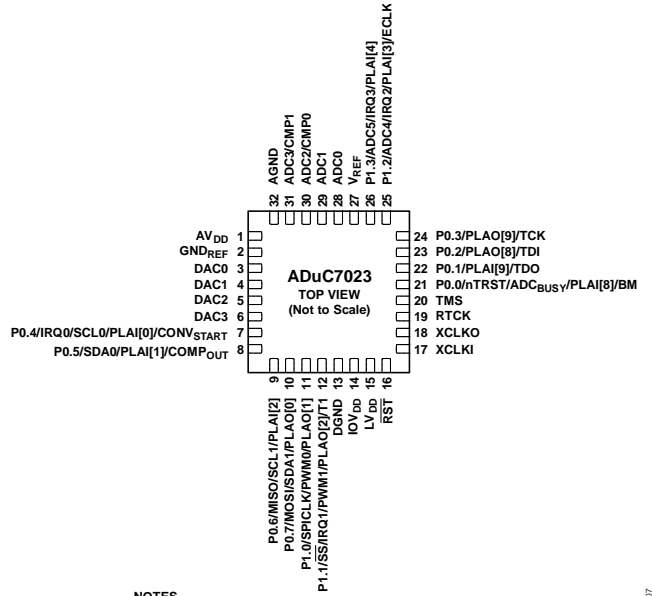
**ESD (electrostatic discharge) sensitive device.** Charged devices and circuit boards can discharge without detection. Although this product features patented or proprietary protection circuitry, damage may occur on devices subjected to high energy ESD. Therefore, proper ESD precautions should be taken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

# PIN CONFIGURATIONS AND FUNCTION DESCRIPTIONS



NOTES  
1. EXPOSED PAD. THE PADDLE NEEDS TO BE SOLDERED AND EITHER CONNECTED TO AGND OR LEFT FLOATING.

Figure 7. 40-Lead LFCSP Pin Configuration



NOTES  
1. EXPOSED PAD. THE PADDLE NEEDS TO BE SOLDERED AND EITHER CONNECTED TO AGND OR LEFT FLOATING.

Figure 8. 32-Lead LFCSP Pin Configuration

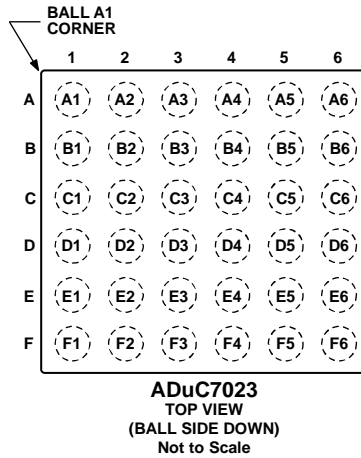


Figure 9. 36-Lead WLCSP Pin Configuration

Table 9. Pin Function Descriptions

| Pin No.  |          |          | Mnemonic           | Description  |
|----------|----------|----------|--------------------|--|
| 40-LFCSP | 32-LFCSP | 36-WLCSP |                    |  |
| 0        | 0        | N/A      | Exposed Paddle     | Exposed Pad. The paddle needs to be soldered and either connected to AGND or left floating.  |
| 36       | 28       | A4       | ADC0               | Single-Ended or Differential Analog Input 0.   |
| 37       | 29       | B4       | ADC1               | Single-Ended or Differential Analog Input 1.   |
| 38       | 30       | A5       | ADC2/CMP0          | Single-Ended or Differential Analog Input 2/Comparator Positive Input.   |
| 39       | 31       | B5       | ADC3/CMP1          | Single-Ended or Differential Analog Input 3/Comparator Negative Input.   |
| 32       | N/A      | B2       | P2.4/ADC9/PLAI[10] | General-Purpose Input and Output Port 2.4/ADC Single-Ended or Differential Analog Input/Programmable Logic Array Input Element 10. By default, this pin is configured as a digital input with a weak pull-up resistor enabled. |

| Pin No.  |          |          | Mnemonic                                   | Description   |
|----------|----------|----------|--|---|
| 40-LFCSP | 32-LFCSP | 36-WLCSP |  |   |
| 31       | N/A      | A1       | P2.3/ADC8/PLAO[7]                          | General-Purpose Input and Output Port 2.3/ADC Single-Ended or Differential Analog Input 8/Programmable Logic Array Output Element 7. By default, this pin is configured as a digital input with a weak pull-up resistor enabled. When used as ADC input, pull-up resistor should be disabled manually.  |
| 30       | N/A      | B1       | P2.2/ADC7/SYNC/PLAO[6]                     | General-Purpose Input and Output Port 2.2/ADC Single-Ended or Differential Analog Input 7/PWM Sync/Programmable Logic Array Output Element 6. By default, this pin is configured as a digital input with a weak pull-up resistor enabled. When used as ADC input, pull-up resistor should be disabled manually.   |
| 8        | N/A      | E6       | P2.0/ADC12/PWM4/PLAI[7]                    | General-Purpose Input and Output Port 2.0/ADC Single-Ended or Differential Analog Input 12/PWM Output 4/Programmable Logic Array Input Element 7. By default, this pin is configured as a digital input with a weak pull-up resistor enabled. When used as an ADC input, it is not possible to disable the internal pull-up resistor. This means that this pin has a higher leakage current value than other analog input pins.   |
| 2        | 2        | C4       | GND <sub>REF</sub>                         | Ground Voltage Reference for the ADC. For optimal performance, the analog power supply should be separated from DGND.   |
| 3        | 3        | C5       | DAC0                                       | DAC0 Voltage Output or ADC Input.   |
| 4        | 4        | C6       | DAC1                                       | DAC1 Voltage Output or ADC Input.   |
| 5        | 5        | D5       | DAC2                                       | DAC2 Voltage Output   |
| 6        | 6        | D6       | DAC3                                       | DAC3 Voltage Output   |
| 24       | 20       | D2       | TMS  | Test Mode Select, JTAG Test Port Input. Debug and download access. This pin has an internal pull-up resistor to IOV <sub>DD</sub> . In some cases an external pull-up resistor is also required to ensure the part does not enter an erroneous state.   |
| 25       | 21       | D1       | P0.0/nTRST/ADC <sub>BUSY</sub> /PLAI[8]/BM | This is a multifunction pin as follows:<br>General-Purpose Input and Output Port 0.0. By default, this pin is configured as GPIO.<br>JTAG Reset Input. Debug and download access. If this pin is held low, JTAG access is not possible because the JTAG interface is held in reset and P0.1/P0.2/P0.3 are configured as GPIO pins.<br>ADC Busy Signal.<br>Programmable Logic Array Input Element 8.<br>Boot Mode Entry Pin. The ADuC7023 enters I <sup>2</sup> C download mode if BM is low at reset with a flash address 0x80014 = 0xFFFFFFFF. The ADuC7023 executes code if BM is pulled high at reset or if BM is low at reset with a flash address 0x80014 not equal to 0xFFFFFFFF. |
| 26       | 22       | C1       | P0.1/PLAI[9]/TDO                           | The default value of this pin depends on the level of P0.0/BM. If P0.0/BM = 0, this pin defaults to a general purpose input. If P0.0/BM = 1, this pin defaults to a JTAG test data output pin and does not work as a GPIO.<br>This is a multifunction pin as follows:<br>General-Purpose Input and Output Port 0.1.<br>Programmable Logic Array Input Element 9.<br>Test Data Out, JTAG Test Port Output. Debug and download access. When debugging the part via JTAG, this pin must not be toggled by user code, and the GPOCON/GPODAT register bits affecting this pin must not be changed as doing so disables JTAG access.  |
| 27       | 23       | C2       | P0.2/PLAO[8]/TDI                           | The default value of this pin depends on the level of P0.0/BM. If P0.0/BM = 0, this pin defaults to a general purpose input. If P0.0/BM = 1, this pin defaults to a JTAG test data input pin and does not work as a GPIO. This is a multifunction pin as follows:<br>General-Purpose Input and Output Port 0.2.<br>Programmable Logic Array Output Element 8.<br>Test Data In, JTAG Test Port Input. Debug and download access. When debugging the part via JTAG, this pin must not be toggled by user code, and the GPOCON/GPODAT register bits affecting this pin must not be changed as doing so disables JTAG access.   |



| Pin No.  |          |          | Mnemonic                                    | Description  |
|----------|----------|----------|---|--|
| 40-LFCSP | 32-LFCSP | 36-WLCSP |   |  |
| 28       | 24       | C3       | P0.3/PLAO[9]/TCK                            | The default value of this pin depends on the level of P0.0/BM. If P0.0/BM = 0, this pin defaults to a general purpose input. If P0.0/BM = 1, this pin defaults to a JTAG test data clock pin. This is a multifunction pin as follows:<br>General-Purpose Input and Output Port 0.3.<br>Programmable Logic Array Output Element 9.<br>Test Clock, JTAG Test Port Clock Input. Debug and download access. When debugging the part via JTAG, this pin must not be toggled by user code and the GP0CON/GP0DAT register bits affecting this pin must not be changed as doing so disables JTAG access. |
| 17       | 13       | E3       | DGND  | Digital Ground.  |
| 18       | 14       | F3       | IOV <sub>DD</sub>                           | 3.3 V Supply for GPIO and Input of the On-Chip Voltage Regulator.  |
| 19       | 15       | D3       | LV <sub>DD</sub>                            | 2.6 V Output of the On-Chip Voltage Regulator. This output must be connected to a 0.47 $\mu$ F capacitor to DGND only.   |
| 20       | 16       | F2       | $\overline{\text{RST}}$                     | Reset Input, Active Low.   |
| 23       | 19       | E1       | RTCK  | Return JTAG Clock Signal. This is not the standard JTAG clock signal. It is an output signal from the JTAG controller. If using a 20-lead JTAG header, connect to Pin 11.  |
| 9        | 7        | F6       | P0.4/IRQ0/SCL0/PLAI[0]/CONV                 | General-Purpose Input and Output Port 0.4/External Interrupt Request 0/ I <sup>2</sup> C0 Clock Signal/Programmable Logic Array Input Element 0/ADC External Convert Start. By default, this pin is configured as a digital input with a weak pull-up resistor enabled.  |
| 10       | 8        | E5       | P0.5/SDA0/PLAI[1]/COMP <sub>OUT</sub>       | General-Purpose Input and Output Port 0.5/I <sup>2</sup> C0 Data Signal/ Programmable Logic Array Input Element 1/Voltage Comparator Output. By default, this pin is configured as a digital input with a weak pull-up resistor enabled.   |
|          | 9        | F5       | P0.6/MISO/SCL1/PLAI[2]                      | General-Purpose Input and Output Port 0.6/SPI MISO Signal/I <sup>2</sup> C1 Clock On 32-Lead and 36-Ball Packages/Programmable Logic Array Input Element 2. By default, this pin is configured as a digital input with a weak pull-up resistor enabled.  |
|          | 10       | D4       | P0.7/MOSI/SDA1/PLAO[0]                      | General-Purpose Input and Output Port 0.7/SPI MOSI Signal/I <sup>2</sup> C1 Data Signal On 32-Lead and 36-Ball Packages/Programmable Logic Array Output Element 0.<br>By default, this pin is configured as a digital input with a weak pull-up resistor enabled.  |
| 11       |          |          | P0.6/MISO/PLAI[2]                           | General-Purpose Input and Output Port 0.6/SPI MISO Signal/Programmable Logic Array Input Element 2. By default, this pin is configured as a digital input with a weak pull-up resistor enabled.  |
| 12       |          |          | P0.7/MOSI/PLAO[0]                           | General-Purpose Input and Output Port 0.7/SPI MOSI Signal/Programmable Logic Array Output Element 0. By default this pin is configured as a digital input with a weak pull-up resistor enabled.  |
| 21       | 17       | F1       | XCLKI                                       | Input to the Crystal Oscillator Inverter and Input to the Internal Clock Generator Circuits. Connect to DGND if unused.  |
| 22       | 18       | E2       | XCLKO                                       | Output from the Crystal Oscillator Inverter. Leave unconnected if unused.  |
| 16       | N/A      | N/A      | P1.7/PWM3/SDA1/PLAI[6]                      | General-Purpose Input and Output Port 1.7/PWM Output 3/I <sup>2</sup> C1 Data Signal/Programmable Logic Array Input Element 6. By default, this pin is configured as a digital input with a weak pull-up resistor enabled.   |
| 15       | N/A      | N/A      | P1.6/PWM2/SCL1/PLAI[5]                      | General-Purpose Input and Output Port 1.6/PWM Output 2/I <sup>2</sup> C1 Clock Signal/Programmable Logic Array Input Element 5. By default, this pin is configured as a digital input with a weak pull-up resistor enabled.  |
| 29       | N/A      | N/A      | P1.5/ADC6/PWM <sub>TRIPINPUT</sub> /PLAO[4] | General-Purpose Input and Output Port 1.5/ADC Single-Ended or Differential Analog Input 6/PWM <sub>TRIPINPUT</sub> /Programmable Logic Array Output Element 4. By default, this pin is configured as a digital input with a weak pull-up resistor enabled. When used as ADC input, the pull-up resistor should be disabled manually.   |
| 7        | N/A      | N/A      | P1.4/ADC10/PLAO[3]                          | General-Purpose Input and Output Port 1.4/ADC Single-Ended or Differential Analog Input 10/Programmable Logic Array Output Element 3.<br>By default, this pin is configured as a digital input with a weak pull-up resistor enabled. When used as ADC input, the pull-up resistor should be disabled manually.   |

| Pin No.  |          |          | Mnemonic                                    | Description  |
|----------|----------|----------|---|--|
| 40-LFCSP | 32-LFCSP | 36-WLCSP |   |  |
| 34       | 26       | A3       | P1.3/ADC5/IRQ3/PLAI[4]                      | General-Purpose Input and Output Port 1.3/ADC Single-Ended or Differential Analog Input 5/External Interrupt Request 3/ Programmable Logic Array Input Element 4.<br>By default, this pin is configured as a digital input with a weak pull-up resistor enabled. When used as ADC input, the pull-up resistor should be disabled manually.                                 |
| 33       | 25       | A2       | P1.2/ADC4/IRQ2/PLAI[3]/ECLK/                | General-Purpose Input and Output Port 1.2/ADC Single-Ended or Differential Analog Input 4/External Interrupt Request 2/ Programmable Logic Array Input Element 3/Input-Output for External Clock.<br>By default, this pin is configured as a digital input with a weak pull-up resistor enabled. When used as ADC input, the pull-up resistor should be disabled manually. |
| 14       | 12       | F4       | P1.1/ $\overline{SS}$ /IRQ1/PWM1/PLAO[2]/T1 | General-Purpose Input and Output Port 1.1/SPI Interface Slave Select (Active Low)/External Interrupt Request 1/PWM Output 1/ Programmable Logic Array Output Element 2/Timer 1 Input Clock. By default, this pin is configured as a digital input with a weak pull-up resistor enabled.  |
| 13       | 11       | E4       | P1.0/SCLK/PWM0/PLAO[1]                      | General-Purpose Input and Output Port 1.0/SPI Interface Clock Signal/ PWM Output 0/Programmable Logic Array Output Element 1. By default, this pin is configured as a digital input with a weak pull-up resistor enabled.  |
| 35       | 27       | B3       | V <sub>REF</sub>                            | 2.5 V Internal Voltage Reference. Must be connected to a 0.47 $\mu$ F capacitor when using the internal reference.   |
| 40       | 32       | A6       | AGND  | Analog Ground. Ground reference point for the analog circuitry.  |
| 1        | 1        | B6       | AV <sub>DD</sub>                            | 3.3 V Analog Power.  |

### TYPICAL PERFORMANCE CHARACTERISTICS

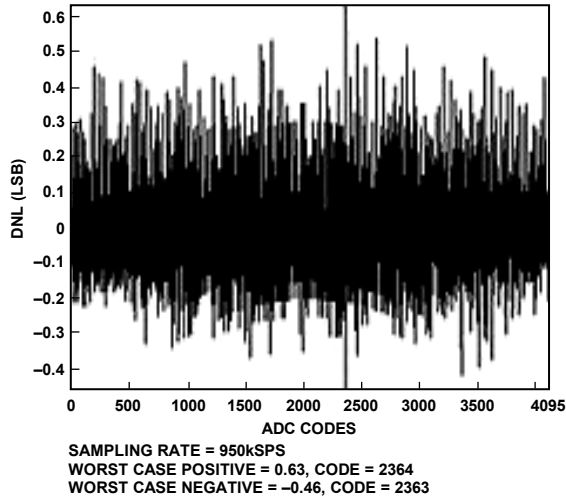


Figure 10. Typical DNL,  $f_{ADC} = 950$  kSPS, Internal Reference Used

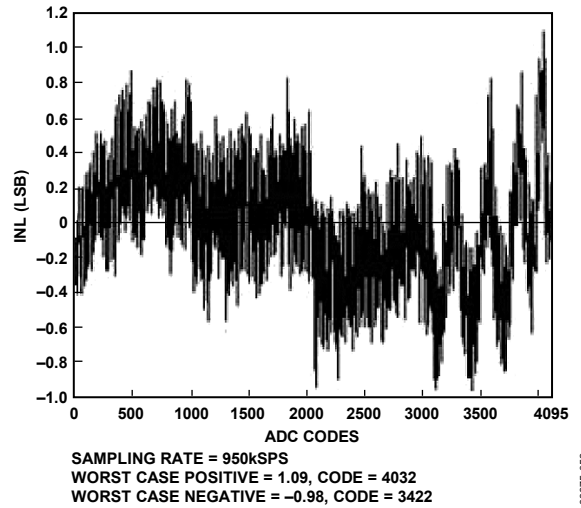


Figure 13. Typical INL,  $f_{ADC} = 950$  kSPS, External 1.0 V Reference Used

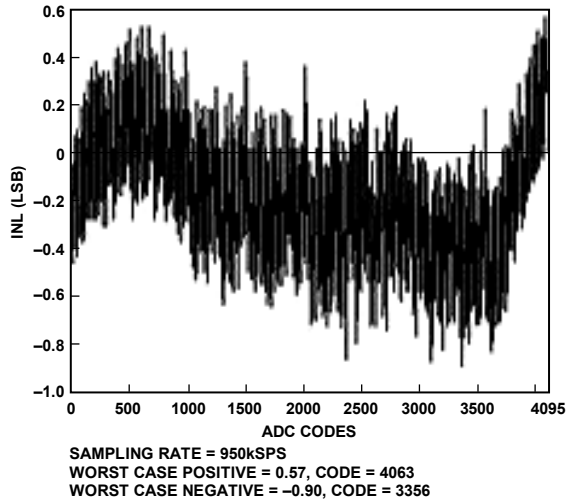


Figure 11. Typical INL,  $f_{ADC} = 950$  kSPS, Internal Reference Used

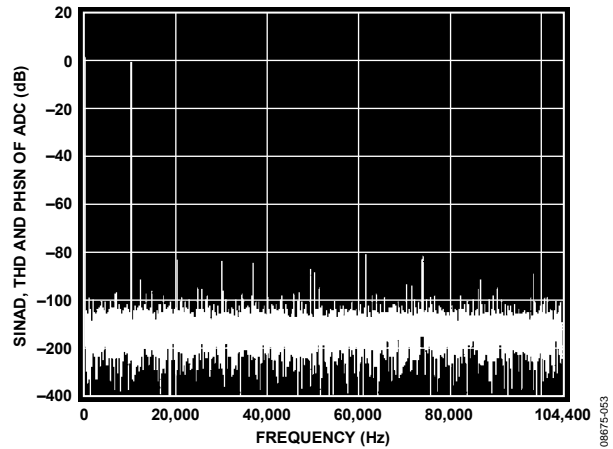


Figure 14. SINAD, THD, and PHSN of ADC, Internal 2.5 V Reference Used

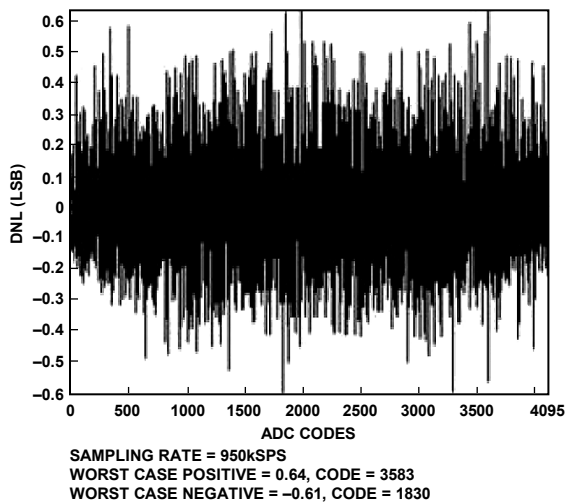


Figure 12. Typical DNL,  $f_{ADC} = 950$  kSPS, External 1.0 V Reference Used

## TERMINOLOGY

### ADC SPECIFICATIONS

#### Integral Nonlinearity (INL)

The maximum deviation of any code from a straight line passing through the endpoints of the ADC 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.

#### Differential Nonlinearity (DNL)

The difference between the measured and the ideal 1 LSB change between any two adjacent codes in the ADC.

#### Offset Error

The deviation of the first code transition (0000 . . . 000) to (0000 . . . 001) from the ideal, that is,  $\pm\frac{1}{2}$  LSB.

#### Gain Error

The deviation of the last code transition from the ideal AIN voltage (full scale – 1.5 LSB) after the offset error has been adjusted out.

#### Signal to (Noise + Distortion) Ratio

The measured ratio of signal to (noise + distortion) at the output of the ADC. The signal is the rms amplitude of the fundamental. Noise is the rms sum of all nonfundamental signals up to half the sampling frequency ( $f_s/2$ ), excluding dc.

The ratio is dependent upon the number of quantization levels in the digitization process; the more levels, the smaller the quantization noise.

The theoretical signal to (noise + distortion) ratio for an ideal N-bit converter with a sine wave input is given by

$$\text{Signal to (Noise + Distortion)} = (6.02 N + 1.76) \text{ dB}$$

Thus, for a 12-bit converter, this is 74 dB.

#### Total Harmonic Distortion

The ratio of the rms sum of the harmonics to the fundamental.

### DAC SPECIFICATIONS

#### Relative Accuracy

Otherwise known as endpoint linearity, relative accuracy is a measure of the maximum deviation from a straight line passing through the endpoints of the DAC transfer function. It is measured after adjusting for zero error and full-scale error.

#### Voltage Output Settling Time

The amount of time it takes the output to settle to within a 1 LSB level for a full-scale input change.

## OVERVIEW OF THE ARM7TDMI CORE

The ARM7<sup>®</sup> core is a 32-bit reduced instruction set computer (RISC). It uses a single 32-bit bus for instruction and data. The length of the data can be 8 bits, 16 bits, or 32 bits. The length of the instruction word is 32 bits.

The ARM7TDMI is an ARM7 core with four additional features: T support for the thumb (16-bit) instruction set, D support for debug, M support for long multiplications, and I includes the EmbeddedICE module to support embedded system debugging.

### THUMB MODE (T)

An ARM instruction is 32 bits long. The ARM7TDMI processor supports a second instruction set that has been compressed into 16 bits, called the Thumb<sup>®</sup> instruction set. Faster execution from 16-bit memory and greater code density can usually be achieved by using the Thumb instruction set instead of the ARM instruction set, which makes the ARM7TDMI core particularly suitable for embedded applications.

However, the Thumb mode has two limitations. Thumb code typically requires more instructions for the same job. As a result, ARM code is usually best for maximizing the performance of time critical code. Also, the Thumb instruction set does not include some of the instructions needed for exception handling, which automatically switches the core to ARM code for exception handling.

See the ARM7TDMI user guide for details on the core architecture, the programming model, and both the ARM and ARM Thumb instruction sets.

### LONG MULTIPLY (M)

The ARM7TDMI instruction set includes four extra instructions that 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. These results are achieved in fewer cycles than required on a standard ARM7 core.

### EmbeddedICE (I)

EmbeddedICE provides integrated on-chip support for the core. The EmbeddedICE module contains the breakpoint and watchpoint registers that allow code to be halted for debugging purposes. These registers are controlled through the JTAG test port.

When a breakpoint or watchpoint is encountered, the processor halts and enters debug state. Once in a debug state, the processor registers can be inspected as well as the Flash/EE, SRAM, and memory mapped registers.

## EXCEPTIONS

ARM supports five types of exceptions and a privileged processing mode for each type. The five types of exceptions are:

- Normal interrupt or IRQ. This is provided to service general-purpose interrupt handling of internal and external events.
- Fast interrupt or FIQ. This is provided to service data transfers or communication channels with low latency. FIQ has priority over IRQ.
- Memory abort.
- Attempted execution of an undefined instruction.
- Software interrupt instruction (SWI). This can be used to make a call to an operating system.

Typically, the programmer defines interrupt as IRQ, but for higher priority interrupt, that is, faster response time, the programmer can define interrupt as FIQ.

## ARM REGISTERS

ARM7TDMI has a total of 37 registers: 31 general-purpose registers and six status registers. Each operating mode has dedicated banked registers.

When writing user-level programs, 15 general-purpose 32-bit registers (R0 to R14), the program counter (R15), and the current program status register (CPSR) are usable. The remaining registers are only used for system-level programming and exception handling.

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 15. The fast interrupt mode has more registers (R8 to R12) for fast interrupt processing. This means the interrupt processing can begin without the need to save or restore these registers, and thus save critical time in the interrupt handling process.

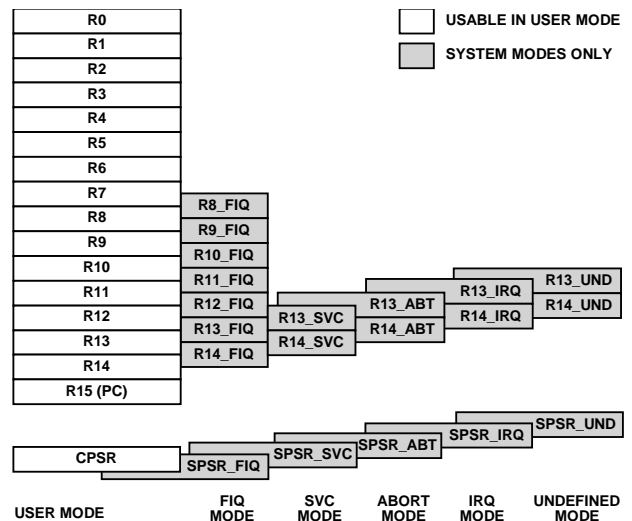


Figure 15. Register Organization

More information relative to the model of the programmer 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 a fast interrupt request (FIQ) consists of the following: the longest time the request can take to pass through the synchronizer, the time for the longest instruction to complete (the longest instruction is an LDM) that loads all the registers including the PC, and the time for the data abort and FIQ entry.

At the end of this time, the ARM7TDMI executes the instruction at 0x1C (FIQ interrupt vector address). The maximum total time is 50 processor cycles, which is just under 1.2  $\mu$ s in a system using a continuous 41.78 MHz processor clock.

The maximum interrupt request (IRQ) latency calculation is similar but must allow for the fact that FIQ has higher priority and could 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 the time is reduced to 22 cycles.

The minimum latency for FIQ or IRQ interrupts is a total of five cycles, which consist of the shortest time the request can take through the synchronizer, plus the time to enter the exception mode.

The ARM7TDMI always runs in ARM (32-bit) mode when in privileged modes, for example, when executing interrupt service routines.

## MEMORY ORGANIZATION

The ADuC7023 incorporates two separate blocks of memory: 8 kB of SRAM and 64 kB of on-chip Flash/EE memory; 62 kB of on-chip Flash/EE memory is available to the user, and the remaining 2 kB are reserved for the factory configured boot page. These two blocks are mapped as shown in Figure 16.

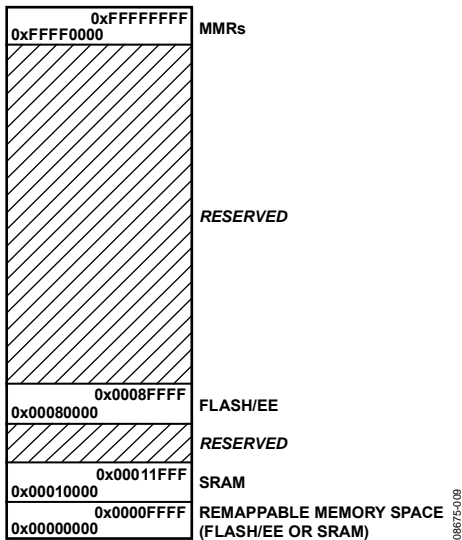


Figure 16. Physical Memory Map

By default, after a reset, the Flash/EE memory is mirrored at Address 0x00000000. It is possible to remap the SRAM at Address 0x00000000 by clearing Bit 0 of the Remap MMR. This remap function is described in more detail in the Flash/EE Memory section.

### MEMORY ACCESS

The ARM7 core sees memory as a linear array of the 2<sup>32</sup> byte location where the different blocks of memory are mapped as outlined in Figure 16.

The ADuC7023 memory organizations are configured in little endian format, which means that the least significant byte is located in the lowest byte address, and the most significant byte is in the highest byte address.

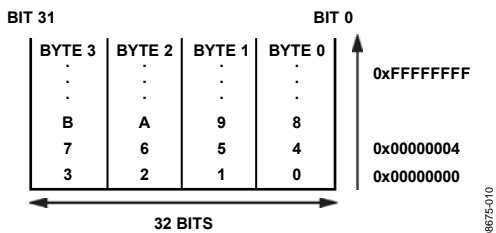


Figure 17. Little Endian Format

### FLASH/EE MEMORY

The total 64 kB of Flash/EE memory is organized as 32k × 16 bits; 31k × 16 bits is user space and 1 k × 16 bits is reserved for the on-chip kernel. The page size of this Flash/EE memory is 512 bytes.

62 kilobytes of Flash/EE memory are available to the user as code and nonvolatile data memory. There is no distinction between data and program because ARM code shares the same space. The real width of the Flash/EE memory is 16 bits, which means that in ARM mode (32-bit instruction), two accesses to the Flash/EE are necessary for each instruction fetch. It is, therefore, recommended to use Thumb mode when executing from Flash/EE memory for optimum access speed. The maximum access speed for the Flash/EE memory is 41.78 MHz in Thumb mode and 20.89 MHz in full ARM mode. More details about Flash/EE access time are outlined later in the Execution Time from SRAM and Flash/EE section.

### SRAM

Eight kilobytes of SRAM are available to the user, organized as 2k × 32 bits, that is, two words. ARM code can run directly from SRAM at 41.78 MHz, given that the SRAM array is configured as a 32-bit wide memory array. More details about SRAM access time are outlined later in the Execution Time from SRAM and Flash/EE section.

### MEMORY MAPPED REGISTERS

The memory mapped register (MMR) space is mapped into the upper two pages of the memory array and 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 18 are unoccupied or reserved locations and should not be accessed by user software. Table 10 to Table 23 show 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: advanced high performance bus (AHB) used for system modules and advanced peripheral bus (APB) used for lower performance peripheral. Access to the AHB is one cycle, and access to the APB is two cycles. All peripherals on the ADuC7023 are on the APB except the Flash/EE memory and the GPIOs.

|            |                            |
|------------|----------------------------|
| 0xFFFFFFFF |                            |
| 0xFFFFF820 | FLASH CONTROL INTERFACE    |
| 0xFFFFF800 |                            |
| 0xFFFFF46C | GPIO                       |
| 0xFFFFF400 |                            |
| 0xFFFF0FBF | PWM                        |
| 0xFFFF0F80 |                            |
| 0xFFFF0B54 | PLA                        |
| 0xFFFF0B00 |                            |
| 0xFFFF0A14 | SPI                        |
| 0xFFFF0A00 |                            |
| 0xFFFF0948 | I <sup>2</sup> C1          |
| 0xFFFF0900 |                            |
| 0xFFFF0848 | I <sup>2</sup> C0          |
| 0xFFFF0800 |                            |
| 0xFFFF0620 | DAC                        |
| 0xFFFF0600 |                            |
| 0xFFFF0538 | ADC                        |
| 0xFFFF0500 |                            |
| 0xFFFF0490 | BAND GAP REFERENCE         |
| 0xFFFF048C |                            |
| 0xFFFF0448 | POWER SUPPLY MONITOR       |
| 0xFFFF0440 |                            |
| 0xFFFF0420 | PLL AND OSCILLATOR CONTROL |
| 0xFFFF0404 |                            |
| 0xFFFF0370 | WATCHDOG TIMER             |
| 0xFFFF0360 |                            |
| 0xFFFF0334 | GENERAL-PURPOSE TIMER      |
| 0xFFFF0320 |                            |
| 0xFFFF0310 | TIMER0                     |
| 0xFFFF0300 |                            |
| 0xFFFF0238 | REMAP AND SYSTEM CONTROL   |
| 0xFFFF0220 |                            |
| 0xFFFF0140 | INTERRUPT CONTROLLER       |
| 0xFFFF0000 |                            |

08675-011

Figure 18. Memory Mapped Registers



Table 10. IRQ Address Base = 0xFFFF0000

| Address | Name     | Byte | Access Type | Default Value | Description   |
|---------|----------|------|-------------|---------------|---|
| 0x0000  | IRQSTA   | 4    | R           | 0x00000000    | Active IRQ source.  |
| 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           |               | MMR to disable IRQ sources.   |
| 0x0010  | SWICFG   | 4    | W           |               | Software interrupt configuration MMR.   |
| 0x0014  | IRQBASE  | 4    | R/W         | 0x00000000    | Base address of all vectors. Points to start of a 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    | This register 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    | This register contains the interrupt priority setting for Interrupt Source 8 to Interrupt Source 15.  |
| 0x0028  | IRQP2    | 4    | R/W         | 0x00000000    | This register contains the interrupt priority setting for Interrupt Source 16 to Interrupt Source 21.   |
| 0x002C  | RESERVED | 4    | R/W         | 0x00000000    | Reserved.   |
| 0x0030  | IRQCONN  | 4    | R/W         | 0x00000000    | Used to enable IRQ and FIQ interrupt nesting.   |
| 0x0034  | IRQCONE  | 4    | R/W         | 0x00000000    | This register 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.  |
| 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           |               | MMR to disable FIQ sources.   |
| 0x011C  | FIQVEC   | 4    | R           | 0x00000000    | FIQ interrupt vector.   |
| 0x013C  | FIQSTAN  | 4    | RW          | 0x00000000    | This register indicates the priority level of an FIQ that has just caused an FIQ exception.   |

Table 11. System Control Address Base = 0xFFFF0200

| Address | Name               | Byte | Access Type | Default Value <sup>1</sup> | Description   |
|---------|--------------------|------|-------------|----------------------------|---|
| 0x0220  | Remap <sup>2</sup> | 1    | R/W         | 0x00                       | Remap control register.   |
| 0x0230  | RSTSTA             | 1    | R/W         | 0x01                       | RSTSTA status MMR.  |
| 0x0234  | RSTCLR             | 1    | W           | 0x00                       | RSTCLR MMR for clearing RSTSTA register.  |
| 0x0248  | RSTKEY1            | 1    | W           | 0xXX                       | 0x76 should be written to this register before writing to RSTCFG.                                 |
| 0x024C  | RSTCFG             | 1    | R/W         | 0x00                       | This register allows the DAC and GPIO outputs to retain state after a watchdog or software reset. |
| 0x0250  | RSTKEY2            | 1    | W           | 0xXX                       | 0xB1 should be written to this register after writing to RSTCFG.                                  |

<sup>1</sup> N/A means not applicable.<sup>2</sup> Updated by kernel.

Table 12. Timer Address Base = 0xFFFF0300

| Address | Name   | Byte | Access Type | Default Value <sup>1</sup> | Description                      |
|---------|--------|------|-------------|----------------------------|----------------------------------|
| 0x0300  | T0LD   | 2    | R/W         | 0x0000                     | Timer0 load register.            |
| 0x0304  | T0VAL  | 2    | R           | 0xFFFF                     | Timer0 value register.           |
| 0x0308  | T0CON  | 2    | R/W         | 0x0000                     | Timer0 control MMR.              |
| 0x030C  | T0CLRI | 1    | W           | 0xXX                       | Timer0 interrupt clear register. |
| 0x0320  | T1LD   | 4    | R/W         | 0x00000000                 | Timer1 load register.            |
| 0x0324  | T1VAL  | 4    | R           | 0xFFFFFFFF                 | Timer1 value register.           |
| 0x0328  | T1CON  | 4    | R/W         | 0x00000000                 | Timer1 control MMR.              |
| 0x032C  | T1CLRI | 1    | W           | 0xXX                       | Timer1 interrupt clear register. |
| 0x0330  | T1CAP  | 4    | R           | 0x00000000                 | Timer1 capture register.         |
| 0x0360  | T2LD   | 2    | R/W         | 0x0000                     | Timer2 load register.            |
| 0x0364  | T2VAL  | 2    | R           | 0xFFFF                     | Timer2 value register.           |
| 0x0368  | T2CON  | 2    | R/W         | 0x0000                     | Timer2 control MMR.              |
| 0x036C  | T2CLRI | 1    | W           | 0xXX                       | Timer2 interrupt clear register. |

<sup>1</sup> N/A means not applicable.

Table 13. PLL/PSM Base Address = 0xFFFF0400

| Address | Name    | Byte | Access Type | Default Value <sup>1</sup> | Description                                    |
|---------|---------|------|-------------|----------------------------|--|
| 0x0404  | POWKEY1 | 2    | W           | 0xFFFF                     | POWCON0 prewrite key.                          |
| 0x0408  | POWCON0 | 1    | R/W         | 0x00                       | 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         | 0x21                       | 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         | 0x0004                     | Power control and core speed control register. |
| 0x043C  | POWKEY4 | 2    | W           | 0xFFFF                     | POWCON1 postwrite key.                         |
| 0x0440  | PSMCON  | 2    | R/W         | 0x0008                     | Power supply monitor control register.         |
| 0x0444  | CMPCON  | 2    | R/W         | 0x0000                     | Comparator control register.                   |

<sup>1</sup> N/A means not applicable.

Table 14. Reference Base Address = 0xFFFF0480

Address: 0x048C  
 Name: REFCON  
 Byte: 1  
 Access type: Read/write  
 Default value: 0x00  
 Description: Reference control register.

Table 15. ADC Address Base = 0xFFFF0500

| Address | Name   | Byte | Access Type | Default Value | Description                              |
|---------|--------|------|-------------|---------------|--|
| 0x0500  | ADCCON | 2    | R/W         | 0x0600        | ADC control MMR.                         |
| 0x0504  | ADCCP  | 1    | R/W         | 0x00          | ADC positive channel selection register. |
| 0x0508  | ADCCN  | 1    | R/W         | 0x01          | ADC negative channel selection register. |
| 0x050C  | ADCSTA | 1    | R           | 0x00          | ADC status MMR.                          |
| 0x0510  | ADCDAT | 4    | R           | 0x00000000    | ADC data output MMR.                     |
| 0x0514  | ADCRST | 1    | R/W         | 0x00          | ADC reset MMR.                           |

| Address | Name    | Byte | Access Type | Default Value      | Description                                  |
|---------|---------|------|-------------|--------------------|--|
| 0x0530  | ADCGN   | 2    | R/W         | Factory configured | ADC gain calibration MMR.                    |
| 0x0534  | ADCOF   | 2    | R/W         | Factory configured | ADC offset calibration MMR.                  |
| 0x0544  | TSCON   | 1    | R/W         | 0x00               | Temperature sensor chopping enable register. |
| 0x0548  | TEMPREF | 2    | R/W         | Factory configured | Temperature sensor reference value.          |

Table 16. DAC Address Base = 0xFFFF0600

| Address | Name     | Byte | Access Type | Default Value | Description           |
|---------|----------|------|-------------|---------------|-----------------------|
| 0x0600  | DAC0CON  | 1    | R/W         | 0x00          | DAC0 control MMR.     |
| 0x0604  | DAC0DAT  | 4    | R/W         | 0x00000000    | DAC0 data MMR.        |
| 0x0608  | DAC1CON  | 1    | R/W         | 0x00          | DAC1 control MMR.     |
| 0x060C  | DAC1DAT  | 4    | R/W         | 0x00000000    | DAC1 data MMR.        |
| 0x0610  | DAC2CON  | 1    | R/W         | 0x00          | DAC2 control MMR.     |
| 0x0614  | DAC2DAT  | 4    | R/W         | 0x00000000    | DAC2 data MMR.        |
| 0x0618  | DAC3CON  | 1    | R/W         | 0x00          | DAC3 control MMR.     |
| 0x061C  | DAC3DAT  | 4    | R/W         | 0x00000000    | DAC3 data MMR.        |
| 0x0654  | DACBCFG  | 1    | R/W         | 0x00          | DAC Configuration MMR |
| 0x0650  | DACBKEY0 | 2    | W           | 0x0000        | DAC Key0 MMR          |
| 0x0658  | DACBKEY1 | 2    | W           | 0x0000        | DAC Key1 MMR          |

Table 17. I<sup>2</sup>C0 Base Address = 0xFFFF0800

| Address | Name      | Byte | Access Type | Default Value | Description  |
|---------|-----------|------|-------------|---------------|--|
| 0x0800  | I2COMCON  | 2    | R/W         | 0x0000        | I <sup>2</sup> C0 master control register.   |
| 0x0804  | I2COMSTA  | 2    | R           | 0x0000        | I <sup>2</sup> C0 master status register.  |
| 0x0808  | I2COMRX   | 1    | R           | 0x00          | I <sup>2</sup> C0 master receive register.   |
| 0x080C  | I2COMTX   | 1    | W           | 0x00          | I <sup>2</sup> C0 master transmit register.  |
| 0x0810  | I2COMCNT0 | 2    | R/W         | 0x0000        | I <sup>2</sup> C0 master read count register. Write the number of required bytes into this register prior to reading from a slave device.            |
| 0x0814  | I2COMCNT1 | 1    | R           | 0x00          | I <sup>2</sup> C0 master current read count register. This register contains the number of bytes already received during a read from slave sequence. |
| 0x0818  | I2C0ADR0  | 1    | R/W         | 0x00          | I <sup>2</sup> C0 address byte register. Write the required slave address in here prior to communications.   |
| 0x081C  | I2C0ADR1  | 1    | R/W         | 0x00          | I <sup>2</sup> C0 address byte register. Write the required slave address in here prior to communications. Used in 10-bit mode only.                 |
| 0x0824  | I2C0DIV   | 2    | R/W         | 0x1F1F        | I <sup>2</sup> C0 clock control register. Used to configure the SCL frequency.   |
| 0x0828  | I2C0SCON  | 2    | R/W         | 0x0000        | I <sup>2</sup> C0 slave control register.  |
| 0x082C  | I2C0SSTA  | 2    | R/W         | 0x0000        | I <sup>2</sup> C0 slave status register.   |
| 0x0830  | I2C0SRX   | 1    | R           | 0x00          | I <sup>2</sup> C0 slave receive register.  |
| 0x0834  | I2C0STX   | 1    | W           | 0x00          | I <sup>2</sup> C0 slave transmit register.   |
| 0x0838  | I2C0ALT   | 1    | R/W         | 0x00          | I <sup>2</sup> C0 hardware general call recognition register.  |
| 0x083C  | I2C0ID0   | 1    | R/W         | 0x00          | I <sup>2</sup> C0 slave ID0 register. Slave bus ID register.   |
| 0x0840  | I2C0ID1   | 1    | R/W         | 0x00          | I <sup>2</sup> C0 slave ID1 register. Slave bus ID register.   |
| 0x0844  | I2C0ID2   | 1    | R/W         | 0x00          | I <sup>2</sup> C0 slave ID2 register. Slave bus ID register.   |
| 0x0848  | I2C0ID3   | 1    | R/W         | 0x00          | I <sup>2</sup> C0 slave ID3 register. Slave bus ID register.   |
| 0x084C  | I2C0FSTA  | 2    | R/W         | 0x0000        | I <sup>2</sup> C0 FIFO status register. Used in both master and slave modes.   |

Table 18. I<sup>2</sup>C1 Base Address = 0xFFFF0900

| Address | Name      | Byte | Access Type | Default Value | Description   |
|---------|-----------|------|-------------|---------------|---|
| 0x0900  | I2C1MCON  | 2    | R/W         | 0x0000        | I <sup>2</sup> C1 master control register.  |
| 0x0904  | I2C1MSTA  | 2    | R           | 0x0000        | I <sup>2</sup> C1 master status register.   |
| 0x0908  | I2C1MRX   | 1    | R           | 0x00          | I <sup>2</sup> C1 master receive register.  |
| 0x090C  | I2C1MTX   | 1    | W           | 0x00          | I <sup>2</sup> C1 master transmit register.   |
| 0x0910  | I2C1MCNT0 | 2    | R/W         | 0x0000        | I <sup>2</sup> C1 master read count register. Write the number of required bytes into this register prior to reading from a slave device. |

| Address | Name      | Byte | Access Type | Default Value | Description  |
|---------|-----------|------|-------------|---------------|--|
| 0x0914  | I2C1MCNT1 | 1    | R           | 0x00          | I <sup>2</sup> C1 master current read count register. This register contains the number of bytes already received during a read from slave sequence. |
| 0x0918  | I2C1ADR0  | 1    | R/W         | 0x00          | I <sup>2</sup> C1 address byte register. Write the required slave address in here prior to communications.   |
| 0x091C  | I2C1ADR1  | 1    | R/W         | 0x00          | I <sup>2</sup> C1 address byte register. Write the required slave address in here prior to communications. Used in 10-bit mode only.                 |
| 0x0924  | I2C1DIV   | 2    | R/W         | 0x1F1F        | I <sup>2</sup> C1 clock control register. Used to configure the SCL frequency.   |
| 0x0928  | I2C1SCON  | 2    | R/W         | 0x0000        | I <sup>2</sup> C1 slave control register.  |
| 0x092C  | I2C1SSTA  | 2    | R/W         | 0x0000        | I <sup>2</sup> C1 slave status register.   |
| 0x0930  | I2C1SRX   | 1    | R           | 0x00          | I <sup>2</sup> C1 slave receive register.  |
| 0x0934  | I2C1STX   | 1    | W           | 0x00          | I <sup>2</sup> C1 slave transmit register.   |
| 0x0938  | I2C1ALT   | 1    | R/W         | 0x00          | I <sup>2</sup> C1 hardware general call recognition register.  |
| 0x093C  | I2C1ID0   | 1    | R/W         | 0x00          | I <sup>2</sup> C1 slave ID0 register. Slave bus ID register.   |
| 0x0940  | I2C1ID1   | 1    | R/W         | 0x00          | I <sup>2</sup> C1 slave ID1 register. Slave bus ID register.   |
| 0x0944  | I2C1ID2   | 1    | R/W         | 0x00          | I <sup>2</sup> C1 slave ID2 register. Slave bus ID register.   |
| 0x0948  | I2C1ID3   | 1    | R/W         | 0x00          | I <sup>2</sup> C1 slave ID3 register. Slave bus ID register.   |
| 0x094C  | I2C1FSTA  | 2    | R/W         | 0x0000        | I <sup>2</sup> C1 FIFO status register. Used in both master and slave modes.   |

Table 19. SPI Base Address = 0xFFFF0A00

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

Table 20. PLA Base Address = 0xFFFF0B00

| Address | Name     | Byte | Access Type | Default Value | Description                       |
|---------|----------|------|-------------|---------------|-----------------------------------|
| 0x0B00  | PLAELM0  | 2    | R/W         | 0x0000        | PLA Element 0 control register.   |
| 0x0B04  | PLAELM1  | 2    | R/W         | 0x0000        | PLA Element 1 control register.   |
| 0x0B08  | PLAELM2  | 2    | R/W         | 0x0000        | PLA Element 2 control register.   |
| 0x0B0C  | PLAELM3  | 2    | R/W         | 0x0000        | PLA Element 3 control register.   |
| 0x0B10  | PLAELM4  | 2    | R/W         | 0x0000        | PLA Element 4 control register.   |
| 0x0B14  | PLAELM5  | 2    | R/W         | 0x0000        | PLA Element 5 control register.   |
| 0x0B18  | PLAELM6  | 2    | R/W         | 0x0000        | PLA Element 6 control register.   |
| 0x0B1C  | PLAELM7  | 2    | R/W         | 0x0000        | PLA Element 7 control register.   |
| 0x0B20  | PLAELM8  | 2    | R/W         | 0x0000        | PLA Element 8 control register.   |
| 0x0B24  | PLAELM9  | 2    | R/W         | 0x0000        | PLA Element 9 control register.   |
| 0x0B28  | PLAELM10 | 2    | R/W         | 0x0000        | PLA Element 10 control register.  |
| 0x0B2C  | PLAELM11 | 2    | R/W         | 0x0000        | PLA Element 11 control register.  |
| 0x0B30  | PLAELM12 | 2    | R/W         | 0x0000        | PLA Element 12 control register.  |
| 0x0B34  | PLAELM13 | 2    | R/W         | 0x0000        | PLA Element 13 control register.  |
| 0x0B38  | PLAELM14 | 2    | R/W         | 0x0000        | PLA Element 14 control register.  |
| 0x0B3C  | PLAELM15 | 2    | R/W         | 0x0000        | PLA Element 15 control register.  |
| 0x0B40  | PLACLK   | 1    | R/W         | 0x00          | PLA clock select register.        |
| 0x0B44  | PLAIRQ   | 4    | R/W         | 0x00000000    | PLA interrupt control register.   |
| 0x0B48  | PLAADC   | 4    | R/W         | 0x00000000    | PLA ADC trigger control register. |
| 0x0B4C  | PLADIN   | 4    | R/W         | 0x00000000    | PLA data in register.             |
| 0x0B50  | PLADOUT  | 4    | R           | 0x00000000    | PLA data out register.            |
| 0x0B54  | PLALCK   | 1    | W           | 0x00          | PLA lock register.                |

Table 21. PWM Base Address = 0xFFFF0F80

| Address | Name     | Byte | Access Type | Default Value | Description   |
|---------|----------|------|-------------|---------------|---|
| 0x0F80  | PWMCON1  | 2    | R/W         | 0x0012        | PWM Control Register 1. 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.  |
| 0x0FA8  | PWM2COM1 | 2    | R/W         | 0x0000        | Compare Register 1 for PWM Output 4.  |
| 0x0FB0  | PWM2LEN  | 2    | R/W         | 0x0000        | Frequency control for PWM Output 4.   |
| 0x0FB8  | PWMCLRI  | 2    | W           | 0x0000        | PWM interrupt clear register. Writing any value to this register clears a PWM interrupt source. |

Table 22. GPIO Base Address = 0xFFFF400

| Address | Name   | Byte | Access Type | Default Value | Description                     |
|---------|--------|------|-------------|---------------|---------------------------------|
| 0xF400  | GP0CON | 4    | R/W         | 0x00001111    | GPIO Port0 control MMR.         |
| 0xF404  | GP1CON | 4    | R/W         | 0x00000000    | GPIO Port1 control MMR.         |
| 0xF408  | GP2CON | 4    | R/W         | 0x00000000    | GPIO Port2 control MMR.         |
| 0xF420  | GP0DAT | 4    | R/W         | 0x000000XX    | GPIO Port0 data control MMR.    |
| 0xF424  | GP0SET | 4    | W           | 0x000000XX    | GPIO Port0 data set MMR.        |
| 0xF428  | GP0CLR | 4    | W           | 0x000000XX    | GPIO Port0 data clear MMR.      |
| 0xF42C  | GP0PAR | 4    | R/W         | 0x22220000    | GPIO Port0 pull-up disable MMR. |
| 0xF430  | GP1DAT | 4    | R/W         | 0x000000XX    | GPIO Port1 data control MMR.    |
| 0xF434  | GP1SET | 4    | W           | 0x000000XX    | GPIO Port1 data set MMR.        |
| 0xF438  | GP1CLR | 4    | W           | 0x000000XX    | GPIO Port1 data clear MMR.      |
| 0xF43C  | GP1PAR | 4    | R/W         | 0x22000022    | GPIO Port1 pull-up disable MMR. |
| 0xF440  | GP2DAT | 4    | R/W         | 0x000000XX    | GPIO Port2 data control MMR.    |
| 0xF444  | GP2SET | 4    | W           | 0x000000XX    | GPIO Port2 data set MMR.        |
| 0xF448  | GP2CLR | 4    | W           | 0x000000XX    | GPIO Port2 data clear MMR.      |
| 0xF44C  | GP2PAR | 4    | R/W         | 0x00000000    | GPIO Port2 pull-up disable MMR. |

Table 23. Flash/EE Base Address = 0xFFFF800

| Address | Name    | Byte | Access Type | Default Value | Description              |
|---------|---------|------|-------------|---------------|--------------------------|
| 0xF800  | FEESTA  | 1    | R           | 0x20          | Flash/EE status MMR.     |
| 0xF804  | FEEMOD  | 2    | R/W         | 0x0000        | Flash/EE control MMR.    |
| 0xF808  | FEECON  | 1    | R/W         | 0x07          | Flash/EE control MMR.    |
| 0xF80C  | FEEDAT  | 2    | R/W         | 0xFFFF        | Flash/EE data MMR.       |
| 0xF810  | FEEADR  | 2    | R/W         | 0x0000        | Flash/EE address MMR.    |
| 0xF818  | FEESIGN | 3    | R           | 0xFFFFFFFF    | Flash/EE LFSR MMR.       |
| 0xF81C  | FEEPRO  | 4    | R/W         | 0x00000000    | Flash/EE protection MMR. |
| 0xF820  | FEEHIDE | 4    | R/W         | 0xFFFFFFFF    | Flash/EE protection MMR. |

## ADC CIRCUIT OVERVIEW

The analog-to-digital converter (ADC) incorporates a fast, multichannel, 12-bit ADC. It can operate from 2.7 V to 3.6 V supplies and is capable of providing a throughput of up to 1 MSPS when the clock source is 41.78 MHz. This block provides the user with a multichannel multiplexer, a differential track-and-hold, an on-chip reference, and an ADC.

The ADC consists of a 12-bit successive approximation converter based around two capacitor DACs. Depending on the input signal configuration, the ADC can operate in one of two different modes: fully differential mode (for small and balanced signals) or single-ended mode (for any single-ended signals).

The converter accepts an analog input range of 0 V to  $V_{REF}$  when operating in single-ended mode. In fully differential mode, the input signal must be balanced around a common-mode voltage ( $V_{CM}$ ) in the 0 V to  $AV_{DD}$  range with a maximum amplitude of  $2V_{REF}$  (see Figure 19).

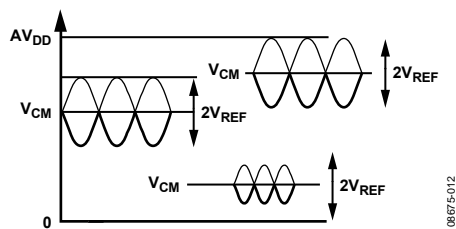


Figure 19. Examples of Balanced Signals in Fully Differential Mode

A high precision, low drift, factory calibrated, 2.5 V reference is provided on chip. An external reference can also be connected as described later in the Band Gap Reference section.

Single or continuous conversion modes can be initiated in the software. An external  $CONV_{START}$  pin, an output generated from the on-chip PLA, or a Timer0 or Timer1 overflow can also be used to generate a repetitive trigger for ADC conversions.

A voltage output from an on-chip band gap reference proportional to absolute temperature can also be routed through the front-end ADC multiplexer. This temperature channel can be selected as an ADC input. This facilitates an internal temperature sensor channel that measures die temperature.

## TRANSFER FUNCTION

### Single-Ended Mode

In single-ended mode, the input range is 0 V to  $V_{REF}$ . The output coding is straight binary in single-ended mode with

$$1 \text{ LSB} = FS/4096, \text{ or}$$

$$2.5 \text{ V}/4096 = 0.61 \text{ mV, or}$$

$$610 \text{ } \mu\text{V} \text{ when } V_{REF} = 2.5 \text{ V}$$

The ideal code transitions occur midway between successive integer LSB values (that is,  $1/2$  LSB,  $3/2$  LSB,  $5/2$  LSB, ...,  $FS - 3/2$  LSB). The ideal input/output transfer characteristic is shown in Figure 20.

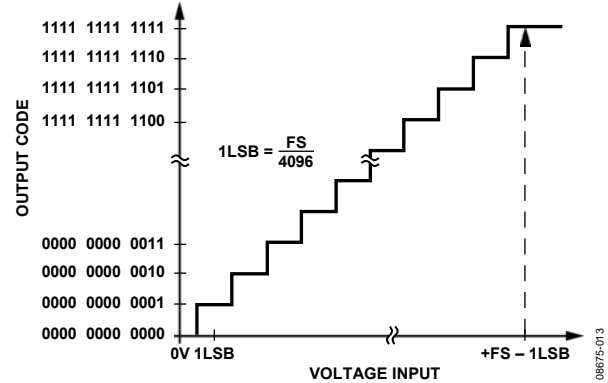


Figure 20. ADC Transfer Function in Single-Ended Mode

### Fully Differential Mode

The amplitude of the differential signal is the difference between the signals applied to the  $V_{IN+}$  and  $V_{IN-}$  pins (that is,  $V_{IN+} - V_{IN-}$ ). The maximum amplitude of the differential signal is, therefore,  $-V_{REF}$  to  $+V_{REF}$  p-p (that is,  $2 \times V_{REF}$ ). This is regardless of the common mode (CM). The common mode is the average of the two signals, for example,  $(V_{IN+} + V_{IN-})/2$ , and is, therefore, the voltage on which the two inputs are centered. This results in the span of each input being  $CM \pm V_{REF}/2$ . This voltage has to be set up externally, and its range varies with  $V_{REF}$  (see the Driving the Analog Inputs section).

The output coding is twos complement in fully differential mode with  $1 \text{ LSB} = 2V_{REF}/4096$  or  $2 \times 2.5 \text{ V}/4096 = 1.22 \text{ mV}$  when  $V_{REF} = 2.5 \text{ V}$ . The output result is  $\pm 11$  bits, but this is shifted by one to the right. This allows the result in the ADCDAT MMR to be declared as a signed integer when writing C code. The designed code transitions occur midway between successive integer LSB values (that is,  $1/2$  LSB,  $3/2$  LSB,  $5/2$  LSB, ...,  $FS - 3/2$  LSB). The ideal input/output transfer characteristic is shown in Figure 21.

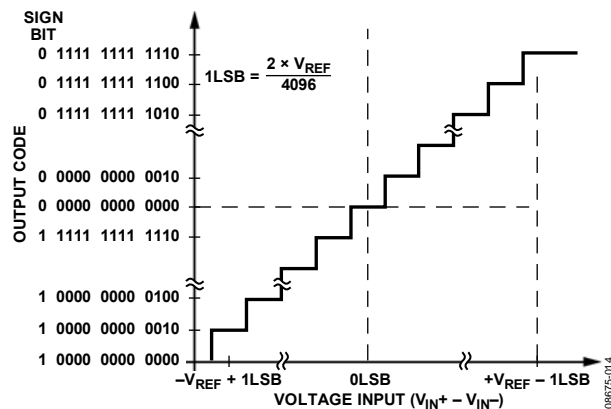


Figure 21. ADC Transfer Function in Differential Mode

**TYPICAL OPERATION**

When configured via the ADC control and channel selection registers, the ADC converts the analog input and provides a 12-bit result in the ADC data register.

The top four bits are the sign bits. The 12-bit result is placed from Bit 16 to Bit 27 as shown in Figure 22. Note that in fully differential mode, the result is represented in twos complement format. In single-ended mode, the result is represented in straight binary format.

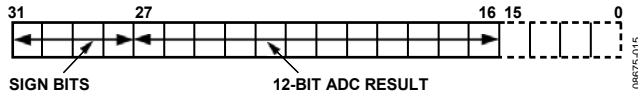


Figure 22. ADC Result Format

The same format is used in DACxDAT, simplifying the software.

**Current Consumption**

The ADC in standby mode, that is, powered up but not converting, typically consumes 640  $\mu$ A. The internal reference adds 140  $\mu$ A. During conversion, the extra current is 0.3  $\mu$ A multiplied by the sampling frequency (in kHz).

**Timing**

Figure 23 gives details of the ADC timing. Users control the ADC clock speed and the number of acquisition clocks in the ADCCON MMR. By default, the acquisition time is eight clocks, and the clock divider is two. The number of extra clocks (such as bit trial or write) is set to 19, which gives a sampling rate of 774 kSPS. For conversion on the temperature sensor, set ADCCON = 0x37A3. When using multiple channels including the temperature sensor, the timing settings revert to the user-defined settings after reading the temperature sensor channel.

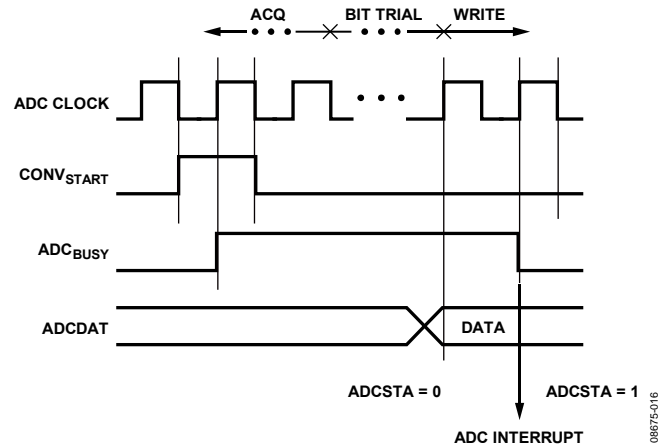


Figure 23. ADC Timing

**MMR INTERFACE**

The ADC is controlled and configured via the eight MMRs described in this section.

**ADCCON Register**

- Name: ADCCON
- Address: 0xFFFF0500
- Default value: 0x0600
- Access: Read/write
- Function: ADCCON is an ADC control register that allows the programmer to enable the ADC peripheral, select the mode of operation of the ADC (either in single-ended mode or fully differential mode), and select the conversion type. This MMR is described in Table 24.

Table 24. ADCCON MMR Bit Designations

| Bit      | Value                                  | Description  |
|----------|--|--|
| 15 to 14 |  | Reserved.  |
| 13       |  | Temperature sensor conversion enable. Set to 1 for temperature sensor conversions and single software conversions. Set to 0 for normal ADC conversions.  |
| 12 to 10 | 000<br>001<br>010<br>011<br>100<br>101 | ADC clock speed.<br>$f_{ADC}/1$ . This divider is provided to obtain 1 MSPS ADC with an external clock <41.78 MHz.<br>$f_{ADC}/2$ (default value).<br>$f_{ADC}/4$ .<br>$f_{ADC}/8$ .<br>$f_{ADC}/16$ .<br>$f_{ADC}/32$ . |
| 9 to 8   | 00<br>01<br>10<br>11                   | ADC acquisition time.<br>2 clocks.<br>4 clocks.<br>8 clocks (default value).<br>16 clocks.   |

| Bit    | Value   | Description  |
|--------|---|--|
| 7      |   | Enable start conversion.<br>This bit is set by the user to start any type of conversion command.<br>This bit is cleared by the user to disable a start conversion (clearing this bit does not stop the ADC when continuously converting).  |
| 6      |   | Reserved   |
| 5      |   | ADC power control.<br>This bit is set by the user to place the ADC in normal mode (the ADC must be powered up for at least 5 $\mu$ s before it converts correctly).<br>This bit is cleared by the user to place the ADC in power-down mode.  |
| 4 to 3 | 00<br>01<br>10<br>11                                | Conversion mode.<br>Single-ended mode.<br>Differential mode.<br>Reserved.<br>Reserved.   |
| 2 to 0 | 000<br>001<br>010<br>011<br><br>100<br>101<br>Other | Conversion type.<br>Enable CONV <sub>START</sub> pin as a conversion input.<br>Enable Timer1 as a conversion input.<br>Enable Timer0 as a conversion input.<br>Single software conversion. This bit is set to 000 after conversion (note that Bit 13 of the ADCCON MMR should be set before starting a single software conversion to avoid further conversions triggered by the CONV <sub>START</sub> pin).<br>Continuous software conversion.<br>PLA conversion.<br>Reserved. |

**ADCCP Register**

|                |   |
|----------------|---|
| Name:          | ADCCP   |
| Address:       | 0xFFFF0504  |
| Default value: | 0x00  |
| Access:        | Read/write  |
| Function:      | ADCCP is an ADC positive channel selection register. This MMR is described in Table 25. |

**Table 25. ADCCP MMR Bit Designation**

| Bit    | Value  | Description                                   |
|--------|--------|---|
| 7 to 5 |        | Reserved.                                     |
| 4 to 0 |        | Positive channel selection bits.              |
|        | 00000  | ADC0.   |
|        | 00001  | ADC1.   |
|        | 00010  | ADC2.   |
|        | 00011  | ADC3.   |
|        | 00100  | ADC4 <sup>1</sup> .                           |
|        | 00101  | ADC5 <sup>1</sup> .                           |
|        | 00110  | ADC6 <sup>1</sup> .                           |
|        | 00111  | ADC7 <sup>1</sup> .                           |
|        | 01000  | ADC8 <sup>1</sup> .                           |
|        | 01001  | ADC9 <sup>1</sup> .                           |
|        | 01010  | ADC10 <sup>1</sup> .                          |
|        | 01011  | Reserved.                                     |
|        | 01100  | ADC12 <sup>1</sup> .                          |
|        | 01101  | Reserved                                      |
|        | 01110  | DAC0  |
|        | 01111  | DAC1  |
|        | 10000  | Temperature sensor.                           |
|        | 10001  | AGND (self-diagnostic feature).               |
|        | 10010  | Internal reference (self-diagnostic feature). |
|        | 10011  | AV <sub>DD</sub> /2.                          |
|        | Others | Reserved.                                     |

<sup>1</sup> When a selected ADC channel is shared with one GPIO, by default, this pin is configured with a weak pull-up resistor enabled. The pull-up resistor should be disabled manually in the appropriate GPxPAR register. Note the internal pull-up resistor on P2.0/AIN12 for 40-lead package cannot be disabled.



**ADCCN Register**

Name: ADCCN  
 Address: 0xFFFFF0508  
 Default value: 0x01  
 Access: Read/write  
 Function: ADCCN is an ADC negative channel selection register. This MMR is described in Table 26.

**ADCSTA Register**

Name: ADCSTA  
 Address: 0xFFFFF050C  
 Default Value: 0x00  
 Access: Read  
 Function: ADCSTA is an ADC status register that indicates when an ADC conversion result is ready. The ADCSTA register contains only one bit, ADCReady (Bit 0), representing the status of the ADC. This bit is set at the end of an ADC conversion, generating an ADC interrupt. It is cleared automatically by reading the ADCDAT MMR. When the ADC is performing a conversion, the status of the ADC can be read externally via the ADC<sub>BUSY</sub> pin. This pin is high during a conversion. When the conversion is finished, ADC<sub>BUSY</sub> goes back low. This information can be available on P0.0 (see the General-Purpose Input/Output section) if enabled in the ADCCON register.

**Table 26. ADCCN MMR Bit Designation**

| Bit    | Value  | Description                                   |
|--------|--------|---|
| 7 to 5 |        | Reserved.                                     |
| 4 to 0 |        | Negative channel selection bits.              |
|        | 00000  | ADC0.   |
|        | 00001  | ADC1.   |
|        | 00010  | ADC2.   |
|        | 00011  | ADC3.   |
|        | 00100  | ADC4.   |
|        | 00101  | ADC5.   |
|        | 00110  | ADC6.   |
|        | 00111  | ADC7.   |
|        | 01000  | ADC8.   |
|        | 01001  | ADC9.   |
|        | 01010  | ADC10.  |
|        | 01011  | Reserved                                      |
|        | 01100  | ADC12.  |
|        | 01101  | Reserved                                      |
|        | 01110  | Reserved                                      |
|        | 01111  | DAC1.   |
|        | 10000  | Temperature sensor.                           |
|        | 10001  | AGND (self-diagnostic feature).               |
|        | 10010  | Internal reference (self-diagnostic feature). |
|        | 10011  | Reserved                                      |
|        | Others | Reserved.                                     |

**ADCDAT Register**

Name: ADCDAT  
 Address: 0xFFFFF0510  
 Default value: 0x00000000  
 Access: Read  
 Function: ADCDAT is an ADC data result register. Hold the 12-bit ADC result as shown in Figure 22.

**ADCRST Register**

Name: ADCRST  
 Address: 0xFFFFF0514  
 Default Value: 0x00  
 Access: Read/write  
 Function: ADCRST resets the digital interface of the ADC. Writing any value to this register resets all the ADC registers to their default value.

## ADCGN Register

Name: ADCGN  
 Address: 0xFFFF0530  
 Default value: Factory configured  
 Access: Read/write  
 Function: ADCGN is a 10-bit gain calibration register.

## ADCOF Register

Name: ADCOF  
 Address: 0xFFFF0534  
 Default value: Factory configured  
 Access: Read/write  
 Function: ADCOF is a 10-bit offset calibration register.

## CONVERTER OPERATION

The ADC incorporates a successive approximation (SAR) architecture involving a charge-sampled input stage. This architecture can operate in two different modes: differential and single-ended.

### Differential Mode

The ADuC7023 contains a successive approximation ADC based on two capacitive DACs. Figure 24 and Figure 25 show simplified schematics of the ADC in acquisition and conversion phase, respectively. The ADC is comprised of control logic, a SAR, and two capacitive DACs. In Figure 24 (the acquisition phase), SW3 is closed and SW1 and SW2 are in Position A. The comparator is held in a balanced condition, and the sampling capacitor arrays acquire the differential signal on the input.

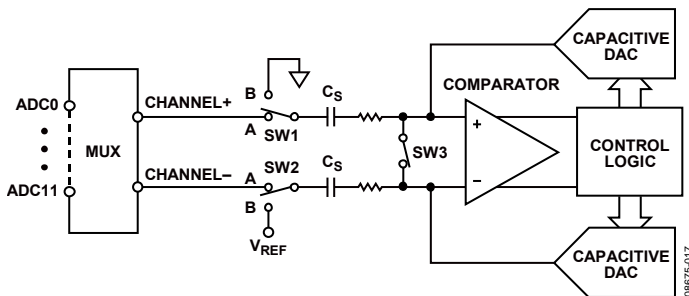


Figure 24. ADC Acquisition Phase

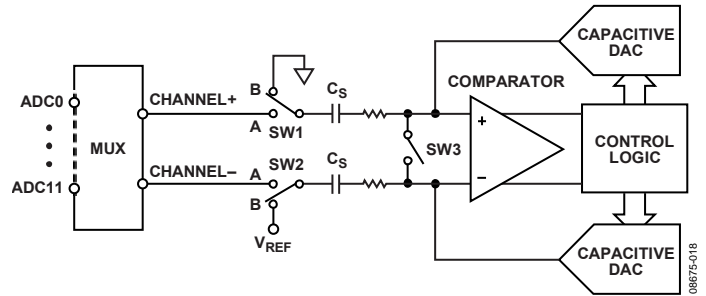


Figure 25. ADC Conversion Phase

When the ADC starts a conversion, as shown in Figure 25, SW3 opens, and then SW1 and SW2 move to Position B. This causes the comparator to become unbalanced. Both inputs are disconnected once the conversion begins. The control logic and the charge redistribution DACs are used to add and subtract fixed amounts of charge from the sampling capacitor arrays to bring the comparator back into a balanced condition. When the comparator is rebalanced, the conversion is complete. The control logic generates the ADC output code. The output impedances of the sources driving the  $V_{IN+}$  and  $V_{IN-}$  pins must be matched; otherwise, the two inputs have different settling times, resulting in errors.

### Single-Ended Mode

In single-ended mode, SW2 is always connected internally to ground. The  $V_{IN-}$  pin can be floating. The input signal range on  $V_{IN+}$  is 0 V to  $V_{REF}$ .

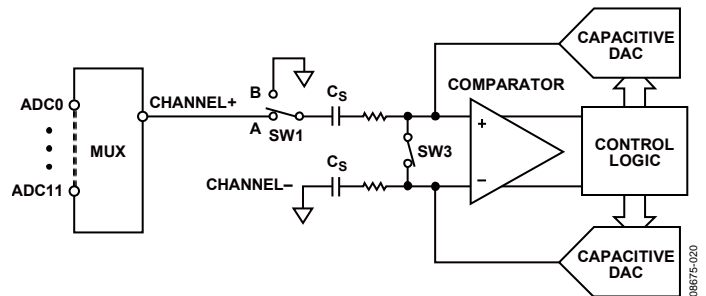


Figure 26. ADC in Single-Ended Mode

### Analog Input Structure

Figure 27 shows the equivalent circuit of the analog input structure of the ADC. The four diodes provide ESD protection for the analog inputs. Care must be taken to ensure that the analog input signals never exceed the supply rails by more than 300 mV; this causes these diodes to become forward-biased and start conducting into the substrate. These diodes can conduct up to 10 mA without causing irreversible damage to the part.

The C1 capacitors in Figure 27 are typically 4 pF and can be primarily attributed to pin capacitance. The resistors are lumped components made up of the on resistance of the switches. The value of these resistors is typically about 100 Ω. The C2 capacitors are the ADC sampling capacitors and typically have a capacitance of 16 pF.

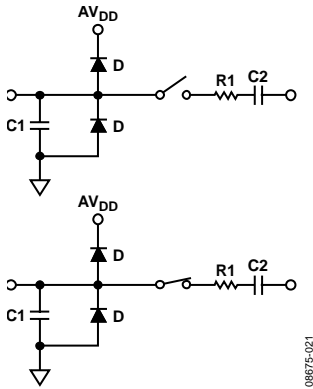


Figure 27. Equivalent Analog Input Circuit Conversion Phase: Switches Open, Track Phase: Switches Closed

For ac applications, removing high frequency components from the analog input signal is recommended by using an RC low-pass filter on the relevant analog input pins. In applications where harmonic distortion and signal-to-noise ratio are critical, the analog input should be driven from a low impedance source. Large source impedances significantly affect the ac performance of the ADC. This can necessitate the use of an input buffer amplifier. The choice of the op amp is a function of the particular application. Figure 28 and Figure 29 give an example of an ADC front end.

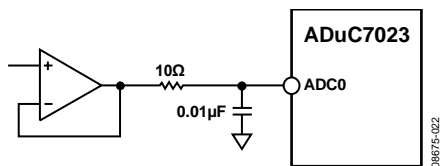


Figure 28. Buffering Single-Ended Differential Input

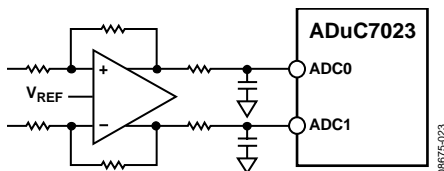


Figure 29. Buffering Differential Inputs

When no amplifier is used to drive the analog input, limit the source impedance to values lower than 1 kΩ. The maximum source impedance depends on the amount of total harmonic distortion (THD) that can be tolerated. The THD increases as the source impedance increases and the performance degrades.

### DRIVING THE ANALOG INPUTS

Internal or external references can be used for the ADC. When operating in differential mode, there are restrictions on the common-mode input signal ( $V_{CM}$ ), which is dependent upon the reference value and supply voltage used to ensure that the

signal remains within the supply rails. Table 27 gives some calculated  $V_{CM}$  minimum and  $V_{CM}$  maximum values.

Table 27.  $V_{CM}$  Ranges

| AV <sub>DD</sub> | V <sub>REF</sub> | V <sub>CM</sub> Min | V <sub>CM</sub> Max | Signal Peak-to-Peak |
|------------------|------------------|---------------------|---------------------|---------------------|
| 3.3 V            | 2.5 V            | 1.25 V              | 2.05 V              | 2.5 V               |
|                  | 2.048 V          | 1.024 V             | 2.276 V             | 2.048 V             |
|                  | 1.25 V           | 0.75 V              | 2.55 V              | 1.25 V              |
| 3.0 V            | 2.5 V            | 1.25 V              | 1.75 V              | 2.5 V               |
|                  | 2.048 V          | 1.024 V             | 1.976 V             | 2.048 V             |
|                  | 1.25 V           | 0.75 V              | 2.25 V              | 1.25 V              |

### CALIBRATION

By default, the factory-set values written to the ADC offset (ADCOF) and gain coefficient registers (ADCGN) yield optimum performance in terms of endpoint errors and linearity for standalone operation of the part (see the Specifications section). If system calibration is required, it is possible to modify the default offset and gain coefficients to improve endpoint errors, but note that any modification to the factory-set ADCOF and ADCGN values can degrade ADC linearity performance.

For system offset error correction, the ADC channel input stage must be tied to AGND. A continuous software ADC conversion loop must be implemented by modifying the value in ADCOF until the ADC result (ADCDAT) reads Code 0 to Code 1. If the ADCDAT value is greater than 1, ADCOF should be decremented until ADCDAT reads Code 0 to Code 1. Offset error correction is done digitally and has a resolution of 0.25 LSB and a range of  $\pm 3.125\%$  of  $V_{REF}$ .

For system gain error correction, the ADC channel input stage must be tied to  $V_{REF}$ . A continuous software ADC conversion loop must be implemented to modify the value in ADCGN until the ADCDAT reads Code 4094 to Code 4095. If the ADCDAT value is less than 4094, ADCGN should be incremented until ADCDAT reads Code 4094 to Code 4095. Similar to the offset calibration, the gain calibration resolution is 0.25 LSB with a range of  $\pm 3\%$  of  $V_{REF}$ .

### TEMPERATURE SENSOR

The ADuC7023 provides a voltage output from an on-chip band gap reference that is proportional to absolute temperature. This voltage output can also be routed through the front-end ADC multiplexer (effectively an additional ADC channel input), facilitating an internal temperature sensor channel, measuring die temperature.

An ADC temperature sensor conversion differs from a standard ADC voltage. The ADC performance specifications do not apply to the temperature sensor.

Chopping of the internal amplifier should be enabled using the TSCON register. To enable this mode, the user must set Bit 0 of TSCON. The user must also take two consecutive ADC readings and average them in this mode.

The ADCCON register must be configured to 0x37A3.  
 To calculate die temperature use the following formula:

$$T - T_{REF} = (V_{ADC} - V_{TREF}) \times K$$

where:

T is the temperature result.

T<sub>REF</sub> is 25°C.

V<sub>ADC</sub> is the average ADC result from two consecutive conversions.

V<sub>TREF</sub> is 1369 mV, which corresponds to T<sub>REF</sub> = 25°C as described in Table 1.

K is the gain of the ADC in temperature sensor mode as determined by characterization data, K = 0.2262°C/mV. This corresponds to 1/V TC specification as shown in Table 1.

Using the default values from Table 1 and without any calibration, this equation becomes

$$T - 25^\circ\text{C} = (V_{ADC} - 1369) \times 0.2262$$

where:

V<sub>ADC</sub> is in millivolts.

For increased accuracy, perform a single point calibration at a controlled temperature value.

For the calculation shown without calibration, (T<sub>REF</sub>, V<sub>TREF</sub>) = (25°C, 1369 mV). The idea of a single point calibration is to use other known (T<sub>REF</sub>, V<sub>TREF</sub>) values to replace the common (25°C, 1369 mV) for every part.

For some users, it is not possible to get such a known pair. For these cases, an ADuC7023 comes with a single point calibration value loaded in the TEMPREF register. For more details on this register, see the TEMPREF Register section.

During production testing of the ADuC7023, the TEMPREF register is loaded with an offset adjustment factor. Each part will have a different value in the TEMPREF register. Using this single point calibration, use the same formula as shown:

$$T - T_{REF} = (V_{ADC} - V_{TREF}) \times K$$

where:

T<sub>REF</sub> is 27°C when using the TEMPREF register method, but is not guaranteed.

T<sub>TREF</sub> can be calculated using the TEMPREF register.

**TSCON Register**

Name: TSCON  
 Address: 0xFFFF0544  
 Default value: 0x00  
 Access: Read/write

**Table 28. TSCON MMR Bit Designations**

| Bit    | Description   |
|--------|---|
| 7 to 1 | Reserved.   |
| 0      | Temperature sensor chop enable bit.<br>This bit is set to 1 to enable chopping of the internal amplifier to the ADC.<br>This bit is cleared to disable chopping.<br>This bit is cleared by default. |

**TEMPREF Register**

Name: TEMPREF  
 Address: 0xFFFF0548  
 Default value: Factory configured  
 Access: Read/write

**Table 29. TEMPREF MMR Bit Designations**

| Bit     | Description  |
|---------|--|
| 15 to 9 | Reserved.  |
| 8       | Temperature reference voltage sign.  |
| 7 to 0  | Temperature sensor offset calibration voltage.<br>To calculate the V <sub>TREF</sub> from the TEMPREF register, perform the following calculation:<br><br>If TEMPREF sign negative, subtract TEMPREF from 2292<br><br>$C_{TREF} = 2292 - TEMPREF[7:0]$ where TEMPREF[8] = 1.<br><br>or<br>If TEMPREF sign positive, add TEMPREF to 2292<br>$C_{TREF} = TEMPREF[7:0] + 2292$ where:<br>TEMPREF[8] = 0.<br><br>Then,<br>$V_{TREF} = (C_{TREF} \times V_{REF}) / 4096 \times 1000$ where:<br>C <sub>TREF</sub> is calculated as above.<br>V <sub>REF</sub> is 2.5 V, internal reference voltage.<br><br>Insert V <sub>TREF</sub> into<br>$T - T_{REF} = (V_{ADC} - V_{TREF}) \times K$ where:<br>T <sub>REF</sub> is 27°C, when using TEMPREF register.<br>V <sub>ADC</sub> is the average ADC result from two consecutive conversions.<br>V <sub>TREF</sub> is calculated as above.<br><br>Note that ADC code value 2292 is a default value when using the TEMPREF register. It is not an exact value and must only be used with the TEMPREF register. |

## BAND GAP REFERENCE

The ADuC7023 provides an on-chip band gap reference of 2.5 V, which can be used for the ADC and DAC. This internal reference also appears on the  $V_{REF}$  pin. When using the internal reference, a 0.47  $\mu$ F capacitor must be connected from the external  $V_{REF}$  pin to AGND to ensure stability and fast response during ADC conversions. This reference can also be connected to an external pin ( $V_{REF}$ ) and used as a reference for other circuits in the system.

An external buffer is required because of the low drive capability of the  $V_{REF}$  output. A programmable option also allows an external reference input on the  $V_{REF}$  pin.

### REFCON Register

|                |   |
|----------------|---|
| Name:          | REFCON  |
| Address:       | 0xFFFF048C  |
| Default value: | 0x00  |
| Access:        | Read/write  |
| Function:      | The band gap reference interface consists of an 8-bit MMR REFCON described in Table 30. |

Table 30. REFCON MMR Bit Designations

| Bit    | Description   |
|--------|---|
| 7 to 1 | Reserved.   |
| 0      | Internal reference output enable.<br>This bit is set by the user to connect the internal 2.5 V reference to the $V_{REF}$ pin. The reference can be used for an external component but needs to be buffered.<br>This bit is cleared by the user to disconnect the reference from the $V_{REF}$ pin. |

To connect an external reference source to the ADuC7023, configure REFCON = 0x01. ADC and the DACs can be configured to use the same or different reference resource. See Table 42.

## NONVOLATILE FLASH/EE MEMORY

The ADuC7023 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.

The 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 ADuC7023, Flash/EE memory technology allows the user to update program code space in-circuit, without needing to replace one-time programmable (OTP) devices at remote operating nodes.

Each part contains a 64 kB array of Flash/EE memory. The lower 62 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:

1. Initial page erase sequence.
2. Read/verify sequence (single Flash/EE).
3. Byte program sequence memory.
4. 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. As indicated in Table 1, the Flash/EE memory endurance qualification is carried out in accordance with JEDEC Retention Lifetime Specification A117 over the industrial temperature range of  $-40^{\circ}$  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 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 the Flash/EE memory is reprogrammed. In addition, note that retention lifetime, based on activation energy of 0.6 eV, derates with  $T_j$  as shown in Figure 30.

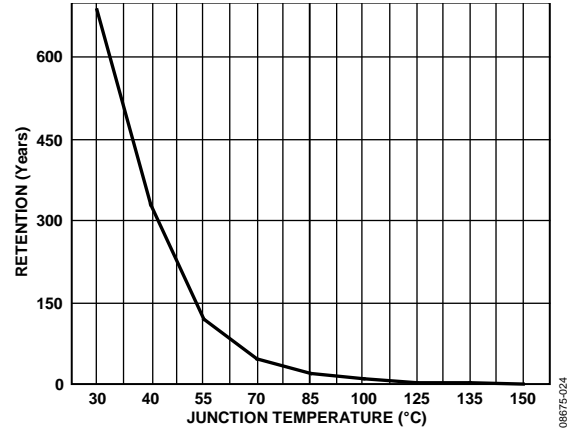


Figure 30. Flash/EE Memory Data Retention

## PROGRAMMING

The 62 kB of Flash/EE memory can be programmed in circuit, using the serial download mode or the provided JTAG mode.

### Downloading (In-Circuit Programming) via I<sup>2</sup>C

The ADuC7023 facilitates code download via the I<sup>2</sup>C port. The parts enter download mode after a reset or power cycle if the BM pin is pulled low through an external 1 k $\Omega$  resistor and Flash Address 0x80014 = 0xFFFFFFFF. Once in download mode, the user can download code to the full 62 kB of Flash/EE memory while the device is in-circuit in its target application hardware. An executable PC I<sup>2</sup>C download is provided as part of the development system for serial downloading via the I<sup>2</sup>C. A USB to I<sup>2</sup>C download dongle can be purchased from Analog Devices, Inc. This board connects to the USB port of a PC and to the I<sup>2</sup>C port of the ADuC7023. The part number is USB-I2C/LIN-CONV-Z.

The AN-806 Application Note describes the protocol for serial downloading via the I<sup>2</sup>C in more detail.

### JTAG Access

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

The JTAG interface is active as long as the part is not in download mode; that is, the P0.0/BM pin = 0 and Address 0x80014 = 0xFFFFFFFF at reset.

When debugging, user code must not write to the bits in GP0CON/GP0DAT corresponding to P0.0/P0.1/P0.2 and P0.3 pins. If user code changes the state of any of these pins, JTAG debug pods are not able to connect to the ADuC7023. In case this happens, the user should have a function in code that can be called externally to mass erase the part. Alternatively, the user should ensure that Flash Address 0x80014 is erased to allow erasing of the part through the I<sup>2</sup>C interface.

## SECURITY

The 62 kB of Flash/EE memory available to the user can be read and write protected.

Bit 31 of the FEEPRO/FEEHIDE MMR (see Table 34) protects the 62 kB from being read through JTAG programming mode. The other 31 bits of this register protect writing to the flash memory. Each bit protects four pages, that is, 2 kB. Write protection is activated for all types of access.

### Three Levels of Protection

Protection can be set and removed by writing directly into FEEHIDE MMR. This protection does not remain after reset.

Protection can be set by writing into FEEPRO MMR. It only takes effect after a save protection command (0x0C) and a reset. The FEEPRO MMR is protected by a key to avoid direct access. The key is saved once and must be entered again to modify FEEPRO. A mass erase sets the key back to 0xFFFF but also erases all the user code.

Flash can be permanently protected by using the FEEPRO MMR and a particular value of key: 0xDEADDEAD. Entering the key again to modify the FEEPRO register is not allowed.

### Sequence to Write the Key

1. Write the bit in FEEPRO corresponding to the page to be protected.
2. Enable key protection by setting Bit 6 of FEEMOD (Bit 5 must equal 0).
3. Write a 32-bit key in FEEADR, FEEDAT.
4. Run the write key command 0x0C in FEECON; wait for the read to be successful by monitoring FEESTA.
5. Reset the part.

**Table 31. FEESTA MMR Bit Designations**

| Bit    | Description  |
|--------|--|
| 7 to 6 | Reserved.  |
| 5      | Reserved.  |
| 4      | Reserved.  |
| 3      | Flash interrupt status bit.<br>This bit is set automatically when an interrupt occurs, that is, when a command is complete and the Flash/EE interrupt enable bit in the FEEMOD register is set.<br>This bit is cleared when reading FEESTA register. |
| 2      | Flash/EE controller busy.<br>This bit is set automatically when the controller is busy.<br>This bit is cleared automatically when the controller is not busy.  |
| 1      | Command fail.<br>This bit is set automatically when a command is not completed.<br>This bit is cleared automatically when reading FEESTA register.   |
| 0      | Command pass.<br>This bit is set by the MicroConverter when a command is completed.<br>This bit is cleared automatically when reading the FEESTA register.   |

To remove or modify the protection, the same sequence is used with a modified value of FEEPRO. If the key chosen is the value 0xDEAD, the memory protection cannot be removed. Only a mass erase unprotects the part, but it also erases all user code.

The sequence to write the key is illustrated in the following example (this protects writing Page 4 to Page 7 of the Flash):

```
FEEPRO=0xFFFFFFFF; //Protect Page 4 to
                        Page 7
```

```
FEEMOD=0x48; //Write key enable
FEEADR=0x1234; //16 bit key value
FEEDAT=0x5678; //16 bit key value
FEECON= 0x0C; //Write key command
```

Follow the same sequence to protect the part permanently with FEEADR = 0xDEAD and FEEDAT = 0xDEAD.

## FLASH/EE CONTROL INTERFACE

Serial and JTAG programming use the Flash/EE control interface, which includes the eight MMRs outlined in this section.

### FEESTA Register

Name: FEESTA

Address: 0xFFFFF800

Default value: 0x20

Access: Read

Function: FEESTA is a read-only register that reflects the status of the flash control interface as described in Table 31.

**FEEMOD Register**

|                |  |
|----------------|--|
| Name:          | FEEMOD   |
| Address:       | 0xFFFFF804   |
| Default value: | 0x0000   |
| Access:        | Read/write   |
| Function:      | FEEMOD sets the operating mode of the flash control interface. Table 32 shows FEEMOD MMR bit designations. |

**Table 32. FEEMOD MMR Bit Designations**

| Bit     | Description  |
|---------|--|
| 15 to 9 | Reserved.  |
| 8       | Reserved. Always set this bit to 0.  |
| 7 to 5  | Reserved. Always set this bit to 0 except when writing keys. See the Sequence to Write the Key section.  |
| 4       | Flash/EE interrupt enable.<br>This bit is set by the user to enable the Flash/EE interrupt. The interrupt occurs when a command is complete.<br>This bit is cleared by the user to disable the Flash/EE interrupt. |
| 3       | Erase/write command protection.<br>This bit is set by the user to enable the erase and write commands.<br>This bit is cleared to protect the Flash/EE against erase/write command.                                 |
| 2 to 0  | Reserved. Always set this bit to 0.  |

**FEECON Register**

|                |  |
|----------------|--|
| Name:          | FEECON   |
| Address:       | 0xFFFFF808   |
| Default value: | 0x07   |
| Access:        | Read/write   |
| Function:      | FEECON is an 8-bit command register. The commands are described in Table 33. |

**Table 33. 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 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 by FEEADR. This operation takes approximately 24 ms.   |
| 0x04 <sup>1</sup> | Single verify | Compare the contents of the location pointed by FEEADR to the data in FEEDAT. The result of the comparison is returned in FEESTA Bit 1.   |
| 0x05 <sup>1</sup> | Single erase  | Erase the page indexed by FEEADR.   |
| 0x06 <sup>1</sup> | Mass erase    | Erase 62 kB of user space. The 2 kB of kernel are protected. This operation takes 2.48 sec. 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     | Give a signature of the 64 kB of Flash/EE in the 24-bit FEESIGN MMR. This operation takes 32,778 clock cycles.  |
| 0x0C              | Protect       | This command can run one time only. The value of FEEPRO is saved and removed only with a mass erase (0x06) or the key (FEEADR/FEEDAT).  |



| Code | Command  | Description                        |
|------|----------|------------------------------------|
| 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**

Name: FEEDAT  
Address: 0xFFFFF80C  
Default value: 0XXXXX  
Access: Read/write  
Function: FEEDAT is a 16-bit data register.

#### **FEEADR Register**

Name: FEEADR  
Address: 0xFFFFF810  
Default value: 0x0000  
Access: Read/write  
Function: FEEADR is another 16-bit address register.

#### **FEESIGN Register**

Name: FEESIGN  
Address: 0xFFFFF818  
Default value: 0FFFFFFF  
Access: Read  
Function: FEESIGN is a 24-bit code signature.

#### **FEEPRO Register**

Name: FEEPRO  
Address: 0xFFFFF81C  
Default value: 0x00000000  
Access: Read/write  
Function: FEEPRO MMR provides protection following a subsequent reset of the MMR. It requires a software key (see Table 34).

#### **FEEHIDE Register**

Name: FEEHIDE  
Address: 0xFFFFF820  
Default value: 0FFFFFFF  
Access: Read/write  
Function: FEEHIDE MMR provides immediate protection. It does not require any software key. The protection settings in FEEHIDE are cleared by a reset (see Table 34).

**Table 34. FEEPRO and FEEHIDE MMR Bit Designations**

| Bit     | Description   |
|---------|---|
| 31      | Read protection.<br>This bit is cleared by the user to protect the code<br>This bit is set by the user to allow reading the code.   |
| 30 to 0 | Write protection for Page 123 to Page 120, Page 119 to Page 116, and Page 0 to Page 3.<br>This bit is cleared by the user to protect the pages in writing.<br>This bit is set by the user to allow writing the pages. |

#### **Command Sequence for Executing a Mass Erase**

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

**EXECUTION TIME FROM SRAM AND FLASH/EE**

**Execution from SRAM**

Fetching instructions from SRAM takes one clock cycle because the access time of the SRAM is 2 ns, and a clock cycle is 22 ns minimum. However, if the instruction involves reading or writing data to memory, one extra cycle must be added if the data is in SRAM (or three cycles if the data is in Flash/EE); one cycle to execute the instruction and two cycles to obtain the 32-bit data from Flash/EE. A control flow instruction (a branch instruction, for example) takes one cycle to fetch but also takes two cycles to fill the pipeline with the new instructions.

**Execution from Flash/EE**

Because the Flash/EE width is 16 bits and the access time for 16-bit words is 22 ns, execution from Flash/EE cannot be completed in one cycle (as can be done from SRAM when the CD bit = 0). Also, some dead times are needed before accessing data for any value of CD bits.

In ARM mode, where instructions are 32 bits, two cycles are needed to fetch any instruction when CD = 0. In thumb mode, where instructions are 16 bits, one cycle is needed to fetch any instruction.

Timing is identical in both modes when executing instructions that involve using the Flash/EE for data memory. If the instruction to be executed is a control flow instruction, an extra cycle is needed to decode the new address of the program counter, and then four cycles are needed to fill the pipeline. A data processing instruction involving only the core register does not require any extra clock cycles. However, if it involves data in Flash/EE, an extra clock cycle is needed to decode the address of the data, and two cycles are needed to get the 32-bit data from Flash/EE. An extra cycle must also be added before fetching another instruction. Data transfer instructions are more complex and are summarized in Table 35.

**Table 35. Execution Cycles in ARM/Thumb Mode**

| Instructions     | Fetch Cycles | Dead Time      | Data Access                | Dead Time      |
|------------------|--------------|----------------|----------------------------|----------------|
| LD <sup>1</sup>  | 2/1          | 1              | 2                          | 1              |
| LDH              | 2/1          | 1              | 1                          | 1              |
| LDM/PUSH         | 2/1          | N <sup>2</sup> | 2 × N <sup>2</sup>         | N <sup>1</sup> |
| STR <sup>1</sup> | 2/1          | 1              | 2 × 20 ns                  | 1              |
| STRH             | 2/1          | 1              | 20 ns                      | 1              |
| STRM/POP         | 2/1          | N <sup>1</sup> | 2 × N × 20 ns <sup>1</sup> | N <sup>1</sup> |

<sup>1</sup> The SWAP instruction combines an LD and STR instruction with only one fetch, giving a total of eight cycles + 40 ns.

<sup>2</sup> N is the number of data to load or store in the multiple load/store instruction (1 < N ≤ 16).

**RESET AND REMAP**

The ARM exception vectors are all situated at the bottom of the memory array, from Address 0x00000000 to Address 0x00000020 as shown in Figure 31.

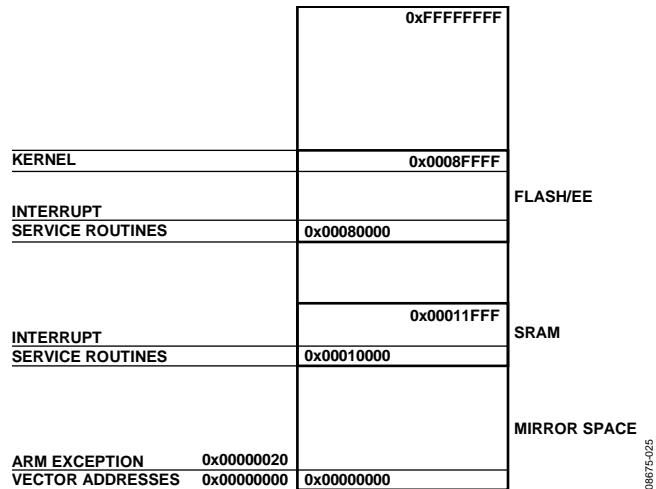


Figure 31. Remap for Exception Execution

By default, and after any reset, the Flash/EE is mirrored at the bottom of the memory array. The remap function allows the programmer to mirror the SRAM at the bottom of the memory array, which facilitates execution of exception routines from SRAM instead of from Flash/EE. This means exceptions are executed twice as fast, being executed in 32-bit ARM mode with 32-bit wide SRAM instead of 16-bit wide Flash/EE memory.

**Remap Operation**

When a reset occurs on the ADuC7023, execution automatically starts in factory programmed, internal configuration code. This kernel is hidden and cannot be accessed by user code. If the part is in normal mode (BM pin is high), it executes the power-on configuration routine of the kernel and then jumps to the reset vector address, 0x00000000, to execute the reset exception routine of the user.

Because the Flash/EE is mirrored at the bottom of the memory array at reset, the reset interrupt routine must always be written in Flash/EE.

The remap is done from Flash/EE by setting Bit 0 of the Remap register. Caution must be taken to execute this command from Flash/EE above Address 0x00080020, and not from the bottom of the array because this is replaced by the SRAM.

This operation is reversible. The Flash/EE can be remapped at Address 0x00000000 by clearing Bit 0 of the Remap MMR. Caution must again be taken to execute the remap function from outside the mirrored area. Any type of reset remaps the Flash/EE memory at the bottom of the array.

**REMAP Register**

Name: REMAP  
 Address: 0xFFFF0220  
 Default value: 0x00  
 Access: Read/write

**Table 36. REMAP MMR Bit Designations**

| Bit    | Name  | Description  |
|--------|-------|--|
| 7 to 5 |       | Reserved.  |
| 4      |       | Read-only bit. Indicates the size of the Flash/EE memory available. If this bit is set, only 32 kB of Flash/EE memory is available.  |
| 3      |       | Read-only bit. Indicates the size of the SRAM memory available. If this bit is set, only 4 kB of SRAM is available.  |
| 2 to 1 | JTAFO | Read only bits. See the P0.0/BM description for further details. The kernel sets these bits to [11] if BM = 0 and 0x80014 ≠ 0xFFFFFFFF at reset.<br>If these bits are set to [00], then P0.1/P0.2/P0.3 are configured as JTAG pins. P0.1/P0.2 cannot be used as GPIO. P0.3 can be used as GPIO, but this disables JTAG access.<br>If these bits are set to [1x], then P0.1/P0.2/P0.3 are configured as GPIO pins. P0.1/P0.2/P0.3 can also be used as JTAG, but JTAG access is disabled if they are used as GPIO. These bits are configured by the kernel after any reset sequence and depend on the state of P0.0 and the value at Address 0x80014 during the last reset sequence. |
| 0      | Remap | Remap bit.<br>This bit is set by the user to remap the SRAM to Address 0x00000000.<br>This bit is cleared automatically after reset to remap the Flash/EE memory to Address 0x00000000.  |

**Reset Operation**

There are four kinds of reset: external, power-on, watchdog expiration, and software force. The RSTSTA register indicates the source of the last reset, and RSTCLR allows clearing of the RSTSTA register. These registers can be used during a reset exception service routine to identify the source of the reset. If RSTSTA is null, the reset is external.

The RSTCFG register allows different peripherals to retain their state after a watchdog or software reset.

**RSTSTA Register**

Name: RSTSTA  
 Address: 0xFFFF0230  
 Default value: 0x01  
 Access: Read/write

**Table 37. RSTSTA MMR Bit Designations**

| Bit    | Description  |
|--------|--|
| 7 to 3 | Reserved.  |
| 2      | Software reset.<br>This bit is set by the user to force a software reset. This bit is cleared by setting the corresponding bit in RSTCLR.          |
| 1      | Watchdog timeout.<br>This bit is set automatically when a watchdog timeout occurs. This bit is cleared by setting the corresponding bit in RSTCLR. |
| 0      | Power-on reset.<br>This bit is set automatically when a power-on reset occurs. This bit is cleared by setting the corresponding bit in RSTCLR.     |

**RSTCLR Register**

Name: RSTCLR  
 Address: 0xFFFF0234  
 Default value: 0x00  
 Access: Write  
 Function: Note that to clear the RSTSTA register, users must write the Value 0x07 to the RSTCLR register.

**RSTCFG Register**

Name: RSTCFG  
 Address: 0xFFFF024C  
 Default value: 0x00  
 Access: Read/write

**Table 38. RSTCFG MMR Bit Designations**

| <b>Bit</b> | <b>Description</b>   |
|------------|--|
| 7 to 3     | Reserved. Always set to 0.   |
| 2          | This bit is set to 1 to configure the DAC outputs to retain their state after a watchdog or software reset. This bit is cleared for the DAC pins and registers to return to their default state. |
| 1          | Reserved. Always set to 0.   |
| 0          | This bit is set to 1 to configure the GPIO pins to retain their state after a watchdog or software reset. This bit is cleared for the GPIO pins and registers to return to their default state.  |

**RSTKEY1 Register**

Name: RSTKEY1  
 Address: 0xFFFF0248  
 Default value: 0xXX  
 Access: Write

**RSTKEY2 Register**

Name: RSTKEY2  
 Address: 0xFFFF0250  
 Default value: 0xXX  
 Access: Write

**Table 39. RSTCFG Write Sequence**

| <b>Name</b> | <b>Code</b> |
|-------------|-------------|
| RSTKEY1     | 0x76        |
| RSTCFG      | User value  |
| RSTKEY2     | 0xB1        |

## OTHER ANALOG PERIPHERALS

### DAC

The ADuC7023 incorporates four, 12-bit voltage output DACs on chip. Each DAC has a rail-to-rail voltage output buffer capable of driving 5 k $\Omega$ /100 pF.

Each DAC has two selectable ranges: 0 V to  $V_{REF}$  (internal band gap 2.5 V reference) and 0 V to  $AV_{DD}$ .

The signal range is 0 V to  $AV_{DD}$ .

By setting RSTCFG Bit 2, the DAC output pins can retain their state during a watchdog or software reset.

#### MMRs Interface

Each DAC is independently configurable through a control register and a data register. These two registers are identical for the four DACs. Only DAC0CON (see Table 40) and DAC0DAT (see Table 41) are described in detail in this section.

#### DACxCON Registers

| Name    | Address    | Default Value | Access |
|---------|------------|---------------|--------|
| DAC0CON | 0xFFFF0600 | 0x00          | R/W    |
| DAC1CON | 0xFFFF0608 | 0x00          | R/W    |
| DAC2CON | 0xFFFF0610 | 0x00          | R/W    |
| DAC3CON | 0xFFFF0618 | 0x00          | R/W    |

Table 40. DAC0CON MMR Bit Designations

| Bit    | Value | Name   | Description   |
|--------|-------|--------|---|
| 7      |       |        | Reserved.   |
| 6      |       | DACBY  | This bit is set to bypass the DAC output buffer.<br>This bit is cleared to enable the DAC output buffer.  |
| 5      |       | DACCLK | DAC update rate.<br>This bit is set by the user to update the DAC using Timer1.<br>This bit is cleared by the user to update the DAC using HCLK (core clock). |
| 4      |       | DACCLR | DAC clear bit.<br>This bit is set by the user to enable normal DAC operation.<br>This bit is cleared by the user to reset data register of the DAC to 0.      |
| 3      |       |        | Reserved. This bit remains at 0.  |
| 2      |       |        | Reserved. This bit remains at 0.  |
| 1 to 0 |       |        | DAC range bits.   |
|        | 00    |        | Power-down mode. The DAC output is in tristate.   |
|        | 01    |        | Reserved.   |
|        | 10    |        | 0 V to $V_{REF}$ (2.5 V) range.   |
|        | 11    |        | 0 V to $AV_{DD}$ range.   |

#### DACxDAT Registers

| Name    | Address    | Default Value | Access |
|---------|------------|---------------|--------|
| DAC0DAT | 0xFFFF0604 | 0x00000000    | R/W    |
| DAC1DAT | 0xFFFF060C | 0x00000000    | R/W    |
| DAC2DAT | 0xFFFF0614 | 0x00000000    | R/W    |
| DAC3DAT | 0xFFFF061C | 0x00000000    | R/W    |

Table 41. DAC0DAT MMR Bit Designations

| Bit      | Description           |
|----------|-----------------------|
| 31 to 28 | Reserved.             |
| 27 to 16 | 12-bit data for DAC0. |
| 15 to 0  | Reserved.             |

#### Using the DACs

The on-chip DAC architecture consists of a resistor string DAC followed by an output buffer amplifier. The functional equivalent is shown in Figure 32.

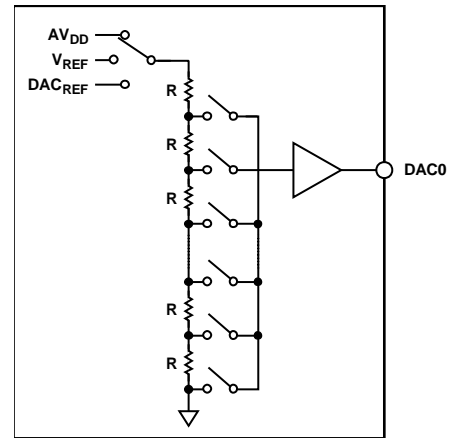


Figure 32. DAC Structure

As illustrated in Figure 32, the reference source for each DAC is user-selectable in software. It can be either  $AV_{DD}$  or  $V_{REF}$ . In 0-to- $AV_{DD}$  mode, the DAC output transfer function spans from 0 V to the voltage at the  $AV_{DD}$  pin. In 0-to- $V_{REF}$  mode, the DAC output transfer function spans from 0 V to the internal 2.5 V reference,  $V_{REF}$ .

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  $AV_{DD}$  and ground. Moreover, the DAC linearity specification (when driving a 5 k $\Omega$  resistive load to ground) is guaranteed through the full transfer function except Code 0 to Code 100, and, in 0-to- $AV_{DD}$  mode only, Code 3995 to Code 4095.

Linearity degradation near ground and  $V_{DD}$  is caused by saturation of the output amplifier, and a general representation of its effects (neglecting offset and gain error) is illustrated in Figure 33. The dotted line in Figure 33 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. Figure 33 represents a transfer function in 0-to- $AV_{DD}$

mode only. In 0-to- $V_{REF}$  mode (with  $V_{REF} < AV_{DD}$ ), the lower nonlinearity is similar. However, the upper portion of the transfer function follows the ideal line right to the end ( $V_{REF}$  in this case, not  $AV_{DD}$ ), showing no signs of endpoint linearity errors.

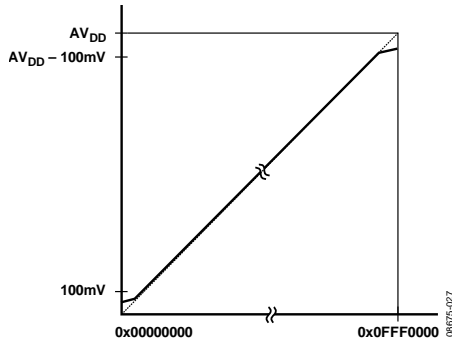


Figure 33. Endpoint Nonlinearities Due to Amplifier Saturation

The endpoint nonlinearities conceptually illustrated in Figure 33 get worse as a function of output loading. Most of the ADuC7023 data sheet specifications 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 of Figure 33 become larger, respectively. With larger current demands, this can significantly limit output voltage swing.

**References to ADC and the DACs**

ADC and DACs can be configured to use internal  $V_{REF}$  or an external reference as a reference source. Internal  $V_{REF}$  must work with an external 0.47 μF capacitor. Note that if an external reference is used, the DACs will no longer meet offset and gain specifications. If an external reference is required for the ADC, then the DACs should be configured to use the 0 to  $AV_{DD}$  range.

Table 42. Reference Source Selection for ADC and DAC

| REFCON Bit 0 | DACxCON[1:0] | Description   |
|--------------|--------------|---|
| 0            | 00           | ADC works with external reference. DACs power down.   |
| 0            | 01           | Reserved.   |
| 0            | 10           | Reserved.   |
| 0            | 11           | ADC works with external reference. DACs work with internal $AV_{DD}$ .  |
| 1            | 00           | ADC works with internal $V_{REF}$ . DACs power down.  |
| 1            | 01           | ADC and DACs work with an external reference. The external reference must be capable of overdriving the internal reference. |
| 1            | 10           | ADC and DACs work with internal $V_{REF}$ .   |
| 1            | 11           | ADC works with internal $V_{REF}$ . DACs work with internal $AV_{DD}$ .   |

**Configuring DAC Buffers in Op Amp Mode**

In op amp mode, the DAC output buffers are used as an op amp with the DAC itself disabled.

If DACBCFG Bit 0 is set, ADC0 is the positive input to the op amp, ADC1 is the negative input, and DAC0 is the output. In this mode, the DAC should be powered down by clearing Bit 0 and Bit 1 of DAC0CON.

If DACBCFG Bit 1 is set, ADC2 is the positive input to the op amp, ADC3 is the negative input, and DAC1 is the output. In this mode, the DAC should be powered down by clearing Bit 0 and Bit 1 of DAC1CON.

If DACBCFG Bit 2 is set, ADC4 is the positive input to the op amp, ADC5 is the negative input, and DAC2 is the output. In this mode, the DAC should be powered down by clearing Bit 0 and Bit 1 of DAC2CON.

If DACBCFG Bit 3 is set, ADC8 is the positive input to the op amp, ADC9 is the negative input, and DAC3 is the output. In this mode, the DAC should be powered down by clearing Bit 0 and Bit 1 of DAC3CON.

**DACBCFG Register**

Name: DACBCFG  
 Address: 0xFFFF0654  
 Default value: 0x00  
 Access: Read/write

Table 43. DACBCFG MMR Bit Designations

| Bit    | Description   |
|--------|---|
| 7 to 4 | Reserved. Always set to 0.  |
| 3      | This bit is set to 1 to configure DAC3 output buffer in op amp mode. This bit is cleared for the DAC buffer to operate as normal. |
| 2      | This bit is set to 1 to configure DAC2 output buffer in op amp mode. This bit is cleared for the DAC buffer to operate as normal. |
| 1      | This bit is set to 1 to configure DAC1 output buffer in op amp mode. This bit is cleared for the DAC buffer to operate as normal. |
| 0      | This bit is set to 1 to configure DAC0 output buffer in op amp mode. This bit is cleared for the DAC buffer to operate as normal. |

**DACBKEY0 Register**

Name: DACBKEY0  
 Address: 0xFFFF0650  
 Default value: 0x0000  
 Access: Write

**DACBKEY1 Register**

Name: DACBKEY1  
 Address: 0xFFFF0658  
 Default value: 0x0000  
 Access: Write

**Table 44. DACBCFG Write Sequence**

| Name     | Code       |
|----------|------------|
| DACBKEY0 | 0x9A       |
| DACBCFG  | User value |
| DACBKEY1 | 0x0C       |

**POWER SUPPLY MONITOR**

The power supply monitor regulates the IOV<sub>DD</sub> supply on the ADuC7023. It indicates when the IOV<sub>DD</sub> supply pin drops below a supply trip point. The monitor function is controlled via the PSMCON register. If enabled in the IRQEN or FIQEN register, the monitor interrupts the core using the PSMI bit in the PSMCON MMR. This bit is immediately cleared when CMP goes high.

This monitor function allows the user to save working registers to avoid possible data loss due to low supply or brownout conditions. It also ensures that normal code execution does not resume until a safe supply level has been established.

**PSMCON Register**

Name: PSMCON  
 Address: 0xFFFF0440  
 Default value: 0x0008  
 Access: Read/write

**Table 45. PSMCON MMR Bit Descriptions**

| Bit | Name  | Description   |
|-----|-------|---|
| 3   | CMP   | Comparator bit. This is a read-only bit that directly reflects the state of the comparator. Read 1 indicates the IOV <sub>DD</sub> supply is above its selected trip point, or the PSM is in power-down mode. Read 0 indicates the IOV <sub>DD</sub> supply is below its selected trip point. This bit should be set before leaving the interrupt service routine.            |
| 2   | TP    | Trip point selection bits.<br>0 = 2.79 V.<br>1 = reserved.  |
| 1   | PSMEN | Power supply monitor enable bit. This bit is set to 1 to enable the power supply monitor circuit. This bit is cleared to 0 to disable the power supply monitor circuit.   |
| 0   | PSMI  | Power supply monitor interrupt bit. This bit is set high by the MicroConverter once CMP goes low, indicating low I/O supply. The PSMI bit can be used to interrupt the processor. Once CMP returns high, the PSMI bit can be cleared by writing a 1 to this location. A 0 write has no effect. There is no timeout delay; PSMI can be immediately cleared once CMP goes high. |

**COMPARATOR**

The ADuC7023 integrates voltage comparators. The positive input is multiplexed with ADC2, and the negative input has two options: ADC3 or DAC0. The output of the comparator can be configured to generate a system interrupt, be routed directly to the programmable logic array, start an ADC conversion, or be on an external pin, COMP<sub>OUT</sub>, as shown in Figure 34.

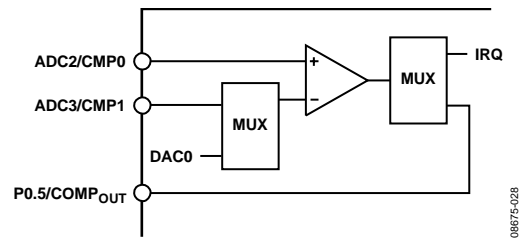


Figure 34. Comparator

**Hysteresis**

Figure 35 shows how the input offset voltage and hysteresis terms are defined. Input offset voltage ( $V_{OS}$ ) is the difference between the center of the hysteresis range and the ground level. This can either be positive or negative. The hysteresis voltage ( $V_H$ ) is  $\frac{1}{2}$  the width of the hysteresis range.

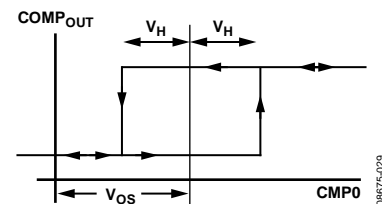


Figure 35. Comparator Hysteresis Transfer Function

**Comparator Interface**

The comparator interface consists of a 16-bit MMR, CMPCON, which is described in Table 46.

**CMPCON Register**

Name: CMPCON  
 Address: 0xFFFF0444  
 Default value: 0x0000  
 Access: Read/write

**Table 46. CMPCON MMR Bit Descriptions**

| Bit      | Value                | Name    | Description   |
|----------|----------------------|---------|---|
| 15 to 11 |                      |         | Reserved.   |
| 10       |                      | COMPEN  | Comparator enable bit.<br>This bit is set by the user to enable the comparator.<br>This bit is cleared by the user to disable the comparator.   |
| 9 to 8   | 00<br>01<br>10<br>11 | COMPIN  | Comparator negative input select bits.<br>AV <sub>DD</sub> /2.<br>ADC3 input.<br>DAC0 output.<br>Reserved.  |
| 7 to 6   | 00<br>01<br>10<br>11 | CMPOC   | Comparator output configuration bits.<br>Reserved.<br>Reserved.<br>Output on COMP <sub>OUT</sub> .<br>IRQ.  |
| 5        |                      | COMPOL  | Comparator output logic state bit. When low, the comparator output is high if the positive input (CMP0) is above the negative input (CMP1). When high, the comparator output is high if the positive input is below the negative input. |
| 4 to 3   | 00<br>11<br>01/10    | COMPRES | Response time.<br>5 μs response time typical for large signals (2.5 V differential).<br>17 μs response time typical for small signals (0.65 mV differential).<br>3 μs typical.<br>Reserved.   |
| 2        |                      | CMPHYST | Comparator hysteresis bit.<br>This bit is set by the user to have a hysteresis of about 7.5 mV.<br>This bit is cleared by the user to have no hysteresis.   |
| 1        |                      | CMPORI  | Comparator output rising edge interrupt.<br>This bit is set automatically when a rising edge occurs on the monitored voltage (CMP0).<br>This bit is cleared by the user by writing a 1 to this bit.                                     |
| 0        |                      | CMPOFI  | Comparator output falling edge interrupt.<br>This bit is set automatically when a falling edge occurs on the monitored voltage (CMP0).<br>This bit is cleared by user.  |



**OSCILLATOR AND PLL—POWER CONTROL**

**Clocking System**

Each ADuC7023 integrates a 32.768 kHz ± 3% oscillator, a clock divider, and a PLL. The PLL locks onto a multiple (1275) of the internal oscillator or an external 32.768 kHz crystal to provide a stable 41.78 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<sup>CD</sup>, is referred to as HCLK. The default core clock is the PLL clock divided by 8 (CD = 3) or 5.22 MHz. The core clock frequency can also come from an external clock on the ECLK pin as described in Figure 36.

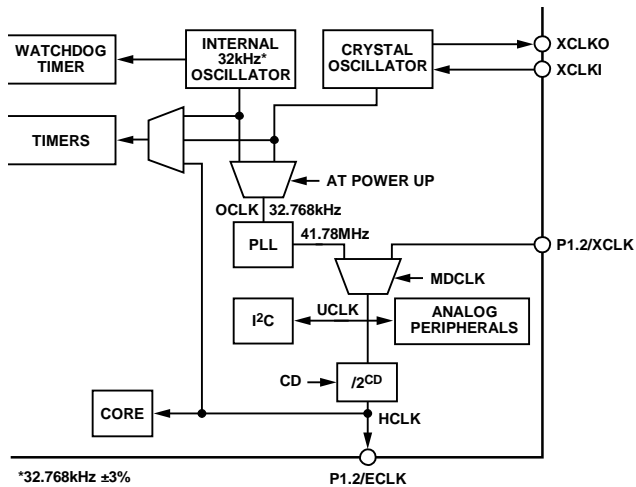


Figure 36. Clocking System

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

In noisy environments, noise can couple to the external crystal pins, and PLL may quickly lose lock. A PLL interrupt is provided in the interrupt controller. The core clock is immediately halted, and this interrupt is only serviced when the lock is restored.

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

**Power Control System**

A choice of operating modes is available on the ADuC7023. Table 47 describes what part is powered on in the different modes and indicates the power-up time.

Table 48 gives some typical values of 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 current consumption of the regulator and other parts on the test board where these values are measured.

Table 47. Operating Modes

| Mode   | Core | Peripherals | PLL | XTAL/T2/T3 | IRQ0 to IRQ3 | Start-Up/Power-On Time           |
|--------|------|-------------|-----|------------|--------------|----------------------------------|
| Active | Yes  | X           | X   | X          | X            | 66 ms at CD = 0                  |
| Pause  |      | X           | X   | X          | X            | 230 ns at CD = 0; 3 μs at CD = 7 |
| Nap    |      |             | X   | X          | X            | 283 ns at CD = 0; 3 μs at CD = 7 |
| Sleep  |      |             |     | X          | X            | 1.23 ms                          |
| Stop   |      |             |     |            | X            | 1.45 ms                          |

X = don't care.

Table 48. Typical Current Consumption at 25°C in mA

| PC[2:0] | Mode   | CD = 0 | CD = 1 | CD = 2 | CD = 3 | CD = 4 | CD = 5 | CD = 6 | CD = 7 |
|---------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| 000     | Active | 28     | 17     | 12     | 11     | 9.3    | 7.5    | 7.2    | 7      |
| 001     | Pause  | 14     | 9      | 7.6    | 5.7    | 4.8    | 4.6    | 4.6    | 4.6    |
| 010     | Nap    | 5      | 4.5    | 4.5    | 4.5    | 4.5    | 4.5    | 4.5    | 4.5    |
| 011     | Sleep  | 0.23   | 0.23   | 0.23   | 0.23   | 0.23   | 0.23   | 0.23   | 0.23   |
| 100     | Stop   | 0.23   | 0.23   | 0.23   | 0.23   | 0.23   | 0.23   | 0.23   | 0.23   |

**MMRs and Keys**

The operating mode, clocking mode, and programmable clock divider are controlled via three MMRs, PLLCON (see Table 49) and POWCONx. PLLCON controls the operating mode of the clock system, POWCON0 controls the core clock frequency and the power-down mode, POWCON1 controls the clock frequency to I<sup>2</sup>C and SPI.

To prevent accidental programming, a certain sequence has to be followed to write to the PLLCON and POWCONx registers.

**PLLKEY1 Register**

Name: PLLKEY1  
 Address: 0xFFFF0410  
 Default value: 0XXXXX  
 Access: Write

**PLLKEY2 Register**

Name: PLLKEY2  
 Address: 0xFFFF0418  
 Default value: 0XXXXX  
 Access: Write

**PLLCON Register**

Name: PLLCON  
 Address: 0xFFFF0414  
 Default value: 0x21  
 Access: Read/write

**Table 49. PLLCON MMR Bit Designations**

| Bit    | Value | Name  | Description  |
|--------|-------|-------|--|
| 7 to 6 |       |       | Reserved.  |
| 5      |       | OSEL  | 32 kHz PLL input selection. This bit is set by the user to select the internal 32 kHz oscillator. This bit is set by default. This bit is cleared by the user to select the external 32 kHz crystal. |
| 4 to 2 |       |       | Reserved.  |
| 1 to 0 |       | MDCLK | Clocking modes.  |
|        | 00    |       | Reserved.  |
|        | 01    |       | PLL default configuration.   |
|        | 10    |       | Reserved.  |
|        | 11    |       | External clock on Pin 33 (40-lead LFCSP)/Pin 25 (32-lead LFCSP).   |

**Table 50. PLLCON Write Sequence**

| Name    | Code       |
|---------|------------|
| PLLKEY1 | 0xAA       |
| PLLCON  | User value |
| PLLKEY2 | 0x55       |

**POWKEY1 Register**

Name: POWKEY1  
 Address: 0xFFFF0404  
 Default value: 0XXXXX  
 Access: Write  
 Function: POWKEY1 prevents accidental programming to POWCON0.

**POWKEY2 Register**

Name: POWKEY2  
 Address: 0xFFFF040C  
 Default value: 0XXXXX  
 Access: Write  
 Function: POWKEY2 prevents accidental programming to POWCON0.

**POWCON0 Register**

Name: POWCON0  
 Address: 0xFFFF0408  
 Default value: 0x00  
 Access: Read/write

**Table 51. POWCON0 MMR Bit Designations**

| Bit    | Value  | Name | Description                                    |
|--------|--------|------|--|
| 7      |        |      | Reserved.                                      |
| 6 to 4 |        | PC   | Operating modes.                               |
|        | 000    |      | Active mode.                                   |
|        | 001    |      | Pause mode.                                    |
|        | 010    |      | Nap.   |
|        | 011    |      | Sleep mode. IRQ0 to IRQ3 can wake up the part. |
|        | 100    |      | Stop mode. IRQ0 to IRQ3 can wake up the part.  |
|        | Others |      | Reserved.                                      |
| 3      |        |      | Reserved.                                      |
| 2 to 0 |        | CD   | CPU clock divider bits.                        |
|        | 000    |      | 41.78 MHz.                                     |
|        | 001    |      | 20.89 MHz.                                     |
|        | 010    |      | 10.44 MHz.                                     |
|        | 011    |      | 5.22 MHz.                                      |
|        | 100    |      | 2.61 MHz.                                      |
|        | 101    |      | 1.31 MHz.                                      |
|        | 110    |      | 653 kHz.                                       |
|        | 111    |      | 326 kHz.                                       |

Table 52. POWCON0 Write Sequence

| Name    | Code       |
|---------|------------|
| POWKEY1 | 0x01       |
| POWCON0 | User value |
| POWKEY2 | 0xF4       |

**POWKEY3 Register**

|                |   |
|----------------|---|
| Name:          | POWKEY3   |
| Address:       | 0xFFFF0434  |
| Default value: | 0XXXXX  |
| Access:        | Write   |
| Function:      | POWKEY3 prevents accidental programming to POWCON1. |

**POWKEY4 Register**

|               |   |
|---------------|---|
| Name          | POWKEY4   |
| Address       | 0xFFFF043C  |
| Default Value | 0XXXXX  |
| Access        | Write   |
| Function:     | POWKEY4 prevents accidental programming to POWCON1. |

**POWCON1 Register**

|                |            |
|----------------|------------|
| Name:          | POWCON1    |
| Address:       | 0xFFFF0438 |
| Default value: | 0x0004     |
| Access:        | Read/write |

Table 53. POWCON1 MMR Bit Designations

| Bit      | Value                | Name       | Description   |
|----------|----------------------|------------|---|
| 15 to 12 |                      |            | Reserved.   |
| 11       |                      | PWMPO      | Clearing this bit powers down the PWM   |
| 9 to 10  |                      |            | Reserved.   |
| 8        |                      | SPIPO      | Clearing this bit powers down the SPI.  |
| 7 to 6   | 00<br>01<br>10<br>11 | SPICLKDIV  | SPI block driving clock divider bits.<br>41.78 MHz.<br>20.89 MHz.<br>10.44 MHz.<br>5.22 MHz.              |
| 5        |                      | I2C1PO     | Clearing this bit powers down the I <sup>2</sup> C1.  |
| 4 to 3   | 00<br>01<br>10<br>11 | I2C1CLKDIV | I <sup>2</sup> C0 block driving clock divider bits.<br>41.78 MHz.<br>10.44 MHz.<br>5.22 MHz.<br>1.31 MHz. |
| 2        |                      | I2C0PO     | Clearing this bit powers down the I <sup>2</sup> C0.  |
| 1 to 0   | 00<br>01<br>10<br>11 | I2C0CLKDIV | I <sup>2</sup> C1 block driving clock divider bits.<br>41.78 MHz.<br>10.44 MHz.<br>5.22 MHz.<br>1.31 MHz. |

<sup>1</sup> Divided clock for SPI/I2C0/I2C1 must be greater than or equal to the CPU clock as selected by POWCON0[2:0]

Table 54. POWCON1 Write Sequence

| Name    | Code       |
|---------|------------|
| POWKEY3 | 0x76       |
| POWCON1 | User value |
| POWKEY4 | 0XB1       |

## DIGITAL PERIPHERALS

### GENERAL-PURPOSE INPUT/OUTPUT

The ADuC7023 provides up to 20 general-purpose, bidirectional I/O (GPIO) pins. All I/O pins apart from the pins shared with the ADC are 5 V tolerant, meaning the GPIOs support an input voltage of 5 V. The shared ADC pins only support an input up to AVDD. In general, many of the GPIO pins have multiple functions (see Table 55 for the pin function definitions). By default, the GPIO pins are configured in GPIO mode.

All GPIO pins have an internal pull-up resistor (of about 100 k $\Omega$ ) and their drive capability is 1.6 mA. Note that a maximum of 20 GPIOs can drive 1.6 mA at the same time. Using the GPxPAR registers, it is possible to enable/disable the pull-up resistors.

The 20 GPIOs are grouped in three ports, Port 0 to Port 2 (Port x). Each port is controlled by four or five MMRs.

The input level of any GPIO can be read at any time in the GPxDAT MMR, even when the pin is configured in a mode other than GPIO. The PLA input is always active.

When the ADuC7023 part enters a power-saving mode, the GPIO pins retain their state. Also note, that by setting RSTCFG bit 0, the GPIO pins can retain their state during a watchdog or software reset.

**Table 55. GPIO Pin Function Descriptions**

| Port | Pin                 | Configuration |                   |                          |          |
|------|---------------------|---------------|-------------------|--------------------------|----------|
|      |                     | 00            | 01                | 10                       | 11       |
| 0    | P0.0 <sup>1</sup>   | GPIO/BM       | nTRST             | ADC <sub>BUSY</sub>      | PLAI[8]  |
|      | P0.1 <sup>1,2</sup> | GPIO          | TDO               |                          | PLAI[9]  |
|      | P0.2 <sup>1,2</sup> | GPIO          | TDI               |                          | PLAO[8]  |
|      | P0.3 <sup>1</sup>   | GPIO          | TCK               |                          | PLAO[9]  |
|      | P0.4                | GPIO/IRQ0     | SCL0              | CONV <sub>START</sub>    | PLAI[0]  |
|      | P0.5                | GPIO          | SDA0              | COMP <sub>OUT</sub>      | PLAI[1]  |
|      | P0.6                | GPIO          | MISO              | SCL1 <sup>3</sup>        | PLAI[2]  |
|      | P0.7                | GPIO          | MOSI              | SDA1 <sup>3</sup>        | PLAO[0]  |
| 1    | P1.0                | GPIO          | SCLK              | PWM0                     | PLAO[1]  |
|      | P1.1                | GPIO/IRQ1     | $\overline{SS}$   | PWM1                     | PLAO[2]  |
|      | P1.2 <sup>4</sup>   | GPIO/IRQ2     | ADC4              | ECLK                     | PLAI[3]  |
|      | P1.3                | GPIO/IRQ3     | ADC5              |                          | PLAI[4]  |
|      | P1.4                | GPIO          | ADC10             |                          | PLAO[3]  |
|      | P1.5                | GPIO          | ADC6              | PWM <sub>TRIPINPUT</sub> | PLAO[4]  |
|      | P1.6                | GPIO          | SCL1 <sup>5</sup> | PWM2                     | PLAI[5]  |
|      | P1.7                | GPIO          | SDA1 <sup>5</sup> | PWM3                     | PLAI[6]  |
| 2    | P2.0                | GPIO          | ADC12             | PWM4                     | PLAI[7]  |
|      | P2.2                | GPIO          | ADC7              | PWM <sub>sync</sub>      | PLAO[6]  |
|      | P2.3                | GPIO          | ADC8              |                          | PLAO[7]  |
|      | P2.4                | GPIO          | ADC9              |                          | PLAI[10] |

<sup>1</sup>These pins should not be used by user code when debugging the part via JTAG. See Table 36 for further details on how to configure these pins for GPIO mode. The default value of these pins depends on the level of the P0.0/BM pin during the last reset sequence.

<sup>2</sup>If the pins are configured for JTAG mode (see Table 36), then these pins cannot be used as GPIO.

<sup>3</sup>I<sup>2</sup>C1 function is only available on the 32-lead and 36-ball packages.

<sup>4</sup>When configured in Mode 2, P1.2 is ECLK by default, or core clock output. To configure it as a clock input, the MDCLK bits in PLLCON must be set to 11.

<sup>5</sup>I<sup>2</sup>C1 function is only available on the 40-lead package.

### GPxCON Registers

| Name   | Address    | Default Value | Access |
|--------|------------|---------------|--------|
| GP0CON | 0xFFFFF400 | 0x00001111    | R/W    |
| GP1CON | 0xFFFFF404 | 0x00000000    | R/W    |
| GP2CON | 0xFFFFF408 | 0x00000000    | R/W    |

GPxCON are the Port x control registers, which select the function of each pin of Port x as described in Table 56.

**Table 56. GPxCON MMR Bit Descriptions**

| Bit      | Description                  |
|----------|------------------------------|
| 31 to 30 | Reserved.                    |
| 29 to 28 | Select function of Px.7 pin. |
| 27 to 26 | Reserved.                    |
| 25 to 24 | Select function of Px.6 pin. |
| 23 to 22 | Reserved.                    |
| 21 to 20 | Select function of Px.5 pin. |
| 19 to 18 | Reserved.                    |
| 17 to 16 | Select function of Px.4 pin. |
| 15 to 14 | Reserved.                    |
| 13 to 12 | Select function of Px.3 pin. |
| 11 to 10 | Reserved.                    |
| 9 to 8   | Select function of Px.2 pin. |
| 7 to 6   | Reserved.                    |
| 5 to 4   | Select function of Px.1 pin. |
| 3 to 2   | Reserved.                    |
| 1 to 0   | Select function of Px.0 pin. |

### GP0PAR Register

|               |   |
|---------------|---|
| Name          | GP0PAR  |
| Address       | 0xFFFFF42C  |
| Default value | 0x22220000  |
| Access        | Read/write  |
| Function      | GP0PAR programs the parameters for Port 0, Port 1, and Port 2. Note that the GP0DAT MMR must always be written after changing the GP0PAR MMR. |

### GP1PAR Register

|               |   |
|---------------|---|
| Name          | GP1PAR  |
| Address       | 0xFFFFF43C  |
| Default value | 0x22000022  |
| Access        | Read/write  |
| Function      | GP1PAR programs the parameters for Port 0, Port 1, and Port 2. Note that the GP1DAT MMR must always be written after changing the GP1PAR MMR. |

**GP2PAR Register**

|               |   |
|---------------|---|
| Name          | GP2PAR  |
| Address       | 0xFFFFF44C  |
| Default value | 0x00000000  |
| Access        | Read/write  |
| Function      | GP2PAR programs the parameters for Port 0, Port 1, and Port 2. Note that the GP2DAT MMR must always be written after changing the GP2PAR MMR. |

**Table 57. GPxPAR MMR Bit Descriptions**

| Bit      | Description           |
|----------|-----------------------|
| 31       | Reserved.             |
| 30 to 29 | Drive strength Px.7.  |
| 28       | Pull-up disable Px.7. |
| 27       | Reserved.             |
| 26 to 26 | Drive strength Px.6.  |
| 24       | Pull-up disable Px.6. |
| 23       | Reserved.             |
| 22 to 21 | Drive strength Px.5.  |
| 20       | Pull-up disable Px.5. |
| 19       | Reserved.             |
| 18 to 17 | Drive strength Px.4.  |
| 16       | Pull-up disable Px.4. |
| 15       | Reserved.             |
| 14 to 13 | Drive strength Px.3.  |
| 12       | Pull-up disable Px.3. |
| 11       | Reserved.             |
| 10 to 9  | Drive strength Px.2.  |
| 8        | Pull-up disable Px.2. |
| 7        | Reserved.             |
| 6 to 5   | Drive strength Px.1.  |
| 4        | Pull-up disable Px.1. |
| 3        | Reserved.             |
| 2 to 1   | Drive strength Px.0.  |
| 0        | Pull-up disable Px.0. |

**Table 58. GPIO Drive Strength Control Bits Descriptions**

| Control Bits Value | Description            |
|--------------------|------------------------|
| 00                 | Medium drive strength. |
| 01                 | Low drive strength.    |
| 1x                 | High drive strength.   |

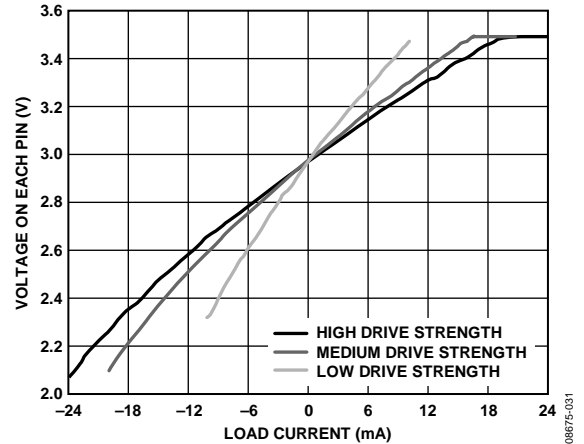


Figure 37. Programmable Strength for High Level

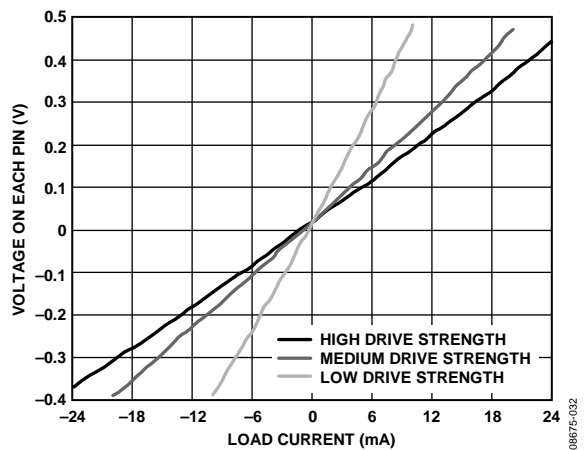


Figure 38. Programmable Strength for Low Level

The drive strength bits can be written one time only after reset. More writing to related bits has no effect on changing drive strength. The GPIO drive strength and pull-up disable is not always adjustable for the GPIO port. Some control bits cannot be changed (see Table 59).

**Table 59. GPxPAR Control Bits Access Descriptions<sup>1</sup>**

| Bit      | GP0PAR   | GP1PAR   | GP2PAR   |
|----------|----------|----------|----------|
| 31       | Reserved | Reserved | Reserved |
| 30 to 29 | R/W      | R/W      | Reserved |
| 28       | R/W      | R/W      | Reserved |
| 27       | Reserved | Reserved | Reserved |
| 26 to 26 | R/W      | R/W      | Reserved |
| 24       | R/W      | R/W      | Reserved |
| 23       | Reserved | Reserved | Reserved |
| 22 to 21 | R/W      | R (b00)  | Reserved |
| 20       | R/W      | R/W      | Reserved |
| 19       | Reserved | Reserved | Reserved |
| 18 to 17 | R (b00)  | R (b00)  | R (b00)  |
| 16       | R/W      | R/W      | R/W      |
| 15       | Reserved | Reserved | Reserved |
| 14 to 13 | R (b00)  | R (b00)  | R (b00)  |
| 12       | R/W      | R/W      | R/W      |
| 11       | Reserved | Reserved | Reserved |

| Bit     | GP0PAR   | GP1PAR   | GP2PAR   |
|---------|----------|----------|----------|
| 10 to 9 | R (b00)  | R (b00)  | R (b00)  |
| 8       | R/W      | R/W      | R/W      |
| 7       | Reserved | Reserved | Reserved |
| 6 to 5  | R (b00)  | R (b00)  | Reserved |
| 4       | R/W      | R/W      | Reserved |
| 3       | Reserved | Reserved | Reserved |
| 2 to 1  | R (b00)  | R (b00)  | R (b00)  |
| 0       | R/W      | R/W      | R (b0)   |

<sup>1</sup> When P2.0 is configured as AIN12, the internal pull-up resistor cannot be disabled.

### GP0DAT Register

| Name   | Address    | Default Value | Access |
|--------|------------|---------------|--------|
| GP0DAT | 0xFFFFF420 | 0x000000XX    | R/W    |
| GP1DAT | 0xFFFFF430 | 0x000000XX    | R/W    |
| GP2DAT | 0xFFFFF440 | 0x000000XX    | R/W    |

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 configured as output, and store the input value of the pins configured as input.

**Table 60. GPxDAT MMR Bit Descriptions**

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

### GP0SET Register

Name: GP0SET  
 Address: 0xFFFFF424  
 Default value: 0x000000XX  
 Access: Write  
 Function: GP0SET is a data set Port x register.

### GP1SET Register

Name: GP1SET  
 Address: 0xFFFFF434  
 Default value: 0x000000XX  
 Access: Write  
 Function: GP1SET is a data set Port x register.

### GP2SET Register

Name: GP2SET  
 Address: 0xFFFFF444  
 Default value: 0x000000XX  
 Access: Write  
 Function: GP2SET is a data set Port x register.

**Table 61. GPxSET MMR Bit Descriptions**

| Bit      | Description  |
|----------|--|
| 31 to 24 | Reserved.  |
| 23 to 16 | Data port x.<br>This bit is set to 1 by the user to set bit on Port x; this bit also sets the corresponding bit in the GPxDAT MMR.<br>This bit is cleared to 0 by the user; this bit does not affect the data out. |
| 15 to 0  | Reserved.  |

### GP0CLR Registers

Name: GP0CLR  
 Address: 0xFFFFF428  
 Default value: 0x000000XX  
 Access: Write  
 Function: GP0CLR is a data clear Port x register.

### GP1CLR Registers

Name: GP1CLR  
 Address: 0xFFFFF438  
 Default value: 0x000000XX  
 Access: Write  
 Function: GP1CLR is a data clear Port x register.

### GP2CLR Registers

Name: GP2CLR  
 Address: 0xFFFFF448  
 Default value: 0x000000XX  
 Access: Write  
 Function: GP2CLR is a data clear Port x register.

**Table 62. GPxCLR MMR Bit Descriptions**

| Bit      | Description  |
|----------|--|
| 31 to 24 | Reserved.  |
| 23 to 16 | Data port x clear bit.<br>This bit is set to 1 by the user to clear the bit on Port x; this bit also clears the corresponding bit in the GPxDAT MMR.<br>This bit is cleared to 0 by the user; this bit does not affect the data out. |
| 15 to 0  | Reserved.  |

## SERIAL PERIPHERAL INTERFACE

The ADuC7023 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 20 Mbps.

The SPI port can be configured for master or slave operation and typically consists of four pins: MISO, MOSI, SCLK, and SPISS.

### 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, MSB 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, MSB first.

### SCLK (Serial Clock I/O) Pin

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

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

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

where:

$f_{UCLK}$  is the clock selected by POWCON1 Bit 7 to Bit 6.

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

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 10 Mbps.

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

### SPI Chip Select ( $\overline{SS}$ Input) Pin

In SPI slave mode, a transfer is initiated by the assertion of  $\overline{SS}$ , 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, the  $\overline{SS}$  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

P1.1 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.

P1.0 is the SCLK pin.

P0.6 is the master in, slave out (MISO) pin.

P0.7 is the master out, slave in (MOSI) pin.

To configure these pins for SPI mode, see the General-Purpose Input/Output section.

### SPI Registers

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

#### SPI Status Register

Name: SPISTA

Address: 0xFFFF0A00

Default value: 0x0000

Access: Read

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

Table 63. SPISTA MMR Bit Designations

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

**SPIRX Register**

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

**SPITX Register**

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



***SPI DIV Register***

Name: SPIDIV  
Address: 0xFFFF0A0C  
Default value: 0x00  
Access: Read/write  
Function: This 6-bit MMR is the SPI baud rate selection register. (Note that the maximum value of this MMR is 0x3E.)

***SPI Control Register***

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

Table 64. SPICON MMR Bit Designations

| Bit      | Name    | Description  |
|----------|---------|--|
| 15 to 14 | SPIMDE  | <p>SPI IRQ mode bits. These bits configure when the Tx/Rx interrupts occur in a transfer.</p> <p>[00] = Tx interrupt occurs when one byte has been transferred. Rx interrupt occurs when one or more bytes have been received into the FIFO.</p> <p>[01] = Tx interrupt occurs when two bytes has been transferred. Rx interrupt occurs when two or more bytes have been received into the FIFO.</p> <p>[10] = Tx interrupt occurs when three bytes has been transferred. Rx interrupt occurs when three or more bytes have been received into the FIFO.</p> <p>[11] = Tx interrupt occurs when four bytes has been transferred. Rx interrupt occurs when the Rx FIFO is full or four bytes present.</p> |
| 13       | SPITFLH | <p>SPI Tx FIFO flush enable bit.</p> <p>This bit is set to flush the Tx 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 Tx FIFO are ignored while this bit is set. This bit is cleared to disable Tx FIFO flushing.</p>   |
| 12       | SPIRFLH | <p>SPI Rx FIFO flush enable bit.</p> <p>This bit is set to flush the Rx 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 Rx FIFO initiates a transfer. This bit is cleared to disable Rx FIFO flushing.</p>   |
| 11       | SPICONT | <p>Continuous transfer enable.</p> <p>This bit is set by the user to enable continuous transfer. In master mode, the transfer continues until no valid data is available in the Tx register. SS is asserted and remains asserted for the duration of each 8-bit serial transfer until Tx is empty. This bit is cleared by the 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 1 serial clock cycle.</p>  |
| 10       | SPILP   | <p>Loop back enable bit.</p> <p>This bit is set by the user to connect MISO to MOSI and test software. This bit is cleared by the user to be in normal mode.</p>   |
| 9        | SPIOEN  | <p>Slave MISO output enable bit.</p> <p>This bit is set for MISO to operate as normal. This bit is cleared to disable the output driver on the MISO pin. The MISO pin is open-drain when this bit is clear.</p>  |
| 8        | SPIROW  | <p>SPIRX overflow overwrite enable.</p> <p>This bit is set by the user; the valid data in the Rx register is overwritten by the new serial byte received. This bit is cleared by the user; the new serial byte received is discarded.</p>  |
| 7        | SPIZEN  | <p>SPI transmit zeros when Tx FIFO is empty.</p> <p>This bit is set to transmit 0x00 when there is no valid data in the Tx FIFO. This bit is cleared to transmit the last transmitted value when there is no valid data in the Tx FIFO.</p>  |
| 6        | SPITMDE | <p>SPI transfer and interrupt mode.</p> <p>This bit is set by the user to initiate transfer with a write to the SPITX register. Interrupt only occurs when Tx is empty. This bit is cleared by the user to initiate transfer with a read of the SPIRX register. Interrupt only occurs when Rx is full.</p>   |
| 5        | SPILF   | <p>LSB first transfer enable bit.</p> <p>This bit is set by the user; the LSB is transmitted first. This bit is cleared by the user; the MSB is transmitted first.</p>   |
| 4        | SPIWOM  | <p>SPI wired or mode enable bit.</p> <p>This bit is set to 1 enable open-drain data output. External pull-ups are required on data out pins. This bit is cleared for normal output levels.</p>   |
| 3        | SPICPO  | <p>Serial clock polarity mode bit.</p> <p>This bit is set by the user; the serial clock idles high. This bit is cleared by the user; the serial clock idles low.</p>   |
| 2        | SPICPH  | <p>Serial clock phase mode bit.</p> <p>This bit is set by the user; the serial clock pulses at the beginning of each serial bit transfer. This bit is cleared by the user; the serial clock pulses at the end of each serial bit transfer.</p>   |

| Bit | Name    | Description  |
|-----|---------|--|
| 1   | SPI MEN | Master mode enable bit.<br>This bit is set by the user to enable master mode.<br>This bit is cleared by the user to enable slave mode. |
| 0   | SPI EN  | SPI enable bit.<br>This bit is set by the user to enable the SPI.<br>This bit is cleared by the user to disable the SPI.               |

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

The ADuC7023 incorporates two I<sup>2</sup>C peripherals that may 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-AND format that allows arbitration in a multimaster system. These pins require external pull-up resistors. Typical pull-up values are between 4.7 kΩ and 10 kΩ.

The I<sup>2</sup>C bus peripheral address in the I<sup>2</sup>C bus system is programmed by the user. This ID can be modified any time 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 only be configured as a master or 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 ADuC7023 includes support for repeated start conditions. In master mode, the ADuC7023 can be programmed to generate a repeated start. In slave mode, the ADuC7023 recognizes repeated start conditions. In master and slave mode, the part recognizes both 7-bit and 10-bit bus addresses. In I<sup>2</sup>C master mode, the ADuC7023 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 ADuC7023. However, the ADuC7023 cannot enable clock stretching. In slave mode, the ADuC7023 can be programmed to return a 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. In loopback mode. The transmit and receive circuits in both master and slave mode 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 pins of the ADuC7023 device are P0.4 and P0.5 for I<sup>2</sup>C0 and P0.6 and P0.7 for I<sup>2</sup>C1.

P0.4 and P0.6 are the I<sup>2</sup>C clock signals and P0.5 and P0.7 are the I<sup>2</sup>C data signals. For instance, to configure I<sup>2</sup>C0 pins (SCL0, SDA0), Bit 16 and Bit 20 of the GPIOCON register must be set to 1 to enable I<sup>2</sup>C mode. On the other hand, to configure I<sup>2</sup>C1 pins (SCL1, SDA1), Bit 25 and Bit 29 of the GPIOCON register must be set to 1 to enable I<sup>2</sup>C mode, as shown in the GPIO section. I<sup>2</sup>C1 function is available at P0.6 and P0.7 on 32-lead and 36-ball packages and available at P1.6 and P1.7 on 40-lead package.

### 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 and the clock selected by POWCON1 Bit 4 to Bit 0.

$DIVH$  is the high period of the clock.

$DIVL$  is the low period of the clock.

Thus, for 100 kHz operation,

$$DIVH = DIVL = 0xCF$$

and for 400 kHz,

$$DIVH = 0x28, DIVL = 0x3C$$

The I2CDIV register corresponds to DIVH:DIVL.

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

#### Slave Mode

In slave mode, the registers I2CxID0, I2CxID1, I2CxID2, and I2CxID3 contain the device IDs. The device compares the four I2CxIDx registers to the address byte received from the bus master. To be correctly addressed, the 7MSBs of either ID register must be identical to that of the 7MSBs of the first received address byte. The LSB of the ID registers (the transfer direction bit) is ignored in the process of address recognition.

The ADuC7023 also supports 10-bit addressing mode. When Bit 1 of I2CxSCON (ADR10EN bit) is set to 1, then one 10-bit address is supported in slave mode and is stored in registers I2CxID0 and I2CxID1. The 10-bit address is derived as follows:

I2CxID0[7:3] must be set to 11110b.

I2CxID0[2:1] = Address Bits[9:8].

I2CxID0[0] is the read/write bit and is not part of the I<sup>2</sup>C address. This must be written as 0.

I2CxID0[7:0] = Address Bits[7:0].

#### Master Mode

In master mode, the I2CxADR0 register is programmed with the I<sup>2</sup>C address of the device.

In 7-bit address mode, I2CxADR0[7:1] are set to the device address. I2CxADR0[0] is the read/write bit.

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

I2CxADR0[7:3] must be set to 11110b.

I2CxADR0[2:1] = Address Bits[9:8].

I2CxADR1[7:0] = Address Bits[7:0].

I2CxADR0[0] is the read/write bit.

In order to perform a read from a slave with a 10-bit address, the master must first send a 10-bit address with the read/write bit cleared. The master must then generate a repeated start and

send only the first byte of the address again, this time with the read/write bit set. A repeated start is generated by writing to I2CxADR0 while the master is still busy.

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

The I<sup>2</sup>C peripheral interfaces consist of a number of MMRs. These are described in the following section.

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

#### I<sup>2</sup>C Master Control Registers, I2CxMCON

Name: I2C0MCON, I2C1MCON

Address: 0xFFFF0800, 0xFFFF0900

Default value: 0x0000, 0x0000

Access: Read/write

Function: These 16-bit MMRs configure the I<sup>2</sup>C peripheral in master mode.

**Table 65. I2CxMCON MMR Bit Designations**

| Bit     | Name       | Description  |
|---------|------------|--|
| 15 to 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>This bit is set to enable an interrupt on detecting a stop condition on the I <sup>2</sup> C bus.<br>This bit clears this interrupt source.  |
| 7       | I2CNACKENI | I <sup>2</sup> C no acknowledge received interrupt enable bit.<br>This bit is set to enable interrupts when the I <sup>2</sup> C master receives a no acknowledge.<br>This bit clears this interrupt source.   |
| 6       | I2CALENI   | I <sup>2</sup> C arbitration lost interrupt enable bit.<br>This bit is set to enable interrupts when the I <sup>2</sup> C master has lost in trying to gain control of the I <sup>2</sup> C bus.<br>This bit clears this interrupt source.                           |
| 5       | I2CMTENI   | I <sup>2</sup> C transmit interrupt enable bit.<br>This bit is set to enable interrupts when the I <sup>2</sup> C master has transmitted a byte.<br>This bit clears this interrupt source.   |
| 4       | I2CMRENI   | I <sup>2</sup> C receive interrupt enable bit.<br>This bit is set to enable interrupts when the I <sup>2</sup> C master receives data.<br>This bit is cleared by the user to disable interrupts when the I <sup>2</sup> C master is receiving data.                  |
| 3       |            | Reserved. Write a value of 0 to this bit.  |
| 2       | I2CILEN    | I <sup>2</sup> C internal loopback enable bit.<br>This bit is set to enable loopback test mode. In this mode, the SCL and SDA signals are connected internally to their respective input signals.<br>This bit is cleared by the user to disable loopback mode.       |
| 1       | I2CBD      | I <sup>2</sup> C master backoff disable bit.<br>This bit is set to allow the device to compete for control of the bus even if another device is currently driving a start condition.<br>This bit is cleared to back off until the I <sup>2</sup> C bus becomes free. |
| 0       | I2CMEN     | I <sup>2</sup> C master enable bit.<br>This bit is set by the user to enable I <sup>2</sup> C master mode.<br>This bit is cleared to disable I <sup>2</sup> C master mode.   |

**I<sup>2</sup>C Master Status Registers, I2CxMSTA**

Name: I2C0MSTA , I2C1MSTA

Address: 0xFFFFF0804, 0xFFFFF0904

Default value: 0x0000, 0x0000

Access: Read

Function: These 16-bit MMRs are the I<sup>2</sup>C status registers in master mode.**Table 66. I2CxMSTA MMR Bit Designations**

| Bit      | Name      | Description   |
|----------|-----------|---|
| 15 to 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 Rx FIFO overflow.<br>This bit is set to 1 when a byte is written to the Rx 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 I2CxMCON is set, an interrupt is generated when this bit is set.<br>This bit clears this interrupt source.        |
| 7        | I2CMNA    | 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 I2CxMCON 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>This bit is set to 1 when the master is busy processing a transaction.<br>This bit is 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 has lost in trying to gain control of the I <sup>2</sup> C bus. If the I2CALENI bit in I2C1MCON 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 I2C1MCON 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 Rx FIFO. If the I2CMRENI in I2C1MCON 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 becomes high if the Tx FIFO is empty or only contains one byte and the master has transmitted an address and write. If the I2CMTENI bit in I2C1MCON is set, an interrupt is generated when this bit is set.<br>This bit is cleared in all other conditions. |
| 1 to 0   | I2CMTFSTA | I <sup>2</sup> C master Tx FIFO status bits.<br>00 = I <sup>2</sup> C master Tx FIFO empty.<br>01 = Reserved.<br>10 = 1 byte in master Tx FIFO.<br>11 = I <sup>2</sup> C master Tx FIFO full.   |

**I<sup>2</sup>C Master Receive Registers, I2CxMRX**

Name: I2C0MRX, I2C1MRX  
 Address: 0xFFFF0808, 0xFFFF0908  
 Default value: 0x00  
 Access: Read only  
 Function: These 8-bit MMRs are the I<sup>2</sup>C master receive registers.

**I<sup>2</sup>C Master Transmit Registers, I2CxMTX**

Name: I2C0MTX, I2C1MTX  
 Address: 0xFFFF080C, 0xFFFF090C  
 Default value: 0x00, 0x00  
 Access: Write only  
 Function: These 8-bit MMRs are the I<sup>2</sup>C master transmit registers

**I<sup>2</sup>C Master Read Count Registers, I2CxMCNT0**

Name: I2C0MCNT0, I2C1MCNT0  
 Address: 0xFFFF0810, 0xFFFF0910  
 Default value: 0x0000, 0x0000  
 Access: Read/write  
 Function: These 16-bit MMRs hold the required number of bytes when the master begins a read sequence from a slave device.

**Table 67. I2CxMCNT0 MMR Bit Descriptions: Address = 0xFFFF0810, 0xFFFF0910. Default Value = 0x0000**

| Bit     | Name     | Description   |
|---------|----------|---|
| 15 to 9 |          | Reserved.   |
| 8       | I2CRECNT | This bit is set if greater than 256 bytes are required from the slave.<br>This bit is cleared when reading 256 bytes or less.                               |
| 7 to 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, these bits should be set to 0. |

**I<sup>2</sup>C Master Current Read Count Registers, I2CxMCNT1**

Name: I2C0MCNT1, I2C1MCNT1  
 Address: 0xFFFF0814, 0xFFFF0914  
 Default value: 0x00, 0x00  
 Access: Read  
 Function: These 8-bit MMRs hold the number of bytes received thus far during a read sequence with a slave device.

**I<sup>2</sup>C Address 0 Registers, I2CxADR0**

Name: I2C0ADR0, I2C1ADR0  
 Address: 0xFFFF0818, 0xFFFF0918  
 Default value: 0x00  
 Access: Read/write  
 Function: These 8-bit MMRs hold the 7-bit slave address and the read/write bit when the master begins communicating with a slave.

**Table 68. I2CxADR0 MMR in 7-Bit Address Mode: Address = 0xFFFF0818, 0xFFFF0918. Default Value = 0x00**

| Bit    | Name   | Description   |
|--------|--------|---|
| 7 to 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 69. I2CxADR0 MMR in 10-Bit Address Mode**

| Bit    | Name    | Description  |
|--------|---------|--|
| 7 to 3 |         | These bits must be set to [11110b] in 10-bit address mode.   |
| 2 to 1 | I2CMADR | These bits contain ADDR[9:8] in 10-bit address 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 Registers, I2CxADR1**

Name: I2C0ADR1, I2C1ADR1

Address: 0xFFFF081C, 0xFFFF091C

Default value: 0x00

Access: Read/write

Function: These 8-bit MMRs are used in 10-bit addressing mode only. These registers contain the least significant byte of the address.

**Table 70. I2CxADR1 MMR in 10-Bit Address Mode**

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

**I<sup>2</sup>C Master Clock Control Register, I2CxDIV**

Name: I2C0DIV, I2C1DIV

Address: 0xFFFF0824, 0xFFFF0924

Default value: 0x1F1F

Access: Read/write

Function: These MMRs control the frequency of the I<sup>2</sup>C clock generated by the master on to the SCL pin. For further details, see the I<sup>2</sup>C initial section.

**Table 72. I2CxSCON MMR Bit Designations**

| Bit      | Name      | Description   |
|----------|-----------|---|
| 15 to 11 |           | Reserved bits.  |
| 10       | I2CSTXENI | Slave transmit interrupt enable bit.<br>This bit is set to enable an interrupt after a slave transmits a byte.<br>This bit clears this interrupt source.  |
| 9        | I2CSRXENI | Slave receive interrupt enable bit.<br>This bit is set to enable an interrupt after the slave receives data.<br>This bit clears this interrupt source.  |
| 8        | I2CSSENI  | I <sup>2</sup> C stop condition detected interrupt enable bit.<br>This bit is set to enable an interrupt on detecting a stop condition on the I <sup>2</sup> C bus.<br>This bit clears this interrupt source.   |
| 7        | I2CNACKEN | I <sup>2</sup> C no acknowledge enable bit.<br>This bit is set to no acknowledge the next byte in the transmission sequence.<br>This bit is cleared to let the hardware control the acknowledge/no acknowledge sequence.  |
| 6        |           | Reserved. Write a value of 0 to this bit.   |
| 5        | I2CSETEN  | I <sup>2</sup> C early transmit interrupt enable bit.<br>This bit is set to enable a transmit request interrupt just after the positive edge of SCL during the read bit transmission.<br>This bit is cleared to enable a transmit request interrupt just after the negative edge of SCL during the read bit transmission. |

**Table 71. I2CxDIV MMR**

| Bit     | Name | Description  |
|---------|------|--|
| 15 to 8 | DIVH | These bits control the duration of the high period of SCL. |
| 7 to 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 Registers, I2CxSCON**

Name: I2C0SCON, I2C1SCON

Address: 0xFFFF0828, 0xFFFF0928

Default value: 0x0000

Access: Read/write

Function: These 16-bit MMRs configure the I<sup>2</sup>C peripheral in slave mode.



| Bit | Name     | Description  |
|-----|----------|--|
| 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 I2CxSSTA register.<br>This bit is cleared at all other times.  |
| 3   | I2CHGCEN | I <sup>2</sup> C hardware general call enable. 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 I2CxALT 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 broadcast message to all master devices on the bus. The ADuC7023 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 I2CxALT register should always be written to 1, as per the I <sup>2</sup> C January 2000 bus specification.<br>This bit and I2CGCEN are set to enable hardware general call recognition in slave mode.<br>This bit is cleared to disable recognition of hardware general call commands. |
| 2   | I2CGCEN  | I <sup>2</sup> C general call enable. This bit is set 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.<br>This bit is set to allow the slave acknowledge I <sup>2</sup> C general call commands.<br>This bit is cleared to disable recognition of general call commands.  |
| 1   | ADR10EN  | I <sup>2</sup> C 10-bit address mode.<br>This bit is set to 1 to enable 10-bit address mode.<br>This bit is cleared to 0 to enable normal address mode.  |
| 0   | I2CSEN   | I <sup>2</sup> C slave enable bit.<br>This bit is set by user to enable I <sup>2</sup> C slave mode.<br>This bit is cleared by the user to disable I <sup>2</sup> C slave mode.  |

### I<sup>2</sup>C Slave Status Registers, I2CxSSTA

Name: I2C0SSTA, I2C1SSTA

Address: 0xFFFFF082C, 0xFFFFF092C

Default value: 0x0000, 0x0000

Access: Read/write

Function: These 16-bit MMRs are the I<sup>2</sup>C status registers in slave mode.

Table 73. I2CxSSTA 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. It is also set if a start byte (0x01) is received. If 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 I2CxSSTA register also clears this bit.  |
| 12 to 11 | I2CID[1:0]   | I <sup>2</sup> C address matching register. These bits indicate which I2CxIDx register matches the received address.<br>[00] = received address matches I2CxID0.<br>[01] = received address matches I2CxID1.<br>[10] = received address matches I2CxID2.<br>[11] = received address matches I2CxID3.   |
| 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.<br>When the I2CSENI bit in I2CxSCON is set, an interrupt is generated.<br>This bit is cleared by reading this register.   |
| 9 to 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>These bits are not cleared by a general call reset command.<br>These bits are cleared by writing a 1 to the I2CGCLR bit in I2CxSCON.  |
| 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 is a reset command, then all registers return to their default state. If the command received is a hardware general call, the Rx FIFO holds the second byte of the command, and this can be compared with the I2CxALT register.<br>This bit is cleared by writing a 1 to the I2CGCLR bit in I2CxSCON.  |
| 6        | I2CSBUSY     | I <sup>2</sup> C slave busy status bit.<br>This bit is set to 1 when the slave receives a start condition.<br>This bit is cleared by hardware if the received address does not match any of the I2CxIDx registers, the slave device receives a stop condition or if a repeated start address does not match any of the I2CxIDx 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 no acknowledge is returned because there is no data in the Tx FIFO or if the I2CNACKEN bit is set in the I2CxSCON register.<br>This bit is cleared in all other conditions.  |
| 4        | I2CSRxFO     | Slave Rx FIFO overflow.<br>This bit is set to 1 when a byte is written to the Rx 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 slave Rx FIFO is not empty. This bit causes an interrupt to occur if the I2CSRXENI bit in I2CxSCON is set.<br>The Rx FIFO must be read or flushed to clear this bit.  |
| 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 I2CxSCON is = 0, this bit goes high just after the negative edge of SCL during the read bit transmission. If the I2CSETEN bit in I2CxSCON 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 I2CxSCON is set.<br>This bit is cleared in all other conditions. |

| Bit | Name     | Description  |
|-----|----------|--|
| 1   | I2CSTFE  | I <sup>2</sup> C slave FIFO underflow status bit.<br>This bit goes high if the Tx 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 I2CxSCON is = 0, this bit goes high if the slave Tx FIFO is empty. If the I2CSETEN bit in I2CxSCON is = 1, this bit goes high just after the positive edge of SCL during the write bit transmission.<br>This bit asserts once only for a transfer.<br>This bit is cleared after being read. |

**I<sup>2</sup>C Slave Receive Registers, I2CxSRX**

Name: I2C0SRX, I2C1SRX  
Address: 0xFFFF0830, 0xFFFF0930  
Default value: 0x00  
Access: Read  
Function: These 8-bit MMRs are the I<sup>2</sup>C slave receive register.

**I<sup>2</sup>C Slave Transmit Registers, I2CxSTX**

Name: I2C0STX, I2C1STX  
Address: 0xFFFF0834, 0xFFFF0934  
Default value: 0x00  
Access: Write  
Function: These 8-bit MMRs are the I<sup>2</sup>C slave transmit registers.

**I<sup>2</sup>C Hardware General Call Recognition Registers, I2CxALT**

Name: I2C0ALT, I2C1ALT  
Address: 0xFFFF0838, 0xFFFF0938  
Default value: 0x00  
Access: Read/write  
Function: These 8-bit MMRs are used with hardware general calls when the I2CxSCON Bit 3 is set to 1. These registers are 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 Registers, I2CxIDx**

Name: I2C0IDx, I2C1IDx  
Addresses: 0xFFFF093C = I2C1ID0  
0xFFFF083C = I2C0ID0  
0xFFFF0940 = I2C1ID1  
0xFFFF0840 = I2C0ID1  
0xFFFF0944 = I2C1ID2  
0xFFFF0844 = I2C0ID2  
0xFFFF0948 = I2C1ID3  
0xFFFF0848 = I2C0ID3  
Default value: 0x00  
Access: Read/write  
Function: These 8-bit MMRs are programmed with I<sup>2</sup>C bus IDs of the slave. See the I<sup>2</sup>C Bus Addresses section for further details.

**I<sup>2</sup>C Common Registers****I<sup>2</sup>C FIFO Status Registers, I2CxFSTA**

Name: I2C0FSTA, I2C1FSTA  
Address: 0xFFFF084C, 0xFFFF094C  
Default value: 0x0000  
Access: Read/write  
Function: These 16-bit MMRs contain the status of the Rx/Tx FIFOs in both master and slave modes.

**Table 74. I2CxFTA MMR Bit Designations**

| Bit      | Name      | Description   |
|----------|-----------|---|
| 15 to 10 |           | Reserved bits.  |
| 9        | I2CFMTX   | This bit is set to 1 to flush the master Tx FIFO.   |
| 8        | I2CFSTX   | This bit is set to 1 to flush the slave Tx FIFO.  |
| 7 to 6   | I2CMRXSTA | I <sup>2</sup> C master receive FIFO status bits.<br>[00] = FIFO empty.<br>[01] = byte written to FIFO.<br>[10] = 1 byte in FIFO.<br>[11] = FIFO full.  |
| 5 to 4   | I2CMTXSTA | I <sup>2</sup> C master transmit FIFO status bits.<br>[00] = FIFO empty.<br>[01] = byte written to FIFO.<br>[10] = 1 byte in FIFO.<br>[11] = FIFO full. |
| 3 to 2   | I2CSRXSTA | I <sup>2</sup> C slave receive FIFO status bits.<br>[00] = FIFO empty<br>[01] = byte written to FIFO<br>[10] = 1 byte in FIFO<br>[11] = FIFO full       |
| 1 to 0   | I2CSTXSTA | I <sup>2</sup> C slave transmit FIFO status bits.<br>[00] = FIFO empty.<br>[01] = byte written to FIFO.<br>[10] = 1 byte in FIFO.<br>[11] = FIFO full.  |

**PROGRAMMABLE LOGIC ARRAY (PLA)**

Every ADuC7023 integrates a fully programmable logic array (PLA) consisting of sixteen PLA elements.

Each PLA element contains a two-input look-up table that can be configured to generate any logic output function based on two inputs and a flip-flop. This is represented in Figure 39.

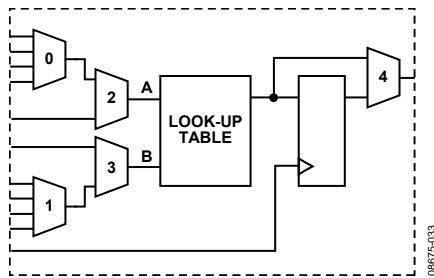


Figure 39. PLA Element

In total, 20 GPIO pins are available on the ADuC7023 for the PLA. These include 11 input pins and nine output pins, which need to be configured in the GPxCON register as PLA pins before using the PLA.

The PLA is configured via a set of user MMRs. The output(s) of the PLA can be routed to the internal interrupt system, to the CONV<sub>START</sub> signal of the ADC, to an MMR, or to any of the eight PLA output pins.

**Table 75. Element Input/Output**

| PLA Block 0 |       |                   | PLA Block 1 |       |                   |
|-------------|-------|-------------------|-------------|-------|-------------------|
| Element     | Input | Output            | Element     | Input | Output            |
| 0           | P0.4  | P0.7              | 8           | P0.0  | P0.2              |
| 1           | P0.5  | P1.0              | 9           | P0.1  | P0.3              |
| 2           | P0.6  | P1.1              | 10          | P2.4  | P2.5 <sup>1</sup> |
| 3           | P1.2  | P1.4              | 11          | NC    | NC                |
| 4           | P1.3  | P1.5              | 12          | NC    | NC                |
| 5           | P1.6  | P2.1 <sup>1</sup> | 13          | NC    | NC                |
| 6           | P1.7  | P2.2              | 14          | NC    | NC                |
| 7           | P2.0  | P2.3              | 15          | NC    | NC                |

<sup>1</sup> Internal pins only. Read via GPxDAT register.

**PLA MMRs Interface**

The PLA peripheral interface consists of the 22 MMRs described in the following sections.

**PLAELMx Registers**

PLAELMx are Element 0 to Element 15 control registers. They configure the input and output mux of each element, select the function in the look-up table, and bypass/use the flip-flop (see Table 77).

**Table 76. PLAELMx Registers**

| Name     | Address    | Default Value | Access |
|----------|------------|---------------|--------|
| PLAELM0  | 0xFFFF0B00 | 0x0000        | R/W    |
| PLAELM1  | 0xFFFF0B04 | 0x0000        | R/W    |
| PLAELM2  | 0xFFFF0B08 | 0x0000        | R/W    |
| PLAELM3  | 0xFFFF0B0C | 0x0000        | R/W    |
| PLAELM4  | 0xFFFF0B10 | 0x0000        | R/W    |
| PLAELM5  | 0xFFFF0B14 | 0x0000        | R/W    |
| PLAELM6  | 0xFFFF0B18 | 0x0000        | R/W    |
| PLAELM7  | 0xFFFF0B1C | 0x0000        | R/W    |
| PLAELM8  | 0xFFFF0B20 | 0x0000        | R/W    |
| PLAELM9  | 0xFFFF0B24 | 0x0000        | R/W    |
| PLAELM10 | 0xFFFF0B28 | 0x0000        | R/W    |
| PLAELM11 | 0xFFFF0B2C | 0x0000        | R/W    |
| PLAELM12 | 0xFFFF0B30 | 0x0000        | R/W    |
| PLAELM13 | 0xFFFF0B34 | 0x0000        | R/W    |
| PLAELM14 | 0xFFFF0B38 | 0x0000        | R/W    |
| PLAELM15 | 0xFFFF0B3C | 0x0000        | R/W    |

Table 77. PLAE<sub>LMx</sub> MMR Bit Descriptions

| Bit      | Value  | Description   |
|----------|--|---|
| 31 to 11 |  | Reserved.   |
| 10 to 9  |  | Mux 0 control (see Table 81).   |
| 8 to 7   |  | Mux 1 control (see Table 81).   |
| 6        |  | Mux 2 control.<br>This bit is set by the user to select the output of Mux 0.<br>This bit is cleared by the user to select the bit value from the PLADIN register.                   |
| 5        |  | Mux 3 control.<br>This bit is set by the user to select the input pin of the particular element.<br>This bit is cleared by the user to select the output of Mux 1.                  |
| 4 to 1   | 0000<br>0001<br>0010<br>0011<br>0100<br>0101<br>0110<br>0111<br>1000<br>1001<br>1010<br>1011<br>1100<br>1101<br>1110<br>1111 | Look-up table control.<br>0.<br>NOR.<br>B and not A.<br>Not A.<br>A and not B.<br>Not B.<br>EXOR.<br>NAND.<br>AND.<br>EXNOR.<br>B.<br>Not A or B.<br>A.<br>A or not B.<br>OR.<br>1. |
| 0        |  | Mux 4 control.<br>This bit is set by the user to bypass the flip-flop.<br>This bit is cleared by the user to select the flip-flop (cleared by default).                             |

**PLACLK Register**

|                |  |
|----------------|--|
| Name:          | PLACLK   |
| Address:       | 0xFFFF0B40   |
| Default value: | 0x00   |
| Access:        | Read/write   |
| Function:      | PLACLK is the clock selection for the flip-flops. The maximum frequency when using the GPIO pins as the clock input for the PLA blocks is 41.78 MHz. |

Table 78. PLACLK MMR Bit Descriptions

| Bit     | Value | Description                  |
|---------|-------|------------------------------|
| 31 to 7 |       | Reserved.                    |
| 6 to 4  |       | Clock source selection.      |
|         | 000   | GPIO clock on P0.5.          |
|         | 001   | GPIO clock on P1.1.          |
|         | 010   | GPIO clock on P1.6.          |
|         | 011   | HCLK.                        |
|         | 100   | External 32.768 kHz crystal. |
|         | 101   | Timer1 overflow.             |
|         | 110   | UCLK.                        |
|         | 111   | Internal 32,768 oscillator.  |
| 3       |       | Reserved.                    |
| 2 to 0  |       | Clock source selection.      |
|         | 000   | GPIO clock on P0.5.          |
|         | 001   | GPIO clock on P1.1.          |
|         | 010   | GPIO clock on P1.6.          |
|         | 011   | HCLK.                        |
|         | 100   | External 32.768 kHz crystal. |
|         | 101   | Timer1 overflow.             |
|         | 110   | UCLK.                        |
|         | 111   | Internal 32,768 oscillator.  |

### PLAIRQ Register

**Name:** PLAIRQ  
**Address:** 0xFFFF0B44  
**Default value:** 0x00000000  
**Access:** Read/write  
**Function:** PLAIRQ enables IRQ0 and/or IRQ1 and selects the source of the IRQ.

**Table 79. PLAIRQ MMR Bit Descriptions**

| Bit      | Value  | Description  |
|----------|--|--|
| 31 to 13 |  | Reserved.  |
| 12       |  | PLA IRQ1 enable bit.   |
| 11 to 8  | 0000<br>0001<br>0010<br>0011<br>0100<br>0101<br>0110<br>0111<br>1000<br>1001<br>1010<br>1011<br>1100<br>1101<br>1110<br>1111 | PLA Element 0.<br>PLA Element 1.<br>PLA Element 2.<br>PLA Element 3.<br>PLA Element 4.<br>PLA Element 5.<br>PLA Element 6.<br>PLA Element 7.<br>PLA Element 8.<br>PLA Element 9.<br>PLA Element 10.<br>PLA Element 11.<br>PLA Element 12.<br>PLA Element 13.<br>PLA Element 14.<br>PLA Element 15. |
| 7 to 5   |  | Reserved.  |
| 4        |  | PLA IRQ0 enable bit.<br>This bit is set by the user to enable IRQ0 output from PLA.<br>This bit is cleared by the user to disable IRQ0 output from PLA.  |
| 3 to 0   | 0000<br>0001<br>0010<br>0011<br>0100<br>0101<br>0110<br>0111<br>1xxx   | PLA IRQ0 source.<br>PLA Element 0.<br>PLA Element 1.<br>PLA Element 2.<br>PLA Element 3.<br>PLA Element 4.<br>PLA Element 5.<br>PLA Element 6.<br>PLA Element 7.<br>Reserved.  |

Table 80. Feedback Configuration

| Bit     | Value | PLAELM0    | PLAELM1 to PLAELM7 | PLAELM8    | PLAELM9 to PLAELM15 |
|---------|-------|------------|--------------------|------------|---------------------|
| 10 to 9 | 00    | Element 15 | Element 0          | Element 7  | Element 8           |
|         | 01    | Element 2  | Element 2          | Element 10 | Element 10          |
|         | 10    | Element 4  | Element 4          | Element 12 | Element 12          |
|         | 11    | Element 6  | Element 6          | Element 14 | Element 14          |
| 8 to 7  | 00    | Element 1  | Element 1          | Element 9  | Element 9           |
|         | 01    | Element 3  | Element 3          | Element 11 | Element 11          |
|         | 10    | Element 5  | Element 5          | Element 13 | Element 13          |
|         | 11    | Element 7  | Element 7          | Element 15 | Element 15          |

**PLAADC Register**

Name: PLAADC

Address: 0xFFFF0B48

Default value: 0x00000000

Access: Read/write

Function: PLAADC is the PLA source for the ADC start conversion signal.

Table 81. PLAADC MMR Bit Descriptions

| Bit     | Value | Description   |
|---------|-------|---|
| 31 to 5 |       | Reserved.   |
| 4       |       | ADC start conversion enable bit.<br>This bit is set by the user to enable ADC start conversion from PLA.<br>This bit is cleared by the user to disable ADC start conversion from PLA. |
| 3 to 0  |       | ADC start conversion source.  |
|         | 0000  | PLA Element 0.  |
|         | 0001  | PLA Element 1.  |
|         | 0010  | PLA Element 2.  |
|         | 0011  | PLA Element 3.  |
|         | 0100  | PLA Element 4.  |
|         | 0101  | PLA Element 5.  |
|         | 0110  | PLA Element 6.  |
|         | 0111  | PLA Element 7.  |
|         | 1000  | PLA Element 8.  |
|         | 1001  | PLA Element 9.  |
|         | 1010  | PLA Element 10.   |
|         | 1011  | PLA Element 11.   |
|         | 1100  | PLA Element 12.   |
|         | 1101  | PLA Element 13.   |
|         | 1110  | PLA Element 14.   |
|         | 1111  | PLA Element 15.   |

**PLADIN Register**

Name: PLADIN

Address: 0xFFFF0B4C

Default value: 0x00000000

Access: Read/write

Function: PLADIN is a data input MMR for PLA.

Table 82. PLADIN MMR Bit Descriptions

| Bit      | Description                           |
|----------|---------------------------------------|
| 31 to 16 | Reserved.                             |
| 15 to 0  | Input bit to Element 15 to Element 0. |

**PLADOUT Register**

Name: PLADOUT

Address: 0xFFFF0B50

Default value: 0x00000000

Access: Read

Function: PLADOUT is a data output MMR for PLA.  
This register is always updated.

Table 83. PLADOUT MMR Bit Descriptions

| Bit      | Description                              |
|----------|--|
| 31 to 16 | Reserved.                                |
| 15 to 0  | Output bit from Element 15 to Element 0. |

**PLALCK Register**

Name: PLALCK

Address: 0xFFFF0B54

Default value: 0x00

Access: Write

Function: PLALCK is a PLA lock option. Bit 0 is written only once. When set, it does not allow modifying any of the PLA MMRs, except PLADIN. A PLA tool is provided in the development system to easily configure PLA.

## PULSE-WIDTH MODULATOR

### PULSE-WIDTH MODULATOR GENERAL OVERVIEW

The ADuC7023 integrates a 5-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   |
|----------|---|
| PWMCON1  | PWM Control Register 1.                               |
| 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                   |
| PWM2COM1 | Compare Register 1 for PWM Output 4                   |
| PWM2LEN  | Frequency control for PWM Output 4.                   |
| PWMCLR1  | 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 40.

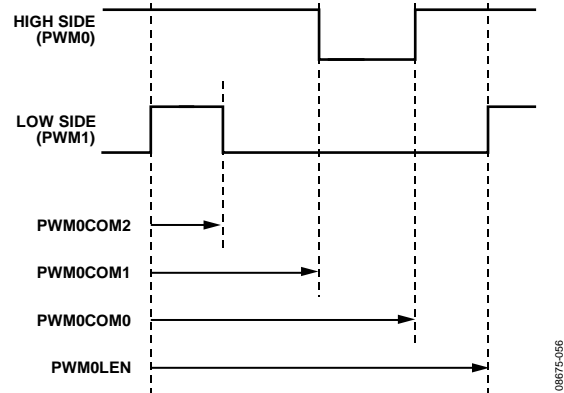


Figure 40. PWM Timing

The PWM clock is selectable via PWMCON1 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 40.

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.

#### PWMCON1 Control Register

Name: PWMCON1

Address: 0xFFFF0F80

Default value: 0x0012

Access: Read and write

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



Table 85. PWMCON1 MMR Bit Designations

| Bit    | Name       | Description   |
|--------|------------|---|
| 14     | SYNC       | Enables PWM synchronization.<br>Set to 1 by the user so that all PWM counters are reset on the next clock edge after the detection of a high-to-low transition on the P2.2/SYNC pin.<br>Cleared by the user to ignore transitions on the P2.2/SYNC pin.                     |
| 13     | Reserved   | Set to 0 by the user.   |
| 12     | PWM3INV    | Set to 1 by the user to invert PWM3.<br>Cleared by the user to use PWM3 in normal mode.   |
| 11     | PWM1INV    | Set to 1 by the user to invert PWM1.<br>Cleared by the user to use PWM1 in normal mode.   |
| 10     | PWMTRIP    | Set to 1 by the user to enable PWM trip interrupt. When the PWM trip input (Pin P1.5/PWM <sub>TRIPINPUT</sub> ) is low, the PWMEN bit is cleared and an interrupt is generated.<br>Cleared by the 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 the user to enable PWM outputs.<br>Cleared by the user to disable PWM outputs.<br>If HOFF = 1 and HMODE = 1, see Table 86.  |
| 8 to 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 the user to invert all PWM outputs.<br>Cleared by the user to use PWM outputs as normal.  |
| 4      | HOFF       | High side off.<br>Set to 1 by the user to force PWM0 and PWM2 outputs high. This also forces PWM1 and PWM3 low.<br>Cleared by the user to use the PWM outputs as normal.  |
| 3      | LCOMP      | Load compare registers.<br>Set to 1 by the 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 the user to use the values previously stored in the internal compare registers. |
| 2      | DIR        | Direction control.<br>Set to 1 by the user to enable PWM0 and PWM1 as the output signals while PWM2 and PWM3 are held low.<br>Cleared by the 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 the user to enable H-bridge mode.<br>Cleared by the user to operate the PWMs in standard mode.   |
| 0      | PWMEN      | Set to 1 by the user to enable all PWM outputs.<br>Cleared by the 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, PWMCON1 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**

| PWMCON1 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    |

**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/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/write  
Function: PWM4 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/write  
Function: PWM2LEN defines the period of PWM4.

**PWMCLRI Register**

Name: PWMCLRI  
Address: 0xFFFF0FB8  
Default value: 0x0000  
Access: Write  
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.

## PROCESSOR REFERENCE PERIPHERALS

### INTERRUPT SYSTEM

There are 22 interrupt sources on the ADuC7023 that are controlled by the interrupt controller. Most interrupts are generated from the on-chip peripherals, such as ADC. Four additional interrupt sources are generated from external interrupt request pins, IRQ0, IRQ1, IRQ2, and IRQ3. The ARM7TDMI CPU core only recognizes interrupts as one of two types, 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 is managed through nine interrupt related registers, four dedicated to IRQ, and four dedicated to FIQ. An additional MMR is used to select the programmed interrupt source. The bits in each IRQ and FIQ registers represent the same interrupt source as described in Table 88.

The ADuC7023 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 is 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.

IRQSTA/FIQSTA should be saved immediately upon entering the interrupt service routine (ISR) to ensure that all valid interrupt sources are serviced.

**Table 88. IRQ/FIQ MMRs Bit Description**

| Bit | Description                      |
|-----|----------------------------------|
| 0   | All interrupts OR'ed (FIQ only). |
| 1   | SWI.                             |
| 2   | Timer0.                          |
| 3   | Timer1.                          |
| 4   | Watchdog timer (Timer 2).        |
| 5   | Flash control.                   |
| 6   | ADC channel.                     |
| 7   | PLL lock.                        |
| 8   | I <sup>2</sup> C0 master.        |
| 9   | I <sup>2</sup> C0 slave.         |
| 10  | I <sup>2</sup> C1 master.        |
| 11  | I <sup>2</sup> C1 slave.         |
| 12  | SPI.                             |
| 13  | External IRQ0.                   |
| 14  | Comparator.                      |
| 15  | PSM.                             |
| 16  | External IRQ1.                   |
| 17  | PLA IRQ0.                        |
| 18  | External IRQ2.                   |
| 19  | External IRQ3.                   |
| 20  | PLA IRQ1.                        |
| 21  | PWM.                             |

### IRQ

The interrupt request (IRQ) is the exception signal to enter the IRQ mode of the processor. It is used to service general-purpose interrupt handling of internal and external events.

The four 32-bit registers dedicated to IRQ are: IRQSTA, IRQSIG, IRQEN, and IRQCLR.

#### IRQSTA Register

|                |  |
|----------------|--|
| Name:          | IRQSTA   |
| Address:       | 0xFFFF0000   |
| Default value: | 0x00000000   |
| Access:        | Read   |
| Function:      | IRQSTA (read-only register) provides the current-enabled IRQ source status. 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. All 32 bits are logically OR'ed to create the IRQ signal to the ARM7TDMI core. |

#### IRQSIG Register

|                |  |
|----------------|--|
| Name:          | IRQSIG   |
| Address:       | 0xFFFF0004   |
| Default value: | 0x00XXX000   |
| Access:        | Read   |
| Function:      | 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 are cleared when the interrupt in the particular peripheral is cleared. All IRQ sources can be masked in the IRQEN MMR. IRQSIG is read-only. |

**IRQEN Register**

|                |  |
|----------------|--|
| Name:          | IRQEN  |
| Address:       | 0xFFFF0008   |
| Default value: | 0x00000000   |
| Access:        | Read/write   |
| Function:      | <p>IRQEN provides the value of the current enable mask. When each bit is set to 1, the source request is enabled to create an IRQ exception. When each bit is set to 0, the source request is disabled or masked, which does not create an IRQ exception.</p> <p>To clear an already enabled interrupt source, users must set the appropriate bit in the IRQCLR register. Clearing an interrupt IRQEN bit does not disable this interrupt.</p> |

**IRQCLR Register**

|                |  |
|----------------|--|
| Name:          | IRQCLR   |
| Address:       | 0xFFFF000C   |
| Default value: | 0x00000000   |
| Access:        | Write  |
| Function:      | <p>IRQCLR (write-only register) clears the IRQEN register to mask an interrupt source. Each bit set to 1 clears the corresponding bit in the IRQEN register without affecting the remaining bits. The pair of registers, IRQEN and IRQCLR, independently manipulate the enable mask without requiring an atomic read-modify-write.</p> |

**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 ORed 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: | 0x00000000 |
| 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/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.

This register should only be used to disable an interrupt source when in the interrupt sources interrupt service routine or if the peripheral is temporarily disabled by its own control register.

This register should not be used to disable an IRQ source if that IRQ source has an interrupt pending or could have an interrupt pending.

**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.

The 32-bit register dedicated to software interrupt is SWICFG described in Table 89. This MMR allows the control of a programmed source interrupt.

**Table 89. SWICFG MMR Bit Designations**

| Bit     | Description   |
|---------|---|
| 31 to 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 and FIQSTA registers.

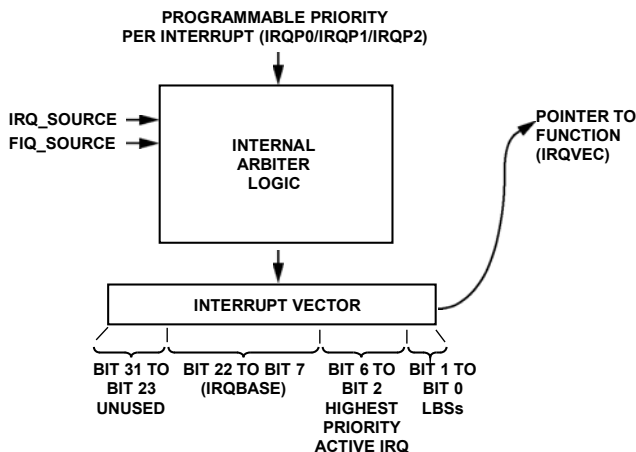


Figure 41. Interrupt Structure

**VECTORED INTERRUPT CONTROLLER (VIC)**

The ADuC7023 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 allow 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, then it is possible to have 16 separate interrupt levels.
- Programmable interrupt priorities, using the IRQP0 to IRQP2 registers, can be assigned an interrupt priority level value between 0 and 7.

**VIC MMRs**

**IRQBASE Register**

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.

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

**Table 90. IRQBASE MMR Bit Designations**

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

**IRQVEC Register**

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 only be read 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 91. IRQVEC MMR Bit Designations**

| Bit      | Type      | Initial Value | Description  |
|----------|-----------|---------------|--|
| 31 to 23 | Read only | 0             | Always read as 0.  |
| 22 to 7  | R/W       | 0             | IRQBASE register value.  |
| 6 to 2   | Read only | 0             | Highest priority source. This is a value between 0 and 21 representing the possible interrupt sources. For example, if the highest currently active IRQ is Timer 2, then these bits are [00100]. |
| 1 to 0   | Reserved  | 0             | Reserved bits.   |

**Priority Registers**

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 only be read when an IRQ occurs and IRQ interrupt nesting has been enabled by setting Bit 0 of the IRQCONN register.



**IRQP0 Register**

Name: IRQP0

Address: 0xFFFF0020

Default value: 0x00000000

Access: Read and write

**Table 92. IRQP0 MMR Bit Designations**

| Bit      | Name     | Description  |
|----------|----------|--|
| 31       | Reserved | Reserved bit   |
| 30 to 28 | PLLPI    | A priority level of 0 to 7 can be set for PLL lock interrupt.                    |
| 27       | Reserved | Reserved bit   |
| 26 to 24 | ADCPI    | A priority level of 0 to 7 can be set for the ADC interrupt source.              |
| 23       | Reserved | Reserved bit   |
| 22 to 20 | FlashPI  | A priority level of 0 to 7 can be set for the Flash controller interrupt source. |
| 19       | Reserved | Reserved bit.  |
| 18 to 16 | T2PI     | A priority level of 0 to 7 can be set for Timer2.                                |
| 15       | Reserved | Reserved bit.  |
| 14 to 12 | T1PI     | A priority level of 0 to 7 can be set for Timer1.                                |
| 11       | Reserved | Reserved bit.  |
| 10 to 8  | TOPI     | A priority level of 0 to 7 can be set for Timer0.                                |
| 7        | Reserved | Reserved bit   |
| 6 to 4   | SWINTP   | A priority level of 0 to 7 can be set for the software interrupt source.         |
| 3 to 0   | Reserved | Interrupt 0 cannot be prioritized.   |

**IRQP1 Register**

Name: IRQP1

Address: 0xFFFF0024

Default value: 0x00000000

Access: Read and write

**Table 93. IRQP1 MMR Bit Designations**

| Bit      | Name     | Description  |
|----------|----------|--|
| 31       | Reserved | Reserved bit.  |
| 30 to 28 | PSMPI    | A priority level of 0 to 7 can be set for the power supply monitor interrupt source. |
| 27       | Reserved | Reserved bit.  |
| 26 to 24 | COMPI    | A priority level of 0 to 7 can be set for comparator.                                |
| 23       | Reserved | Reserved bit.  |
| 22 to 20 | IRQ0PI   | A priority level of 0 to 7 can be set for IRQ0.                                      |
| 19       | Reserved | Reserved bit.  |

| Bit      | Name     | Description   |
|----------|----------|---|
| 18 to 16 | SPIPI    | A priority level of 0 to 7 can be set for SPI.                      |
| 15       | Reserved | Reserved bit.   |
| 14 to 12 | I2C1SPI  | A priority level of 0 to 7 can be set for I <sup>2</sup> C1 slave.  |
| 11       | Reserved | Reserved bit.   |
| 10 to 8  | I2C1MPI  | A priority level of 0 to 7 can be set for I <sup>2</sup> C1 master. |
| 7        | Reserved | Reserved bits.  |
| 6 to 4   | I2C0SPI  | A priority level of 0 to 7 can be set for I <sup>2</sup> C0 slave.  |
| 3        | Reserved | Reserved bits.  |
| 2 to 0   | I2C0MPI  | A priority level of 0 to 7 can be set for I <sup>2</sup> C0 master. |

**IRQP2 Register**

Name: IRQP2

Address: 0xFFFF0028

Default value: 0x00000000

Access: Read and write

**Table 94. IRQP2 MMR Bit Designations**

| Bit      | Name     | Description   |
|----------|----------|---|
| 31 to 23 | Reserved | Reserved bit.                                       |
| 22 to 20 | PWMPI    | A priority level of 0 to 7 can be set for PWM.      |
| 19       | Reserved | Reserved bit.                                       |
| 18 to 16 | PLA1PI   | A priority level of 0 to 7 can be set for PLA IRQ1. |
| 15       | Reserved | Reserved bit.                                       |
| 14 to 12 | IRQ3PI   | A priority level of 0 to 7 can be set for IRQ3.     |
| 11       | Reserved | Reserved bit.                                       |
| 10 to 8  | IRQ2PI   | A priority level of 0 to 7 can be set for IRQ2.     |
| 7        | Reserved | Reserved bit.                                       |
| 6 to 4   | PLA0PI   | A priority level of 0 to 7 can be set for PLA IRQ0. |
| 3        | Reserved | Reserved bit.                                       |
| 2 to 0   | IRQ1PI   | A priority level of 0 to 7 can be set for IRQ1.     |

**IRQCONN Register**

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, then FIQs and IRQs may 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.

Name: IRQCONN  
Address: 0xFFFF0030  
Default value: 0x00000000  
Access: Read and write

**Table 95. IRQCONN MMR Bit Designations**

| Bit     | Name     | Description   |
|---------|----------|---|
| 31 to 2 | Reserved | These bits are reserved and should not be written to.   |
| 1       | ENFIQN   | This bit is set to 1 to enable nesting of FIQ interrupts. This bit is cleared to mean no nesting or prioritization of FIQs is allowed.        |
| 0       | ENIRQN   | This bit is set to 1 to enable nesting of IRQ interrupts. When this bit is cleared, it means no nesting or prioritization of IRQs is allowed. |

**IRQSTAN Register**

If IRQCONN Bit 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. If the IRQ is of 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 only possible to clear one bit at a time. For example, if this register is set to 0x09, then writing 0xFF changes the register to 0x08, and writing 0xFF a second time changes the register to 0x00.

Name: IRQSTAN  
Address: 0xFFFF003C  
Default value: 0x00000000  
Access: Read and write

**Table 96. IRQSTAN MMR Bit Designations**

| Bit     | Name     | Description   |
|---------|----------|---|
| 31 to 8 | Reserved | These bits are reserved and should not be written to.   |
| 7 to 0  |          | This bit is set to 1 to enable nesting of FIQ interrupts. When this bit is cleared, it means no nesting or prioritization of FIQs is allowed. |

**FIQVEC Register**

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 only be read when an FIQ occurs and FIQ interrupt nesting has been enabled by setting Bit 1 of the IRQCONN register.

Name: FIQVEC  
Address: 0xFFFF011C  
Default value: 0x00000000  
Access: Read only

**Table 97. FIQVEC MMR Bit Designations**

| Bit      | Type      | Initial Value | Description   |
|----------|-----------|---------------|---|
| 31 to 23 | Read only | 0             | Always read as 0.   |
| 22 to 7  | R/W       | 0             | IRQBASE register value.   |
| 6 to 2   |           | 0             | Highest priority source. This is a value between 0 and 27 that represents the possible interrupt sources. For example, if the highest currently active FIQ is Timer 2, then these bits are [00100]. |
| 1 to 0   | Reserved  | 0             | Reserved bits.  |

**FIQSTAN Register**

If IRQCONN Bit 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, then Bit 0 asserts. If the FIQ is of 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 only possible to clear one bit at a time. For example, if this register is set to 0x09, then writing 0xFF changes the register to 0x08 and writing 0xFF a second time changes the register to 0x00.

Name: FIQSTAN

Address: 0xFFFF013C

Default value: 0x00000000

Access: Read/write

**Table 98. FIQSTAN MMR Bit Designations**

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

**External Interrupts and PLA interrupts**

The ADuC7023 provides up to four external interrupt sources and two PLA interrupt sources. These external interrupts can be individually configured as level or rising/falling edge triggered.

To enable the external interrupt source or the PLA interrupt source, the appropriate bit must 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 or an edge-based PLA interrupt, set the appropriate bit in the IRQCLRE register.

**IRQCONE Register**

Name: IRQCONE

Address: 0xFFFF0034

Default value: 0x00000000

Access: Read and write

**Table 99. IRQCONE MMR Bit Designations**

| Bit      | Value | Name         | Description   |
|----------|-------|--------------|---|
| 31 to 12 |       | Reserved     | These bits are reserved and should not be written to. |
| 11 to 10 | 11    | PLA1SRC[1:0] | PLA IRQ1 triggers on falling edge.                    |
|          | 10    |              | PLA IRQ1 triggers on rising edge.                     |
|          | 01    |              | PLA IRQ1 triggers on low level.                       |
|          | 00    |              | PLA IRQ1 triggers on high level.                      |
| 9 to 8   | 11    | IRQ3SRC[1:0] | External IRQ3 triggers on falling edge.               |
|          | 10    |              | External IRQ3 triggers on rising edge.                |
|          | 01    |              | External IRQ3 triggers on low level.                  |
|          | 00    |              | External IRQ3 triggers on high level.                 |
| 7 to 6   | 11    | IRQ2SRC[1:0] | External IRQ2 triggers on falling edge.               |
|          | 10    |              | External IRQ2 triggers on rising edge.                |
|          | 01    |              | External IRQ2 triggers on low level.                  |
|          | 00    |              | External IRQ2 triggers on high level.                 |
| 5 to 4   | 11    | PLA0SRC[1:0] | PLA IRQ0 triggers on falling edge.                    |
|          | 10    |              | PLA IRQ0 triggers on rising edge.                     |
|          | 01    |              | PLA IRQ0 triggers on low level.                       |
|          | 00    |              | PLA IRQ0 triggers on high level.                      |
| 3 to 2   | 11    | IRQ1SRC[1:0] | External IRQ1 triggers on falling edge.               |
|          | 10    |              | External IRQ1 triggers on rising edge.                |
|          | 01    |              | External IRQ1 triggers on low level.                  |
|          | 00    |              | External IRQ1 triggers on high level.                 |
| 1 to 0   | 11    | IRQ0SRC[1:0] | External IRQ0 triggers on falling edge.               |
|          | 10    |              | External IRQ0 triggers on rising edge.                |
|          | 01    |              | External IRQ0 triggers on low level.                  |
|          | 00    |              | External IRQ0 triggers on high level.                 |

**IRQCLRE Register**

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

**Table 100. IRQCLRE MMR Bit Designations**

| Bit      | Name     | Description   |
|----------|----------|---|
| 31 to 21 | Reserved | These bits are reserved and should not be written to.   |
| 20       | PLA1CLRI | A 1 must be written to this bit in the PLA IRQ1 interrupt service routine to clear an edge triggered PLA IRQ1 interrupt.  |
| 19       | IRQ3CLRI | A 1 must be written to this bit in the external IRQ3 interrupt service routine to clear an edge triggered IRQ3 interrupt. |
| 18       | IRQ2CLRI | A 1 must be written to this bit in the external IRQ2 interrupt service routine to clear an edge triggered IRQ2 interrupt. |
| 17       | PLA0CLRI | A 1 must be written to this bit in the PLA IRQ0 interrupt service routine to clear an edge triggered PLA IRQ0 interrupt.  |
| 16       | IRQ1CLRI | A 1 must be written to this bit in the external IRQ1 interrupt service routine to clear an edge triggered IRQ1 interrupt. |
| 15 to 14 | Reserved | These bits are reserved and should not be written to.   |
| 13       | IRQ0CLRI | A 1 must be written to this bit in the external IRQ0 interrupt service routine to clear an edge triggered IRQ0 interrupt. |
| 12 to 0  | Reserved | These bits are reserved and should not be written to.   |

**TIMERS**

The ADuC7023 has three general-purpose timer/counters: Timer0, Timer1, and Timer2 or Watchdog Timer.

These three timers in their normal mode of operation can be either free-running or periodic.

In free-running mode, the counter decreases from the maximum value until zero scale and starts again at the minimum value. (It also increases from the minimum value until full scale and starts again at the maximum 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.

The timer interval is calculated as follows.

If the timer is set to count down,

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

If the timer is set to count up,

$$Interval = \frac{(FullScale - TxLD) \times Prescaler}{Source\ Clock}$$

The value of a counter can be read at any time by accessing its value register (TxVAL). When a timer is being clocked from a clock other than core clock, an incorrect value may be read (due to asynchronous clock system). In this configuration, TxVAL should always be read twice. If the two readings are different, it should be read a third time to get the correct value.

Timers are started by writing in the control register of the corresponding timer (TxCON).

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

When using an asynchronous clock-to-clock timer, the interrupt in the timer block can take more time to clear than the time it takes for the code in the interrupt routine to execute. Ensure that the interrupt signal is cleared before leaving the interrupt service routine. This can be done by checking the IRQSTA MMR.

**Hours, Minutes, Seconds, and 1/128 Format**

To use the timer in hours, minutes, seconds, and hundreds format, select the 32768 kHz clock and a prescaler of 256. The hundreds field does not represent milliseconds but 1/128 of a seconds (256/32,768). The bits representing the hour, minute, and second are not consecutive in the register. This arrangement applies to T1LD and T1VAL when using the Hr:Min:Sec:hundreds format as set in T1CON[5:4]. See Table 101 for more details.

**Table 101. Hours, Minutes, Seconds, and 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 (RTOS Timer)**

Timer0 is a general-purpose, 16-bit timer (count-down) with a programmable prescaler (see Figure 42). The prescaler source is the core clock frequency (HCLK) and can be scaled by factors of 1, 16, or 256.

Timer0 can be used to start ADC conversions as shown in the block diagram in Figure 42.

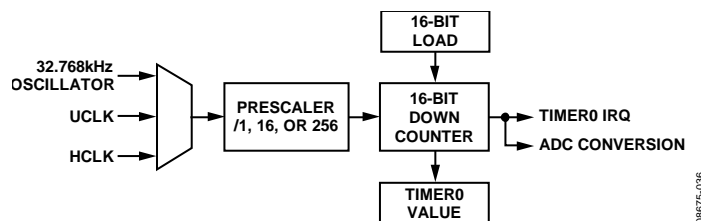


Figure 42. Timer0 Block Diagram

The Timer0 interface consists of four MMRs: TOLD, T0VAL, T0CON, and T0CLRI.

**TOLD Register**

Name: TOLD  
 Address: 0xFFFF0300  
 Default value: 0x0000  
 Access: Read/write

TOLD is a 16-bit load register that holds the 16-bit value that is loaded into the counter.

**T0VAL Register**

Name: T0VAL  
 Address: 0xFFFF0304  
 Default Value: 0xFFFF  
 Access: Read

T0VAL is a 16-bit read-only register representing the current state of the counter.

**T0CON Register**

Name: T0CON  
 Address: 0xFFFF0308  
 Default value: 0x0000  
 Access: R/W

T0CON is the configuration MMR described in Table 102.

**Table 102. T0CON MMR Bit Descriptions**

| Bit     | Value                | Description  |
|---------|----------------------|--|
| 15 to 8 |                      | Reserved.  |
| 7       |                      | Timer0 enable bit. This bit is set by the user to enable Timer0. This bit is cleared by the user to disable Timer0 by default.                       |
| 6       |                      | Timer0 mode. This bit is set by the user to operate in periodic mode. This bit is cleared by the user to operate in free-running mode. Default mode. |
| 5 to 4  | 00<br>01<br>10<br>11 | Clock select bits.<br>HCLK.<br>UCLK.<br>Internal 32768 Hz oscillator.<br>Reserved.   |
| 3 to 2  | 00<br>01<br>10<br>11 | Source clock/1. Default value.<br>Source clock/16.<br>Source clock/256.<br>Undefined. Equivalent to 00.  |
| 1 to 0  |                      | Reserved.  |

**T0CLRI Register**

Name: T0CLRI  
 Address: 0xFFFF030C  
 Default value: 0xXX  
 Access: Write

T0CLRI is an 8-bit register. Writing any value to this register clears the interrupt.

The following is the recommended procedure for servicing the Timer 0 interrupt:

```
void IRQ_Handler(void) __irq
{
    if(IRQSTA & BIT2) // Timer0 IRQ?
    {
        T0CLRI = 0; //clear Timer0 interrupt
        T0CON = 0x00; //disable Timer0 interrupt
        T0CON = 0xC8; //enable Timer0 interrupt
    }
}
```

**Timer1 (General-Purpose Timer)**

Timer1 is a general-purpose, 32-bit timer (count down or count up) with a programmable prescaler. The source can be the 32 kHz external crystal, the undivided system, the core clock, or P1.1 (maximum frequency 44 MHz). This source can be scaled by a factor of 1, 16, 256, or 32,768.

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

Timer1 has a capture register (T1CAP) that can be triggered by a selected IRQ source initial assertion. This feature can be used to determine the assertion of an event more accurately than the precision allowed by the RTOS timer when the IRQ is serviced.

Timer1 can be used to start ADC conversions as shown in the block diagram in Figure 43.

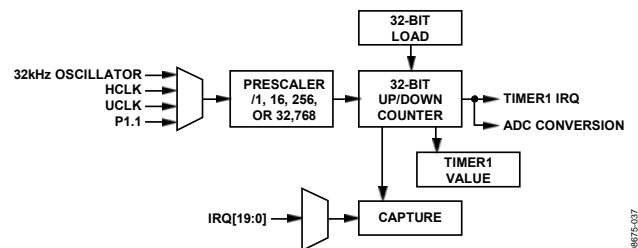


Figure 43. Timer1 Block Diagram

The Timer1 interface consists of five MMRs: T1LD, T1VAL, T1CON, T1CLRI, and T1CAP.

**T1LD Register**

Name: T1LD  
 Address: 0xFFFF0320  
 Default value: 0x00000000  
 Access: Read/write

T1LD is a 32-bit load register that holds the 32-bit value that is loaded into the counter.

**T1VAL Register**

Name: T1VAL  
 Address: 0xFFFF0324  
 Default value: 0xFFFFFFFF  
 Access: Read

T1VAL is a 32-bit read-only register that represents the current state of the counter.

**T1CON Register**

Name: T1CON  
 Address: 0xFFFF0328  
 Default value: 0x00000000  
 Access: Read/write

T1CON is the configuration MMR described in Table 103.

**Table 103. T1CON MMR Bit Descriptions**

| Bit      | Value                    | Description  |
|----------|--------------------------|--|
| 31 to 18 |                          | Reserved.  |
| 17       |                          | Event select bit. This bit is set by the user to enable time capture of an event. This bit is cleared by the user to disable time capture of an event.                         |
| 16 to 12 |                          | Event select range, 0 to 31. These events are as described in Table 88. All events are offset by two, that is, Event 2 in Table 88 becomes Event 0 for the purposes of Timer1. |
| 11 to 9  | 000<br>001<br>010<br>011 | Clock select.<br>Core clock (HCLK).<br>Internal 32.768 kHz crystal<br>UCLK<br>P1.1 raising edge triggered.   |
| 8        |                          | Count up. This bit is set by the user for Timer1 to count up. This bit is cleared by the user for Timer1 to count down by default.   |
| 7        |                          | Timer1 enable bit. This bit is set by the user to enable Timer1. This bit is cleared by the user to disable Timer1 by default.   |

| Bit    | Value                        | Description  |
|--------|------------------------------|--|
| 6      |                              | Timer1 mode. This bit is set by the user to operate in periodic mode. This bit is cleared by the user to operate in free-running mode. Default mode.       |
| 5 to 4 | 00<br>01<br>10<br>11         | Format.<br>Binary.<br>Reserved.<br>Hours, minutes, seconds, hundredths (23 hours to 0 hour).<br>Hours, minutes, seconds, hundredths (255 hours to 0 hour). |
| 3 to 0 | 0000<br>0100<br>1000<br>1111 | Prescale.<br>Source clock/1.<br>Source clock/16.<br>Source clock/256.<br>Source clock/32,768.  |

**T1CLRI Register**

Name: T1CLRI  
 Address: 0xFFFF032C  
 Default value: 0xFF  
 Access: Write

T1CLRI is an 8-bit register. Writing any value to this register clears the Timer1 interrupt.

**T1CAP Register**

Name: T1CAP  
 Address: 0xFFFF0330  
 Default value: 0x00000000  
 Access: Read

T1CAP is a 32-bit register. It holds the value contained in T1VAL when a particular event occurs. This event must be selected in T1CON.

**Timer2 (Watchdog Time)**

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 processor reset.

**Normal Mode**

Timer2 in normal mode is identical to Timer0, except for the clock source and the count-up functionality. The clock source is 32 kHz from the PLL and can be scaled by a factor of 1, 16, or 256 (see Figure 44).

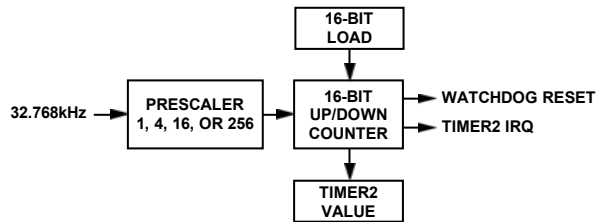


Figure 44. Timer2 Block Diagram

**Watchdog Mode**

Watchdog mode is entered by setting Bit 5 in the T2CON MMR. Timer2 decreases from the value present in the T2LD register until 0. T2LD is used as the timeout. The maximum timeout can be 512 sec using the prescaler/256, and full-scale in T2LD. Timer3 is clocked by the internal 32 kHz crystal when operating in the watchdog mode. To enter watchdog mode successfully, Bit 5 in the T2CON MMR must be set after writing to the T2LD MMR.

If the timer reaches 0, a reset or an interrupt occurs, depending on Bit 1 in the T2CON register. To avoid reset or interrupt, any value must be written to T2CLRI before the expiration period. 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 reset clears the watchdog enable bit, which causes Timer2 to exit watchdog mode.

The Timer2 interface consists of four MMRs: T2LD, T2VAL, T2CON, and T2CLRI.

**T2LD Register**

Name: T2LD  
 Address: 0xFFFF0360  
 Default value: 0x0000  
 Access: Read/write

T2LD is a 16-bit register load register that holds the 16-bit value that is loaded into the counter.

**T2VAL Register**

Name: T2VAL  
 Address: 0xFFFF0364  
 Default value: 0xFFFF  
 Access: Read

T2VAL is a 16-bit read-only register that represents the current state of the counter.

**T2CON Register**

Name: T2CON  
 Address: 0xFFFF0368  
 Default value: 0x0000  
 Access: Read/write

T2CON is the configuration MMR described in Table 104.

**Table 104. T2CON MMR Bit Descriptions**

| Bit     | Value                | Description   |
|---------|----------------------|---|
| 15 to 9 |                      | Reserved.   |
| 8       |                      | Count up. This bit is set by the user for Timer2 to count up. This bit is cleared by the user for Timer2 to count down by default.  |
| 7       |                      | Timer2 enable bit. This bit is set by the user to enable Timer2. This bit is cleared by user to disable Timer2 by default.  |
| 6       |                      | Timer2 mode. This bit is set by user to operate in periodic mode. This bit is cleared by the user to operate in free-running mode. Default mode.                                  |
| 5       |                      | Watchdog mode enable bit. This bit is set by the user to enable watchdog mode. This bit is cleared by the user to disable watchdog mode by default.                               |
| 4       |                      | Secure clear bit. This bit is set by the user to use the secure clear option. This bit is cleared by the user to disable the secure clear option by default.                      |
| 3 to 2  | 00<br>01<br>10<br>11 | Prescale. Source clock/1 by default. Source clock/16. Source clock/256. Undefined. Equivalent to 00.  |
| 1       |                      | Watchdog IRQ Option Bit. This bit is set by the user to produce an IRQ instead of a reset when the watchdog reaches 0. This bit is cleared by the user to disable the IRQ option. |
| 0       |                      | Reserved.   |

**T2CLRI Register**

Name: T2CLRI  
 Address: 0xFFFF036C  
 Default value: 0xXX  
 Access: Write

T2CLRI is an 8-bit register. Writing any value to this register on successive occasions clears the Timer2 interrupt in normal mode or resets a new timeout period in watchdog mode.

The user must perform successive writes to this register to ensure resetting the timeout period.

**Secure Clear Bit (Watchdog Mode Only)**

The secure clear bit is provided for a higher level of protection. When set, a specific sequential value must be written to T2CLRI to avoid a watchdog reset. The value is a sequence generated by the 8-bit linear feedback shift register (LFSR) polynomial =  $X^8 + X^6 + X^5 + X + 1$  shown in Figure 45.

The initial value or seed is written to T2CLRI before entering watchdog mode. After entering watchdog mode, a write to T2CLRI must match this expected value. If it matches, the LFSR is advanced to the next state when the counter reload happens. If it fails to match the expected state, a reset is immediately generated, even if the count has not yet expired.

The value 0x00 should not be used as an initial seed due to the properties of the polynomial. The value 0x00 is always guaranteed to force an immediate reset. The value of the LFSR cannot be read; it must be tracked/generated in software.

An example of a sequence follows:

1. Enter initial seed, 0xAA, in T2CLRI before starting Timer2 in watchdog mode.
2. Enter 0xAA in T2CLRI; Timer2 is reloaded.
3. Enter 0x37 in T2CLRI; Timer2 is reloaded.
4. Enter 0x6E in T2CLRI; Timer2 is reloaded.
5. Enter 0x66. 0xDC was expected; the watchdog resets the chip.

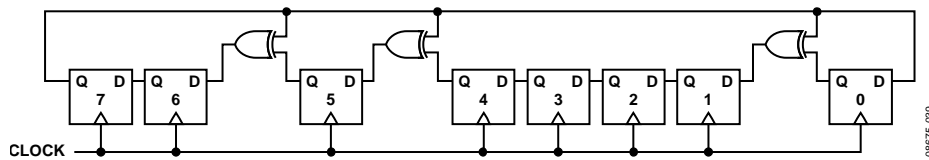


Figure 45. 8-Bit LFSR

018575-03B



## HARDWARE DESIGN CONSIDERATIONS

### POWER SUPPLIES

The ADuC7023 operational power supply voltage range is 2.7 V to 3.6 V. Separate analog and digital power supply pins ( $AV_{DD}$  and  $IOV_{DD}$ , respectively) allow  $AV_{DD}$  to be kept relatively free of noisy digital signals often present on the system  $IOV_{DD}$  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 an  $IOV_{DD}$  voltage level of 3.3 V while the  $AV_{DD}$  level can be at 3 V, or vice versa. A typical split supply configuration is shown in Figure 46.

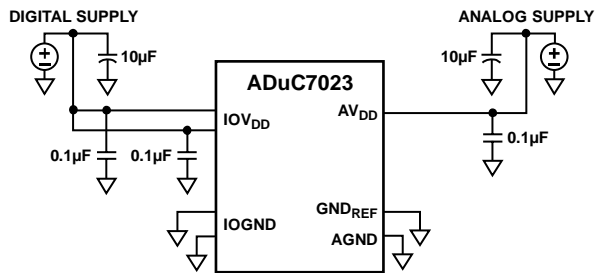


Figure 46. External Dual Supply Connections

As an alternative to providing two separate power supplies, the user can reduce noise on  $AV_{DD}$  by placing a small series resistor and/or ferrite bead between  $AV_{DD}$  and  $IOV_{DD}$ , and then decoupling  $AV_{DD}$  separately to ground. An example of this configuration is shown in Figure 47. With this configuration, other analog circuitry (such as op amps, voltage reference, and others) can be powered from the  $AV_{DD}$  supply line as well.

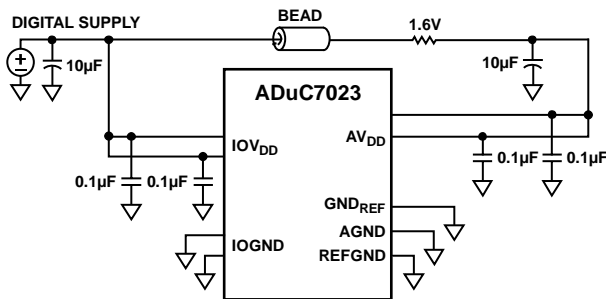


Figure 47. External Single Supply Connections

In both Figure 46 and Figure 47, a large value (10  $\mu\text{F}$ ) reservoir capacitor sits on  $IOV_{DD}$ , and a separate 10  $\mu\text{F}$  capacitor sits on  $AV_{DD}$ . In addition, local small-value (0.1  $\mu\text{F}$ ) capacitors are located at each  $AV_{DD}$  and  $IOV_{DD}$  pin of the chip. As per standard design practice, include all of these capacitors and ensure the smaller capacitors are close to each  $AV_{DD}$  pin with trace lengths as short as possible. Connect the ground terminal of each of these capacitors directly to the underlying ground plane.

Finally, the analog and digital ground pins on the ADuC7023 must be referenced to the same system ground reference point at all times.

### $IOV_{DD}$ Supply Sensitivity

The  $IOV_{DD}$  supply is sensitive to high frequency noise because it is the supply source for the internal oscillator and PLL circuits. When the internal PLL loses lock, the clock source is removed by a gating circuit from the CPU, and the ARM7TDMI core stops executing code until the PLL regains lock. This feature is to ensure that no flash interface timings or ARM7TDMI timings are violated.

Typically, frequency noise greater than 50 kHz and 50 mV p-p on top of the supply causes the core to stop working.

If decoupling values recommended in the Power Supplies section do not sufficiently dampen all noise sources below 50 mV on  $IOV_{DD}$ , a filter such as the one shown in Figure 48 is recommended.

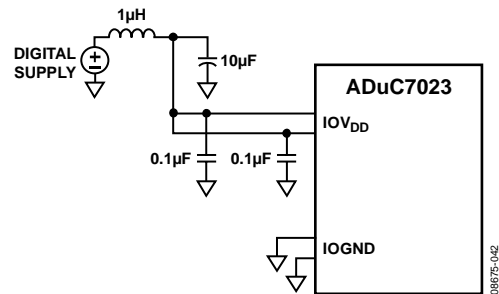


Figure 48. Recommended  $IOV_{DD}$  Supply Filter

### Linear Voltage Regulator

Each ADuC7023 requires a single 3.3 V supply, but the core logic requires a 2.6 V supply. An on-chip linear regulator generates the 2.6 V from  $IOV_{DD}$  for the core logic. The  $LV_{DD}$  pin is the 2.6 V supply for the core logic. An external compensation capacitor of 0.47  $\mu\text{F}$  must be connected between  $LV_{DD}$  and DGND (as close as possible to these pins) to act as a tank of charge, as shown in Figure 49.

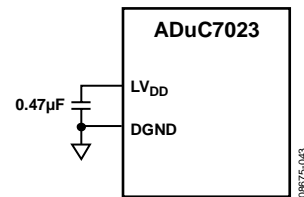


Figure 49. Voltage Regulator Connections

The  $LV_{DD}$  pin should not be used for any other chip. It is also recommended to use excellent power supply decoupling on  $IOV_{DD}$  to help improve line regulation performance of the on-chip voltage regulator.

**GROUNDING AND BOARD LAYOUT RECOMMENDATIONS**

As with all high resolution data converters, special attention must be paid to grounding and PC board layout of the ADuC7023-based designs to achieve optimum performance from the ADCs and DACs.

Although the parts have separate pins for analog and digital ground (AGND and DGND), the user must not tie these to two separate ground planes unless the two ground planes are connected very close to the part. This is illustrated in the simplified example shown in Figure 50a. In systems where digital and analog ground planes are connected together somewhere else (at the system power supply, for example), the planes cannot be reconnected near the part because a ground loop would result. In these cases, tie all the ADuC7023 AGND and DGND pins to the analog ground plane, as illustrated in Figure 50b. In systems with only one ground plane, ensure that the digital and analog components are physically separated onto separate halves of the board so that digital return currents do not flow near analog circuitry (and vice versa).

The ADuC7023 can then be placed between the digital and analog sections, as illustrated in Figure 50c.

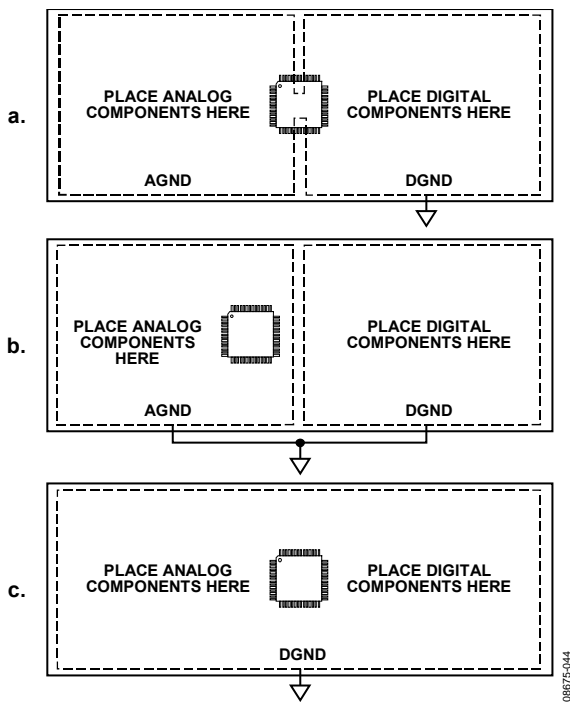


Figure 50. System Grounding Schemes

In all of these scenarios, and in more complicated real-life applications, users should pay particular attention to the flow of current from the supplies and back to ground. Make sure the return paths for all currents are as close as possible to the paths the currents took to reach their destinations.

For example, do not power components on the analog side (as seen in Figure 50b) with IOV<sub>DD</sub> because that would force return currents from IOV<sub>DD</sub> to flow through AGND. Avoid digital currents flowing under analog circuitry, which can occur if a noisy digital chip is placed on the left half of the board (shown in Figure 50c). If possible, avoid large discontinuities in the ground plane(s) such as those formed by a long trace on the same layer, because they force return signals to travel a longer path. In addition, make all connections to the ground plane directly, with little or no trace separating the pin from its via to ground.

When connecting fast logic signals (rise/fall time < 5 ns) to any of the ADuC7023 digital inputs, add a series resistor to each relevant line to keep rise and fall times longer than 5 ns at the input pins of the part. A value of 100 Ω or 200 Ω is usually sufficient enough to prevent high speed signals from coupling capacitively into the part and affecting the accuracy of ADC conversions.

**CLOCK OSCILLATOR**

The clock source for the ADuC7023 can be generated by the internal PLL or by an external clock input. To use the internal PLL, connect a 32.768 kHz parallel resonant crystal between XCLKI and XCLKO, and connect a capacitor from each pin to ground, as shown in Figure 51. The crystal allows the PLL to lock correctly to give a frequency of 41.78 MHz. If no external crystal is present, the internal oscillator is used to give a typical frequency of 41.78 MHz ± 3%.

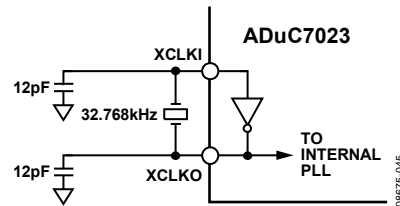


Figure 51. External Parallel Resonant Crystal Connections

To use an external source clock input instead of the PLL (see Figure 52), Bit 1 and Bit 0 of PLLCON must be modified. The external clock uses P1.1 and XCLK.

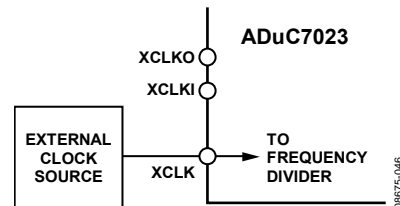


Figure 52. Connecting an External Clock Source

Using an external clock source, the ADuC7023 specified operational clock speed range is 50 kHz to 44 MHz ± 1%, which ensures correct operation of the analog peripherals and Flash/EE.

### POWER-ON RESET OPERATION

An internal power-on reset (POR) is implemented on the [ADuC7023](#). For  $LV_{DD}$  below 2.40 V typical, the internal POR holds the part in reset. As  $LV_{DD}$  rises above 2.40 V, an internal timer times out for typically 64 ms before the part is released from reset. The user must ensure that the power supply  $IOV_{DD}$  has reached a stable 2.7 V minimum level by this time. Likewise, on power-down, the internal POR holds the part in reset until  $LV_{DD}$  has dropped below 2.40 V.

Figure 53 illustrates the operation of the internal POR in detail.

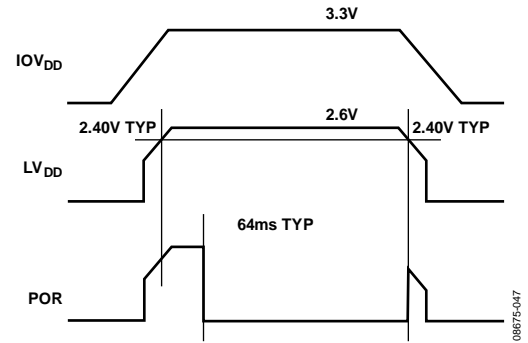


Figure 53. Internal Power-On Reset Operation

## TYPICAL SYSTEM CONFIGURATION

A typical ADuC7023 configuration is shown in Figure 54. It summarizes some of the hardware considerations. The bottom of the LFCSP package has an exposed pad that needs to be soldered to a metal plate on the board for mechanical reasons. The metal plate of the board can be connected to ground.

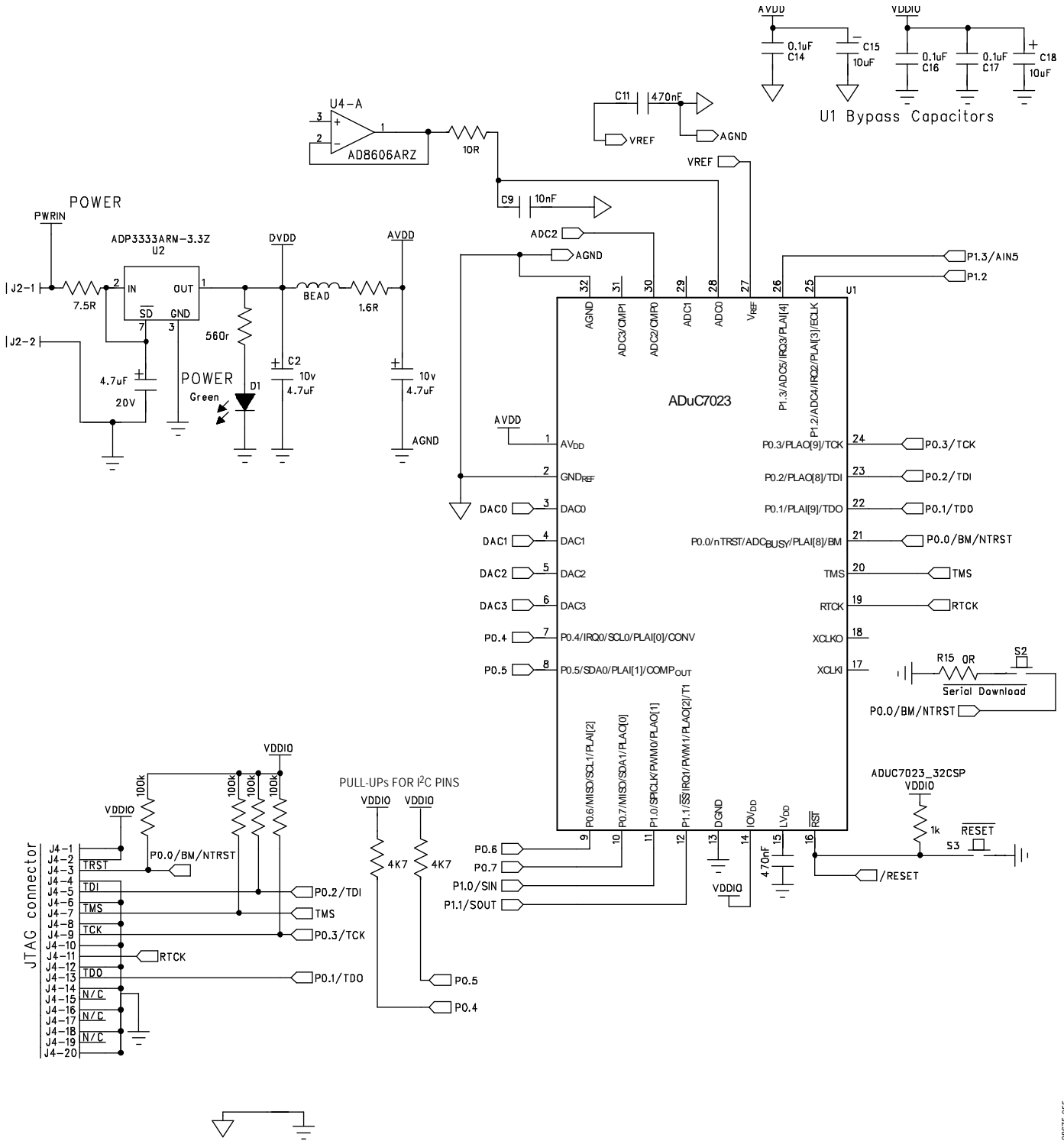


Figure 54. Typical System Configuration

## DEVELOPMENT TOOLS

### PC-BASED TOOLS

Four types of development systems are available for the [ADuC7023](#) family. The [ADuC7023](#) QuickStart Plus is intended for new users who want to have a comprehensive hardware development environment.

These systems consist of the following PC-based (Windows® compatible) hardware and software development tools.

#### **Hardware**

The hardware system uses the [ADuC7023](#) evaluation board, a serial port programming cable, and a RDI-compliant JTAG emulator (included in the [ADuC7023](#) QuickStart Plus only).

#### **Software**

The software system has an integrated development environment, incorporating an assembler, compiler, and nonintrusive JTAG-based debugger. The software system uses a serial downloader software and example code.

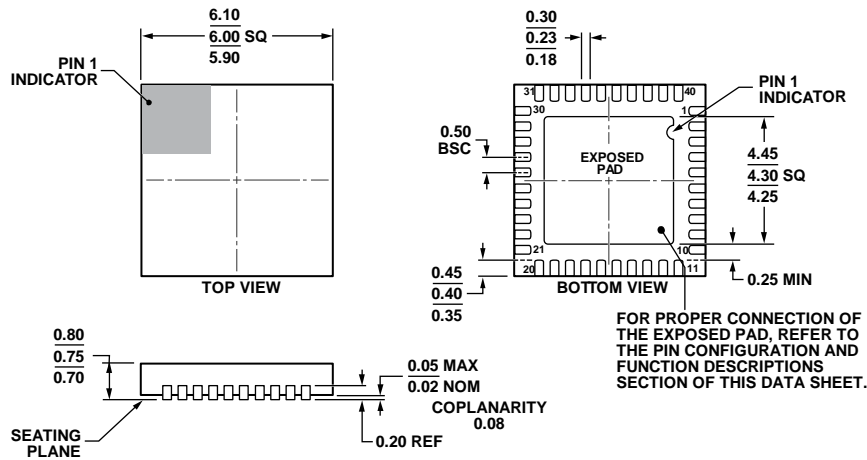
#### **Miscellaneous**

The miscellaneous systems use CD-ROM documentation.

### IN-CIRCUIT I<sup>2</sup>C DOWNLOADER

An I<sup>2</sup>C-based serial downloader is available at [www.analog.com](http://www.analog.com). This software requires an USB-to-I<sup>2</sup>C adaptor board available from Analog Devices. The part number for this USB-to-I<sup>2</sup>C adapter is USB-I2C/LIN-CONV-Z.

OUTLINE DIMENSIONS



COMPLIANT TO JEDEC STANDARDS MO-220-WJJD.

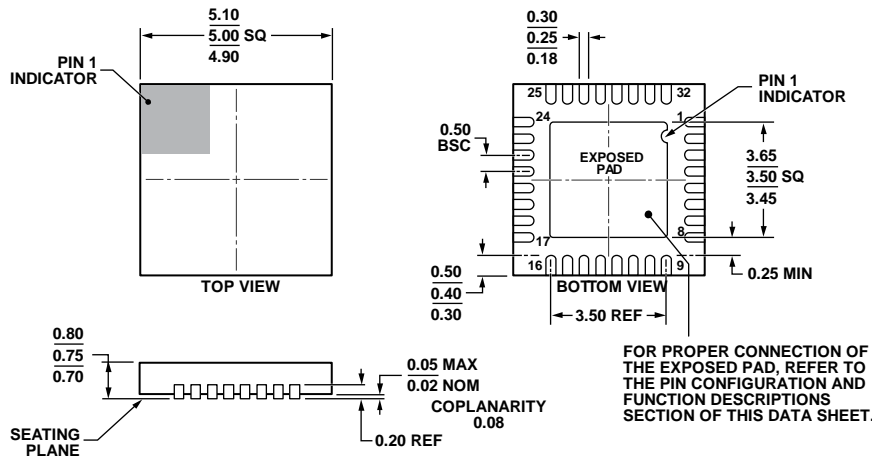
Figure 55. 40-Lead Frame Chip Scale Package [LFCSP\_WQ]

6 mm × 6 mm Body, Very Very Thin Quad

(CP-40-10)

Dimensions shown in millimeters

05-06-2011-A



COMPLIANT TO JEDEC STANDARDS MO-220-WHHD.

Figure 56. 32-Lead Lead Frame Chip Scale Package [LFCSP\_WQ]

5 mm × 5 mm Body, Very Very Thin Quad

(CP-32-11)

Dimensions shown in millimeters

04-02-2012-A

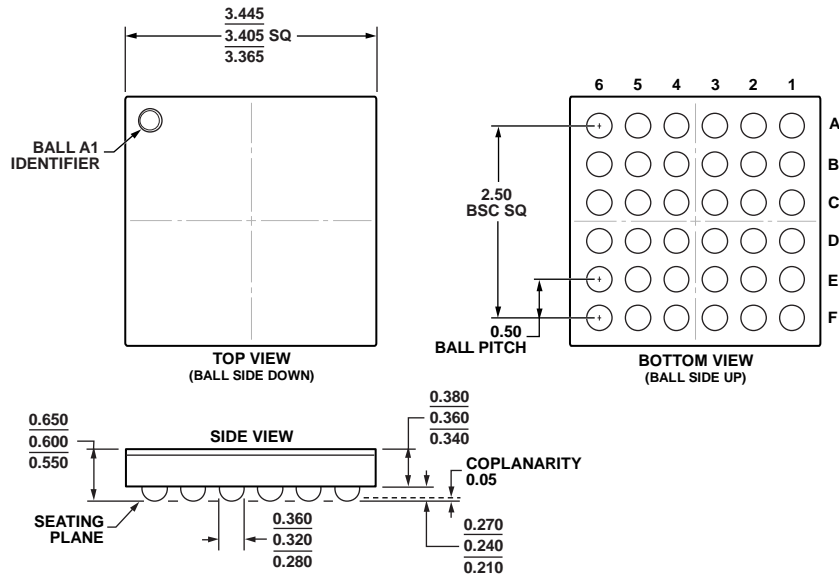


Figure 57. 36-Ball Wafer Level Chip Scale Package [WLCSP]  
(CB-36-3)  
Dimensions shown in millimeters

08-01-2012-A

## ORDERING GUIDE

| Model <sup>1</sup> | ADC Channels | DAC Channels | FLASH/ RAM | GPIO | Downloader       | Temperature Range | Package Description  | Package Option | Ordering Quantity |
|--------------------|--------------|--------------|------------|------|------------------|-------------------|--|----------------|-------------------|
| ADuC7023BCP6Z62I   | 12           | 4            | 62 kB/8 kB | 20   | I <sup>2</sup> C | −40°C to +125°C   | 40-Lead LFCSP_WQ   | CP-40-10       | 490               |
| ADuC7023BCP6Z62IRL | 12           | 4            | 62 kB/8 kB | 20   | I <sup>2</sup> C | −40°C to +125°C   | 40-Lead LFCSP_WQ   | CP-40-10       | 2,500             |
| ADuC7023BCP6Z62IR7 | 12           | 4            | 62 kB/8 kB | 20   | I <sup>2</sup> C | −40°C to +125°C   | 40-Lead LFCSP_WQ   | CP-40-10       | 750               |
| ADuC7023BCPZ62I    | 6            | 4            | 62 kB/8 kB | 12   | I <sup>2</sup> C | −40°C to +125°C   | 32-Lead LFCSP_WQ   | CP-32-11       | 490               |
| ADuC7023BCPZ62I-RL | 6            | 4            | 62 kB/8 kB | 12   | I <sup>2</sup> C | −40°C to +125°C   | 32-Lead LFCSP_WQ   | CP-32-11       | 5,000             |
| ADuC7023BCPZ62I-R7 | 6            | 4            | 62 kB/8 kB | 12   | I <sup>2</sup> C | −40°C to +125°C   | 32-Lead LFCSP_WQ   | CP-32-11       | 1,500             |
| ADuC7023BCBZ62I-R7 | 10           | 4            | 62 kB/8 kB | 16   | I <sup>2</sup> C | −40°C to +125°C   | 36-Ball WLCSP  | CB-36-03       | 1,500             |
| EVAL-ADuC7023QSPZ  |              |              |            |      |                  |                   | ADuC7023<br>QuickStart Plus<br>Development System<br>Using 32-Pin<br>ADuC7023  |                |                   |
| EVAL-ADuC7023QSPZ1 |              |              |            |      |                  |                   | ADuC7023<br>QuickStart Plus<br>Development System<br>Using 40-Pin<br>ADuC7023  |                |                   |
| EVAL-ADuC7023QSPZ2 |              |              |            |      |                  |                   | ADuC7023<br>QuickStart Plus<br>Development System<br>Using 36-Ball<br>ADuC7023 |                |                   |

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



## NOTES

I<sup>2</sup>C refers to a communications protocol originally developed by Philips Semiconductors (now NXP Semiconductors).

# Mouser Electronics

Authorized Distributor

Click to View Pricing, Inventory, Delivery & Lifecycle Information:

## [Analog Devices Inc.:](#)

[EVAL-ADUC7023QSPZ](#) [EVAL-ADUC7023QSPZ1](#) [ADUC7022BCPZ62](#) [ADUC7024BCPZ62-RL7](#)  
[ADUC7020BCPZ62-RL7](#) [ADUC7029BBCZ62-RL](#) [ADUC7023BCP6Z62I](#) [ADUC7025BCPZ32](#) [ADUC7021BCPZ32](#)  
[ADUC7021BCPZ62I](#) [ADUC7024BCPZ62](#) [ADUC7019BCPZ62I](#) [ADUC7029BBCZ62](#) [ADUC7020BCPZ62I](#)  
[ADUC7021BCPZ62-RL7](#) [ADUC7023BCPZ62I](#) [ADUC7028BBCZ62](#) [ADUC7024BCPZ62I](#) [ADUC7025BSTZ62-RL](#)  
[ADUC7026BSTZ62I-RL](#) [ADUC7022BCPZ32](#) [ADUC7022BCPZ32-RL](#) [ADUC7024BCPZ62I-RL](#) [ADUC7026BSTZ62](#)  
[ADUC7026BSTZ62-RL](#) [ADUC7025BCPZ62-RL](#) [ADUC7027BSTZ62](#) [ADUC7023BCP6Z62IRL](#) [ADUC7026BSTZ62I](#)  
[ADUC7021BCPZ62](#) [ADUC7028BBCZ62-RL](#) [ADUC7025BSTZ62](#) [ADUC7029BBCZ62I](#) [ADUC7029BBCZ62I-RL](#)  
[ADUC7020BCPZ62](#) [ADUC7021BCPZ62-RL](#) [ADUC7020BCPZ62IRL7](#) [ADUC7025BCPZ32-RL](#) [ADUC7023BCPZ62I-](#)  
[RL](#) [ADUC7025BCPZ62](#) [ADUC7021BCPZ62I-RL](#) [ADUC7020BCPZ62I-RL](#) [ADUC7021BCPZ32-RL7](#)  
[ADUC7024BSTZ62-RL](#) [ADUC7022BCPZ62-RL7](#) [ADUC7027BSTZ62-RL](#) [ADUC7019BCPZ62IRL7](#)  
[ADUC7024BSTZ62](#) [ADUC7023BCPZ62I-R7](#) [ADUC7023BCBZ62I-R7](#) [EVAL-ADUC7023QSPZ2](#)  
[ADUC7023BCP6Z62IR7](#)