# MicroConverter®, Small Package 12-Bit ADC with Embedded Flash MCU

# ADuC814

## FEATURES

**ANALOG I/O**
- 6-channel 247 kSPS ADC
  - 12-bit resolution
  - ADC high speed data capture mode
  - Programmable reference via on-chip DAC for low level inputs, ADC performance specified to $V_{REF}$ = 1 V
- Dual voltage output DACs
  - 12-bit resolution, 15 µs settling time

**Memory**
- 8 kbytes on-chip Flash/EE program memory
- 640 bytes on-chip Flash/EE data memory
- Flash/EE, 100 year retention, 100 kcycle endurance
- 3 levels of Flash/EE program memory security
- In-circuit serial downlaod (no external hardware)
- 256 bytes on-chip data RAM

**8051 based core**
- 8051 compatible instruction set
- 32 kHz external crystal, on-chip programmable PLL (16.78 MHz max)
- Three 16-bit timer/counters
- 11 programmable I/O lines
- 11 interrupt sources, 2 priority levels

**Power**
- Specified for 3 V and 5 V operation
- Normal: 3 mA @ 3 V (core CLK = 2.1 MHz)
- Power-down: 15 µA (32 kHz oscillator running)

**On-chip peripherals**
- Power-on reset circuit (no need for external POR device)
- Temperature monitor (±1.5°C accuracy)
- Precision voltage reference
- Time interval counter (wake-up/RTC timer)
- UART serial I/O
- SPI®/I²C® compatible serial I/O
- Watchdog timer (WDT), power supply monitor (PSM)

**Package and temperature range**
- 28-lead TSSOP 4.4 mm × 9.7 mm package
- Fully specified for −40°C to +125°C operation

## APPLICATIONS

Optical networking—laser power control
Base station systems—power amplifier bias control
Precision instruments, smart sensors
Battery-powered systems, precision system monitors

## FUNCTIONAL BLOCK DIAGRAM

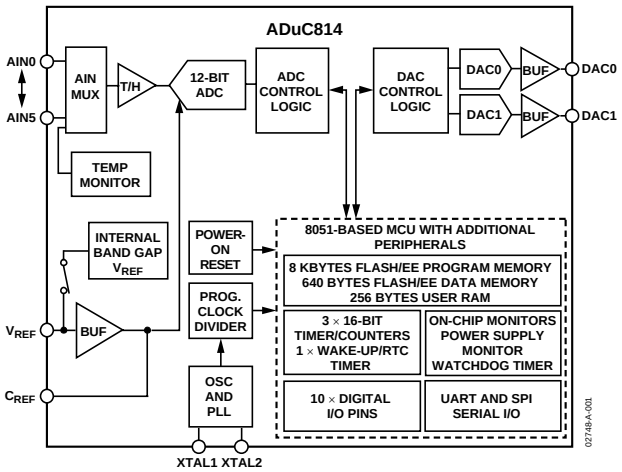



*Figure 1.*

## GENERAL DESCRIPTION

The ADuC814 is a fully integrated 247 kSPS, 12-bit data acquisition system incorporating a high performance multichannel ADC, an 8-bit MCU, and program/data Flash/EE memory on a single chip.

This low power device operates from a 32 kHz crystal with an on-chip PLL generating a high frequency clock of 16.78 MHz. This clock is, in turn, routed through a programmable clock divider from which the MCU core clock operating frequency is generated.

The microcontroller core is an 8052 and is compatible with an 8051 instruction. 8 kBytes of nonvolatile Flash/EE program memory are provided on-chip. 640 bytes of nonvolatile Flash/EE data memory and 256 bytes RAM are also integrated on-chip.

The ADuC814 also incorporates additional analog functionality with dual 12-bit DACs, a power supply monitor, and a band gap reference. On-chip digital peripherals include a watchdog timer, time interval counter, three timer/counters, and two serial I/O ports (SPI and UART).

On-chip factory firmware supports in-circuit serial download and debug modes (via UART), as well as single-pin emulation mode via the DLOAD pin. The ADuC814 is supported by a QuickStart™ Development System.

The part operates from a single 3 V or 5 V supply over the extended temperature range −40°C to +125°C. When operating from 3 V supplies, the power dissipation for the part is below 10 mW. The ADuC814 is housed in a 28-lead TSSOP package.

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

**One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.**
**Tel: 781.329.4700** **www.analog.com**
**Fax: 781.326.8703** **© 2003 Analog Devices, Inc. All rights reserved.**

# TABLE OF CONTENTS


Specifications ... 4
Absolute Maximum Ratings ... 9
ESD Caution ... 9
Pin Configuration and Function Description ... 10
Terminology ... 12
ADC Specifications ... 12
DAC Specifications ... 12
Typical Performance Curves ... 13
ADuC814 Architecture, Main Features ... 16
Memory Organization ... 17
Overview of MCU-Related SFRs ... 18
Accumulator SFR ... 18
B SFR ... 18
Stack Pointer SFR ... 18
Data Pointer ... 18
Program Status Word SFR ... 18
Power Control SFR ... 19
Special Function Registers ... 20
ADC Circuit Information ... 21
General Overview ... 21
ADC Transfer Function ... 21
ADC Data Output Format ... 21
SFR Interface to ADC Block ... 22
ADCCON1 (ADC Control SFR 1) ... 22
ADCCON2 (ADC Control SFR 2) ... 23
ADCCON3 (ADC Control SFR 3) ... 24
Driving the ADC ... 25
Voltage Reference Connections ... 26
Configuring the ADC ... 26
Initiating ADC Conversions ... 27
ADC High Speed Data Capture Mode ... 27
ADC Offset and Gain Calibration Overview ... 28
ADC Offset and Gain Calibration Coefficients ... 28
Calibrating the ADC ... 29
Initiating Calibration in Code ... 29
Nonvolitile Flash/EE Memory ... 30
Flash/EE Memory Overview ... 30
Flash/EE Memory and the ADuC814 ... 30
ADuC814 Flash/EE Memory Reliability ... 30
Using Flash/EE Program Memory ... 31
Serial Downloading (In-Circuit Programming) ... 31
Parallel Programming ... 31
Flash/EE Program Memory Security ... 31
Lock Mode ... 31
Secure Mode ... 31
Serial Safe Mode ... 31
Using Flash/EE Data Memory ... 32
ECON—Flash/EE Memory Control SFR ... 32
Flash/EE Memory Timing ... 33
Using the Flash/EE Memory Interface ... 33
Programming a Byte ... 33
User Interface to Other On-Chip ADuC814 Peripherals ... 34
DACs ... 34
Using the DACs ... 35
On-Chip PLL ... 37
Time Interval Counter (TIC) ... 38
Watchdog Timer ... 41
Power Supply Monitor ... 42
ADuC814 Configuration Register (CFG814) ... 43
Serial Peripheral Interface ... 43
External Clock ... 43

Serial Peripheral Interface....44
MISO (Master In, Slave Out Data I/O Pin)....44
MOSI (Master Out, Slave In Pin)....44
SCLOCK (Serial Clock I/O Pin)....44
$\overline{SS}$ (Slave Select Input Pin)....44
Using the SPI Interface....45
SPI Interface—Master Mode....45
SPI Interface—Slave Mode....45
$I^2C$ Compatible Interface....46
8051 Compatible On-Chip Peripherals....47
Parallel I/O Ports 1 and 3....47
Additional Digital Outputs Pins....47
Timers/Counters....48
Timer/Counter 0 and 1 Data Registers....49
Timer/Counter 0 and 1 Operating Modes....50
Mode 0 (13-Bit Timer/Counter)....50
Mode 1 (16-Bit Timer/Counter)....50
Mode 2 (8-Bit Timer/Counter with Autoreload)....50
Mode 3 (Two 8-Bit Timer/Counters)....50
Timer/Counter 2 Data Registers....51
Timer/Counter 2 Operating Modes....52
16-Bit Autoreload Mode....52
16-Bit Capture Mode....52
UART Serial Interface....53
SBUF....53
Mode 0: 8-Bit Shift Register Mode....54
Mode 1: 8-Bit UART, Variable Baud Rate....54
Mode 2: 9-Bit UART with Fixed Baud Rate....55
Mode 3: 9-Bit UART with Variable Baud Rate....55
UART Serial Port Baud Rate Generation....55
Timer 2 Generated Baud Rates....56
Interrupt System....57
Interrupt Priority....59
Interrupt Vectors....59
ADuC814 Hardware Design Considerations....60
Clock Oscillator....60
Power Supplies....60
Power Consumption....60
Power-Saving Modes....61
Power-On Reset....61
Grounding and Board Layout Recommendations....61
Other Hardware Considerations....62
In-Circuit Serial Download Access....62
Embedded Serial Port Debugger....62
Single-Pin Emulation Mode....63
Timing Specifications....64
Outline Dimensions....70
Ordering Guide....71

## REVISION HISTORY

12/03 – Data Sheet Changed from REV. 0 to REV. A

Added detailed description of product....Universal
Changes to Specifications....4
Updated Outline Dimensions....70
Changes to Ordering Guide....71

# SPECIFICATIONS

**Table 1. $AV_{DD}$ = $DV_{DD}$ = 2.7 V to 3.3 V or 4.5 V to 5.5 V, $V_{REF}$ = 2.5 V internal reference, XTAL1/XTAL2 = 32.768 kHz crystal. All specifications $T_{MIN}$ to $T_{MAX}$, unless otherwise specified[1]**

| Parameter | $V_{DD}$ = 5 V | $V_{DD}$ = 3 V | Unit | Test Conditions |
|---|---|---|---|---|
| ADC CHANNEL SPECIFICATIONS | | | | |
| **A GRADE** | | | | |
| DC ACCURACY[2,3] | | | | $f_{SAMPLE}$ = 147 kHz |
| Resolution | 12 | 12 | Bits | |
| Integral Nonlinearity | 2 | 2 | LSB max | 2.5 V internal reference |
| | 1 | 1 | LSB typ | |
| | 2.5 | 2.5 | LSB typ | 1.0 V external reference |
| Differential Nonlinearity | 4 | 4 | LSB max | 2.5 V internal reference |
| | 2 | 2 | LSB typ | |
| | 5 | 5 | LSB typ | 1.0 V external reference |
| CALIBRATED ENDPOINT ERRORS[4, 5] | | | | |
| Offset Error | 5 | 5 | LSB max | |
| Offset Error Match | 1 | 1 | LSB typ | |
| Gain Error | 5 | 5 | LSB max | |
| Gain Error Match | 1 | 1 | LSB typ | |
| DYNAMIC PERFORMANCE[6] | | | | $f_{IN}$ = 10 kHz sine wave |
| | | | | $f_{SAMPLE}$ = 147 kHz |
| Signal to Noise Ratio (SNR)[7] | 62.5 | 62.5 | dB typ | |
| Total Harmonic Distortion (THD) | –65 | –65 | dB typ | |
| Peak Harmonic or Spurious Noise | –65 | –65 | dB typ | |
| Channel-to-Channel Crosstalk[8] | –80 | –80 | dB typ | |
| **B GRADE** | | | | |
| DC ACCURACY[2, 3] | | | | $f_{SAMPLE}$ = 147 kHz |
| Resolution | 12 | 12 | Bits | |
| Integral Nonlinearity | 1 | 1 | LSB max | 2.5 V internal reference |
| | 0.3 | 0.3 | LSB typ | |
| | 1.5 | 1.5 | LSB max | 1.0 V external reference[11] |
| Differential Nonlinearity | 0.9 | 0.9 | LSB max | 2.5 V internal reference |
| | 0.25 | 0.25 | LSB typ | |
| | +1.5/–0.9 | 1.5/–0.9 | LSB max | 1.0 V external reference[11] |
| Code Distribution | 1 | 1 | LSB typ | ADC input is a dc voltage |
| CALIBRATED ENDPOINT ERRORS[4, 5] | | | | |
| Offset Error | 2 | 3 | LSB max | |
| Offset Error Match | 1 | 1 | LSB typ | |
| Gain Error | 2 | 3 | LSB max | |
| Gain Error Match | 1 | 1 | LSB typ | |
| DYNAMIC PERFORMANCE[6] | | | | $f_{IN}$ = 10 kHz sine wave |
| | | | | $f_{SAMPLE}$ = 147 kHz |
| Signal to Noise Ratio (SNR)[7] | 71 | 71 | dB typ | |
| Total Harmonic Distortion (THD) | –85 | –85 | dB typ | |
| Peak Harmonic or Spurious Noise | –85 | –85 | dB typ | |
| Channel-to-Channel Crosstalk[8] | –80 | –80 | dB typ | |
| ANALOG INPUT | | | | |
| Input Voltage Ranges | 0 to $V_{REF}$ | 0 to $V_{REF}$ | V | |
| Leakage Current | 1 | 1 | µA max | |
| Input Capacitance | 32 | 32 | pF typ | |

| Parameter | $V_{DD}$ = 5 V | $V_{DD}$ = 3 V | Unit | Test Conditions |
|---|---|---|---|---|
| TEMPERATURE MONITOR[9] | | | | |
| Voltage Output at 25°C | 650 | 650 | mV typ | |
| Voltage TC | –2 | –2 | mV/°C typ | |
| Accuracy | 3 | 3 | °C typ | 2.5 V internal reference |
| Accuracy | 1.5 | 1.5 | °C typ | 2.5 V external reference |
| DAC CHANNEL SPECIFICATIONS | | | | DAC Load to AGND RL = 10 kΩ, CL = 100 pF |
| DC ACCURACY[10] | | | | |
| Resolution | 12 | 12 | Bits | |
| Relative Accuracy | +3 | +3 | LSB typ | |
| Differential Nonlinearity[11] | –1 | –1 | LSB max | Guaranteed montonic |
| | 1/2 | 1/2 | LSB typ | |
| Offset Error | 50 | 50 | mV max | $V_{REF}$ range |
| Gain Error | 1 | 1 | % max | $V_{REF}$ range |
| | 1 | 1 | % typ | $AV_{DD}$ range |
| Gain Error Mismatch | 0.5 | 0.5 | % typ | Of full scale on DAC1 |
| ANALOG OUTPUTS | | | | |
| Voltage Range_0 | 0 to $V_{REF}$ | | Volts | DAC $V_{REF}$ = 2.5 V |
| Voltage Range_1 | 0 to $V_{DD}$ | | Volts | DAC $V_{REF}$ = $V_{DD}$ |
| Output Impedance | 0.5 | 0.5 | Ω typ | |
| $I_{SINK}$ | 50 | 50 | µA typ | |
| DAC AC Specifications | | | | |
| Voltage Output Settling Time | 15 | 15 | µs typ | Full-scale settling time to within ½ LSB of final value |
| Digital-to-Analog Glitch Energy | 10 | 10 | nVs typ | 1 LSB change at major carry |
| REFERENCE INPUT/OUTPUT | | | | |
| REFERENCE OUTPUT | | | | |
| Output Voltage ($V_{REF}$) | 2.5 | 2.5 | V | |
| Accuracy | 2.5 | 2.5 | % max | Of $V_{REF}$ measured at the $C_{REF}$ pin |
| Power Supply Rejection | 47 | 57 | dB typ | |
| Reference Tempco | 100 | 100 | ppm/°C typ | |
| Internal $V_{REF}$ Power-On Time[12] | 80 | 80 | ms typ | |
| EXTERNAL REFERENCE INPUT[13] | | | | Internal band gap reference deselected via ADCCON2.6 |
| Voltage Range ($V_{REF}$)[14] | 1.0 | 1.0 | V min | |
| | $V_{DD}$ | $V_{DD}$ | V max | |
| Input Impedance | 20 | 20 | kΩ typ | |
| Input Leakage | 10 | 10 | µA max | |
| POWER SUPPLY MONITOR (PSM) | | | | |
| $V_{DD}$ Trip Point Selection Range | 2.63 | 2.63 | V | |
| | 2.93 | 2.93 | V | Four trip points selectable in this range |
| | 3.08 | 3.08 | V | programmed via TP1–0 in PSMCON |
| | 4.63 | | V | |
| $V_{DD}$ Power Supply Trip Point Accuracy | 3.5 | 3.5 | % max | |
| WATCH DOG TIMER (WDT)[14] | | | | |
| Timeout Period | 0 | 0 | ms min | Nine time-out periods selectable in this range |
| | 2000 | 2000 | ms max | programmed via PRE3–0 in WDCON |
| LOGIC INPUTS | | | | |
| INPUT VOLTAGES[14] | | | | |
| All Inputs except SCLOCK, RESET, and XTAL1 | | | | |
| $V_{INL}$, Input Low Voltage | 0.8 | 0.4 | V max | |
| $V_{INH}$, Input High Voltage | 2.0 | 2.0 | V min | |

| Parameter | $V_{DD}$ = 5 V | $V_{DD}$ = 3 V | Unit | Test Conditions |
|---|---|---|---|---|
| SCLOCK and RESET Only[14] (Schmitt-Triggered Inputs) | | | | |
| $V_{T+}$ | 1.3 | 0.95 | V min | |
| | 3.0 | 2.5 | V max | |
| $V_{T-}$ | 0.8 | 0.4 | V min | |
| | 1.4 | 1.1 | V max | |
| $V_{T+} - V_{T-}$ | 0.3 | 0.3 | V min | |
| | 0.85 | 0.85 | V max | |
| INPUT CURRENTS | | | | |
| P1.2–P1.7, DLOAD | ±10 | ±10 | µA max | $V_{IN}$ = 0 V or $V_{DD}$ |
| SCLOCK[15] | –10 | –3 | µA min | $V_{IN}$ = 0 V, internal pull-up |
| | –40 | –15 | µA max | $V_{IN}$ = 0 V, internal pull-up |
| | ±10 | ±10 | µA max | $V_{IN}$ = $V_{DD}$ |
| RESET | ±10 | ±10 | µA max | $V_{IN}$ = 0 V |
| | 20 | 10 | µA min | $V_{IN}$ = 5 V, 3 V internal pull-down |
| | 105 | 35 | µA max | $V_{IN}$ = 5 V, 3 V internal pull-down |
| P1.0, P1.1, Port 3[15] | ±10 | ±10 | µA max | $V_{IN}$ = 5 V, 3 V |
| (includes MISO, MOSI/SDATA and $\overline{SS}$) | 1 | 1 | µA typ | |
| | –180 | –70 | µA min | $V_{IN}$ = 2 V, $V_{DD}$ = 5 V, 3 V |
| | –660 | –200 | µA max | |
| | –360 | –100 | µA typ | |
| | –20 | –5 | µA min | $V_{IN}$ = 450 mV, $V_{DD}$ = 5 V, 3 V |
| | –75 | –25 | µA max | |
| | –38 | –12 | µA typ | |
| INPUT CAPACITANCE | 5 | 5 | pF typ | All digital inputs |
| CRYSTAL OSCILLATOR (XTAL1 AND XTAL2) | | | | |
| Logic Inputs, XTAL1 Only | | | | |
| $V_{INL}$, Input Low Voltage | 0.8 | 0.4 | V typ | |
| $V_{INH}$, Input High Voltage | 3.5 | 2.5 | V typ | |
| XTAL1 Input Capacitance | 18 | 18 | pF typ | |
| XTAL2 Output Capacitance | 18 | 18 | pF typ | |
| DIGITAL OUTPUTS | | | | |
| Output High Voltage ($V_{OH}$) | 2.4 | 2.4 | V min | $I_{SOURCE}$ = 80 mA |
| Output Low Voltage ($V_{OL}$) | | | | |
| Port 1.0 and Port 1.1 | 0.4 | 0.4 | V max | $I_{SINK}$ = 10 mA, $T_{MAX}$ = 85°C |
| Port 1.0 and Port 1.1 | 0.4 | 0.4 | V max | $I_{SINK}$ = 10 mA, $T_{MAX}$ = 125°C |
| SCLOCK, MISO/MOSI | 0.4 | 0.4 | V max | $I_{SINK}$ = 4 mA |
| All Other Outputs | 0.4 | 0.4 | V max | $I_{SINK}$ = 1.6 mA |
| MCU CORE CLOCK | | | | |
| MCU Clock Rate | 131.1 | 131.1 | kHz min | Clock rate generated via on-chip PLL, programmable via CD2-0 in PLLCON |
| | 16.78 | 16.78 | MHz max | |
| START UP TIME | | | | |
| At Power-On | 500 | 500 | ms typ | |
| From Idle Mode | 100 | 100 | µs typ | |
| From Power-Down Mode | | | | |
| Oscillator Running | | | | OSC_PD = 0 in PLLCON SFR |
| Wake-Up with $\overline{INT0}$ Interrupt | 100 | 100 | µs typ | |
| Wake-Up with SPI/$I^2C$ Interrupt | 100 | 100 | µs typ | |
| Wake-Up with TIC Interrupt | 100 | 100 | µs typ | |
| Wake-Up with External RESET | 3 | 3 | ms typ | |

| Parameter | $V_{DD}$ = 5 V | $V_{DD}$ = 3 V | Unit | Test Conditions |
|---|---|---|---|---|
| Oscillator Powered Down[16] | | | | OSC_PD = 1 in PLLCON SFR |
| Wake-Up with $\overline{INT0}$ Interrupt | 150 | 400 | ms typ | |
| Wake-Up with SPI/I²C Interrupt | 150 | 400 | ms typ | |
| Wake-Up with External RESET | 150 | 400 | ms typ | |
| After External RESET in Normal Mode | 3 | 3 | ms typ | |
| After WDT Reset in Normal Mode | 3 | 3 | ms typ | Controlled via WDCON SFR |
| FLASH/EE MEMORY RELIABILITY CHARACTERISTICS[17] | | | | |
| Endurance[18] | 100,000 | 100,000 | Cycles min | |
| Data Retention[19] | 100 | 100 | Years min | |
| POWER REQUIREMENTS[20, 21] | | | | |
| Power Supply Voltages | | | | |
| $AV_{DD}/DV_{DD}$ – AGND | | 2.7 | V min | $AV_{DD}/DV_{DD}$ = 3 V nom |
| | | 3.3 | V max | |
| | 4.5 | | V min | $AV_{DD}/DV_{DD}$ = 5 V nom |
| | 5.5 | | V max | |
| Power Supply Currents, Normal Mode | | | | |
| $D_{VDD}$ Current[14] | 5 | 2.5 | mA max | Core CLK = 2.097 MHz |
| | 4 | 2 | mA typ | (CD bits in PLLCON = 3) |
| $A_{VDD}$ Current[14] | 1.7 | 1.7 | mA max | |
| $D_{VDD}$ Current | 20 | 10 | mA max | Core CLK = 16.78MHz (max) |
| | 16 | 8 | mA typ | (CD bits in PLLCON = 0) |
| $A_{VDD}$ Current | 1.7 | 1.7 | mA max | |
| $D_{VDD}$ Current[14] | 3.5 | 1.5 | mA max | Core CLK = 131.2 kHz (min) |
| | 2.8 | 1.2 | mA typ | (CD bits in PLLCON = 7) |
| $A_{VDD}$ Current | 1.7 | 1.7 | mA max | |
| Power Supply Currents, Idle Mode | | | | |
| $D_{VDD}$ Current[14] | 1.7 | 1.2 | mA max | Core CLK = 2.097 MHz |
| | 1.5 | 1 | mA typ | (CD Bits in PLLCON = 3) |
| $AV_{DD}$ Current[14] | 0.15 | 0.15 | mA max | |
| $DV_{DD}$ Current[14] | 6 | 3 | mA max | Core CLK = 16.78 MHz (max) |
| | 4 | 2.5 | mA typ | (CD bits in PLLCON = 0) |
| $AV_{DD}$ Current[14] | 0.15 | 0.15 | mA max | |
| $DV_{DD}$ Current[14] | 1.25 | 1 | mA max | Core CLK = 131 kHz (min) |
| | 1.1 | 0.7 | mA typ | (CD bits in PLLCON = 7) |
| $AV_{DD}$ Current[14] | 0.15 | 0.15 | mA max | |
| Power Supply Currents, Power-Down Mode | | | | Core CLK = 2.097 MHz or 16.78 MHz (CD bits in PLLCON = 3 or 0) |
| $DV_{DD}$ Current[14] | | 20 | µA max | Oscillator on |
| | 40 | 14 | µA typ | |
| $AV_{DD}$ Current | 1 | 1 | µA typ | |
| $DV_{DD}$ Current | | 15 | µA max | Oscillator off |
| | 20 | 10 | µA typ | |
| $AV_{DD}$ Current | 1 | 1 | µA typ | |
| Typical Additional Power Supply Currents | | | | Core CLK = 2.097 MHz, (CD bits in PLLCON = 3) $AV_{DD}$ = $DV_{DD}$ = 5 V |
| PSM Peripheral | 50 | | µA typ | |
| ADC | 1.5 | | mA typ | |
| DAC | 150 | | µA typ | |

[1]Temperature range –40ºC to +125ºC.
[2]ADC linearity is guaranteed when operating in nonpipelined mode, i.e., ADC conversion followed sequentially by a read of the ADC result. ADC linearity is also guaranteed during normal MicroConverter core operation.
[3]ADC LSB size = $V_{REF}$ /$2^{12}$, i.e., for internal $V_{REF}$ = 2.5 V, 1 LSB = 610 µV, and for external $V_{REF}$ = 1 V, 1 LSB = 244 µV.
[4]Offset and gain error and offset and gain error match are measured after factory calibration.
[5]Based on external ADC system components the user may need to execute a system calibration to remove additional external channel errors and achieve these specifications.
[6]Measured with coherent sampling system using external 16.77 MHz clock via P3.5 (Pin 22).
[7]SNR calculation includes distortion and noise components.
[8]Channel-to-channel crosstalk is measured on adjacent channels.
[9]The temperature monitor gives a measure of the die temperature directly; air temperature can be inferred from this result.
[10]DAC linearity is calculated using a reduced code range of 48 to 4095, 0 V to $V_{REF}$ range; a reduced code range of 48 to 3950, 0 V to $V_{DD}$ range. DAC output load = 10 kΩ and 100 pF.
[11]DAC differential nonlinearity specified on 0 V to $V_{REF}$ and 0 to $V_{DD}$ ranges.
[12]Measured with $V_{REF}$ and $C_{REF}$ pins decoupled with 0.1 µF capacitors to ground. Power-up time for the internal reference is determined by the value of the decoupling capacitor chosen for both the $V_{REF}$ and $C_{REF}$ pins.
[13]When using an external reference device, the internal band gap reference input can be bypassed by setting the ADCCON1.6 bit. In this mode, the $V_{REF}$ and $C_{REF}$ pins need to be shorted together for correct operation.
[14]These numbers are not production tested but are guaranteed by design and/or characterization data on production release.
[15]Pins configured in $I^2C$ compatible mode or SPI mode; pins configured as digital inputs during this test.
[16]These typical specifications assume no loading on the XTAL2 pin. Any additional loading on the XTAL2 pin increases the power-on times.
[17]Flash/EE memory reliability characteristics apply to both the Flash/EE program memory and the Flash/EE data memory.
[18]Endurance is qualified to 100 kcycles as per JEDEC Std. 22, Method A117 and measured at –40ºC, +25°C, and +125°C; typical endurance at +25°C is 700 kcycles.
[19]Retention lifetime equivalent at junction temperature ($T_J$) = 55°C as per JEDEC Std. 22, Method A117. Retention lifetime based on an activation energy of 0.6 eV derates with junction temperature as shown in Figure 33 in the Flash/EE memory description section.
[20]Power supply current consumption is measured in normal, idle, and power-down modes under the following conditions:

Normal Mode: Reset and all digital I/O pins = open circuit, core Clk changed via CD bits in PLLCON, core executing internal software loop.
Idle Mode: Reset and all digital I/O pins = open circuit, core Clk changed via CD bits in PLLCON, PCON.0 = 1, core execution suspended in idle mode.
Power-Down Mode: Reset and all P1.2–P1.7 pins = 0.4 V; all other digital I/O pins are open circuit, Core Clk changed via CD bits in PLLCON, PCON.1 = 1, Core execution suspended in power-down mode, OSC turned on or off via OSC_PD bit (PLLCON.7) in PLLCON SFR.

[21]$DV_{DD}$ power supply current increases typically by 3 mA (3 V operation) and 10 mA (5 V operation) during a Flash/EE memory program or erase cycle.

# ABSOLUTE MAXIMUM RATINGS

**Table 2. Temperature = 25°C, unless otherwise noted**

| Parameter | Rating |
|---|---|
| $AV_{DD}$ to AGND | −0.3 V to +7 V |
| $DV_{DD}$ to AGND | −0.3 V to +7 V |
| $AV_{DD}$ to $DV_{DD}$ | −0.3 V to +0.3 V |
| AGND to DGND[1] | −0.3 V to +0.3 V |
| Analog Input Voltage to AGND[2] | −0.3 V to $AV_{DD}$ + 0.3 V |
| Reference Input Voltage to AGND | −0.3 V to $AV_{DD}$ + 0.3 V |
| Analog Input Current (Indefinite) | 30 mA |
| Reference Input Current (Indefinite) | 30 mA |
| Digital Input Voltage to DGND | −0.3 V to $DV_{DD}$ + 0.3 V |
| Digital Output Voltage to DGND | −0.3 V to $DV_{DD}$ + 0.3 V |
| Operating Temperature Range | −40°C to +125°C |
| Storage Temperature Range | −65°C to +150°C |
| Junction Temperature | 150°C |
| $\theta_{JA}$ Thermal Impedance | 97.9°C/W |
| Lead Temperature, Soldering | |
| Vapor Phase (60 sec) | 215°C |
| Infrared (15 sec) | 220°C |

[1] AGND and DGND are shorted internally on the ADuC814.
[2] Applies to Pins P1.2 to P1.7 operating in analog or digital input mode.

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

## ESD CAUTION

ESD (electrostatic discharge) sensitive device. Electrostatic charges as high as 4000 V readily accumulate on the human body and test equipment and can discharge without detection. Although this product features proprietary ESD protection circuitry, permanent damage may occur on devices subjected to high energy electrostatic discharges. Therefore, proper ESD precautions are recommended to avoid performance degradation or loss of functionality.

# PIN CONFIGURATION AND FUNCTION DESCRIPTION

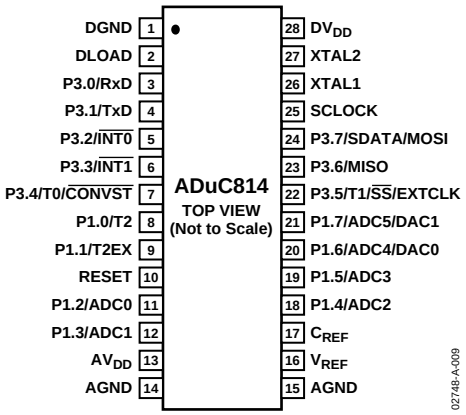



*Figure 2. Pin Configuration*

**Table 3. Pin Descriptions**

| Pin No. | Mnemonic | Type | Function |
|---|---|---|---|
| 1 | DGND | S | Digital Ground. Ground reference point for the digital circuitry. |
| 2 | DLOAD | I | Debug/Serial Download Mode. Enables when pulled high through a resistor on power-on or RESET. In this mode, DLOAD may also be used as an external emulation I/O pin, therefore the voltage level at this pin must not be changed during this mode of operation because it may cause an emulation interrupt that halts code execution. User code is executed when this pin is pulled low on power-on or RESET. |
| 3–7 | P3.0 – P3.4 | I/O | Bidirectional Port Pins with Internal Pull-Up Resistors. Port 3 pins that have 1s written to them are pulled high by the internal pull-up resistors, and in that state they can be used as inputs. As inputs, with Port 3 pins being pulled low externally, they source current because of the internal pull-up resistors. When driving a 0-to-1 output transition, a strong pull-up is active during S1 of the instruction cycle. Port 3 pins also have various secondary functions which are described next. |
| 3 | P3.0/RxD | I/O | Receiver Data Input (asynchronous) or Data Input/Output (synchronous) in Serial (UART) Mode. |
| 4 | P3.1/TxD | I/O | Transmitter Data Output (asynchronous) or Clock Output (synchronous) in Serial (UART) Mode. |
| 5 | P3.2/$\overline{\text{INT0}}$ | I/O | Interrupt 0, programmable edge or level-triggered interrupt input, which can be programmed to one of two priority levels. This pin can also be used as agate control input to Timer 0. |
| 6 | P3.3/$\overline{\text{INT1}}$ | I/O | Interrupt 1, programmable edge or level-triggered interrupt input, which can be programmed to one of two priority levels. This pin can also be used as agate control input to Timer 1. |
| 7 | P3.4/T0/$\overline{\text{CONVST}}$ | I/O | Timer/Counter 0 Input and External Trigger Input for ADC Conversion Start. |
| 8–9 | P1.0–P1.1 | I/O | Bidirectional Port Pins with Internal Pull-Up Resistors. Port 1 pins that have 1s written to them are pulled high by the internal pull-up resistors, and in that state they can be used as inputs. As inputs ,with Port 1 pins being pulled low externally, they source current because of the internal pull-up resistors When driving a 0-to-1 output transition a strong pull-up is active during S1 of the instruction cycle. Port 1 pins also have various secondary functions which are described as follows. |
| 8 | P1.0/T2 | I/O | Timer 2 Digital Input. Input to Timer/Counter 2. When enabled, Counter 2 is incremented in response to a 1 to 0 transition of the T2 input. |
| 9 | P1.1/T2EX | I/O | Digital Input. Capture/Reload trigger for Counter 2. |
| 10 | RESET | I | Reset Input. A high level on this pin while the oscillator is running resets the device. There is an internal weak pull-down and a Schmitt-trigger input stage on this pin. |
| 11–12 | P1.2–P1.3 | I | Port 1.2 to P1.3. These pins have no digital output drivers, i.e., they can only function as digital inputs, for which 0 must be written to the port bit. These port pins also have the following analog functionality: |
| 11 | P1.2/ADC0 | I | ADC Input Channel 0. Selected via ADCCON2 SFR. |
| 12 | P1.3/ADC1 | I | ADC Input Channel 1. Selected via ADCCON2 SFR. |
| 13 | $AV_{DD}$ | S | Analog Positive Supply Voltage, 3 V or 5 V. |
| 14–15 | AGND | G | Analog Ground. Ground reference point for the analog circuitry. |
| 16 | $V_{REF}$ | I/O | Reference Input/Output. This pin is connected to the internal reference through a switch and is the reference source for the analog to digital converter. The nominal internal reference voltage is 2.5 V and this appears at the pin. This pin can be used to connect an external reference to the analog to digital converter by setting ADCCON1.6 to 1. Connect 0.1 µF between this pin and AGND. |

| Pin No. | Mnemonic | Type | Function |
|---|---|---|---|
| 17 | $C_{REF}$ | I | Decoupling Input for On-Chip Reference. Connect 0.1 µF between this pin and AGND. |
| 18–21 | P1.4–P1.7 | I | Port 1.4 to P1.7. These pins have no digital output drivers, i.e., they can only function as digital inputs, for which 0 must be written to the port bit. These port pins also have the following analog functionality: |
| 18 | P1.4/ADC2 | I | ADC Input Channel 2. Selected via ADCCON2 SFR. |
| 19 | P1.5/ADC3 | I | ADC Input Channel 2. Selected via ADCCON2 SFR. |
| 20 | P1.6/ADC4/ DAC0 | I/O | ADC Input Channel 4. Selected via ADCCON2 SFR. The voltage DAC Channel 0 can also be configured to appear on P1.6. |
| 21 | P1.7/ ADC5/DAC1 | I/O | ADC Input Channel 5, selected via ADCCON2 SFR. The voltage DAC Channel 1 can also be configured to appear on P1.7. |
| 22–24 | P3.5–P3.7 | I/O | Bidirectional Port Pins with Internal Pull-Up Resistors. Port 3 pins that have 1s written to them are pulled high by the internal pull-up resistors, and in that state they can be used as inputs. As inputs ,with Port 3 pins being pulled low externally, they source current because of the internal pull-up resistors. When driving a 0-to-1 output transition a strong pull-up is active during S1 of the instruction cycle. Port 3 pins also have various secondary functions which are described as follows. |
| 22 | P3.5/T1 | | I/O Timer/Counter 1 Input. P3.5–P3.7 pins also have SPI interface functions. To enable these functions, Bit 0 of the CFG814 SFR must be set to 1. |
| 22 | P3.5/$\overline{SS}$ /EXTCLK | I/O | This pin also functions as the Slave Select input for the SPI interface when the device is operated in slave mode. P3.5 can also function as an input for an external clock. This clock effectively bypasses the PLL. This function is enabled by setting Bit 1 of the CFG814 SFR. |
| 23 | P3.6/MISO | I/O | SPI Master Input/Slave Output Data Input/Output Pin. |
| 24 | P3.7/SDATA/ MOSI | I/O | SPI Master Output/Slave Input Data Input/Output Pin. |
| 25 | SCLOCK | I/O | Serial Clock Pin for SPI Serial Interface Clock. |
| 26 | XTAL1 | I | Input to the Crystal Oscillator Inverter. |
| 27 | XTAL2 | O | Output from the Crystal Oscillator Inverter. |
| 28 | $DV_{DD}$ | S | Analog Positive Supply Voltage, 3 V or 5 V. |

I = Input, O = Output, S = Supply, G - Ground.

The following notes apply to the entire data sheet:

- In bit designation tables, *set* implies a Logic 1 state, and *cleared* implies a Logic 0 state, unless otherwise stated.
- *Set* and *cleared* also imply that the bit is set or cleared by the ADuC814 hardware, unless otherwise stated.
- User software should not write to reserved or unimplemented bits as they may be used in future products.

# TERMINOLOGY

## ADC SPECIFICATIONS

### Integral Nonlinearity

This is 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 point1/2 LSB below the first code transition and full scale, a point 1/2 LSB above the last code transition.

### Differential Nonlinearity

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

### Offset Error

This is the deviation of the first code transition (0000 … 000) to (0000 … 001) from the ideal, i.e., +1/2 LSB.

### Full-Scale Error

This is the deviation of the last code transition from the ideal AIN voltage (full-scale error has been adjusted out).

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

This is 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

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

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

### Total Harmonic Distortion (THD)

Total harmonic distortion is the ratio of the rms sum of the harmonics to the fundamental.

### Peak Harmonic or Spurious Noise

Peak harmonic or spurious noise is defined as the ratio of the rms value of the next largest component in the ADC output spectrum (up to $f_S/2$ and including dc) to the rms value of the fundamental. Normally, the value of this specification is determined by the largest harmonic in the spectrum, but for ADCs where the harmonics are buried in the noise floor, it is the noise peak.

## DAC SPECIFICATIONS

### Relative Accuracy

Relative accuracy or endpoint linearity 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-scale error and full-scale error.

### Voltage Output Settling Time

This is the amount of time it takes for the output to settle to a specified level for a full-scale input change.

### Digital-to-Analog Glitch Impulse

This is the amount of charge injected into the analog output when the inputs change state. It is specified as the area of the glitch in nV-sec.

# TYPICAL PERFORMANCE CURVES

The typical performance plots presented in this section illustrate typical performance of the ADuC814 under various operating conditions. Note that all typical plots in this section were generated using the ADuC814BRU, i.e., the B-grade part.

Figure 3 and Figure 4 show typical ADC integral nonlinearity (INL) errors from ADC Code 0 to Code 4095 at 5 V and 3 V supplies, respectively. The ADC is using its internal reference (2.5 V) and operating at a sampling rate of 152 kHz. The typical worst-case errors in both plots are just less than 0.3 LSBs.

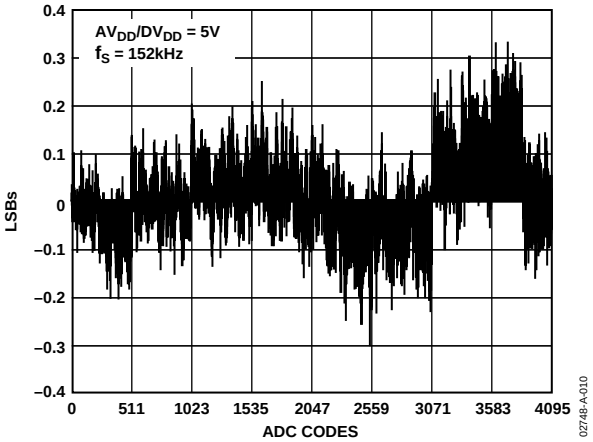



*Figure 3. Typical INL Error, $V_{DD}$ = 5 V*

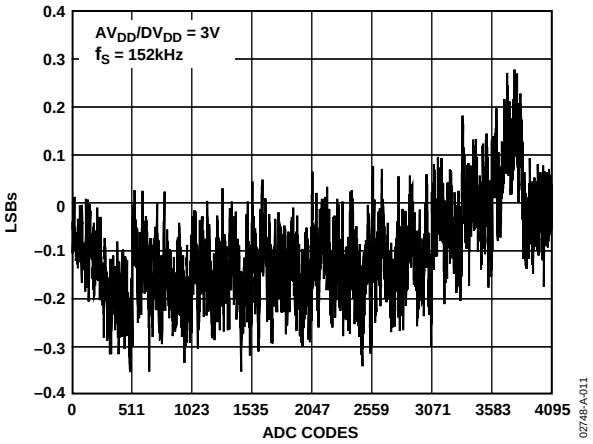



*Figure 4. Typical INL Error, $V_{DD}$ = 3 V*

Figure 5 and Figure 6 show the variation in worst-case positive (WCP) INL and worst-case negative (WCN) INL versus external reference input voltage.

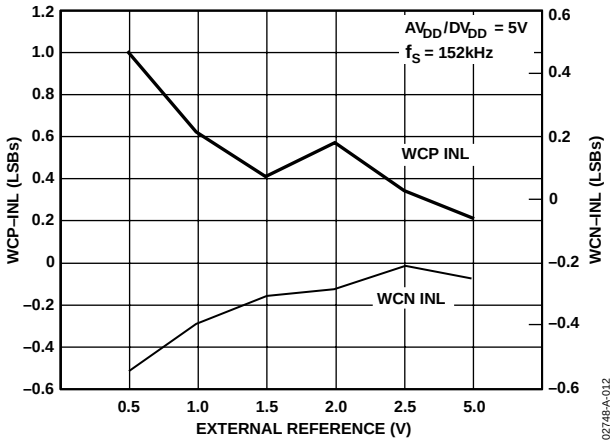



*Figure 5. Typical Worst-Case INL Error vs. $V_{REF}$, $V_{DD}$ = 5 V*

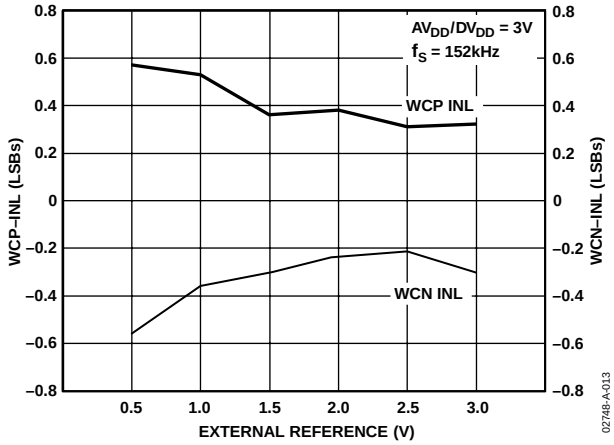



*Figure 6. Typical Worst-Case INL Error vs. $V_{REF}$, $V_{DD}$ = 3 V*

Figure 7 and Figure 8 show typical ADC differential nonlinearity (DNL) errors from ADC Code 0 to Code 4095 at 5 V and 3 V supplies, respectively. The ADC is using its internal reference (2.5 V) and operating at a sampling rate of 152 kHz. The typical worst-case errors in both plots are just less than 0.2 LSBs.

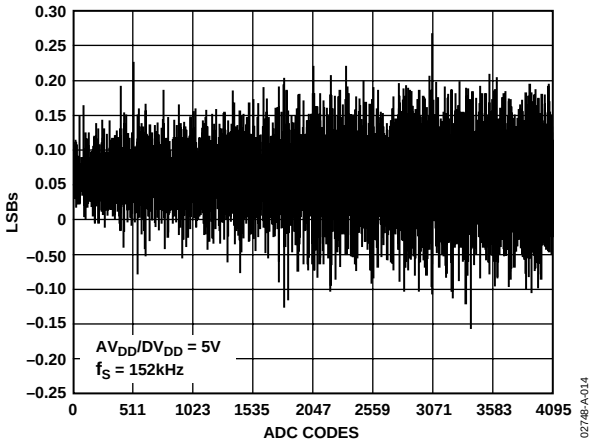



*Figure 7. Typical DNL Error, $V_{DD}$ = 5 V*

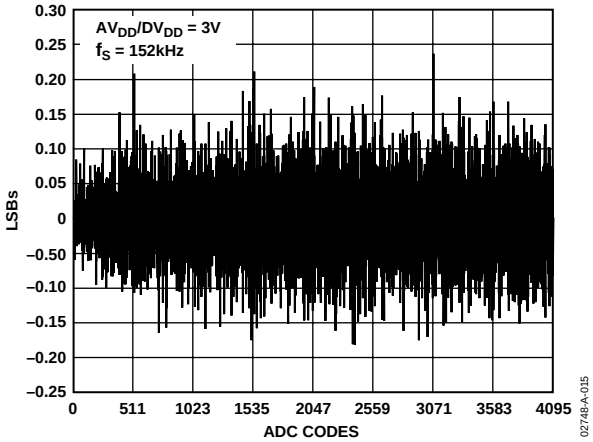



*Figure 8. Typical DNL Error, $V_{DD}$ = 3 V*

Figure 9 and Figure 10 show the variation in worst-case positive (WCP) DNL and worst-case negative (WCN) DNL versus external reference input voltage.

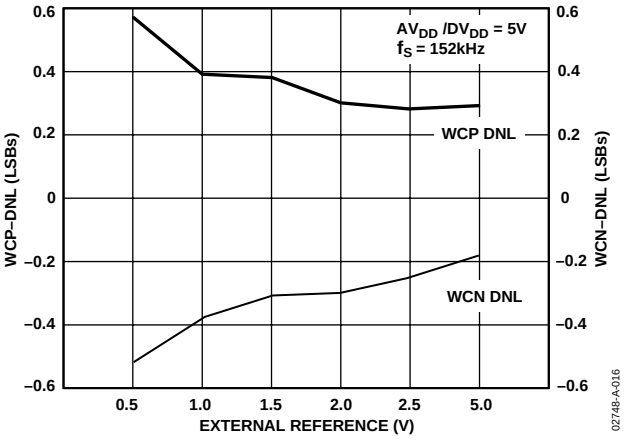



*Figure 9. Typical Worst-Case DNL Error vs. $V_{REF}$, $V_{DD}$ = 5 V*

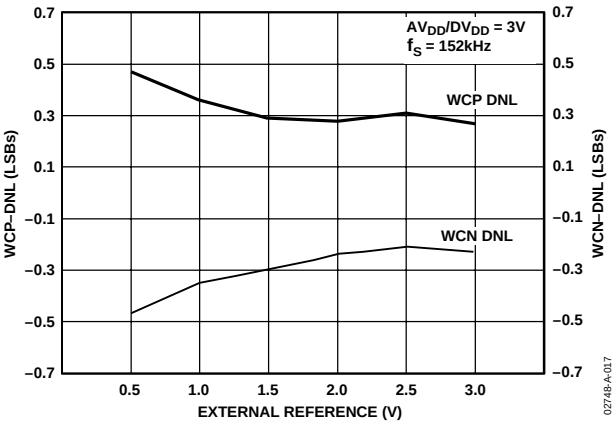



*Figure 10. Typical Worst-Case DNL Error vs. $V_{REF}$, $V_{DD}$ = 3 V*

Figure 11 shows a histogram plot of 10,000 ADC conversion results on a dc input with $V_{DD}$ = 5 V. The plot illustrates an excellent code distribution pointing to the low noise performance of the on-chip precision ADC.

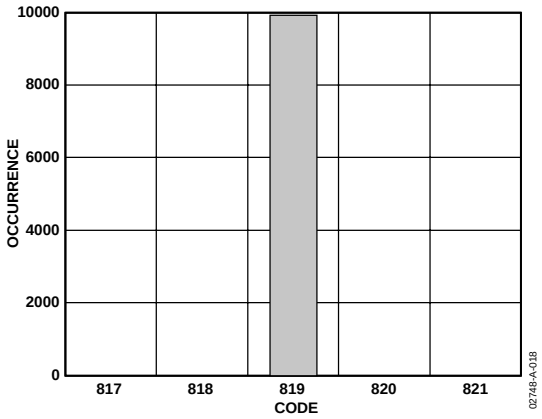



*Figure 11. Code Histogram plot, $V_{DD}$ = 5 V*

Figure 12 shows a histogram plot of 10,000 ADC conversion results on a dc input for $V_{DD}$ = 3 V. The plot again illustrates a very tight code distribution of 1 LSB with the majority of codes appearing in one output bin.

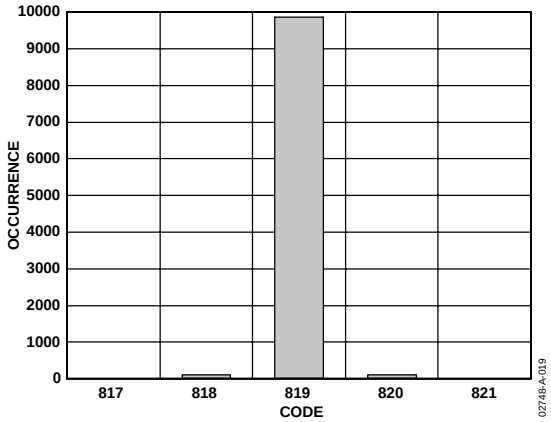



*Figure 12. Code Histogram Plot, $V_{DD}$ = 3 V*

Figure 13 and Figure 14 show typical FFT plots for the ADuC814. These plots were generated using an external clock input via P3.5 to achieve coherent sampling. The ADC is using its internal reference (2.5 V) sampling a full-scale, 10 kHz sine wave test tone input at a sampling rate of 149.79 kHz. The resultant FFTs shown at 5 V and 3 V supplies illustrate an excellent 100 dB noise floor, a 71 dB signal-to-noise ratio (SNR), and a THD greater than −80 dB.

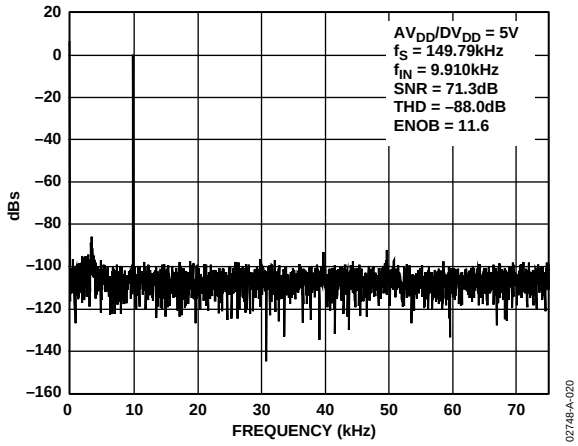



*Figure 13. ADuC814 Dynamic Performance at $V_{DD}$ = 5 V*

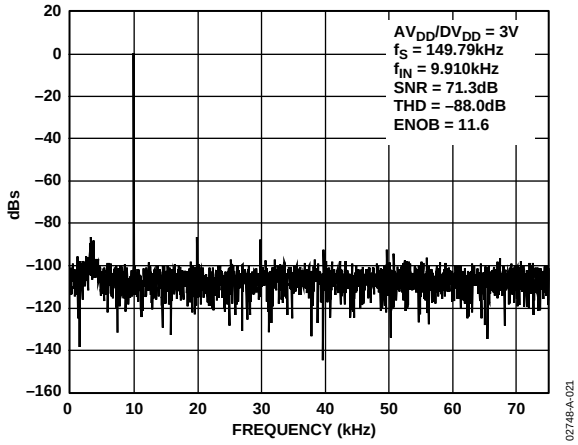



*Figure 14. ADuC814 Dynamic Performance at $V_{DD}$ = 3 V*

Figure 15 and Figure 16 show typical dynamic performance versus external reference voltages. Again excellent ac performance can be observed in both plots with some roll-off being observed as $V_{REF}$ falls below 1 V.

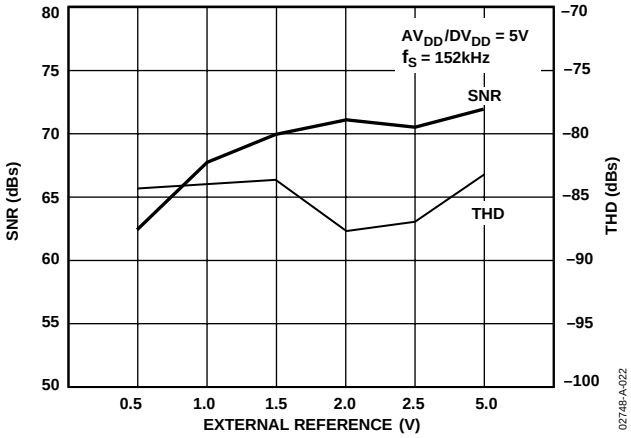



*Figure 15. Typical Dynamic Performance vs. $V_{REF}$, $V_{DD}$ = 5 V*

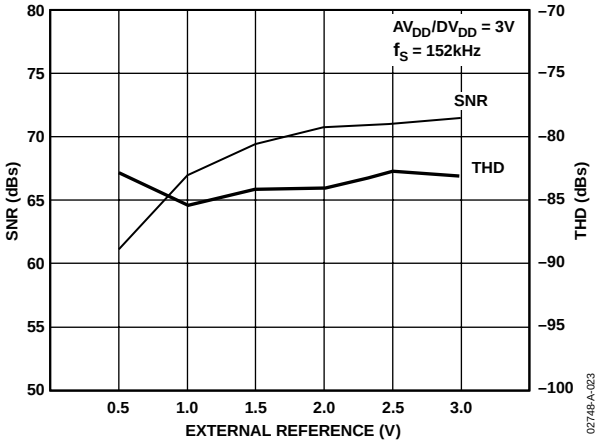



*Figure 16. Typical Dynamic Performance vs. $V_{REF}$, $V_{DD}$ = 3 V*

# ADuC814 ARCHITECTURE, MAIN FEATURES

The ADuC814 is a fully integrated 247 kSPS 12-bit data acquisition system incorporating a high performance multi-channel ADC, an 8-bit MCU, and program/data Flash/EE memory on a single chip.

This low power device operates from a 32 kHz crystal with an on-chip PLL generating a high frequency clock of 16.78 MHz. This clock is, in turn, routed through a programmable clock divider from which the MCU core clock operating frequency is generated.

The microcontroller core is an 8052, and therefore 8051, instruction set compatible. The microcontroller core machine cycle consists of 12 core clock periods of the selected core operating frequency. Eight kbytes of nonvolatile Flash/EE program memory are provided on-chip. 640 bytes of nonvolatile Flash/EE data memory and 256 bytes RAM are also integrated on-chip.

The ADuC814 also incorporates additional analog functionality with dual 12-bit DACs, a power supply monitor, and a band gap reference. On-chip digital peripherals include a watchdog timer, time interval counter, three timer/counters, and three serial I/O ports (SPI, UART, $I^2C$).

On-chip factory firmware supports in-circuit serial download and debug modes (via UART), as well as single-pin emulation mode via the DLOAD pin. A detailed functional block diagram of the ADuC814 is shown in Figure 17.

The ADuC814 is supported by a QuickStart Development System. This is a full-featured, low cost system, consisting of PC-based (Windows compatible) hardware and software development tools.

The part operates from a single 3 V or 5 V supply. When operating from 3 V supplies, the power dissipation for the part is below 10 mW. The ADuC814 is housed in a 28-lead TSSOP package and is specified for operation over an extended temperature range −40°C to +125°C.

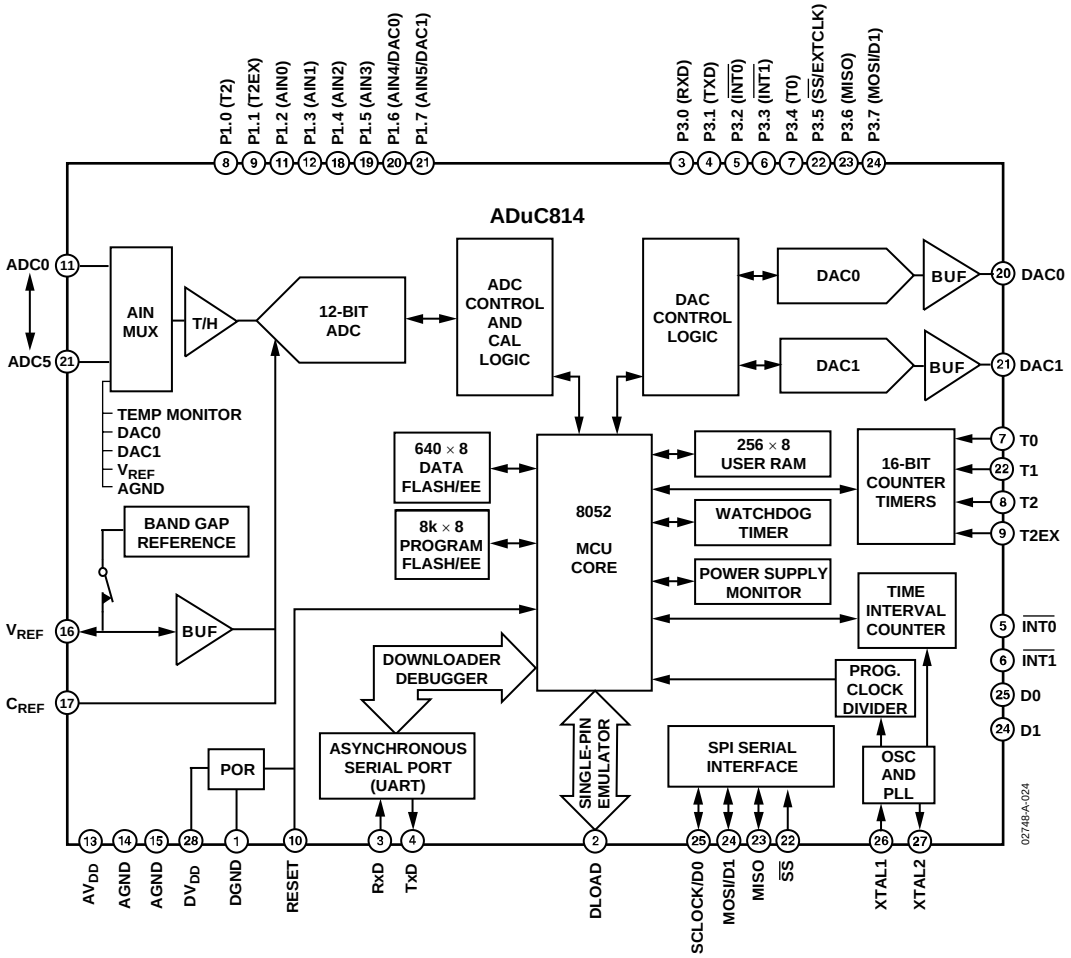



*Figure 17. ADuC814 Block Diagram*

## MEMORY ORGANIZATION

The ADuC814 does not have Port 0 and Port 2 pins and therefore does not support external program or data memory interfaces. The device executes code from the internal 8-kByte Flash/EE program memory. This internal code space can be programmed via the UART serial port interface while the device is in-circuit. The program memory space of the ADuC814 is shown in Figure 18.

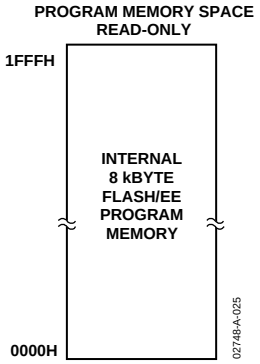



*Figure 18. Program Memory Map*

The data memory address space consists of internal memory only. The internal memory space is divided into four physically separate and distinct blocks, namely the lower 128 bytes of RAM, the upper 128 bytes of RAM, the 128 bytes of special function register (SFR) area, and a 640-byte Flash/EE data memory. While the upper 128 bytes of RAM and the SFR area share the same address locations, they are accessed through different addressing modes.

The lower 128 bytes of data memory can be accessed through direct or indirect addressing, the upper 128 bytes of RAM can be accessed through indirect addressing, and the SFR area is accessed through direct addressing.

Also, as shown in Figure 19, an additional 640 bytes of Flash/EE data memory are available to the user and can be accessed indirectly via a group of control registers mapped into the SFR area. Access to the Flash/EE data memory is discussed in detail later as part of the Flash/EE Memory section.

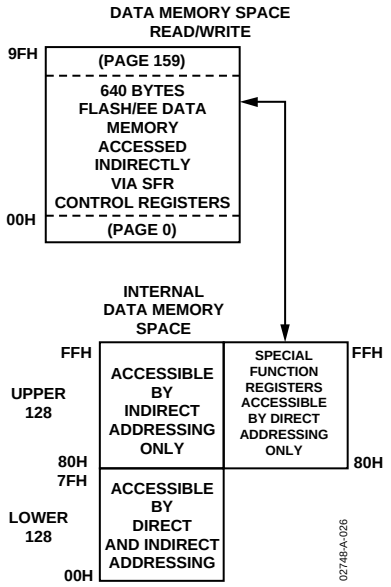



*Figure 19. Data Memory Map*

The lower 128 bytes of internal data memory are mapped as shown in Figure 20. The lowest 32 bytes are grouped into four banks of eight registers addressed as R0 to R7. The next 16 bytes (128 bits), locations 20H to 2FH above the register banks, form a block of directly addressable bit locations at bit addresses 00H through 7FH. The stack can be located anywhere in the internal memory address space, and the stack depth can be expanded up to 256 bytes.

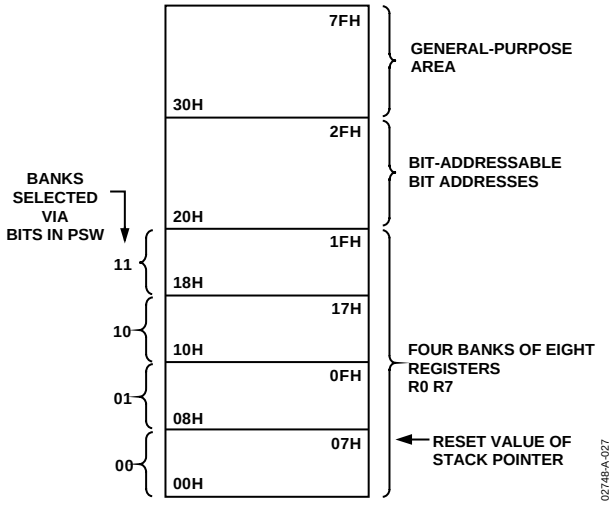



*Figure 20. Lower 128 Bytes of Internal Data Memory*

RESET initializes the stack pointer to location 07H and increments it once to start from location 08H, which is also the first register (R0) of Register Bank 1. If more than one register bank is being used, the stack pointer should be initialized to an area of RAM not used for data storage.

The SFR space is mapped to the upper 128 bytes of internal data memory space and is accessed by direct addressing only. It provides an interface between the CPU and all on-chip peripherals. A block diagram showing the programming model of the ADuC814 via the SFR area is shown in Figure 21. A complete SFR map is shown in Figure 22.

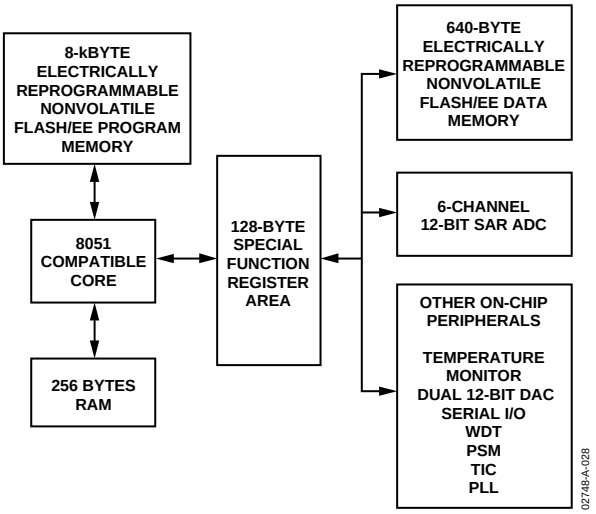



*Figure 21. Programming Model*

## OVERVIEW OF MCU-RELATED SFRS

### *Accumulator SFR*

ACC is the accumulator register and is used for math operations including addition, subtraction, integer multiplication and division, and Boolean bit manipulations. The mnemonics for accumulator-specific instructions refer to the accumulator as A.

### *B SFR*

The B register is used with the ACC for multiplication and division operations. For other instructions it can be treated as a general-purpose scratchpad register.

### *Stack Pointer SFR*

The SP register is the stack pointer and is used to hold an internal RAM address called the *top of the stack*. The SP register is incremented before data is stored during PUSH and CALL executions. While the stack may reside anywhere in on-chip RAM, the SP register is initialized to 07H after a reset. This causes the stack to begin at location 08H.

### *Data Pointer*

The data pointer is made up of two 8-bit registers, named DPH (high byte) and DPL (low byte). These registers provide memory addresses for internal code access. The pointer may be manipulated as a 16-bit register (DPTR = DPH, DPL), or as two independent 8-bit registers (DPH, DPL).

### *Program Status Word SFR*

The program status word (PSW) register is the program status word that contains several bits reflecting the current status of the CPU as detailed in Table 4.

| | |
|---|---|
| SFR Address | D0H |
| Power-On Default | 00H |
| Bit Addressable | Yes |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| CY | AC | F0 | RS1 | RS0 | OV | F1 | P |

**Table 4. PSW SFR Bit Designations**

| Bit No. | Name | Description |
|---|---|---|
| 7 | CY | Carry Flag. |
| 6 | AC | Auxiliary Carry Flag. |
| 5 | F0 | General-Purpose Flag. |
| 4 | RS1 | Register Bank Select Bits. |
| 3 | RS0 | RS1 RS0 Selected Bank<br>0 0 0<br>0 1 1<br>1 0 2<br>1 1 3 |
| 2 | OV | Overflow Flag. |
| 1 | F1 | General-Purpose Flag. |
| 0 | P | Parity Bit. |

### *Power Control SFR*

The power control (PCON) register contains bits for power-saving options and general-purpose status flags as shown in Table 5.

| | |
|---|---|
| SFR Address | 87H |
| Power-On Default | 00H |
| Bit Addressable | No |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| SMOD | SERIPD | INT0PD | --- | GF1 | GF0 | PD | IDL |

**Table 5. PCON SFR Bit Designations**

| Bit No. | Name | Description |
|---|---|---|
| 7 | SMOD | Double UART Baud Rate. |
| 6 | SERIPD | SPI Power-Down Interrupt Enable. |
| 5 | INT0PD | $\overline{\text{INT0}}$ Power-Down Interrupt Enable. |
| 4 | RSVD | Reserved. |
| 3 | GF1 | General-Purpose Flag Bit. |
| 2 | GF0 | General-Purpose Flag Bit. |
| 1 | PD | Power-Down Mode Enable. |
| 0 | IDL | Idle Mode Enable. |

## SPECIAL FUNCTION REGISTERS

All registers, except the program counter and the four general-purpose register banks, reside in the SFR area. The SFR registers include control, configuration, and data registers that provide an interface between the CPU and all on-chip peripherals.

Figure 22 shows a full SFR memory map and SFR contents on RESET; NOT USED indicates unoccupied SFR locations.

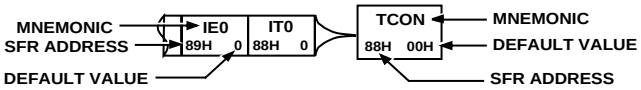
Unoccupied locations in the SFR address space are not implemented, i.e., no register exists at this location. If an unoccupied location is read, an unspecified value is returned. SFR locations reserved for future use are shaded (RESERVED) and should not be accessed by the user software.

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| ISPI FFH 0 | WCOL FEH 0 | SPE FDH 0 | SPIM FCH 0 | CPOL FBH 0 | CPHA FAH 1 | SPR1 F9H 0 | SPR0[5] F8H 0 | BITS |
| F7H 0 | F6H 0 | F5H 0 | F4H 0 | F3H 0 | F2H 0 | F1H 0 | F0H 0 | BITS |
| D1 EFH 0 | D1EN EEH 0 | D0 EDH 0 | ECH 0 | D0EN EBH 0 | EAH 0 | E9H 0 | E8H 0 | BITS |
| E7H 0 | E6H 0 | E5H 0 | E4H 0 | E3H 0 | E2H 0 | E1H 0 | E0H 0 | BITS |
| ADCI DFH 0 | ADCSPI DEH 0 | CCONV DDH 0 | SCONV DCH 0 | CS3 DBH 0 | CS2 DAH 0 | CS1 D9H 0 | CS0 D8H 0 | BITS |
| CY D7H 0 | AC D6H 0 | F0 D5H 0 | RS1 D4H 0 | RS0 D3H 0 | OV D2H 0 | FI D1H 0 | P D0H 0 | BITS |
| TF2 CFH 0 | EXF2 CEH 0 | RCLK CDH 0 | TCLK CCH 0 | EXEN2 CBH 0 | TR2 CAH 0 | CNT2 C9H 0 | CAP2 C8H 0 | BITS |
| PRE3 C7H 0 | PRE2 C6H 0 | PRE1 C5H 0 | PRE0 C4H 1 | WDIR C3H 0 | WDS C2H 0 | WD C1H 0 | WDWR C0H 0 | BITS |
| PSI BFH 0 | PADC BEH 0 | PT2 BDH 0 | PS BCH 0 | PT1 BBH 0 | PX1 BAH 0 | PT0 B9H 0 | PX0 B8H 0 | BITS |
| RD B7H 1 | WR B6H 1 | T1 B5H 1 | T0 B4H 1 | INT1 B3H 1 | INT0 B2H 1 | TxD B1H 1 | RxD B0H 1 | BITS |
| EA AFH 0 | EADC AEH 0 | ET2 ADH 0 | ES ACH 0 | ET1 ABH 0 | EX1 AAH 0 | ET0 A9H 0 | EX0 A8H 0 | BITS |
| | | | | | | | | |
| SM0 9FH 0 | SM1 9EH 0 | SM2 9DH 0 | REN 9CH 0 | TB8 9BH 0 | RB8 9AH 0 | TI 99H 0 | RI 98H 0 | BITS |
| 97H 1 | 96H 1 | 95H 1 | 94H 1 | 93H 1 | 92H 1 | T2EX 91H 1 | T2 90H 1 | BITS |
| TF1 8FH 0 | TR1 8EH 0 | TF0 8DH 0 | TR0 8CH 0 | IE1 8BH 0 | IT1 8AH 0 | IE0 89H 0 | IT0 88H 0 | BITS |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| SPICON[1] F8H 04H | DAC0L F9H 00H | DAC0H FAH 00H | DAC1L FBH 00H | DAC1H FCH 00H | DACCON FDH 04H | RESERVED | RESERVED |
| B[1] F0H 00H | ADCOFSL F1H 00H | ADCOFSH F2H 20H | ADCGAINL F3H 00H | ADCGAINH F4H 00H | ADCCON3 F5H 00H | RESERVED | SPIDAT F7H 00H |
| DCON[1] E8H 00H | RESERVED | RESERVED | RESERVED | RESERVED | RESERVED | RESERVED | ADCCON1 EFH 00H |
| ACC[1] E0H 00H | RESERVED | RESERVED | RESERVED | RESERVED | RESERVED | RESERVED | RESERVED |
| ADCCON2[1] D8H 00H | ADCDATAL D9H 00H | ADCDATAH DAH 00H | RESERVED | RESERVED | RESERVED | RESERVED | PSMCON DFH DEH |
| PSW[1] D0H 00H | RESERVED | RESERVED | RESERVED | RESERVED | RESERVED | RESERVED | PLLCON D7H 53H |
| T2CON[1] C8H 00H | RESERVED | RCAP2L CAH 00H | RCAP2H CBH 00H | TL2 CCH 00H | TH2 CDH 00H | RESERVED | RESERVED |
| WDCON[1] C0H 10H | RESERVED | CHIPID C2H 0XH | RESERVED | NOT USED | RESERVED | EDARL C6H 00H | RESERVED |
| IP[1] B8H 00H | ECON B9H 00H | ETIM1 BAH 00H | ETIM2 BBH 00H | EDATA1 BCH 00H | EDATA2 BDH 00H | EDATA3 BEH 00H | EDATA4 BFH 00H |
| P3[1] B0H FFH | RESERVED | RESERVED | RESERVED | RESERVED | RESERVED | RESERVED | NOT USED |
| IE[1] A8H 00H | IEIP2 A9H A0H | RESERVED | RESERVED | RESERVED | RESERVED | RESERVED | RESERVED |
| NOT USED | TIMECON A1H 00H | HTHSEC A2H 00H | SEC A3H 00H | MIN A4H 00H | HOUR A5H 00H | INTVAL A6H 00H | NOT USED |
| SCON[1] 98H 00H | SBUF 99H 00H | I2CDAT 9AH 00H | I2CADD 9BH 55H | CFG814 9CH 04H | NOT USED | NOT USED | NOT USED |
| P1[1,2] 90H FFH | NOT USED | NOT USED | NOT USED | NOT USED | NOT USED | NOT USED | NOT USED |
| TCON[1] 88H 00H | TMOD 89H 00H | TL0 8AH 00H | TL1 8BH 00H | TH0 8CH 00H | TH1 8DH 00H | RESERVED | RESERVED |
| NOT USED | SP 81H 07H | DPL 82H 00H | DPH 83H 00H | RESERVED | RESERVED | RESERVED | PCON 87H 00H |

SFR MAP KEY:

THESE BITS ARE CONTAINED IN THIS BYTE.




02748-0-029

*Figure 22. Special Function Register Locations and Reset Values*

Note the following about SFRs:

- SFRs whose address ends in 0H or 8H are bit addressable.
- Only P1.0 and P1.1 can operate as digital I/O pins. P1.2–P1.7 can be configured as analog inputs (ADC inputs) or as digital inputs.
- The CHIPID SFR contains the silicon revision ID byte and may change for future silicon revisions.
- These registers are reconfigured at power-on with factory calculated calibration coefficients that can be overwritten by user code. See the calibration options in ADCCON3 SFR.
- When the SPIM bit in the SPICON SFR is cleared, the SPR0 bit reflects the level on the $\overline{SS}$ pin (Pin 22).

# ADC CIRCUIT INFORMATION

## GENERAL OVERVIEW

The ADC block incorporates a 4.05 msec, 6-channel, 12-bit resolution, single-supply ADC. This block provides the user with a multichannel multiplexer, track-and-hold amplifier, on-chip reference, offset calibration features and ADC. All components in this block are easily configured via a 3-register SFR interface.

The ADC consists of a conventional successive-approximation converter based around a capacitor DAC. The converter accepts an analog input range of 0 V to $V_{REF}$. A precision, factory calibrated 2.5 V reference is provided on-chip. An external reference may also be used via the external $V_{REF}$ pin. This external reference can be in the range 1.0 V to $AV_{DD}$.

Single or continuous conversion modes can be initiated in software. In hardware, a convert signal can be applied to an external pin (CONVST), or alternatively Timer 2 can be configured to generate a repetitive trigger for ADC conversions.

The ADuC814 has a high speed ADC to SPI interface data capture logic implemented on-chip. Once configured, this logic transfers the ADC data to the SPI interface without the need for CPU intervention.

The ADC has six external input channels. Two of the ADC channels are multiplexed with the DAC outputs, ADC4 with DAC0, and ADC5 with DAC1. When the DAC outputs are in use, any ADC conversion on these channels represents the DAC output voltage. Due care must be taken to ensure that no external signal is trying to drive these ADC/DAC channels while the DAC outputs are enabled.

In addition to the six external channels of the ADC, five internal signals are also routed through the front end multiplexer. These signals include a temperature monitor, DAC0, DAC1, $V_{REF}$, and AGND. The temperature monitor is a voltage output from an on-chip band gap reference, which is proportional to absolute temperature. These internal channels can be selected similarly to the external channels via CS3–CS0 bits in the ADCCON2 SFR.

The ADuC814 is shipped with factory programmed offset and gain calibration coefficients that are automatically downloaded to the ADC on a power-on or RESET event, ensuring optimum ADC performance. The ADC core contains automatic endpoint self-calibration and system calibration options that allow the user to overwrite the factory programmed coefficients if desired and tailor the ADC transfer function to the system in which it is being used.

## ADC TRANSFER FUNCTION

The analog input range for the ADC is 0 V to $V_{REF}$. For this range, the designed code transitions occur midway between successive integer LSB values, i.e., 1/2 LSB, 3/2 LSBs, 5/2 LSBs . . . FS –3/2 LSBs. The output coding is straight binary with 1 LSB = FS/4096 or 2.5 V/4096 = 0.61 mV when $V_{REF}$ = 2.5 V. The ideal input/output transfer characteristic for the 0 V to $V_{REF}$ range is shown in Figure 23.

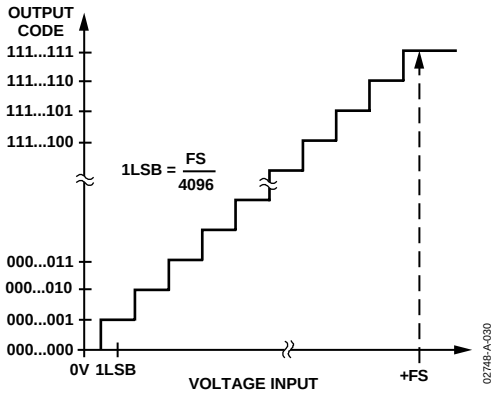



*Figure 23. ADuC814 ADC Transfer Function*

### *ADC Data Output Format*

Once configured via the ADCCON1–3 SFRs, the ADC converts the analog input and provides an ADC 12-bit result word in the ADCDATAH/L SFRs. The ADCDATAL SFR contains the bottom 8 bits of the 12-bit result. The bottom nibble of the ADCDATAH SFR contains the top 4 bits of the result, while the top nibble contains the channel ID of the ADC channel which has been converted on. This ID corresponds to the channel selection bits CD3–CD0 in the ADCCON2 SFR. The format of the ADC 12-bit result word is shown in Figure 24.

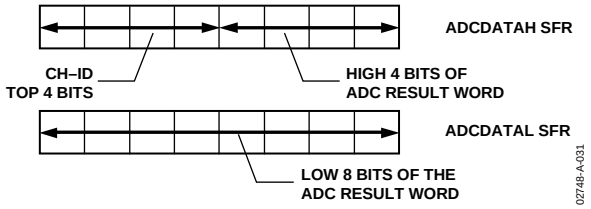



*Figure 24. ADC Result Format*

## SFR INTERFACE TO ADC BLOCK

The ADC operation is fully controlled via three SFRs: ADCCON1, ADCCON2, and ADCCON3. These three registers control the mode of operation.

## ADCCON1 (ADC CONTROL SFR 1)

The ADCCON1 register controls conversion and acquisition times, hardware conversion modes, and power-down modes as detailed below.

| | |
|---|---|
| SFR Address | EFH |
| SFR Power-on Default | 00H |
| Bit Addressable | No |

| MODE | EXT_REF | CK1 | CK0 | AQ1 | AQ0 | T2C | EXC |
|---|---|---|---|---|---|---|---|

**Table 6. ADCCON1 SFR Bit Designations**

| Bit No. | Name | Description |
|---|---|---|
| 7 | MODE | Mode Bit.<br>This bit selects the operating mode of the ADC.<br>Set to 1 by the user to power on the ADC.<br>Set to 0 by the user to power down the ADC. |
| 6 | EXT_REF | External Reference Select Bit.<br>This bit selects which reference the ADC uses when performing a conversion.<br>Set to 1 by the user to switch in an external reference.<br>Set to 0 by the user to switch in the on-chip band gap reference. |
| 5 | CK1 | ADC Clock Divide Bits. |
| 4 | CK0 | CK1 and CK0 combine to select the divide ratio for the PLL master clock used to generate the ADC clock. To ensure correct ADC operation, the divider ratio must be chosen to reduce the ADC clock to 4.5 MHz and below. The divider ratio is selected as follows:<br>CK1 CK0 PLL Divider<br>0 0 8<br>0 1 4<br>1 0 16<br>1 1 32 |
| 3 | AQ1 | The ADC Acquisition Time Select Bits. |
| 2 | AQ0 | AQ1 and AQ0 combine to select the number of ADC clocks required for the input track-and-hold amplifier to acquire the input signal. The acquisition time is selected as follows:<br>AQ1 AQ0 No. ADC Clks<br>0 0 1<br>0 1 2<br>1 0 3<br>1 1 4 |
| 1 | T2C | The Timer2 Conversion Bit.<br>T2C is set to enable the Timer2 overflow bit to be used as the ADC convert start trigger input. |
| 0 | EXC | The External Trigger Enable Bit.<br>EXC is set to allow the external CONVST pin be used as the active low convert start trigger input. When enabled, a rising edge on this input pin trigger a conversion. This pin should remain low for a minimum pulse width of 100 nsec at the required sample rate. |

## ADCCON2 (ADC CONTROL SFR 2)

The ADCCON2 (byte addressable) register controls ADC channel selection and conversion modes as detailed below.

| | |
|---|---|
| SFR Address | D8H |
| SFR Power-On Default | 00H |
| Bit Addressable | Yes |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| ADCI | ADCSPI | CCONV | SCOVC | CS3 | CS2 | CS1 | CS0 |

**Table 7. ADCCON2 SFR Bit Designations**

| Bit No. | Name | Description |
|---|---|---|
| 7 | ADCI | ADC Interrupt Bit.<br>ADCI is set at the end of a single ADC conversion cycle. If the ADC interrupt is enabled, the ADCI bit is cleared when user code vectors to the ADC interrupt routine. Otherwise the ADCI bit should be cleared by the user code. |
| 6 | ADCSPI | ADCSPI Mode Enable Bit.<br>ADCSPI is set to enable the ADC conversion results to be transferred directly to the SPI data buffer (SPIDAT) without intervention from the CPU. |
| 5 | CCONV | Continuous Conversion Bit.<br>CCONV is set to initiate the ADC into a continuous mode of conversion. In this mode the ADC starts converting based on the timing and channel configuration already set up in the ADCCON SFRs. The ADC automatically starts another conversion once a previous conversion cycle has completed. When operating in this mode from 3 V supplies, the ADC should be configured for ADC clock divide of 16 using CK1 and CK0 bits in ADCCON1, and ADC acquisition time should be set to four ADC clocks using AQ1, AQ0 bits in ADCCON1 SFR. |
| 4 | SCONV | Single Conversion Bit.<br>SCONV is set to initiate a single conversion cycle. The SCONV bit is automatically reset to 0 on completion of the single conversion cycle. When operating in this mode from 3 V supplies, the maximum ADC sampling rate should not exceed 147 kSPS. |
| 3 | CS3 | Channel Selection Bits. |
| 2 | CS2 | CS3–CS0 allow the user to program the ADC channel selection under software control. Once a conversion is initiated, the channel converted is pointed to by these channel selection bits. |
| 1 | CS1 | |
| 0 | CS0 | The Channel Select bits operate as follows: |

| CS3 | CS2 | CS1 | CS0 | CHANNEL | |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | |
| 0 | 0 | 0 | 1 | 1 | |
| 0 | 0 | 1 | 0 | 2 | |
| 0 | 0 | 1 | 1 | 3 | |
| 0 | 1 | 0 | 0 | 4 | |
| 0 | 1 | 0 | 1 | 5 | |
| 0 | 1 | 1 | 0 | X | Not a vaild selection. No ADC channel selected. |
| 0 | 1 | 1 | 1 | X | Not a valid selection. No ADC channel selected. |
| 1 | 0 | 0 | 0 | Temperature Sensor | |
| 1 | 0 | 0 | 1 | DAC0 | |
| 1 | 0 | 1 | 0 | DAC1 | |
| 1 | 0 | 1 | 1 | AGND | |
| 1 | 1 | 0 | 0 | $V_{REF}$ | |

## ADCCON3 (ADC CONTROL SFR 3)

The ADCCON3 register controls the operation of various calibration modes as well as giving an indication of ADC busy status.

SFR Address F5H
SFR Power-On Default 00H

| BUSY | GNCLD | AVGS1 | AVGS0 | OFCLD | MODCAL | TYPECAL | SCAL |
|---|---|---|---|---|---|---|---|

**Table 8. ADCCON3 SFR Bit Designations**

| Bit No. | Name | Description |
|---|---|---|
| 7 | BUSY | ADC Busy Status Bit.<br>BUSY is a read-only status bit that is set during a valid ADC conversion or calibration cycle.<br>Busy is automatically cleared by the core at the end of a conversion or calibration cycle. |
| 6 | GNCLD | Gain Calibration Disable Bit.<br>This bit enables/disables the gain calibration coefficients from affecting the ADC results.<br>Set to 0 to enable gain calibration coefficient<br>Set to 1 to disable gain calibration coefficient. |
| 5 | AVGS1 | Number of Averages Selection Bits. |
| 4 | AVGS0 | This bit selects the number of ADC readings averaged for each bit decision during a calibration cycle.<br>AVGS1 AVGS0 Number of Averages<br>0 0 15<br>0 1 1<br>1 0 31<br>1 1 63 |
| 3 | OFCLD | Offset Calibration Disable Bit.<br>This bit enables/disables the offset calibration coefficients from affecting the ADC results.<br>Set to 0 to enable offset calibration coefficient.<br>Set to 1 to disable the offset calibration coefficient |
| 2 | MODCAL | Calibration Mode Select Bit.<br>This bit should be set to 1 for all calibration cycles. |
| 1 | TYPECAL | Calibration Type Select Bit.<br>This bit selects between offset (zero-scale) and gain (full-scale) calibration.<br>Set to 0 for offset calibration.<br>Set to 1 for gain calibration. |
| 0 | SCAL | Start Calibration Cycle Bit.<br>When set, this bit starts the selected calibration cycle.<br>It is automatically cleared when the calibration cycle is completed. |

# DRIVING THE ADC

The ADC incorporates a successive approximation architecture (SAR) involving a charge-sampled input stage. Each ADC conversion is divided into two distinct phases as defined by the position of the switches in Figure 25. During the sampling phase (with SW1 and SW2 in the track position), a charge proportional to the voltage on the analog input is developed across the input sampling capacitor. During the conversion phase (with both switches in the hold position), the capacitor DAC is adjusted via internal SAR logic until the voltage on Node A is zero, indicating that the sampled charge on the input capacitor is balanced out by the charge being output by the capacitor DAC. The digital value finally contained in the SAR is then latched out as the result of the ADC conversion. Control of the SAR, and timing of acquisition and sampling modes, is handled automatically by built-in ADC control logic. Acquisition and conversion times are also fully configurable under user control.

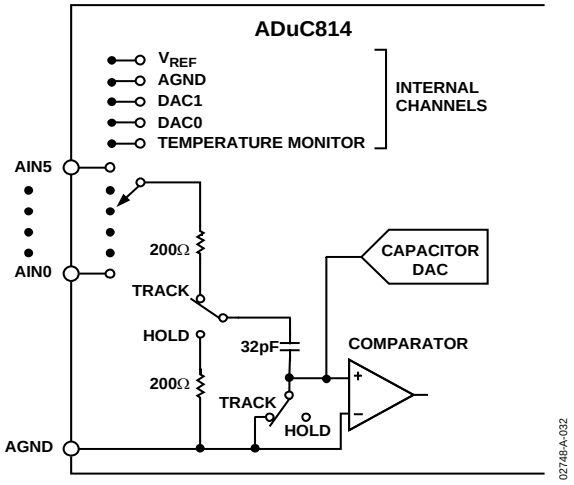



*Figure 25. Internal ADC Structure*

Note that whenever a new input channel is selected, a residual charge from the 32 pF sampling capacitor places a transient on the newly selected input. The signal source must be capable of recovering from this transient before the sampling switches click into hold mode. Delays can be inserted in software (between channel selection and conversion request) to account for input stage settling, but a hardware solution alleviates this burden from the software design task and ultimately results in a cleaner system implementation.

One hardware solution is to choose a very fast settling op amp to drive each analog input. Such an op amp would need to fully settle from a small signal transient in less than 300 ns in order to guarantee adequate settling under all software configurations. A better solution, recommended for use with any amplifier, is shown in Figure 26.

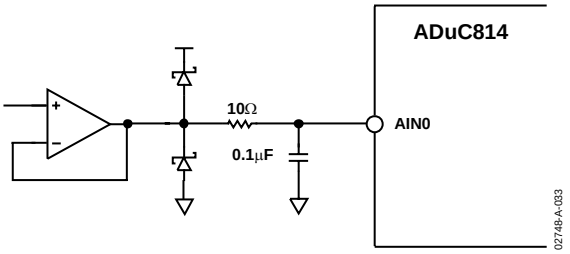



*Figure 26. Buffering Analog Inputs*

At first glance the circuit in Figure 26 may look like a simple anti-aliasing filter, it actually serves no such purpose. Though the R/C does help to reject some incoming high frequency noise, its primary function is to ensure that the transient demands of the ADC input stage are met. It does so by providing a capacitive bank from which the 32 pF sampling capacitor can draw its charge. Since the 0.1 μF capacitor in Figure 26 is more than 3000 times the size of the 32 pF sampling capacitor, its voltage does not change by more than one count of the 12-bit transfer function when the 32 pF charge from a previous channel is dumped onto it. A larger capacitor can be used if desired, but care needs to be taken if choosing a larger resistor (see Table 9).

The Schottky diodes in Figure 26 may be necessary to limit the voltage applied to the analog input pin as per the Absolute Maximum Ratings. They are not necessary if the op amp is powered from the same supply as the ADuC814 because, in that case, the op amp is unable to generate voltages above $V_{DD}$ or below ground. An op amp of some kind is necessary unless the signal source is very low impedance to begin with. DC leakage currents at the ADuC814 analog inputs can cause measurable dc errors with external source impedances as little as 100 Ω or so. To ensure accurate ADC operation, keep the total source impedance at each analog input less than 61 Ω. Table 9 illustrates examples of how source impedance can affect dc accuracy.

**Table 9. Source Impedance Errors**

| Source Impedance | Error from 1 μA Leakage Current | Error from 10 μA Leakage Current |
|---|---|---|
| 61 Ω | 61 μV = 0.1 LSB | 610 μV = 1 LSB |
| 610 Ω | 610 μV = 1 LSB | 6.1 mV = 10 LSB |

Although Figure 26 shows the op amp operating at a gain of 1, you can configure it for any gain needed. Also, you can just as easily use an instrumentation amplifier in its place to condition differential signals. Use any modern amplifier that is capable of delivering the signal (0 V to $V_{REF}$) with minimal saturation. Some single-supply, rail-to-rail op-amps that are useful for this purpose include, but are certainly not limited to, the ones given in Table 10. Check the Analog Devices literature (CD ROM data book, etc.) for details on these and other op amps and instrumentation amps.

**Table 10. Some Single-Supply Op Amps**

| Op Amp Model | Characteristics |
|---|---|
| OP281/OP481 | Micropower |
| OP191/OP291/OP491 | I/O good up to $V_{DD}$, low cost |
| OP196/OP296/OP496 | I/O to $V_{DD}$, micropower, low cost |
| OP183/OP283 | High gain-bandwidth product |
| OP162/OP262/OP462 | High GBP, micropackage |
| AD820/OP822/OP824 | FET input, low cost |
| AD823 | FET input, high GBP |

Keep in mind that the ADC's transfer function is 0 V to $V_{REF}$, and any signal range lost to amplifier saturation near ground impacts dynamic range. Though the op amps in Table 10 are capable of delivering output signals very closely approaching ground, no amplifier can deliver signals all the way to ground when powered by a single supply. Therefore, if a negative supply is available, one could consider using it to power the front end amplifiers. If you do, however, be sure to include the Schottky diodes shown in Figure 26 (or at least the lower of the two diodes) to protect the analog input from undervoltage conditions.

## VOLTAGE REFERENCE CONNECTIONS

The on-chip 2.5 V band gap voltage reference can be used as the reference source for the ADC and DACs. To ensure the accuracy of the voltage reference, decouple the $V_{REF}$ and the $C_{REF}$ pin to ground with 0.1 µF capacitors as shown in Figure 27.

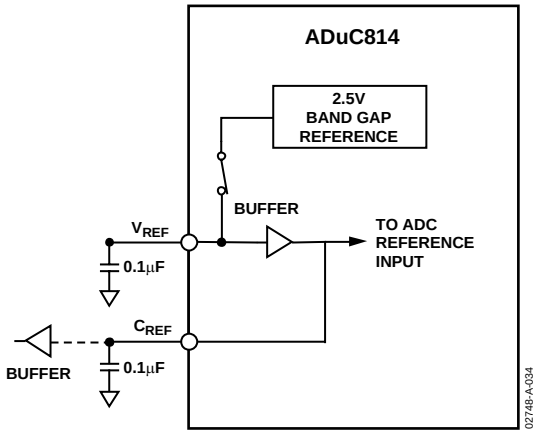



*Figure 27. Decoupling $V_{REF}$ and $C_{REF}$*

If the internal voltage reference is to be used as a reference for external circuitry, the $C_{REF}$ output should be used. However, a buffer must be used in this case to ensure that no current is drawn from the $C_{REF}$ pin itself. The voltage on the $C_{REF}$ pin is that of an internal node within the buffer block, and its voltage is critical to ADC and DAC accuracy. As outlined in the Reference Input/Output section of the Specifications table, the internal band gap reference takes typically 80 msecs to power on and settle to its final value. To ensure accurate ADC operation, one should wait for the ADC to settle after power-on.

If an external voltage reference is preferred, it should be connected to the $V_{REF}$ and $C_{REF}$ pins as shown in Figure 28. Bit 6 of the ADCCON1 SFR must be set to 1 to switch in the external reference voltage. To ensure accurate ADC operation, the voltage applied to $V_{REF}$ must be between 1.0 V and $AV_{DD}$. In situations where analog input signals are proportional to the power supply (such as some strain gage applications) it can be desirable to connect the $V_{REF}$ pin directly to $AV_{DD}$.

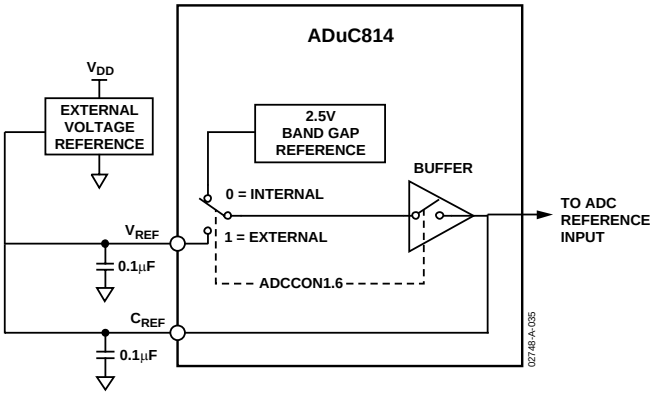



*Figure 28. Using an External Voltage Reference*

Operation of the ADC with a reference voltage below 1.0 V, however, may incur loss of accuracy eventually resulting in missing codes or non-monotonicity. For that reason, do not use a reference voltage less than 1.0 V.

## CONFIGURING THE ADC

In configuring the ADC a number of parameters need to be set up. These parameters can be configured using the three SFRs: ADCCON1, ADCCON2, and ADCCON3, which are detailed in the following sections.

The ADCCLK determines the speed at which the ADC logic runs while performing an ADC conversion. All ADC timing parameters are calculated from the ADCCLK frequency. On the ADuC814, the ADCCLK is derived from the maximum core frequency ($F_{CORE}$), 16.777216 MHz. The ADCCLK frequency is selected via ADCCON1 Bits 5 and 4, which provide four core clock divide ratios of 8, 4, 16, and 32, generating ADCCLK values of 2 MHz, 4 MHz, 1 MHz, and 500 kHz, respectively.

The acquisition time ($T_{ACQ}$) is the number of ADCCLKs that the ADC input circuitry uses to sample the input signal. In most cases, an acquisition time of one ADCCLK provides more than adequate time for the ADuC814 to acquire its signal before switching the internal track-and-hold amplifier into hold mode. The only exception is a high source impedance analog input, but this should be buffered first anyway because high source impedances can cause significant dc errors (see Table 6). ADCCON1 Bits 3 and 2 are used to select acquisition times of 1, 2, 3, and 4 ADCCLKs.

Both the ADCCLK frequency and the acquisition time are used in determining the ADC conversion time. Two other parameters are also used in this calculation. To convert the acquired signal into its corresponding digital output word takes 15 ADCCLK periods ($T_{CONV}$). When a conversion is initiated, the start of conversion signal is synchronized to the ADCCLK. This synchronization ($T_{SYNC}$) can take from 0.5 to 1.5 ADCCLKs to occur. The total ADC conversion time $T_{ADC}$ is calculated using the following formula:

$$T_{ADC} = T_{SYNC} + T_{ACQ} + T_{CONV}$$

Assuming $T_{SYNC} = 1$, $T_{ACQ} = 1$ and $F_{CORE}$/ADCCLK divider of 4. The total conversion time is calculated by

$$T_{ADC} = (1 + 1 + 15) \times (1 / 4194304)$$

$$T_{ADC} = 4.05\ \mu s$$

These settings allow a maximum conversion speed or sampling rate of 246.7 kHz.

When converting on the temperature monitor channel, the conversion time is not controlled via the ADCCON registers. It is controlled in hardware and sets the ADCCLK to $F_{CORE}$ /32 and uses four acquisition clocks, giving a total ADC conversion time of

$$T_{ADC} = (1 + 4 + 15) \times (1 / 524288) = 38.14\ \mu s$$

Increasing the conversion time on the temperature monitor channel improves the accuracy of the reading. To further improve the accuracy, an external reference with low temperature drift should also be used.

## INITIATING ADC CONVERSIONS

After the ADC has been turned on and configured, there are four methods of initiating ADC conversions.

Single conversions can be initiated in software by setting the SCONV bit in the ADCCON2 register via user code. This causes the ADC to perform a single conversion and puts the result into the ADCDATAH/L SFRs. The SCONV bit is cleared as soon as the ADCDATA SFRs have been updated.

Continuous conversion mode can be initiated by setting the CCONV bit in ADCCON2 via user code. This performs back-to-back conversions at the configured rate (246.7 kHz for the settings detailed previously). In continuous mode, the ADC results must be read from the ADCDATA SFRs before the next conversion is completed to avoid loss of data. Continuous mode can be stopped by clearing the CCONV bit.

An external signal can also be used to initiate ADC conversions. Setting Bit 0 in ADCCON1 enables the logic to allow an external start-of-conversion signal on Pin 7 ($\overline{CONVST}$). This active low pulse should be at least 100 ns wide. The rising edge of this signal initiates the conversion.

Timer 2 can also be used to initiate conversions. Setting Bit 1 of ADCCON1 enables the Timer 2 overflow signal to start a conversion. For Timer 2 configuration information, see the Timers/Counters section.

For both external $\overline{CONVST}$ and Timer 2 overflow, the conversion rate must be equal to or greater than the conversion time ($T_{ADC}$) to avoid incorrect ADC results.

When initiating conversions, the user must ensure that only one of the trigger modes is active at any one time. Initiating conversions with more than one of the trigger modes active results in erratic ADC behavior.

## ADC HIGH SPEED DATA CAPTURE MODE

The on-chip ADC has been designed to run at a maximum conversion speed of 4.05 µs (247 kHz sampling rate). When converting at this rate, the ADuC814 MCU has 4.05 µs to read the ADC result and store it in memory for further post processing; otherwise the next ADC sample could be lost. The time to complete a conversion and store the ADC results without errors is known as the throughput rate. In an interrupt driven routine, the MCU also has to jump to the ADC interrupt service routine, which decreases the throughput rate of the ADuC814. In applications where the ADuC814 standard operating mode throughput is not fast enough, an ADC high speed data capture (HSDC) mode is provided.

In HSDC mode, ADC results are transferred to the SPI logic without intervention from the ADuC814 core logic. In applications where the ADC throughput is slow, the HSDC logic operates in non-pipelined mode (Figure 29). In this mode, there is adequate time for the ADC conversion and the ADC-to-SPI data transfer to complete before the next start of conversion. As the ADC throughput increases, the HSDC logic begins to operate in pipelined mode as shown in Figure 30.

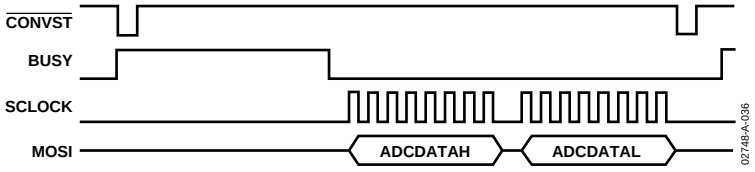



*Figure 29. High Speed Data Capture Logic Timing (Non-Pipelined Mode)*

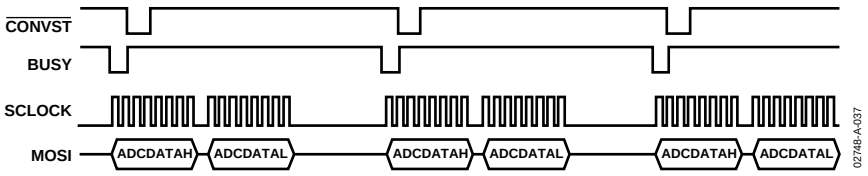



*Figure 30. High Speed Data Capture Logic Timing (Pipelined Mode)*

In this mode, the ADC to SPI data transfer occurs during the next ADC conversion. To avoid loss of an ADC result, the user must ensure that the ADC to SPI transfer rate is complete before the current ADC conversion ends.

To enable HSDC mode, Bit 6 in ADCCON2 (ADCSPI) must be set and to enable the ADuC814 to capture a contiguous sample stream at full ADC update rates (247 kHz).

To configure the ADuC814 in HSDC mode:

1. The ADC must be put into one of its conversion modes.
2. The SPI interface must be configured. (The SPI configuration is detailed in the Serial Peripheral Interface section).
3. Enable HSDC by setting the ADCSPI bit in the ADCCON2 SFR.
4. Apply trigger signal to the ADC to perform conversions.

Once configured and enabled, the ADC results are transferred from the ADCDATAH/L SFRs to the SPIDAT register. Figure 31 shows the HSDC logic configuration once the mode is enabled. The ADC result is transmitted most significant bit first. In this case, the channel ID is transmitted first, followed by the 12-bit ADC result. When this mode is enabled, normal SPI and Port 3 operation is disabled; however, the core is free to continue code execution, including general housekeeping and communication tasks. This mode is disabled by clearing the ADCSPI bit.

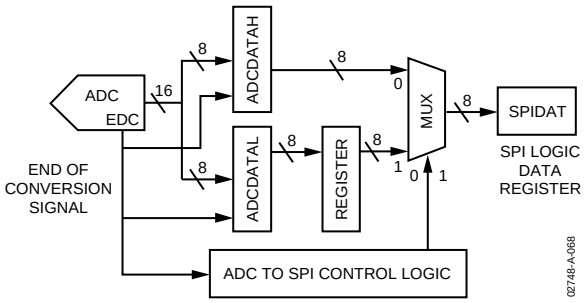



*Figure 31. High Speed Data Capture Logic*

## ADC OFFSET AND GAIN CALIBRATION OVERVIEW

The ADC block incorporates calibration hardware and associated SFRs, which ensures optimum offset and gain performance from the ADC at all times.

As part of internal factory final test routines, the ADuC814 is calibrated to its offset and gain specifications. The offset and gain coefficients obtained from this factory calibration are stored in non-volatile Flash/EE memory. These are downloaded from the Flash/EE memory to offset and gain calibration registers automatically on a power-up or a reset event.

In many applications these factory-generated calibration coefficients suffice. However, the ADuC814 ADC offset and gain accuracy may vary from system to system due to board layout, grounding, clock speed, or system configuration, and so on. To get the best ADC accuracy in your system, an ADC calibration should be performed.

Two main advantages are derived from ensuring the ADC calibration registers are initialized correctly. First, the internal errors in the ADC can be reduced significantly to give superior dc performance; and second, system offset and gain errors can be removed. This allows the user to remove reference errors (whether an internal or external reference) and to use the full dynamic range of the ADC by adjusting the analog input range of the part for a specific system.

## ADC OFFSET AND GAIN CALIBRATION COEFFICIENTS

The ADuC814 has two ADC calibration coefficients, one for offset calibration and one for gain calibration. Both the offset and gain calibration coefficients are 14-bit words, and each is stored in two registers located in the special function register (SFR) area. The offset calibration coefficient is divided into ADCOFSH (6 bits) and ADCOFSL (8 bits), and the gain calibration coefficient is divided into ADCGAINH (6 bits) and ADCGAINL (8 bits).

The offset calibration coefficient compensates for dc offset errors in both the ADC and the input signal. Increasing the offset coefficient compensates for positive offset, and effectively pushes down the ADC transfer function. Decreasing the offset coefficient compensates for negative offset, and effectively pushes up the ADC transfer function. The maximum offset that can be compensated is typically ± 3.5% of $V_{REF}$, which equates to typically ±87.5 mV with a 2.5 V reference.

Similarly, the gain calibration coefficient compensates for dc gain errors in both the ADC and the input signal. Increasing the gain coefficient compensates for a smaller analog input signal range and scales up the ADC transfer function, effectively increasing the slope of the transfer function. Decreasing the

gain coefficient compensates for a larger analog input signal range and scales down the ADC transfer function, effectively decreasing the slope of the transfer function. The maximum analog input signal range for which the gain coefficient can compensate is 1.035 × $V_{REF}$, and the minimum input range is 0.965 × $V_{REF}$, which equates to typically ±3.5% of the reference voltage.

## CALIBRATING THE ADC

The ADuC814 has two hardware calibration modes, device calibration and system calibration, that can be easily initiated by the user software. The ADCCON3 SFR is used to calibrate the ADC. See Table 8.

Device calibration is so called because the relevant signals used for the calibration are available internally to the ADC. This calibration method can be used to compensate for significant changes in operating conditions, such as core frequency, analog input range, reference voltage and supply voltages. In this calibration mode, offset calibration uses internal AGND selected via the ADCCON2 register bits CS3–CS0 (1011), and gain calibration uses internal $V_{REF}$ selected by CS3–CS0 (1100). Offset calibration should be executed first, followed by gain calibration.

System calibration is so called because the AGND and $V_{REF}$ required for calibration must be the system AGND and $V_{REF}$ signals. These must be supplied in turn, externally, to the ADC inputs. This calibration method can be used to compensate for both internal and external system errors. To perform system calibration using an external reference, tie system ground and reference to any two of the six selectable inputs. Enable external reference mode (ADCCON1.6). Select the channel connected to AGND via CS3–CS0 and perform system offset calibration. Select the channel connected to $V_{REF}$ via CS3–CS0 and perform system gain calibration.

## INITIATING CALIBRATION IN CODE

When calibrating the ADC, ADCCON1 should be set to the configuration in which the ADC is used. The ADCCON3 register can then be used to configure and execute the offset and gain calibration required in sequence.

Configure the ADC as required. In this case, ADCCLK = /4, acquisition time is set to 1 clock ($T_{ACQ}$), and ADC is enabled.

```
MOV ADCCON1,#0D0H      ;ADC on, ADCCLK set to
                       ;divide by 4, 1 acquisition
                       ;clock (Tacq)
```

**To perform device offset calibration:**

```
MOV ADCCON2,#0BH       ;select internal AGND
MOV ADCCON3,#25H       ;select offset calibration,
                       ;31 averages per bit,
                       ;offset calibration
```

**To perform device gain calibration:**

```
MOV ADCCON2,#0CH       ;select internal VREF MOV
ADCCON3,#27H           ;select offset calibration,
                       ;31 averages per bit,
                       ;offset calibration
```

**To perform system offset calibration:**

Connect system AGND to an ADC input (Channel 0 in this case).

```
MOV ADCCON2,#00H       ;select external AGND
MOV ADCCON3,#25H       ;select offset calibration,
                       ;31 averages per bit
```

**To perform system gain calibration:**

Connect system $V_{REF}$ to an ADC input (Channel 1 in this case).

```
MOV ADCCON2,#01H       ;select external VREF MOV
ADCCON3,#27H           ;select offset calibration,
                       ;31 averages / bit (NUMAV),
                       ;offset calibration
```

The calibration cycle time $T_{CAL}$ is calculated by

$$T_{CAL} = 14 \times ADCCLK \times NUMAV \times (16 + T_{ACQ})$$

For an ADCCLK/$F_{CORE}$, divide ratio of 4, a $T_{ACQ}$ = 1 ADCCLK, NUMAV = 31, the calibration cycle time is

$$T_{CAL} = 14 \times (1 / 4194304) \times 31 \times (16 + 1)$$

$$T_{CAL} = 1.76\ mS$$

In a calibration cycle, the ADC busy flag (Bit 7), instead of framing an individual ADC conversion as in normal mode, goes high at the start of calibration and returns to zero only at the end of the calibration cycle. It can therefore be monitored in code to indicate when the calibration cycle is completed. The following code can be used to monitor the BUSY signal during a calibration cycle.

```
WAIT:    MOV A, ADCCON3    ;move ADCCON3 to A
         JB ACC.7, WAIT    ;If bit 7 is set jump to
                            WAIT
                           ;else continue
```

# NONVOLITILE FLASH/EE MEMORY

## FLASH/EE MEMORY OVERVIEW

The ADuC814 incorporates Flash/EE memory technology on-chip to provide the user with nonvolatile, in-circuit reprogrammable code and data memory space.

Flash/EE memory takes the flexible in-circuit reprogrammable features of EEPROM and combines them with the space efficient/density features of EPROM (see Figure 32).

Because Flash/EE technology is based on a single transistor cell architecture, a Flash memory array such as EPROM can be implemented to achieve the space efficiencies or memory densities required by a given design.

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

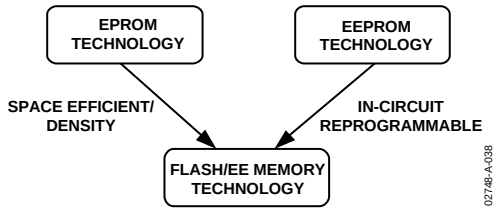



*Figure 32. Flash/EE Memory Development*

Incorporated in the ADuC814, Flash/EE memory technology allows the user to update program code space in-circuit without the need to replace one-time programmable (OTP) devices at remote operating nodes.

## FLASH/EE MEMORY AND THE ADUC814

The ADuC814 provides two arrays of Flash/EE memory for user applications. There are 8 kbytes of Flash/EE program space provided on-chip to facilitate code execution, therefore removing the requirement for an external discrete ROM device. The program memory can be programmed using conventional third-party memory programmers. This array can also be programmed in-circuit, using the serial download mode provided.

A 640-byte Flash/EE data memory space is also provided on-chip. This may be used as a general-purpose nonvolatile scratchpad area. User access to this area is via a group of six SFRs. This space can be programmed at a byte level, although it must first be erased in 4-byte pages.

## ADUC814 FLASH/EE MEMORY RELIABILITY

The Flash/EE program and data memory arrays on the ADuC814 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. In real terms, a single endurance cycle is composed of four independent, sequential events:

1. Initial page erase sequence
2. Read/verify sequence
3. Byte program sequence
4. Second read/verify sequence

In reliability qualification, every byte in both the program and data Flash/EE memory is cycled from 00H to FFH until a first fail is recorded signifying the endurance limit of the on-chip Flash/EE memory.

As indicated in the Specifications tables, the ADuC814 Flash/EE memory endurance qualification has been carried out in accordance with JEDEC Specification A117 over the industrial temperature range of –40°C to +125°C. The results allow the specification of a minimum endurance figure over supply and a temperature of 100,000 cycles, with an endurance figure of 700,000 cycles being typical of operation at 25°C.

Retention quantifies the ability of the Flash/EE memory to retain its programmed data over time. Again, the ADuC814 has been qualified in accordance with the formal JEDEC Retention Lifetime Specification (A117) at a specific junction temperature ($T_J$ = 55°C). As part of this qualification procedure, the Flash/EE memory is cycled to its specified endurance limit described above, before data retention is characterized. This means that the Flash/EE memory is guaranteed to retain its data for its full specified retention lifetime every time the Flash/EE memory is reprogrammed. It should be noted that retention lifetime, based on an activation energy of 0.6 eV, derates with $T_J$ as shown in Figure 33.

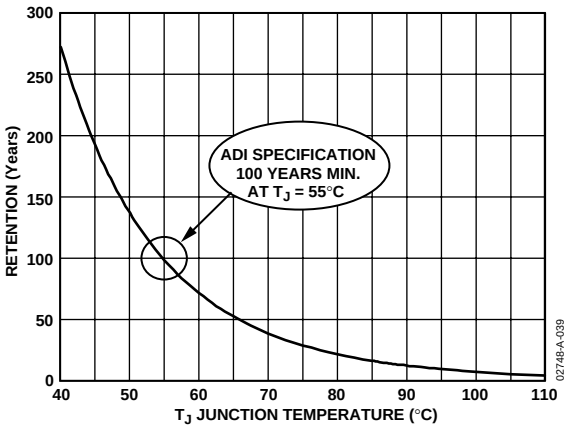



*Figure 33. Flash/EE Memory Data Retention*

## USING FLASH/EE PROGRAM MEMORY

The Flash/EE program memory array can be programmed in one of two modes: serial downloading and parallel programming.

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

As part of its factory boot code, the ADuC814 facilitates code download via the standard UART serial port. Serial download mode is automatically entered on power-up or during a hardware RESET operation if the external DLOAD pin is pulled high through an external resistor, as shown in Figure 34. Once in this mode, the user can download code to the program memory array while the device is sited in its target application hardware. A PC serial download executable is provided as part of the ADuC814 QuickStart development system. The Serial Download protocol is detailed in a MicroConverter Applications Note uC004 available from the ADI MicroConverter website at www.analog.com/microconverter.

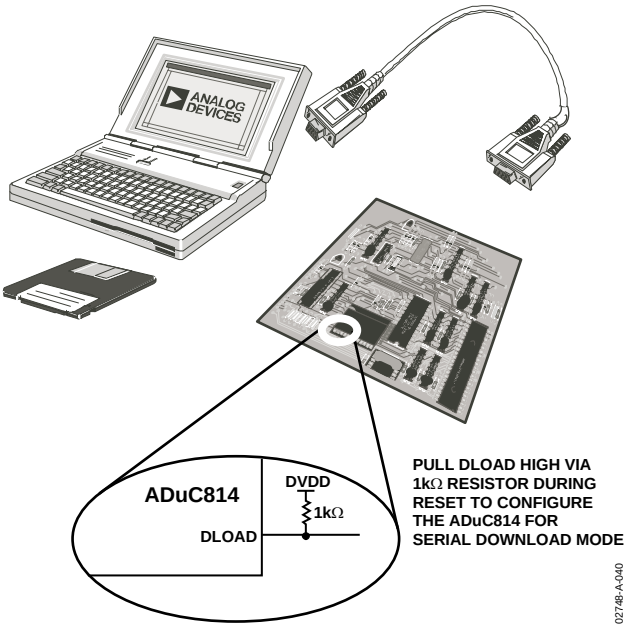



*Figure 34. Flash/EE Memory Serial Download Mode Programming*

### *Parallel Programming*

The parallel programming mode is fully compatible with conventional third-party Flash or EEPROM device programmers. A block diagram of the external pin configuration required to support parallel programming is shown in Figure 35.

The high voltage (12 V) supply required for Flash/EE programming is generated using on-chip charge pumps to supply the high voltage program lines.

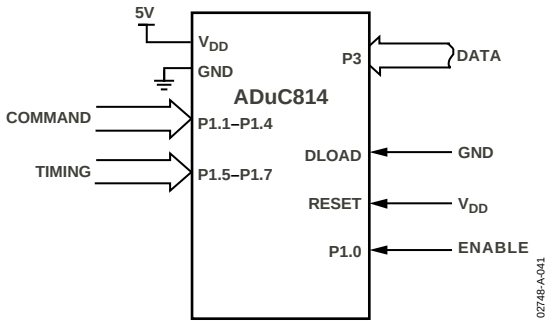



*Figure 35. Flash/EE Memory Parallel Programming*

## FLASH/EE PROGRAM MEMORY SECURITY

The ADuC814 facilitates three modes of Flash/EE program memory security, which are described in the following sections. These modes can be independently activated, restricting access to the internal code space. These security modes can be enabled as part of the user interface available on all ADuC814 serial or parallel programming tools referenced on the MicroConverter website at www.analog.com/microconverter.

### *Lock Mode*

This mode locks code in memory, disabling parallel programming of the program memory, although reading the memory in parallel mode is still allowed. This mode is deactivated by initiating a CODE-ERASE command in serial download or parallel programming modes.

### *Secure Mode*

This mode locks code in memory, disabling parallel programming (program and VERIFY/READ commands). This mode is deactivated by initiating a CODE-ERASE command in serial download or parallel programming modes.

### *Serial Safe Mode*

This mode disables serial download capability on the device. If serial safe mode is activated and an attempt is made to reset the part into serial download mode, i.e., RESET asserted and de-asserted with DLOAD high, the part interprets the serial download reset as a normal reset only. It therefore does not enter serial download mode but only executes a normal reset sequence. Serial safe mode can be disabled only by initiating a CODE-ERASE command in parallel programming mode.

## USING FLASH/EE DATA MEMORY

The user Flash/EE data memory array consists of 640 bytes that are configured into 160 (00H to 9FH) 4-byte pages as shown in Figure 36.

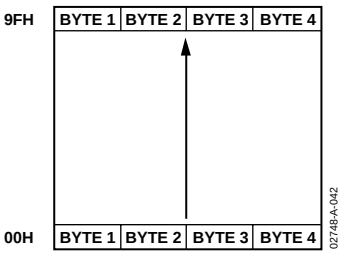



*Figure 36. Flash/EE Data Memory Configuration*

As with other ADuC814 user-peripheral circuits, the interface to this memory space is via a group of registers mapped in the SFR space. EADRL is used to hold the 8-bit address of the page to be accessed. A group of four data registers (EDATA1–4) is used to hold 4-byte page data just accessed. Finally, ECON is an 8-bit control register that may be written with one of five Flash/EE memory access commands to trigger various read, write, erase, and verify functions. These registers can be summarized as follows:

**ECON**

| | |
|---|---|
| SFR Address | B9H |
| Function | Controls access to 640 bytes Flash/EE data space. |
| Default | 00H |

**EADRL**

| | |
|---|---|
| SFR Address | C6H |
| Function | Holds the Flash/EE data page address. (640 bytes = > 160 page addresses) |
| Default | 00H |

**EDATA1–4**

| | |
|---|---|
| SFR Address | BCH to BFH, respectively |
| Function | Holds Flash/EE data memory page write or page read data bytes. |
| Default | EDATA1–4 > 00H |

A block diagram of the SFR interface to the Flash/EE data memory array is shown in Figure 37.

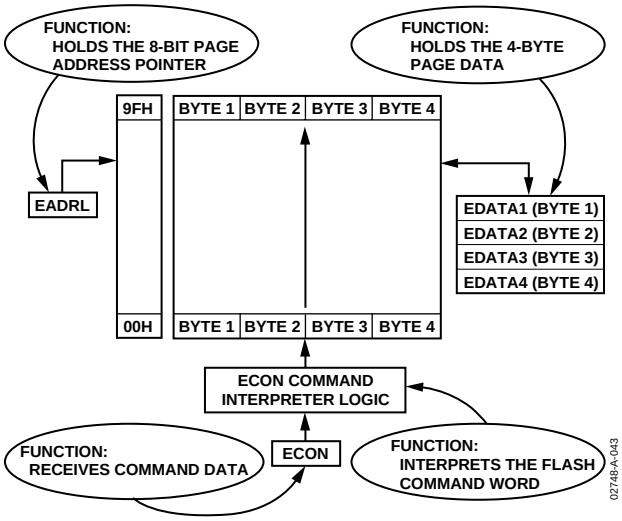



*Figure 37. Flash/EE Data Memory Control and Configuration*

### *ECON—Flash/EE Memory Control SFR*

This SFR acts as a command interpreter and may be written with one of five command modes to enable various read, program, and erase cycles as detailed in Table 11.

**Table 11. ECON–Flash/EE Memory Control Register Command Modes**

| Command Byte | Command Mode | Description |
|---|---|---|
| 01H | READ | Results in 4 bytes being read into EDATA1–4 from memory page address contained in EADRL. |
| 02H | PROGRAM | Results in 4 bytes (EDATA1–4) being written to memory page address in EADRL. This write command assumes the designated write page has been erased. |
| 03H | Reserved | For internal use. 03H should not be written to the ECON SFR. |
| 04H | VERIFY | Allows the user to verify if data in EDATA1–4 is contained in page address designated by EADRL. A subsequent read of the ECON SFR results in a zero being read if the verification is valid, a nonzero value is read to indicate an invalid verification. |
| 05H | ERASE | Results in an erase of the 4-byte page designated in EADRL. |
| 06H | ERASE-ALL | Results in an erase of the full Flash/EE aata memory, 160-page (640 bytes) array. |
| 07H to FFH | Reserved | For future use. |

## FLASH/EE MEMORY TIMING

The typical program/erase times for the Flash/EE data memory are

| | |
|---|---|
| Erase Full Array (640 bytes) | 2 ms |
| Erase Single Page (4 bytes) | 2 ms |
| Program Page (4 bytes) | 250 µs |
| Read Page (4 bytes) | Within single instruction cycle |

### *Using the Flash/EE Memory Interface*

As with all Flash/EE memory architectures, the array can be programmed in-system at a byte level, although it must be erased first, the erasure being performed in page blocks (4-byte pages in this case).

A typical access to the Flash/EE data array involves setting up the page address to be accessed in the EADRL SFR, configuring the EDATA1–4 with data to be programmed to the array (the EDATA SFRs are not written to for read accesses), and finally, writing the ECON command word, which initiates one of the modes shown Table 11.

Note that a given mode of operation is initiated as soon as the command word is written to the ECON SFR. The core micro-controller operation on the ADuC814 is idled until the requested program/read or erase mode is completed. In practice, this means that even though the Flash/EE memory mode of operation is typically initiated with a two-machine cycle MOV instruction (to write to the ECON SFR), the next instruction cannot be executed until the Flash/EE operation is complete (250 µs or 2 ms later). This means that the core does not respond to interrupt requests until the Flash/EE operation is complete, though the core peripheral functions like counter/timers continue to count and time as configured throughout this period. Although the 640-byte user Flash/EE array is shipped from the factory pre-erased, i.e., byte locations set to FFH, it is nonetheless good programming practice to include an ERASE-ALL routine as part of any configuration/setup code running on the ADuC814. An ERASE-ALL command consists of writing 06H to the ECON SFR, which initiates an erasure of all 640 byte locations in the Flash/EE array. This command coded in 8051 assembly appears as

```
MOV ECON, #06H          ; Erase all Command
                        ; 2 ms Duration
```

### *Programming a Byte*

In general terms, a byte in the Flash/EE array can be programmed only if it has been erased previously. To be more specific, a byte can only be programmed if it already holds the value FFH. Because of the Flash/EE architecture, this erasure must happen at a page level; therefore, a minimum of four bytes (1 page) are erased when an ERASE command is initiated.

A more specific example of the program-byte process is shown below. In this example the user writes F3H into the second byte on Page 03H of the Flash/EE data memory space while preserving the other three bytes already in this page. Because the user is required to modify only one of the page bytes, the full page must be first read so that this page can then be erased without the existing data being lost. This example, coded in 8051 assembly, appears as

```
MOV EADRL,#03H          ; Set Page Address
                        ; Pointer
MOV ECON,#01H           ; Read Page
MOV EDATA2,#0F3H        ; Write New Byte
MOV ECON,#05H           ; Erase Page
MOV ECON,#02H           ; Write Page
                        ; Program Flash/EE)
```

# USER INTERFACE TO OTHER ON-CHIP ADuC814 PERIPHERALS

This section gives a brief overview of the various peripherals also available on-chip. A summary of the SFRs used to control and configure these peripherals is also given.

## DACS

The ADuC814 incorporates two 12-bit, voltage output DACs on-chip. Each DAC has a rail-to-rail voltage output buffer capable of driving 10 kΩ/100 pF. They have two selectable ranges, 0 V to $V_{REF}$ (an external or the internal band gap 2.5 V reference) and 0 V to $AV_{DD}$, and can operate in 12-bit or 8-bit modes. DAC operation is controlled by a single special function (SFR) register, DACCON. Each DAC has two data registers, DACxH/L. The DAC0 and DAC1 outputs share pins with ADC inputs ADC4 and ADC5, respectively. When both DACs are on, the number of analog inputs is reduced to four. Note that in 12-bit mode, the DAC voltage output is updated as soon as the DACL data SFR has been written; therefore, the DAC data registers should be updated as DACH first, followed by DACL.

When using the DACs on the $V_{REF}$ range it is necessary to power up the ADC to enable the reference to the DAC section. See Note 1.

| **DACCON** | **DAC Control Register** |
|---|---|
| SFR Address | FDH |
| Power-On Default | 04H |
| Bit Addressable | No |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MODE | RNG1 | RNG0 | $\overline{CLR1}$ | $\overline{CLR0}$ | SYNC | $\overline{PD1}$ | $\overline{PD0}$ |

**Table 12. DACCON SFR Bit Designations**

| Bit No. | Name | Description |
|---|---|---|
| 7 | MODE | Mode Select Bit. Selects either 12-bit or 8-bit mode for both DACs.<br>Set to 1 by the user to enable 8-bit mode (DACxL is the active data register).<br>Set to 0 by the user to enable 12-bit mode. |
| 6 | RNG1 | DAC1 Output Voltage Range Select Bit.<br>Set to 1 by the user to configure DAC1 range of 0 V to $AV_{DD}$.<br>Set to 0 by the user to configure DAC1 range of 0 V to 2.5 V ($V_{REF}$ range) [1]. |
| 5 | RNG0 | DAC0 Output Voltage Range Select Bit.<br>Set to 1 by he user to configure DAC0 range of 0 V to $AV_{DD}$.<br>Set to 0 by the user to configure DAC0 range of 0 V to 2.5 V ($V_{REF}$ range) [1]. |
| 4 | $\overline{CLR1}$ | DAC1 Clear Bit.<br>Set to 1 by the user to enable normal DAC1 operation.<br>Set to 0 by the user to force DAC1 output voltage to 0 V. |
| 3 | $\overline{CLR0}$ | DAC0 Clear Bit.<br>Set to 1 by the user to enable normal DAC0 operation.<br>Set to 0 by the user to force DAC0 output voltage to 0 V. |
| 2 | SYNC | DAC0/1 Update Synchronization Bit.<br>Set to 1 by the user to enable asynchronous update mode. The DAC outputs update as soon as the DACxL SFRs are written.<br>Set to 0 by the user to enable synchronous update mode. The user can simultaneously update both DACs by first updating the DACxH/L SFRs while SYNC is 0. Both DACs then update simultaneously when the SYNC bit is set to 1. |
| 1 | $\overline{PD1}$ | DAC1 Power-Down Bit.<br>Set to 1 by the user to power up DAC1.<br>Set to 0 by the user to power down DAC1. |
| 0 | $\overline{PD0}$ | DAC0 Power-Down Bit.<br>Set to 1 by the user to power up DAC0.<br>Set to 0 by the user to power down DAC0. |

[1]For correct DAC operation on the 2.5 V to $V_{REF}$ range, the ADC must be powered on.

| DACxH/L | DAC0 and DAC1 Data Registers |
|---|---|
| Function | DAC Data Registers, written by the user to update the DAC outputs. |
| SFR Address | DAC0L (DAC0 data low byte) –> F9H DAC0H (DAC0 data high byte) –> FAH; |
| | DAC1L (DAC1 data low byte) –> FBH DAC1H (DAC1 data high byte) –> FCH |
| Power-On Default | 00H –> Both DAC0 and DAC1 data registers. |
| Bit Addressable | No –> Both DAC0 and DAC1 data registers. |

The 12-bit DAC data should be written into DACxH/L right-justified such that DACL contains the lower eight bits, and the lower nibble of DACH contains the upper four bits.

### *Using the DACs*

The on-chip DAC architecture consists of a resistor string DAC followed by an output buffer amplifier, the functional equivalent of which is illustrated in Figure 38. Features of this architecture include inherent guaranteed monotonicity and excellent differential linearity.

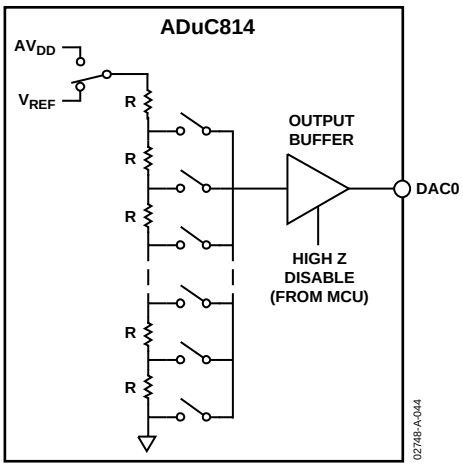



*Figure 38. Resistor String DAC Functional Equivalent*

As illustrated in Figure 38, the reference source for each DAC is user selectable in software. It can be either $AV_{DD}$ or $V_{REF}$. In 0 V-to-$AV_{DD}$ mode, the DAC output transfer function spans from 0 V to the voltage at the $AV_{DD}$ pin. In 0 V-to-$V_{REF}$ mode, the DAC output transfer function spans from 0 V to the internal $V_{REF}$, or if an external reference is applied, the voltage at the $V_{REF}$ pin. The DAC output buffer features a true rail-to-rail output stage implementation. This means that, unloaded, each output is capable of swinging to within less than 100 mV of both $AV_{DD}$ and ground. Moreover, the DAC's linearity specification (when driving a 10 kΩ resistive load to ground) is guaranteed through the full transfer function except Codes 0 to 48, and, in 0 V-to-$AV_{DD}$ mode only, Codes 3945 to 4095. Linearity degradation near ground and $V_{DD}$ is caused by saturation of the output buffer, and a general representation of its effects (neglecting offset and gain error) is illustrated in Figure 39. The dotted line in Figure 39 indicates the ideal transfer function, and the solid line represents what the transfer function might look like with endpoint nonlinearities due to saturation of the output buffer. Note that Figure 39 represents a transfer function in 0 V-to-$V_{DD}$ mode only. In 0 V-to-$V_{REF}$ mode (with $V_{REF} < V_{DD}$), the lower nonlinearity would be similar, but the upper portion of the transfer function would follow the ideal line right to the end ($V_{REF}$ in this case, not $V_{DD}$), showing no signs of upper endpoint linearity error.

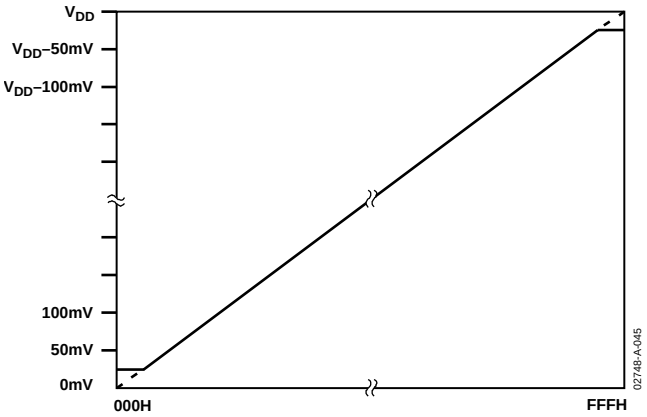



*Figure 39. Endpoint Nonlinearities Due to Amplifier Saturation*

The endpoint nonlinearities conceptually illustrated in Figure 39 get worse as a function of output loading. Most ADuC814 specifications assume a 10 kΩ resistive load to ground at the DAC output. As the output is forced to source or sink more current, the nonlinear regions at the top or bottom (respectively) of Figure 39 become larger. With larger current demands, this can significantly limit output voltage swing. Figure 40 and Figure 41 illustrate this behavior. Note that the upper trace in each of these figures is valid only for an output range selection of 0 V-to-$AV_{DD}$. In 0 V-to-$V_{REF}$ mode, DAC loading does not cause high-side voltage drops as long as the reference voltage remains below the upper trace in the corresponding figure. For example, if $AV_{DD}$ = 3 V and $V_{REF}$ = 2.5 V, the high-side voltage is not affected by loads less than 5 mA. But around 7 mA, the upper curve in Figure 41 drops below 2.5 V ($V_{REF}$), indicating that at these higher currents the output cannot reach $V_{REF}$.

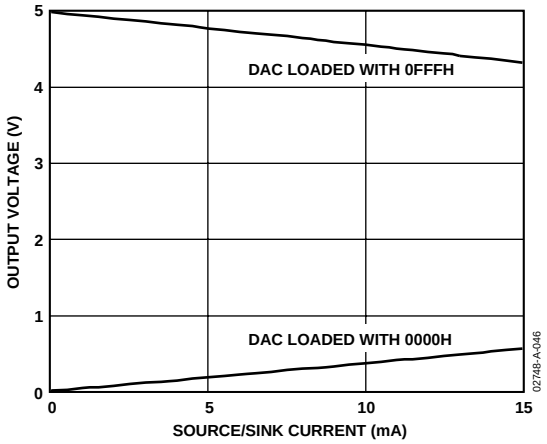



*Figure 40. Source and Sink Current Capability with $V_{REF} = V_{DD} = 5$ V*

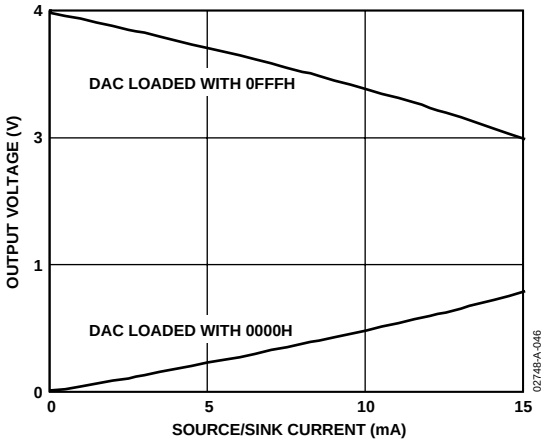



*Figure 41. Source and Sink Current Capability with $V_{REF} = V_{DD} = 3$ V*

For larger loads, the current drive capability may not be sufficient. To increase the source and sink current capability of the DACs, an external buffer should be added, as shown in Figure 42.

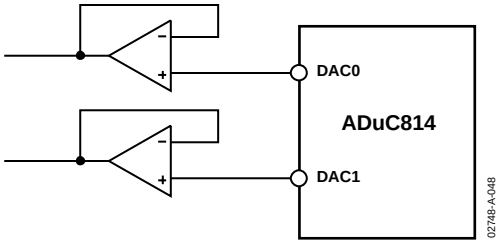



*Figure 42. Buffering the DAC Outputs*

The DAC output buffer also features a high impedance disable function. In the chip's default power-on state, both DACs are disabled, and their outputs are in a high impedance state (or three-state) where they remain inactive until enabled in software. This means that if a zero output is desired during power-up or power-down transient conditions, then a pull-down resistor must be added to each DAC output. Assuming this resistor is in place, the DAC outputs remain at ground potential whenever the DAC is disabled.

## ON-CHIP PLL

The ADuC814 is intended for use with a 32.768 kHz watch crystal. An on-board PLL locks onto a multiple (512) of this 32.768kHz frequency to provide a stable 16.777216 MHz clock for the system. The core can operate at this frequency or at binary submultiples of it to allow power saving in cases where maximum core performance is not required. The default core clock is the PLL clock divided by 8 ($2^{CD} = 2^3$) or 2.097152 MHz. The PLL is controlled via the PLLCON special function register.

| **PLLCON** | **PLL Control Register** |
|---|---|
| SFR Address | D7H |
| Power-On Default | 03H |
| Bit Addressable | No |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| OSC_PD | LOCK | --- | --- | FINT | CD2 | CD1 | CD0 |

**Table 13. PLLCON SFR Bit Designations**

| Bit No. | Name | Description |
|---|---|---|
| 7 | OSC_PD | Oscillator Power-Down Bit.<br>Set by the user to halt the 32 kHz oscillator in power-down mode.<br>Cleared by the user to enable the 32 kHz oscillator in power-down mode. This feature allows the oscillator to continue clocking the TIC even in power-down mode. |
| 6 | LOCK | PLL Lock Bit. This is a read-only bit.<br>Set automatically at power-on to indicate that the PLL loop is correctly tracking the crystal clock. If the external crystal becomes subsequently disconnected, the PLL rails and the core halts.<br>Cleared automatically at power-on to indicate that the PLL is not correctly tracking the crystal clock. This may be due to the absence of a crystal clock or an external crystal at power-on. In this mode, the PLL output is 16.78 MHz ± 20%. |
| 5 | --- | Reserved. Should be written with 0. |
| 4 | --- | Reserved. Should be written with 0. |
| 3 | FINT | Fast Interrupt Response Bit.<br>Set by the user to enable the response to any interrupt to be executed at the fastest core clock frequency, regardless of the configuration of the CD2–CD0 bits (see below). Once user code has returned from an interrupt, the core resumes code execution at the core clock selected by the CD2–CD0 bits.<br>Cleared by the user to disable the fast interrupt response feature. |
| 2 | CD2 | CPU (Core Clock) Divider Bits. |
| 1 | CD1 | This number determines the frequency at which the microcontroller core operates. |
| 0 | CD0 | |

| CD2 | CD1 | CD0 | Core Clock Frequency (MHz) |
|---|---|---|---|
| 0 | 0 | 0 | 16.777216 |
| 0 | 0 | 1 | 8.388608 |
| 0 | 1 | 0 | 4.194304 |
| 0 | 1 | 1 | 2.097152 (Default Core Clock Frequency) |
| 1 | 0 | 0 | 1.048576 |
| 1 | 0 | 1 | 0.524288 |
| 1 | 1 | 0 | 0.262144 |
| 1 | 1 | 1 | 0.131072 |

## TIME INTERVAL COUNTER (TIC)

A time interval counter is provided on-chip for counting longer intervals than the standard 8051 compatible timers are capable of. The TIC is capable of time-out intervals ranging from 1/128th second to 255 hours. Furthermore, this counter is clocked by the crystal oscillator rather than by the PLL and thus has the ability to remain active in power-down mode and to time long power-down intervals. This has obvious applications for remote battery-powered sensors where regular, widely spaced readings are required.

Six SFRs are associated with the time interval counter, TIMECON being its control register. Depending on the configuration of the IT0 and IT1 bits in TIMECON, the selected time counter register overflow clocks the interval counter. When this counter is equal to the time interval value loaded in the INTVAL SFR, the TII bit (TIMECON.2) is set and generates an interrupt if enabled. (See IEIP2 SFR description under the Interrupt System section.) If the ADuC814 is in power-down mode, again with TIC interrupt enabled, the TII bit wakes up the device and resumes code execution by vectoring directly to the TIC interrupt service vector address at 0053H. The TIC-related SFRs are described in Table 14. Note that the time based SFRs can be written initially with the current time; the TIC can then be controlled and accessed by the user software. In effect, this facilitates the implementation of a real-time clock. A block diagram of the TIC is shown in Figure 43.

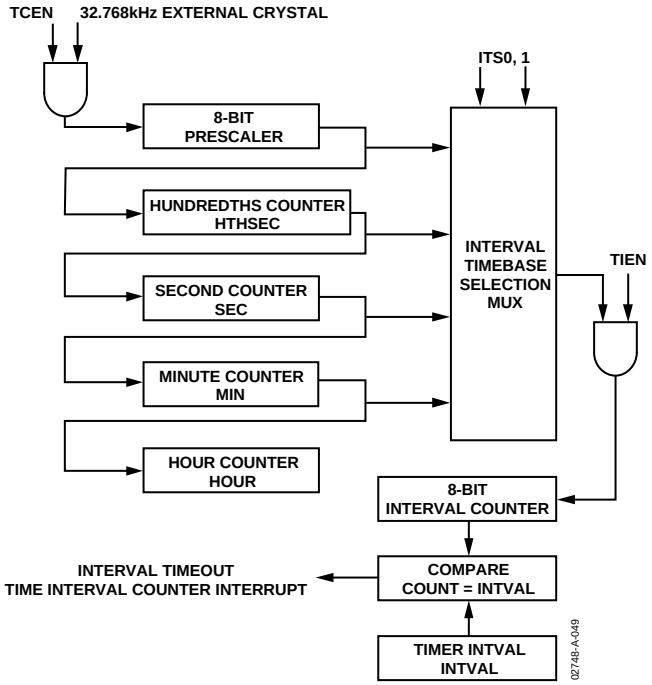



*Figure 43. Time Interval Counter, Simplified Block Diagram*

| | |
|---|---|
| **TIMECON** | **TIC CONTROL REGISTER** |
| SFR Address | A1H |
| Power-On Default | 00H |
| Bit Addressable | No |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| --- | TFH | ITS1 | ITS0 | STI | TII | TIEN | TCEN |

**Table 14. TIMECON SFR Bit Designations**

| Bit No. | Name | Description |
|---|---|---|
| 7 | --- | Reserved. |
| 6 | TFH | Twenty-Four Hour Select Bit.<br>Set by the user to enable the HOUR counter to count from 0 to 23.<br>Cleared by the user to enable the HOUR counter to count from 0 to 255.<br>The time interval counter continues to count after a reset when in hours/min/sec mode. If the part is in 24 hour mode though, this bit is reset and the part now counts in 255 hour mode. The following code segment can be used to set the TIC back into 24 hour mode after a RESET event.<br>`MOV A,TIMECON ;Move contents of TIMECON into ACC`<br>`RRC A ;Rotate ACC right by 1 place into Carry`<br>`JNC NOTSET ;If CARRY bit is ! = 1 jump to NOTSET, else continue with next line`<br>`ORL TIMECON,#01000000B`<br>`;If CARRY bit = 1 for last line, then logical OR TIMECON with 40H`<br>`NOTSE: ;continuation of normal code from here` |
| | ITS1 | Interval Timebase Selection Bits. |
| 4 | ITS0 | Written by the user to determine the interval counter update rate.<br>ITS1 ITS0 Interval Timebase<br>0 0 1/128 Second<br>0 1 Seconds<br>1 0 Minutes<br>1 1 Hours |
| 3 | STI | Single Time Interval Bit.<br>Set by the user to generate a single interval timeout. If set, a timeout clears the TIEN bit.<br>Cleared by the user to allow the interval counter to be automatically reloaded and start counting again at each interval timeout. |
| 2 | TII | TIC Interrupt Bit.<br>Set when the 8-bit interval counter matches the value in the INTVAL SFR.<br>Cleared by the user software. |
| 1 | TIEN | Time Interval Enable Bit.<br>Set by the user to enable the 8-bit time interval counter.<br>Cleared by the user to disable and clear the contents of the interval counter. |
| 0 | TCEN | Time Clock Enable Bit.<br>Set by the user to enable the time clock to the time interval counters.<br>Cleared by the user to disable the clock to the time interval counters and clear the time interval SFRs.<br>The time registers (HTHSEC, SEC, MIN and HOUR) can be written while TCEN is low. |

| | |
|---|---|
| **INTVAL** | **User Time Interval Select Register** |
| Function | User code writes the required time interval to this register. When the 8-bit interval counter is equal to the time interval value loaded in the INTVAL SFR, the TII bit (TIMECON.2) is set and generates an interrupt if enabled. (See the IEIP2 SFR description in the Interrupt System section.) |
| SFR Address | A6H |
| Power-On Default | 00H |
| Bit Addressable | No |
| Valid Value | 0 to 255 decimal |

| | |
|---|---|
| **HTHSEC** | **Hundredths Seconds Time Register** |
| Function | This register is incremented in 1/128 second intervals once TCEN in TIMECON is active. The HTHSEC SFR counts from 0 to 127 before rolling over to increment the SEC time register. |
| SFR Address | A2H |
| Power-On Default | 00H |
| Bit Addressable | No |
| Valid Value | 0 to 127 decimal |

| | |
|---|---|
| **SEC** | **Seconds Time Register** |
| Function | This register is incremented in 1-second intervals once TCEN in TIMECON is active. The SEC SFR counts from 0 to 59 before rolling over to increment the MIN time register. |
| SFR Address | A3H |
| Power-On Default | 00H |
| Bit Addressable | No |
| Valid Value | 0 to 59 decimal |

| | |
|---|---|
| **MIN** | **Minutes Time Register** |
| Function | This register is incremented in 1-minute intervals once TCEN in TIMECON is active. The MIN SFR counts from 0 to 59 before rolling over to increment the HOUR time register. |
| SFR Address | A4H |
| Power-On Default | 00H |
| Bit Addressable | No |
| Valid Value | 0 to 59 decimal |

| | |
|---|---|
| **HOUR** | **Hours Time Register** |
| Function | This register is incremented in 1-hour intervals once TCEN in TIMECON is active. If the TFH bit (TIMECON.6) is set to 1 the HOUR SFR counts from 0 to 23 before rolling over to 0. If the TFH bit is set to 0, the HOUR SFR counts from 0 to 255 before rolling over to 0. |
| SFR Address | A5H |
| Power-On Default | 00H |
| Bit Addressable | No |
| Valid Value | 0 to 23 decimal |

## WATCHDOG TIMER

The purpose of the watchdog timer is to generate a device reset or interrupt within a reasonable amount of time if the ADuC814 enters an erroneous state, possibly due to a programming error, electrical noise, or RFI. The watchdog function can be disabled by clearing the WDE (watchdog enable) bit in the watchdog control (WDCON) SFR. When enabled, the watchdog circuit generates a system reset or interrupt (WDS) if the user program fails to set the watchdog (WDE) bit within a predetermined amount of time (see PRE3–0 bits in WDCON). The watchdog timer itself is a 16-bit counter that is clocked at 32.768 kHz. The watchdog timeout interval can be adjusted via the PRE3–0 bits in WDCON. Full control and status of the watchdog timer function can be controlled via the watchdog timer control SFR (WDCON). The WDCON SFR can be written only by the user software if the double write sequence (WDWR) described in Table 15 is initiated on every write access to the WDCON SFR.

| **WDCON** | **Watchdog Timer Control Register** |
|---|---|
| SFR Address | C0H |
| Power-On Default | 10H |
| Bit Addressable | Yes |

| PRE3 | PRE2 | PRE1 | PRE0 | WDIR | WDS | WDE | WDWR |
|---|---|---|---|---|---|---|---|

**Table 15. WDCON SFR Bit Designation**

| Bit No. | Name | Description |
|---|---|---|
| 7 | PRE3 | Watchdog Timer Prescale Bits. |
| 6 | PRE2 | The watchdog timeout period is given by the equation |
| 5 | PRE1 | $t_{WD} = (2^{PRE} \times (2^9/f_{PLL}))$ |
| | | where $f_{PLL}$ = 32.768 kHz and PRE is defined as follows: |
| 4 | PRE0 | PRE3 PRE2 PRE1 PRE0 Timeout Period (ms) Action |
| | | 0 0 0 0 15.6 Reset or Interrupt |
| | | 0 0 0 1 31.2 Reset or Interrupt |
| | | 0 0 1 0 62.5 Reset or Interrupt |
| | | 0 0 1 1 125 Reset or Interrupt |
| | | 0 1 0 0 250 Reset or Interrupt |
| | | 0 1 0 1 500 Reset or Interrupt |
| | | 0 1 1 0 1000 Reset or Interrupt |
| | | 0 1 1 1 2000 Reset or Interrupt |
| | | 1 0 0 0 0.0 Immediate Reset |
| | | PRE3–0 > 1001 Reserved |
| 3 | WDIR | Watchdog Interrupt Request. |
| | | If this bit is set by the user, the watchdog generates an interrupt response instead of a system reset when the watchdog timeout period has expired. This interrupt is not disabled by the CLR EA instruction and it is also a fixed, high-priority interrupt. |
| | | If the watchdog is not being used to monitor the system, it can alternatively be used as a timer. The prescaler is used to set the timeout period in which an interrupt is generated. (See Table 33, Note 1, in the Interrupt System section.) |
| 2 | WDS | Watchdog Status Bit. |
| | | Set by the watchdog controller to indicate that a watchdog timeout has occurred. |
| | | Cleared by writing a 0 or by an external hardware reset. It is not cleared by a watchdog reset. |
| 1 | WDE | Watchdog Enable Bit. |
| | | Set by the user to enable the watchdog and clear its counters. If this bit is not set by the user within the watchdog timeout period, the watchdog generates a reset or interrupt, depending on WDIR. |
| | | Cleared under the following conditions: User writes 0, Watchdog Reset (WDIR = 0); Hardware Reset; PSM Interrupt. |
| 0 | WDWR | Watchdog Write Enable Bit. |
| | | To write data into the WDCON SFR involves a double instruction sequence. The WDWR bit must be set and followed immediately by a write instruction to the WDCON SFR. For example: |

```
CLR EA           ; disable interrupts while writing to WDT
SETB WDWR        ; allow write to WDCON
MOV WDCON, #72H  ; enable WDT for 2.0s timeout
SET B EA         ; enable interrupts again (if rqd)
```

## POWER SUPPLY MONITOR

As its name suggests, the power supply monitor, once enabled, monitors the supply ($DV_{DD}$) on the ADuC814. It indicates when any of the supply pins drop below one of four user-selectable voltage trip points from 2.63 V to 4.63 V. For correct operation of the power supply monitor function, $DV_{DD}$ must be equal to or greater than 2.7 V. Monitor function is controlled via the PSMCON SFR. If enabled via the IEIP2 SFR, the monitor interrupts the core using the PSMI bit in the PSMCON SFR.

This bit is not cleared until the failing power supply has returned above the trip point for at least 250 ms. This monitor function allows the user to save working registers to avoid possible data loss due to the low supply condition, and also ensures that normal code execution does not resume until a safe supply level is well established. The supply monitor is also protected against spurious glitches triggering the interrupt circuit.

| **PSMCON** | **Power Supply Monitor Control Register** |
|---|---|
| SFR Address | DFH |
| Power-On Default | DEH |
| Bit Addressable | No |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| ---- | CMPD | PSMI | TPD1 | TPD0 | ---- | ---- | PSMEN |

**Table 16. PSMCON SFR Bit Designations**

| Bit No. | Name | Description |
|---|---|---|
| 7 | PSMCON.7 | Reserved. |
| 6 | CMPD | $DV_{DD}$ Comparator Bit.<br>This is a read-only bit and directly reflects the state of the $DV_{DD}$ comparator.<br>Read 1 indicates that the $DV_{DD}$ supply is above its selected trip point.<br>Read 0 indicates that the $DV_{DD}$ supply is below its selected trip point. |
| 5 | PSMI | Power Supply Monitor Interrupt Bit.<br>This bit is set high by the MicroConverter if CMPD is low, indicating low digital supply. The PSMI bit can be used to interrupt the processor. Once CMPD returns and remains high, a 250 ms counter is started. When this counter times out, the PSMI interrupt is cleared. PSMI can also be written by the user; however, if the comparator output is low, it is not possible for the user to clear PSMI. |
| 4 | TPD1 | $DV_{DD}$ Trip Point Selection Bits. |
| 3 | TPD0 | These bits select the $DV_{DD}$ trip-point voltage as follows:<br>TPD1 TPD0 Selected $DV_{DD}$ Trip Point (V)<br>0 0 4.63<br>0 1 3.08<br>1 0 2.93<br>1 1 2.63 |
| 2 | PSMCON.2 | Reserved. |
| 1 | PSMCON.1 | Reserved. |
| 0 | PSMEN | Power Supply Monitor Enable Bit.<br>Set to 1 by the user to enable the power supply monitor circuit.<br>Cleared to 0 by the user to disable the power supply monitor circuit. |

## ADuC814 CONFIGURATION REGISTER (CFG814)

The ADuC814 is housed in a 28-lead TSSOP package. To maintain as much functional compatibility with other MicroConverter products, some pins share multiple I/O functionality. Switching between these functions is controlled via the ADuC814 configuration SFR, CFG814, located at SFR address 9CH. A summary of these functions is described and a detailed bit designation for the CFG814 SFR is given in Table 17.

### *Serial Peripheral Interface*

The SPI interface on the ADuC814 shares the same pins as digital outputs P3.5, P3.6, and P3.7. The SPE bit in SPICON is used to select which interface is active at any one time. This is described in greater detail in the next section. By default, these pins operate as standard Port 3 pins. Bit 0 of the CFG814 SFR must be set to 1 to enable the SPI interface on these Port 3 pins.

### *External Clock*

The ADuC814 is intended for use with a 32.768 kHz watch crystal. The on-chip PLL locks onto a multiple of this to provide a stable 16.777216 MHz clock for the device. On the ADuC814, P3.5 alternate functions include T1 input and slave select in SPI master mode. P3.5 also functions as external clock input, EXTCLK, selected via Bit 1 of the CFG814 SFR. When selected, this external clock bypasses the PLL and is used as the clock for the device, therefore allowing the ADuC814 to be synchronized to the rest of the application system. The maximum input frequency of this external clock is 16.777216 MHz. If selected, the EXTCLK signal affects the timing of the majority of peripherals on the ADuC814 including the ADC, EEPROM controller, watchdog timer, SPI interface clock, and the MicroConverter core clock.

| **CFG814** | **ADuC814 Configuration Register** |
|---|---|
| SFR Address | 9CH |
| Power-On Default | 04H |
| Bit Addressable | No |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | | | | EXTCLK | SER_EN |

**Table 17. CFG814 SFR Bit Designations**

| Bit No. | Name | Description |
|---|---|---|
| 1 | EXTCLK | External Clock Selection Bit.<br>Set to 1 to enable EXTCLK as MCU core clock.<br>Cleared to 0 to enable XTAL and PLL as the MCU core clock. |
| 0 | SER_EN | Serial Interface Enable Bit.<br>Set to 1 by the user to enable the SPI interface onto the P3.5, P3.6, and P3.7 pins.<br>Cleared to 0 by the user to enable standard Port 3 functionality on the P3.5, P3.6, and P3.7 pins. |

## SERIAL PERIPHERAL INTERFACE

The ADuC814 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 received simultaneously, i.e., full duplex. Note that the SPI pins MISO and MOSI are multiplexed with digital outputs P3.6 and P3.7. These pins are controlled via the CFG814.0 bit in the CFG814 SFR (Table 17), which configures the relevant Port 3 pins for normal operation or serial port operation. When the relevant Port 3 pins are configured for serial interface operation via the CFG814 SFR, the SPE bit in the SPICON SFR configures SPI or $I^2C$ operation (see SPE bit description in Table 18). SPI can be configured for master or slave operation, and typically consists of four pins described next.

### *MISO (Master In, Slave Out Data I/O Pin)*

The MISO pin (Pin 23) is configured as an input line in master mode and as an output line in slave mode. The MISO line on the master (data in) 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 (Pin 24) is configured as an output line in master mode and as an input line in slave mode. The MOSI line on the master (data out) 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.

### *SCLOCK (Serial Clock I/O Pin)*

The SCLOCK pin (Pin 25) is used to synchronize the data being transmitted and received through the MOSI and MISO data lines. A single data bit is transmitted and received in each SCLOCK period. Therefore, a byte is transmitted/received after eight SCLOCK periods. The SCLOCK pin is configured as an output in master mode and as an input in slave mode.

In master mode, the bit rate, polarity, and phase of the clock are controlled by the CPOL, CPHA, SPR0, and SPR1 bits in the SPICON SFR (see Table 18). In slave mode, the SPICON register must be configured with the phase and polarity (CPHA and CPOL) of the expected input clock. In both master and slave modes, the data is transmitted on one edge of the SCLOCK signal and sampled on the other. It is important, therefore, that the CPHA and CPOL are configured the same for the master and slave devices.

### *$\overline{SS}$ (Slave Select Input Pin)*

The $\overline{SS}$ input pin (Pin 22) is used only when the ADuC814 is configured in slave mode to enable the SPI peripheral. This line is active low. Data is received or transmitted in slave mode only when the $\overline{SS}$ pin is low, allowing the ADuC814 to be used in single master, multislave SPI configurations. If CPHA = 1, then the $\overline{SS}$ input may be permanently pulled low. With CPHA = 0, the $\overline{SS}$ input must be driven low before the first bit in a byte-wide transmission or reception and return high again after the last bit in that byte-wide transmission or reception. In SPI slave mode, the logic level on the external $\overline{SS}$ pin can be read via the SPR0 bit in the SPICON SFR.

The following SFR registers are used to control the SPI interface.

| **SPICON** | **SPI Control Register** |
|---|---|
| SFR Address | F8H |
| Power-On Default | 04H |
| Bit Addressable | Yes |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| ISPI | WCOL | SPE | SPM | CPOL | CPHA | SPR1 | SPR0 |

**Table 18. SPICON SFR Bit Designations**

| Bit No. | Name | Description |
|---|---|---|
| 7 | ISPI | SPI Interrupt Bit.<br>Set by the MicroConverter at the end of each SPI transfer.<br>Cleared directly by the user code or indirectly by reading the SPIDAT SFR. |
| 6 | WCOL | Write Collision Error Bit.<br>Set by the MicroConverter if SPIDAT is written to while an SPI transfer is in progress.<br>Cleared by the user. |
| 5 | SPE | SPI Interface Enable Bit.<br>Set by the user to enable the SPI interface.<br>Cleared by the user to enable the $I^2C$ interface. |
| 4 | SPIM | SPI Master/Slave Mode Select Bit.<br>Set by the user to enable master mode operation (SCLOCK is an output).<br>Cleared by the user to enable slave mode operation (SCLOCK is an input). |
| 3 | CPOL[1] | Clock Polarity Select Bit.<br>Set by the user if SCLOCK idles high. Cleared by the user if SCLOCK idles low. |

| Bit No. | Name | Description |
|---|---|---|
| 2 | CPHA[1] | Clock Phase Select Bit.<br>Set by the user if the leading SCLOCK edge is to transmit data.<br>Cleared by the user if the trailing SCLOCK edge is to transmit data. |
| 1 | SPR1 | SPI Bit Rate Select Bits. |
| 0 | SPR0 | These bits select the SCLOCK rate (bit rate) in master mode as follows:<br>SPR1 SPR0 Selected Bit Rate<br>0 0 $f_{CORE}/2$<br>0 1 $f_{CORE}/4$<br>1 0 $f_{CORE}/8$<br>1 1 fcore/16<br>In SPI slave mode,where SPIM = 0, the logic level on the external $\overline{SS}$ pin (Pin 22), can be read via the SPR0 bit. |

[1]The CPOL and CPHA bits should both contain the same values for master and slave devices.

| **SPIDAT** | **SPI Data Register** |
|---|---|
| Function | The SPIDAT SFR is written by the user to transmit data over the SPI interface or read by the user code to read data just received by the SPI interface. |
| SFR Address | F7H |
| Power-On Default | 00H |
| Bit Addressable | No |

### *Using the SPI Interface*

Depending on the configuration of the bits in the SPICON SFR shown in Table 18, the ADuC814 SPI interface transmits or receives data in a number of possible modes. Figure 44 shows all possible ADuC814 SPI configurations and the timing relationships and synchronization between the signals involved.

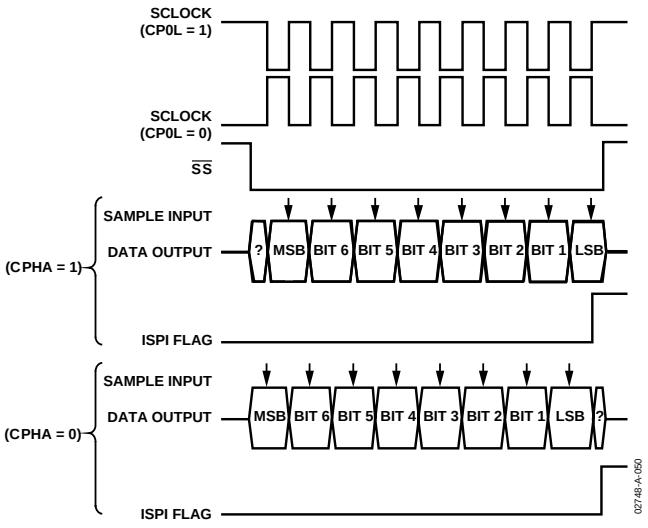



*Figure 44. ADuC814, SPI Timing, All Modes*

### *SPI Interface—Master Mode*

In master mode, the SCLOCK pin is always an output and generates a burst of eight clocks whenever user code writes to the SPIDAT register. The SCLOCK bit rate is determined by SPR0 and SPR1 in SPICON. It should also be noted that the $\overline{SS}$ pin is not used in master mode. If the ADuC814 needs to assert the $\overline{SS}$ pin on an external slave device, a port digital output pin should be used. In master mode, a byte transmission or reception is initiated by a write to SPIDAT. Eight clock periods are generated via the SCLOCK pin and the SPIDAT byte being transmitted via MOSI. With each SCLOCK period, a data bit is also sampled via MISO. After eight clocks, the byte is completely transmitted, and the input byte is waiting in the input shift register. The ISPI flag is set automatically and an interrupt occurs if enabled. The value in the shift register is latched into SPIDAT.

### *SPI Interface—Slave Mode*

In slave mode, the SCLOCK is an input. The $\overline{SS}$ pin must also be driven low externally during the byte communication. Transmission is also initiated by a write to SPIDAT. In slave mode, a data bit is transmitted via MISO, and a data bit is received via MOSI on each input SCLOCK. After eight clocks, the byte is completely transmitted and the input byte is waiting in the input shift register. The ISPI flag is set automatically and an interrupt occurs if enabled. The value in the shift register is latched into SPIDAT only when the transmission/reception of a byte is complete. The end of transmission occurs after the eighth clock is received, if CPHA = 1, or when $\overline{SS}$ returns high if CPHA = 0.

## I²C COMPATIBLE INTERFACE

The ADuC814 supports a 2-wire serial interface mode that is I²C compatible. The I²C compatible interface shares its pins with the on-chip SPI interface, and therefore the user can enable only one interface or the other at any given time (see the SPE bit in SPICON SFR, Table 18). Application Note uC001 describes the operation of this interface as implemented, and is available on the MicroConverter website at www.analog.com/microconverter. This interface can be configured as a software master or hardware slave, and uses two pins in the interface.

**SDATA** (Pin 24) Serial Data I/O

**SCLOCK** (Pin 25) Serial Clock

Three SFRs are used to control the I²C compatible interface.

| **I2CCON** | **I²C Control Register** |
|---|---|
| SFR Address | E8H |
| Power-On Default | 00H |
| Bit Addressable | Yes |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| MDO | MDE | MCO | MDI | I2CM | I2CRS | I2CTX | I2CI |

**Table 19. I2CCON SFR Bit Designations**

| Bit No. | Name | Description |
|---|---|---|
| 7 | MDO | I²C Software Master Data Output Bit (Master Mode Only).<br>This data bit is used to implement a master I²C transmitter interface in software. Data written to this bit is output on the SDATA pin if the data output enable (MDE) bit is set. |
| 6 | MDE | I²C Software Master Data Output Enable Bit (Master Mode Only).<br>Set by the user to enable the SDATA pin as an output (Tx).<br>Cleared by the user to enable SDATA pin as an input (Rx). |
| 5 | MCO | I²C Software Master Clock Output Bit (Master Mode Only).<br>This data bit is used to implement a master I²C transmitter interface in software. Data written to this bit is output on the SCLOCK pin. |
| 4 | MDI | I²C Software Master Data Input Bit (Master Mode Only).<br>This data bit is used to implement a master I²C receiver interface in software. Data on the SDATA pin is latched into this bit on SCLOCK if the data output enable (MDE) bit is 0. |
| 3 | I2CM | I²C Master/Slave Mode Bit.<br>Set by the user to enable I²C software master mode.<br>Cleared by the user to enable I²C hardware slave mode. |
| 2 | I2CRS | I²C Reset Bit (Slave Mode Only).<br>Set by the user to reset the I²C interface.<br>Cleared by the user code for normal I²C operation. |
| 1 | I2CTX | I²C Direction Transfer Bit (Slave Mode Only).<br>Set by the MicroConverter if the interface is transmitting.<br>Cleared by the MicroConverter if the interface is receiving. |
| 0 | I2CI | I²C Interrupt Bit (Slave Mode Only).<br>Set by the MicroConverter after a byte has been transmitted or received.<br>Cleared automatically when user code reads the I2CDAT SFR (see the I2CDAT SFR description that follows). |

| **I2CADD** | **I²C Address Register** |
|---|---|
| Function | Holds the I²C peripheral address for the part. It may be overwritten by the user code. Application Note uC001 at www.analog.com/microconverter describes the format of the 7-bit address in detail. |
| SFR Address | 9BH |
| Power-On Default | 55H |
| Bit Addressable | No |

| **I2CDAT** | **I²C Data Register** |
|---|---|
| Function | The I2CDAT SFR is written to by the user code to transmit data over the I²C interface, or it is read by the user code to read data just received via the I²C interface. Accessing the I2CDAT register automatically clears any pending I²C interrupts and the I2CI bit in the I2CCON SFR. |
| SFR Address | 9AH |
| Power-On Default | 00H |
| Bit Addressable | No |

## 8051 COMPATIBLE ON-CHIP PERIPHERALS

This section gives a brief overview of the various secondary peripheral circuits that are available to the user on-chip. These functions are fully 8051 compatible and are controlled via standard 8051 SFR bit definitions.

### *Parallel I/O Ports 1 and 3*

The ADuC814 has two input/output ports. In addition to performing general-purpose I/O, some ports are multiplexed with an alternate function for the peripheral features on the device. In general, when a peripheral is enabled, that pin may not be used as a general purpose I/O pin.

Port 1 is an 8-bit port directly controlled via the P1 SFR (SFR address = 90H). The Port 1 pins are divided into two distinct pin groupings.

P1.0 and P1.1 pins on Port 1 are bidirectional digital I/O pins with internal pull-ups. If P1.0 and P1.1 have 1s written to them via the P1 SFR, these pins are pulled high by the internal pull-up resistors. In this state they can also be used as inputs. As input pins being externally pulled low, they source current because of the internal pull-ups. With 0s written to them, both of these pins drive a logic low output voltage (VOL) and are capable of sinking 10 mA compared to the standard 1.6 mA sink capability on the other port pins. These pins also have various secondary functions described in Table 20.

**Table 20. Port 1, Alternate Pin Functions**

| Pin No. | Alternate Function |
|---|---|
| P1.0 | T2 (Timer/Counter 2 External Input) |
| P1.1 | T2EX (Timer/Counter 2 Capture/Reload Trigger) |

The remaining Port 1 pins (P1.2–P1.7) can be configured only as analog input (ADC), analog output (DAC) or digital input pins. By default (power-on) these pins are configured as analog inputs, that is, 1 written in the corresponding Port 1 register bit. To configure any of these pins as digital inputs, the user should write a 0 to these port bits to configure the corresponding pin as a high impedance digital input.

Port 3 is a bidirectional port with internal pull-ups directly controlled via the P3 SFR (SFR address = B0H). Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and in that state, they can be used as inputs. As inputs, Port 3 pins being pulled externally low sources current because of the internal pull-ups. Port 3 pins also have various secondary functions described in Table 21.

**Table 21. Port 3, Alternate Pin Functions**

| Pin No. | Alternate Function |
|---|---|
| P3.0 | RxD (UART Input Pin) |
| | (or Serial Data I/O in Mode 0) |
| P3.1 | TxD (UART Output Pin) |
| | (or Serial Clock Output in Mode 0) |
| P3.2 | INT0 (External Interrupt 0) |
| P3.3 | INT1 (External Interrupt 1) |
| P3.4 | T0 (Timer/Counter 0 External Input) |
| P3.5 | T1 (Timer/Counter 1 External Input) |
| | $\overline{SS}$ (Slave Select in SPI Slave Mode) |
| P3.6 | MISO (Master in Slave Out in SPI Mode) |
| P3.7 | MOSI (Master Out Slave In in SPI Mode) |

### *Additional Digital Outputs Pins*

Pins P1.0 and P1.1 can be used to provide high current (10 mA sink) general-purpose I/O.

## TIMERS/COUNTERS

The ADuC814 has three 16-bit timer/counters: Timer 0, Timer 1, and Timer 2. The timer/counter hardware has been included on-chip to relieve the processor core of the overhead inherent in implementing timer/counter functionality in software. Each timer/counter consists of two 8-bit registers, THx and TLx (x = 0, 1 and 2). All three can be configured to operate either as timers or event counters.

In timer function, the TLx register is incremented every machine cycle. Thus one can think of it as counting machine cycles. Since a machine cycle consists of 12 core clock periods, the maximum count rate is 1/12 of the core clock frequency.

In counter function, the TLx register is incremented by a 1-to-0 transition at its corresponding external input pin, T0, T1, or T2. In this function, the external input is sampled during S5P2 of every machine cycle. When the samples show a high in one cycle and a low in the next cycle, the count is incremented. The new count value appears in the register during S3P1 of the cycle following the one in which the transition was detected. Since it takes two machine cycles (16 core clock periods) to recognize a 1-to-0 transition, the maximum count rate is 1/16 of the core clock frequency. There are no restrictions on the duty cycle of the external input signal, but to ensure that a given level is sampled at least once before it changes, it must be held for a minimum of one full machine cycle. Remember that the core clock frequency is programmed via the CD0–CD2 selection bits in the PLLCON SFR. User configuration and control of all timer operating modes is achieved via three SFRs: TMOD, TCON, and T2CON.

| **TMOD** | **Timer/Counter 0 and 1 Mode Register** |
|---|---|
| SFR Address | 89H |
| Power-On Default | 00H |
| Bit Addressable | No |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| GATE | C/$\overline{T}$ | M1 | M0 | GATE | C/$\overline{T}$ | M1 | M0 |

**Table 22. TMOD SFR Bit Designations**

| Bit | Name | Description |
|---|---|---|
| 7 | GATE | Timer 1 Gating Control.<br>Set by software to enable Timer/Counter 1 only while $\overline{INT1}$ pin is high and TR1 control bit is set.<br>Cleared by software to enable Timer 1 whenever TR1 control bit is set. |
| 6 | C/$\overline{T}$ | Timer 1 Timer or Counter Select Bit.<br>Set by software to select counter operation (input from T1 pin).<br>Cleared by software to select timer operation (input from internal system clock). |
| 5 | M1 | Timer 1 Mode Select Bit 1 (Used with M0 Bit). |
| 4 | M0 | Timer 1 Mode Select Bit 0.<br>M1 M0<br>0 0 TH1 operates as an 8-bit timer/counter. TL1 serves as 5-bit prescaler.<br>0 1 16-Bit Timer/Counter. TH1 and TL1 are cascaded; there is no prescaler.<br>1 0 8-Bit Auto-Reload Timer/Counter. TH1 holds a value which is to be reloaded into TL1 each time it overflows.<br>1 1 Timer/Counter 1 Stopped. |
| 3 | Gate | Timer 0 Gating Control.<br>Set by software to enable Timer/Counter 0 only while $\overline{INT0}$ pin is high and the TR0 control bit is set.<br>Cleared by software to enable Timer 0 whenever the TR0 control bit is set. |
| 2 | C/$\overline{T}$ | Timer 0 Timer or Counter Select Bit.<br>Set by software to select counter operation (input from T0 pin).<br>Cleared by software to select timer operation (input from internal system clock). |
| 1 | M1 | Timer 0 Mode Select Bit 1. |
| 0 | M0 | Timer 0 Mode Select Bit 0.<br>M1 M0<br>0 0 TH0 operates as an 8-bit timer/counter. TL0 serves as 5-bit prescaler.<br>0 1 16-Bit Timer/Counter. TH0 and TL0 are cascaded; there is no prescaler.<br>1 0 8-Bit Autoreload Timer/Counter. TH0 holds a value which is to be reloaded into TL0 each time it overflows.<br>1 1 TL0 is an 8-bit timer/counter controlled by the standard Timer 0 control bits. |

**TCON** **Timer/Counter 0 and 1 Control Register**
SFR Address 88H
Power-On Default 00H
Bit Addressable Yes

| TF1 | TR1 | TF0 | TR0 | IE1[1] | IT1[1] | IE0 | IT0[1] |
|---|---|---|---|---|---|---|---|

[1] These bits are not used in the control of Timer/Counter 0 and 1, but are used instead in the control and monitoring of the external $\overline{INT0}$ and $\overline{INT1}$ interrupt pins.

**Table 23. TCON SFR Bit Designations**

| Bit No. | Name | Description |
|---|---|---|
| 7 | TF1 | Timer 1 Overflow Flag.<br>Set by hardware on a Timer/Counter 1 overflow.<br>Cleared by hardware when the program counter (PC) vectors to the interrupt service routine. |
| 6 | TR1 | Timer 1 Run Control Bit.<br>Set by the user to turn on Timer/Counter 1.<br>Cleared by the user to turn off Timer/Counter 1. |
| 5 | TF0 | Timer 0 Overflow Flag.<br>Set by hardware on a Timer/Counter 0 overflow.<br>Cleared by hardware when the PC vectors to the interrupt service routine. |
| 4 | TR0 | Timer 0 Run Control Bit.<br>Set by the user to turn on Timer/Counter 0.<br>Cleared by the user to turn off Timer/Counter 0. |
| 3 | IE1 | External Interrupt 1 ($\overline{INT1}$) Flag.<br>Set by hardware by a falling edge or zero level being applied to external interrupt pin $\overline{INT1}$, depending on bit IT1 state.<br>Cleared by hardware when the when the PC vectors to the interrupt service routine only if the interrupt was transition-activated. If level-activated, the external requesting source controls the request flag, rather than the on-chip hardware. |
| 2 | IT1 | External Interrupt 1 (IE1) Trigger Type.<br>Set by software to specify edge-sensitive detection, that is, 1-to-0 transition.<br>Cleared by software to specify level-sensitive detection, that is, zero level. |
| 1 | IE0 | External Interrupt 0 ($\overline{INT0}$) Flag.<br>Set by hardware by a falling edge or zero level being applied to external interrupt pin $\overline{INT0}$, depending on bit IT0 state.<br>Cleared by hardware when the PC vectors to the interrupt service routine, but only if the interrupt was transition-activated. If level-activated, the external requesting source controls the request flag, rather than the on-chip hardware. |
| 0 | IT0 | External Interrupt 0 (IE0) Trigger Type.<br>Set by software to specify edge-sensitive detection, that is, 1-to-0 transition.<br>Cleared by software to specify level-sensitive detection, that is, zero level. |

### *Timer/Counter 0 and 1 Data Registers*

Each timer consists of two 8-bit registers. These can be used as independent registers or combined to be a single 16-bit register, depending on the timer mode configuration.

**TH0 and TL0** Timer 0 high byte and low byte.
SFR Address 8CH, 8AH, respectively

**TH1 and TL1** Timer 1 high byte and low byte.
SFR Address 8DH, 8BH, respectively

## TIMER/COUNTER 0 AND 1 OPERATING MODES

The following paragraphs describe the operating modes for Timer/Counters 0 and 1. Unless otherwise noted, it should be assumed that these modes of operation are the same for both Timer 0 and Timer 1.

### *Mode 0 (13-Bit Timer/Counter)*

Mode 0 configures an 8-bit timer/counter with a divide-by-32 prescaler. Figure 45 shows Mode 0 operation.

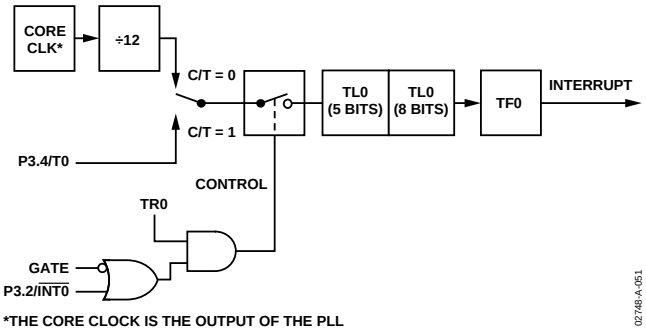



*Figure 45. Timer/Counter 0, Mode 0*

In this mode, the timer register is configured as a 13-bit register. As the count rolls over from all 1s to all 0s, it sets the timer overflow flag TF0. The overflow flag, TF0, can then be used to request an interrupt. The counted input is enabled to the timer when TR0 = 1 and either Gate = 0 or $\overline{INT0}$ = 1.

Setting Gate = 1 allows the timer to be controlled by external input $\overline{INT0}$ to facilitate pulse width measurements. TR0 is a control bit in the special function register TCON; Gate is in TMOD. The 13-bit register consists of all eight bits of TH0 and the lower five bits of TL0. The upper three bits of TL0 are indeterminate and should be ignored. Setting the run flag (TR0) does not clear the registers.

### *Mode 1 (16-Bit Timer/Counter)*

Mode 1 is the same as Mode 0, except that the timer register is running with all 16 bits. Mode 1 is shown in Figure 46.

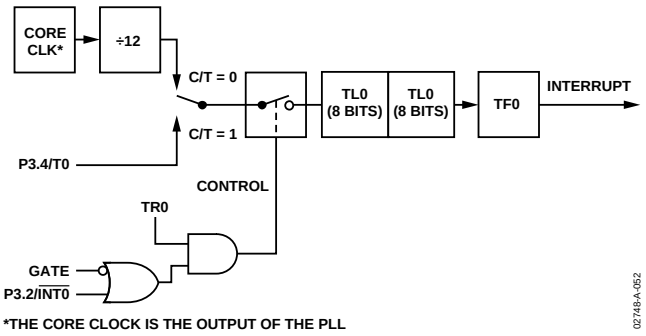



*Figure 46. Timer/Counter 0, Mode 1*

### *Mode 2 (8-Bit Timer/Counter with Autoreload)*

Mode 2 configures the timer register as an 8-bit counter (TL0) with automatic reload, as shown in Figure 47. Overflow from TL0 not only sets TF0, but also reloads TL0 with the contents of TH0, which is preset by software. The reload leaves TH0 unchanged.

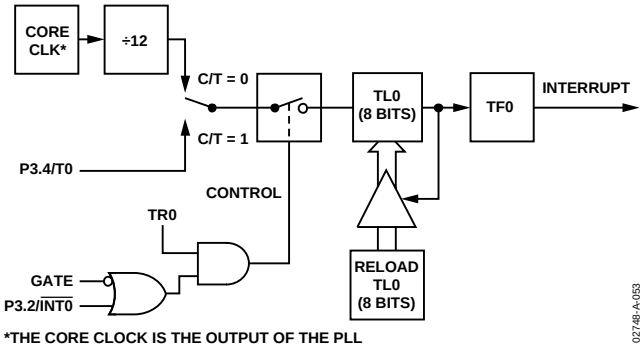



*Figure 47. Timer/Counter 0, Mode 2*

### *Mode 3 (Two 8-Bit Timer/Counters)*

Mode 3 has different effects on Timer 0 and Timer 1. Timer 1 in Mode 3 simply holds its count. The effect is the same as setting TR1 = 0. Timer 0 in Mode 3 establishes TL0 and TH0 as two separate counters. This configuration is shown in Figure 48. TL0 uses the Timer 0 control bits: C/T, GATE, TR0, $\overline{INT0}$, and TF0. TH0 is locked into a timer function (counting machine cycles) and takes over the use of TR1 and TF1 from Timer 1. Thus, TH0 controls the Timer 1 interrupt. Mode 3 is provided for applications requiring an extra 8-bit timer or counter. When Timer 0 is in Mode 3, Timer 1 can be turned on and off by switching it out of, and into, its own Mode 3, or can still be used by the serial interface as a baud rate generator. In fact, it can be used in any application not requiring an interrupt from Timer 1 itself.

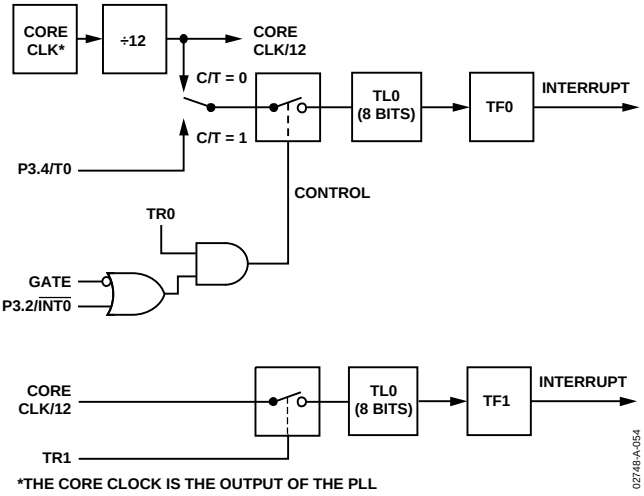



*Figure 48. Timer/Counter 0, Mode 3*

| | |
|---|---|
| **T2CON** | **Timer/Counter 2 Control Register** |
| SFR Address | C8H |
| Power-On Default | 00H |
| Bit Addressable | Yes |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| TF2 | EXF2 | RCLK | TCLK | EXEN2 | TR2 | CNT2 | CAP2 |

**Table 24. T2CON SFR Bit Designations**

| Bit No. | Name | Description |
|---|---|---|
| 7 | TF2 | Timer 2 Overflow Flag.<br>Set by hardware on a Timer 2 overflow. TF2 is not set when either RCLK or TCLK = 1.<br>Cleared by the user software. |
| 6 | EXF2 | Timer 2 External Flag.<br>Set by hardware when either a capture or reload is caused by a negative transition on T2EX and EXEN2 = 1.<br>Cleared by the user software. |
| 5 | RCLK | Receive Clock Enable.<br>Set by the user to enable the serial port to use Timer 2 overflow pulses for its receive clock in serial port in Modes 1 and 3.<br>Cleared by the user to enable Timer 1 overflow to be used for the receive clock. |
| 4 | TCLK | Transmit Clock Enable Bit.<br>Set by the user to enable the serial port to use Timer 2 overflow pulses for its transmit clock in serial port Modes 1 and 3.<br>Cleared by the user to enable Timer 1 overflow to be used for the transmit clock. |
| 3 | EXEN2 | Timer 2 External Enable Flag.<br>Set by the user to enable a capture or reload to occur as a result of a negative transition on T2EX if Timer 2 is not being used to clock the serial port.<br>Cleared by the user for Timer 2 to ignore events at T2EX. |
| 2 | TR2 | Timer 2 Start/Stop Control Bit.<br>Set by the user to start Timer 2.<br>Cleared by the user to stop Timer 2. |
| 1 | CNT2 | Timer 2 timer or counter function select bit.<br>Set by the user to select counter function (input from external T2 pin).<br>Cleared by the user to select timer function (input from on-chip core clock). |
| 0 | CAP2 | Timer 2 Capture/Reload Select Bit.<br>Set by the user to enable captures on negative transitions at T2EX if EXEN2 = 1.<br>Cleared by the user to enable autoreloads with Timer 2 overflows or negative transitions at T2EX when EXEN2 = 1.<br>When either RCLK = 1 or TCLK = 1, this bit is ignored and the timer is forced to autoreload on Timer 2 overflow. |

### *Timer/Counter 2 Data Registers*

Timer/Counter 2 also has two pairs of 8-bit data registers associated with it. These are used as both timer data registers and timer capture/reload registers.

| | |
|---|---|
| **TH2 and TL2** | Timer 2, data high byte and low byte. |
| SFR Address | CDH, CCH, respectively |
| **RCAP2H and RCAP2L** | Timer 2, Capture/Reload byte and low byte. |
| SFR Address | CBH, CAH, respectively |

## TIMER/COUNTER 2 OPERATING MODES

This section describes the operating modes for Timer/Counter 2. The operating modes are selected by bits in the T2CON SFR as shown in Table 27.

**Table 25. Mode Selection in T2CON**

| RCLK (or) TCLK | CAP2 | TR2 | MODE |
|---|---|---|---|
| 0 | 0 | 1 | 16-Bit Autoreload |
| 0 | 1 | 1 | 16-Bit Capture |
| 1 | X | 1 | Baud Rate |
| X | X | 0 | OFF |

### *16-Bit Autoreload Mode*

In autoreload mode, there are two options that are selected by bit EXEN2 in T2CON. If EXEN2 = 0, then when Timer 2 rolls over, it not only sets TF2 but also causes the Timer 2 registers to be reloaded with the 16-bit value in registers RCAP2L and RCAP2H, which are preset by software. If EXEN2 = 1, then Timer 2 still performs the above, but with the added feature that a 1-to-0 transition at external input T2EX also triggers the 16-bit reload and sets EXF2. The autoreload mode is illustrated in Figure 49.

### *16-Bit Capture Mode*

In the capture mode, there are again two options that are selected by bit EXEN2 in T2CON. If EXEN2 = 0, then Timer 2 is a 16-bit timer or counter which, upon overflowing, sets bit TF2, the Timer 2 overflow bit, which can be used to generate an interrupt. If EXEN2 = 1, then Timer 2 still performs the above, but a 1-to-0 transition on external input T2EX causes the current value in the Timer 2 registers, TL2 and TH2, to be captured into registers RCAP2L and RCAP2H, respectively. In addition, the transition at T2EX causes bit EXF2 in T2CON to be set, and EXF2, like TF2, can generate an interrupt. Capture mode is illustrated in Figure 50.

The baud rate generator mode is selected by RCLK = 1 and/or TCLK = 1. In either case, if Timer 2 is being used to generate the baud rate, the TF2 interrupt flag does not occur. Therefore Timer 2 interrupts do not occur so they do not have to be disabled. In this mode, the EXF2 flag, however, can still cause interrupts; this can be used as a third external interrupt. Baud rate generation is described as part of the UART Serial Interface section that follows.

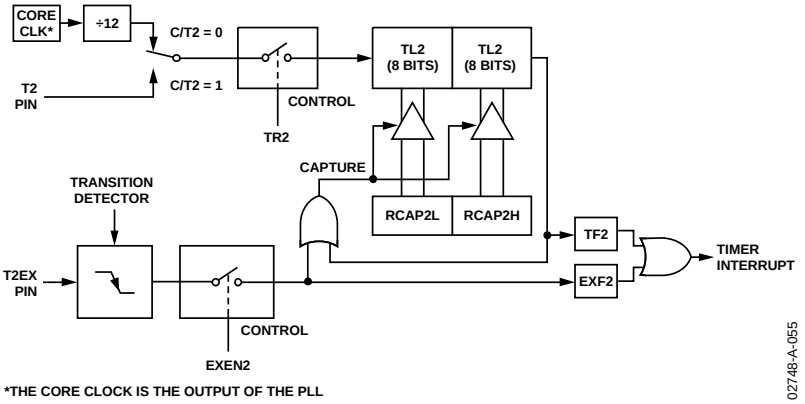



*Figure 49. Timer/Counter 2, 16-Bit Autoreload Mode*

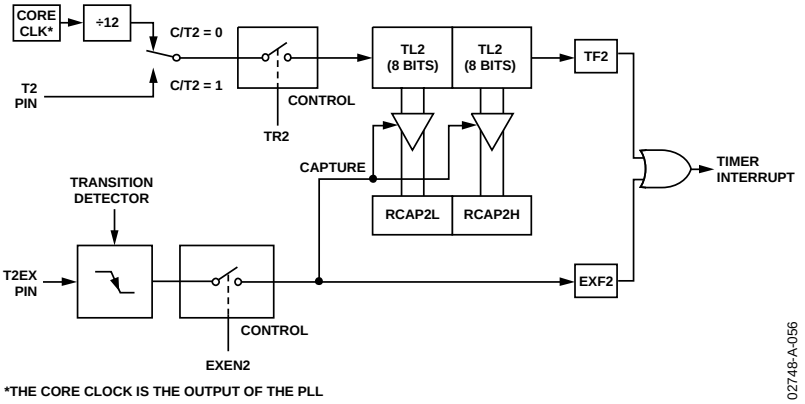



*Figure 50. Timer/Counter 2, 16-Bit Capture Mode*

## UART SERIAL INTERFACE

The serial port is full duplex, meaning it can transmit and receive simultaneously. It is also receive-buffered, meaning it can begin receiving a second byte before a previously received byte has been read from the receive register. However, if the first byte still has not been read by the time reception of the second byte is complete, the first byte is lost. The physical interface to the serial data network is via Pins RxD (P3.0) and TxD (P3.1), while the SFR interface to the UART is comprised of the following registers.

### *SBUF*

The serial port receive and transmit registers are both accessed through the SBUF SFR (SFR address = 99H). Writing to SBUF loads the transmit register and reading SBUF accesses a physically separate receive register.

| **SCON** | **UART Serial Port Control Register** |
|---|---|
| SFR Address | 98H |
| Power-On Default | 00H |
| Bit Addressable | Yes |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| SM0 | SM1 | SM2 | REN | TB8 | RB8 | TI | RI |

**Table 26. SCON SFR Bit Designations**

| Bit No. | Name | Description |
|---|---|---|
| 7 | SM0 | UART Serial Mode Select Bits. |
| 6 | SM1 | These bits select the serial port operating mode as follows:<br>SM0 SM1 Selected Operating Mode<br>0 0 Mode 0: Shift Register, fixed baud rate (Core_Clk/2)<br>0 1 Mode 1: 8-bit UART, variable baud rate<br>1 0 Mode 2: 9-bit UART, fixed baud rate (Core_Clk/64) or (Core_Clk/32)<br>1 1 Mode 3: 9-bit UART, variable baud rate |
| 5 | SM2 | Multiprocessor Communication Enable Bit.<br>Enables multiprocessor communication in Modes 2 and 3.<br>In Mode 0, SM2 should be cleared.<br>In Mode 1, if SM2 is set, RI is not activated if a valid stop bit is not received. If SM2 is cleared, RI is set as soon as the byte of data is received.<br>In Mode 2 or 3, if SM2 is set, RI is not activated if the received ninth data bit in RB8 is 0. If SM2 is cleared, RI is set as soon as the byte of data is received. |
| 4 | REN | Serial Port Receive Enable Bit.<br>Set by the user software to enable serial port reception.<br>Cleared by the user software to disable serial port reception. |
| 3 | TB8 | Serial Port Transmit (Bit 9).<br>The data loaded into TB8 is the ninth data bit that is transmitted in Modes 2 and 3. |
| 2 | RB8 | Serial port Receiver Bit 9.<br>The ninth data bit received in Modes 2 and 3 is latched into RB8. For Mode 1, the stop bit is latched into RB8. |
| 1 | TI | Set by hardware at the end of the eighth bit in Mode 0, or at the beginning of the stop bit in Modes 1, 2, and 3.<br>Cleared by the user software. |
| 0 | RI | Serial Port Receive Interrupt Flag.<br>Set by hardware at the end of the eighth bit in Mode 0, or halfway through the stop bit in Modes 1, 2, and 3.<br>Cleared by user software. |

### Mode 0: 8-Bit Shift Register Mode

Mode 0 is selected by clearing both the SM0 and SM1 bits in the SFR SCON. Serial data enters and exits through RxD. TxD outputs the shift clock. Eight data bits are transmitted or received. Transmission is initiated by any instruction that writes to SBUF. The data is shifted out of the RxD line. The eight bits are transmitted with the least-significant bit (LSB) first, as shown in Figure 52.

Reception is initiated when the receive enable bit (REN) is 1 and the receive interrupt bit (RI) is 0. When RI is cleared the data is clocked into the RxD line and the clock pulses are output from the TxD line.

### Mode 1: 8-Bit UART, Variable Baud Rate

Mode 1 is selected by clearing SM0 and setting SM1. Each data byte (LSB first) is preceded by a start bit (Bit 0) and followed by a stop bit (Bit 1). Therefore 10 bits are transmitted on TxD or received on RxD. The baud rate is set by the Timer 1 or Timer 2 overflow rate, or a combination of the two (one for transmission and the other for reception).

Transmission is initiated by writing to SBUF. The write-to-SBUF signal also loads a 1 (stop bit) into the ninth bit position of the transmit shift register. The data is output bit by bit until the stop bit appears on TxD and the transmit interrupt flag (TI) is automatically set as shown in Figure 51.

Reception is initiated when a 1-to-0 transition is detected on RxD. Assuming a valid start bit was detected, character reception continues. The start bit is skipped and the eight data bits are clocked into the serial port shift register. When all eight bits have been clocked in, the following events occur:

- The eight bits in the receive shift register are latched into SBUF.
- The ninth bit (stop bit) is clocked into RB8 in SCON. The receiver interrupt flag (RI) is set if, and only if, the following conditions are met at the time the final shift pulse is generated:
  - RI = 0
  - Either SM2 = 0 or SM2 = 1 and the received stop bit = 1

  If either of these conditions is not met, the received frame is irretrievably lost, and RI is not set.

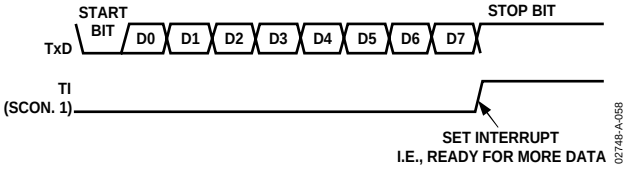



*Figure 51. UART Serial Port Transmission, Mode 1*

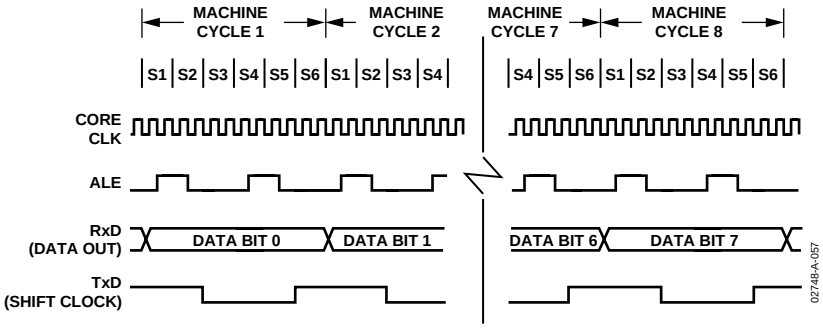



*Figure 52. UART Serial Port Transmission, Mode 0*

### Mode 2: 9-Bit UART with Fixed Baud Rate

Mode 2 is selected by setting SM0 and clearing SM1. In this mode, the UART operates in 9-bit mode with a fixed baud rate. The baud rate is fixed at Core_Clk/64 by default, although by setting the SMOD bit in PCON, the frequency can be doubled to Core_Clk/32. Eleven bits are transmitted or received, a start bit (Bit 0), eight data bits, a programmable ninth bit, and a stop bit (Bit 1). The ninth bit is most often used as a parity bit, although it can be used for anything, including a ninth data bit if required.

To transmit, the eight data bits must be written into SBUF. The ninth bit must be written into TB8 in SCON. When transmission is initiated, the eight data bits (from SBUF) are loaded into the transmit shift register (LSB first). The contents of TB8 are loaded into the ninth bit position of the transmit shift register. The transmission starts at the next valid baud rate clock. The TI flag is set as soon as the stop bit appears on TxD.

Reception for Mode 2 is similar to that of Mode 1. The eight data bytes are input at RxD (LSB first) and loaded into the receive shift register. When all eight bits have been clocked in, the following events occur:

- The eight bits in the receive shift register are latched into SBUF.
- The ninth data bit is latched into RB8 in SCON.
- The receiver interrupt flag (RI) is set if, and only if, the following conditions are met at the time the final shift pulse is generated:
  - RI = 0
  - Either SM2 = 0 or
    SM2 = 1 and the received stop bit = 1

  If either of these conditions is not met, the received frame is irretrievably lost, and RI is not set.

### Mode 3: 9-Bit UART with Variable Baud Rate

Mode 3 is selected by setting both SM0 and SM1. In this mode, the 8051 UART serial port operates in 9-bit mode with a variable baud rate determined by either Timer 1 or Timer 2. The operation of the 9-bit UART is the same as for Mode 2 but the baud rate can be varied as for Mode 1.

In all four modes, transmission is initiated by any instruction that uses SBUF as a destination register. Reception is initiated in Mode 0 by the condition RI = 0 and REN = 1. Reception is initiated in the other modes by the incoming start bit if REN = 1.

### UART Serial Port Baud Rate Generation

In these descriptions that follow, *Core Clock Frequency* refers to the core clock frequency selected via the CD0–CD2 bits in the PLLCON SFR.

**Mode 0 Baud Rate Generation**

The baud rate in Mode 0 is fixed:

$$\textit{Mode 0 Baud Rate} = (\textit{Core Clock Frequency}/12)$$

**Mode 2 Baud Rate Generation**

The baud rate in Mode 2 depends on the value of the SMOD bit in the PCON SFR. If SMOD = 0, the baud rate is 1/64 of the core clock. If SMOD = 1, the baud rate is 1/32 of the core clock:

$$\textit{Mode 2 Baud Rate} = (2^{SMOD}/64) \times (\textit{Core Clock Frequency})$$

**Mode 1 and 3 Baud Rate Generation**

The baud rates in Modes 1 and 3 are determined by the overflow rate in Timer 1 or Timer 2, or both (one for transmit and the other for receive).

**Timer 1 Generated Baud Rates**

When Timer 1 is used as the baud rate generator, the baud rates in Modes 1 and 3 are determined by the Timer 1 overflow rate and the value of SMOD as follows:

$$\textit{Modes 1 and 3 Baud Rate} = (2^{SMOD}/32) \times (\textit{Timer 1 Overflow Rate})$$

The Timer 1 interrupt should be disabled in this application. The timer itself can be configured for either timer or counter operation, and in any of its three running modes. In the most typical application, it is configured for timer operation, in the autoreload mode (high nibble of TMOD = 0100 binary). In that case, the baud rate is given by the formula

$$\textit{Modes 1 and 3 Baud Rate} = (2^{SMOD}/32) \times (\textit{Core Clock}/(12 \times [256 - \textit{TH1}]))$$

A very low baud rate can also be achieved with Timer 1 by leaving the Timer 1 interrupt enabled, and configuring the timer to run as a 16-bit timer (high nibble of TMOD = 0100 binary), and using the Timer 1 interrupt to do a 16-bit software reload. Table 27 shows some commonly used baud rates and how they might be calculated from a core clock frequency of 2.0971 MHz and 16.78 MHz. Generally speaking, a 5% error is tolerable using asynchronous (start/stop) communications.

**Table 27. Commonly Used Baud Rates, Timer 1**

| Ideal Baud | Core CLK | SMOD Value | TH1-Reload Value | Actual Baud | % Error |
|---|---|---|---|---|---|
| 9600 | 16.78 | 1 | –9 (F7H) | 9709 | 1.14 |
| 2400 | 16.78 | 1 | –36 (DCH) | 2427 | 1.14 |
| 1200 | 16.78 | 1 | –73 (B7H) | 1197 | 0.25 |
| 1200 | 2.10 | 1 | –9 (F7H) | 1213 | 1.14 |

### *Timer 2 Generated Baud Rates*

Baud rates can also be generated using Timer 2. Using Timer 2 is similar to using Timer 1 in that the timer must overflow 16 times before a bit is transmitted/received. Because Timer 2 has a 16-bit autoreload mode, a wide range of baud rates is possible using Timer 2.

$$\textit{Modes 1 and 3 Baud Rate} = (1/16) \times (\textit{Timer 2 Overflow Rate})$$

Therefore, when Timer 2 is used to generate baud rates, the timer increments every two clock cycles and not every core machine cycle as before. Therefore, it increments six times faster than Timer 1, and therefore baud rates six times faster are possible. Because Timer 2 has 16-bit autoreload capability, very low baud rates are still possible. Timer 2 is selected as the baud rate generator by setting the CLK and/or RCLK in T2CON. The baud rates for transmit and receive can be simultaneously different. Setting RCLK and/or TCLK puts Timer 2 into its baud rate generator mode as shown in Figure 53.

In this case, the baud rate is given by the formula

$$\textit{Modes 1 and 3 Baud Rate} = (\textit{Core Clk})/(32 \times [65536 - (\textit{RCAP2H}, \textit{RCAP2L})])$$

Table 28 shows some commonly used baud rates and how they could be calculated from a core clock frequency of 2.0971 MHz and 16.7772 MHz.

**Table 28. Commonly Used Baud Rates, Timer 2**

| Ideal Baud | Core CLK | RCAP2H Value | RCAP2L Value | Actual Baud | % Error |
|---|---|---|---|---|---|
| 19200 | 16.78 | –1 (FFH) | –27 (E5H) | 19418 | 1.14 |
| 9600 | 16.78 | –1 (FFH) | –55 (C9H) | 9532 | 0.7 |
| 2400 | 16.78 | –1 (FFH) | –218 (26H) | 2405 | 0.21 |
| 1200 | 16.78 | –2 (FEH) | –181 (4BH) | 1199 | 0.02 |
| 9600 | 2.10 | –1 (FFH) | –7 (FBH) | 9362 | 2.4 |
| 2400 | 2.10 | –1 (FFH) | –27 (ECH) | 2427 | 1.14 |
| 1200 | 2.10 | –1 (FFH) | –55 (D7H) | 1191 | 0.7 |

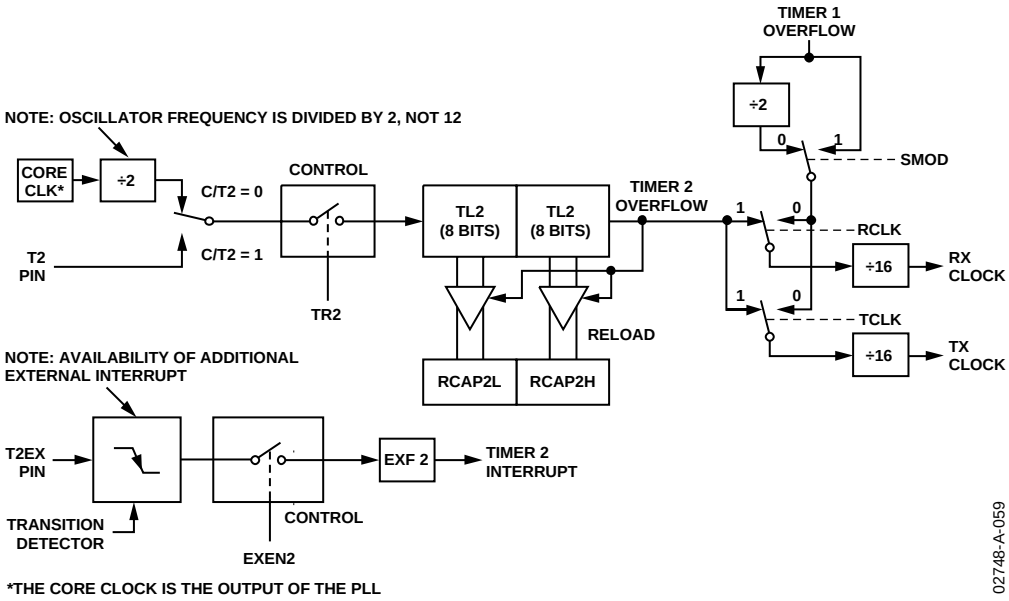



*Figure 53. Timer 2, UART Baud Rates*

## INTERRUPT SYSTEM

The ADuC814 provides a total of twelve interrupt sources with two priority levels. The control and configuration of the interrupt system is carried out through three interrupt-related SFRs.

| | |
|---|---|
| IE | Interrupt Enable Register |
| IP | Interrupt Priority Register |
| IEIP2 | Secondary Interrupt Enable and Priority Register |

| **IE** | **Interrupt Enable Register** |
|---|---|
| SFR Address | A8H |
| Power-On Default | 00H |
| Bit Addressable | Yes |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| EA | EADC | ET2 | ES | ET1 | EX1 | ET0 | EX0 |

**Table 29. IE SFR Bit Designations**

| Bit No. | Name | Description |
|---|---|---|
| 7 | EA | Global Interrupt Enable.<br>Set by the user to enable all interrupt sources.<br>Cleared by the user to disable all interrupt sources. |
| 6 | EADC | ADC Interrupt.<br>Set by the user to enable the ADC interrupt.<br>Cleared by the user to disable the ADC interrupt. |
| 5 | ET2 | Timer 2 Interrupt.<br>Set by the user to enable the Timer 2 interrupt.<br>Cleared by the user to disable the Timer 2 interrupt. |
| 4 | ES | UART Serial Port Interrupt.<br>Set by the user to enable the UART serial port interrupt.<br>Cleared by the user to disable the UART serial port interrupt. |
| 3 | ET1 | Timer 1 Interrupt.<br>Set by the user to enable the Timer 1 interrupt.<br>Cleared by the user to disable the Timer 1 interrupt. |
| 2 | EX1 | INT1 Interrupt.<br>Set by the user to enable the External Interrupt 1.<br>Cleared by the user to disable the External Interrupt 1. |
| 1 | ET0 | Timer 0 Interrupt.<br>Set by the user to enable the Timer 0 interrupt.<br>Cleared by the user to disable the Timer 0 interrupt. |
| 0 | EX0 | INT0 Interrupt.<br>Set by the user to enable the External Interrupt 0.<br>Cleared by the user to disable the External Interrupt 0. |

**IP** **Interrupt Priority Register**
SFR Address B8H
Power-On Default 00H
Bit Addressable Yes

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| --- | PADC | PT2 | PS | PT1 | PX1 | PT0 | PX0 |

**Table 30. IP SFR Bit Designations**

| Bit No. | Name | Description |
|---|---|---|
| 7 | --- | Reserved. |
| 6 | PADC | ADC Interrupt Priority.<br>Written to by user to set interrupt priority level (1 = High; 0 = Low). |
| 5 | PT2 | Timer 2 Interrupt Priority.<br>Written to by the user to set interrupt priority level (1 = High; 0 = Low). |
| 4 | PS | UART Serial Port Interrupt Priority.<br>Written to by the user to set interrupt priority level (1 = High; 0 = Low). |
| 3 | PT1 | Timer 1 Interrupt Priority.<br>Written to by the user to set interrupt priority level (1 = High; 0 = Low). |
| 2 | PX1 | External Interrupt 1 Priority (INT1).<br>Written to by the user to set interrupt priority level (1 = High; 0 = Low). |
| 1 | PT0 | Timer 0 Interrupt Priority.<br>Written to by the user to set interrupt priority level (1 = High; 0 = Low). |
| 0 | PX0 | External Interrupt 0 Priority (INT0).<br>Written to by the user to set interrupt priority level (1 = High; 0 = Low). |

**IEIP2** **Secondary Interrupt Enable and Priority Register**
SFR Address A9H
Power-On Default A0H
Bit Addressable No

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| --- | PT1 | PPSM | PSI | --- | ETI | EPSM | ESI |

**Table 31. IEIP2 SFR Bit Designations**

| Bit No. | Name | Description |
|---|---|---|
| 7 | --- | Reserved. |
| 6 | PTI | Time Interval Counter Interrupt Priority.<br>Written to by the user to set TIC interrupt priority (1 = High; 0 = Low). |
| 5 | PPSM | PSM Interrupt Priority.<br>Written to by the user to select power supply monitor interrupt priority (1 = High; 0 = Low). |
| 4 | PSI | SPI Serial Port Interrupt Priority.<br>Written to by the user to select SPI serial port interrupt priority (1 = High; 0 = Low). |
| 3 | --- | Reserved. This bit must be 0. |
| 2 | ETI | TIC Interrupt.<br>Set by the user to enable the TIC interrupt.<br>Cleared by the user to disable the TIC interrupt. |
| 1 | EPSM | Power Supply Monitor Interrupt.<br>Set by the user to enable the power supply monitor interrupt.<br>Cleared by the user to disable the power supply monitor interrupt. |
| 0 | ESI | SPI Serial Port Interrupt.<br>Set by the user to enable the SPI serial port interrupt.<br>Cleared by the user to disable the SPI serial port interrupt. |

### *Interrupt Priority*

The interrupt enable registers are written by the user to enable individual interrupt sources, while the interrupt priority registers allow the user to select one of two priority levels for each interrupt. An interrupt of a high priority may interrupt the service routine of a low priority interrupt, and if two interrupts of different priority occur at the same time, the higher level interrupt is serviced first. An interrupt cannot be interrupted by another interrupt of the same priority level. If two interrupts of the same priority level occur simultaneously, a polling sequence is observed as shown in Table 32.

**Table 32. Priority within an Interrupt Level**

| Source | Priority | Description |
|---|---|---|
| PSMI | 1 (Highest) | Power Supply Monitor Interrupt |
| WDS | 2 | Watchdog Interrupt |
| IE0 | 3 | External Interrupt 0 |
| RDY0/RDY1 | 4 | ADC Interrupt |
| TF0 | 5 | Timer/Counter 0 Interrupt |
| IE1 | 6 | External Interrupt 1 |
| TF1 | 7 | Timer/Counter 1 Interrupt |
| ISPI | 8 | SPI Interrupt |
| RI + TI | 9 | Serial Interrupt |
| TF2 + EXF2 | 10 | Timer/Counter 2 Interrupt |
| TII | 11 (Lowest) | Time Interval Counter Interrupt |

### *Interrupt Vectors*

When an interrupt occurs, the program counter is pushed onto the stack, and the corresponding interrupt vector address is loaded into the program counter. The interrupt vector addresses are shown in Table 33.

**Table 33. Interrupt Vector Addresses**

| Source | Vector Address |
|---|---|
| IE0 | 0003H |
| TF0 | 000BH |
| IE1 | 0013H |
| TF1 | 001BH |
| RI + TI | 0023H |
| TF2 + EXF2 | 002BH |
| RDY0/RDY1 (ADC) | 0033H |
| ISPI | 003BH |
| PSMI | 0043H |
| TII | 0053H |
| WDS (WDIR = 1)[1] | 005BH |

[1] The watchdog can be configured to generate an interrupt instead of a reset when it times out. This is used for logging errors or to examine the internal status of the microcontroller core to understand, from a software debug point of view, why a watchdog timeout occurred. The watchdog interrupt is slightly different from normal interrupts in that its priority level is always set to 1, and it is not possible to disable the interrupt via the global disable bit (EA) in the IE SFR. This is done to ensure that the interrupt is always responded to if a watchdog timeout occurs. The watchdog produces an interrupt only if the watchdog timeout is greater than zero.

# ADuC814 HARDWARE DESIGN CONSIDERATIONS

This section outlines some key hardware design considerations for integrating the ADuC814 into any hardware system.

## CLOCK OSCILLATOR

As described earlier, the core clock frequency for the ADuC814 is generated from an on-chip PLL that locks onto a multiple (512 times) of 32.768 kHz. The latter is generated from an internal clock oscillator. To use the internal clock oscillator, connect a 32.768 kHz parallel resonant crystal between XTAL1 and XTAL2 pins (Pins 26 and 27) as shown in Figure 54.

As shown in the typical external crystal connection diagram in Figure 54, two internal 12 pF capacitors are provided on-chip. These are connected internally, directly to the XTAL1 and XTAL2 pins. The total input capacitances at both pins is detailed in the Specifications table. The value of the total load capacitance required for the external crystal should be the value recommended by the crystal manufacturer for use with that specific crystal. In many cases, because of the on-chip capacitors, additional external load capacitors are not required.

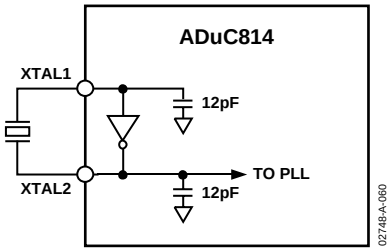



*Figure 54. External Parallel Resonant Crystal Connections*

As an alternative to providing two separate power supplies, $AV_{DD}$ can be kept quiet by placing a small series resistor and/or ferrite bead between it and $DV_{DD}$, and then decoupling $AV_{DD}$ separately to ground. An example of this configuration is shown in Figure 56. With this configuration, other analog circuitry (such as op amps and voltage reference) can be powered from the $AV_{DD}$ supply line as well.

## POWER SUPPLIES

The ADuC814's operational power supply voltage range is 2.7 V to 5.5 V. Although the guaranteed data sheet specifications are given only for power supplies within 2.7 V to 3.3 V or 4.5 V to 5.5 V, (±10% of the nominal level), the chip can function equally well at any power supply level between 2.7 V and 5.5 V.

Users should separate analog and digital power supply pins ($AV_{DD}$ and $DV_{DD}$) and allow $AV_{DD}$ to be kept relatively free of noisy digital signals often present on the system $DV_{DD}$ line. In this mode, the part can also operate with split supplies as long as the supply voltages are within 0.3 V of each other. A typical split-supply configuration is show in Figure 55.

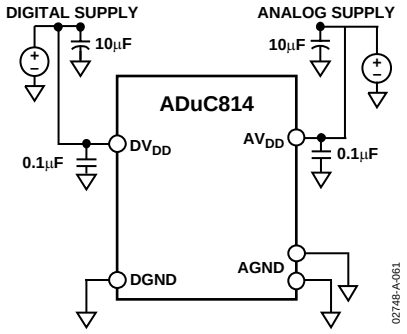



*Figure 55. External Dual-Supply Connections*

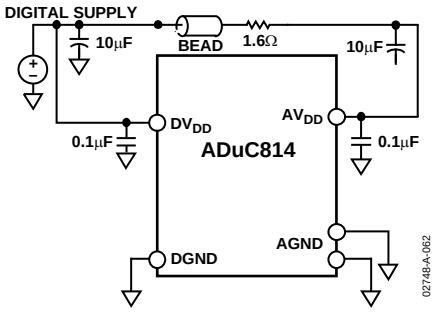



*Figure 56. External Single-Supply Connections*

Notice that in both Figure 55 and Figure 56, a large value (10 μF) reservoir capacitor sits on $DV_{DD}$ and a separate 10 μF capacitor sits on AVDD. Also, local small-value (0.1 μF) capacitors are located at each $V_{DD}$ pin of the chip. As per standard design practice, be sure to include all of these capacitors, and ensure that the smaller capacitors are closest 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, note that, at all times, the analog and digital ground pins on the ADuC814 should be referenced to the same system ground reference point.

## POWER CONSUMPTION

The CORE values given represent the current drawn by $DV_{DD}$, while the rest (ADC and DAC) are pulled by the $AV_{DD}$ pin and can be disabled in software when not in use. The other on-chip peripherals (such as watchdog timer and power supply monitor) consume negligible current and are therefore lumped in with the CORE operating current here. Of course, the user must add any currents sourced by the parallel and serial I/O pins, and that sourced by the DAC, in order to determine the total current needed at the ADuC814's supply pins. Also, current drawn from the DVDD supply increases by approximately 5 mA during the Flash/EE erase and program cycles.

### *Power-Saving Modes*

Setting the idle and power-down mode bits, PCON.0 and PCON.1, respectively, in the PCON SFR described in Table 5, allows the chip to be switched from normal mode to idle mode, and also to full power-down mode.

In idle mode, the oscillator continues to run, but the core clock generated from the PLL is halted. The on-chip peripherals continue to receive the clock and remain functional. The CPU status is preserved with the stack pointer, program counter, and all other internal registers maintaining their data during idle mode. Port pins and DAC output pins also retain their states in this mode. The chip recovers from idle mode upon receiving any enabled interrupt, or upon receiving a hardware reset.

In power-down mode, both the PLL and the clock to the core are stopped. The on-chip oscillator can be halted or can continue to oscillate depending on the state of the oscillator power-down bit (OSC_PD) in the PLLCON SFR. The TIC, being driven directly from the oscillator, can also be enabled during power-down. All other on-chip peripherals however, are shut down. Port pins retain their logic levels in this mode, but the DAC output goes to a high impedance state (three-state). During full power-down mode, the ADuC814 consumes a total of 5 µA typically. There are five ways of terminating power-down mode, as described in the next sections.

#### Asserting RESET (Pin 10)

Returns to normal mode and all registers are set to their default state. Program execution starts at the reset vector once the RESET pin is de-asserted.

#### Cycling Power

All registers are set to their default state and program execution starts at the reset vector.

#### Time Interval Counter (TIC) Interrupt

Power-down mode is terminated and the CPU services the TIC interrupt. The RETI at the end of the TIC interrupt service routine returns the core to the instruction following the one that enabled power-down.

#### SPI Interrupt

Power-down mode is terminated and the CPU services the SPI interrupt. The RETI at the end of the ISR returns the core to the instruction following the one that enabled power-down. Note that the SPI power-down interrupt enable bit (SERIPD) in the PCON SFR must first be set to allow this mode of operation.

#### $\overline{\text{INT0}}$ Interrupt

Power-down mode is terminated and the CPU services the INT0 interrupt. The RETI at the end of the ISR returns the core to the instruction following the one that enabled power-down. The INT0 pin must not be driven low during or within two machine cycles of the instruction that initiates power-down mode. Note that the $\overline{\text{INT0}}$ power-down interrupt enable bit (INT0PD) in the PCON SFR must first be set to allow this mode of operation.

### *Power-On Reset*

An internal POR (power-on-reset) is implemented on the ADuC814. For $DV_{DD}$ below 2.45 V, the internal POR holds the ADuC814 in reset. As $DV_{DD}$ rises above 2.45 V, an internal timer times out for approximately128 ms before the part is released from reset. The user must ensure that the power supply has reached a stable 2.7 V minimum level by this time. Likewise on power-down, the internal POR holds the ADuC814 in reset until the power supply has dropped below 1 V. Figure 57 illustrates the operation of the internal POR in detail.

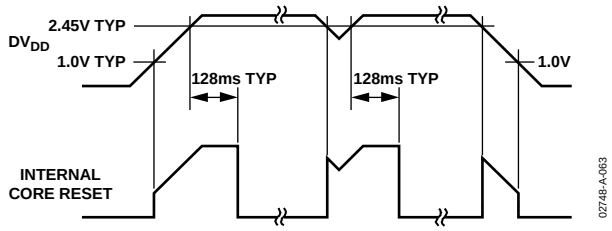



*Figure 57. Internal POR Operation*

### *Grounding and Board Layout Recommendations*

As with all high resolution data converters, special attention must be paid to grounding and PC board layout of ADuC814 based designs in order to achieve optimum performance from the ADCs and DAC. Although the ADuC814 has 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 together very close to the ADuC814, as illustrated in the simplified example of Figure 58a. In systems where digital and analog ground planes are connected together somewhere else (at the system's power supply for example), they cannot be connected again near the ADuC814 because a ground loop would result. In these cases, tie all of the ADuC814's AGND and DGND pins to the analog ground plane, as illustrated in Figure 58b. In systems with only one ground plane, ensure that the digital and analog components are physically separated onto separate halves of the board such that digital return currents do not flow near analog circuitry and vice versa. The ADuC814 can then be placed between the digital and analog sections, as illustrated in Figure 58c.

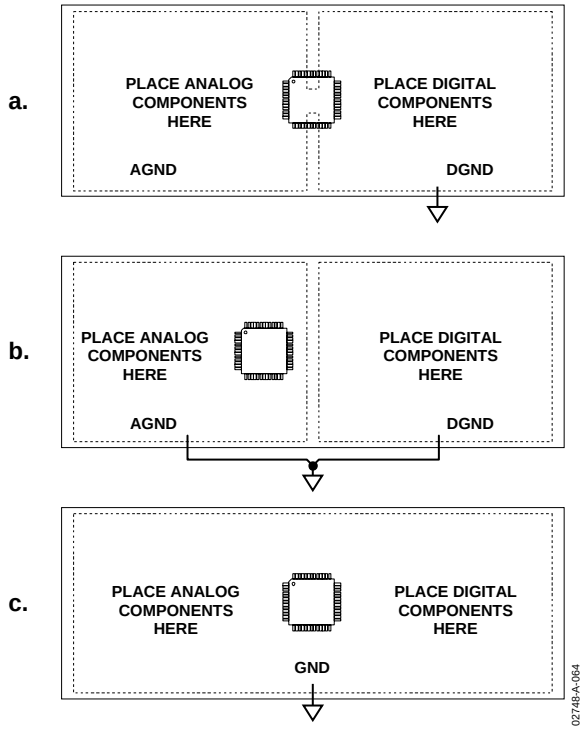



*Figure 58. System Grounding Schemes*

In all of these scenarios, and in more complicated real-life applications, keep in mind 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 put power components on the analog side of Figure 58b with $DV_{DD}$ because that would force return currents from $DV_{DD}$ to flow through AGND. Also, try to avoid digital currents flowing under analog circuitry, which could happen if a noisy digital chip is placed on the left half of the board in Figure 58c. Whenever possible, avoid large discontinuities in the ground plane(s) (such as are formed by a long trace on the same layer), because they force return signals to travel a longer path. And of course, make all connections to the ground plane directly, with little or no trace separating the pin from its via to ground.

If the user plans to connect fast logic signals (rise/fall time < 5 ns) to any of the ADuC814's digital inputs, add a series resistor to each relevant line to keep rise and fall times longer than 5 ns at the ADuC814 input pins. A value of 100 Ω or 200 Ω is usually sufficient to prevent high speed signals from coupling capacitively into the ADuC814 and affecting the accuracy of ADC conversions.

## OTHER HARDWARE CONSIDERATIONS

To facilitate in-circuit programming, in-circuit debug, and emulation options, users should implement some simple connection points in their hardware. A typical ADuC814 connection diagram is shown in Figure 59.

### *In-Circuit Serial Download Access*

Nearly all ADuC814 designs will want to take advantage of the in-circuit reprogrammability of the chip. This is accomplished by a connection from the ADuC814's UART to a PC, which requires an external RS-232 chip for level translation. If users would rather not design an RS-232 chip onto a board, refer to the Application Note uC006, *A 4-Wire UART-to-PC Interface* (available at www.analog.com/microconverter) for a simple (and zero-cost-per-board) method of gaining in-circuit serial download access to the ADuC814.

In addition to the basic UART connections, users also need a way to trigger the chip into download mode. This is accomplished via a 1 kΩ pull-up resistor that can be jumpered onto the DLOAD pin. To get the ADuC814 into download mode, simply connect this jumper and power-cycle the device (or manually reset the device, if a manual reset button is available), and it will be ready to receive a new program serially. To enable the device to enter normal mode (and run the program) whenever power is cycled or RESET is toggled, the DLOAD pin must be pulled low through a 1 kΩ resistor.

### *Embedded Serial Port Debugger*

From a hardware perspective, entry to serial port debug mode is identical to the serial download entry sequence described in the preceding section. In fact, both serial download and serial port debug modes can be thought of as essentially one mode of operation used in two different ways. Note that the serial port debugger is fully contained on the ADuC814 device, unlike ROM monitor type debuggers.

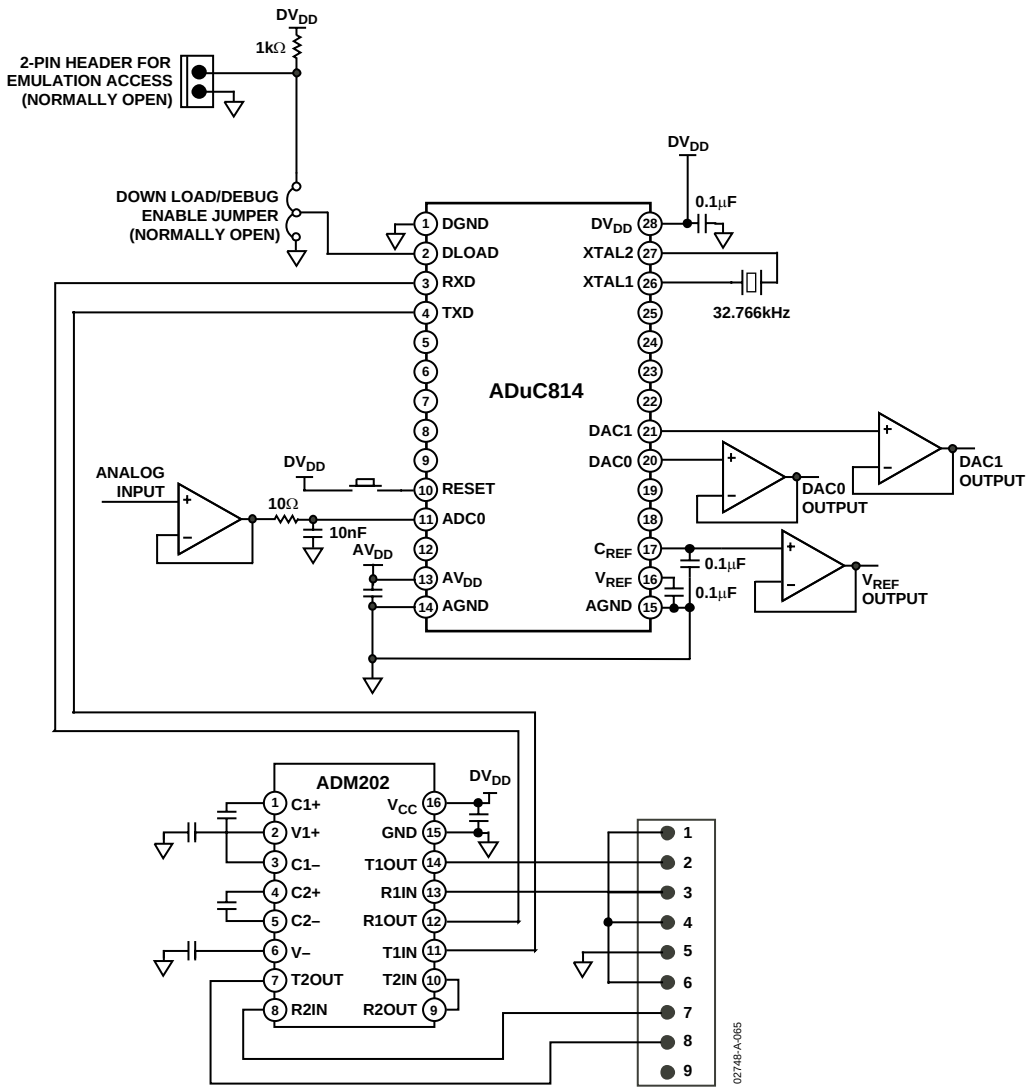



*Figure 59. Typical ADuC814 System Connection Diagram*

### *Single-Pin Emulation Mode*

Also built into the ADuC814 is a dedicated controller for single-pin in-circuit emulation (ICE) using standard production ADuC814 devices. In this mode, emulation access is gained by connection to a single pin, again the DLOAD pin is used for this function. As described previously, this pin is either high to enable entry into serial download and serial debug modes or low to select normal code execution. To enable single-pin emulation mode, however, users need to pull the DLOAD pin high through a 1 kΩ resistor. The emulator then connects to the 2-pin header. To be compatible with the standard connector that comes with the single-pin emulator available from Accutron Limited (www.accutron.com), use a 2-pin 0.1-inch pitch friction lock header from Molex (www.molex.com) such as their part number 22-27-2021. Be sure to observe the polarity of this header. When the friction lock tab is at the right, the ground pin should be the lower of the two pins (when viewed from the top).

# TIMING SPECIFICATIONS[1,2,3]

**Table 34. Clock Input (External Clock Driven XTAL1)**

**$AV_{DD}$ = 2.7 V to 3.3 V or 4.75 V to 5.25 V, $DV_{DD}$ = 2.7 V to 3.3 V or 4.75 V to 5.25 V; all specifications $T_{MIN}$ to $T_{MAX}$, unless otherwise noted**

| Parameter | | 32.768 kHz External Crystal | | | Unit |
|---|---|---|---|---|---|
| | | Min | Typ | Max | |
| $t_{CK}$ | XTAL1 Period | | 30.52 | | µs |
| $t_{CKL}$ | XTAL1 Width Low | | 15.16 | | µs |
| $t_{CKH}$ | XTAL1 Width High | | 15.16 | | µs |
| $t_{CKR}$ | XTAL1 Rise Time | | 20 | | ns |
| $t_{CKF}$ | XTAL1 Fall Time | | 20 | | ns |
| $1/t_{CORE}$ | ADuC814 Core Clock Frequency[4] | 0.131 | | 16.78 | MHz |
| $t_{CORE}$ | ADuC814 Core Clock Period[5] | | 0.476 | | µs |
| $t_{CYC}$ | ADuC814 Machine Cycle Time[6] | 0.72 | 5.7 | 91.55 | µs |

[1] AC inputs during testing are driven at $DV_{DD}$ – 0.5 V for a Logic 1, and at 0.45 V for a Logic 0. Timing measurements are made at $V_{IH}$ min for a Logic 1, and at $V_{IL}$ max for a Logic 0 as shown in Figure 61.
[2] For timing purposes, a port pin is no longer floating when a 100 mV change from load voltage occurs. A port pin begins to float when a 100 mV change from the loaded $V_{OH}/V_{OL}$ level occurs as shown in Figure 61.
[3] $C_{LOAD}$ for all outputs = 80 pF, unless otherwise noted.
[4] ADuC814 internal PLL locks onto a multiple (512 times) the external crystal frequency of 32.768 kHz to provide a stable 16.777216 MHz internal clock for the system. The core can operate at this frequency or at a binary submultiple called Core_Clk, selected via the PLLCON SFR.
[5] This number is measured at the default Core_Clk operating frequency of 2.09 MHz.
[6] ADuC814 Machine Cycle Time is nominally defined as 12/Core_CLK.

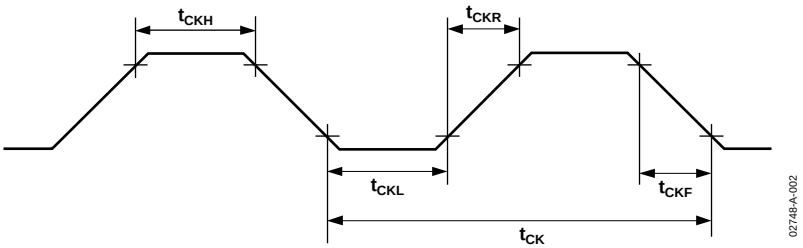



*Figure 60. XTAL1 Input*

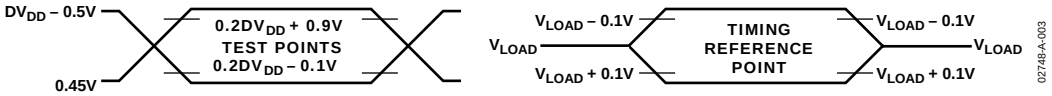



*Figure 61. Timing Waveform Characteristics*

**Table 35. UART Timing (Shift Register Mode)**

| Parameter | | 16.78 MHz Core_Clk | | | Variable Core_Clk | | | Unit |
|---|---|---|---|---|---|---|---|---|
| | | Min | Typ | Max | Min | Typ | Max | |
| $t_{XLXL}$ | Serial Port Clock Cycle Time | | 715 | | | 12 $t_{CORE}$ | | µs |
| $t_{QVXH}$ | Output Data Setup to Clock | 463 | | | 10 $t_{CORE}$ | −133 | | ns |
| $t_{DVXH}$ | Input Data Setup to Clock | 252 | | | 2 $t_{CORE}$ | +133 | | ns |
| $t_{XHDX}$ | Input Data Hold after Clock | 0 | | | 0 | | | ns |
| $t_{XHQX}$ | Output Data Hold after Clock | 22 | | | 2 $t_{CORE}$ | −117 | | ns |

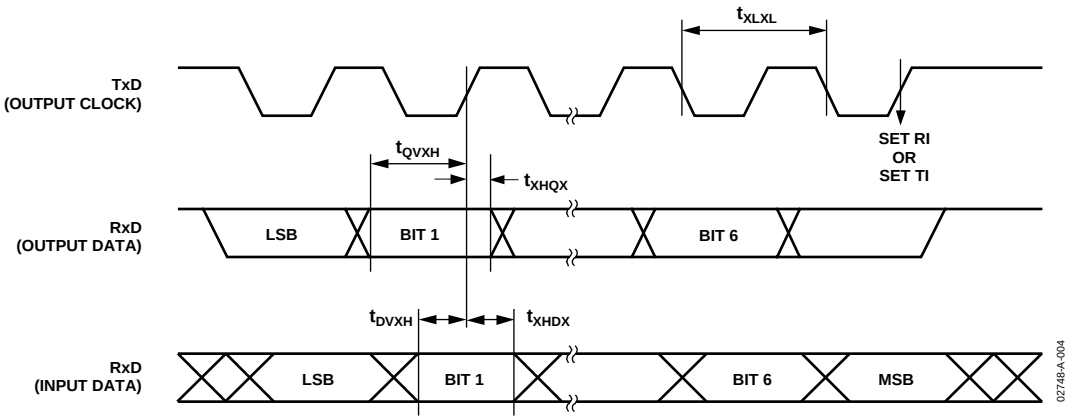



*Figure 62. UART Timing in Shift Register Mode*

**Table 36. SPI Master Mode Timing (CPHA = 1)**

| Parameter | | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| $t_{SL}$ | SCLOCK Low Pulse Width[1] | | 630 | | ns |
| $t_{SH}$ | SCLOCK High Pulse Width[1] | | 630 | | ns |
| $t_{DAV}$ | Data Output Valid after SCLOCK Edge | | | 50 | ns |
| $t_{DSU}$ | Data Input Setup Time before SCLOCK Edge | 100 | | | ns |
| $t_{DHD}$ | Data Input Hold Time after SCLOCK Edge | 100 | | | ns |
| $t_{DF}$ | Data Output Fall Time | | 10 | 25 | ns |
| $t_{DR}$ | Data Output Rise Time | | 10 | 25 | ns |
| $t_{SR}$ | SCLOCK Rise Time | | 10 | 25 | ns |
| $t_{SF}$ | SCLOCK Fall Time | | 10 | 25 | ns |

[1] Characterized under the following conditions:
a. Core clock divider bits CD2, CD1, and CD0 in PLLCON SFR set to 0, 1, and 1, respectively, i.e., core clock frequency = 2.09 MHz.
b. SPI bit-rate selection bits SPR1 and SPR0 bits in SPICON SFR set to 0 and 0, respectively.

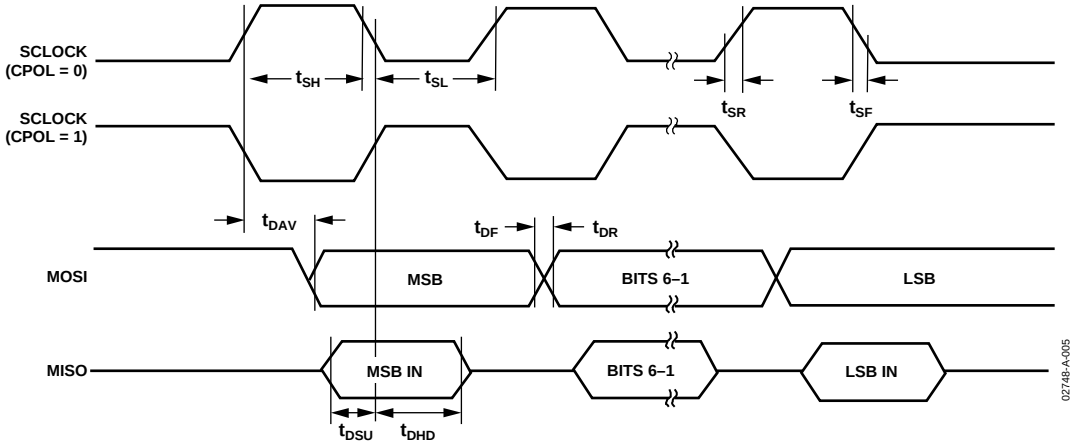



*Figure 63. SPI Master Mode Timing (CPHA = 1)*

**Table 37. SPI Master Mode Timing (CPHA = 0)**

| Parameter | | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| $t_{SL}$ | SCLOCK Low Pulse Width[1] | | 630 | | ns |
| $t_{SH}$ | SCLOCK High Pulse Width[1] | | 630 | | ns |
| $t_{DAV}$ | Data Output Valid after SCLOCK Edge | | | 50 | ns |
| $t_{DOSU}$ | Data Output Setup before SCLOCK Edge | | | 150 | ns |
| $t_{DSU}$ | Data Input Setup Time before SCLOCK Edge | 100 | | | ns |
| $t_{DHD}$ | Data Input Hold Time after SCLOCK Edge | 100 | | | ns |
| $t_{DF}$ | Data Output Fall Time | | 10 | 25 | ns |
| $t_{DR}$ | Data Output Rise Time | | 10 | 25 | ns |
| $t_{SR}$ | SCLOCK Rise Time | | 10 | 25 | ns |
| $t_{SF}$ | SCLOCK Fall Time | | 10 | 25 | ns |

[1] Characterized under the following conditions:
a. Core clock divider bits CD2, CD1, and CD0 bits in PLLCON SFR set to 0, 1, and 1, respectively, i.e., core clock frequency = 2.09 MHz.
b. SPI bit-rate selection bits SPR1 and SPR0 bits in SPICON SFR set to 0 and 0, respectively.

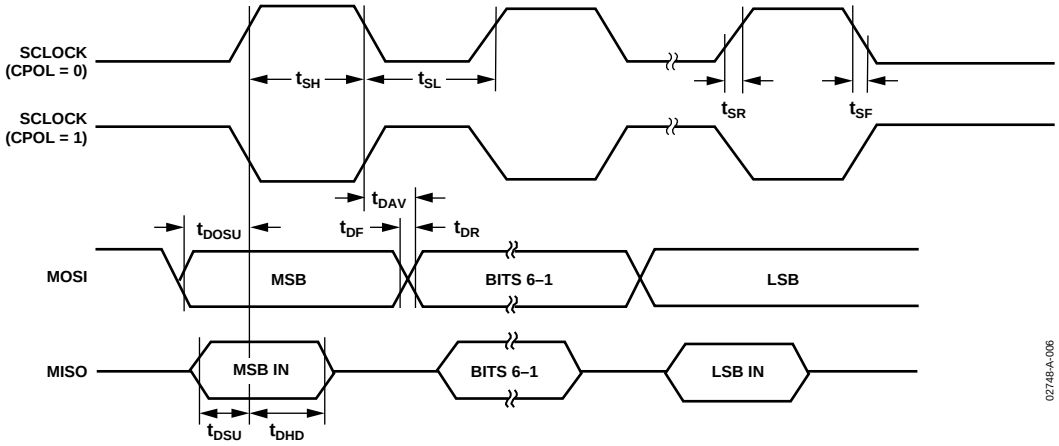



*Figure 64. SPI Master Mode Timing (CPHA = 0)*

**Table 38. SPI Slave Mode Timing (CPHA = 1)**

| Parameter | | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| $t_{SS}$ | $\overline{SS}$ to SCLOCK Edge | 0 | | | ns |
| $t_{SL}$ | SCLOCK Low Pulse Width | | 330 | | ns |
| $t_{SH}$ | SCLOCK High Pulse Width | | 330 | | ns |
| $t_{DAV}$ | Data Output Valid after SCLOCK Edge | | | 50 | ns |
| $t_{DSU}$ | Data Input Setup Time before SCLOCK Edge | 100 | | | ns |
| $t_{DHD}$ | Data Input Hold Time after SCLOCK Edge | 100 | | | ns |
| $t_{DF}$ | Data Output Fall Time | | 10 | 25 | ns |
| $t_{DR}$ | Data Output Rise Time | | 10 | 25 | ns |
| $t_{SR}$ | SCLOCK Rise Time | | 10 | 25 | ns |
| $t_{SF}$ | SCLOCK Fall Time | | 10 | 25 | ns |
| $t_{SFS}$ | $\overline{SS}$ High after SCLOCK Edge | 0 | | | ns |

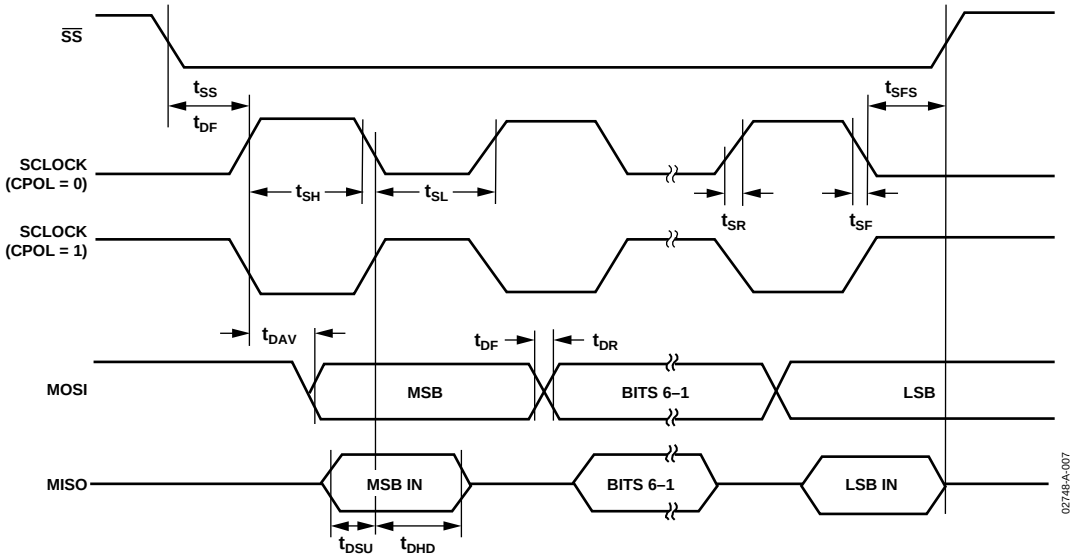



*Figure 65. SPI Slave Mode Timing (CPHA = 1)*

**Table 39. SPI Slave Mode Timing (CPHA = 0)**

| Parameter | | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| $t_{SS}$ | $\overline{SS}$ to SCLOCK Edge | 0 | | | |
| $t_{SL}$ | SCLOCK Low Pulse Width | | 330 | | ns |
| $t_{SH}$ | SCLOCK High Pulse Width | | 330 | | ns |
| $t_{DAV}$ | Data Output Valid after SCLOCK Edge | | | 50 | ns |
| $t_{DSU}$ | Data Input Setup Time before SCLOCK Edge | | 100 | | ns |
| $t_{DHD}$ | Data Input Hold Time after SCLOCK Edge | 100 | | | ns |
| $t_{DF}$ | Data Output Fall Time | | 10 | 25 | ns |
| $t_{DR}$ | Data Output Rise Time | | 10 | 25 | ns |
| $t_{SR}$ | SCLOCK Rise Time | | 10 | 25 | ns |
| $t_{SF}$ | SCLOCK Fall Time | | 10 | 25 | ns |
| $t_{SSR}$ | $\overline{SS}$ to SCLOCK Edge | | | 50 | ns |
| $t_{DOSS}$ | Data Output Valid after $\overline{SS}$ Edge | | | 20 | ns |
| $t_{SFS}$ | $\overline{SS}$ High after SCLOCK Edge | 0 | | | ns |

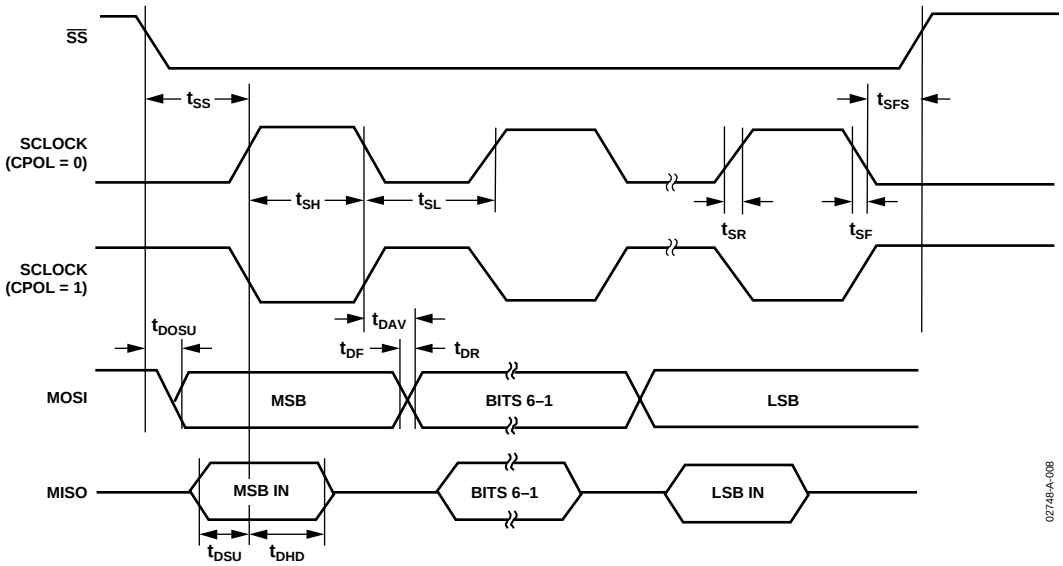



*Figure 66. SPI Slave Mode Timing (CPHA = 0)*

# OUTLINE DIMENSIONS

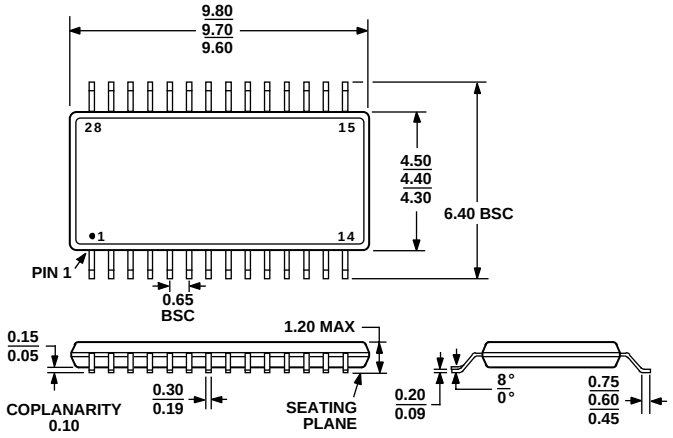



*Figure 67. 28-Lead Thin Shrink Small Outline Package (TSSOP) (RU-28)*
*Dimensions shown in mm*

## ORDERING GUIDE

| Model | Temperature Range | Package Description | Package Option |
|---|---|---|---|
| ADuC814ARU | −40°C to +125°C | Thin Shrink Small Outline (TSSOP) | RU-28 |
| ADuC814ARU-REEL | −40°C to +125°C | Thin Shrink Small Outline (TSSOP) | RU-28 |
| ADuC814ARU-REEL7 | −40°C to +125°C | Thin Shrink Small Outline (TSSOP) | RU-28 |
| ADuC814BRU | −40°C to +125°C | Thin Shrink Small Outline (TSSOP) | RU-28 |
| ADuC814BRU-REEL | −40°C to +125°C | Thin Shrink Small Outline (TSSOP) | RU-28 |
| ADuC814BRU-REEL7 | −40°C to +125°C | Thin Shrink Small Outline (TSSOP) | RU-28 |

| QuickStart Development System Model | Description |
|---|---|
| EVAL-ADUC814QS | Development System for the ADuC814 MicroConverter |
| EVAL-ADUC814QSP[1] | QuickStart PLUS Development System |

[1]Only available to order through the web.

# NOTES

**Purchase of licensed I²C components of Analog Devices or one of its sublicensed Associated Companies conveys a license for the purchaser under the Philips I²C Patent Rights to use these components in an I²C system, provided that the system conforms to the I²C Standard Specification as defined by Philips.**

**© 2003 Analog Devices, Inc. All rights reserved. Trademarks and registered trademarks are the property of their respective owners.**
**C02748-0-12/03(A)**

ANALOG DEVICES

www.analog.com

Компания «ЭлектроПласт» предлагает заключение долгосрочных отношений при поставках импортных электронных компонентов на взаимовыгодных условиях!

Наши преимущества:

- Оперативные поставки широкого спектра электронных компонентов отечественного и импортного производства напрямую от производителей и с крупнейших мировых складов;
- Поставка более 17-ти миллионов наименований электронных компонентов;
- Поставка сложных, дефицитных, либо снятых с производства позиций;
- Оперативные сроки поставки под заказ (от 5 рабочих дней);
- Экспресс доставка в любую точку России;
- Техническая поддержка проекта, помощь в подборе аналогов, поставка прототипов;
- Система менеджмента качества сертифицирована по Международному стандарту ISO 9001;
- Лицензия ФСБ на осуществление работ с использованием сведений, составляющих государственную тайну;
- Поставка специализированных компонентов (Xilinx, Altera, Analog Devices, Intersil, Interpoint, Microsemi, Aeroflex, Peregrine, Syfer, Eurofarad, Texas Instrument, Miteq, Cobham, E2V, MA-COM, Hittite, Mini-Circuits,General Dynamics и др.);

Помимо этого, одним из направлений компании «ЭлектроПласт» является направление «Источники питания». Мы предлагаем Вам помощь Конструкторского отдела:

- Подбор оптимального решения, техническое обоснование при выборе компонента;
- Подбор аналогов;
- Консультации по применению компонента;
- Поставка образцов и прототипов;
- Техническая поддержка проекта;
- Защита от снятия компонента с производства.

**Как с нами связаться**

**Телефон:** 8 (812) 309 58 32 (многоканальный)
**Факс:** 8 (812) 320-02-42
**Электронная почта:** org@eplast1.ru
**Адрес:** 198099, г. Санкт-Петербург, ул. Калинина, дом 2, корпус 4, литера А.