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

Preliminary Technical Data — ADuC814

FEATURES

ANALOG I/O
- 6 Channel 247kSPS ADC
  - 12 Bit Resolution
  - High Speed ADC to Serial RAM Capture
- Dual Voltage Output DAC's
  - 12 Bit Resolution, 15uS Settling Time

Memory
- 8 Kbytes On-Chip Flash/EE Program Memory
- 640 Bytes On-Chip Flash/EE Data Memory
- Flash/EE, 100 Yr Retention, 100 Kcycles Endurance
- 256 Bytes On-Chip Data RAM

8051-Based Core
- 8051-Compatible Instruction Set (16.78 MHz Max)
- 32 kHz External Crystal, On-Chip Programmable PLL
- Three 16-Bit Timer/Counters
- 11 Programmable I/O Lines
- 11 Interrupt Sources, Two 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
- On-Chip Temperature Monitor
- Precision Voltage Reference
- Time Interval Counter (TIC)
- UART Serial I/O
- I^2C®-Compatible and SPI® Serial I/O
- Watchdog Timer (WDT), Power Supply Monitor (PSM)

Small Footprint Package
- 28 Pin TSSOP 4.4mm × 9.7mm body package

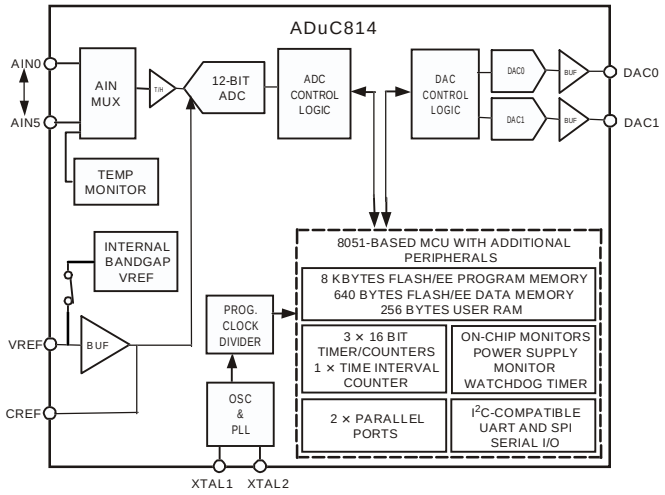
APPLICATIONS

Smart Sensors

Battery Powered Systems (Portable PC's, Instruments, Monitors)

Small Footprint Sensors and Acquisition Systems

FUNCTIONAL BLOCK DIAGRAM



GENERAL DESCRIPTION

The ADuC814 is a fully integrated 247kSPS 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 32kHz 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. 8Kbytes 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 DAC's, power supply monitor, and a bandgap reference. On-chip digital peripherals include a watchdog timer, time interval counter, three timer/counters, and three serial I/O ports (SPI, UART, and I^2C-compatible).

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 functional block diagram of the ADuC814 is shown above with a more detailed block diagram shown in Figure 9 (page18).

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 tiny 28-lead TSSOP package.

REV. PrE 10/01

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

TABLE OF CONTENTS

FEATURES 1
GENERAL DESCRIPTION 1
SPECIFICATIONS 3
TIMING SPECIFICATIONS 7
ABSOLUTE MAXIMUM RATINGS 14
ORDERING GUIDE 14
PIN CONFIGURATION 14
PIN FUNCTION DESCRIPTIONS 15
BLOCK DIAGRAM 18
MEMORY ORGANIZATION 19
OVERVIEW OF MCU-RELATED SFRS 20
Accumulator (ACC) 20
B SFR (B) 20
Stack Pointer (SP) 20
Data Pointer (DPTR) 20
Program Status Word (PSW) 20
Power Control (PCON) 20
SPECIAL FUNCTION REGISTERS 21
ADC CIRCUIT INFORMATION 22
General Overview 22
ADC Transfer Function 22
ADC Output Result Format 22
ADC SFR INTERFACE 23
ADCCON1 23
ADCCON2 24
ADCCON3 25
DRIVING THE A/D CONVERTER 26
Voltage Reference 27
Configuring the ADC 28
Initiating ADC Conversions 28
ADC High Speed Data Capture Mode 28
ADC Calibration Overview 22
ADC Offset and Gain Coefficients 29
Calibrating the ADC 29

NONVOLATILE FLASH/EE MEMORY 30
Overview 30
ADuC814 Flash/EE Memory Reliability 31
Using the Flash/EE Program Memory 31
Serial-Downloading 31
Parallel Programming 31
Flash/EE Program Memory Security 32
Using the Flash/ee Data Memory 32
Flash/EE Memory Timing 33
Using the Flash/EE Data Memory Interface 33

USER INTERFACE TO OTHER ON-CHIP ADuC814 PERIPHERALS 34
DAC 34
Using the D/A Converter 35
On-Chip PLL 37
Time Interval Counter 38
Watch Dog Timer 41
Power Supply Monitor 42
ADuC814 Configuration Register (CFG812S) 43
Serial Peripheral Interface 44
I^2C-Compatible Interface 46

8051-COMPATIBLE ON-CHIP PERIPHERALS 47
Parallel I/O Ports 1 and 3 47
Timers/Counters 47
Timers/Counter Operating Modes 50
UART Serial Interface 53
Interrupt System 56
HARDWARE DESIGN CONSIDERATIONS 58
Clock Oscillator 58
Power Supplies 58
Power Consumption 58
Power-Saving Modes 58
Grounding and Board Layout Recommendations 59
OTHER HARDWARE CONSIDERATIONS 60
In-Circuit Serial Download Access 60
Embedded Serial Port Debugger 60
Single-Pin Emulation Mode 60
Enhanced-Hooks Emulation Mode 60
Typical System Configuration 60
QUICKSTART DEVELOPMENT SYSTEM 61
OUTLINE DIMENSIONS 62

ADuC814–SPECIFICATIONS[1]

(AV_{DD} = DVDD = 3.0V or 5.0V ± 10%, 2.5V Internal Reference, XTAL1/XTAL2 = 32.768 kHz Crystal; DAC Vout Load to AGND; RL=10k, CL = 100pF,all specifications T_{MIN} to T_{MAX} unless otherwise noted.)

Parameter	ADuC814ARU V_{DD}=5V	ADuC814ARU V_{DD}=3V	Units	Test Conditions
ADC CHANNEL SPECIFICATIONS				
DC ACCURACY [2]				f_{SAMPLE} = 147kHz
Resolution	12	12	Bits	
Integral Nonlinearity	±2	±2	LSB max.	
	±1	±1	LSB typ.	
Differential Nonlinearity	±4	±4	LSB max.	
	±2	±2	LSB typ.	
CALIBRATED ENDPOINT ERRORS [3,4]				
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				f_{IN} = 10kHz Sine Wave f_{SAMPLE} = 147kHz
Signal to Noise Ratio (SNR) [5]	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 [16]	-80	-80	dB typ	
ANALOG INPUT				
Input Voltage Ranges	0 to V_{REF}	0 to V_{REF}	Volts	
Leakage Current	±1	±1	uA max	
Input Capacitance	32	32	pF typ	
TEMPERATURE MONITOR [6]				
Voltage Output at 25°C	600	600	mV typ	
Voltage TC	-3.5	-3.5	mV/°C typ.	
DAC CHANNEL SPECIFICATIONS				
DC ACCURACY [7]				
Resolution	12	12	Bits	
Relative Accuracy	±3	±3	LSB typ	
Differential Nonlinearity	-1	-1	LSB max	Guaranteed 12-Bit Monotonic
	±1/2	±1/2	LSB typ	
Offset Error	±50	±50	mV max	
Gain Error	±1	±1	% max	AVDD Range
	±1	±1	% typ	VREF Range
Gain Error Mismatch	0.5	0.5	% typ	of Fullscale on DAC1
ANALOG OUTPUTS				
Voltage Range_0	0 to V_{REF}	0 to V_{REF}	Volts	DAC V_{REF} = 2.5V
Voltage Range_1	0 to V_{DD}	0 to V_{DD}	Volts	DAC V_{REF} = VDD
Resistive Load	10	10	KΩ	
Capacitive Load	100	100	pF	
Output Impedance	0.5	0.5	Ω typ	
I sink	50	50	uA typ	
DAC AC Specifications				
Voltage Output Settling Time	15	15	us typ	Fullscale Settling Time to Within 1/2LSB of Final Value
Digital-to-Analog Glitch Energy	10	10	nVsec typ	1 LSB change at major carry

ADuC814–SPECIFICATIONS[1]

Parameter	ADuC814ARU V_{DD}=5V	ADuC814ARU V_{DD}=3V	Units	Test Conditions
REFERENCE INPUT / OUTPUT				
REFERENCE OUTPUT [8]				
Output Voltage (V_{REF})	2.5	2.5	V typ	
Accuracy	± 2.5	± 2.5	% max	of V_{REF} measured at the CREF pin.
Power Supply Rejection	47	57	dB typ	
Reference Tempco	± 100	± 100	ppm/°C typ	
EXTERNAL REFERENCE INPUT [9]				
Voltage Range (V_{REF})	1.8	1.8	V min.	
	VDD	VDD	V max.	
Input Impedance	20	20	kΩ typ.	
Input Leakage	10	10	µA max	Internal Bandgap Deselected via ADCCON2.6
POWER SUPPLY MONITOR (PSM)				
VDD Trip Point Selection Range	2.63	2.63	V min	
	4.63		Vmax	Four Trip Points Selectable in This Range Programmed via TP1-0 in PSMCON
VDD Power Supply Trip Point Accuracy	± 3.5	± 3.5	% max	
WATCH DOG TIMER (WDT)				
Time-out Period	0	0	mS min	Nine Time-out Periods Selectable in This Range Programmed via PRE3-0 in WDCON
	2000	2000	mS max.	
LOGIC INPUTS				
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	
SCLOCK and RESET Only (Schmitt-Triggered Inputs)[2]				
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, SDATA/MOSI, MISO, $\overline{SS}$ [10]	–10	-5	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
	35	10	µA min	V_{IN} = 5V,3V Internal Pull Down
	105	35	µA max	V_{IN} = 5V,3V Internal Pull Down
P1.0, P1.1, Port 3	±10	±10	µA max	V_{IN} = 5V,3V
	± 1	± 1	µA typ	
	–180	-100	µA min	V_{IN} = 2 V, VDD=5V, 3V
	–660	–200	µA max	
	-420	-100	µA typ	
	–20	-5	µA min	V_{IN} = 450 mV, VDD = 5V, 3V
	–75	-25	µA max	
	-38	-12	µA typ	
Input Capacitance	5	5	pF typ	All Digital Inputs

Parameter	ADuC814ARU V_{DD}=5V	V_{DD}=3V	Units	Test Conditions
CRYSTAL OSCILLATOR (XTAL1 AND XTAL2)				
Logic Inputs, XTAL1 Only				
VINL, Input Low Voltage	0.8	0.4	V typ	
VINH, 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 (Voh)	2.4	2.4	Vmin	Isource = 1.6mA
Output Low Voltage (Vol)				
Port 1.0 and Port 1.1	0.4	0.4	V max	Isink = 1.6mA
SCLOCK, SDATA/MOSI	0.4	0.4	V max	Isink = 4mA
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	Clock rate generated via
	16.78	16.78	MHz	on-chip PLL, programmable via CD2-0 in PLLCON
START UP TIME				
At Power-On	500	500	ms typ	
From Idle Mode	1	1	ms typ	
From Power-Down Mode				
Oscillator Running				OSC_PD Bit = 0 in PLLCON SFR
Wakeup with $\overline{INT0}$ Interrupt	1	1	ms typ	
Wakeup with SPI/I^2C Interrupt	1	1	ms typ	
Wakeup with TIC Interrupt	1	1	ms typ	
Wakeup with External RESET	3.4	3.4	ms typ	
Oscillator Powered Down				OSC_PD Bit = 1 in PLLCON SFR
Wakeup with External RESET	0.9	0.9	sec typ	
After External RESET in Normal Mode	3.3	3.3	ms typ	
After WDT Reset in Normal Mode	3.3	3.3	ms typ	Controlled via WDCON SFR
FLASH/EE MEMORY RELIABILITY CHARACTERISTICS[11]				
Endurance[12]	100,000	100,000	Cycles min	
Data Retention[13]	100	100	Years min	
POWER REQUIREMENTS [14,15]				
Power Supply Voltages				
AV_{DD} / DV_{DD} - AGND		2.7	V min.	AV_{DD} / DV_{DD} = 3V nom.
		3.3	V max.	
	4.5		V min.	AV_{DD} / DV_{DD} = 5V nom.
	5.5		V max.	
Power Supply Currents Normal Mode				
D_{VDD} Current	6	2	mA max	Core CLK=2.097MHz
	3.5	1.5	mA typ	
A_{VDD} Current	1.4	1.4	mA max	
D_{VDD} Current	20	10	mA max	Core CLK=16.78MHz
	14	7	mA typ	
A_{VDD} Current	1.4	1.4	mA max	
Power Supply Currents Idle Mode				
D_{VDD} Current	2.5	1.5	mA max	Core CLK=2.097MHz
	1.4	0.9	mA typ	
A_{VDD} Current	0.11	0.11	mA typ	
D_{VDD} Current	6	3	mA max	Core CLK=16.78MHz
	4	1	mA typ	
A_{VDD} Current	0.11	0.11	mA typ	

ADuC814–SPECIFICATIONS[1]

Parameter	ADuC814ARU V_{DD}=5V	V_{DD}=3V	Units	Test Conditions
POWER REQUIREMENTS (Cont'd)				
Power Supply Currents Power Down Mode				Core CLK=2.097MHz or 16.78 MHz
D_{VDD} Current	80	25	µA max	Osc. On
	38	14	µA typ	
A_{VDD} Current	2	1	µA typ	
D_{VDD} Current	35	15	µA max	Osc. Off
	25	12	µA typ	
Typical Additional Power Supply Currents				Core CLK=2.097MHz, AVDD = DVDD = 5V
PSM Peripheral	50		µA typ	
ADC	1.5		mA typ	
DAC	150		µA typ	

NOTES

[1] Temperature Range -40°C to +85°C.

[2] ADC Linearity is guaranteed during normal MicroConverter Core operation.

[3] Offset and Gain Error and Offset and Gain Error Match are measured after calibration.

[4] User may need to execute Hardware Calibration Options (Ref: ADC Section) to achieve these specifications.

[5] SNR calculation includes distortion and noise components.

[6] The Temperature Monitor will give a measure of the die temperature directly, air temperature can be inferred from this result.

[7] DAC linearity is calculated using :

reduced code range of 48 to 4095, 0 to Vref range.

reduced code range of 48 to 3950, 0 to V_{DD} range.

DAC Output Load = 10K Ohms and 100 pF.

[8] Power-up time for the Internal Reference will be determined on the value of the decoupling capacitor chosen for both the Vref and Cref pins.

[9] When using an External Reference device, the internal bandgap reference input can be bypassed by setting the ADCCON1.6 bit. In this mode the Vref and Cref pins need to be shorted together for correct operation.

[10] Pins configured in I2C-compatible mode or SPI mode, pins configured as digital inputs during this test.

[11] Flash/EE Memory Reliability Characteristics apply to both the Flash/EE program memory and the Flash/EE data memory.

[12] Endurance is qualified to 100 Kcycles as per JEDEC Std. 22 method A117 and measured at -40°C, +25°C, and +85°C, typical endurance at 25°C is 700 Kcycles.

[13] Retention lifetime equivalent at junction temperature (Tj) = 55°C as per JEDEC Std. 22 method A117. Retention lifetime based on an activation energy of 0.6eV will derate with junction temperature as shown in Figure 27 in the Flash/EE Memory description section of this data sheet.

[14] Power Supply current consumption is measured in Normal, Idle, and Power-Down Modes under the following conditions:

Normal Mode: Reset = 0.4 V, Digital I/O pins = open circuit, Core Clk changed via CD bits in PLLCON, Core Executing internal software loop.

Idle Mode: Reset = 0.4 V, 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 = 0.4 V, 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.

[15] D_{VDD} power supply current will increase typically by 3 mA (3 V operation) and 10 mA (5 V operation) during a Flash/EE memory program or erase cycle.

[16] Channel to Channel Crosstalk is measure on adjacent channels.

Typical specifications are not production tested, but are supported by characterization data at initial product release.

Specifications subject to change without notice.

TIMING SPECIFICATIONS[1, 2, 3]

(AV_{DD} = 2.7 V to 3.6 V or 4.75 V to 5.25 V, DV_{DD} = 2.7 V to 3.6 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 Min	Typ	Max	Unit	Figure
CLOCK INPUT (External Clock Driven XTAL1)						
t_{CK}	XTAL1 Period		30.52		µs	1
t_{CKL}	XTAL1 Width Low		15.16		µs	1
t_{CKH}	XTAL1 Width High		15.16		µs	1
t_{CKR}	XTAL1 Rise Time		20		ns	1
t_{CKF}	XTAL1 Fall Time		20		ns	1
$1/t_{CORE}$	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	

NOTES

[1]AC inputs during testing are driven at DV_{DD} – 0.5 V for a Logic 1, and 0.45 V for a Logic 0. Timing measurements are made at V_{IH} min for a Logic 1, and V_{IL} max for a Logic 0 as shown in Figure 2.

[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 2.

[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 16.78 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.

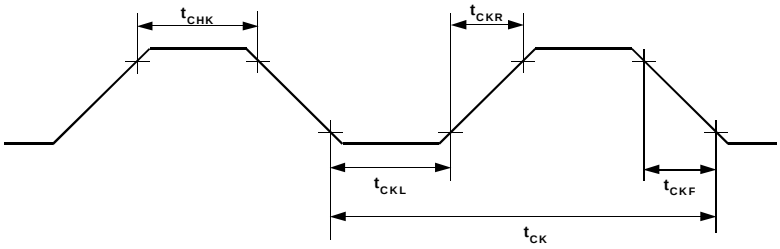


Figure 1. XTAL1 Input

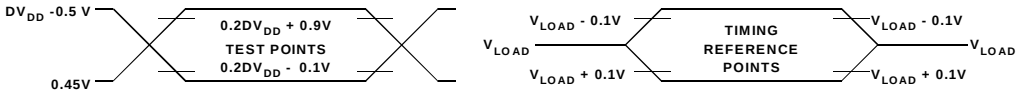


Figure 2. Timing Waveform Characteristics

Parameter		16.78 MHz Core_Clk Min	Typ	Max	Variable Core_Clk Min	Typ	Max	Unit	Figure
UART TIMING (Shift Register Mode)									
t_{XLXL}	Serial Port Clock Cycle Time		715			$12t_{CORE}$		μs	6
t_{QVXH}	Output Data Setup to Clock	463			$10t_{CORE} - 133$			ns	6
t_{DVXH}	Input Data Setup to Clock	252			$2t_{CORE} + 133$			ns	6
t_{XHDX}	Input Data Hold after Clock	0			0			ns	6
t_{XHQX}	Output Data Hold after Clock	22			$2t_{CORE} - 117$			ns	6

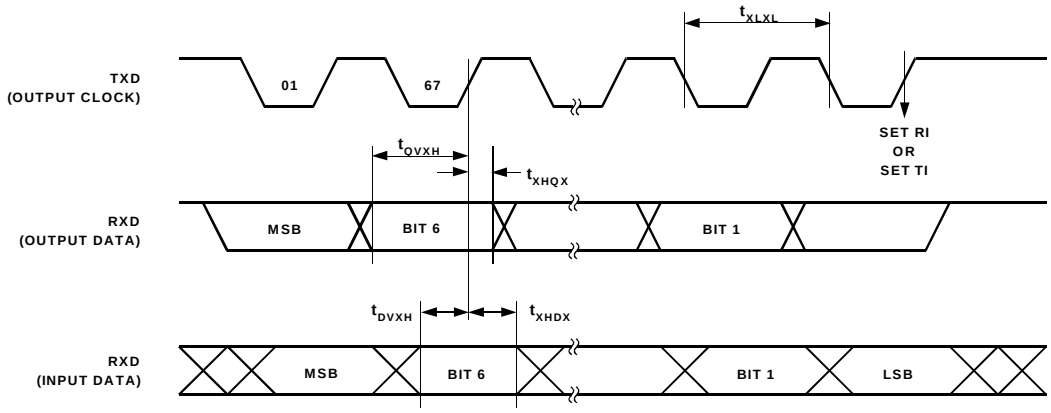


Figure 3. UART Timing in Shift Register Mode

Parameter		Min	Max	Unit	Figure
I²C-COMPATIBLE INTERFACE TIMING					
t_L	SCLOCK Low Pulsewidth	4.7		µs	7
t_H	SCLOCK High Pulsewidth	4.0		µs	7
t_{SHD}	Start Condition Hold Time	0.6		µs	7
t_{DSU}	Data Setup Time	100		µs	7
t_{DHD}	Data Hold Time		0.9	µs	7
t_{RSU}	Setup Time for Repeated Start	0.6		µs	7
t_{PSU}	Stop Condition Setup Time	0.6		µs	7
t_{BUF}	Bus Free Time Between a STOP Condition and a START Condition	1.3		µs	7
t_R	Rise Time of Both SCLOCK and SDATA		300	ns	7
t_F	Fall Time of Both SCLOCK and SDATA		300	ns	7
t_{SUP}[1]	Pulsewidth of Spike Suppressed		50	ns	7

NOTE
[1]Input filtering on both the SCLOCK and SDATA inputs suppresses noise spikes less than 50 ns.

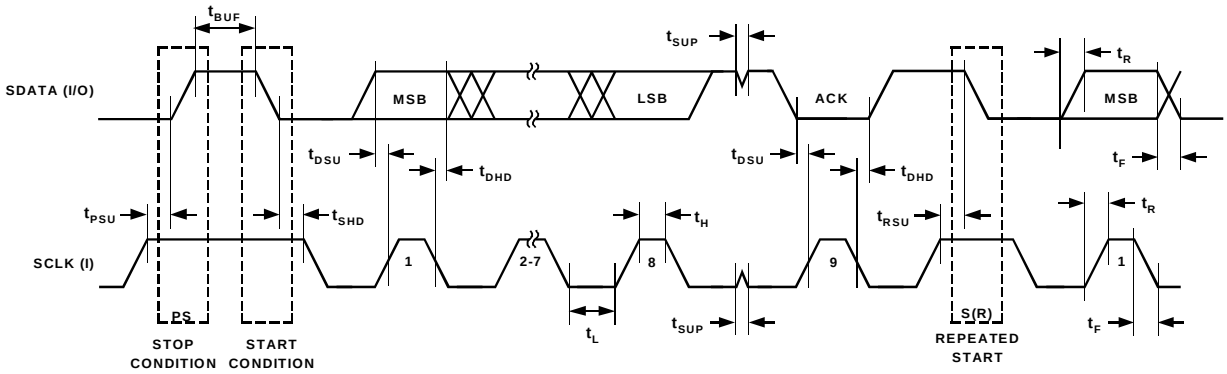


Figure 4 I²C-Compatible Interface Timing

Parameter		Min	Typ	Max	Unit	Figure
SPI MASTER MODE TIMING (CPHA = 1)						
t_{SL}	SCLOCK Low Pulsewidth[1]		630		ns	8
t_{SH}	SCLOCK High Pulsewidth[1]		630		ns	8
t_{DAV}	Data Output Valid after SCLOCK Edge			50	ns	8
t_{DSU}	Data Input Setup Time before SCLOCK Edge	100			ns	8
t_{DHD}	Data Input Hold Time after SCLOCK Edge	100			ns	8
t_{DF}	Data Output Fall Time		10	25	ns	8
t_{DR}	Data Output Rise Time		10	25	ns	8
t_{SR}	SCLOCK Rise Time		10	25	ns	8
t_{SF}	SCLOCK Fall Time		10	25	ns	8

NOTE
[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 and
b. SPI bit-rate selection bits SPR1 and SPR0 bits in SPICON SFR set to 0 and 0 respectively.

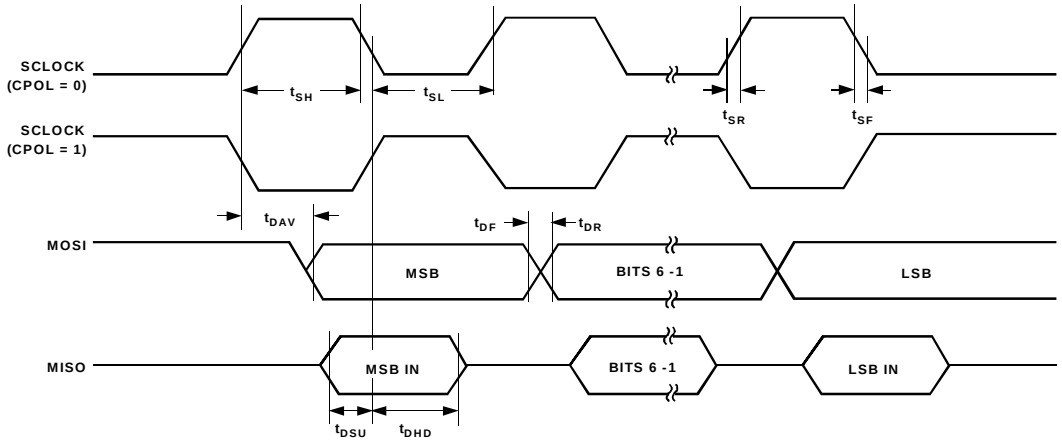


Figure 5. SPI Master Mode Timing (CPHA = 1)

Parameter		Min	Typ	Max	Unit	Figure
SPI MASTER MODE TIMING (CPHA = 0)						
t_{SL}	SCLOCK Low Pulsewidth[1]		630		ns	9
t_{SH}	SCLOCK High Pulsewidth[1]		630		ns	9
t_{DAV}	Data Output Valid after SCLOCK Edge			50	ns	9
t_{DOSU}	Data Output Setup before SCLOCK Edge			150	ns	9
t_{DSU}	Data Input Setup Time before SCLOCK Edge	100			ns	9
t_{DHD}	Data Input Hold Time after SCLOCK Edge	100			ns	9
t_{DF}	Data Output Fall Time		10	25	ns	9
t_{DR}	Data Output Rise Time		10	25	ns	9
t_{SR}	SCLOCK Rise Time		10	25	ns	9
t_{SF}	SCLOCK Fall Time		10	25	ns	9

NOTE
[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 and
b. SPI bit-rate selection bits SPR1 and SPR0 bits in SPICON SFR set to 0 and 0 respectively.

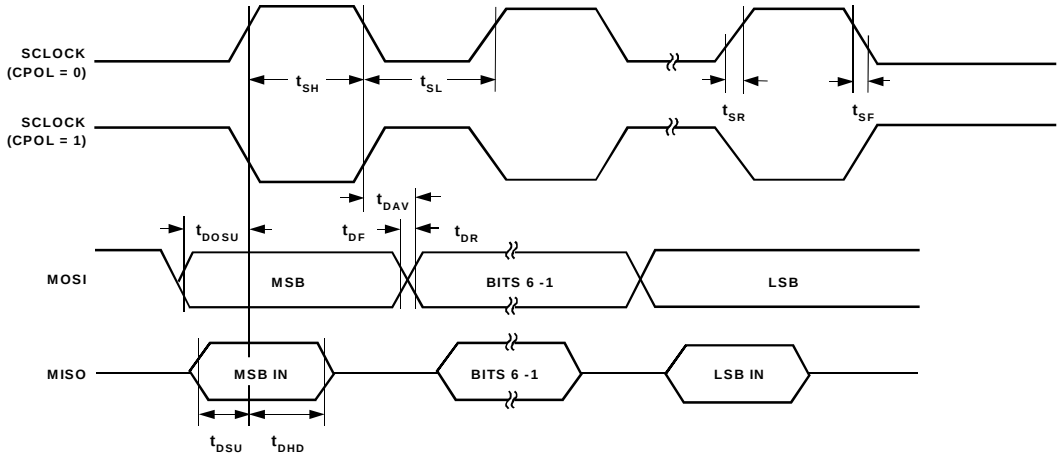


Figure 6. SPI Master Mode Timing (CPHA = 0)

Parameter		Min	Typ	Max	Unit	Figure
SPI SLAVE MODE TIMING (CPHA = 1)						
t_{SS}	$\overline{SS}$ to SCLOCK Edge	0			ns	10
t_{SL}	SCLOCK Low Pulsewidth		330		ns	10
t_{SH}	SCLOCK High Pulsewidth		330		ns	10
t_{DAV}	Data Output Valid after SCLOCK Edge			50	ns	10
t_{DSU}	Data Input Setup Time before SCLOCK Edge	100			ns	10
t_{DHD}	Data Input Hold Time after SCLOCK Edge	100			ns	10
t_{DF}	Data Output Fall Time		10	25	ns	10
t_{DR}	Data Output Rise Time		10	25	ns	10
t_{SR}	SCLOCK Rise Time		10	25	ns	10
t_{SF}	SCLOCK Fall Time		10	25	ns	10
t_{SFS}	$\overline{SS}$ High after SCLOCK Edge	0			ns	10

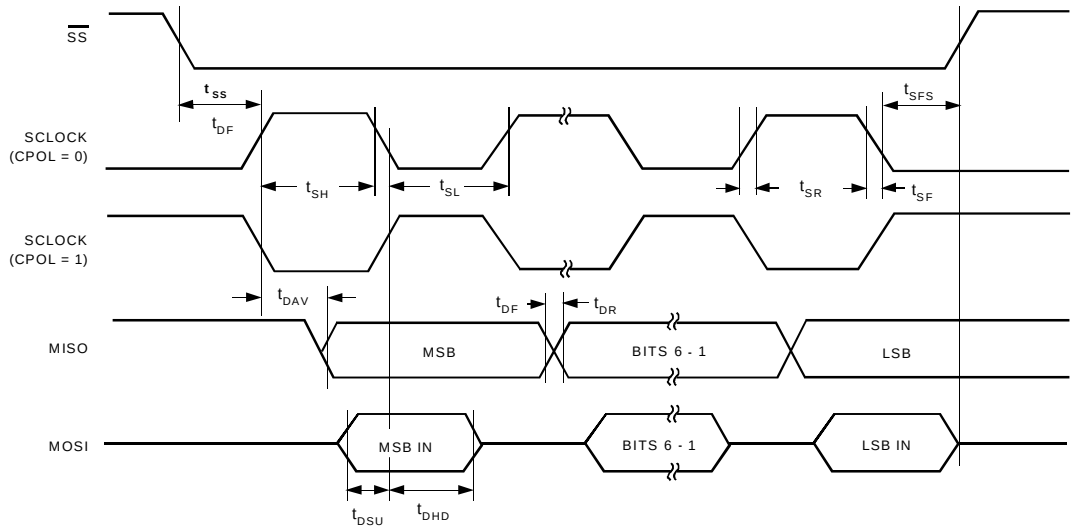


Figure 7. SPI Slave Mode Timing (CPHA = 1)

Parameter		Min	Typ	Max	Unit	Figure
SPI SLAVE MODE TIMING (CPHA = 0)						
t_{SS}	$\overline{SS}$ to SCLOCK Edge	0			ns	11
t_{SL}	SCLOCK Low Pulsewidth		330		ns	11
t_{SH}	SCLOCK High Pulsewidth		330		ns	11
t_{DAV}	Data Output Valid after SCLOCK Edge			50	ns	11
t_{DSU}	Data Input Setup Time before SCLOCK Edge	100			ns	11
t_{DHD}	Data Input Hold Time after SCLOCK Edge	100			ns	11
t_{DF}	Data Output Fall Time		10	25	ns	11
t_{DR}	Data Output Rise Time		10	25	ns	11
t_{SR}	SCLOCK Rise Time		10	25	ns	11
t_{SF}	SCLOCK Fall Time		10	25	ns	11
t_{SSR}	$\overline{SS}$ to SCLOCK Edge			50	ns	11
t_{DOSS}	Data Output Valid after $\overline{SS}$ Edge			20	ns	11
t_{SFS}	$\overline{SS}$ High after SCLOCK Edge	0			ns	11

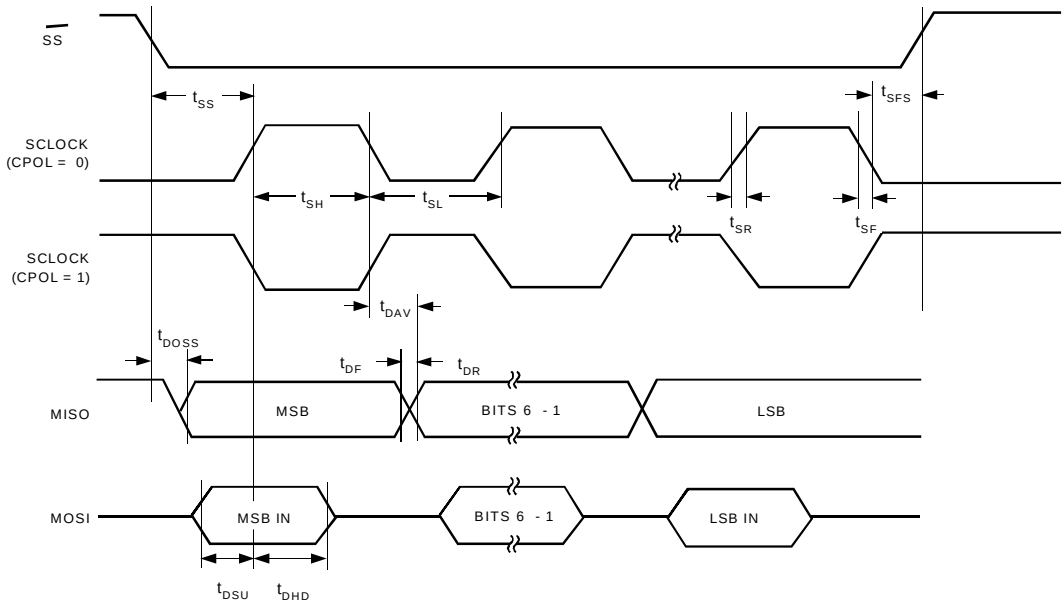


Figure 8. SPI Slave Mode Timing (CPHA = 0)

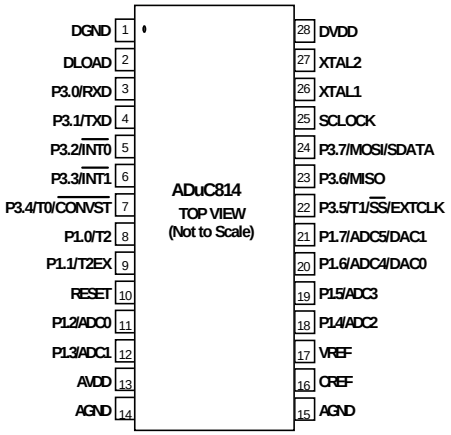
ABSOLUTE MAXIMUM RATINGS[1]

(T_A = 25°C unless otherwise noted)

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[2]	–0.3 V to +0.3 V
Analog Input Voltage to AGND[3]	–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 +85°C
Storage Temperature Range	–65°C to +150°C
Junction Temperature	150°C
θ_{JA} Thermal Impedance	90°C/W
Lead Temperature, Soldering	
Vapor Phase (60 sec)	215°C
Infrared (15 sec)	220°C

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

PIN CONFIGURATION
28-Lead TSSOP



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 the ADuC814 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.

ORDERING GUIDE

Model	Temperature Range	Package Description	Package Option
ADuC814ARU	–40°C to +85°C	Tiny Shrink Small outline Package	RU-28

QuickStart Development System Model	Description
EVAL-ADUC814QS	Development System for the ADuC814 MicroConverter, Containing: Evaluation Board Serial Port Cable Windows® Serial Downloader (WSD) Windows Debugger (DeBug) Windows ADuC814 Simulator (ADSIM) Windows ADC Analysis Software Program (WASP) 8051 Assembler (Metalink) Example Code Documentation

Windows is a registered trademark of Microsoft Corporation.

PIN FUNCTION DESCRIPTION

28 PIN TSSOP

Pin Number	Pin Name
1	DGND
2	DLOAD
3	P3.0/RXD
4	P3.1/TXD
5	P3.2/$\overline{\text{INT0}}$
6	P3.3/$\overline{\text{INT1}}$
7	P3.4/T0/$\overline{\text{CONVST}}$
8	P1.0/T2
9	P1.1/T2EX
10	RESET
11	P1.2/ADC0
12	P1.3/ADC1
13	AV_{DD}
14	AGND
15	AGND
16	VREF
17	CREF
18	P1.4/ADC2
19	P1.5/ADC3
20	P1.6/ADC4/DAC0
21	P1.7/ADC5/DAC1
22	P3.5/T1/$\overline{\text{SS}}$/EXTCLK
23	P3.6/MISO
24	P3.7/MOSI/SDATA
25	SCLOCK
26	XTAL1
27	XTAL2
28	DV_{DD}

PIN FUNCTION DESCRIPTION

Mnemonic	Type	Function
DGND	S	Digital ground, ground reference point for the digital circuitry.
DLOAD	I	Enables Debug / Serial Download mode when pulled high through a resistor on power-on or RESET. User code is executed when this pin is pulled low on power-on or RESET.
P3.0 - P3.4	I/O	P3.0 - P3.4 are 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 Port 3 pins being pulled externally low will 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 below.
P3.0 / RXD	I/O	Receiver Data Input (asynchronous) or Data Input/Output (synchronous) in serial (UART) mode.
P3.1 / TXD	I/O	Transmitter Data Output (asynchronous) or Clock Output (synchronous) in serial (UART) mode.
P3.2 / 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 a gate control input to Timer0.
P3.3 / 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 a gate control input to Timer1.
P3.4/T0/CONVST	I/O	Timer / Counter 0 Input and External Trigger input for ADC conversion start.
P1.0 - P1.1	I/O	P1.0 - P1.1 are 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 Port 1 pins being pulled externally low will 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 below.
P1.0 / T2	I/O	Timer 2 Digital Input. Input to Timer/Counter 2. When Enabled, Counter2 is incremented in response to a 1 to 0 transition of the T2 input.
P1.1 / T2EX	I/O	Digital Input. Capture / Reload trigger for Counter 2.
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:
P1.2/ADC0	I	ADC Input Channel 0, selected via ADCCON2 SFR
P1.3/ADC1	I	ADC Input Channel 1, selected via ADCCON2 SFR
AVDD	S	Analog Positive Supply Voltage, +3V or +5V.
AGND	G	Analog Ground. Ground Reference point for the analog circuitry.

Mnemonic	Type	Function
V_{REF}	I/O	Reference Input/Output. This pin is connected to the internal reference through a series resistor 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 over driven by an external reference. Connect 0.1uF between this pin and AGND.
C_{REF}	I	Decoupling input for on-chip reference. Connect 0.1uF between this pin and AGND.
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:
P1.4/ADC2	I	ADC Input Channel 2, selected via ADCCON2 SFR
P1.5/ADC3	I	ADC Input Channel 2, selected via ADCCON2 SFR
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
P1.7/ADC5/DAC1	I/O	ADC Input Channel 5, selected via ADCCON2 SFR. The voltage DAC channel 0 can also be configured to appear on P1.7
P3.5 - P3.7	I/O	P3.5 - P3.7 are 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 Port 3 pins being pulled externally low will 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 below.
P3.5/T1	I/O	Timer / Counter 1 Input.
		P3.5 - P3.7 pins also have I2C compatible and SPI interface functions. To enable these functions, bit 0 of the CFG812S SFR must be set to 1.
P3.5/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 CFG812S SFR.
P3.6/MISO	I/O	SPI Master Input / Slave Output Data Input/Output pin.
P3.7/MOSI/SDATA	I/O	User selectable, I2C Compatible Data Input/Output pin or SPI Master Output / Slave Input Data Input/Output pin.
SCLOCK	I/O	Serial Clock pin for I2C Compatible or SPI serial interface clock.
XTAL1	I	Input to the crystal oscillator inverter.
XTAL2	O	Output from the crystal oscillator inverter
DVDD	S	Analog Positive Supply Voltage, +3V or +5V.

*I = Input, O = Output, S = Supply.

NOTES

1. In the following descriptions, SET implies a Logic 1 state and CLEARED implies a Logic 0 state unless otherwise stated.
2. In the following descriptions, SET and CLEARED also imply that the bit is set or automatically cleared by the ADuC814 hardware unless otherwise stated.
3. User software should not write 1s to reserved or unimplemented bits as they may be used in future products.

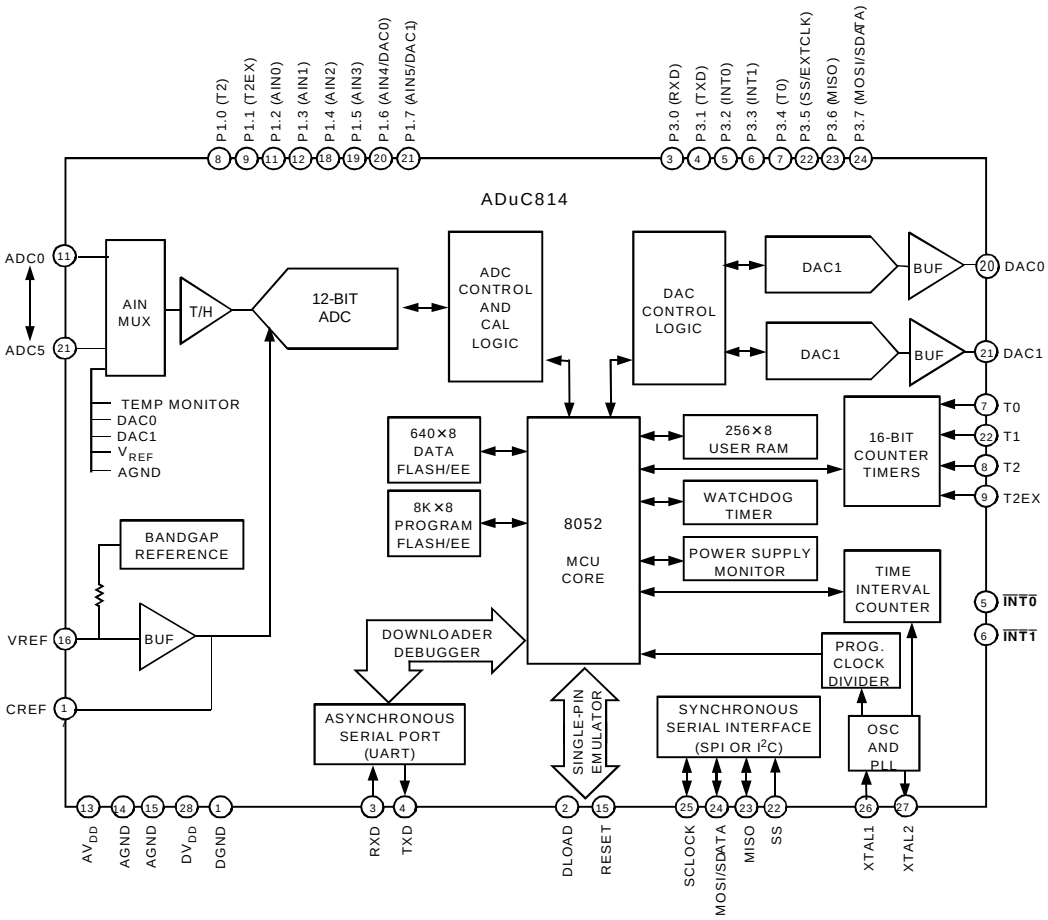


Figure 9. ADuC814 Block Diagram

MEMORY ORGANIZATION

The ADuC814 does not have Port 0 and Port 2 pins and therefore does not support external Program and Data memory interfaces. The device executes code from the internal 8K Byte Flash/EE program memory. This internal code space can be programmed via the UART serial port while the device is in-circuit. The program memory space of the ADuC814 is shown in figure 10 below.

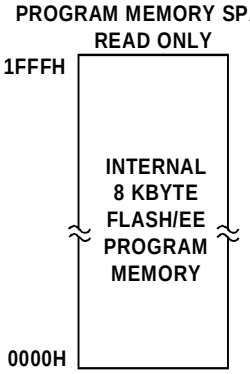


Figure 10. Program Memory Map

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 11, the 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 Special Function Register (SFR) area. Access to the Flash/EE Data memory is discussed in detail later as part of the Flash/EE memory section in this data sheet.

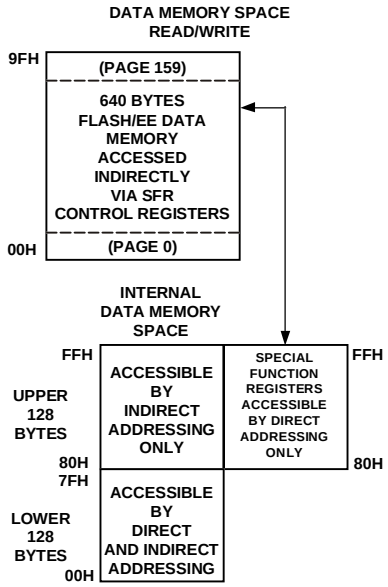


Figure 11. Data Memory Map

The lower 128 bytes of internal data memory are mapped as shown in Figure 12. The lowest 32 bytes are grouped into four banks of eight registers addressed as R0 through R7. The next 16 bytes (128 bits), locations 20Hex through 2FHex 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.

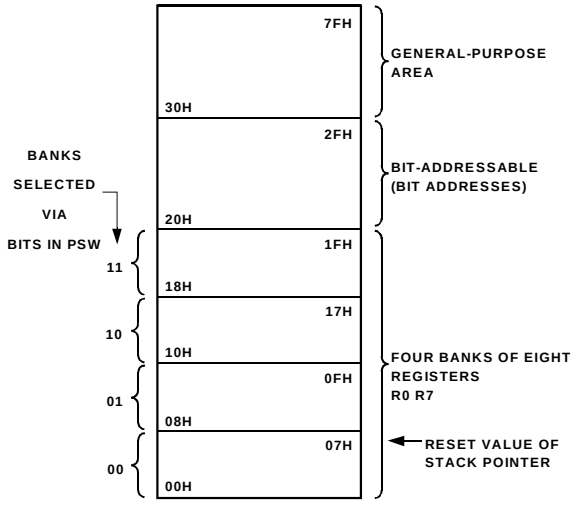


Figure 12. Lower 128 Bytes of Internal Data Memory

Reset initializes the stack pointer to location 07 hex and increments it once to start from locations 08 hex 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 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 13. A complete SFR map is shown in Figure 14.

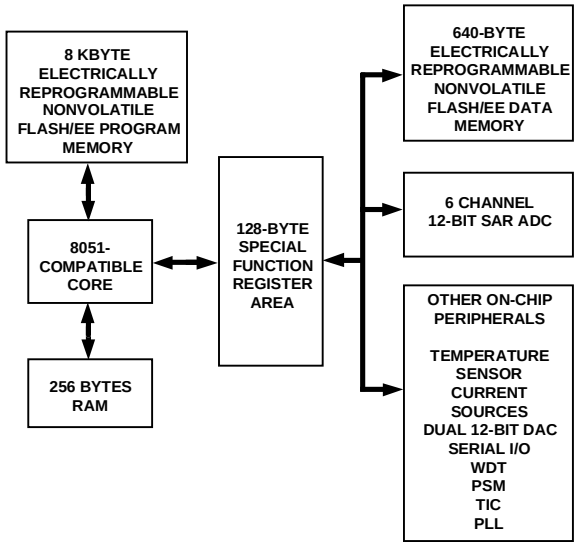


Figure 13. Programming Model

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 that is 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 are used to provide memory addresses for internal code access. It 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 PSW register is the Program Status Word which contains several bits reflecting the current status of the CPU as detailed in Table I.

SFR Address	D0H
Power ON Default Value	00H
Bit Addressable	Yes

CY	AC	F0	RS1	RS0	OV	F1	P

Table I. PSW SFR Bit Designations

Bit	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
		0	0	0
		0	1	1
		1	0	2
		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 II.

SFR Address	87H
Power ON Default Value	00H
Bit Addressable	No

SMOD	SERIPD	INT0PD		GF1	GF0	PD	IDL

Table II. PCON SFR Bit Designations

Bit	Name	Description
7	SMOD	Double UART Baud Rate
6	SERIPD	I^2C/SPI Power-Down Interrupt Enable
5	INT0PD	$\overline{INT0}$ Power-Down Interrupt Enable
4	RSVD	Reserved for Future use
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 17 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 user software.

ISPI FFH 0	WCOL FEH 0	SPE FDH 0	SPIM FCH 0	CPOL FBH 0	CPHA FAH 0	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
MDO EFH 0	MDE EEH 0	MCO EDH 0	MDI ECH 0	I2CM EBH 0	I2CRS EAH 0	I2CTX E9H 0	I2CI E8H 0	BITS
E7H 0	E6H 0	E5H 0	E4H 0	E3H 0	E2H 0	E1H 0	E0H 0	BITS
ADCI DFH 0	DMA 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	RSI 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 0	WDIR C3H 0	WDS C2H 0	WDE C1H 0	WDWR C0H 0	BITS
PS1 BFH 0	PADC BEH 0	PT2 BDH 0	PS BCH 0	PT1 BBH 0	PX1 BAH	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	EADC AEH	ET2 ADH	ES ACH 0	ET1 ABH 0	EX1 AAH	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	T1 99H 0	R1 98H 0	BITS
97H 0	96H 0	95H 0	94H 0	93H 0	92H 0	T2EX 91H 0	T2 90H 0	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 00H	RESERVED	RESERVED
B[1] F0H 00H	ADCOFSL F1H 00H	ADCOFSH F2H 00H	ADCGAINL F3H 00H	ADCGAINH F4H 00H	ADCCON3 F5H 00H	RESERVED	SPIDAT F7H 00H
I2CCON[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 03H
T2CON[1] C8H 00H	RESERVED	RCAP2L CAH 00H	RCAP2H CBH 00H	TL2 CCH 00H	TH2 CDH 00H	RESERVED	RESERVED
WDCON[1] C0H 00H	RESERVED	RESERVED	RESERVED	NOT USED	RESERVED	EADRL 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[3] 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	INT VAL A6H 00H	NOT USED
SCON[1] 98H 00H	SBUF 99H 00H	I2CDAT 9AH 00H	I2CADD 9BH 55H	CFG812S 9CH 00H	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

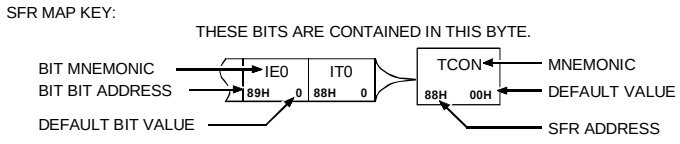


Figure 14. Special Function Register Locations and Reset Values

SFR Notes :

[1] SFRs whose address ends in 0H or 8H are bit addressable.

[2] Only P1.0 and P1.1 can operate as Digital I/O Pins. P1.2 - P1.7 can be configured as analog inputs (i.e. adc inputs) or digital inputs.

[3] The CHIPID SFR contains the silicon revision id byte and may change for future silicon revisions.

[4] These registers are reconfigured at power-on with factory calculated calibration coefficients, which can be overwritten by user code. See calibration options in ADCCON3 SFR.

[5] When the SPIM bit in the SPICON SFR is Cleared, then the SPR0 bit will reflect the level on the SS pin (pin#22).

ADC CIRCUIT INFORMATION

GENERAL OVERVIEW

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

The A/D converter consists of a conventional successive-approximation converter based around a capacitor DAC. The converter accepts an analog input range of 0 to $+V_{REF}$. A high precision, low drift and factory calibrated 2.5V 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 1.8V 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 ($\overline{CONVST}$) or alternatively Timer 2 can be configured to generate a repetitive trigger for ADC conversions.

The ADuC814 has 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 will represent 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 frontend multiplexor. These signals include a Temperature Monitor, DAC0, DAC1, V_{REF}, and AGND. The temperature monitor is a voltage output from an On-Chip bandgap 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 which 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 will 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 to V_{REF} range is shown in Figure 15.

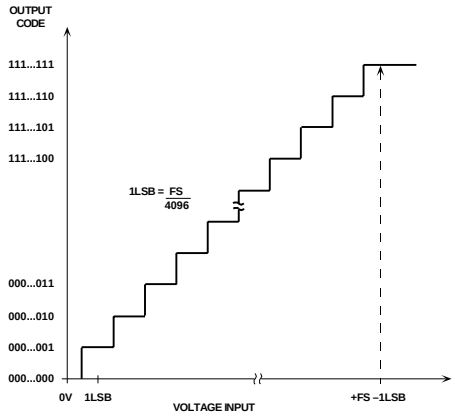


Figure 15. ADuC814 ADC Transfer Function

ADC Result Output Format

Once configured via the ADCCON 1-3 SFRs the ADC will convert the analog input and provide an ADC 12-bit result word in the ADCDATAH/L SFRs. The ADCDATAL SFR contains the bottom 8bits 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. This ID will correspond to the channel selection bits CD3 - CD0 in the ADCCON2 SFR. The format of the ADC 12-bit result word is shown in Figure 16.

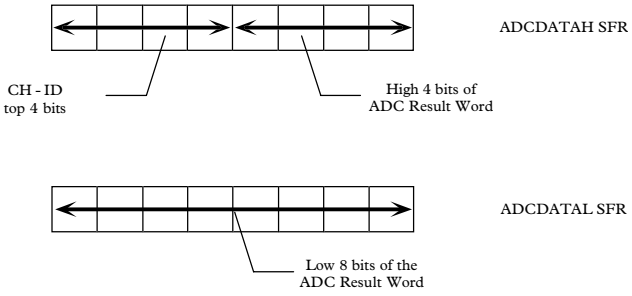


Figure 16. ADC Result Format

SFR INTERFACE TO ADC BLOCK

The ADC operation is fully controlled via 3 SFR's, ADCCON1, ADCCON2, and ADCCON3. These 3 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 Value : 00H

MODE	EXT_REF	CK1	CK0	AQ1	AQ0	T2C	EXC

Bit	Name	Description
ADCCON1.7	MODE	The Mode bit This bit selects the operating mode of the ADC Set to '1' = Power on the ADC. Set to '0' = Power down the ADC.
ADCCON1.6	EXT_REF	External Reference Select bit This bit selects which reference the ADC will use when performing a conversion. Set to '1' = Switches in an External Reference. Set to '0' = The On-Chip Bandgap Reference will be used.
ADCCON1.5 ADCCON1.4	CK1 CK0	The ADC Clock Divide bits 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 4.5MHz and below. The divider ratio is selected as follows: **CK1 CK0 PLL Divider** 0 0 8 0 1 4 1 0 16 1 1 32
ADCCON1.3 ADCCON1.2	AQ1 AQ0	The ADC Acquisition Time select bits AQ1 and AQ0 combine to select the number of ADC clocks required for the input track/hold amplifier to acquire the input signal. The Acquisition time is selected as follows: **AQ1 AQ0 #ADC Clks** 0 0 1 0 1 2 1 0 3 1 1 4
ADCCON1.1	T2C	The Timer2 conversion bit T2C is set to enable the Timer2 overflow bit to be used as the ADC convert start trigger input.
ADCCON1.0	EXC	The External Trigger enable bit 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 remaiin low for a minimum pulse width of 100nSecs at the required sample rate.

Table 6. ADCCON1 SFR bit designations

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 Value : 00H

Bit Addressable : YES

ADCI	ADCSPI	CCONV	SCONV	CS3	CS2	CS1	CS0

Bit Location	Bit Mnemonic	Description
ADCCON2.7	ADCI	The ADC interrupt bit 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 user code.
ADCCON2.6	ADCSPI	The ADCSPI Mode enable bit ADCSPI is set to enable the ADC conversion results to be transferred directly to the SPI data buffer (SPIDAT) without intervention from the CPU.
ADCCON2.5	CCONV	The Continuous Conversion bit 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 setup in the ADCCON SFRs. The ADC automatically starts another conversion once a previous conversion cycle has completed.
ADCCON2.4	SCONV	The Single Conversion bit SCONV is set to initiate a single conversion cycle. The SCONV bit is automatically reset to '0' on completion of the single conversion cycle.
ADCCON2.3 ADCCON2.2 ADCCON2.1 ADCCON2.0	CS3 CS2 CS1 CS0	The Channel Selection bits CS3 - CS0 allow the user to program the ADC channel selection under software control. Once a conversion is initiated the channel converted will be pointed to by these channel selection bits. The Channel Select bits operate as follows: CS3 CS2 CS1 CS0 CH# 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 No ADC Channel selected 0 1 1 1 X No ADC Channel selected 1 0 0 0 Temp Monitor 1 0 0 1 DAC0 1 0 1 0 DAC1 1 0 1 1 AGND 1 1 0 0 VREF

Table 7. ADCCON2 SFR bit designations

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 Value : 00H

BUSY	GNCLD	AVGS1	AVGS0	CALCLK	MODCAL	TYPCAL	SCAL

Bit Location	Bit Mnemonic	Description
ADCCON3.7	BUSY	The ADC busy status bit BUSY is a read only status bit that is set during a valid ADC conversion or calibration cycle. Busy is automatically cleared by the core at the end of a conversion or calibration.
ADCCON3.6	GNCLD	Gain Calibration disable bit. Set to 0 to enable Gain calibration Set to 1 to disable Gain calibration
ADCCON3.5 ADCCON3.4	AVGS1 AVGS0	Number of Averages selection bits. This bit selects the number of ADC readings averaged during a calibration cycle. AVGS1 AVGS0 Number of Averages 0 0 15 0 1 1 1 0 31 1 1 63
ADCCON3.3	CALCLK	Calibration Clock Divide bit. This bit selects the ratio of the Calibration Clock to the ADCCLK. Set to 0 for ADCCLK / 1 Set to 1 for ADCCLK / 2
ADCCON3.2	MODCAL	Calibration Mode select bit. This bit selects between internal (Device) and external (System) voltages when a calibration cycle is being performed. Set to 0 for Device Calibration Set to 1 for System Calibration
ADCCON3.1	TYPECAL	Calibration Type select bit. This bit selects between Offset (zeroscale) and gain (fullscale) calibration. Set to 0 for Offset calibration Set to 1 for Gain calibration
ADCCON3.0	SCAL	Start Calibration Cycle bit. When set, this bit starts the selected calibration cycle. It is automatically cleared when the calibration cycle is completed.

Table 13. ADCCON3 SFR bit designations

DRIVING THE A/D CONVERTER

The ADC incorporates a successive approximation (SAR) architecture involving a charge-sampled input stage. Figure 17 shows the equivalent circuit of the analog input section. Each ADC conversion is divided into two distinct phases as defined by the position of the switches in Figure 17. 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.

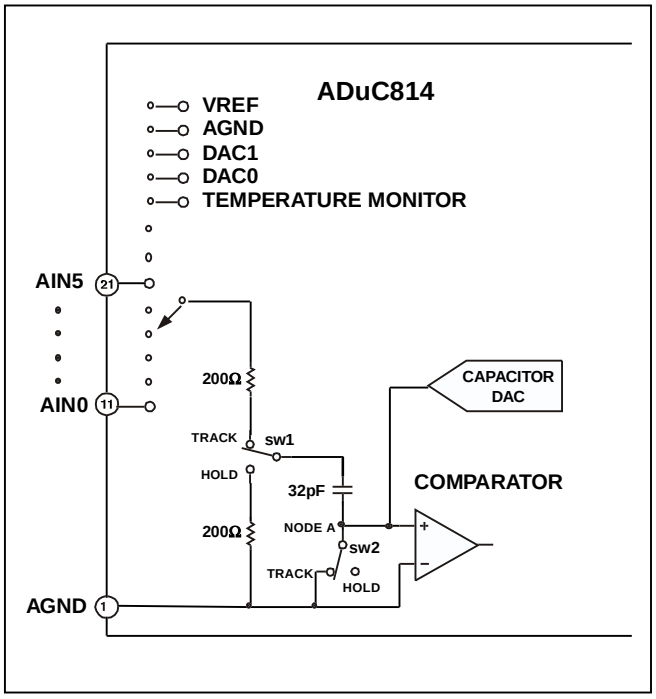


Figure 17. Internal ADC structure

Note that whenever a new input channel is selected, a residual charge from the 32pF 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 will alleviate this burden from the software design task and will ultimately result in a cleaner system implementation. One hardware solution would be 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 300ns in order to guarantee adequate settling under all software configurations. A better solution, recommended for use with any amplifier, is shown in Figure 18.

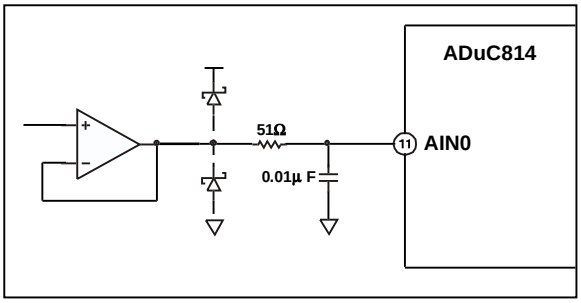


Figure 18. Buffering Analog Inputs

Though at first glance the circuit in Figure 18 may look like a simple anti-aliasing filter, it actually serves no such purpose since its corner frequency is well above the Nyquist frequency, even at a 200kHz sample rate. 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 32pF sampling capacitor can draw its charge. Since the 0.01µF capacitor in Figure 18 is more than 312 times the size of the 32pF sampling capacitor, its voltage will not change by more than one count (1 / 312) of the 12-bit transfer function when the 32pF charge from a previous channel is dumped onto it. A larger capacitor can be used if desired, but not a larger resistor (for reasons described below).

The Schottky diodes in Figure 18 may be necessary to limit the voltage applied to the analog input pin as per the datasheet absolute maximum ratings. They are not necessary if the op amp is powered from the same supply as the ADuC814 since in that case the op amp is unable to generate voltages above VDD 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's analog inputs can cause measurable DC errors with external source impedances as little as 100Ω or so. To ensure accurate ADC operation, keep the total source impedance at each analog input less than 61Ω. Table 1 illustrates examples of how source impedance can affect DC accuracy.

Table 1: Source Impedance Errors

Source Impedance	Error from 1µA Leakage Current	Error from10µA Leakage Current
61Ω	61µV = 0.1 LSB	610µV = 1 LSB
610Ω	610µV = 1 LSB	6.1mV = 10 LSB

Although Figure 18 shows the op amp operating at a gain of 1, you can of course 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 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 2. Check Analog Devices literature (CD ROM data book, etc.) for details on these and other op amps and instrumentation amps.

Table VI – Some single-supply op amps

Op Amp Model	Characteristics
OP181/281/481	micropower
OP191/291/491	I/O good up to V_{DD}, low cost
OP196/296/496	I/O to V_{DD}, micropwr, low cost
OP183/283	high gain-bandwidth product
OP162/262/462	high GBP, micro package
AD820/822/824	FET input, low cost
AD823	FET input, high GBP

Keep in mind that the ADC's transfer function is 0 to V_{REF}, and any signal range lost to amplifier saturation near ground will impact dynamic range. Though the op amps in Table 2 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, you might consider using it to power the front-end amplifiers. If you do, however, be sure to include the Schottky diodes shown in Figure 18 (or at least the lower of the two diodes) to protect the analog input from under-voltage conditions. To summarize this section, use the circuit of Figure 18 to drive the analog input pins of the ADuC814.

VOLTAGE REFERENCE CONNECTIONS

The on-chip 2.5V bandgap voltage reference can be used as the reference source for the ADC and DACs. In order to ensure the accuracy of the voltage reference you must decouple the V_{REF} pin to ground with 10µF and 0.1µF capacitors, and the C_{REF} pin to ground with 0.1µF capacitors as shown in Figure 19.

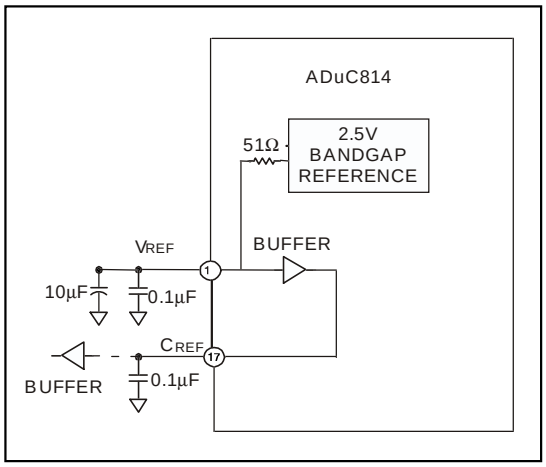


Figure 19. Decoupling V_{REF} and C_{REF}.

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.

The ADuC814 powers up with its internal voltage reference in the "on" state as some of the circuitry is used by the on-chip PLL.

If an external voltage reference is preferred, it should be connected to the V_{REF} and C_{REF} pins as shown in Figure 20. 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 1.8V 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}.

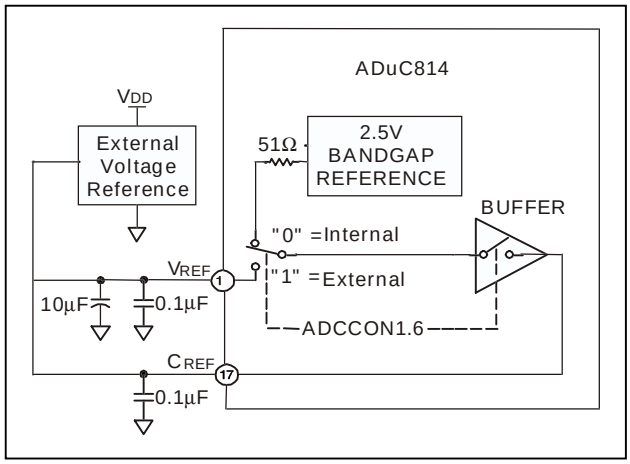


Figure 20. Using an external Voltage Reference.

Operation of the ADC or DACs with a reference voltage below 1.8V, 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 1.8V.

CONFIGURING THE ADC

In configuring the ADC a number of parameters need to be setup. These parameters can be configured using the three SFR'S ADCCON1, ADCCON2, and ADCCON3 and are detailed below.

The ADCCLK determines the speed at which the ADC logic runs while performing an ADC conversion. All ADC timing parameters are calculted from the ADCCLK frequency. On the ADuC814, the ADCCLK is derived from the maximum core frequency (Fcore), 16.777216Mhz. The ADCCLK frequency is selected via ADCCON1 bits 5 and 4. These 2 bits provide 4 core clock divide ratios of 8, 4, 16, and 32, which provide ADCCLK values of 2MHz, 4MHz, 1MHz and 500KHz respectively.

The Acquisition time (Tacq) is the number of ADCCLKs that the ADC input circuitry uses to sample the input signal. In nearly all cases, an acquisition time of 1 ADCCLK will provide more than adequate time for the ADuC814 to acquire its signal before switching the internal track&hold amplifier into hold mode. The only exception would be a high source impedance analog input, but these should be buffered first anyway since source impedances of greater than 610Ω can cause DC errors as well. ADCCON1 bits 3 and 2 are used to select acqusition 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 (Tconv).
When a conversion is initiated this start-of-conversion signal is synchronised to the ADCCLK. This synchronisation (Tsync) may take from 0.5 to 1.5 ADCCLKs to occur. The total ADC conversion time Tadc is calculated using the following formula...

Tadc = Tsync + Tacq + Tconv

Assuming Tsync = 1, Tacq = 1 and Fcore/ADCCLK divider of 8. The total conversion time...

Tadc = (1 + 1 + 15) * (1 / 2097152)

Tadc = 8.11 µS

These settings would allow a maximum conversion mode rate of 123.361kHz.

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 will cause the ADC to perform a single conversion and put the result into the ADCDATAH/L SFRs. The SCONV bit will get 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 will perform back-to-back conversions at the configured rate (123.361KHz for the settings detailed previously.

In this 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 pin7 ($\overline{\text{CONVST}}$). This active low pulse should be at least 100nS wide. The rising edge of this signal initiates the conversion.
Timer 2 can also be used to initiate converions. Setting bit 1 of ADCCON1 enables the Timer 2 overflow signal to start a conversion. For Timer 2 configuration information, see the Timers section later in this datasheet.

For both external $\overline{\text{CONVST}}$ and Timer 2 overflow, the conversion rate must be greater than the conversion time (Tadc).

ADC HIGH SPEED DATA CAPTURE MODE.

The on-chip ADC has been designed to run at a maximum conversion speed of 4.05µS (247kHz sampling rate). When converting at this rate the ADuC814 micro has 4.05µS to read the ADC result and store the result in memory for further post processing otherwise the next ADC sample could be lost. In an interrupt driven routine the micro would also have to jump to the ADC Interrupt Service routine which will also increase the time required to store the ADC results. In applications where the ADuC814 cannot sustain the interrupt rate, an ADC High Speed Data Capture (HSDC) mode is provided.
To enable HSDC mode, bit 6 in ADCCON2 (ADCSPI) must be set. This allows the ADC results to be written directly to the SPI interface without any interaction from the ADuC814 core. This mode allows the ADuC814 to capture a contiguous sample stream at full ADC update rates (247kHz).

A typical HSDC Mode configuration example

To configure the ADuC814 into HSDC mode the following steps must be followed.

1. The ADC must be put into one of its conversion modes

2. The SPI Interface must be configured (SPI interface configuration is detailed later in this datasheet).

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 will be transferred from the ADCDATAH/L SFRs to the SPIDAT register. Figure 21 and 22 show the HSDC logic and timing for a single external CONVST pulse. The ADC result is transmitted most significant bit first. In this case the Channel ID will be transmitted first followed by the 12 bit ADC result. While this mode is enabled , normal SPI or port 3 operation is disabled, while the MicroConverter core is free to continue code execution, including general housekeeping and communication tasks. This mode is disabled by clearing the ADCSPI bit.

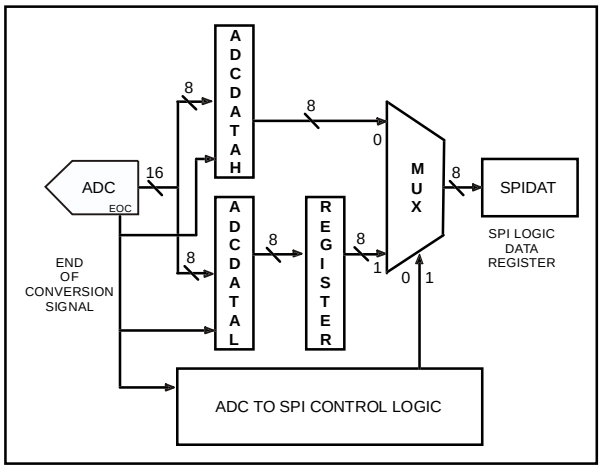


Figure 21. High Speed Data Capture Logic.

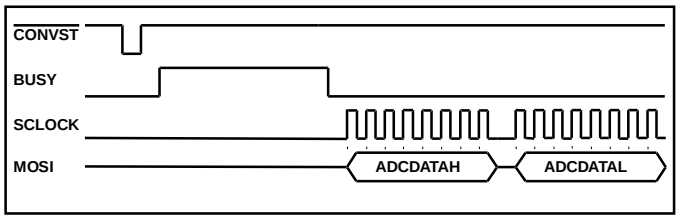


Figure 22. High Speed Data Capture Logic Timing

ADC OFFSET AND GAIN CALIBRATION OVERVIEW

The ADC block incorporates calibration hardware and associated SFR's that ensures optimum offset and gain performance from the ADC at all times.

As part of the ADuC814 internal factory final test routines, this calibration hardware is used to calibrate the 12-bit ADC to its offset and gain specifications. The offset and gain coefficients obtained from this factory calibration are stored in nonvolatile Flash memory. These are downloaded automatically on a power-up or reset event to initialize the ADC offset and gain calibration registers.

In many applications this auto-calibration download function suffices. However, the ADuC814 ADC offset and gain accuracy may vary from system to system due to board layout, grounding, clock speed, or system configuration. To get the best ADC accuracy in your system, you should perform an ADC calibration.

Two main advantages are derived from ensuring the ADC calibration registers are initialized correctly. Firstly, the internal errors in the ADC can be reduced significantly to give superior dc performance; and secondly, system offset and gain errors can be removed. This allows the user to remove reference errors (whether it be internal or external reference) and to make use of 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 are each 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 the ADC Transfer Function DOWN. Decreasing the offset coefficient compensates for negative offset, and effectively pushes the ADC Transfer Function UP. The maximum offset that can be compensated is typically ± 5% of VREF, which equates to typically ±125mV with a 2.5V 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 the ADC Transfer Function UP, effectively increasing the slope of the transfer function. Decreasing the gain coefficient, compensates for a larger analog input signal range and scales the ADC Transfer Function DOWN, effectively decreasing the slope of the transfer function. The maximum analog input signal range for which the gain coefficient can compensate is 1.025 * VREF and the minimum input range is 0.975 * VREF which equates to typically ± 2.5% of the reference voltage.

CALIBRATING THE ADC

There are two hardware calibration modes provided which can be easily initiated by user software. The ADCCON3 SFR is used to calibrate the ADC. Bit 1 (CALTYP) and bit 2 (CALMOD) setup the calibration mode. The following table indicates the mode setup by these bits.

CALMOD	CALTYP	Calibration Mode
0	0	Device Offset
0	1	Device Gain
1	0	System Offset
1	1	System Gain

Device calibration can be initiated to compensate for significant changes in operating conditions (CLK frequency, analog input range, reference voltage and supply voltages). In this calibration mode offset calibration should be executed first, followed by gain calibration. Internal AGND and VREF signals are used device calibration.

System calibration can be initiated to compensate for both internal and external system errors. To perform a system calibration using internal signals, select AGND via CD3-CD0 (1011) and perform system offset calibration. Select VREF via CD3-CD0 (1100) and perform system gain calibration.

To perform system calibration using an external reference, tie system ground and reference to any two of the the six selectable inputs. Enable external reference mode (ADCCON1.6). Select the channel connected to AGND via CD3-CD0 and perform system offset calibration. Select the channel connected to VREF via CD3-CD0 and perform system gain calibration.

Bit 3 (CALCLK) selects whether the CALCLK frequency used is ADCCLK (0) or ADCCLK/2 (1). Setting this bit will double the amount of time available to each bit decision during the calibration process. However, this also doubles the time the calibration cycle takes to complete.

Bits 5 (AVGS1) and 4 (AVGS0) select the number of ADC readings averaged (NUMAV) to calibrate each bit in the calibration coefficients.The following table indicates the number of averages selected by these bits.

AVGS1	AVGS0	NUMAV
0	0	15 (default)
0	1	1
1	0	31
1	1	63

These 2 bits along with the ADCCON1 settings will determine the time to complete an Offset or Gain calibration cycle. This time Tcal can be derived using the following formula...

Tcal =
14 × ADCCLK × 2^CALCLK × NUMAV × (16+Tacq)

For an ADCCLK/Fcore divide ratio of 4, a Tacq = 1 ADCCLK, NUMAV = 15, and CALCLK = 0, the calibration cycle time...

Tcal = 14 × (1 / 4194304) × 2^0 × 15 × (16 + 1)
Tcal = 851.15 µS

Setting bit 6 (GNCLD) disables the gain calibration coefficients being used during normal ADC conversions. This disables any gain calibration previously performed on the device affecting the ADC transfer function result. Clearing bit 6 will allow the coefficients in the ADCGAINH/L SFRs registers to be used when calculating the ADC result.

As the ADCCON3 SFR is a byte accessable register all of the above parameters have to be set up at the same time. When writing the parameters to the ADCCON3 SFR bit 0 (SCAL) must be set to 1 to initiate a calibration cycle.

The ADC Busy Flag (bit 7), instead of framing an individual ADC conversion as in normal mode, will go high at the start of calibration and only return to zero at the end of the calibration cycle. It can therefore be monitored in code to indicate when the calibration cycle is completed.

NONVOLATILE 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 is a relatively recent type of nonvolatile memory technology and is based on a single transistor cell architecture.

This technology is basically an outgrowth of EPROM technology and was developed through the late 1980s. 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 23).

Because Flash/EE technology is based on a single transistor cell architecture, a Flash memory array, like 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.

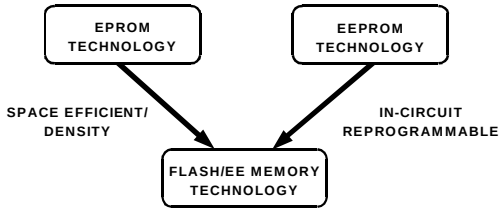


Figure 23. Flash/EE Memory Development

Overall, Flash/EE memory represents a step closer to the ideal memory device that includes nonvolatility, in-circuit programmability, high density and low cost. Incorporated in the ADuC814, Flash/EE memory technology allows the user to update program code space in-circuit, without the need to replace onetime 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. 8K bytes of Flash/EE Program space are provided on-chip to facilitate code execution without any external discrete ROM device requirements. 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, namely 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. These events are defined as:

a. Initial page erase sequence
b. Read/verify sequence
c. Byte program sequence
d. Second read/verify sequence

A single Flash/EE Memory Endurance Cycle

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

As indicated in the specification pages of this data sheet, 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 +85°C. The results allow the specification of a minimum endurance figure over supply and 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 also be noted that retention lifetime, based on an activation energy of 0.6 eV, will derate with T_J as shown in Figure 24.

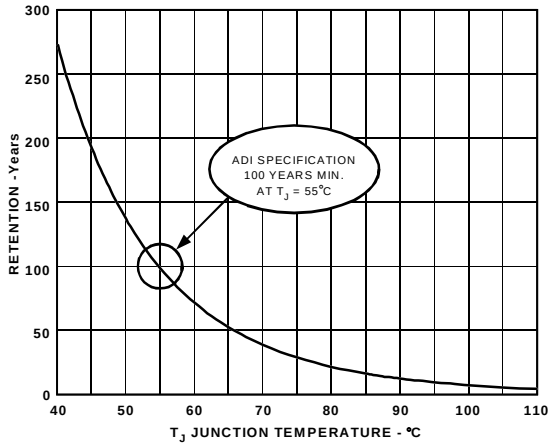


Figure 24. Flash/EE Memory Data Retention

Using the Flash/EE Program Memory

The 8 Kbyte Flash/EE Program Memory array is mapped into the lower 8 Kbytes of the 64 Kbytes program space addressable by the ADuC814, and is used to hold user code in typical applications.

The program memory Flash/EE memory arrays can be programmed in one of two modes, namely:

Serial Downloading (In-Circuit Programming)

As part of its factory boot code, the ADuC814 facilitates serial code download via the standard UART serial port. Serial download mode is automatically entered on power-up if the external pin, DLOAD, is pulled high through an external resistor as shown in Figure 25. 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.

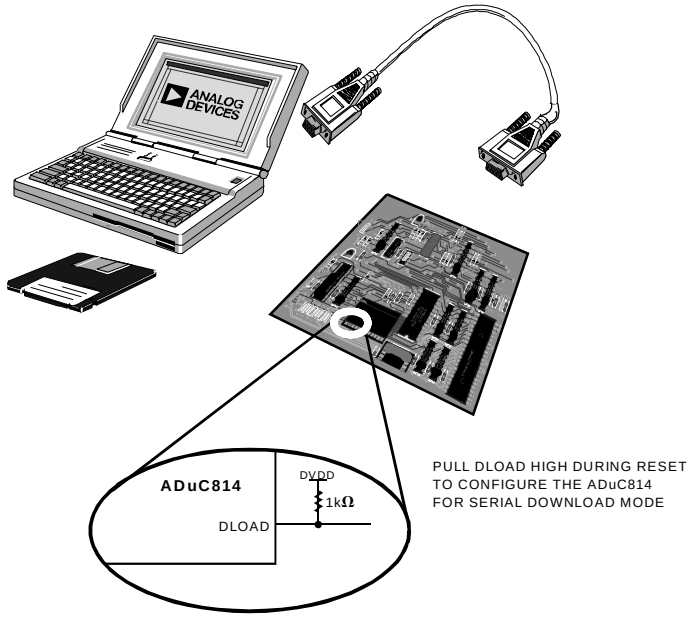


Figure 25. Flash/EE Memory Serial Download Mode Programming

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 26.

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.

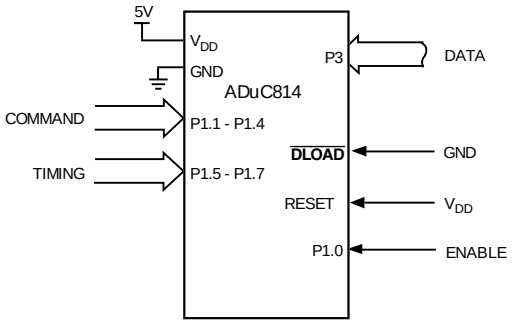


Figure 26. Flash/EE Memory Parallel Programming

Flash/EE Program Memory Security

The ADuC814 facilitates three modes of Flash/EE program memory security. 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 web page at www.analog.com/microconverter. The security modes available on the ADuC814 are described as follows:

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 will interpret the serial download reset as a normal reset only. It will therefore not enter serial download mode but only execute a normal reset sequence. Serial Safe mode can only be disabled by initiating a code-erase command in parallel programming mode.

Using the 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 27.

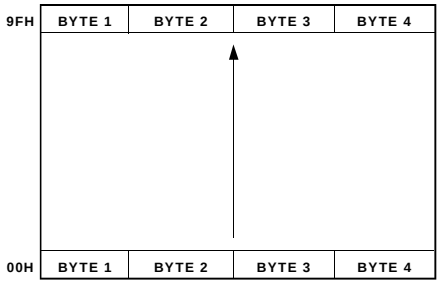


Figure 27. Flash/EE Data Memory Configuration

As with other ADuC814 user-peripheral circuits, the interface to this memory space is via a group of registers mapped in the SFR space. A group of four data registers (EDATA1–4) are used to hold 4-byte page data just accessed. EADRL is used to hold the 8-bit address of the page to be 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
EDATA 1–4:		
	SFR Address:	BCH to BFH respectively
	Function:	Holds Flash/EE Data memory page write or page read data bytes.
	Default :	EDATA1–2 –> 00H EDATA3–4 –> 00H

A block diagram of the SFR interface to the Flash/EE Data Memory array is shown in Figure 28.

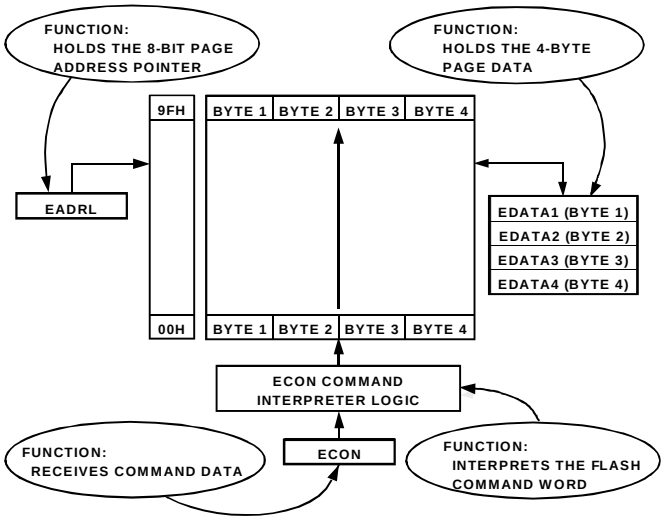


Figure 28. Flash/EE Data Memory Control and Configuration

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 XIII:

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

Command Byte	Command Mode
01H	READ COMMAND. Results in four bytes being read into EDATA1–4 from memory page address contained in EADRL.
02H	PROGRAM COMMAND. Results in four bytes (EDATA1–4) being written to memory page address in EADRL. This write command assumes the designated "write" page has been pre-erased.
03H	RESERVED FOR INTERNAL USE. 03H should not be written to the ECON SFR.
04H	VERIFY COMMAND. 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 will result in a "zero" being read if the verification is valid, a nonzero value will be read to indicate an invalid verification.
05H	ERASE COMMAND. Results in an erase of the 4-byte page designated in EADRL.
06H	ERASE-ALL COMMAND. Results in erase of the full Flash/EE Data memory 160-page (640 bytes) array.
07H to FFH	RESERVED COMMANDS. Commands 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 will involve 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 will not be written for read accesses) and finally, writing the ECON command word which initiates one of the six modes shown in Table XIII.

It should be noted that a given mode of operation is initiated as soon as the command word is written to the ECON SFR. The core microcontroller 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 will not be executed until the Flash/EE operation is complete (250 µs or 2 ms later). This means that the core will not respond to Interrupt requests until the Flash/EE operation is complete, though the core peripheral functions like Counter/Timers will continue to count and time as configured throughout this period.

Erase-All
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 erase of all 640 byte locations in the Flash/EE array. This command coded in 8051 assembly would appear as:

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

Program a Byte
In general terms, a byte in the Flash/EE array can only be programmed if it has previously been erased. 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) will be 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. As the user is only required to modify 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, would appear 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

The following 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.

DAC

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} (the internal bandgap 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 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. It should be noted that in 12-bit mode, the DAC voltage output will be 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.

DACCON	DAC Control Register
SFR Address	FDH
Power-On Default Value	00H
Bit Addressable	No

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

Table XIV. DACCON SFR Bit Designations

Bit	Name	Description
7	MODE	Mode Select bit. Selects between 12-bit and 8-bit mode for both DACs Set to "1" = 8-bit mode (Write 8 bits to DACxL SFR Set to "0" = 12-bit mode
6	RNG1	DAC1 Output Voltage range select bit. Set to "1" = Configure DAC1 range of 0 – AV_{DD}. Set to "0" = Configure DAC1 range of 0 – 2.5 V.
5	RNG0	DAC0 Output Voltage range select bit. Set to "1" = Configure DAC0 range of 0 – AV_{DD}. Set to "0" = Configure DAC0 range of 0 – 2.5 V.
4	$\overline{CLR1}$	DAC1 Clear Bit. Set to "1" = Normal DAC1 operation. Set to "0" = Force DAC1 Output Voltage to zero volts.
3	$\overline{CLR0}$	DAC0 Clear Bit. Set to "1" = Normal DAC0 operation. Set to "0" = Force DAC0 Output Voltage to zero volts. Cleared by user to operate the DAC in its normal 12-bit mode of operation.
2	SYNC	DAC0/1 Update Synchronization Bit. Set to "1" = Asynchronous Update Mode. The DAC outputs update as soon as the DACxL SFRs are written. Set to "0" = Synchronous Update Mode. The user can simultaneously update both DACs by first updating the DACxH/L SFRs while SYNC is "0". Both DAC will then update simultaneously when the SYNC bit is set to "1".
1	$\overline{PD1}$	DAC1 Power-Down Bit. Set to "1" = Power Up DAC1. Set to "0" = Power Down DAC1.
0	$\overline{PD0}$	DAC0 Power-Down Bit. Set to "1" = Power Up DAC0. Set to "0" = Power Down DAC0.

DACxH/L	DAC0 and DAC1 Data Registers
Function	DAC Data Registers, written by user to update the DAC outputs.
SFR Address	DAC0L (DAC0 Data Low Byte)–>F9H DAC0L (DAC0 Data Low Byte)–>FBH DAC1H (DAC1 Data High Byte)–>FAH DAC1H (DAC1 Data High Byte)–>FCH
Power-On Default Value	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 D/A Converter

The on-chip D/A converter architecture consists of a resistor string DAC followed by an output buffer amplifier, the functional equivalent of which is illustrated in Figure 29. Features of this architecture include inherent guaranteed monotonicity and excellent differential linearity.

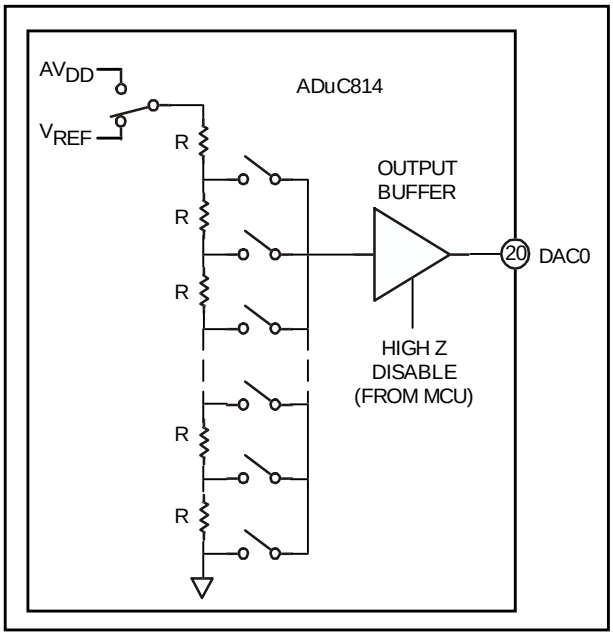


Figure 29 - Resistor String DAC Functional Equivalent

As illustrated in Figure 29, the reference source for each DAC is user selectable in software. It can be either AV_{DD} or $V_{REF.}$ In 0-to-AV_{DD} mode, the DAC output transfer function spans from 0V to the voltage at the AV_{DD} pin. In 0-to-V_{REF} mode, the DAC output transfer function spans from 0V to the internal V_{REF} or if an external reference is applied the voltage at the V_{REF} pin.The DAC output buffer amplifier features a true rail-to-rail output stage implementation. This means that, unloaded, each output is capable of swinging to within less than 100mV of both AV_{DD} and ground. Moreover, the DAC's linearity specification (when driving a 10KΩ resistive load to ground) is guaranteed through the full transfer function *except* codes 0 to 48, and, in 0-to-AV_{DD} mode only, codes 3945 to 4095. Linearity degradation near ground and V_{DD} is caused by saturation of the output amplifier, and a general representation of its effects (neglecting offset & gain error) is illustrated in Figure 30. The dotted line in Figure 19 indicates the *ideal* transfer function, and the solid line represents what the transfer function might look like with endpoint non-linearities due to saturation of the output amplifier. Note that Figure 30 represents a transfer function in 0-to-V_{DD} mode only. In 0-to-V_{REF} mode (with $V_{REF} < V_{DD}$) the lower non-linearity 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 endpoint linearity errors.

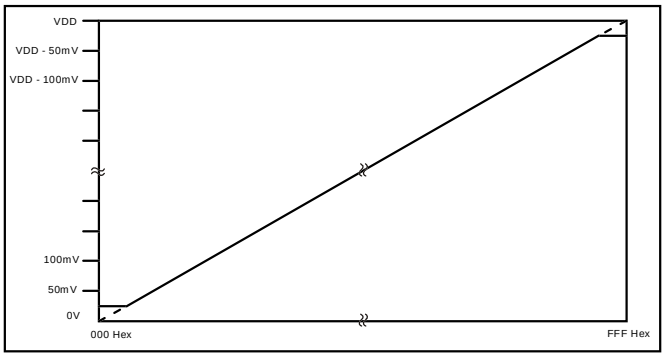


Figure 30 - Endpoint Non-linearities due to Amplifier Saturation.

The endpoint non-linearities conceptually illustrated in Figure 30 get worse as a function of output loading. Most of the ADuC814's datasheet specifications assume a 10KΩ 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 30 become larger. With larger current demands, this can significantly limit output voltage swing. Figure 31 & Figure 32 illustrate this behavior. It should be noted that the upper trace in each of these figures is only valid for an output range selection of 0-to-AV_{DD}. In 0-to-V_{REF} mode, DAC loading will 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}=3V & V_{REF}=2.5V, the high-side voltage will not be affected by loads less than 5mA. But somewhere around 7mA the upper curve in Figure 32 drops below 2.5V (V_{REF}) indicating that at these higher currents the output will not be capable of reaching V_{REF}.

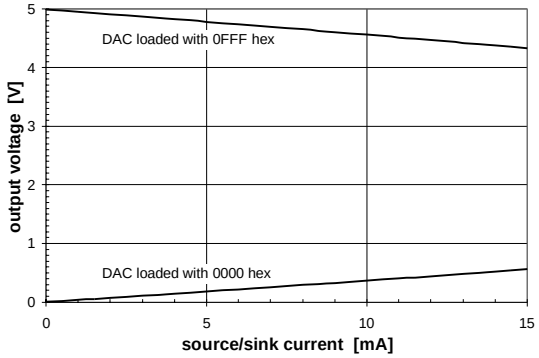


Figure 31 - Source & Sink Current Capability with V_{REF} = V_{DD} = 5V

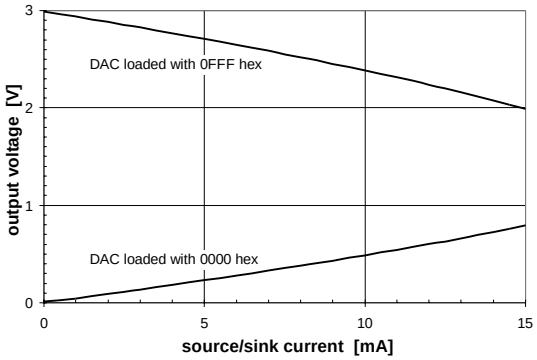


Figure 32 - Source & Sink Current Capability with $V_{REF} = V_{DD} = 3V$

For larger loads the current drive capability may not be sufficient. In order to increase the Source & Sink current capability of the DACs an external buffer should be added, as shown in figure 33.

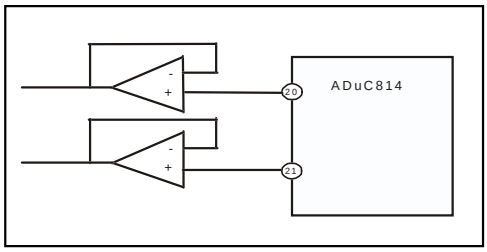


Figure 33 - Buffering the DAC outputs

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 "tri-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 will 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. A PLL locks onto a multiple (512) of this 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 or 2.097152 MHz. The PLL is controlled via the PLLCON special function register.

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

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

Table XV. PLLCON SFR Bit Designations

Bit	Name	Description
7	OSC_PD	Oscillator Power-down Bit. Set by user to halt the 32 kHz oscillator in power-down mode. Cleared by 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. Set automatically at power-on to indicate the PLL loop is correctly tracking the crystal clock. If the external crystal becomes subsequently disconnected the PLL will rail and the core will halt. Cleared automatically at power-on to indicate 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 will be 16.78 MHz ± 20%.
5	---	Reserved for future use; should be written with '0.'
4	---	Reserved for future use; should be written with '0.'
3	FINT	Fast Interrupt Response Bit. Set by user enabling the response to any interrupt to be executed at the fastest core clock frequency, regardless of the configuration of the CD2–0 bits (see below). Once user code has returned from an interrupt, the core resumes code execution at the core clock selected by the CD2–0 bits. Cleared by 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 will operate.
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 the PLL and thus has the ability to remain active in power-down mode and 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 will clock 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 Interrupt System later in this data sheet.) If the ADuC814 is in power-down mode, again with TIC interrupt enabled, the TII bit will wake up the device and resume code execution by vectoring directly to the TIC interrupt service vector address at 0053 hex. The TIC-related SFRs are described in Table XVI. Note also that the timebase SFRs can be written initially with the current time, the TIC can then be controlled and accessed by user software. In effect, this facilitates the implementation of a real-time clock. A block diagram of the TIC is shown in Figure 34.

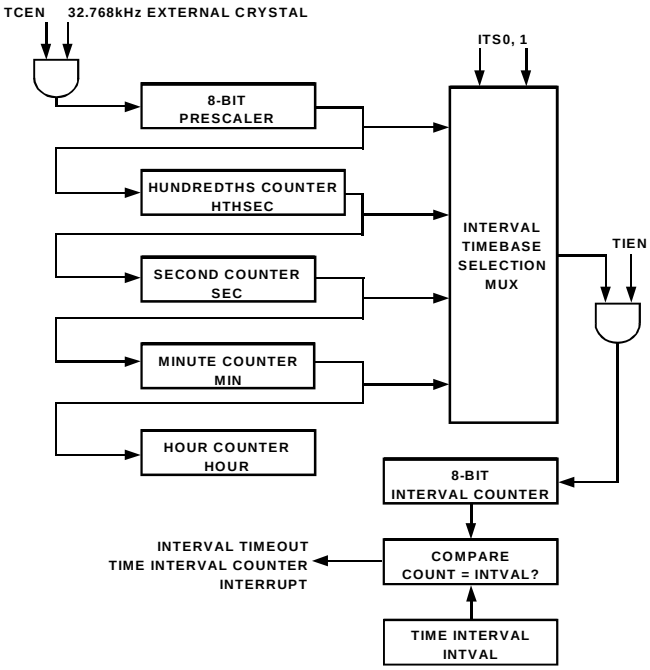


Figure 34. TIC, Simplified Block Diagram

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

----	TFH	ITS1	ITS0	STI	TII	TIEN	TCEN

Table XVI. TIMECON SFR Bit Designations

Bit	Name	Description
7	---	Reserved for Future Use.
6	TFH	Twenty four hour Select bit. Set by user to enable the HOUR counter to count from 0 to 23. Cleared by user to enable the HOUR counter to count from 0 to 255.
5	ITS1	Interval Timebase Selection Bits.
4	ITS0	Written by user to determine the interval counter update rate. ITS1 ITS0 Interval Timebase 0 0 1/128 Second 0 1 Seconds 1 0 Minutes 1 1 Hours
3	STI	Single Time Interval Bit. Set by user to generate a single interval time-out. If set, a timeout will clear the TIEN bit. Cleared by user to allow the interval counter to be automatically reloaded and start counting again at each interval timeout.
2	TII	TIC Interrupt Bit. Set when the 8-bit Interval Counter matches the value in the INTVAL SFR. *Cleared* by user software.
1	TIEN	Time Interval Enable Bit. Set by user to enable the 8-bit time interval counter. Cleared by user to disable and clear the contents of the interval counter.
0	TCEN	Time Clock Enable Bit. Set by user to enable the time clock to the time interval counters. Cleared by user to disable the clock to the time interval counters and clear the time interval SFRs. 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) bit is set and generates an interrupt if enabled. (See IEIP2 SFR description under Interrupt System later in this data sheet.)
SFR Address	A6H
Power-On Default Value	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 Value	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 Value	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 counts from 0 to 59 before rolling over to increment the HOUR time register.
SFR Address	A4H
Power-On Default Value	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 Value	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 will generate 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 time-out 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 only be written by user software if the double write sequence described in WDWR below is initiated on every write access to the WDCON SFR.

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

PRE3	PRE2	PRE1	PRE0	WDIR	WDS	WDE	WDWR

Table XVII. WDCON SFR Bit Designations

Bit	Name	Description
7	PRE3	Watchdog Timer Prescale Bits.
6	PRE2	The Watchdog timeout period is given by the equation: $t_{WD} = (2^{PRE} \times (2^9/f_{PLL}))$
5	PRE1	(0 - PRE - 7; f_{PLL} = 32.768 kHz)
4	PRE0	PRE3 PRE2 PRE1 PRE0 Time-out 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 Response Enable Bit. If this bit is set by the user, the watchdog will generate 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 will be generated. (See also Note 1, Table XXXIV 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 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 will generate 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 the very next instruction must be a write instruction to the WDCON SFR.
		e.g., `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 (DVDD) on the ADuC814. It will indicate 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 will interrupt the core using the PSMI bit in the PSMCON SFR. This bit will not be 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 will not resume until a safe supply level has been 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 Value	DEH
Bit Addressable	No

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

Table XVIII. PSMCON SFR Bit Designations

Bit	Name	Description
7	PSMCON.7	Reserved for future use.
6	CMPD	DVDD Comparator Bit. This is a read-only bit and directly reflects the state of the DVDD comparator. Read '1' indicates the DVDD supply is above its selected trip point. Read '0' indicates the DVDD supply is below its selected trip point.
5	PSMI	Power Supply Monitor Interrupt Bit. This bit will be 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 and/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	DVDD Trip Point Selection Bits.
3	TPD0	These bits select the DVDD trip-point voltage as follows: TPD1 TPD0 Selected DVDD Trip Point (V) 0 0 4.63 0 1 3.08 1 0 2.93 1 1 2.63
2	PSMCON.2	Reserved for future use.
1	PSMCON.1	Reserved for future use.
0	PSMEN	Power Supply Monitor Enable Bit. Set to '1' by the user to enable the Power Supply Monitor Circuit. Cleared to '0' by the user to disable the Power Supply Monitor Circuit.

ADUC814 CONFIGURATION REGISTER (CFG812S)

The ADuC814 is housed in a 28 pin 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 Config SFR, CFG812S, located at SFR address 9CH. A summary of these functions is described below and a detailed bit designation for the CFG812S SFR is also outlined below.

Serial Interfaces

The SPI and the I2C compatible interfaces on the ADuC814 share the same pins. 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 SPI and I2C compatible sections later in this data sheet. On the ADuC814, these serial interfaces also multiplexed with P3.5, P3.6 and P3.7. By default, these pins operate as standard port 3 pins. Bit 0 of the CFG812S SFR must be set to 1 to enable these serial interfaces onto 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 CFG812S SFR. When selected, this external clock bypasses the PLL and is used as the clock for the device, therefore, allowing the ADuC814 to be synchronised to the rest of the application system. The maximum input frequency of this external clock is 16.777216MHz. As can be seen in figure 35 below, if selected, the EXTCLK signal will affect 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. When using this feature, it is up to the user to ensure that timing critical parameters of each peripheral used are met.

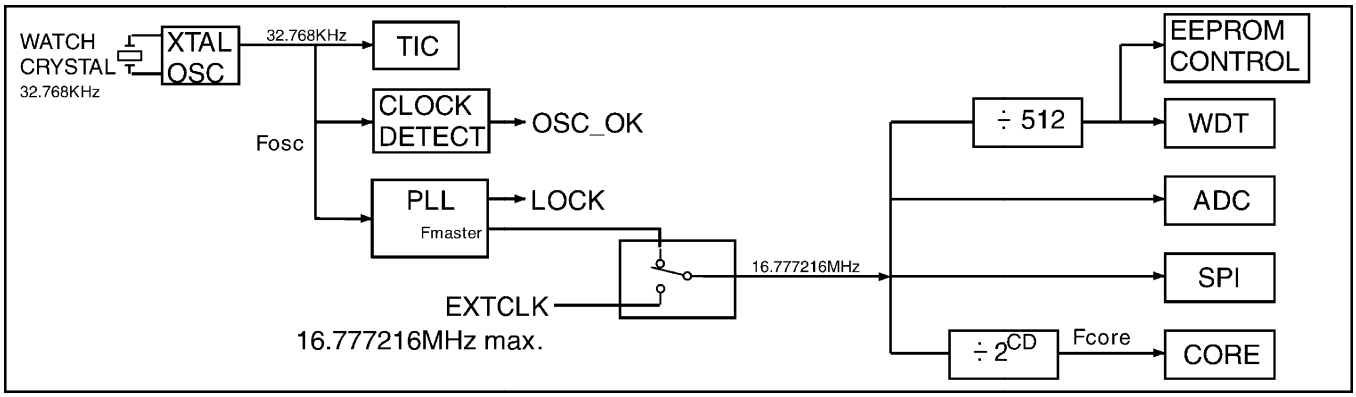


Figure 35. ADuC814 Clock Distribution

CFG812S: ADuC814 Config Register

SFR Address 9CH
Power-On Default Value 04H
Bit Addressable No

						EXTCLK	SER_EN

CFG812S SFR Bit Designations

Bit	Name	Description
1	EXTCLK	External Clock Selection Bit. Set to 1 to enable EXTCLK as MCU core clock. Cleared to 0 enable XTAL + PLL as the MCU core clock.
0	SER_EN	Serial Interface Enable Bit. Set to 1 to enable the SPI and I2C compatible interfaces onto the P3.5, P3.6 and P3.7 pins. Cleared to 0 to enable standard Port 3 functionality on P3.5, P3.6 and P3.7.

SERIAL PERIPHERAL INTERFACE

The ADuC814 integrates a complete hardware Serial Peripheral Interface (SPI) interface 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. It should be noted that the SPI physical interface is shared with the I^2C interface and therefore the user can only enable one or the other interface at any given time (see SPE in SPICON below). The system can be configured for Master or Slave operation and typically consists of four pins, namely:

MISO (Master In, Slave Out Data I/O Pin), Pin#23

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

MOSI (Master Out, Slave In Pin), Pin#24

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

SCLOCK (Serial Clock I/O Pin), Pin#25

The master clock (SCLOCK) 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 XIX below). In slave mode the SPICON register will have to be configured with the phase and polarity (CPHA and CPOL) of the expected input clock. In both master and slave mode 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), Pin#22

The Slave Select ($\overline{SS}$) input pin is only used when the ADuC814 is configured in slave mode to enable the SPI peripheral. This line is active low. Data is only received or transmitted in slave mode 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 then 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 (Pin# 22), 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 Value	04H
Bit Addressable	Yes

ISPI	WCOL	SPE	SPIM	CPOL	CPHA	SPR1	SPR0

Table XIX. SPICON SFR Bit Designations

Bit	Name	Description
7	ISPI	SPI Interrupt Bit. Set by MicroConverter at the end of each SPI transfer. Cleared directly by user code or indirectly by reading the SPIDAT SFR
6	WCOL	Write Collision Error Bit. Set by MicroConverter if SPIDAT is written to while an SPI transfer is in progress. Cleared by user code.
5	SPE	SPI Interface Enable Bit. Set by user to enable the SPI interface. Cleared by user to enable the I^2C interface.
4	SPIM	SPI Master/Slave Mode Select Bit. Set by user to enable Master Mode operation (SCLOCK is an output). Cleared by user to enable Slave Mode operation (SCLOCK is an input).
3	CPOL	Clock Polarity Select Bit. Set by user if SCLOCK idles high. Cleared by user if SCLOCK idles low.
2	CPHA	Clock Phase Select Bit. Set by user if leading SCLOCK edge is to transmit data. Cleared by user if trailing SCLOCK edge is to transmit data.

Table XIX. SPICON SFR Bit Designations (continued)

Bit	Name	Description
1 0	SPR1 SPR0	SPI Bit-Rate Select Bits. These bits select the SCLOCK rate (bit-rate) in Master Mode as follows: SPR1 SPR0 Selected Bit Rate 0 0 $f_{CORE}/2$ 0 1 $f_{CORE}/4$ 1 0 $f_{CORE}/8$ 1 1 fcore/16 In SPI Slave Mode, i.e., SPIM = 0, the logic level on the external $\overline{SS}$ pin (Pin# 22), can be read via the SPR0 bit.

NOTE
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 user code to read data just received by the SPI interface.

SFR Address F7H

Power-On Default Value 00H

Bit Addressable No

Using the SPI Interface

Depending on the configuration of the bits in the SPICON SFR shown in Table XIX, the ADuC814 SPI interface will transmit or receive data in a number of possible modes. Figure 36 shows all possible ADuC814 SPI configurations and the timing relationships and synchronization between the signals involved. Also shown in this figure is the SPI interrupt bit (ISPI) and how it is triggered at the end of each byte-wide communication.

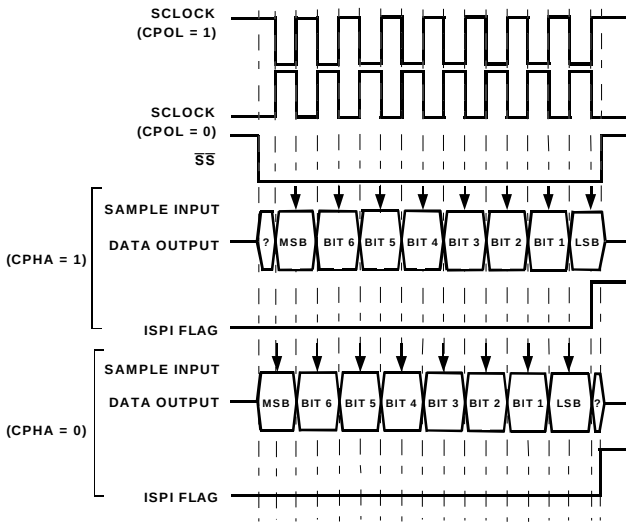


Figure 36. ADuC814, SPI Timing, All Modes

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 transmitted byte will have been completely transmitted and the input byte will be waiting in the input shift register. The ISPI flag will be set automatically and an interrupt will occur if enabled. The value in the shift register will be 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 through each input SCLOCK period. After eight clocks, the transmitted byte will have been completely transmitted and the input byte will be waiting in the input shift register. The ISPI flag will be set automatically and an interrupt will occur if enabled. The value in the shift register will be latched into SPIDAT only when the transmission/reception of a byte has been completed. The end of transmission occurs after the eighth clock has been 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 which is I²C compatible. The I²C-compatible interface shares its pins with the on-chip SPI interface and therefore the user can only enable one or the other interface at any given time (see SPE in SPICON previously). An Application Note describing the operation of this interface as implemented is available from 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 Pin
SCLOCK (Pin 25)	Serial Clock

Three SFRs are used to control the I²C-compatible interface. These are described below:

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

MDO	MDE	MCO	MDI	I2CM	I2CRS	I2CTX	I2CI

Table XX. I2CCON SFR Bit Designations

Bit	Name	Description
7	MDO	I²C Software Master Data Output Bit (MASTER MODE ONLY). This data bit is used to implement a master I²C transmitter interface in software. Data written to this bit will be outputted 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). Set by user to enable the SDATA pin as an output (Tx). Cleared by the user to enable SDATA pin as an input (Rx).
5	MCO	I²C Software Master Clock Output Bit (MASTER MODE ONLY). This data bit is used to implement a master I²C transmitter interface in software. Data written to this bit will be outputted on the SCLOCK pin.
4	MDI	I²C Software Master Data Input Bit (MASTER MODE ONLY). 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. Set by user to enable I²C software master mode. Cleared by user to enable I²C hardware slave mode.
2	I2CRS	I²C Reset Bit (SLAVE MODE ONLY). Set by user to reset the I²C interface. *Cleared* by user code for normal I²C operation.
1	I2CTX	I²C Direction Transfer Bit (SLAVE MODE ONLY). Set by the MicroConverter if the interface is transmitting. Cleared by the MicroConverter if the interface is receiving.
0	I2CI	I²C Interrupt Bit (SLAVE MODE ONLY). Set by the MicroConverter after a byte has been transmitted or received. Cleared automatically when user code reads the I2CDAT SFR (see I2CDAT below).

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

I2CDAT	I²C Data Register
Function	The I2CDAT SFR is written by the user to transmit data over the I²C interface or read by user code to read data just received by the I²C interface Accessing I2CDAT automatically clears any pending I²C interrupt and the I2CI bit in the I2CCON SFR. User software should only access I2CDAT once per interrupt cycle.
SFR Address	9AH
Power-On Default Value	00H
Bit Addressable	No

8051-COMPATIBLE ON-CHIP PERIPHERALS

This section gives a brief overview of the various secondary peripheral circuits are also available to the user on-chip. These remaining 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 capable of external memory operations; others 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 also an 8-bit port directly controlled via the P1 SFR (SFR address = 90 hex). 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 will source current because of the internal pull-ups. With 0s written to them, both these pins will drive a logic low output voltage (VOL) and will be 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 XXI.

Table XXI. Port 1, Alternate Pin Functions

Pin	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 only be configured as Analog Input (ADC), Analog Output (DAC) or Digital Input pins. By (power-on) default these pins are configured as Analog Inputs, i.e., '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 P2 SFR (SFR address = B0 hex). 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 will source current because of the internal pull-ups. Port 3 pins also have various secondary functions described in Table XXII.

Table XXII. Port 3, Alternate Pin Functions

Pin	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	$\overline{\text{INT0}}$ (External Interrupt 0)
P3.3	$\overline{\text{INT1}}$ (External Interrupt 1)
P3.4	T0 (Timer/Counter 0 External Input)
P3.5	T1 (Timer/Counter 1 External Input)
P3.6	$\overline{\text{WR}}$ (External Data Memory Write Strobe)
P3.7	$\overline{\text{RD}}$ (External Data Memory Read Strobe)

The alternate functions of P1.0, P1.1, and Port 3 pins can only be activated if the corresponding bit latch in the P1 and P3 SFRs contains a 1. Otherwise, the port pin is stuck at 0.

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 CD0–2 selection bits in the PLLCON SFR.

User configuration and control of all Timer operating modes is achieved via three SFRs namely:

TMOD, TCON: Control and configuration for Timers 0 and 1.
T2CON: Control and configuration for Timer 2.

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

GATE	C/T	M1	M0	GATE	C/T	M1	M0

Table XXIII. TMOD SFR Bit Designations

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

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

TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

NOTE
[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 XXIV. TCON SFR Bit Designations

Bit	Name	Description
7	TF1	Timer 1 Overflow Flag. Set by hardware on a timer/counter 1 overflow. Cleared by hardware when the Program Counter (PC) vectors to the interrupt service routine.
6	TR1	Timer 1 Run Control Bit. Set by user to turn on timer/counter 1. Cleared by user to turn off timer/counter 1.
5	TF0	Timer 0 Overflow Flag. Set by hardware on a timer/counter 0 overflow. Cleared by hardware when the PC vectors to the interrupt service routine.
4	TR0	Timer 0 Run Control Bit. Set by user to turn on timer/counter 0. Cleared by user to turn off timer/counter 0.
3	IE1	External Interrupt 1 ($\overline{INT1}$) Flag. Set by hardware by a falling edge or zero level being applied to external interrupt pin $\overline{INT1}$, depending on bit IT1 state. 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. Set by software to specify edge-sensitive detection (i.e., 1-to-0 transition). Cleared by software to specify level-sensitive detection (i.e., zero level).
1	IE0	External Interrupt 0 ($\overline{INT0}$) Flag. Set by hardware by a falling edge or zero level being applied to external interrupt pin $\overline{INT0}$, depending on bit IT0 state. Cleared by hardware 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.
0	IT0	External Interrupt 0 (IE0) Trigger Type. Set by software to specify edge-sensitive detection (i.e., 1-to-0 transition). Cleared by software to specify level-sensitive detection (i.e., 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 = 8Chex, 8Ahex respectively.

TH1 and TL1

Timer 1 high byte and low byte.
SFR Address = 8Dhex, 8Bhex 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 timer 0 as for timer 1.

Mode 0 (13-Bit Timer/Counter)

Mode 0 configures an 8-bit timer/counter with a divide-by-32 prescaler. Figure 37 shows mode 0 operation.

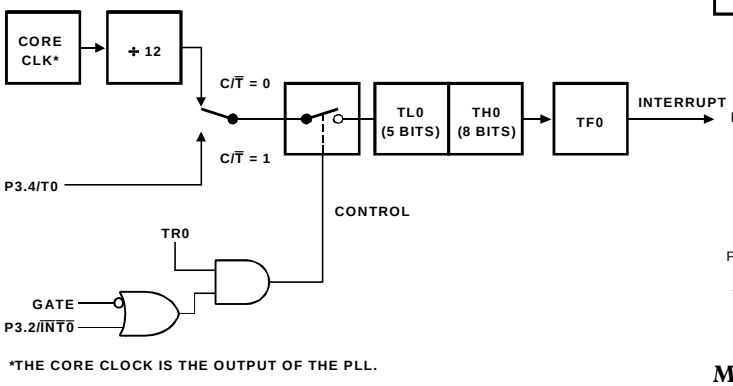


Figure 37. Timer/Counter 0, Mode 0

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 pulsewidth 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 38.

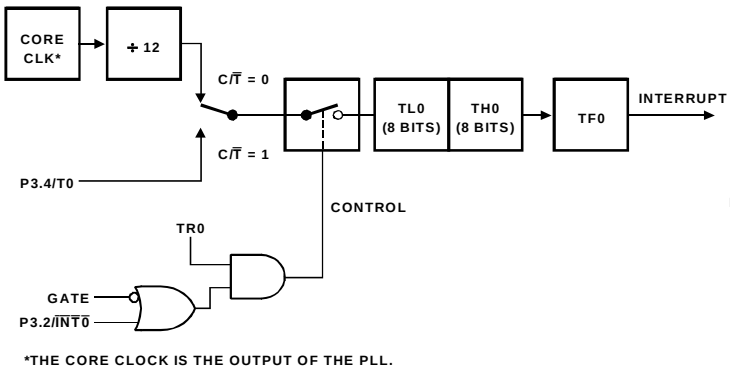


Figure 38. Timer/Counter 0, Mode 1

Mode 2 (8-Bit Timer/Counter with Auto Reload)

Mode 2 configures the timer register as an 8-bit counter (TL0) with automatic reload, as shown in Figure 39. 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.

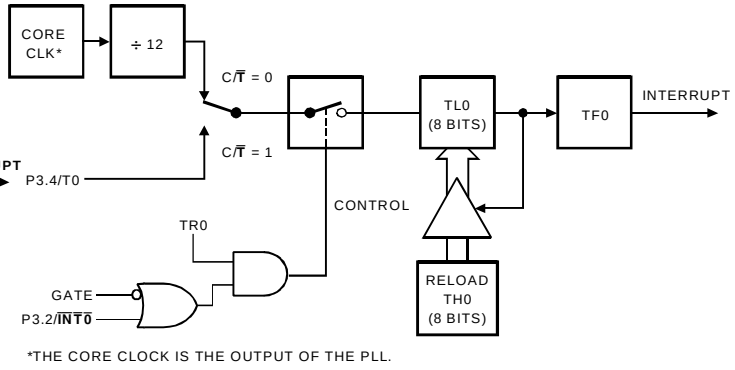


Figure 39. Timer/Counter 0, Mode 2

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 40. 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 now 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.

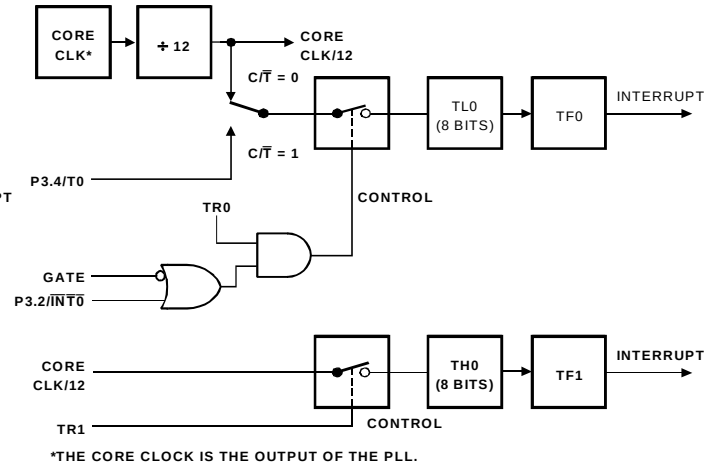


Figure 40. Timer/Counter 0, Mode 3

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

TF2	EXF2	RCLK	TCLK	EXEN2	TR2	CNT2	CAP2

Table XXV. T2CON SFR Bit Designations

Bit	Name	Description
7	TF2	Timer 2 Overflow Flag. Set by hardware on a timer 2 overflow. TF2 will not be set when either RCLK or TCLK = 1. Cleared by user software.
6	EXF2	Timer 2 External Flag. Set by hardware when either a capture or reload is caused by a negative transition on T2EX and EXEN2 = 1. Cleared by user software.
5	RCLK	Receive Clock Enable Bit. Set by user to enable the serial port to use timer 2 overflow pulses for its receive clock in serial port Modes 1 and 3. Cleared by user to enable timer 1 overflow to be used for the receive clock.
4	TCLK	Transmit Clock Enable Bit. Set by user to enable the serial port to use timer 2 overflow pulses for its transmit clock in serial port Modes 1 and 3. Cleared by user to enable timer 1 overflow to be used for the transmit clock.
3	EXEN2	Timer 2 External Enable Flag. Set by 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. Cleared by user for Timer 2 to ignore events at T2EX.
2	TR2	Timer 2 Start/Stop Control Bit. Set by user to start timer 2. Cleared by user to stop timer 2.
1	CNT2	Timer 2 timer or counter function select bit. Set by user to select counter function (input from external T2 pin). Cleared by user to select timer function (input from on-chip core clock).
0	CAP2	Timer 2 Capture/Reload Select Bit. Set by user to enable captures on negative transitions at T2EX if EXEN2 = 1. Cleared by user to enable auto-reloads with Timer 2 overflows or negative transitions at T2EX when EXEN2 = 1. 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 = CDhex, CChex respectively.

RCAP2H and RCAP2L

Timer 2, Capture/Reload byte and low byte.
SFR Address = CBhex, CAhex respectively.

Timer/Counter 2 Operating Modes

The following paragraphs describe the operating modes for timer/counter 2. The operating modes are selected by bits in the T2CON SFR as shown in Table XXVI.

Table XXVI. TIMECON SFR Bit Designations

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, which 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 will also trigger the 16-bit reload and set EXF2. The autoreload mode is illustrated in Figure 41 below.

16-Bit Capture Mode

In the 'Capture' mode, there are again two options, which 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. The Capture Mode is illustrated in Figure 42.

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 will not occur. Hence Timer 2 interrupts will not occur so they do not have to be disabled. In this mode the EXF2 flag, however, can still cause interrupts and this can be used as a third external interrupt.

Baud rate generation will be described as part of the UART serial port operation in the following pages.

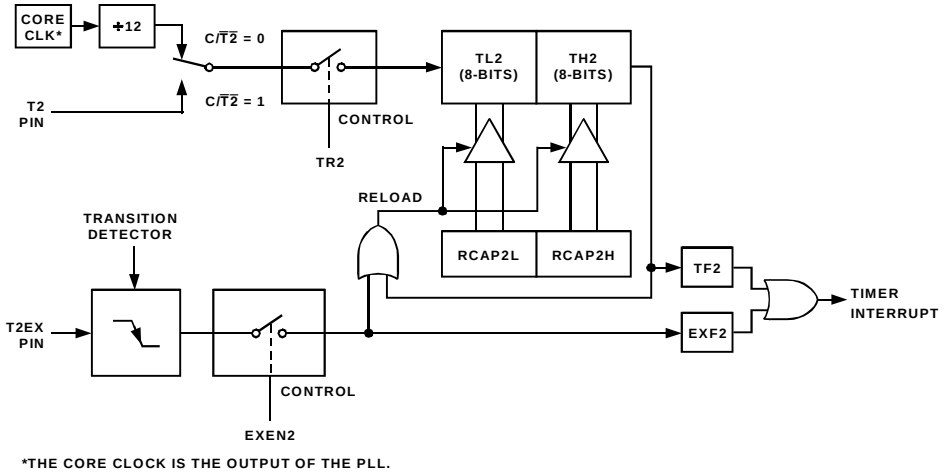


Figure 41. Timer/Counter 2, 16-Bit Autoreload Mode

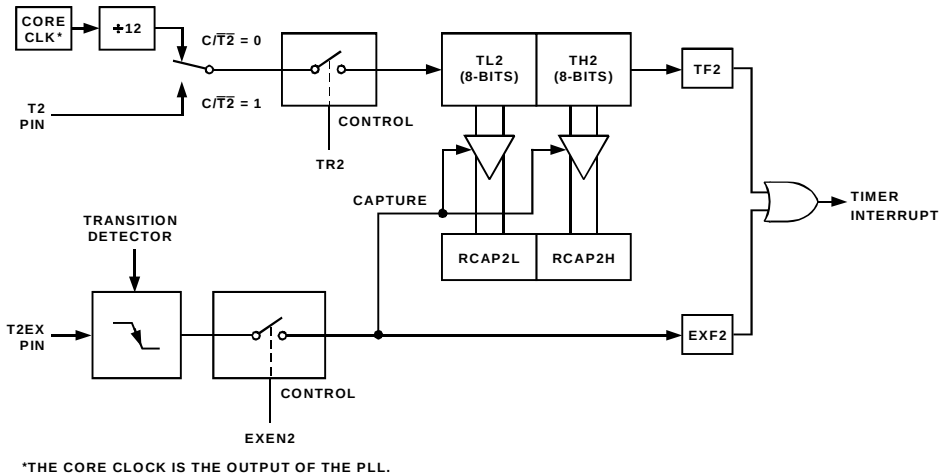


Figure 42. Timer/Counter 2, 16-Bit Capture Mode

UART SERIAL INTERFACE

The serial port is full duplex, meaning it can transmit and receive simultaneously. It is also receive-buffered, meaning it can commence reception of 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 will be 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 = 99 hex). 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 Value	00H
Bit Addressable	Yes

SM0	SM1	SM2	REN	TB8	RB8	TI	RI

Table XXVII. SCON SFR Bit Designations

Bit	Name	Description
7	SM0	UART Serial Mode Select Bits.
6	SM1	These bits select the Serial Port operating mode as follows: SM0 SM1 Selected Operating Mode 0 0 Mode 0: Shift Register, fixed baud rate (Core_Clk/2) 0 1 Mode 1: 8-bit UART, variable baud rate 1 0 Mode 2: 9-bit UART, fixed baud rate (Core_Clk/64) or (Core_Clk/32) 1 1 Mode 3: 9-bit UART, variable baud rate
5	SM2	Multiprocessor Communication Enable Bit. Enables multiprocessor communication in Modes 2 and 3. In Mode 0, SM2 should be cleared. In Mode 1, if SM2 is set, RI will not be activated if a valid stop bit was not received. If SM2 is cleared, RI will be set as soon as the byte of data has been received. In Modes 2 or 3, if SM2 is set, RI will not be activated if the received ninth data bit in RB8 is 0. If SM2 is cleared, RI will be set as soon as the byte of data has been received.
4	REN	Serial Port Receive Enable Bit. Set by user software to enable serial port reception. Cleared by user software to disable serial port reception.
3	TB8	Serial Port Transmit (Bit 9). The data loaded into TB8 will be the ninth data bit that will be transmitted in Modes 2 and 3.
2	RB8	Serial port Receiver Bit 9. 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	Serial Port Transmit Interrupt Flag. 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. TI must be cleared by user software.
0	RI	Serial Port Receive Interrupt Flag. 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. RI must be cleared by 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 43.

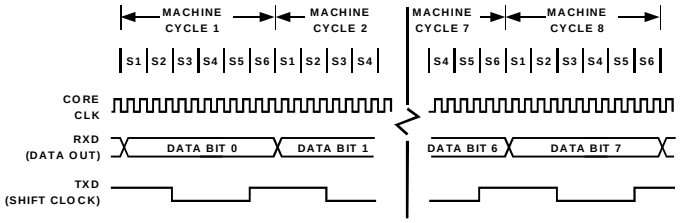


Figure 43. UART Serial Port Transmission, Mode 0.

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(0) and followed by a stop 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 44.

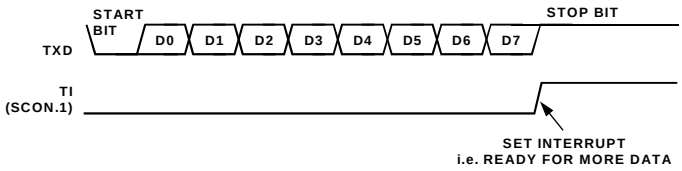


Figure 44. UART Serial Port Transmission, Mode 0.

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, and

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 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(0), eight data bits, a programmable ninth bit and a stop 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 to TB8 in SCON. When transmission is initiated the eight data bits (from SBUF) are loaded onto the transmit shift register (LSB first). The contents of TB8 are loaded into the ninth bit position of the transmit shift register. The transmission will start 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 onto 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, and

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

Mode 0 Baud Rate Generation
The baud rate in Mode 0 is fixed:

Mode 0 Baud Rate = (Core Clock Frequency[1]/12)

NOTE
[1]In these descriptions Core Clock Frequency refers to the core clock frequency selected via the CD0–2 bits in the PLLCON SFR.

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:

Mode 2 Baud Rate = $(2^{SMOD}/64) \infty$ (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:

Modes 1 and 3 Baud Rate =
$(2^{SMOD}/32) \times$ (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 = 0100Binary). In that case, the baud rate is given by the formula:

Modes 1 and 3 Baud Rate =
$(2^{SMOD}/32) \times$ (Core Clock/(12 ∞ [256-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 = 0100Binary), and using the Timer 1 interrupt to do a 16-bit software reload. Table XXVIII below, 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 XXVIII. 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 wider range of baud rates is possible using Timer 2.

Modes 1 and 3 Baud Rate = (1/16) × (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. Hence, 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 45.

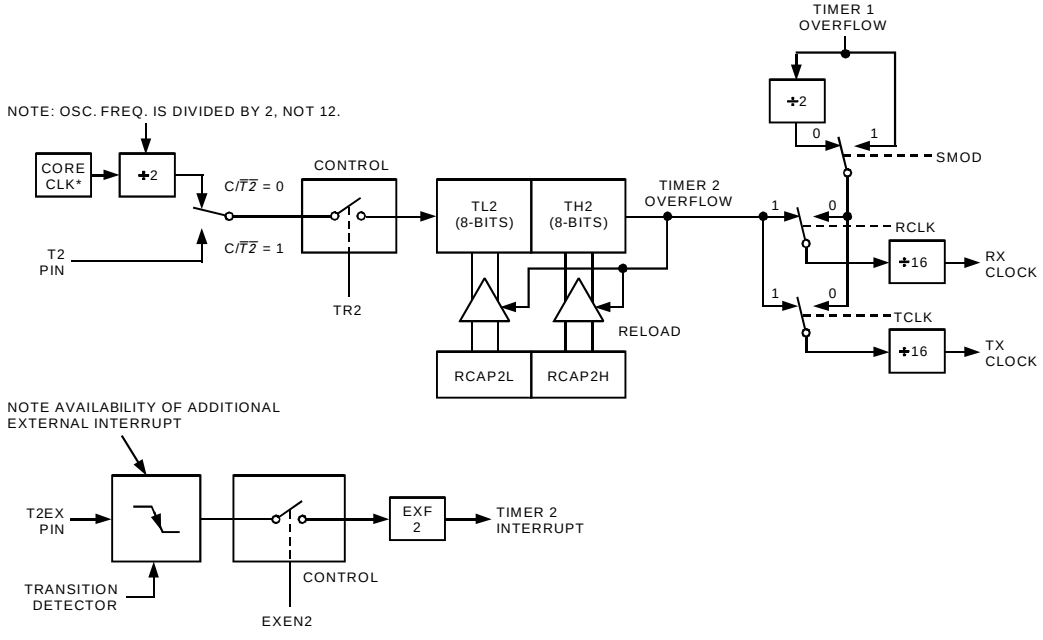
In this case, the baud rate is given by the formula:

Modes 1 and 3 Baud Rate
= (Core Clk)/(32 × [65536 – (RCAP2H, RCAP2L)])

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

Table XXIX. 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



*THE CORE CLOCK IS THE OUTPUT OF THE PLL AS DESCRIBED ON PAGE 42.

Figure 45. 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 Priority-Interrupt Register.

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

EA	EADC	ET2	ES	ET1	EX1	ET0	EXO

Table XXX. IE SFR Bit Designations

Bit	Name	Description
7	EA	Written by User to Enable '1' or Disable '0' All Interrupt Sources
6	EADC	Written by User to Enable '1' or Disable '0' ADC Interrupt
5	ET2	Written by User to Enable '1' or Disable '0' Timer 2 Interrupt
4	ES	Written by User to Enable '1' or Disable '0' UART Serial Port Interrupt
3	ET1	Written by User to Enable '1' or Disable '0' Timer 1 Interrupt
2	EX1	Written by User to Enable '1' or Disable '0' External Interrupt 1
1	ET0	Written by User to Enable '1' or Disable '0' Timer 0 Interrupt
0	EX0	Written by User to Enable '1' or Disable '0' External Interrupt 0

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

----	PADC	PT2	PS	PT1	PX1	PT0	PX0

Table XXXI. IP SFR Bit Designations

Bit	Name	Description
7	---	Reserved for Future Use.
6	PADC	Written by User to Select ADC Interrupt Priority ('1' = High; '0' = Low)
5	PT2	Written by User to Select Timer 2 Interrupt Priority ('1' = High; '0' = Low)
4	PS	Written by User to Select UART Serial Port Interrupt Priority ('1' = High; '0' = Low)
3	PT1	Written by User to Select Timer 1 Interrupt Priority ('1' = High; '0' = Low)
2	PX1	Written by User to Select External Interrupt 1 Priority ('1' = High; '0' = Low)
1	PT0	Written by User to Select Timer 0 Interrupt Priority ('1' = High; '0' = Low)
0	PX0	Written by User to Select External Interrupt 0 Priority ('1' = High; '0' = Low)

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

----	PTI	PPSM	PSI	----	ET1	EPSM	ES1

Table XXXII. IEIP2 SFR Bit Designations

Bit	Name	Description
7	---	Reserved for Future Use.
6	PTI	Written by User to Select TIC Interrupt Priority ('1' = High; '0' = Low).
5	PPSM	Written by User to Select Power Supply Monitor Interrupt Priority ('1' = High; '0' = Low).
4	PSI	Written by User to Select SPI/I^2C Serial Port Interrupt Priority ('1' = High; '0' = Low).
3	---	Reserved, This Bit Must Be '0.'
2	ETI	Written by User to Enable '1' or Disable '0' TIC Interrupt.
1	EPSM	Written by User to Enable '1' or Disable '0' Power Supply Monitor Interrupt.
0	ESI	Written by User to Enable '1' or Disable '0' SPI/I^2C 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 will be 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 XXXIII.

Table XXXIII. 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
I2CI + ISPI	8	I^2C/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 XXXIV.

Table XXXIV. Interrupt Vector Addresses

Source	Vector Address
IE0	0003 Hex
TF0	000B Hex
IE1	0013 Hex
TF1	001B Hex
RI + TI	0023 Hex
TF2 + EXF2	002B Hex
RDY0/RDY1 (ADC)	0033 Hex
II^2C + ISPI	003B Hex
PSMI	0043 Hex
TII	0053 Hex
WDS (WDIR = 1)[1]	005B Hex

Notes
[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 the 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 will always be responded to if a watchdog timeout occurs. The watchdog will only produce an interrupt if the watchdog timeout is greater than zero.

ADuC814 HARDWARE DESIGN CONSIDERATIONS

This section outlines some of the key hardware design considerations that must be addressed when 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 (26 and 27) as shown in Figure 46.

As shown in the typical external crystal connection diagram in Figure 46, two internal 12 pF capacitors are provided on-chip. These are connected internally, directly to the XTAL1 and XTAL2 pins and the total input capacitances at both pins is detailed in the specification section of this data sheet. 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 will not be required.

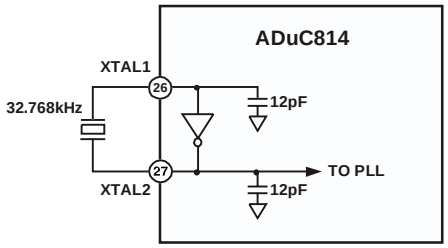


Figure 46. External Parallel Resonant Crystal Connections

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 ±10% of the nominal 5 V level, the chip will function equally well at any power supply level between 2.7 V and 5.5 V.

Separate analog and digital power supply pins (AV_{DD} and DV_{DD} respectively) allow AV_{DD} to be kept relatively free of noisy digital signals often present on the system DVDD line. In this mode the part can also operate with split supplies as long as the supply voltages are within 0.3 volts of each other. A typical split supply configuration is show in Figure 47.

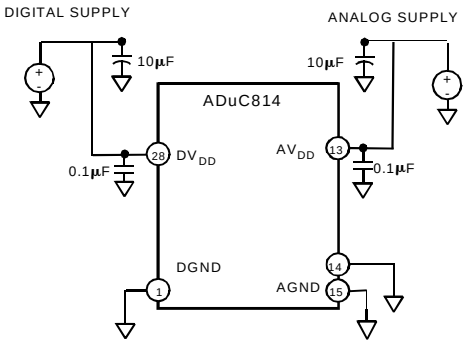


Figure 47. External Dual Supply Connections

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 48. With this configuration other analog circuitry (such as op-amps, voltage reference, etc.) can be powered from the AV_{DD} supply line as well.

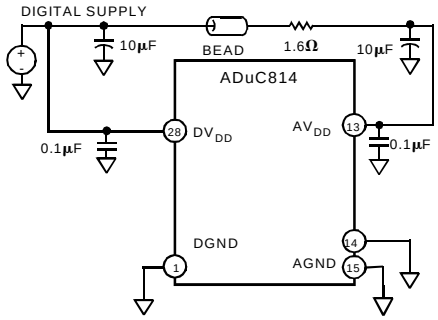


Figure 48. External Single Supply Connections

Notice that in both Figure 47 and Figure 48, a large value (10 µF) reservoir capacitor sits on DV_{DD} and a separate 10 µF capacitor sits on AV_{DD}. Also, local small-value (0.1 µF) capacitors are located at each VDD pin of the chip. As per standard design practice, be sure to include all of these capacitors, and ensure 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, it should also be noticed 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 (watchdog timer, power supply monitor, etc.) 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 will increase by approximately 5 mA during 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 II, allows the chip to be switched from normal mode into idle mode, and also into 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 maintain their data during idle mode. Port pins and DAC output pins also retain their states in this mode. The chip will recover from idle mode upon receiving any enabled interrupt, or on 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:

Asserting the RESET pin (#10)
Returns to normal mode all registers are set to their default state and 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 will return the core to the instruction after that which enabled power down.

I^2C or SPI Interrupt
Power-down mode is terminated and the CPU services the I^2C/SPI interrupt. The RETI at the end of the ISR will return the core to the instruction after that which enabled power down. It should be noted that the I^2C/SPI power down interrupt enable bit (SERIPD) in the PCON SFR must first be set to allow this mode of operation.

$\overline{INT0}$ Interrupt
Power-down mode is terminated and the CPU services the $\overline{INT0}$ interrupt. The RETI at the end of the ISR will return the core to the instruction after that which enabled power-down. It should be noted that the $\overline{INT0}$ power-down interrupt enable bit (INT0PD) in the PCON SFR must first be set to allow this mode of 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 49a. 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 since a ground loop would result. In these cases, tie the ADuC814's AGND and DGND pins all to the analog ground plane, as illustrated in Figure 49b. 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 49c.

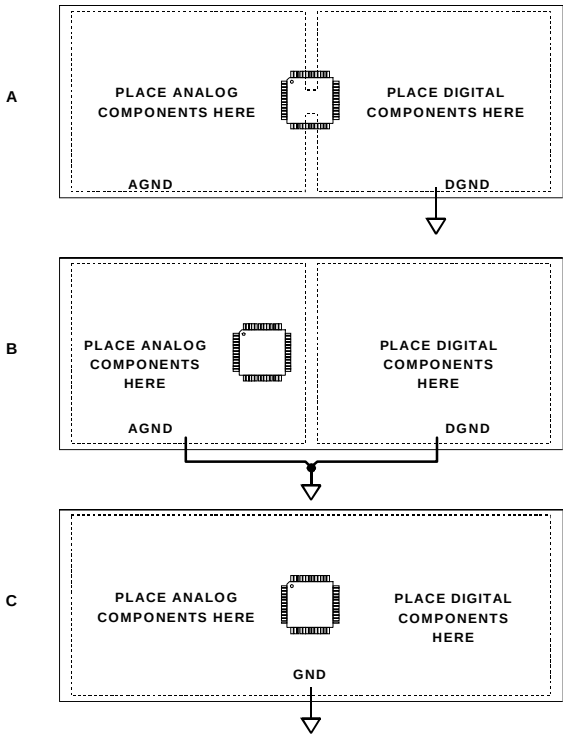


Figure 49. System Grounding Schemes

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 power components on the analog side of Figure 49b with DV_{DD} since 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 the user placed a noisy digital chip on the left half of the board in Figure 49c. Whenever possible, avoid large discontinuities in the ground plane(s) (such as are formed by a long trace on the same layer), since 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, plus in-circuit debug and emulation options, users will want to implement some simple connection points in their hardware that will allow easy access to download, debug, and emulation modes.

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 to the ADuC814's UART, which requires an external RS-232 chip for level translation if downloading code from a PC. 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"[1] for a simple (and zero-cost-per-board) method of gaining in-circuit serial download access to the ADuC814.

Note
[1]Application note uC006 is available at www.analog.com/microconverter

In addition to the basic UART connections, users will 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 above. 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).

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 mention previously, this pin is either high or low to enable entry into serial download and serial debug modes or select normal code execution, as described earlier. To enable single-pin emulation mode, however, users will need to pull the DLOAD pin high through a 1 ký resistor. The emulator will then connect 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).

QUICKSTART DEVELOPMENT SYSTEM

The QuickStart Development System is a full featured, low cost development tool suite supporting the ADuC814. The system consists of the following PC-based (Windows-compatible) hardware and software development tools.

Hardware:	ADuC814 Evaluation Board and Serial Port Cable
Code Development:	8051 Assembler
Code Functionality:	ADSIM, Windows MicroConverter Code Simulator
In-Circuit Code Download:	Serial Downloader
In-Circuit Debugger:	Serial Port Debugger
Misc. Other:	CD-ROM Documentation

Figures 50 shows the typical components of a QuickStart Development System while Figure 51 shows a typical debug session. A brief description of some of the software tools' components in the QuickStart Development System is given below.

Figure 50. Components of the QuickStart Development System

Download In-Circuit Serial Downloader

The Serial Downloader is a software program that allows the user to serially download an assembled program (Intel Hex format file) to the on-chip program FLASH memory via the serial COM1 port on a standard PC. An Application Note (uC004) detailing this serial download protocol is available from www.analog.com/microconverter.

The QuickStart development tool-suite software is freely available at the Analog Devices MicroConverter Website www.analog.com/microconverter.

DeBug-In-Circuit Debugger

The Debugger is a Windows application that allows the user to debug code execution on silicon using the MicroConverter UART serial port. The debugger provides access to all on-chip peripherals during a typical debug session as well as single-step and break-point code execution control.

ADSIM-Windows Simulator

The Simulator is a Windows application that fully simulates all the MicroConverter functionality including ADC and DAC peripherals. The simulator provides an easy-to-use, intuitive, interface to the MicroConverter functionality and integrates many standard debug features; including multiple breakpoints, single stepping; and code execution trace capability. This tool can be used both as a tutorial guide to the part as well as an efficient way to prove code functionality before moving to a hardware platform.

The QuickStart development tool-suite software is freely available at the Analog Devices MicroConverter Website www.analog.com/microconverter.

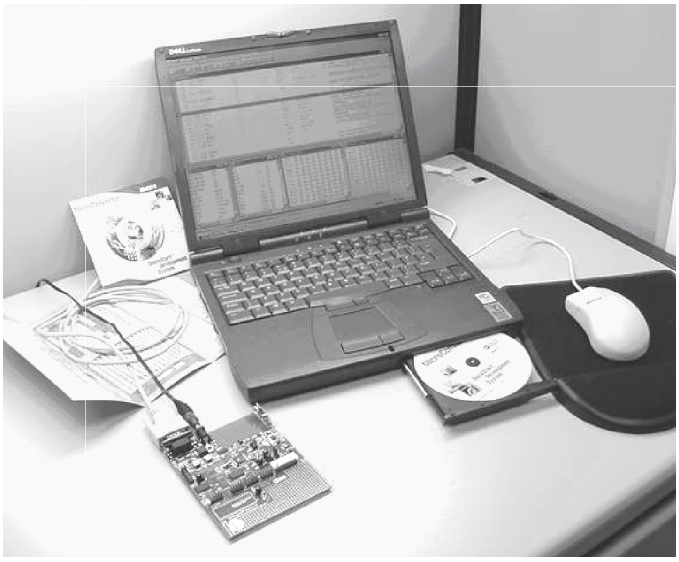
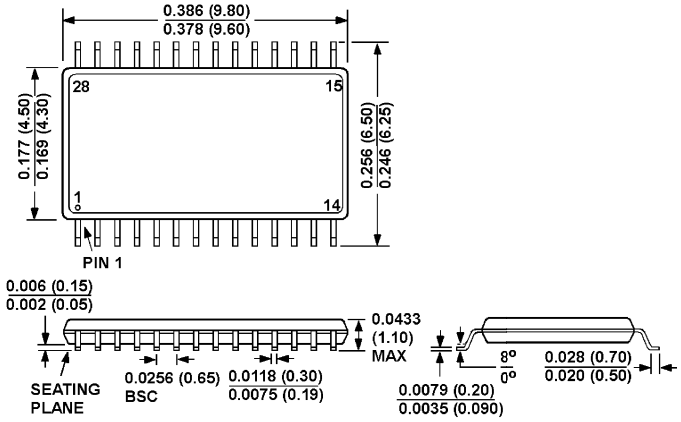


Figure. 51. Typical Debug Session

OUTLINE DIMENSIONS

Dimensions shown in inches and (mm).



C02345–2.5–12/00 (rev. 0)

PRINTED IN U.S.A.