MicroConverter®, Dual 16-Bit/24-Bit Σ-Δ ADCs with Embedded 62 kB Flash MCU

ADuC834

FEATURES

High Resolution Σ-Δ ADCs
- 2 Independent ADCs (16-Bit and 24-Bit Resolution)
- 24-Bit No Missing Codes, Primary ADC
- 21-Bit rms (18.5-Bit p-p) Effective Resolution @ 20 Hz
- Offset Drift 10 nV/°C, Gain Drift 0.5 ppm/°C

Memory
- 62 Kbytes On-Chip Flash/EE Program Memory
- 4 Kbytes On-Chip Flash/EE Data Memory
- Flash/EE, 100 Year Retention, 100 Kcycles Endurance
- 3 Levels of Flash/EE Program Memory Security
- In-Circuit Serial Download (No External Hardware)
- High Speed User Download (5 Seconds)
- 2304 Bytes On-Chip Data RAM

8051-Based Core
- 8051 Compatible Instruction Set
- 32 kHz External Crystal
- On-Chip Programmable PLL (12.58 MHz Max)
- 3 × 16-Bit Timer/Counter
- 26 Programmable I/O Lines
- 11 Interrupt Sources, Two Priority Levels
- Dual Data Pointer, Extended 11-Bit Stack Pointer

On-Chip Peripherals
- Internal Power on Reset Circuit
- 12-Bit Voltage Output DAC
- Dual 16-Bit Σ-Δ DACs/PWMs
- On-Chip Temperature Sensor
- Dual Excitation Current Sources
- Time Interval Counter (Wake-Up/RTC Timer)
- UART, SPI®, and I²C® Serial I/O
- High Speed Baud Rate Generator (Including 115,200)
- Watchdog Timer (WDT)
- Power Supply Monitor (PSM)

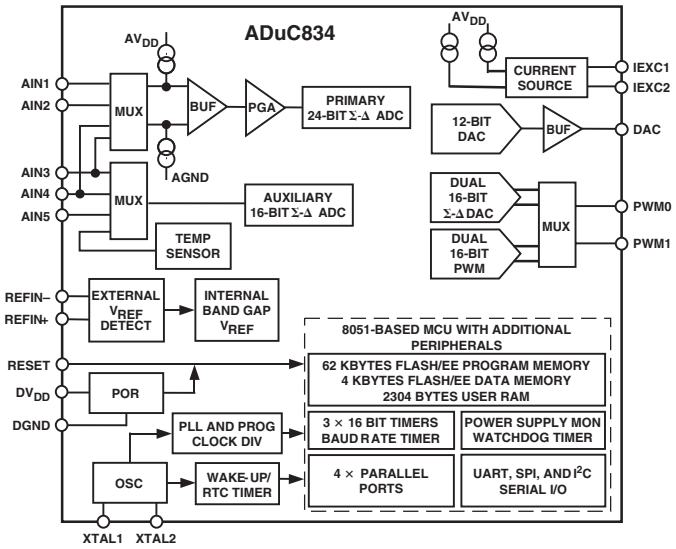
Power
- Normal: 2.3 mA Max @ 3.6 V (Core CLK = 1.57 MHz)
- Power-Down: 20 μA Max with Wake-Up Timer Running
- Specified for 3 V and 5 V Operation

Package and Temperature Range
- 52-Lead MQFP (14 mm × 14 mm), –40°C to +125°C
- 56-Lead LFCSP (8 mm × 8 mm), –40°C to +85°C

APPLICATIONS

Intelligent Sensors
Weigh Scales
Portable Instrumentation, Battery-Powered Systems
4–20 mA Transmitters
Data Logging
Precision System Monitoring

FUNCTIONAL BLOCK DIAGRAM



GENERAL DESCRIPTION

The ADuC834 is a complete smart transducer front end, integrating two high resolution Σ-Δ ADCs, an 8-bit MCU, and program/data Flash/EE memory on a single chip.

The two independent ADCs (primary and auxiliary) include a temperature sensor and a PGA (allowing direct measurement of low level signals). The ADCs with on-chip digital filtering and programmable output data rates are intended for the measurement of wide dynamic range, low frequency signals, such as those in weigh scale, strain-gage, pressure transducer, or temperature measurement applications.

The device operates from a 32 kHz crystal with an on-chip PLL generating a high frequency clock of 12.58 MHz. This clock is routed through a programmable clock divider from which the MCU core clock operating frequency is generated. The microcontroller core is an 8052 and therefore 8051 instruction set compatible with 12 core clock periods per machine cycle.

62 Kbytes of nonvolatile Flash/EE program memory, 4 Kbytes of nonvolatile Flash/EE data memory, and 2304 bytes of data RAM are provided on-chip. The program memory can be configured as data memory to give up to 60 Kbytes of NV data memory in data logging applications.

On-chip factory firmware supports in-circuit serial download and debug modes (via UART), as well as single-pin emulation mode via the $\overline{EA}$ pin. The ADuC834 is supported by a QuickStart™ development system featuring low cost software and hardware development tools.

REV. A

Information furnished by Analog Devices is believed to be accurate and reliable. However, no responsibility is assumed by Analog Devices for its use, nor for any infringements of patents or other rights of third parties that may result from its use. No license is granted by implication or otherwise under any patent or patent rights of Analog Devices. Trademarks and registered trademarks are the property of their respective companies.

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.
Tel: 781/329-4700 **www.analog.com**
Fax: 781/326-8703 **© 2003 Analog Devices, Inc. All rights reserved.**

TABLE OF CONTENTS

FEATURES 1

APPLICATIONS 1

FUNCTIONAL BLOCK DIAGRAM 1

GENERAL DESCRIPTION 1

SPECIFICATIONS 3

ABSOLUTE MAXIMUM RATINGS 9

ORDERING GUIDE 9

PIN CONFIGURATIONS 9

DETAILED BLOCK DIAGRAM 10

PIN FUNCTION DESCRIPTIONS 10

MEMORY ORGANIZATION 13

SPECIAL FUNCTION REGISTERS (SFRS) 14
- Accumulator (ACC) 14
- B SFR (B) 14
- Data Pointer (DPTR) 14
- Stack Pointer (SP and SPH) 15
- Program Status Word (PSW) 15
- Power Control SFR (PCON) 15
- ADuC834 Configuration SFR (CFG834) 15
- Complete SFR Map 16

ADC SFR INTERFACE
- ADCSTAT 17
- ADCMODE 18
- ADC0CON 19
- ADC1CON 19
- ADC0H/ADC0M/ADC0L/ADC1H/ADC1L 20
- OF0H/OF0M/OF0L/OF1H/OF1L 20
- GN0H/GN0M/GN0L/GN1H/GN1L 20
- SF 21
- ICON 21

PRIMARY AND AUXILIARY ADC NOISE PERFORMANCE 22

PRIMARY AND AUXILIARY ADC CIRCUIT DESCRIPTION
- Overview 23
- Primary ADC 23
- Auxiliary ADC 24
- Analog Input Channels 24
- Primary and Auxiliary ADC Inputs 25
- Analog Input Ranges 25
- Programmable Gain Amplifier 25
- Bipolar/Unipolar Inputs 25
- Reference Input 26
- Burnout Currents 26
- Excitation Currents 26
- Reference Detect 26
- Σ-Δ Modulator 26
- Digital Filter 27
- ADC Chopping 28
- Calibration 28

NONVOLATILE FLASH/EE MEMORY
- Flash/EE Memory Overview 28
- Flash/EE Memory and the ADuC834 28
- ADuC834 Flash/EE Memory Reliability 29
- Flash/EE Program Memory 30
- Serial Downloading 30
- Parallel Programming 30
- User Download Mode (ULOAD) 30
- Flash/EE Program Memory Security 31
- Lock, Secure, and Serial Safe Modes 31
- Using the Flash/EE Data Memory 31
- ECON 32
- Programming the Flash/EE Data Memory 33
- Flash/EE Memory Timing 33

OTHER ON-CHIP PERIPHERALS
- DAC 34
- Pulsewidth Modulator (PWM) 36
- On-Chip PLL 39
- Time Interval Counter (Wake-Up/RTC Timer) 40
- Watchdog Timer 42
- Power Supply Monitor 43
- Serial Peripheral Interface (SPI) 44
- I^2C Serial Interface 46
- Dual Data Pointer 48

8052 COMPATIBLE ON-CHIP PERIPHERALS
- Parallel I/O Ports 0–3 49
- Timers/Counters 52
- UART Serial Interface 57
- UART Operating Modes 57
- Baud Rate Generation Using Timer 1 and Timer 2 59
- Baud Rate Generation Using Timer 3 60
- Interrupt System 61

HARDWARE DESIGN CONSIDERATIONS
- External Memory Interface 63
- Power Supplies 64
- Power-On Reset (POR) Operation 64
- Power Consumption 64
- Power Saving Modes 65
- Wake-Up from Power-Down Latency 65
- Grounding and Board Layout Recommendations 65
- ADuC834 System Self-Identification 66
- Clock Oscillator 66

OTHER HARDWARE CONSIDERATIONS
- In-Circuit Serial Download Access 67
- Embedded Serial Port Debugger 67
- Single-Pin Emulation Mode 67
- Typical System Configuration 68

QUICKSTART DEVELOPMENT SYSTEM 69

TIMING SPECIFICATIONS 70

OUTLINE DIMENSIONS 80

SPECIFICATIONS[1]

(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, REFIN(+) = 2.5 V, REFIN(–) = AGND; AGND = DGND = 0 V; XTAL1/XTAL2 = 32.768 kHz Crystal; all specifications T_{MIN} to T_{MAX} unless otherwise noted.)

Parameter	ADuC834	Test Conditions/Comments	Unit
ADC SPECIFICATIONS			
Conversion Rate	5.4	On Both Channels	Hz min
	105	Programmable in 0.732 ms Increments	Hz max
Primary ADC			
No Missing Codes[2]	24	20 Hz Update Rate	Bits min
Resolution	13.5	Range = ±20 mV, 20 Hz Update Rate	Bits p-p typ
	18.5	Range = ±2.56 V, 20 Hz Update Rate	Bits p-p typ
Output Noise	See Tables X and XI in ADuC834 ADC Description	Output Noise Varies with Selected Update Rate and Gain Range	
Integral Nonlinearity	±15	1 LSB_{16}	ppm of FSR max
Offset Error[3]	±3		μV typ
Offset Error Drift	±10		nV/°C typ
Full-Scale Error[4]	±10		μV typ
Gain Error Drift[5]	±0.5		ppm/°C typ
ADC Range Matching	±2	AIN = 18 mV	μV typ
Power Supply Rejection (PSR)	113	AIN = 7.8 mV, Range = ±20 mV	dBs typ
	80	AIN = 1 V, Range = ±2.56 V	dBs min
Common-Mode DC Rejection			
On AIN	95	At DC, AIN = 7.8 mV, Range = ±20 mV	dBs min
	113	At DC, AIN = 1 V, Range = ±2.56 V	dBs typ
On REFIN	125	At DC, AIN = 1 V, Range = ±2.56 V	dBs typ
Common-Mode 50 Hz/60 Hz Rejection[2]		20 Hz Update Rate	
On AIN	95	50 Hz/60 Hz ±1 Hz, AIN = 7.8 mV, Range = ±20 mV	dBs min
	90	50 Hz/60 Hz ±1 Hz, AIN = 1 V, Range = ±2.56 V	dBs min
On REFIN	90	50 Hz/60 Hz ±1 Hz, AIN = 1 V, Range = ±2.56 V	dBs min
Normal Mode 50 Hz/60 Hz Rejection[2]			
On AIN	60	50 Hz/60 Hz ±1 Hz, 20 Hz Update Rate	dBs min
On REFIN	60	50 Hz/60 Hz ±1 Hz, 20 Hz Update Rate	dBs min
Auxiliary ADC			
No Missing Codes[2]	16		Bits min
Resolution	16	Range = ±2.5 V, 20 Hz Update Rate	Bits p-p typ
Output Noise	See Table XII in ADuC834 ADC Description	Output Noise Varies with Selected Update Rate	
Integral Nonlinearity	±15		ppm of FSR max
Offset Error[3]	–2		LSB typ
Offset Error Drift	1		μV/°C typ
Full-Scale Error[6]	–2.5		LSB typ
Gain Error Drift[5]	±0.5		ppm/°C typ
Power Supply Rejection (PSR)	80	AIN = 1 V, 20 Hz Update Rate	dBs min
Normal Mode 50 Hz/60 Hz Rejection[2]			
On AIN	60	50 Hz/60 Hz ±1 Hz	dBs min
On REFIN	60	50 Hz/60 Hz ±1 Hz, 20 Hz Update Rate	dBs min
DAC PERFORMANCE			
DC Specifications[7]			
Resolution	12		Bits
Relative Accuracy	±3		LSB typ
Differential Nonlinearity	–1	Guaranteed 12-Bit Monotonic	LSB max
Offset Error	±50		mV max
Gain Error[8]	±1	AV_{DD} Range	% max
	±1	V_{REF} Range	% typ
AC Specifications[2, 7]			
Voltage Output Settling Time	15	Settling Time to 1 LSB of Final Value	μs typ
Digital-to-Analog Glitch Energy	10	1 LSB Change at Major Carry	nVs typ

SPECIFICATIONS (continued)

Parameter	ADuC834	Test Conditions/Comments	Unit
INTERNAL REFERENCE			
ADC Reference			
Reference Voltage	1.25 ± 1%	Initial Tolerance @ 25°C, V_{DD} = 5 V	V min/max
Power Supply Rejection	45		dBs typ
Reference Tempco	100		ppm/°C typ
DAC Reference			
Reference Voltage	2.5 ± 1%	Initial Tolerance @ 25°C, V_{DD} = 5 V	V min/max
Power Supply Rejection	50		dBs typ
Reference Tempco	±100		ppm/°C typ
ANALOG INPUTS/REFERENCE INPUTS			
Primary ADC			
Differential Input Voltage Ranges[9, 10]		External Reference Voltage = 2.5 V RN2, RN1, RN0 of ADC0CON Set to	
Bipolar Mode (ADC0CON3 = 0)	±20	0 0 0 (Unipolar Mode 0 mV to 20 mV)	mV
	±40	0 0 1 (Unipolar Mode 0 mV to 40 mV)	mV
	±80	0 1 0 (Unipolar Mode 0 mV to 80 mV)	mV
	±160	0 1 1 (Unipolar Mode 0 mV to 160 mV)	mV
	±320	1 0 0 (Unipolar Mode 0 mV to 320 mV)	mV
	±640	1 0 1 (Unipolar Mode 0 mV to 640 mV)	mV
	±1.28	1 1 0 (Unipolar Mode 0 V to 1.28 V)	V
	±2.56	1 1 1 (Unipolar Mode 0 V to 2.56 V)	V
Analog Input Current[2]	±1	T_{MAX} = 85°C	nA max
	±5	T_{MAX} = 125°C	nA max
Analog Input Current Drift	±5	T_{MAX} = 85°C	pA/°C typ
	±15	T_{MAX} = 125°C	pA/°C typ
Absolute AIN Voltage Limits[2]	AGND + 100 mV		V min
	AV_{DD} – 100 mV		V max
Auxiliary ADC			
Input Voltage Range[9, 10]	0 to V_{REF}	Unipolar Mode, for Bipolar Mode See Note 11	V
Average Analog Input Current	125	Input Current Will Vary with Input	nA/V typ
Average Analog Input Current Drift[2]	±2	Voltage on the Unbuffered Auxiliary ADC	pA/V/°C typ
Absolute AIN Voltage Limits[2, 11]	AGND – 30 mV		V min
	AV_{DD} + 30 mV		V max
External Reference Inputs			
REFIN(+) to REFIN(–) Range[2]	1		V min
	AV_{DD}		V max
Average Reference Input Current	1	Both ADCs Enabled	μA/V typ
Average Reference Input Current Drift	±0.1		nA/V/°C typ
'NO Ext. REF' Trigger Voltage	0.3	NOXREF Bit Active if V_{REF} < 0.3 V	V min
	0.65	NOXREF Bit Inactive if V_{REF} > 0.65 V	V max
ADC SYSTEM CALIBRATION			
Full-Scale Calibration Limit	+1.05 × FS		V max
Zero-Scale Calibration Limit	–1.05 × FS		V min
Input Span	0.8 × FS		V min
	2.1 × FS		V max
ANALOG (DAC) OUTPUT			
Voltage Range	0 to V_{REF}	DACRN = 0 in DACCON SFR	V typ
	0 to AV_{DD}	DACRN = 1 in DACCON SFR	V typ
Resistive Load	10	From DAC Output to AGND	kΩ typ
Capacitive Load	100	From DAC Output to AGND	pF typ
Output Impedance	0.5		Ω typ
I_{SINK}	50		μA typ
TEMPERATURE SENSOR			
Accuracy	±2		°C typ
Thermal Impedance (θ_{JA})	90	MQFP Package	°C/W typ
	52	CSP Package (Base Floating)[12]	°C/W typ

Parameter	ADuC834	Test Conditions/Comments	Unit
TRANSDUCER BURNOUT CURRENT SOURCES			
AIN+ Current	–100	AIN+ Is the Selected Positive Input to the Primary ADC	nA typ
AIN– Current	+100	AIN– Is the Selected Negative Input to the Auxiliary ADC	nA typ
Initial Tolerance @ 25°C	±10		% typ
Drift	0.03		%/°C typ
EXCITATION CURRENT SOURCES			
Output Current	–200	Available from Each Current Source	μA typ
Initial Tolerance @ 25°C	±10		% typ
Drift	200		ppm/°C typ
Initial Current Matching @ 25°C	±1	Matching between Both Current Sources	% typ
Drift Matching	20		ppm/°C typ
Line Regulation (AV_{DD})	1	AV_{DD} = 5 V + 5%	μA/V typ
Load Regulation	0.1		μA/V typ
Output Compliance[2]	AV_{DD} – 0.6		V max
	AGND		min
LOGIC INPUTS			
All Inputs Except SCLOCK, RESET, and XTAL1[2]			
V_{INL}, Input Low Voltage	0.8	DV_{DD} = 5 V	V max
	0.4	DV_{DD} = 3 V	V max
V_{INH}, Input High Voltage	2.0		V min
SCLOCK and RESET Only (Schmitt-Triggered Inputs)[2]			
V_{T+}	1.3/3	DV_{DD} = 5 V	V min/V max
	0.95/2.5	DV_{DD} = 3 V	V min/V max
V_{T-}	0.8/1.4	DV_{DD} = 5 V	V min/V max
	0.4/1.1	DV_{DD} = 3 V	V min/V max
$V_{T+} - V_{T-}$	0.3/0.85	DV_{DD} = 5 V	V min/V max
	0.3/0.85	DV_{DD} = 3 V	V min/V max
Input Currents			
Port 0, P1.2–P1.7, $\overline{EA}$	±10	V_{IN} = 0 V or V_{DD}	μA max
SCLOCK, MOSI, MISO, $\overline{SS}$[13]	–10 min, –40 max	V_{IN} = 0 V, DV_{DD} = 5 V, Internal Pull-Up	μA min/μA max
	±10	V_{IN} = V_{DD}, DV_{DD} = 5 V	μA max
RESET	±10	V_{IN} = 0 V, DV_{DD} = 5 V	μA max
	35 min, 105 max	V_{IN} = V_{DD}, DV_{DD} = 5 V, Internal Pull-Down	μA min/μA max
P1.0, P1.1, Ports 2 and 3	±10	V_{IN} = V_{DD}, DV_{DD} = 5 V	μA max
	–180	V_{IN} = 2 V, DV_{DD} = 5 V	μA min
	–660		μA max
	–20	V_{IN} = 450 mV, DV_{DD} = 5 V	μA min
	–75		μA max
Input Capacitance	5	All Digital Inputs	pF typ
CRYSTAL OSCILLATOR (XTAL1 AND XTAL2)			
Logic Inputs, XTAL1 Only[2]			
V_{INL}, Input Low Voltage	0.8	DV_{DD} = 5 V	V max
	0.4	DV_{DD} = 3 V	V max
V_{INH}, Input High Voltage	3.5	DV_{DD} = 5 V	V min
	2.5	DV_{DD} = 3 V	V min
XTAL1 Input Capacitance	18		pF typ
XTAL2 Output Capacitance	18		pF typ

SPECIFICATIONS (continued)

Parameter	ADuC834	Test Conditions/Comments	Unit
LOGIC OUTPUTS (Not Including XTAL2)[2]			
V_{OH}, Output High Voltage	2.4	V_{DD} = 5 V, I_{SOURCE} = 80 μA	V min
	2.4	V_{DD} = 3 V, I_{SOURCE} = 20 μA	V min
V_{OL}, Output Low Voltage[14]	0.4	I_{SINK} = 8 mA, SCLOCK, MOSI/SDATA	V max
	0.4	I_{SINK} = 10 mA, P1.0 and P1.1	V max
	0.4	I_{SINK} = 1.6 mA, All Other Outputs	V max
Floating State Leakage Current[2]	±10		μA max
Floating State Output Capacitance	5		pF typ
POWER SUPPLY MONITOR (PSM)			
AV_{DD} Trip Point Selection Range	2.63	Four Trip Points Selectable in This Range	V min
	4.63	Programmed via TPA1–0 in PSMCON	V max
AV_{DD} Power Supply Trip Point Accuracy	±3.0	T_{MAX} = 85°C	% max
	±4.0	T_{MAX} = 125°C	% max
DV_{DD} Trip Point Selection Range	2.63	Four Trip Points Selectable in This Range	V min
	4.63	Programmed via TPD1–0 in PSMCON	V max
DV_{DD} Power Supply Trip Point Accuracy	±3.0	T_{MAX} = 85°C	% max
	±4.0	T_{MAX} = 125°C	% max
WATCHDOG TIMER (WDT)			
Timeout Period	0	Nine Timeout Periods in This Range	ms min
	2000	Programmed via PRE3–0 in WDCON	ms max
MCU CORE CLOCK RATE		Clock Rate Generated via On-Chip PLL	
MCU Clock Rate[2]	98.3	Programmable via CD2–0 Bits in PLLCON SFR	kHz min
	12.58		MHz max
START-UP TIME			
At Power-On	300		ms typ
After External RESET in Normal Mode	3		ms typ
After WDT Reset in Normal Mode	3	Controlled via WDCON SFR	ms typ
From Idle Mode	10		μs typ
From Power-Down Mode			
Oscillator Running		OSC_PD Bit = 0 in PLLCON SFR	
Wake-Up with $\overline{INT0}$ Interrupt	20		μs typ
Wake-Up with SPI Interrupt	20		μs typ
Wake-Up with TIC Interrupt	20		μs typ
Wake-Up with External RESET	3		ms typ
Oscillator Powered Down		OSC_PD Bit = 1 in PLLCON SFR	
Wake-Up with $\overline{INT0}$ Interrupt	20		μs typ
Wake-Up with SPI Interrupt	20		μs typ
Wake-Up with External RESET	5		ms typ
FLASH/EE MEMORY RELIABILITY CHARACTERISTICS[15]			
Endurance[16]	100,000		Cycles min
Data Retention[17]	100		Years min

Parameter	ADuC834	Test Conditions/Comments	Unit
POWER REQUIREMENTS		DV_{DD} and AV_{DD} Can Be Set Independently	
Power Supply Voltages			
AV_{DD}, 3 V Nominal Operation	2.7		V min
	3.6		V max
AV_{DD}, 5 V Nominal Operation	4.75		V min
	5.25		V max
DV_{DD}, 3 V Nominal Operation	2.7		V min
	3.6		V max
DV_{DD}, 5 V Nominal Operation	4.75		V min
	5.25		V max
5 V POWER CONSUMPTION		DV_{DD} = 4.75 V to 5.25 V, AV_{DD} = 5.25 V	
Power Supply Currents Normal Mode[18, 19]			
DV_{DD} Current	4	Core CLK = 1.57 MHz	mA max
DV_{DD} Current	13	Core CLK = 12.58 MHz	mA typ
	16	Core CLK = 12.58 MHz	mA max
AV_{DD} Current	180	Core CLK = 1.57 MHz or 12.58 MHz	μA max
Typical Additional Power Supply Currents (AI_{DD} and DI_{DD})		Core CLK = 1.57 MHz	
PSM Peripheral	50		μA typ
Primary ADC	1		mA typ
Auxiliary ADC	500		μA typ
DAC	150		μA typ
Dual Current Sources	400		μA typ
3 V POWER CONSUMPTION		DV_{DD} = 2.7 V to 3.6 V	
Power Supply Currents Normal Mode[18, 19]			
DV_{DD} Current	2.3	Core CLK = 1.57 MHz	mA max
DV_{DD} Current	8	Core CLK = 12.58 MHz	mA typ
	10	Core CLK = 12.58 MHz	mA max
AV_{DD} Current	180	AV_{DD} = 5.25 V, Core CLK = 1.57 MHz or 12.58 MHz	μA max
Power Supply Currents Power-Down Mode[18, 19]		Core CLK = 1.57 MHz or 12.58 MHz	
DV_{DD} Current	20	T_{MAX} = 85°C; Osc. On, TIC On	μA max
	40	T_{MAX} = 125°C; Osc. On, TIC On	μA max
DV_{DD} Current	10	Osc. Off	μA typ
AV_{DD} Current	1	AV_{DD} = 5.25 V; T_{MAX} = 85°C; Osc. On or Osc. Off	μA max
	3	AV_{DD} = 5.25 V; T_{MAX} = 125°C; Osc. On or Osc. Off	μA max

NOTES

[1] Temperature Range for ADuC834BS (MQFP package) is –40°C to +125°C.
Temperature Range for ADuC834BCP (CSP package) is –40°C to +85°C.

[2] These numbers are not production tested but are guaranteed by design and/or characterization data on production release.

[3] System Zero-Scale Calibration can remove this error.

[4] The primary ADC is factory calibrated at 25°C with $AV_{DD} = DV_{DD} = 5$ V yielding this full-scale error of 10 μV. If user power supply or temperature conditions are significantly different from these, an Internal Full-Scale Calibration will restore this error to 10 μV. A system zero-scale and full-scale calibration will remove this error altogether.

[5] Gain Error Drift is a span drift. To calculate Full-Scale Error Drift, add the Offset Error Drift to the Gain Error Drift times the full-scale input.

[6] The auxiliary ADC is factory calibrated at 25°C with $AV_{DD} = DV_{DD} = 5$ V yielding this full-scale error of –2.5 LSB. A system zero-scale and full-scale calibration will remove this error altogether.

[7] DAC linearity and ac specifications are calculated using: reduced code range of 48 to 4095, 0 to V_{REF}; reduced code range of 100 to 3950, 0 to V_{DD}.

[8] Gain Error is a measure of the span error of the DAC.

[9] In general terms, the bipolar input voltage range to the primary ADC is given by $RangeADC = \pm(V_{REF}\ 2^{RN})/125$, where:
V_{REF} = REFIN(+) to REFIN(–) voltage and V_{REF} = 1.25 V when internal ADC V_{REF} is selected. RN = decimal equivalent of RN2, RN1, RN0, e.g., V_{REF} = 2.5 V and RN2, RN1, RN0 = 1, 1, 0 the $Range_{ADC} = \pm 1.28$ V. In unipolar mode, the effective range is 0 V to 1.28 V in our example.

[10] 1.25 V is used as the reference voltage to the ADC when internal V_{REF} is selected via XREF0 and XREF1 bits in ADC0CON and ADC1CON, respectively.

[11] In bipolar mode, the Auxiliary ADC can only be driven to a minimum of AGND – 30 mV as indicated by the Auxiliary ADC absolute AIN voltage limits. The bipolar range is still $–V_{REF}$ to $+V_{REF}$; however, the negative voltage is limited to –30 mV.

[12] The ADuC834BCP (CSP Package) has been qualified and tested with the base of the CSP Package floating.

[13] Pins configured in SPI Mode, pins configured as digital inputs during this test.

[14] Pins configured in I^2C Mode only.

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

[16] Endurance is qualified to 100 Kcycles as per JEDEC Std. 22 method A117 and measured at –40°C, +25°C, +85°C, and +125°C. Typical endurance at 25°C is 700 Kcycles.

[17] Retention lifetime equivalent at junction temperature (T_J) = 55°C as per JEDEC Std. 22, Method A117. Retention lifetime based on an activation energy of 0.6eV will derate with junction temperature as shown in Figure 16 in the Flash/EE Memory section of this data sheet.

[18] 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 P0 pins and 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.

[19] DV_{DD} 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.

Specifications subject to change without notice.

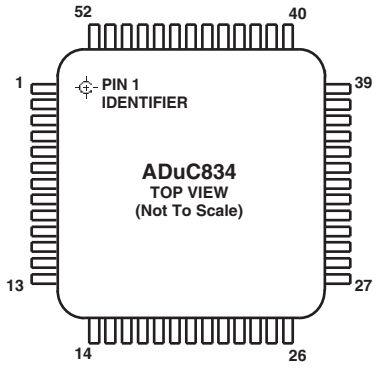
ABSOLUTE MAXIMUM RATINGS[1]

(T_A = 25°C, unless otherwise noted.)

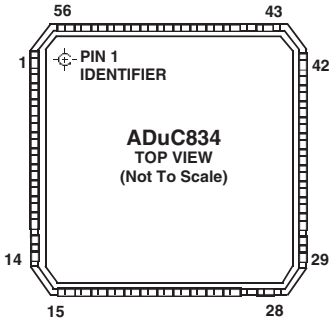
AV_{DD} to AGND ..... –0.3 V to +7 V
AV_{DD} to DGND ..... –0.3 V to +7 V
DV_{DD} to AGND ..... –0.3 V to +7 V
DV_{DD} to DGND ..... –0.3 V to +7 V
AGND to DGND[2] ..... –0.3 V to +0.3 V
AV_{DD} to DV_{DD} ..... –2 V to +5 V
Analog Input Voltage to AGND[3] .... –0.3 V to AV_{DD} + 0.3 V
Reference Input Voltage to AGND .. –0.3 V to AV_{DD} + 0.3 V
AIN/REFIN Current (Indefinite) ..... 30 mA
Digital Input Voltage to DGND .... –0.3 V to DV_{DD} + 0.3 V
Digital Output Voltage to DGND ... –0.3 V to DV_{DD} + 0.3 V
Operating Temperature Range ..... –40°C to +125°C
Storage Temperature Range ..... –65°C to +150°C
Junction Temperature ..... 150°C
θ_{JA} Thermal Impedance (MQFP) ..... 90°C/W
θ_{JA} Thermal Impedance (LFCSP Base Floating) ..... 52°C/W
Lead Temperature, Soldering
Vapor Phase (60 sec) ..... 215°C
Infrared (15 sec) ..... 220°C

NOTES
[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 ADuC834.
[3]Applies to P1.2 to P1.7 pins operating in analog or digital input modes.

PIN CONFIGURATION
52-Lead MQFP



56-Lead LFCSP



ORDERING GUIDE

Model	Temperature Range	Package Description	Package Option
ADuC834BS	–40°C to +125°C	52-Lead Metric Quad Flat Package	S-52
ADuC834BCP	–40°C to +85°C	56-Lead Frame Chip Scale Package	CP-56
EVAL-ADuC834QS		QuickStart Development System	

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 ADuC834 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.

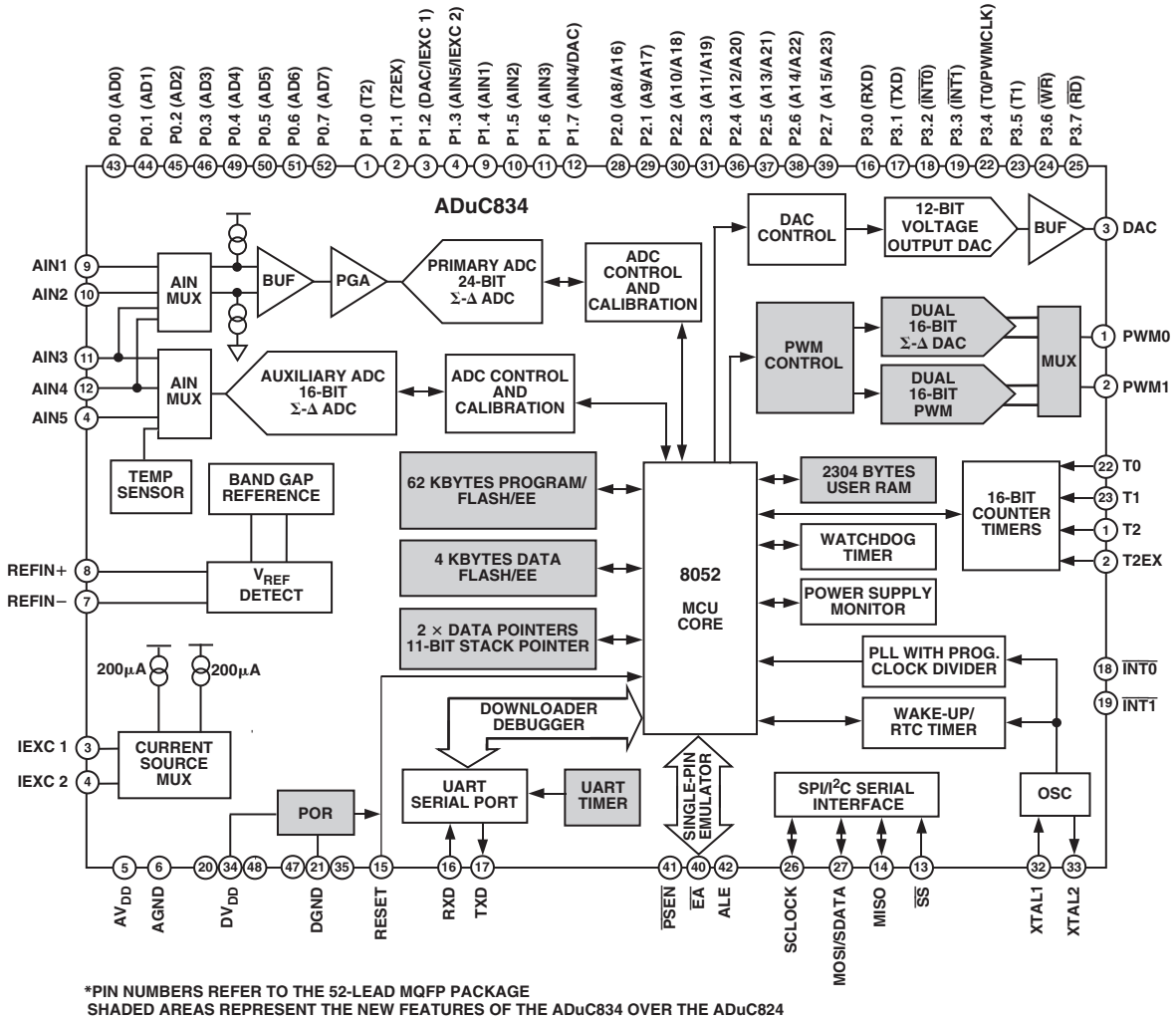


Figure 1. Detailed Block Diagram

PIN FUNCTION DESCRIPTIONS

Pin No. 52-Lead MQFP	Pin No. 56-Lead CSP	Mnemonic	Type*	Description
1, 2	56, 1	P1.0/P1.1	I/O	P1.0 and P1.1 can function as a digital inputs or digital outputs and have a pull-up configuration as described below for Port 3. P1.0 and P1.1 have an increased current drive sink capability of 10 mA.
		P1.0/T2/PWM0	I/O	P1.0 and P1.1 also have various secondary functions as described below. P1.0 can also be used to provide a clock input to Timer 2. When enabled, counter 2 is incremented in response to a negative transition on the T2 input pin. If the PWM is enabled, the PWM0 output will appear at this pin.
		P1.1/T2EX/PWM1	I/O	P1.1 can also be used to provide a control input to Timer 2. When enabled, a negative transition on the T2EX input pin will cause a Timer 2 capture or reload event. If the PWM is enabled, the PWM1 output will appear at this pin.

PIN FUNCTION DESCRIPTIONS (continued)

Pin No. 52-Lead MQFP	Pin No. 56-Lead CSP	Mnemonic	Type*	Description
3–4, 9–12	2–3, 11–14	P1.2–P1.7	I	Port 1.2 to Port 1.7 have no digital output driver; they can function as a digital input for which '0' must be written to the port bit. As a digital input, these pins must be driven high or low externally. These pins also have the following analog functionality:
		P1.2/DAC/IEXC1	I/O	The voltage output from the DAC or one or both current sources (200 μA or 2 × 200 μA) can be configured to appear at this pin.
		P1.3/AIN5/IEXC2	I/O	Auxiliary ADC Input or one or both current sources can be configured at this pin.
		P1.4/AIN1	I	Primary ADC, Positive Analog Input
		P1.5/AIN2	I	Primary ADC, Negative Analog Input
		P1.6/AIN3	I	Auxiliary ADC Input or Muxed Primary ADC, Positive Analog Input
		P1.7/AIN4/DAC	I/O	Auxiliary ADC Input or Muxed Primary ADC, Negative Analog Input. The voltage output from the DAC can also be configured to appear at this pin.
5	4, 5	AV_{DD}	S	Analog Supply Voltage, 3 V or 5 V
6	6, 7, 8	AGND	S	Analog Ground. Ground reference pin for the analog circuitry.
7	9	REFIN(–)	I	Reference Input, Negative Terminal
8	10	REFIN(+)	I	Reference Input, Positive Terminal
13	15	$\overline{SS}$	I	Slave Select Input for the SPI Interface. A weak pull-up is present on this pin.
14	16	MISO	I/O	Master Input/Slave Output for the SPI Interface. There is a weak pull-up on this input pin.
15	17	RESET	I	Reset Input. A high level on this pin for 16 core clock cycles while the oscillator is running resets the device. There is an internal weak pull-down and a Schmitt trigger input stage on this pin.
16–19, 22–25	18–21, 24–27	P3.0–P3.7	I/O	Bidirectional port pins with internal pull-up resistors. Port 3 pins that have 1s written to them are pulled high by the internal pull-up resistors, and in that state 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 for two core clock periods of the instruction cycle. Port 3 pins also have various secondary functions including:
		P3.0/RXD	I/O	Receiver Data for UART Serial Port
		P3.1/TXD	I/O	Transmitter Data for UART Serial Port
		P3.2/$\overline{INT0}$	I/O	External Interrupt 0. This pin can also be used as a gate control input to Timer 0.
		P3.3/$\overline{INT1}$	I/O	External Interrupt 1. This pin can also be used as a gate control input to Timer 1.
		P3.4/T0/ PWMCLK	I/O	Timer/Counter 0 External Input. If the PWM is enabled, an external clock may be input at this pin.
		P3.5/T1	I/O	Timer/Counter 1 External Input
		P3.6/$\overline{WR}$	I/O	External Data Memory Write Strobe. Latches the data byte from Port 0 into an external data memory.
		P3.7/$\overline{RD}$	I/O	External Data Memory Read Strobe. Enables the data from an external data memory to Port 0.
20, 34, 48	22, 36, 51	DV_{DD}	S	Digital Supply, 3 V or 5 V.
21, 35, 47	23, 37, 38, 50	DGND	S	Digital Ground. Ground reference point for the digital circuitry.
26		SCLOCK	I/O	Serial Interface Clock for Either the I^2C or SPI Interface. As an input, this pin is a Schmitt-triggered input and a weak internal pull-up is present on this pin unless it is outputting logic low. This pin can also be directly controlled in software as a digital output pin.
27		MOSI/SDATA	I/O	Serial Data I/O for the I^2C Interface or Master Output/Slave Input for the SPI Interface. A weak internal pull-up is present on this pin unless it is outputting logic low. This pin can also be directly controlled in software as a digital output pin.

PIN FUNCTION DESCRIPTIONS (continued)

Pin No. 52-Lead MQFP	Pin No. 56-Lead CSP	Mnemonic	Type*	Description
28–31 36–39	30–33 39–42	P2.0–P2.7 (A8–A15) (A16–A23)	I/O	Port 2 is a bidirectional port with internal pull-up resistors. Port 2 pins that have 1s written to them are pulled high by the internal pull-up resistors, and in that state can be used as inputs. As inputs, Port 2 pins being pulled externally low will source current because of the internal pull-up resistors. Port 2 emits the high order address bytes during fetches from external program memory and middle and high order address bytes during accesses to the 24-bit external data memory space.
32	34	XTAL1	I	Input to the Crystal Oscillator Inverter
33	35	XTAL2	O	Output from the Crystal Oscillator Inverter. (See "Hardware Design Considerations" for description.)
40	43	$\overline{EA}$	I/O	External Access Enable, Logic Input. When held high, this input enables the device to fetch code from internal program memory locations 0000h to F7FFh. When held low, this input enables the device to fetch all instructions from external program memory. To determine the mode of code execution, i.e., internal or external, the $\overline{EA}$ pin is sampled at the end of an external RESET assertion or as part of a device power cycle. $\overline{EA}$ may also be used as an external emulation I/O pin, and therefore the voltage level at this pin must not be changed during normal mode operation as it may cause an emulation interrupt that will halt code execution.
41	44	$\overline{PSEN}$	O	Program Store Enable, Logic Output. This output is a control signal that enables the external program memory to the bus during external fetch operations. It is active every six oscillator periods except during external data memory accesses. This pin remains high during internal program execution. $\overline{PSEN}$ can also be used to enable serial download mode when pulled low through a resistor at the end of an external RESET assertion or as part of a device power cycle.
42	45	ALE	O	Address Latch Enable, Logic Output. This output is used to latch the low byte (and page byte for 24-bit data address space accesses) of the address to external memory during external code or data memory access cycles. It is activated every six oscillator periods except during an external data memory access. It can be disabled by setting the PCON.4 bit in the PCON SFR.
43–46 49–52	46–49 52–55	P0.0–P0.7 (AD0–AD3) (AD4–AD7)	I/O	P0.0–P0.7, these pins are part of Port0, which is an 8-bit, open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them float and in that state can be used as high impedance inputs. An external pull-up resistor will be required on P0 outputs to force a valid logic high level externally. Port 0 is also the multiplexed low-order address and databus during accesses to external program or data memory. In this application, it uses strong internal pull-ups when emitting 1s.

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

MEMORY ORGANIZATION

The ADuC834 contains four different memory blocks, namely:

- 62 Kbytes of On-Chip Flash/EE Program Memory
- 4 Kbytes of On-Chip Flash/EE Data Memory
- 256 bytes of General-Purpose RAM
- 2 Kbytes of Internal XRAM

(1) Flash/EE Program Memory

The ADuC834 provides 62 Kbytes of Flash/EE program memory to run user code. The user can choose to run code from this internal memory or run code from an external program memory.

If the user applies power or resets the device while the $\overline{EA}$ pin is pulled low externally, the part will execute code from the external program space; otherwise, if $\overline{EA}$ is pulled high externally, the part defaults to code execution from its internal 62 Kbytes of Flash/EE program memory.

Unlike the ADuC824, where code execution can overflow from the internal code space to external code space once the PC becomes greater than 1FFFH, the ADuC834 does not support the rollover from F7FFH in internal code space to F800H in external code space. Instead, the 2048 bytes between F800H and FFFFH will appear as NOP instructions to user code.

Permanently embedded firmware allows code to be serially downloaded to the 62 Kbytes of internal code space via the UART serial port while the device is in-circuit. No external hardware is required.

56 Kbytes of the program memory can be reprogrammed during runtime; thus the code space can be upgraded in the field using a user defined protocol or it can be used as a data memory. This will be discussed in more detail in the Flash/EE Memory section of the data sheet.

(2) Flash/EE Data Memory

4 Kbytes of Flash/EE Data Memory are available to the user and can be accessed indirectly via a group of 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.

(3) General-Purpose RAM

The general-purpose RAM is divided into two separate memories, namely the upper and the lower 128 bytes of RAM. The lower 128 bytes of RAM can be accessed through direct or indirect addressing; the upper 128 bytes of RAM can only be accessed through indirect addressing as it shares the same address space as the SFR space, which can only be accessed through direct addressing.

The lower 128 bytes of internal data memory are mapped as shown in Figure 2. The lowest 32 bytes are grouped into four banks of eight registers addressed as R0 through R7. The next 16 bytes (128 bits), locations 20H through 2FH above the register banks, form a block of directly addressable bit locations at bit addresses 00H through 7FH. The stack can be located anywhere in the internal memory address space, and the stack depth can be expanded up to 2048 bytes.

Reset initializes the stack pointer to location 07H. Any CALL or PUSH pre-increments the SP before loading the stack. Therefore, loading the stack starts from locations 08H, which is also the first register (R0) of register bank 1. Thus, if one is going to use more than one register bank, the stack pointer should be initialized to an area of RAM not used for data storage.

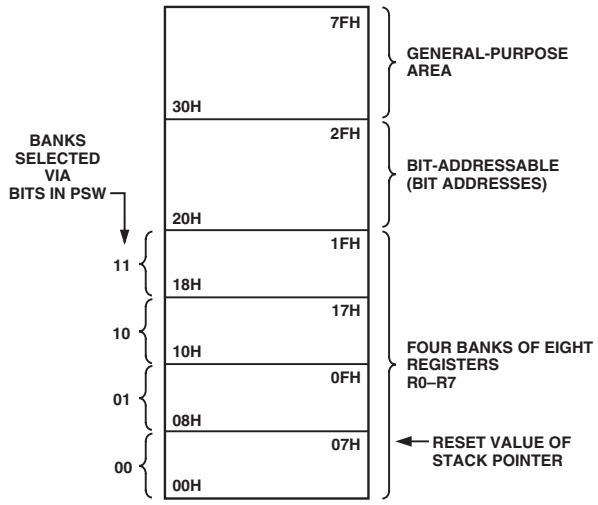


Figure 2. Lower 128 Bytes of Internal Data Memory

(4) Internal XRAM

The ADuC834 contains 2 Kbytes of on-chip extended data memory. This memory, although on-chip, is accessed via the MOVX instruction. The 2 Kbytes of internal XRAM are mapped into the bottom 2 Kbytes of the external address space if the CFG834.0 bit is set. Otherwise, access to the external data memory will occur just like a standard 8051.

Even with the CFG834.0 bit set, access to the external XRAM will occur once the 24-bit DPTR is greater than 0007FFH.

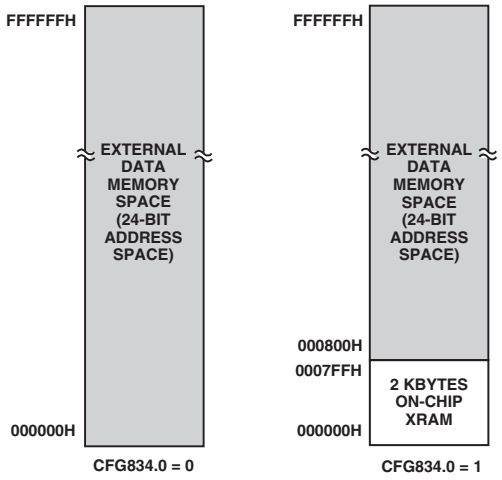


Figure 3. Internal and External XRAM

GENERAL NOTES PERTAINING TO THIS DATA SHEET

1. SET implies a Logic 1 state and CLEARED implies a Logic 0 state unless otherwise stated.
2. SET and CLEARED also imply that the bit is set or automatically cleared by the ADuC834 hardware unless otherwise stated.
3. User software should not write 1s to reserved or unimplemented bits as they may be used in future products.
4. Any pin numbers used throughout this data sheet refer to the 52-lead MQFP package, unless otherwise stated.

When accessing the internal XRAM, the P0 and P2 port pins, as well as the $\overline{RD}$ and $\overline{WR}$ strobes, will not be output as per a standard 8051 MOVX instruction. This allows the user to use these port pins as standard I/O.

The upper 1792 bytes of the internal XRAM can be configured to be used as an extended 11-bit stack pointer. By default, the stack will operate exactly like an 8052 in that it will roll over from FFH to 00H in the general-purpose RAM. On the ADuC834 however, it is possible (by setting CFG834.7) to enable the 11-bit extended stack pointer. In this case, the stack will roll over from FFH in RAM to 0100H in XRAM. The 11-bit stack pointer is visible in the SP and SPH SFRs. The SP SFR is located at 81H as with a standard 8052. The SPH SFR is located at B7H. The 3 LSBs of this SFR contain the three extra bits necessary to extend the 8-bit stack pointer into an 11-bit stack pointer.

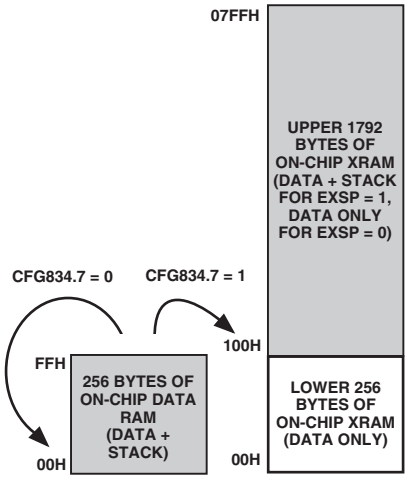


Figure 4. Extended Stack Pointer Operation

External Data Memory (External XRAM)
Just like a standard 8051 compatible core, the ADuC834 can access external data memory using a MOVX instruction. The MOVX instruction automatically outputs the various control strobes required to access the data memory.

The ADuC834 however, can access up to 16 Mbytes of external data memory. This is an enhancement of the 64 Kbytes external data memory space available on a standard 8051 compatible core.

The external data memory is discussed in more detail in the ADuC834 Hardware Design Considerations section.

SPECIAL FUNCTION REGISTERS (SFRS)
The SFR space is mapped into 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 ADuC834 via the SFR area is shown in Figure 5.

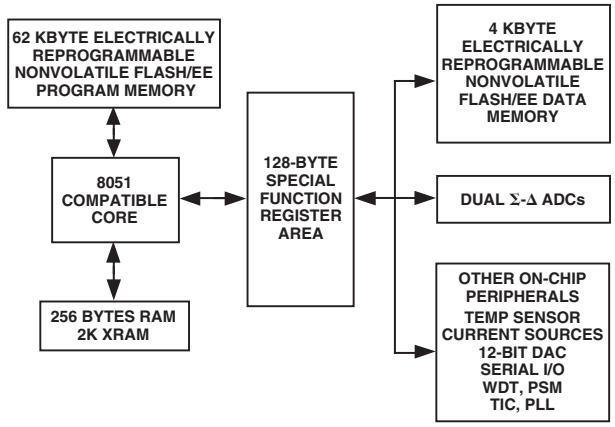


Figure 5. Programming Model

All registers, except the Program Counter (PC) 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.

Accumulator SFR (ACC)
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 (B)
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.

Data Pointer (DPTR)
The Data Pointer is made up of three 8-bit registers, named DPP (page byte), DPH (high byte) and DPL (low byte). These are used to provide memory addresses for internal and external code access and external data access. It may be manipulated as a 16-bit register (DPTR = DPH, DPL), although INC DPTR instructions will automatically carry over to DPP, or as three independent 8-bit registers (DPP, DPH, DPL).

The ADuC834 supports dual data pointers. Refer to the Dual Data Pointer section in this data sheet.

Stack Pointer (SP and SPH)
The SP SFR 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.

As mentioned earlier, the ADuC834 offers an extended 11-bit stack pointer. The three extra bits to make up the 11-bit stack pointer are the 3 LSBs of the SPH byte located at B7H.

Program Status Word (PSW)
The PSW SFR 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

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

The TIC (wake-up/RTC timer) can be used to accurately wake up the ADuC834 from power-down at regular intervals. To use the TIC to wake up the ADuC834 from power-down, the OSC_PD bit in the PLLCON SFR must be clear and the TIC must be enabled.

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

Table II. PCON SFR Bit Designations

Bit	Name	Description
7	SMOD	Double UART Baud Rate
6	SERIPD	SPI Power-Down Interrupt Enable
5	INT0PD	$\overline{\text{INT0}}$ Power-Down Interrupt Enable
4	ALEOFF	Disable ALE Output
3	GF1	General-Purpose Flag Bit
2	GF0	General-Purpose Flag Bit
1	PD	Power-Down Mode Enable
0	IDL	Idle Mode Enable

ADuC834 CONFIGURATION SFR (CFG834)
The CFG834 SFR contains the necessary bits to configure the internal XRAM and the extended SP. By default it configures the user into 8051 mode, i.e., extended SP is disabled, internal XRAM is disabled.

SFR Address	AFH
Power-On Default Value	00H
Bit Addressable	No

Table III. CFG834 SFR Bit Designations

Bit	Name	Description
7	EXSP	Extended SP Enable. If this bit is set, the stack will roll over from SPH/SP = 00FFH to 0100H. If this bit is clear, the SPH SFR will be disabled and the stack will roll over from SP = FFH to SP = 00H
6	—	Reserved for Future Use
5	—	Reserved for Future Use
4	—	Reserved for Future Use
3	—	Reserved for Future Use
2	—	Reserved for Future Use
1	—	Reserved for Future Use
0	XRAMEN	XRAM Enable Bit. If this bit is set, the internal XRAM will be mapped into the lower 2 Kbytes of the external address space. If this bit is clear, the internal XRAM will not be accessible and the external data memory will be mapped into the lower 2 Kbytes of external data memory. (See Figure 3.)

COMPLETE SFR MAP

Figure 6 shows a full SFR memory map and the SFR contents after 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 that are 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 1	SPR1 F9H 0	SPR0 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
RDY0 DFH 0	RDY1 DEH 0	CAL DDH 0	NOXREF DCH 0	ERR0 DBH 0	ERR1 DAH 0	D9H 0	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 1	WDIR C3H 0	WDS C2H 0	WDE C1H 0	WDWR C0H 0	BITS
BFH 0	PADC BEH 0	PT2 BDH 0	PS BCH 0	PT1 BBH 0	PX1 BAH	PT0 B9H 0	PX0 B8H 0	BITS
$\overline{RD}$ B7H 1	$\overline{WR}$ B6H 1	T1 B5H 1	T0 B4H 1	$\overline{INT1}$ B3H 1	$\overline{INT0}$ B2H 1	TXD B1H 1	RXD B0H 1	BITS
EA AFH 0	EADC AEH 0	ET2 ADH 0	ES ACH 0	ET1 ABH 0	EX1 AAH 0	ET0 A9H 0	EX0 A8H 0	BITS
A7H 1	A6H 1	A5H 1	A4H 1	A3H 1	A2H 1	A1H 1	A0H 1	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 1	96H 1	95H 1	94H 1	93H 1	92H 1	T2EX 91H 1	T2 90H 1	BITS
TF1 8FH 0	TR1 8EH 0	TF0 8DH 0	TR0 8CH 0	IE1 8BH 0	IT1 8AH 0	IE0 89H 0	IT0 88H 0	BITS
87H 1	86H 1	85H 1	84H 1	83H 1	82H 1	81H 1	80H 1	BITS

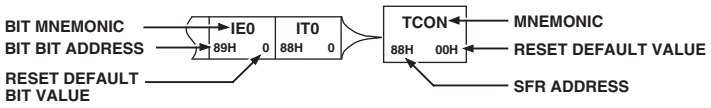
SPICON F8H 04H	RESERVED	RESERVED	DACL FBH 00H	DACH FCH 00H	DACCON FDH 00H	RESERVED	RESERVED
B F0H 00H	RESERVED	RESERVED	NOT USED	RESERVED	RESERVED	RESERVED	SPIDAT F7H 00H
I2CCON E8H 00H	GN0L[1] E9H 55H	GN0M[1] EAH 55H	GN0H[1] EBH 53H	GN1L[1] ECH 9AH	GN1H[1] EDH 59H	RESERVED	RESERVED
ACC E0H 00H	OF0L E1H 00H	OF0M E2H 00H	OF0H E3H 80H	OF1L E4H 00H	OF1H E5H 80H	RESERVED	RESERVED
ADCSTAT D8H 00H	ADC0L D9H 00H	ADC0M DAH 00H	ADC0H DBH 00H	ADC1L DCH 00H	ADC1H DDH 00H	RESERVED	PSMCON DFH DEH
PSW D0H 00H	ADCMODE D1H 00H	ADC0CON D2H 07H	ADC1CON D3H 00H	SF D4H 45H	ICON D5H 00H	RESERVED	PLLCON D7H 03H
T2CON C8H 00H	RESERVED	RCAP2L CAH 00H	RCAP2H CBH 00H	TL2 CCH 00H	TH2 CDH 00H	RESERVED	RESERVED
WDCON C0H 10H	RESERVED	CHIPID C2H 2×H	RESERVED	RESERVED	RESERVED	EADRL C6H 00H	EADRH C7H 00H
IP B8H 00H	ECON B9H 00H	RESERVED	RESERVED	EDATA1 BCH 00H	EDATA2 BDH 00H	EDATA3 BEH 00H	EDATA4 BFH 00H
P3 B0H FFH	PWM0L B1H 00H	PWM0H B2H 00H	PWM1L B3H 00H	PWM1H B4H 00H	RESERVED	RESERVED	SPH B7H 00H
IE A8H 00H	IEIP2 A9H A0H	RESERVED	RESERVED	RESERVED	RESERVED	PWMCON AEH 00H	CFG834 AFH 00H
P2 A0H FFH	TIMECON A1H 00H	HTHSEC[2] A2H 00H	SEC[2] A3H 00H	MIN[2] A4H 00H	HOUR[2] A5H 00H	INTVAL A6H 00H	DPCON A7H 00H
SCON 98H 00H	SBUF 99H 00H	RESERVED	RESERVED	NOT USED	T3FD 9DH 00H	T3CON 9EH 00H	RESERVED
P1 90H FFH	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED
TCON 88H 00H	TMOD 89H 00H	TL0 8AH 00H	TL1 8BH 00H	TH0 8CH 00H	TH1 8DH 00H	RESERVED	RESERVED
P0 80H FFH	SP 81H 07H	DPL 82H 00H	DPH 83H 00H	DPP 84H 00H	RESERVED	RESERVED	PCON 87H 00H

NOTES

[1]CALIBRATION COEFFICIENTS ARE PRECONFIGURED AT POWER-UP TO FACTORY CALIBRATED VALUES.

[2]THESE SFRS MAINTAIN THEIR PRERESET VALUES AFTER A RESET IF TIMECON.0 = 1.

SFR MAP KEY:



SFR NOTE:

SFRs WHOSE ADDRESSES END IN 0H OR 8H ARE BIT-ADDRESSABLE.

Figure 6. Special Function Register Locations and Their Reset Default Values

ADC SFR INTERFACE

Both ADCs are controlled and configured via a number of SFRs that are mentioned here and described in more detail in the following pages.

ADCSTAT	ADC Status Register. Holds general status of the primary and auxiliary ADCs.
ADCMODE	ADC Mode Register. Controls general modes of operation for primary and auxiliary ADCs.
ADC0CON	Primary ADC Control Register. Controls specific configuration of primary ADC.
ADC1CON	Auxiliary ADC Control Register. Controls specific configuration of auxiliary ADC.
SF	Sinc Filter Register. Configures the decimation factor for the $Sinc^3$ filter and thus the primary and auxiliary ADC update rates.
ICON	Current Source Control Register. Allows user control of the various on-chip current source options.
ADC0L/M/H	Primary ADC 24-bit conversion result is held in these three 8-bit registers.
ADC1L/H	Auxiliary ADC 16-bit conversion result is held in these two 8-bit registers.
OF0L/M/H	Primary ADC 24-bit Offset Calibration Coefficient is held in these three 8-bit registers.
OF1L/H	Auxiliary ADC 16-bit Offset Calibration Coefficient is held in these two 8-bit registers.
GN0L/M/H	Primary ADC 24-bit Gain Calibration Coefficient is held in these three 8-bit registers.
GN1L/H	Auxiliary ADC 16-bit Gain Calibration Coefficient is held in these two 8-bit registers.

ADCSTAT—(ADC Status Register)

This SFR reflects the status of both ADCs including data ready, calibration, and various (ADC-related) error and warning conditions including reference detect and conversion overflow/underflow flags.

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

Table IV. ADCSTAT SFR Bit Designations

Bit	Name	Description
7	RDY0	Ready Bit for primary ADC. Set by hardware on completion of ADC conversion or calibration cycle. Cleared directly by the user or indirectly by write to the mode bits to start another primary ADC conversion or calibration. The primary ADC is inhibited from writing further results to its data or calibration registers until the RDY0 bit is cleared.
6	RDY1	Ready Bit for auxiliary ADC. Same definition as RDY0 referred to the auxiliary ADC.
5	CAL	Calibration Status Bit. Set by hardware on completion of calibration. Cleared indirectly by a write to the mode bits to start another ADC conversion or calibration.
4	NOXREF	No External Reference Bit *(only active if primary or auxiliary ADC is active)*. Set to indicate that one or both of the REFIN pins is floating or the applied voltage is below a specified threshold. When Set, conversion results are clamped to all ones, if using external reference. Cleared to indicate valid V_{REF}.
3	ERR0	Primary ADC Error Bit. Set by hardware to indicate that the result written to the primary ADC data registers has been clamped to all zeros or all ones. After a calibration, this bit also flags error conditions that caused the calibration registers not to be written. Cleared by a write to the mode bits to initiate a conversion or calibration.
2	ERR1	Auxiliary ADC Error Bit. Same definition as ERR0 referred to the auxiliary ADC.
1	—	Reserved for Future Use
0	—	Reserved for Future Use

ADCMODE (ADC Mode Register)
Used to control the operational mode of both ADCs.

SFR Address	D1H
Power-On Default Value	00H
Bit Addressable	No

Table V. ADCMODE SFR Bit Designations

Bit	Name	Description			
7	—	Reserved for Future Use			
6	—	Reserved for Future Use			
5	ADC0EN	Primary ADC Enable. Set by the user to enable the primary ADC and place it in the mode selected in MD2–MD0 below. Cleared by the user to place the primary ADC in power-down mode.			
4	ADC1EN	Auxiliary ADC Enable. Set by the user to enable the auxiliary ADC and place it in the mode selected in MD2–MD0 below. Cleared by the user to place the auxiliary ADC in power-down mode.			
3	—	Reserved for Future Use			
2	MD2	Primary and auxiliary ADC Mode bits.			
1	MD1	These bits select the operational mode of the enabled ADC as follows:			
0	MD0	MD2	MD1	MD0	
		0	0	0	ADC Power-Down Mode (Power-On Default)
		0	0	1	Idle Mode. In Idle Mode, the ADC filter and modulator are held in a reset state although the modulator clocks are still provided.
		0	1	0	Single Conversion Mode. In Single Conversion Mode, a single conversion is performed on the enabled ADC. On completion of the conversion, the ADC data registers (ADC0H/M/L and/or ADC1H/L) are updated, the relevant flags in the ADCSTAT SFR are written, and power-down is re-entered with the MD2–MD0 accordingly being written to 000.
		0	1	1	Continuous Conversion. In Continuous Conversion Mode, the ADC data registers are regularly updated at the selected update rate (see SF Register).
		1	0	0	Internal Zero-Scale Calibration. Internal short automatically connected to the enabled ADC input(s).
		1	0	1	Internal Full-Scale Calibration Internal or External V_{REF} (as determined by XREF0 and XREF1 bits in ADC0/1CON) is automatically connected to the enabled ADC input(s) for this calibration.
		1	1	0	System Zero-Scale Calibration. User should connect system zero-scale input to the enabled ADC input(s) as selected by CH1/CH0 and ACH1/ACH0 bits in the ADC0/1CON Register.
		1	1	1	System Full-Scale Calibration. User should connect system full-scale input to the enabled ADC input(s) as selected by CH1/CH0 and ACH1/ACH0 bits in the ADC0/1CON Register.

NOTES
1. Any change to the MD bits will immediately reset both ADCs. A write to the MD2–0 Bits with no change is also treated as a reset. (See exception to this in Note 3 below.)
2. If ADC0CON is written when ADC0EN = 1, or if ADC0EN is changed from 0 to 1, then both ADCs are also immediately reset. In other words, the primary ADC is given priority over the auxiliary ADC and any change requested on the primary ADC is immediately responded to.
3. On the other hand, if ADC1CON is written or if ADC1EN is changed from 0 to 1, only the auxiliary ADC is reset. For example, if the primary ADC is continuously converting when the auxiliary ADC change or enable occurs, the primary ADC continues undisturbed. Rather than allow the auxiliary ADC to operate with a phase difference from the primary ADC, the auxiliary ADC will fall into step with the outputs of the primary ADC. The result is that the first conversion time for the auxiliary ADC will be delayed up to three outputs while the auxiliary ADC update rate is synchronized to the primary ADC.
4. Once ADCMODE has been written with a calibration mode, the RDY0/1 bits (ADCSTAT) are immediately reset and the calibration commences. On completion, the appropriate calibration registers are written, the relevant bits in ADCSTAT are written, and the MD2–0 bits are reset to 000 to indicate the ADC is back in power-down mode.
5. Any calibration request of the auxiliary ADC while the temperature sensor is selected will fail to complete. Although the RDY1 bit will be set at the end of the calibration cycle, no update of the calibration SFRs will take place and the ERR1 bit will be set.
6. Calibrations are performed at maximum SF (see SF SFR) value guaranteeing optimum calibration operation.

ADC0CON (Primary ADC Control Register) and ADC1CON (Auxiliary ADC Control Register)

The ADC0CON and ADC1CON SFRs are used to configure the primary and auxiliary ADC for reference and channel selection, unipolar or bipolar coding and, in the case of the primary ADC, for range (the auxiliary ADC operates on a fixed input range of $\pm V_{REF}$).

ADC0CON	**Primary ADC Control SFR**	**ADC1CON**	**Auxiliary ADC Control SFR**
SFR Address	D2H	SFR Address	D3H
Power-On Default Value	07H	Power-On Default Value	00H
Bit Addressable	No	Bit Addressable	No

Table VI. ADC0CON SFR Bit Designations

Bit	Name	Description
7	—	Reserved for Future Use
6	XREF0	Primary ADC External Reference Select Bit. Set by user to enable the primary ADC to use the external reference via REFIN(+)/REFIN(–). Cleared by user to enable the primary ADC to use the internal band gap reference (V_{REF} = 1.25 V).
5	CH1	Primary ADC Channel Selection Bits
4	CH0	Written by the user to select the differential input pairs used by the primary ADC as follows: CH1 CH0 Positive Input Negative Input 0 0 AIN1 AIN2 0 1 AIN3 AIN4 1 0 AIN2 AIN2 (Internal Short) 1 1 AIN3 AIN2
3	UNI0	Primary ADC Unipolar Bit. Set by user to enable unipolar coding, i.e., zero differential input will result in 000000H output. Cleared by user to enable bipolar coding, i.e., zero differential input will result in 800000H output.
2	RN2	Primary ADC Range Bits.
1	RN1	Written by the user to select the primary ADC input range as follows:
0	RN0	RN2 RN1 RN0 Selected Primary ADC Input Range (V_{REF} = 2.5 V) 0 0 0 ±20 mV (0 mV–20 mV in Unipolar Mode) 0 0 1 ±40 mV (0 mV–40 mV in Unipolar Mode) 0 1 0 ±80 mV (0 mV–80 mV in Unipolar Mode) 0 1 1 ±160 mV (0 mV–160 mV in Unipolar Mode) 1 0 0 ±320 mV (0 mV–320 mV in Unipolar Mode) 1 0 1 ±640 mV (0 mV–640 mV in Unipolar Mode) 1 1 0 ±1.28 V (0 V–1.28 V in Unipolar Mode) 1 1 1 ±2.56 V (0 V–2.56 V in Unipolar Mode)

Table VII. ADC1CON SFR Bit Designations

Bit	Name	Description
7	—	Reserved for Future Use
6	XREF1	Auxiliary ADC External Reference Bit. Set by user to enable the auxiliary ADC to use the external reference via REFIN(+)/REFIN(–). Cleared by user to enable the auxiliary ADC to use the internal band gap reference.
5	ACH1	Auxiliary ADC Channel Selection Bits.
4	ACH0	Written by the user to select the single-ended input pins used to drive the auxiliary ADC as follows: ACH1 ACH0 Positive Input Negative Input 0 0 AIN3 AGND 0 1 AIN4 AGND 1 0 Temp Sensor AGND (Temp Sensor routed to the ADC input) 1 1 AIN5 AGND
3	UNI1	Auxiliary ADC Unipolar Bit. Set by user to enable unipolar coding, i.e., zero input will result in 0000H output. Cleared by user to enable bipolar coding, i.e., zero input will result in 8000H output.
2	—	Reserved for Future Use
1	—	Reserved for Future Use
0	—	Reserved for Future Use

NOTES

1. When the temperature sensor is selected, user code must select internal reference via XREF1 bit above and clear the UNI1 bit (ADC1CON.3) to select bipolar coding.
2. The temperature sensor is factory calibrated to yield conversion results 8000H at 0°C.
3. A +1°C change in temperature will result in a +1 LSB change in the ADC1H Register ADC conversion result.

ADC0H/ADC0M/ADC0L (Primary ADC Conversion Result Registers)
These three 8-bit registers hold the 24-bit conversion result from the primary ADC.

SFR Address	ADC0H	High Data Byte	DBH
	ADC0M	Middle Data Byte	DAH
	ADC0L	Low Data Byte	D9H
Power-On Default Value	00H	ADC0H, ADC0M, ADC0L	
Bit Addressable	No	ADC0H, ADC0M, ADC0L	

ADC1H/ADC1L (Auxiliary ADC Conversion Result Registers)
These two 8-bit registers hold the 16-bit conversion result from the auxiliary ADC.

SFR Address	ADC1H	High Data Byte	DDH
	ADC1L	Low Data Byte	DCH
Power-On Default Value	00H	ADC1H, ADC1L	
Bit Addressable	No	ADC1H, ADC1L	

OF0H/OF0M/OF0L (Primary ADC Offset Calibration Registers*)
These three 8-bit registers hold the 24-bit offset calibration coefficient for the primary ADC. These registers are configured at power-on with a factory default value of 800000H. However, these bytes will be automatically overwritten if an internal or system zero-scale calibration of the primary ADC is initiated by the user via MD2–0 bits in the ADCMODE Register.

SFR Address	OF0H	Primary ADC Offset Coefficient High Byte	E3H
	OF0M	Primary ADC Offset Coefficient Middle Byte	E2H
	OF0L	Primary ADC Offset Coefficient Low Byte	E1H
Power-On Default Value	800000H	OF0H, OF0M, OF0L, respectively	
Bit Addressable	No	OF0H, OF0M, OF0L	

OF1H/OF1L (Auxiliary ADC Offset Calibration Registers*)
These two 8-bit registers hold the 16-bit offset calibration coefficient for the auxiliary ADC. These registers are configured at power-on with a factory default value of 8000H. However, these bytes will be automatically overwritten if an internal or system zero-scale calibration of the auxiliary ADC is initiated by the user via the MD2–0 bits in the ADCMODE Register.

SFR Address	OF1H	Auxiliary ADC Offset Coefficient High Byte	E5H
	OF1L	Auxiliary ADC Offset Coefficient Low Byte	E4H
Power-On Default Value	8000H	OF1H and OF1L, respectively	
Bit Addressable	No	OF1H, OF1L	

GN0H/GN0M/GN0L (Primary ADC Gain Calibration Registers*)
These three 8-bit registers hold the 24-bit gain calibration coefficient for the primary ADC. These registers are configured at power-on with a factory-calculated internal full-scale calibration coefficient. Every device will have an individual coefficient. However, these bytes will be automatically overwritten if an internal or system full-scale calibration of the primary ADC is initiated by the user via MD2–0 bits in the ADCMODE Register.

SFR Address	GN0H	Primary ADC Gain Coefficient High Byte	EBH
	GN0M	Primary ADC Gain Coefficient Middle Byte	EAH
	GN0L	Primary ADC Gain Coefficient Low Byte	E9H
Power-On Default Value		Configured at Factory Final Test; see Notes above.	
Bit Addressable	No	GN0H, GN0M, GN0L	

GN1H/GN1L (Auxiliary ADC Gain Calibration Registers*)
These two 8-bit registers hold the 16-bit gain calibration coefficient for the auxiliary ADC. These registers are configured at power-on with a factory-calculated internal full-scale calibration coefficient. Every device will have an individual coefficient. However, these bytes will be automatically overwritten if an internal or system full-scale calibration of the auxiliary ADC is initiated by the user via MD2–0 bits in the ADCMODE Register.

SFR Address	GN1H	Auxiliary ADC Gain Coefficient High Byte	EDH
	GN1L	Auxiliary ADC Gain Coefficient Low Byte	ECH
Power-On Default Value		Configured at Factory Final Test; see Notes above.	
Bit Addressable	No	GN1H, GN1L	

*These registers can be overwritten by user software only if Mode bits MD0–2 (ADCMODE SFR) are zero.

SF (Sinc Filter Register)
The number in this register sets the decimation factor and thus the output update rate for the primary and auxiliary ADCs. This SFR cannot be written by user software while either ADC is active. The update rate applies to both primary and auxiliary ADCs and is calculated as follows:

$$f_{ADC} = \frac{1}{3} \times \frac{1}{8 \times SF} \times f_{MOD}$$

Where: f_{ADC} = ADC Output Update Rate

f_{MOD} = Modulator Clock Frequency = 32.768 kHz

SF = Decimal Value of SF Register

The allowable range for SF is 0DH to FFH. Examples of *SF* values and corresponding conversion update rates (f_{ADC}) and conversion times (t_{ADC}) are shown in Table VIII. The power-on default value for the SF Register is 45H, resulting in a default ADC update rate of just under 20 Hz. Both ADC inputs are chopped to minimize offset errors, which means that the settling time for a single conversion, or the time to a first conversion result in Continuous Conversion mode, is 2 × t_{ADC}. As mentioned earlier, all calibration cycles will be carried out automatically with a maximum, i.e., FFH, SF value to ensure optimum calibration performance. Once a calibration cycle has completed, the value in the SF Register will be that programmed by user software.

Table VIII. SF SFR Bit Designations

SF(dec)	SF(hex)	f_{ADC}(Hz)	t_{ADC}(ms)
13	0D	105.3	9.52
69	45	19.79	50.34
255	FF	5.35	186.77

ICON (Current Sources Control Register)
Used to control and configure the various excitation and burnout current source options available on-chip.

SFR Address D5H
Power-On Default Value 00H
Bit Addressable No

Table IX. ICON SFR Bit Designations

Bit	Name	Description
7	—	Reserved for Future Use
6	BO	Burnout Current Enable Bit. Set by user to enable both transducer burnout current sources in the primary ADC signal paths. Cleared by user to disable both transducer burnout current sources.
5	ADC1IC	Auxiliary ADC Current Correction Bit. Set by user to allow scaling of the auxiliary ADC by an internal current source calibration word.
4	ADC0IC	Primary ADC Current Correction Bit. Set by user to allow scaling of the primary ADC by an internal current source calibration word.
3	I2PIN*	Current Source-2 Pin Select Bit. Set by user to enable current source-2 (200 μA) to external Pin 3 (P1.2/DAC/IEXC1). Cleared by user to enable current source-2 (200 μA) to external Pin 4 (P1.3/AIN5/IEXC2).
2	I1PIN*	Current Source-1 Pin Select Bit. Set by user to enable current source-1 (200 μA) to external Pin 4 (P1.3/AIN5/IEXC2). Cleared by user to enable current source-1 (200 μA) to external Pin 3 (P1.2/DAC/IEXC1).
1	I2EN	Current Source-2 Enable Bit. Set by user to turn on excitation current source-2 (200 μA). Cleared by user to turn off excitation current source-2 (200 μA).
0	I1EN	Current Source-1 Enable Bit. Set by user to turn on excitation current source-1 (200 μA). Cleared by user to turn off excitation current source-1 (200 μA).

*Both current sources can be enabled to the same external pin, yielding a 400 μA current source.

PRIMARY AND AUXILIARY ADC NOISE PERFORMANCE

Tables X, XI, and XII show the output rms noise in μV and output peak-to-peak resolution in bits (rounded to the nearest 0.5 LSB) for some typical output update rates on both the primary and auxiliary ADCs. The numbers are typical and are generated at a differential input voltage of 0 V. The output update rate is selected via the Sinc Filter (SF) SFR. It is important to note that the peak-to-peak resolution figures represent the resolution for which there will be no code flicker within a six-sigma limit.

The QuickStart Development system PC software comes complete with an ADC noise evaluation tool. This tool can be easily used with the evaluation board to see these figures from silicon.

Table X. Primary ADC, Typical Output RMS Noise (μV)
Typical Output RMS Noise vs. Input Range and Update Rate; Output RMS Noise in μV

SF Word	Data Update Rate (Hz)	Input Range							
		±20 mV	±40 mV	±80 mV	±160 mV	±320 mV	±640 mV	±1.28 V	±2.56 V
13	105.3	1.50	1.50	1.60	1.75	3.50	4.50	6.70	11.75
69	19.79	0.60	0.65	0.65	0.65	0.65	0.95	1.40	2.30
255	5.35	0.35	0.35	0.37	0.37	0.37	0.51	0.82	1.25

Table XI. Primary ADC, Peak-to-Peak Resolution (Bits)
Peak-to-Peak Resolution vs. Input Range and Update Rate; Peak-to-Peak Resolution in Bits

SF Word	Data Update Rate (Hz)	Input Range							
		±20 mV	±40 mV	±80 mV	±160 mV	±320 mV	±640 mV	±1.28 V	±2.56 V
13	105.3	12	13	14	15	15	15.5	16	16
69	19.79	13.5	14	15	16	17	17.5	18	18.5
255	5.35	14	15	16	17	18	18.5	19	19.5

Typical RMS Resolution vs. Input Range and Update Rate: RMS Resolution in Bits*

SF Word	Data Update Rate (Hz)	Input Range							
		±20 mV	±40 mV	±80 mV	±160 mV	±320 mV	±640 mV	±1.28 V	±2.56 V
13	105.3	14.7	15.7	16.7	17.7	17.7	18.2	18.7	18.7
69	19.79	16.2	16.7	17.7	18.7	19.7	20.2	20.7	21.2
255	5.35	16.7	17.7	18.7	19.7	20.7	21.2	21.7	22.2

*Based on a six-sigma limit, the rms resolution is 2.7 bits greater than the peak-to-peak resolution.

Table XII. Auxiliary ADC

Typical Output RMS Noise vs. Update Rate*
Output RMS Noise in μV

SF Word	Data Update Rate (Hz)	Input Range 2.5 V
13	105.3	10.75
69	19.79	2.00
255	5.35	1.15

*ADC converting in Bipolar mode

Peak-to-Peak Resolution vs. Update Rate[1]
Peak-to-Peak Resolution in Bits

SF Word	Data Update Rate (Hz)	Input Range 2.5 V
13	105.3	16[2]
69	19.79	16
255	5.35	16

NOTES
[1]ADC converting in Bipolar mode
[2]In Unipolar mode, peak-to-peak resolution at 105 Hz is 15 bits.

PRIMARY AND AUXILIARY ADC CIRCUIT DESCRIPTION

Overview

The ADuC834 incorporates two independent Σ-Δ ADCs (primary and auxiliary) with on-chip digital filtering intended for the measurement of wide dynamic range, low frequency signals such as those in weigh-scale, strain gage, pressure transducer, or temperature measurement applications.

Primary ADC

This ADC is intended to convert the primary sensor input. The input is buffered and can be programmed for one of eight input ranges from ±20 mV to ±2.56 V being driven from one of three differential input channel options AIN1/2, AIN3/4, or AIN3/2. The input channel is internally buffered, allowing the part to handle significant source impedances on the analog input and allowing R/C filtering (for noise rejection or RFI reduction) to be placed on the analog inputs if required. On-chip burnout currents can also be turned on. These currents can be used to check that a transducer on the selected channel is still operational before attempting to take measurements.

The ADC employs a Σ-Δ conversion technique to realize up to 24 bits of no missing codes performance. The Σ-Δ modulator converts the sampled input signal into a digital pulse train whose duty cycle contains the digital information. A Sinc^3 programmable low-pass filter is then employed to decimate the modulator output data stream to give a valid data conversion result at programmable output rates from 5.35 Hz (186.77 ms) to 105.03 Hz (9.52 ms). A chopping scheme is also employed to minimize ADC offset errors. A block diagram of the primary ADC is shown in Figure 7.

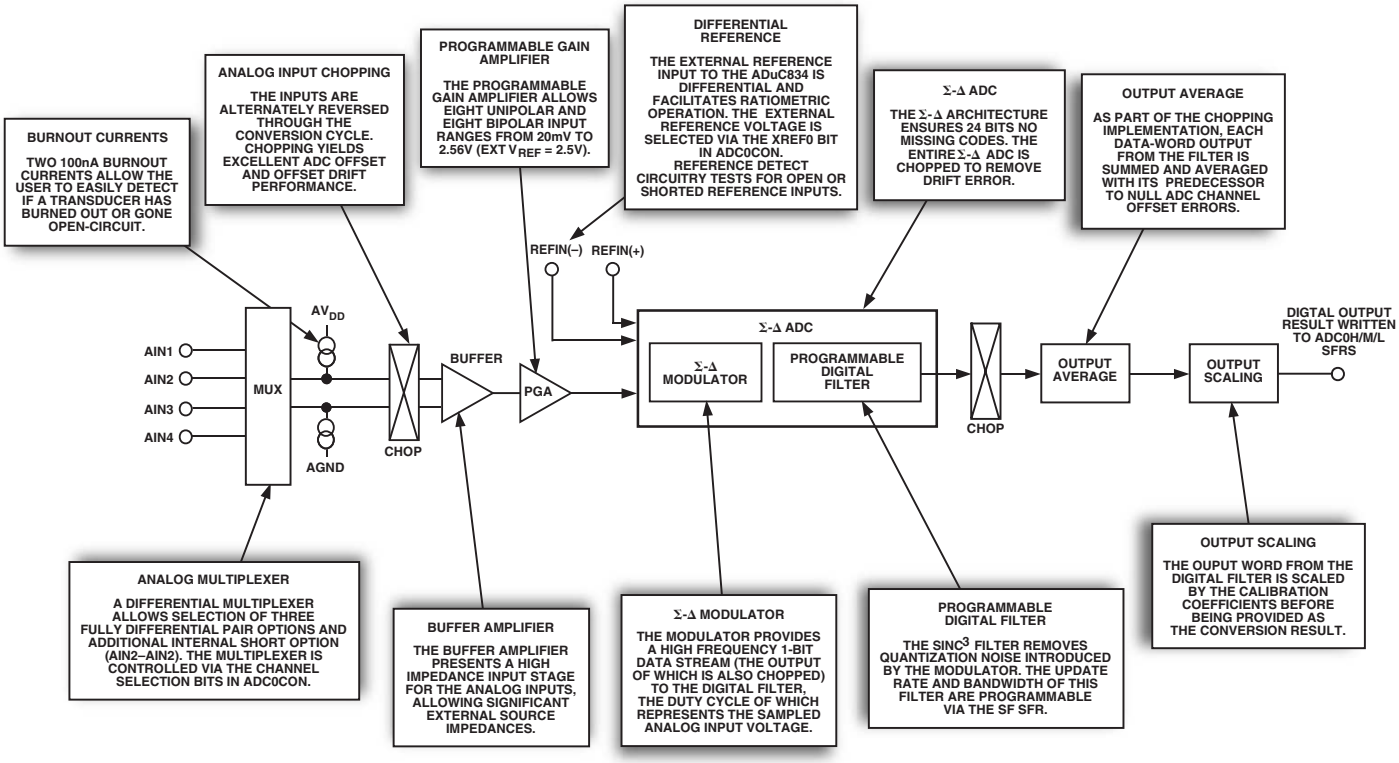


Figure 7. Primary ADC Block Diagram

Auxiliary ADC

The auxiliary ADC is intended to convert supplementary inputs such as those from a cold junction diode or thermistor. This ADC is not buffered and has a fixed input range of 0 V to 2.5 V (assuming an external 2.5 V reference). The single-ended inputs can be driven from AIN3, AIN4, or AIN5 Pins, or directly from the on-chip temperature sensor voltage. A block diagram of the auxiliary ADC is shown in Figure 8.

Analog Input Channels

The primary ADC has four associated analog input pins (labelled AIN1 to AIN4) that can be configured as two fully differential input channels. Channel selection bits in the ADC0CON SFR detailed in Table VI allow three combinations of differential pair selection as well as an additional shorted input option (AIN2–AIN2).

The auxiliary ADC has three external input pins (labelled AIN3 to AIN5) as well as an internal connection to the on-chip temperature sensor. All inputs to the auxiliary ADC are single-ended inputs referenced to the AGND on the part. Channel selection bits in the ADC1CON SFR previously detailed in Table VII allow selection of one of four inputs.

Two input multiplexers switch the selected input channel to the on-chip buffer amplifier in the case of the primary ADC and directly to the Σ-Δ modulator input in the case of the auxiliary ADC. When the analog input channel is switched, the settling time of the part must elapse before a new valid word is available from the ADC.

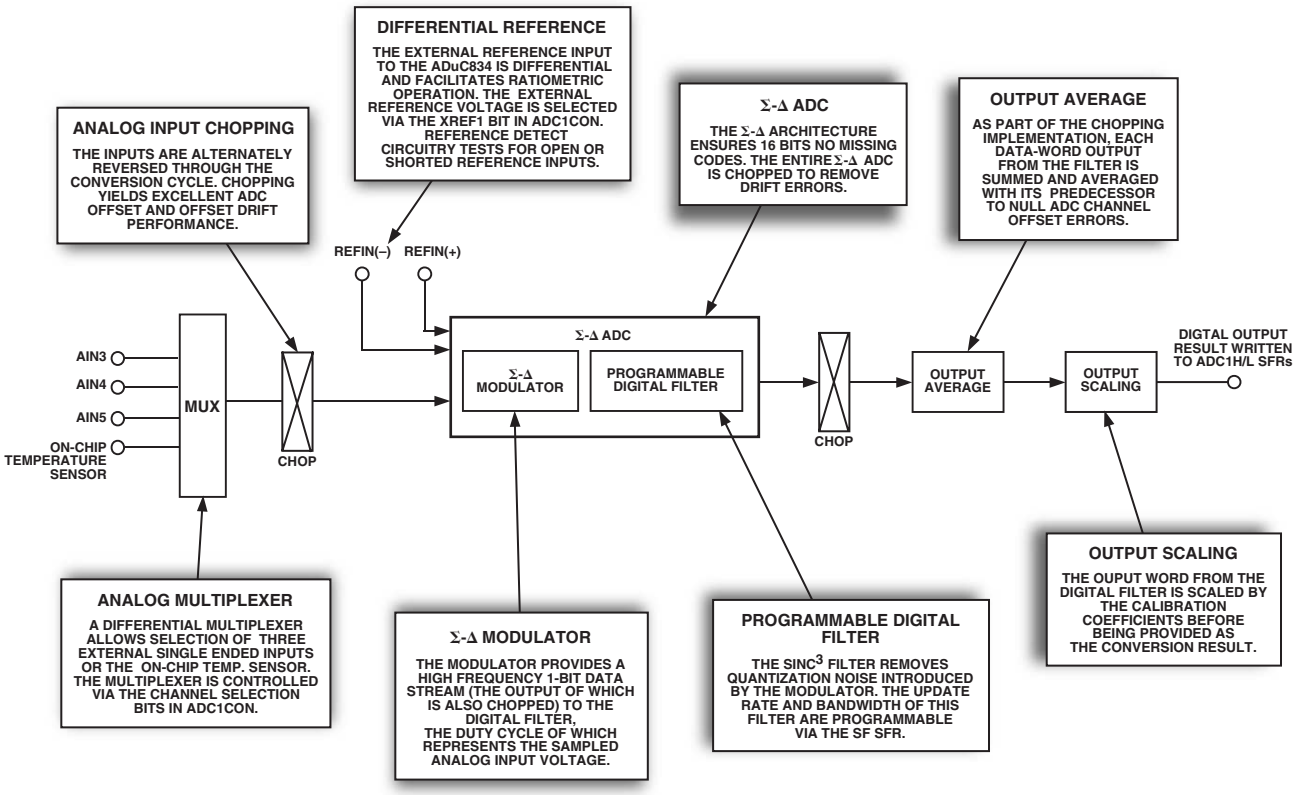


Figure 8. Auxiliary ADC Block Diagram

Primary and Auxiliary ADC Inputs

The output of the primary ADC multiplexer feeds into a high impedance input stage of the buffer amplifier. As a result, the primary ADC inputs can handle significant source impedances and are tailored for direct connection to external resistive-type sensors like strain gages or Resistance Temperature Detectors (RTDs).

The auxiliary ADC, however, is unbuffered, resulting in higher analog input current on the auxiliary ADC. It should be noted that this unbuffered input path provides a dynamic load to the driving source. Therefore, resistor/capacitor combinations on the input pins can cause dc gain errors depending on the output impedance of the source that is driving the ADC inputs.

Analog Input Ranges

The absolute input voltage range on the primary ADC is restricted to between AGND + 100 mV to AV_{DD} – 100 mV. Care must be taken in setting up the common-mode voltage and input voltage range so that these limits are not exceeded; otherwise there will be a degradation in linearity performance.

The absolute input voltage range on the auxiliary ADC is restricted to between AGND – 30 mV to AV_{DD} + 30 mV. The slightly negative absolute input voltage limit does allow the possibility of monitoring small signal bipolar signals using the single-ended auxiliary ADC front end.

Programmable Gain Amplifier

The output from the buffer on the primary ADC is applied to the input of the on-chip programmable gain amplifier (PGA). The PGA can be programmed through eight different unipolar input ranges and bipolar ranges. The PGA gain range is programmed via the range bits in the ADC0CON SFR. With the external reference select bit set in the ADC0CON SFR and an external 2.5 V reference, the unipolar ranges are 0 mV to 20 mV, 0 mV to 40 mV, 0 mV to 80 mV, 0 mV to 160 mV, 0 mV to 320 mV, 0 mV to 640 mV, 0 V to 1.28 V, and 0 to 2.56 V; the bipolar ranges are ±20 mV, ±40 mV, ±80 mV, ±160 mV, ±320 mV, ±640 mV, ±1.28 V, and ±2.56 V. These are the nominal ranges that should appear at the input to the on-chip PGA. An ADC range matching specification of 2 μV (typ) across all ranges means that calibration need only be carried out at a single gain range and does not have to be repeated when the PGA gain range is changed.

Typical matching across ranges is shown in Figure 9. Here, the primary ADC is configured in bipolar mode with an external 2.5 V reference, while just greater than 19 mV is forced on its inputs. The ADC continuously converts the dc input voltage at an update rate of 5.35 Hz, i.e., SF = FFH. In total, 800 conversion results are gathered. The first 100 results are gathered with the primary ADC operating in the ±20 mV range. The ADC range is then switched to ±40 mV, 100 more conversion results are gathered, and so on until the last group of 100 samples is gathered with the ADC configured in the ±2.56 V range. From Figure 9, the variation in the sample mean through each range, i.e., the range matching, is seen to be of the order of 2 μV.

The auxiliary ADC does not incorporate a PGA and is configured for a fixed single input range of 0 to V_{REF}.

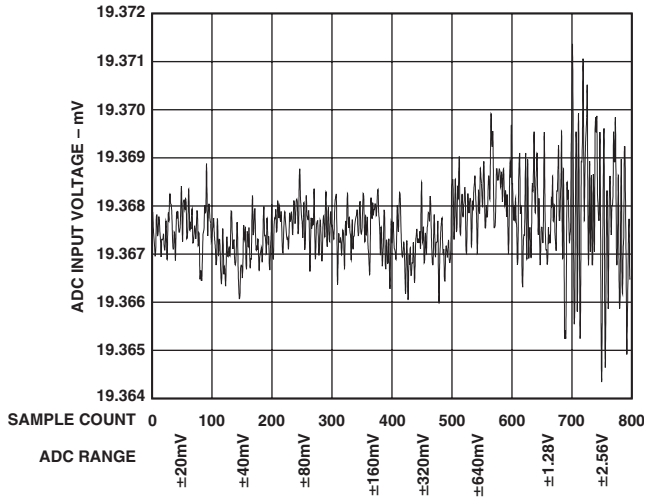


Figure 9. Primary ADC Range Matching

Bipolar/Unipolar Inputs

The analog inputs on the ADuC834 can accept either unipolar or bipolar input voltage ranges. Bipolar input ranges do not imply that the part can handle negative voltages with respect to system AGND.

Unipolar and bipolar signals on the AIN(+) input on the primary ADC are referenced to the voltage on the respective AIN(–) input. For example, if AIN(–) is 2.5 V and the primary ADC is configured for an analog input range of 0 mV to 20 mV, the input voltage range on the AIN(+) input is 2.5 V to 2.52 V. If AIN(–) is 2.5 V and the ADuC834 is configured for an analog input range of 1.28 V, the analog input range on the AIN(+) input is 1.22 V to 3.78 V (i.e., 2.5 V ± 1.28 V).

As mentioned earlier, the auxiliary ADC input is a single-ended input with respect to the system AGND. In this context, a bipolar signal on the auxiliary ADC can only span 30 mV negative with respect to AGND before violating the voltage input limits for this ADC.

Bipolar or unipolar options are chosen by programming the primary and auxiliary Unipolar enable bits in the ADC0CON and ADC1CON SFRs respectively. This programs the relevant ADC for either unipolar or bipolar operation. Programming for either unipolar or bipolar operation does not change any of the input signal conditioning; it simply changes the data output coding and the points on the transfer function where calibrations occur. When an ADC is configured for unipolar operation, the output coding is natural (straight) binary with a zero differential input voltage resulting in a code of 000 . . . 000, a midscale voltage resulting in a code of 100 . . . 000, and a full-scale input voltage resulting in a code of 111 . . . 111. When an ADC is configured for bipolar operation, the coding is offset binary with a negative full-scale voltage resulting in a code of 000 . . . 000, a zero differential voltage resulting in a code of 100 . . . 000, and a positive full-scale voltage resulting in a code of 111 . . . 111.

Reference Input
The ADuC834's reference inputs, REFIN(+) and REFIN(–), provide a differential reference input capability. The common-mode range for these differential inputs is from AGND to AV_{DD}. The nominal reference voltage, V_{REF} (REFIN(+) – REFIN(–)), for specified operation is 2.5 V with the primary and auxiliary reference enable bits set in the respective ADC0CON and/or ADC1CON SFRs.

The part is also functional (although not specified for performance) when the XREF0 or XREF1 bits are 0, which enables the on-chip internal band gap reference. In this mode, the ADCs will see the internal reference of 1.25 V, therefore halving all input ranges. As a result of using the internal reference voltage, a noticeable degradation in peak-to-peak resolution will result. Therefore, for best performance, operation with an external reference is strongly recommended.

In applications where the excitation (voltage or current) for the transducer on the analog input also drives the reference voltage for the part, the effect of the low frequency noise in the excitation source will be removed as the application is ratiometric. If the ADuC834 is not used in a ratiometric application, a low noise reference should be used. Recommended reference voltage sources for the ADuC834 include the AD780, REF43, and REF192.

It should also be noted that the reference inputs provide a high impedance, dynamic load. Because the input impedance of each reference input is dynamic, resistor/capacitor combinations on these inputs can cause dc gain errors depending on the output impedance of the source that is driving the reference inputs. Reference voltage sources, like those recommended above (e.g., AD780), will typically have low output impedances and therefore decoupling capacitors on the REFIN(+) input would be recommended. Deriving the reference input voltage across an external resistor, as shown in Figure 66, will mean that the reference input sees a significant external source impedance. External decoupling on the REFIN(+) and REFIN(–) pins would not be recommended in this type of circuit configuration.

Burnout Currents
The primary ADC on the ADuC834 contains two 100 nA constant current generators, one sourcing current from AV_{DD} to AIN(+), and one sinking from AIN(–) to AGND. The currents are switched to the selected analog input pair. Both currents are either on or off, depending on the Burnout Current Enable (BO) bit in the ICON SFR (see Table IX). These currents can be used to verify that an external transducer is still operational before attempting to take measurements on that channel. Once the burnout currents are turned on, they will flow in the external transducer circuit, and a measurement of the input voltage on the analog input channel can be taken. If the resultant voltage measured is full-scale, it indicates that the transducer has gone open-circuit. If the voltage measured is 0 V, it indicates that the transducer has short circuited. For normal operation, these burnout currents are turned off by writing a 0 to the BO bit in the ICON SFR. The current sources work over the normal absolute input voltage range specifications.

Excitation Currents
The ADuC834 also contains two identical, 200 μA constant current sources. Both source current from AV_{DD} to Pin 3 (IEXC1) or Pin 4 (IEXC2). These current sources are controlled via bits in the ICON SFR shown in Table IX. They can be configured to source 200 μA individually to both pins or a combination of both currents, i.e., 400 μA, to either of the selected pins. These current sources can be used to excite external resistive bridge or RTD sensors.

Reference Detect
The ADuC834 includes on-chip circuitry to detect if the part has a valid reference for conversions or calibrations. If the voltage between the external REFIN(+) and REFIN(–) pins goes below 0.3 V or either the REFIN(+) or REFIN(–) inputs is open circuit, the ADuC834 detects that it no longer has a valid reference. In this case, the NOXREF bit of the ADCSTAT SFR is set to a 1. If the ADuC834 is performing normal conversions and the NOXREF bit becomes active, the conversion results revert to all 1s. It is not necessary to continuously monitor the status of the NOXREF bit when performing conversions. It is only necessary to verify its status if the conversion result read from the ADC Data Register is all 1s.

If the ADuC834 is performing either an offset or gain calibration and the NOXREF bit becomes active, the updating of the respective calibration registers is inhibited to avoid loading incorrect coefficients to these registers, and the appropriate ERR0 or ERR1 bits in the ADCSTAT SFR are set. If the user is concerned about verifying that a valid reference is in place every time a calibration is performed, the status of the ERR0 or ERR1 bit should be checked at the end of the calibration cycle.

Σ-Δ Modulator
A Σ-Δ ADC generally consists of two main blocks, an analog modulator and a digital filter. In the case of the ADuC834 ADCs, the analog modulators consist of a difference amplifier, an integrator block, a comparator, and a feedback DAC as illustrated in Figure 10.

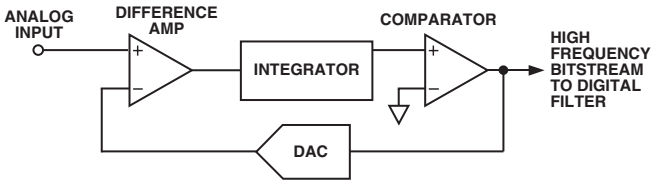


Figure 10. Σ-Δ Modulator Simplified Block Diagram

In operation, the analog signal sample is fed to the difference amplifier along with the output of the feedback DAC. The difference between these two signals is integrated and fed to the comparator. The output of the comparator provides the input to the feedback DAC so the system functions as a negative feedback loop that tries to minimize the difference signal. The digital data that represents the analog input voltage is contained in the duty cycle of the pulse train appearing at the output of the comparator. This duty cycle data can be recovered as a data-word using a subsequent digital filter stage. The sampling frequency of the modulator loop is many times higher than the bandwidth of the input signal. The integrator in the modulator shapes the quantization noise (which results from the analog-to-digital conversion) so that the noise is pushed toward one-half of the modulator frequency.

Digital Filter

The output of the Σ-Δ modulator feeds directly into the digital filter. The digital filter then band-limits the response to a frequency significantly lower than one-half of the modulator frequency. In this manner, the 1-bit output of the comparator is translated into a band-limited, low noise output from the ADuC834 ADCs.

The ADuC834 filter is a low-pass, $Sinc^3$ or (SIN x/x)3 filter whose primary function is to remove the quantization noise introduced at the modulator. The cutoff frequency and decimated output data rate of the filter are programmable via the SF (Sinc Filter) SFR as described in Table VIII.

Figure 11 shows the frequency response of the ADC channel at the default SF word of 69 dec or 45H, yielding an overall output update rate of just under 20 Hz.

It should be noted that this frequency response allows frequency components higher than the ADC Nyquist frequency to pass through the ADC, in some cases without significant attenuation. These components may, therefore, be aliased and appear in-band after the sampling process.

It should also be noted that rejection of mains-related frequency components, i.e., 50 Hz and 60 Hz, is seen to be at a level of >65 dB at 50 Hz and >100 dB at 60 Hz. This confirms the data sheet specifications for 50 Hz/60 Hz Normal Mode Rejection (NMR) at a 20 Hz update rate.

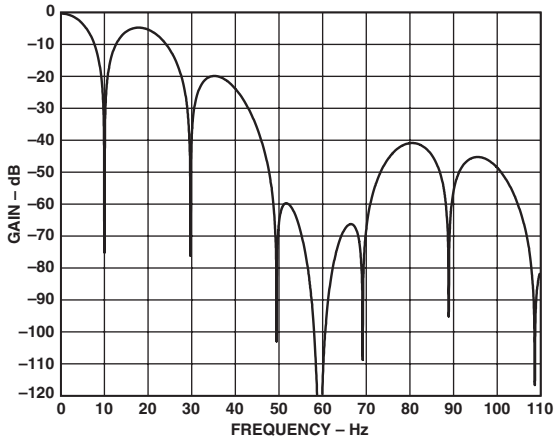


Figure 11. Filter Response, SF = 69 dec

The response of the filter, however, will change with SF word as can be seen in Figure 12, which shows >90 dB NMR at 50 Hz and >70 dB NMR at 60 Hz when SF = 255 dec.

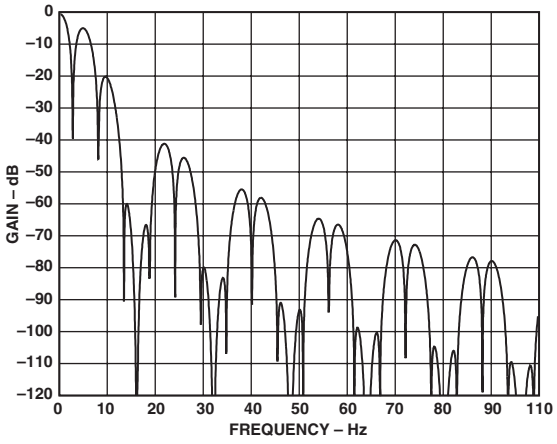


Figure 12. Filter Response, SF = 255 dec

Figures 13 and 14 show the NMR for 50 Hz and 60 Hz across the full range of SF word, i.e., SF = 13 dec to SF = 255 dec.

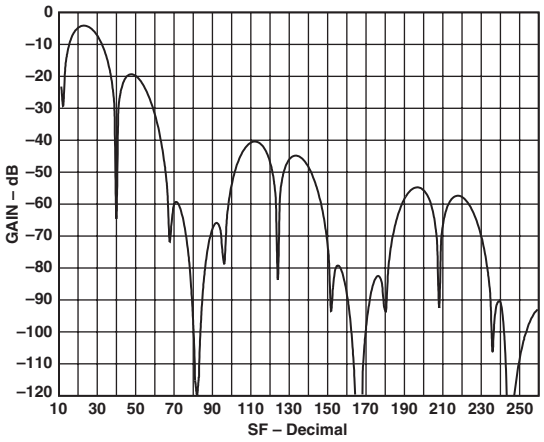


Figure 13. 50 Hz Normal Mode Rejection vs. SF

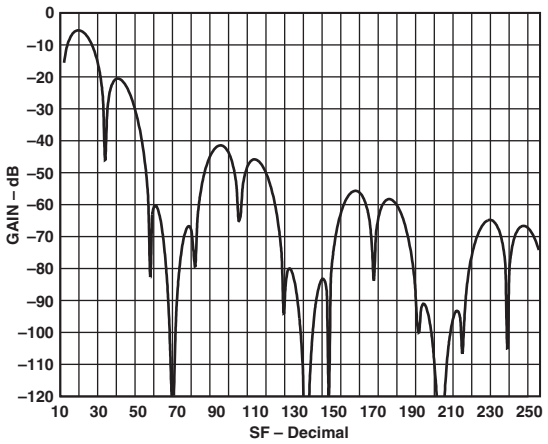


Figure 14. 60 Hz Normal Mode Rejection vs. SF

ADC Chopping

Both ADCs on the ADuC834 implement a chopping scheme whereby the ADC repeatedly reverses its inputs. The decimated digital output words from the Sinc3 filters therefore have a positive offset and negative offset term included.

As a result, a final summing stage is included in each ADC so that each output word from the filter is summed and averaged with the previous filter output to produce a new valid output result to be written to the ADC data SFRs. In this way, while the ADC throughput or update rate is as discussed earlier and illustrated in Table VIII, the full settling time through the ADC (or the time to a first conversion result), will actually be given by $2 \times t_{ADC}$.

The chopping scheme incorporated in the ADuC834 ADC results in excellent dc offset and offset drift specifications and is extremely beneficial in applications where drift, noise rejection, and optimum EMI rejection are important factors.

Calibration

The ADuC834 provides four calibration modes that can be programmed via the mode bits in the ADCMODE SFR detailed in Table V. In fact, every ADuC834 has already been factory calibrated. The resultant Offset and Gain calibration coefficients for both the primary and auxiliary ADCs are stored on-chip in manufacturing-specific Flash/EE memory locations. At power-on or after reset, these factory calibration coefficients are automatically downloaded to the calibration registers in the ADuC834 SFR space. Each ADC (primary and auxiliary) has dedicated calibration SFRs, these have been described earlier as part of the general ADC SFR description. However, the factory calibration values in the ADC calibration SFRs will be overwritten if any one of the four calibration options are initiated and that ADC is enabled via the ADC enable bits in ADCMODE.

Even though an internal offset calibration mode is described below, it should be recognized that both ADCs are chopped. This chopping scheme inherently minimizes offset and means that an internal offset calibration should never be required. Also, because factory 5 V/25°C gain calibration coefficients are automatically present at power-on, an internal full-scale calibration will only be required if the part is being operated at 3 V or at temperatures significantly different from 25°C.

The ADuC834 offers internal or system calibration facilities. For full calibration to occur on the selected ADC, the calibration logic must record the modulator output for two different input conditions. These are zero-scale and full-scale points. These points are derived by performing a conversion on the different input voltages provided to the input of the modulator during calibration. The result of the zero-scale calibration conversion is stored in the Offset Calibration Registers for the appropriate ADC. The result of the full-scale calibration conversion is stored in the Gain Calibration Registers for the appropriate ADC. With these readings, the calibration logic can calculate the offset and the gain slope for the input-to-output transfer function of the converter.

During an internal zero-scale or full-scale calibration, the respective zero-scale input and full-scale input are automatically connected to the ADC input pins internally to the device. A system calibration, however, expects the system zero-scale and system full-scale voltages to be applied to the external ADC pins before the calibration mode is initiated. In this way, external ADC errors are taken into account and minimized as a result of system calibration. It should also be noted that to optimize calibration accuracy, all ADuC834 ADC calibrations are carried out automatically at the slowest update rate.

Internally in the ADuC834, the coefficients are normalized before being used to scale the words coming out of the digital filter. The offset calibration coefficient is subtracted from the result prior to the multiplication by the gain coefficient.

From an operational point of view, a calibration should be treated like another ADC conversion. A zero-scale calibration (if required) should always be carried out before a full-scale calibration. System software should monitor the relevant ADC RDY0/1 bit in the ADCSTAT SFR to determine end of calibration via a polling sequence or interrupt driven routine.

NONVOLATILE FLASH/EE MEMORY

Flash/EE Memory Overview

The ADuC834 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 15).

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.

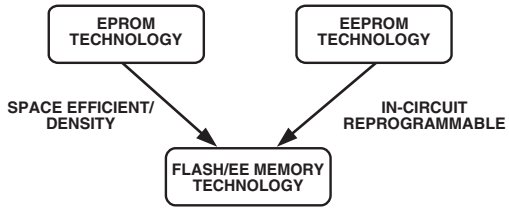


Figure 15. Flash/EE Memory Development

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 ADuC834, 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 ADuC834

The ADuC834 provides two arrays of Flash/EE memory for user applications. 62 Kbytes 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 in-circuit, using the serial download mode provided, using conventional third party memory programmers, or via any user defined protocol in User Download (ULOAD) Mode.

A 4 Kbyte 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 seven SFRs. This space can be programmed at a byte level, although it must first be erased in 4-byte pages.

ADuC834 Flash/EE Memory Reliability

The Flash/EE Program and Data Memory arrays on the ADuC834 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 00H to FFH 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 ADuC834 Flash/EE memory endurance qualification has been carried out in accordance with JEDEC Specification A117 over the industrial temperature range of –40°C, +25°C, +85°C, and +125°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 ADuC834 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 16.

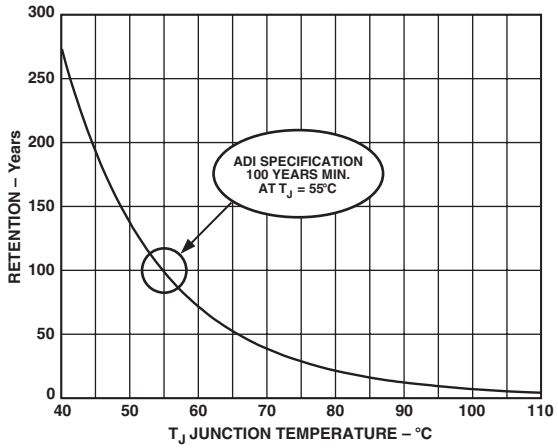


Figure 16. Flash/EE Memory Data Retention

Flash/EE Program Memory

The ADuC834 contains a 64 Kbyte array of Flash/EE program memory. The lower 62 Kbytes of this program memory is available to the user, and can be used for program storage or indeed as additional NV data memory.

The upper 2 Kbytes of this Flash/EE program memory array contain permanently embedded firmware, allowing in circuit serial download, serial debug and nonintrusive single pin emulation. These 2 Kbytes of embedded firmware also contain a power-on configuration routine that downloads factory calibrated coefficients to the various calibrated peripherals (ADC, temperature sensor, current sources, bandgap references and so on).

This 2 Kbyte embedded firmware is hidden from user code. Attempts to read this space will read 0s, i.e., the embedded firmware appears as NOP instructions to user code.

In normal operating mode (power up default) the 62 Kbytes of user Flash/EE program memory appear as a single block. This block is used to store the user code as shown in Figure 17.

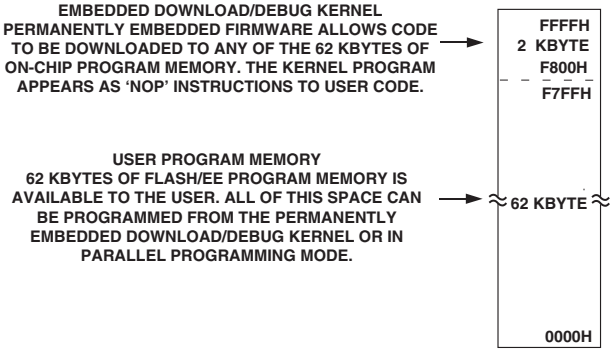


Figure 17. Flash/EE Program Memory Map in Normal Mode

In Normal Mode, the 62 Kbytes of Flash/EE program memory can be programmed programmed in two ways, namely:

(1) Serial Downloading (In-Circuit Programming)

The ADuC834 facilitates code download via the standard UART serial port. The ADuC834 will enter Serial Download mode after a reset or power cycle if the $\overline{\text{PSEN}}$ pin is pulled low through an external 1 kΩ resistor. Once in serial download mode, the hidden embedded download kernel will execute. This allows the user to download code to the full 62 Kbytes of Flash/EE program memory while the device is in circuit in its target application hardware.

A PC serial download executable is provided as part of the ADuC834 QuickStart development system. Appliction Note uC004 fully describes the serial download protocol that is used by the embedded download kernel. This Appliction Note is available at www.analog.com/microconverter.

(2) 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 18. In this mode, Ports 0, and 2 operate as the external address bus interface, P3 operates as the external data bus interface and P1.0 operates as the Write Enable strobe. Port 1.1, P1.2, P1.3, and P1.4 are used as a general configuration port that configures the device for various program and erase operations during parallel programming.

Table XIII. Flash/EE Memory Parallel Programming Modes

Port 1 Pins				
P1.4	P1.3	P1.2	P1.1	Programming Mode
0	0	0	0	Erase Flash/EE Program, Data, and Security Modes
1	0	0	1	Read Device Signature/ID
1	0	1	0	Program Code Byte
0	0	1	0	Program Data Byte
1	0	1	1	Read Code Byte
0	0	1	1	Read Data Byte
1	1	0	0	Program Security Modes
1	1	0	1	Read/Verify Security Modes
All other codes				Redundant

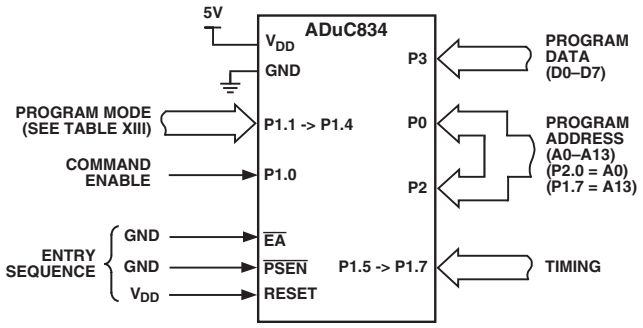


Figure 18. Flash/EE Memory Parallel Programming

User Download Mode (ULOAD)

In Figure 17 we can see that it was possible to use the 62 Kbytes of Flash/EE program memory available to the user as one single block of memory. In this mode all of the Flash/EE memory is read only to user code.

However, the Flash/EE program memory can also be written to during runtime simply by entering ULOAD mode. In ULOAD mode, the lower 56 Kbytes of program memory can be erased and reprogrammed by user software as shown in Figure 19. ULOAD mode can be used to upgrade your code in the field via any user defined download protocol. Configuring the SPI port on the ADuC834 as a slave, it is possible to completely reprogram the 56 Kbytes of Flash/EE program memory in only 5 seconds. See Application Note uC007.

Alternatively ULOAD Mode can be used to save data to the 56 Kbytes of Flash/EE memory. This can be extremely useful in datalogging applications where the ADuC834 can provide up to 60 Kbytes of NV data memory on-chip (4 Kbytes of dedicated Flash/EE data memory also exist).

The upper 6 Kbytes of the 62 Kbytes of Flash/EE program memory is only programmable via serial download or parallel programming. This means that this space appears as read only to user code. Therefore, it cannot be accidently erased or reprogrammed by erroneous code execution. This makes it very suitable to use the 6 Kbytes as a bootloader. A Bootload Enable option exists in the serial downloader to "Always RUN from E000h after Reset." If using a bootloader, this option is recommended to ensure that the bootloader always executes correct code after reset.

Programming the Flash/EE program memory via ULOAD mode is described in more detail in the description of ECON and also in Application Note uC007.

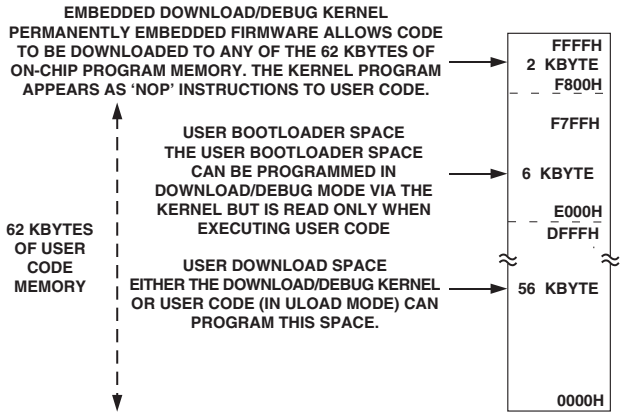


Figure 19. Flash/EE Program Memory Map in ULOAD Mode

Flash/EE Program Memory Security

The ADuC834 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 serial download protocol, as described in Application Note uC004, or via parallel programming. The ADuC834 offers the following security modes:

Lock Mode

This mode locks the code memory, disabling parallel programming of the program memory. However, reading the memory in Parallel Mode and reading the memory via a MOVC command from external memory are still allowed. This mode is deactivated by initiating an "erase code and data" command in Serial Download or Parallel Programming modes.

Secure Mode

This mode locks the code memory, disabling parallel programming of the program memory. Reading/Verifying the memory in Parallel Mode and reading the internal memory via a MOVC command from external memory is also disabled. This mode is deactivated by initiating an "erase code and data" 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 deasserted with $\overline{\text{PSEN}}$ low, 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 an "erase code and data" command in parallel programming mode.

Using the Flash/EE Data Memory

The 4 Kbytes of Flash/EE data memory is configured as 1024 pages, each of 4 bytes. As with the other ADuC834 peripherals, 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) is used to hold the 4 bytes of data at each page. The page is addressed via the two registers EADRH and EADRL. Finally, ECON is an 8-bit control register that may be written with one of nine Flash/EE memory access commands to trigger various read, write, erase, and verify functions.

A block diagram of the SFR interface to the Flash/EE data memory array is shown in Figure 20.

ECON—Flash/EE Memory Control SFR

Programming of either the Flash/EE data memory or the Flash/EE program memory is done through the Flash/EE Memory Control SFR (ECON). This SFR allows the user to read, write, erase or verify the 4 Kbytes of Flash/EE data memory or the 56 Kbytes of Flash/EE program memory.

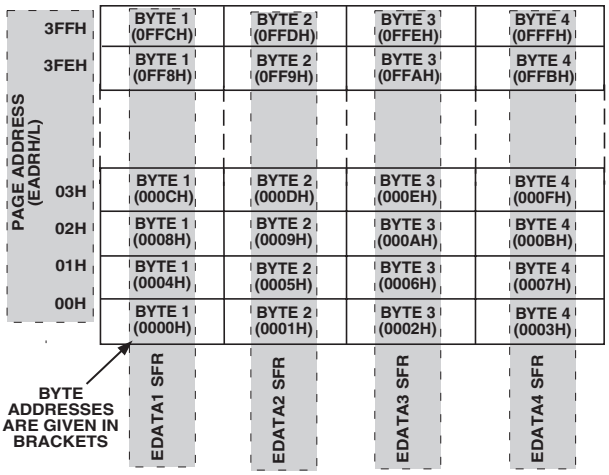


Figure 20. Flash/EE Data Memory Control and Configuration

Table XIV. ECON—Flash/EE Memory Commands

ECON Value	Command Description (Normal Mode) (Power-On Default)	Command Description (ULOAD Mode)
01H READ	Results in 4 bytes in the Flash/EE data memory, addressed by the page address EADRH/L, being read into EDATA 1 to 4.	Not Implemented. Use the MOVC instruction.
02H WRITE	Results in 4 bytes in EDATA1–4 being written to the Flash/EE data memory, at the page address given by EADRH. (0 ≤ EADRH < 0400H) Note: The 4 bytes in the page being addressed must be pre-erased.	Results in bytes 0–255 of internal XRAM being written to the 256 bytes of Flash/EE program memory at the page address given by EADRH/L (0 ≤ EADRH/L < E0H) Note: The 256 bytes in the page being addressed must be pre-erased.
03H	Reserved Command	Reserved Command
04H VERIFY	Verifies if the data in EDATA1–4 is contained in the page address given by EADRH/L. A subsequent read of the ECON SFR will result in a 0 being read if the verification is valid, or a nonzero value being read to indicate an invalid verification.	Not Implemented. Use the MOVC and MOVX instructions to verify the WRITE in software.
05H ERASE PAGE	Results in the erase of the 4 bytes page of Flash/EE data memory addressed by the page address EADRH/L	Results in the 64-bytes page of Flash/EE program memory, addressed by the byte address EADRH/L being erased. EADRL can equal any of 64 locations within the page. A new page starts whenever EADRL is equal to 00H, 40H, 80H, or C0H
06H ERASE ALL	Results in the erase of entire 4 Kbytes of Flash/EE data memory.	Results in the erase of the entire 56 Kbytes of ULOAD Flash/EE program memory
81H READBYTE	Results in the byte in the Flash/EE data memory, addressed by the byte address EADRH/L, being read into EDATA1. (0 ≤ EADRH/L ≤ 0FFFH).	Not Implemented. Use the MOVC command.
82H WRITEBYTE	Results in the byte in EDATA1 being written into Flash/EE data memory, at the byte address EADRH/L.	Results in the byte in EDATA1 being written into Flash/EE program memory at the byte address EADRH/L (0 ≤ EADRH/L ≤ DFFFH)
0FH EXULOAD	Leaves the ECON instructions to operate on the Flash/EE data memory.	Enters normal mode directing subsequent ECON instructions to operate on the Flash/EE data memory
F0H ULOAD	Enters ULOAD mode, directing subsequent ECON instructions to operate on the Flash/EE program memory.	Leaves the ECON Instructions to operate on the Flash/EE program memory.

Programming the Flash/EE Data Memory
A user wishes to program 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.

A typical program of the Flash/EE data array will involve:

1. setting EADRH/L with the page address
2. writing the data to be programmed to the EDATA1–4
3. writing the ECON SFR with the appropriate command

Step 1: Set Up the Page Address
The two address registers EADRH and EADRL hold the high byte address and the low byte address of the page to be addressed. The assembly language to set up the address may appear as:

```
MOV   EADRH,#0   ;  Set Page Address Pointer
MOV   EADRL,#03H
```

Step 2: Set Up the EDATA Registers
The four values to be written into the page into the 4 SFRs EDATA1–4. Unfortunately we do not know three of them. Thus it is necessary to read the current page and overwrite the second byte.

```
MOV   ECON,#1       ;  Read Page into EDATA1-4
MOV   EDATA2,#0F3H  ;  Overwrite byte 2
```

Step 3: Program Page
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 erase must happen at a page level. Therefore, a minimum of 4 bytes (1 page) will be erased when an erase command is initiated. Once the page is erased, we can program the 4 bytes in-page and then perform a verification of the data.

```
MOV ECON,#5       ;  ERASE Page
MOV ECON,#2       ;  WRITE Page
MOV ECON,#4       ;  VERIFY Page
MOV A,ECON        ;  Check if ECON=0 (OK!)
JNZ ERROR
```

Note: although the 4 Kbytes of Flash/EE data memory 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 ADuC834. An "ERASE-ALL" command consists of writing "06H" to the ECON SFR, which initiates an erase of the 4-Kbyte Flash/EE array. This command coded in 8051 assembly would appear as:

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

Flash/EE Memory Timing
Typical program and erase times for the ADuC834 are as follows:

Normal Mode (operating on Flash/EE data memory)

READPAGE (4 bytes)	– 5 machine cycles
WRITEPAGE (4 bytes)	– 380 μs
VERIFYPAGE (4 bytes)	– 5 machine cycles
ERASEPAGE (4 bytes)	– 2 ms
ERASEALL (4 Kbytes)	– 2 ms
READBYTE (1 byte)	– 3 machine cycles
WRITEBYTE (1 byte)	– 200 μs

ULOAD Mode (operating on Flash/EE program memory)

WRITEPAGE (256 bytes)	– 15 ms
ERASEPAGE (64 bytes)	– 2 ms
ERASEALL (56 Kbytes)	– 2 ms
WRITEBYTE (1 byte)	– 200 μs

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 ADuC834 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. This means that the core will not respond to interrupt requests until the Flash/EE operation is complete, although the core peripheral functions like Counter/Timers will continue to count and time as configured throughout this period.

DAC

The ADuC834 incorporates a 12-bit, voltage output DAC on-chip. It has a rail-to-rail voltage output buffer capable of driving 10 kΩ/100 pF. It has two selectable ranges, 0 V to V_{REF} (the internal bandgap 2.5 V reference) and 0 V to AV_{DD}. It can operate in 12-bit or 8-bit mode. The DAC has a control register, DACCON, and two data registers, DACH/L. The DAC output can be programmed to appear at Pin 3 or Pin 12. 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. The 12-bit DAC data should be written into DACH/L right-justified such that DACL contains the lower eight bits, and the lower nibble of DACH contains the upper four bits.

Table XV. DACCON SFR Bit Designations

Bit	Name	Description
7	—	Reserved for Future Use
6	—	Reserved for Future Use
5	—	Reserved for Future Use
4	DACPIN	DAC Output Pin Select. Set by the user to direct the DAC output to Pin 12 (P1.7/AIN4/DAC). Cleared by user to direct the DAC output to Pin 3 (P1.2/DAC/IEXC1).
3	DAC8	DAC 8-bit Mode Bit. Set by user to enable 8-bit DAC operation. In this mode, the 8-bits in DACL SFR are routed to the 8 MSBs of the DAC, and the 4 LSBs of the DAC are set to zero. Cleared by user to operate the DAC in its normal 12-bit mode of operation.
2	DACRN	DAC Output Range Bit. Set by user to configure DAC range of 0–AV_{DD}. Cleared by user to configure DAC range of 0 V–2.5 V (V_{REF}).
1	$\overline{\text{DACCLR}}$	DAC Clear Bit. Set to 1 by user to enable normal DAC operation. Cleared to 0 by user to reset DAC data registers DACL/H to zero.
0	DACEN	DAC Enable Bit. Set to 1 by user to enable normal DAC operation. Cleared to 0 by user to power down the DAC.

DACH/L	**DAC Data Registers**	
Function	DAC Data Registers, written by user to update the DAC output.	
SFR Address	DACL (DAC Data Low Byte)	→ FBH
	DACH (DAC Data High Byte)	→ FCH
Power-On Default Value	00H	→ Both Registers
Bit Addressable	No	→ Both Registers

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 21.

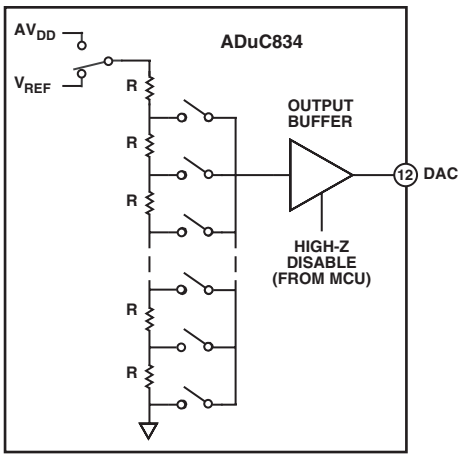


Figure 21. Resistor String DAC Functional Equivalent

Features of this architecture include inherent guaranteed monotonicity and excellent differential linearity. As illustrated in Figure 21, the reference source for the DAC is user selectable in software. It can be either AV_{DD} or V_{REF}. In 0-to-AV_{DD} mode, the DAC output transfer function spans from 0 V to the voltage at the AV_{DD} pin. In 0-to-V_{REF} mode, the DAC output transfer function spans from 0 V to the internal V_{REF} (2.5 V). 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 100 mV of both AV_{DD} and ground. Moreover, the DAC's linearity specification (when driving a 10 kΩ resistive load to ground) is guaranteed through the full transfer function except codes 0 to 48 in 0-to-V_{REF} mode and 0 to 100 and 3950 to 4095 in 0-to-V_{DD} mode.

Linearity degradation near ground and V_{DD} is caused by saturation of the output amplifier, and a general representation of its effects (neglecting offset and gain error) is illustrated in Figure 22. The dotted line in Figure 22 indicates the ideal transfer function, and the solid line represents what the transfer function might look like with endpoint nonlinearities due to saturation of the output amplifier.

Note that Figure 22 represents a transfer function in 0-to-V_{DD} mode only. In 0-to-V_{REF} mode (with $V_{REF} < V_{DD}$), the lower nonlinearity would be similar, but the upper portion of the transfer function would follow the "ideal" line right to the end, showing no signs of endpoint linearity errors.

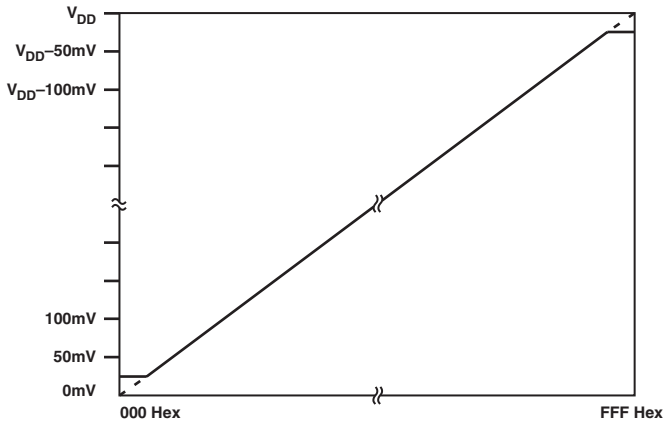


Figure 22. Endpoint Nonlinearities Due to Amplifier Saturation

The endpoint nonlinearities conceptually illustrated in Figure 22 get worse as a function of output loading. Most of the ADuC834 data sheet specifications assume a 10 kΩ resistive load to ground at the DAC output. As the output is forced to source or sink more current, the nonlinear regions at the top or bottom (respectively) of Figure 22 become larger. With larger current demands, this can significantly limit output voltage swing. Figures 23 and 24 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} = 3 V and V_{REF} = 2.5 V, the high-side voltage will not be affected by loads less than 5 mA. But somewhere around 7 mA, the upper curve in Figure 24 drops below 2.5 V (V_{REF}) indicating that at these higher currents, the output will not be capable of reaching V_{REF}.

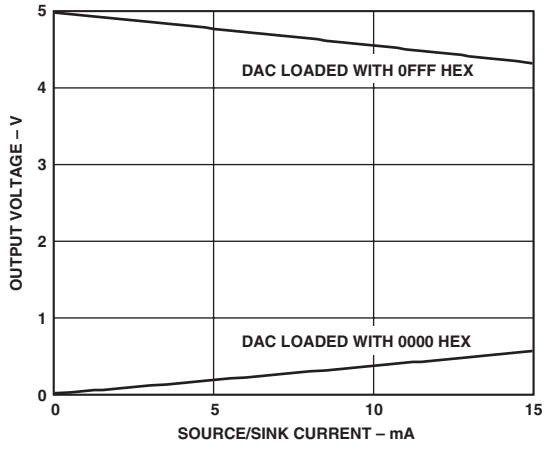


Figure 23. Source and Sink Current Capability with $V_{REF} = AV_{DD} = 5$ V

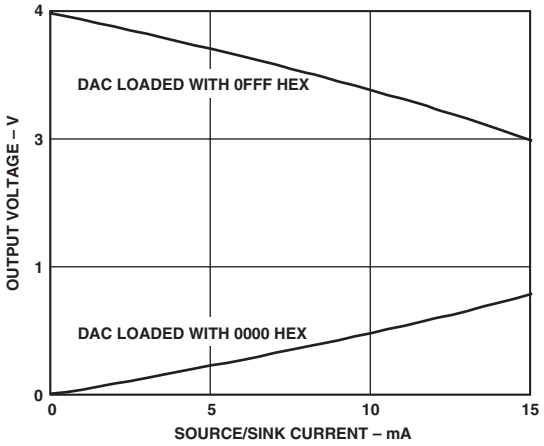


Figure 24. Source and Sink Current Capability with $V_{REF} = V_{DD} = 3$ V

For larger loads, the current drive capability may not be sufficient. In order to increase the source and sink current capability of the DAC, an external buffer should be added, as shown in Figure 25.

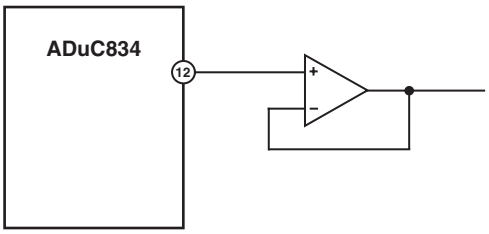


Figure 25. Buffering the DAC Output

The DAC output buffer also features a high impedance disable function. In the chip's default power-on state, the DAC is disabled and its output is in a high impedance state (or "three-state") where they remain inactive until enabled in software.

This means that if a zero output is desired during power-up or power-down transient conditions, a pull-down resistor must be added to each DAC output. Assuming this resistor is in place, the DAC output will remain at ground potential whenever the DAC is disabled.

PULSEWIDTH MODULATOR (PWM)

The PWM on the ADuC834 is a highly flexible PWM offering programmable resolution and input clock, and can be configured for any one of six different modes of operation. Two of these modes allow the PWM to be configured as a Σ-Δ DAC with up to 16 bits of resolution. A block diagram of the PWM is shown in Figure 26.

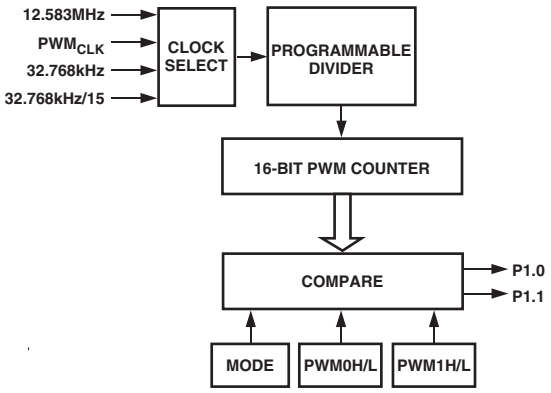


Figure 26. PWM Block Diagram

The PWM uses five SFRs: the control SFR, PWMCON, and four data SFRs PWM0H, PWM0L, PWM1H, and PWM1L.

PWMCON (as described below) controls the different modes of operation of the PWM as well as the PWM clock frequency. PWM0H/L and PWM1H/L are the data registers that determine the duty cycles of the PWM outputs at P1.0 and P1.1.

To use the PWM user software, first write to PWMCON to select the PWM mode of operation and the PWM input clock. Writing to PWMCON also resets the PWM counter. In any of the 16-bit modes of operation (Modes 1, 3, 4, 6), user software should write to the PWM0L or PWM1L SFRs first. This value is written to a hidden SFR. Writing to the PWM0H or PWM1H SFRs updates both the PWMxH and the PWMxL SFRs but does not change the outputs until the end of the PWM cycle in progress. The values written to these 16-bit registers are then used in the next PWM cycle.

PWMCON	**PWM Control SFR**
SFR Address	AEH
Power-On Default Value	00H
Bit Addressable	No

Table XVI. PWMCON SFR Bit Designations

Bit	Name	Description
7	—	Reserved for Future Use
6	MD2	PWM Mode Bits
5	MD1	The MD2/1/0 bits choose the PWM mode as follows:
4	MD0	MD2 MD1 MD0 Mode 0 0 0 Mode 0: PWM Disabled 0 0 1 Mode 1: Single Variable Resolution PWM 0 1 0 Mode 2: Twin 8-bit PWM 0 1 1 Mode 3: Twin 16-bit PWM 1 0 0 Mode 4: Dual NRZ 16-bit Σ-Δ DAC 1 0 1 Mode 5: Dual 8-bit PWM 1 1 0 Mode 6: Dual RZ 16-bit Σ-Δ DAC 1 1 1 Reserved for Future Use
3	CDIV1	PWM Clock Divider
2	CDIV0	Scale the clock source for the PWM counter as follows: CDIV1 CDIV0 Description 0 0 PWM Counter = Selected Clock /1 0 1 PWM Counter = Selected Clock /4 1 0 PWM Counter = Selected Clock /16 1 1 PWM Counter = Selected Clock /64
1	CSEL1	PWM Clock Divider
0	CSEL0	Select the clock source for the PWM as follows: CSEL1 CSEL0 Description 0 0 PWM Clock = f_{XTAL}/15 0 1 PWM Clock = f_{XTAL} 1 0 PWM Clock = External Input at P3.4/T0/PWMCLK 1 1 PWM Clock = f_{VCO} (12.58 MHz)

PWM MODES OF OPERATION

Mode 0: PWM Disabled

The PWM is disabled, allowing P1.0 and P1.1 be used as normal.

Mode 1: Single-Variable Resolution PWM

In Mode 1, both the pulse length and the cycle time (period) are programmable in user code, allowing the resolution of the PWM to be variable.

PWM1H/L sets the period of the output waveform. Reducing PWM1H/L reduces the resolution of the PWM output but increases the maximum output rate of the PWM (e.g., setting PWM1H/L to 65536 gives a 16-bit PWM with a maximum output rate of 192 Hz (12.583 MHz/65536). Setting PWM1H/L to 4096 gives a 12-bit PWM with a maximum output rate of 3072 Hz (12.583 MHz/4096)).

PWM0H/L sets the duty cycle of the PWM output waveform, as shown in Figure 27.

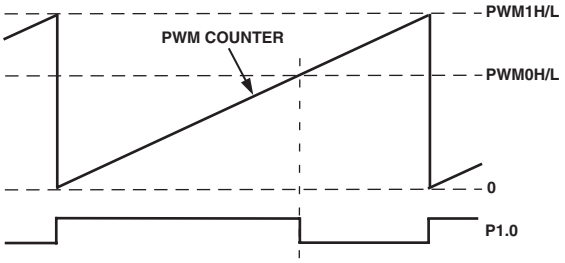


Figure 27. PWM in Mode 1

Mode 2: Twin 8-Bit PWM

In Mode 2, the duty cycle of the PWM outputs and the resolution of the PWM outputs are both programmable. The maximum resolution of the PWM output is eight bits.

PWM1L sets the period for both PWM outputs. Typically this will be set to 255 (FFh) to give an 8-bit PWM, although it is possible to reduce this as necessary. A value of 100 could be loaded here to give a percentage PWM (i.e., the PWM is accurate to 1%).

The outputs of the PWM at P1.0 and P1.1 are shown in the diagram below. As can be seen, the output of PWM0 (P1.0) goes low when the PWM counter equals PWM0L. The output of PWM1 (P1.1) goes high when the PWM counter equals PWM1H and goes low again when the PWM counter equals PWM0H. Setting PWM1H to 0 ensures that both PWM outputs start simultaneously.

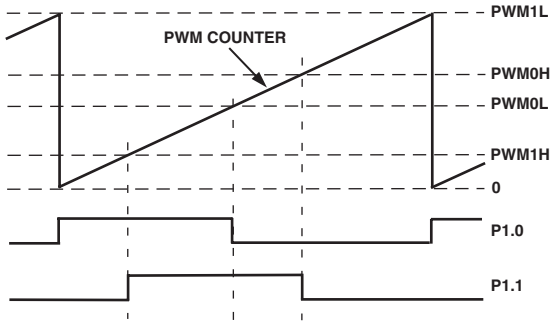


Figure 28. PWM Mode 2

Mode 3: Twin 16-Bit PWM

In Mode 3, the PWM counter is fixed to count from 0 to 65536 giving a fixed 16-bit PWM. Operating from the 12.58 MHz core clock results in a PWM output rate of 192 Hz. The duty cycle of the PWM outputs at P1.0 and P1.1 are independently programmable.

As shown below, while the PWM counter is less than PWM0H/L, the output of PWM0 (P1.0) is high. Once the PWM counter equals PWM0H/L, then PWM0 (P1.0) goes low and remains low until the PWM counter rolls over.

Similarly, while the PWM counter is less than PWM1H/L, the output of PWM1 (P1.1) is high. Once the PWM counter equals PWM1H/L, then PWM1 (P1.1) goes low and remains low until the PWM counter rolls over.

In this mode, both PWM outputs are synchronized (i.e., once the PWM counter rolls over to 0, both PWM0 (P1.0) and PWM1 (P1.1) will go high).

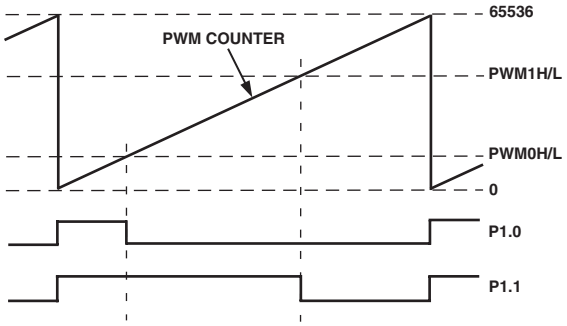


Figure 29. PWM Mode 3

Mode 4: Dual NRZ 16-Bit Σ-Δ DAC
Mode 4 provides a high speed PWM output similar to that of a Σ-Δ DAC. Typically, this mode will be used with the PWM clock equal to 12.58 MHz.

In this mode, P1.0 and P1.1 are updated every PWM clock (80 ns in the case of 12.58 MHz). Over any 65536 cycles (16-bit PWM) PWM0 (P1.0) is high for PWM0H/L cycles and low for (65536 – PWM0H/L) cycles. Similarly PWM1 (P1.1) is high for PWM1H/L cycles and low for (65536 – PWM1H/L) cycles.

If PWM1H is set to 4010H (slightly above one quarter of FS), then typically P1.1 will be low for three clocks and high for one clock (each clock is approximately 80 ns). Over every 65536 clocks, the PWM will compromise for the fact that the output should be slightly above one quarter of full scale by having a high cycle followed by only two low cycles.

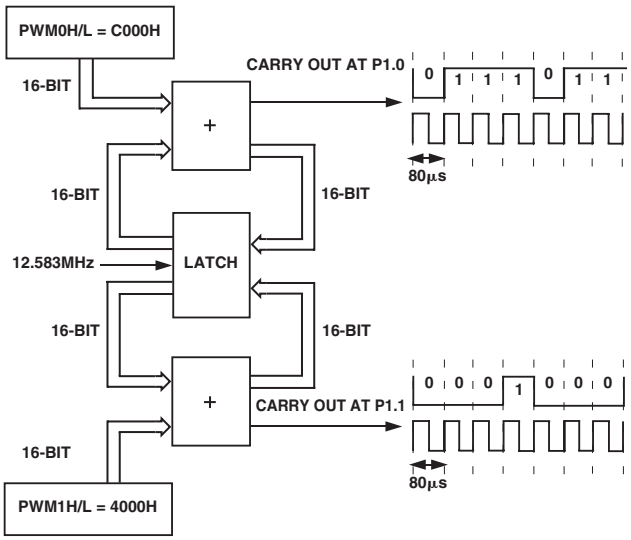


Figure 30. PWM Mode 4

For faster DAC outputs (at lower resolution), write 0s to the LSBs that are not required with a 1 in the LSB position. If, for example, only 12-bit performance is required, write "0001" to the 4 LSBs. This means that a 12-bit accurate Σ-Δ DAC output can occur at 3 kHz. Similarly, writing 00000001 to the 8 LSBs gives an 8-bit accurate Σ-Δ DAC output at 49 kHz.

Mode 5: Dual 8-Bit PWM
In Mode 5, the duty cycle of the PWM outputs and the resolution of the PWM outputs are individually programmable. The maximum resolution of the PWM output is eight bits.

The output resolution is set by the PWM1L and PWM1H SFRs for the P1.0 and P1.1 outputs respectively. PWM0L and PWM0H sets the duty cycles of the PWM outputs at P1.0 and P1.1, respectively. Both PWMs have the same clock source and clock divider.

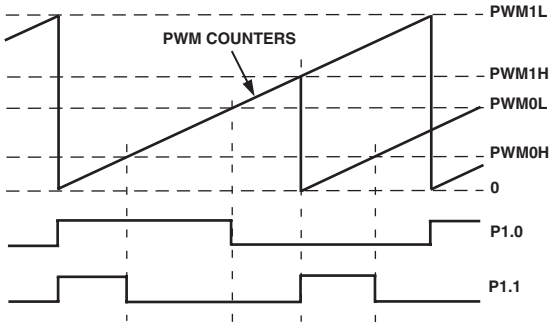


Figure 31. PWM Mode 5

Mode 6: Dual RZ 16-Bit Σ-Δ DAC
Mode 6 provides a high speed PWM output similar to that of a Σ-Δ DAC. Mode 6 operates very similarly to Mode 4. However, the key difference is that Mode 6 provides return to zero (RZ) Σ-Δ DAC output. Mode 4 provides non-return-to-zero Σ-Δ DAC outputs. The RZ mode ensures that any difference in the rise and fall times will not affect the Σ-Δ DAC INL. However, the RZ mode halves the dynamic range of the Σ-Δ DAC outputs from $0 \rightarrow AV_{DD}$ to $0 \rightarrow AV_{DD}/2$. For best results, this mode should be used with a PWM clock divider of 4.

If PWM1H is set to 4010H (slightly above one quarter of FS) then typically P1.1 will be low for three full clocks (3×80 ns), high for half a clock (40 ns) and then low again for half a clock (40 ns) before repeating itself. Over every 65536 clocks, the PWM will compromise for the fact that the output should be slightly above one quarter of full scale by leaving the output high for two half clocks in four every so often.

For faster DAC outputs (at lower resolution), write 0s to the LSBs that are not required with a 1 in the LSB position. If, for example, only 12-bit performance is required, write "0001" to the 4 LSBs. This means that a 12-bit accurate Σ-Δ DAC output can occur at 3 kHz. Similarly, writing 00000001 to the 8 LSBs gives an 8-bit accurate Σ-Δ DAC output at 49 kHz.

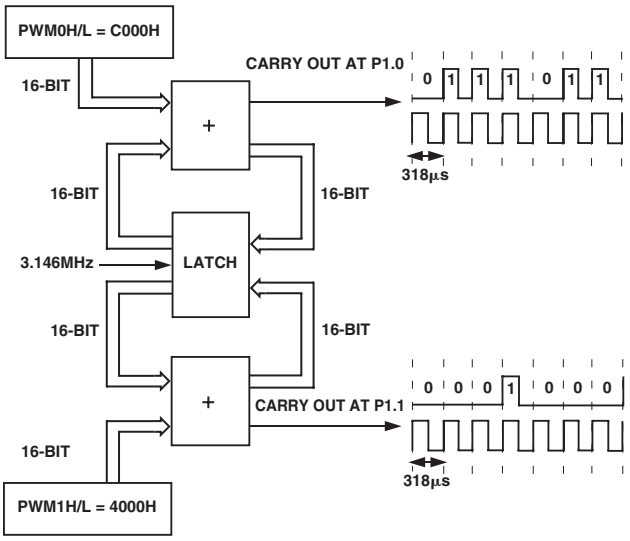


Figure 32. PWM Mode 6

ON-CHIP PLL

The ADuC834 is intended for use with a 32.768 kHz watch crystal. A PLL locks onto a multiple (384) of this to provide a stable 12.582912 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 1.572864 MHz. The ADC clocks are also derived from the PLL clock, with the modulator rate being the same as the crystal oscillator frequency. The above choice of frequencies ensures that the modulators and the core will be synchronous, regardless of the core clock rate. The PLL control register is PLLCON.

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

Table XVII. 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 TIC to continue counting 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. After power down, this bit can be polled to wait for the PLL to lock. 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 can be 12.58 MHz ± 20%. After the ADuC834 wakes up from power-down, user code may poll this bit, to wait for the PLL to lock. If LOCK = 0, then the PLL is not locked.
5	—	Reserved for Future Use; Should Be Written with '0'
4	$\overline{\text{LTEA}}$	Reading this bit returns the state of the external $\overline{\text{EA}}$ pin latched at reset or power-on.
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). After 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 12.582912 0 0 1 6.291456 0 1 0 3.145728 0 1 1 1.572864 (Default Core Clock Frequency) 1 0 0 0.786432 1 0 1 0.393216 1 1 0 0.196608 1 1 1 0.098304

TIME INTERVAL COUNTER (WAKE-UP/RTC TIMER)

A time interval counter (TIC) is provided on-chip for:

- periodically waking the part up from power-down
- implementing a Real-Time Clock
- counting longer intervals than the standard 8051 compatible timers are capable of

The TIC is capable of timeout 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.

The TIC counter can easily be used to generate a real-time clock. The hardware will count in seconds, minutes, and hours; however, user software will have to count in days, months, and years. The current time can be written to the timebase SFRs (HTHSEC, SEC, MIN, and HOUR) while TCEN is low. When the RTC timer is enabled (TCEN is set), the TCEN bit itself and the HTHSEC, SEC, MIN, and HOUR Registers are not reset to 00H after a hardware or watchdog timer reset. This is to prevent the need to recalibrate the real-time clock after a reset. However, these registers will be reset to 00H after a power cycle (independent of TCEN) or after any reset if TCEN is clear.

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 in this data sheet.) If the ADuC834 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 0053H. The TIC-related SFRs are described below with a block diagram of the TIC shown in Figure 33.

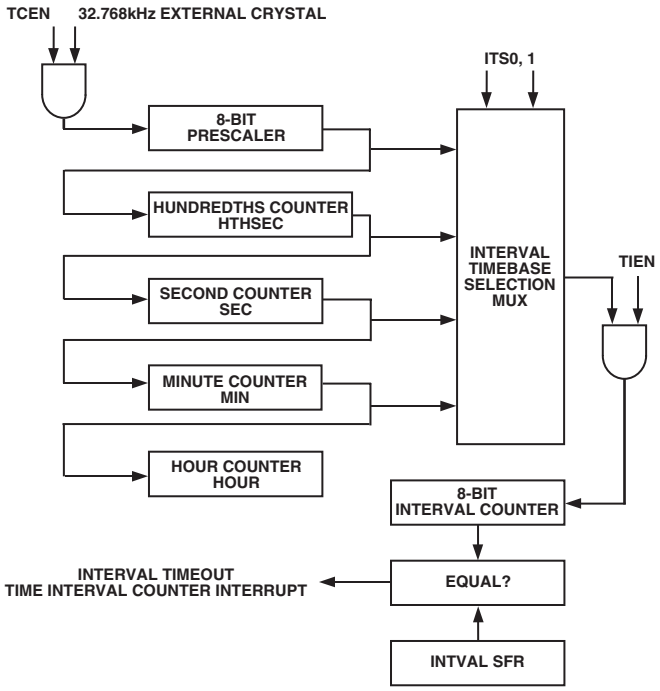


Figure 33. TIC, Simplified Block Diagram

Table XVIII. TIMECON SFR Bit Designations

Bit	Name	Description
7	—	Reserved for Future Use
6	—	Reserved for Future Use. For future product code compatibility, this bit should be written as a '1.'
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 timeout. 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 8-bit interval counter. To ensure that the 8-bit interval counter is cleared TIEN must be held low for at least 30.5 μs (32 kHz).
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 32 kHz clock to the TIC and clear the 8-bit prescaler and the HTHSEC, SEC, MIN and HOURS SFRs. To ensure that these registers are cleared, TCEN must be held low for at least 30.5 μs (32 kHz). The time registers (HTHSEC, SEC, MIN, and HOUR) can only be written while TCEN is low.

INTVAL **User Time Interval Select Register**

Function	User code writes the required time interval to this register. When the 8-bit interval counter is equal to the time interval value loaded in the INTVAL SFR, the TII bit (TIMECON.2) is set and generates generates an interrupt if enabled. (See IEIP2 SFR description under Interrupt System in this data sheet.)
SFR Address	A6H
Power-On Default Value	00H
Reset 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
Reset Default Value	00H if TCEN = 0, Previous Value before reset if TCEN = 1
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
Reset Default Value	00H if TCEN = 0, Previous Value before reset if TCEN = 1
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
Reset Default Value	00H if TCEN = 0, Previous Value before reset if TCEN = 1
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. The HOUR SFR counts from 0 to 23 before rolling over to 0.
SFR Address	A5H
Power-On Default Value	00H
Reset Default Value	00H if TCEN = 0, Previous Value before reset if TCEN = 1
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 ADuC834 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 timeout interval can be adjusted via the PRE3–0 bits in WDCON. Full control and status of the watchdog timer function can be controlled via the Watchdog Timer Control SFR (WDCON). The WDCON SFR can 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

Table XIX. 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	($0 \le PRE \le 7$; f_{PLL} = 32.768 kHz)
4	PRE0	PRE3 PRE2 PRE1 PRE0 Timeout Period (ms) Action 0 0 0 0 15.6 Reset or Interrupt 0 0 0 1 31.2 Reset or Interrupt 0 0 1 0 62.5 Reset or Interrupt 0 0 1 1 125 Reset or Interrupt 0 1 0 0 250 Reset or Interrupt 0 1 0 1 500 Reset or Interrupt 0 1 1 0 1000 Reset or Interrupt 0 1 1 1 2000 Reset or Interrupt 1 0 0 0 0.0 Immediate Reset PRE3–0 > 1001 Reserved
3	WDIR	Watchdog Interrupt 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 $\overline{EA}$ 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 XXXIX 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 a 1 is not written to this bit 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. For example:
		`CLR EA ; disable interrupts while writing`
		`; to WDT`
		`SETB WDWR ; allow write to WDCON`
		`MOV WDCON, #72h ; enable WDT for 2.0s timeout`
		`SETB EA ; enable interrupts again (if rqd)`

POWER SUPPLY MONITOR

As its name suggests, the Power Supply Monitor, once enabled, monitors both supplies (AV_{DD} or DV_{DD}) on the ADuC834. 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, AV_{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

Table XX. PSMCON SFR Bit Designations

Bit	Name	Description
7	CMPD	DV_{DD} Comparator Bit. This is a read-only bit and directly reflects the state of the DV_{DD} comparator. Read 1 indicates the DV_{DD} supply is above its selected trip point. Read 0 indicates the DV_{DD} supply is below its selected trip point.
6	CMPA	AV_{DD} Comparator Bit. This is a read-only bit and directly reflects the state of the AV_{DD} comparator. Read 1 indicates the AV_{DD} supply is above its selected trip point. Read 0 indicates the AV_{DD} supply is below its selected trip point.
5	PSMI	Power Supply Monitor Interrupt Bit. This bit will be set high by the MicroConverter if either CMPA or CMPD are low, indicating low analog or digital supply. The PSMI bit can be used to interrupt the processor. Once CMPD and/or CMPA return (and remain) 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 either comparator output is low, it is not possible for the user to clear PSMI.
4	TPD1	DV_{DD} Trip Point Selection Bits.
3	TPD0	These bits select the DV_{DD} trip point voltage as follows: TPD1 TPD0 Selected DV_{DD} Trip Point (V) 0 0 4.63 0 1 3.08 1 0 2.93 1 1 2.63
2	TPA1	AV_{DD} Trip Point Selection Bits.
1	TPA0	These bits select the AV_{DD} trip point voltage as follows: TPA1TPA0 Selected AV_{DD} Trip Point (V) 0 0 4.63 0 1 3.08 1 0 2.93 1 1 2.63
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.

SERIAL PERIPHERAL INTERFACE

The ADuC834 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 pins SCLOCK and MOSI are multiplexed with the I^2C pins SCLOCK and SDATA. The pins are controlled via the I2CCON SFR only if SPE is clear. SPI can be configured for master or slave operation and typically consists of four pins, namely:

SCLOCK (Serial Clock I/O Pin), Pin 26

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 XXI). In Slave mode the SPICON register will have to be configured with the phase and polarity (CPHA and CPOL) as the master as for both Master and Slave mode the data is transmitted on one edge of the SCLOCK signal and sampled on the other.

MISO (Master In, Slave Out Data I/O Pin), Pin 14

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 27

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.

$\overline{SS}$ (Slave Select Input Pin), Pin 13

The Slave Select ($\overline{SS}$) input pin is only used when the ADuC834 is configured in SPI Slave mode. This line is active low. Data is only received or transmitted in Slave mode when the $\overline{SS}$ pin is low, allowing the ADuC834 to be used in single master, multislave SPI configurations. If CPHA = 1, the $\overline{SS}$ input may be permanently pulled low. With CPHA = 0, the $\overline{SS}$ input must be driven low before the first bit in a byte wide transmission or reception and return high again after the last bit in that byte wide transmission or reception. In SPI Slave mode, the logic level on the external $\overline{SS}$ pin (Pin 13), can be read via the SPR0 bit in the SPICON SFR.

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

Table XXI. 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.
1	SPR1	SPI Bit-Rate Select Bits.
0	SPR0	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 $f_{CORE}/16$ In SPI Slave mode, i.e., SPIM = 0, the logic level on the external $\overline{SS}$ pin (Pin 13), can be read via the SPR0 bit.

*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 XXI, the ADuC834 SPI interface will transmit or receive data in a number of possible modes. Figure 34 shows all possible ADuC834 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.

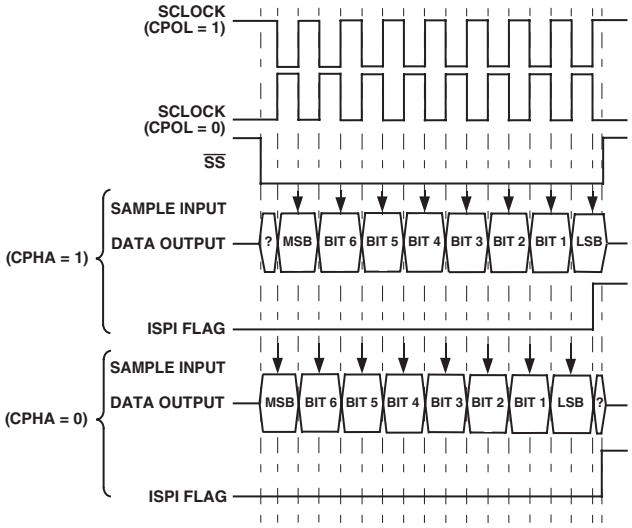


Figure 34. SPI Timing, All Modes

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 ADuC834 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 SERIAL INTERFACE

The ADuC834 supports a fully licensed* I²C serial interface. The I²C interface is implemented as a full hardware slave and software master. SDATA (Pin 27) is the data I/O pin and SCLOCK (Pin 26) is the serial clock. These two pins are shared with the MOSI and SCLOCK pins of the on-chip SPI interface. Therefore the user can only enable one or the other interface at any given time (see SPE in Table XXI). Application Note uC001 describes the operation of this interface as implemented and is available from the MicroConverter website at www.analog.com/microconverter.

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

I2CCON — **I²C Control Register**

SFR Address	E8H
Power-On Default Value	00H
Bit Addressable	Yes

Table XXII. 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 the 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 the 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 the 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 the 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 the user code. Application 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

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

The main features of the MicroConverter I²C interface are:

- Only two bus lines are required; a serial data line (SDATA) and a serial clock line (SCLOCK).
- An I²C master can communicate with multiple slave devices. Because each slave device has a unique 7-bit address then single master/slave relationships can exist at all times even in a multislave environment (Figure 35).
- On-chip filtering rejects <50 ns spikes on the SDATA and the SCLOCK lines to preserve data integrity.

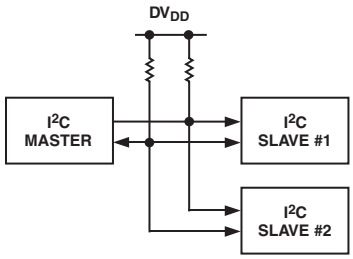


Figure 35. Typical I²C System

Software Master Mode

The ADuC834 can be used as a I²C master device by configuring the I²C peripheral in master mode and writing software to output the data bit by bit, which is referred to as a software master. Master mode is enabled by setting the I2CM bit in the I2CCON register.

To transmit data on the SDATA line, MDE must be set to enable the output driver on the SDATA pin. If MDE is set then the SDATA pin will be pulled high or low depending on whether the MDO bit is set or cleared. MCO controls the SCLOCK pin and is always configured as an output in Master mode. In Master mode, the SCLOCK pin will be pulled high or low depending on the whether MCO is set or cleared.

To receive data, MDE must be cleared to disable the output driver on SDATA. Software must provide the clocks by toggling the MCO bit and read SDATA pin via the MDI bit. If MDE is cleared, MDI can be used to read the SDATA pin. The value of the SDATA pin is latched into MDI on a rising edge of SCLOCK. MDI is set if the SDATA pin was high on the last rising edge of SCLOCK. MDI is clear if the SDATA pin was low on the last rising edge of SCLOCK.

Software must control MDO, MCO, and MDE appropriately to generate the START condition, slave address, acknowledge bits, data bytes, and STOP conditions appropriately. These functions are provided in Application Note uC001.

Hardware Slave Mode

After reset, the ADuC834 defaults to hardware slave mode. The I²C interface is enabled by clearing the SPE bit in SPICON. Slave mode is enabled by clearing the I2CM bit in I2CCON. The ADuC834 has a full hardware slave. In slave mode, the I²C address is stored in the I2CADD register. Data received or to be transmitted is stored in the I2CDAT register.

Once enabled in I²C slave mode, the slave controller waits for a START condition. If the ADuC834 detects a valid start condition, followed by a valid address, and by the R/$\overline{W}$ bit, the I2CI interrupt bit will get automatically set by hardware.

The I²C peripheral will only generate a core interrupt if the user has preconfigured the I²C interrupt enable bit in the IEIP2 SFR as well as the global interrupt bit EA in the IE SFR, i.e.,

```
; Enabling I2C Interrupts for the ADuC834
MOV  IEIP2,#01h     ; enable I2C interrupt
SETB EA
```

On the ADuC834 an auto clear of the I2CI bit is implemented so this bit is cleared automatically on a read or write access to the I2CDAT SFR.

```
  MOV     I2CDAT, A   ; I2CI auto-cleared MOV
A, I2CDAT   ; I2CI auto-cleared
```

If for any reason the user tries to clear the interrupt more than once, i.e., access the data SFR more than once per interrupt, then the I²C controller will halt. The interface will then have to be reset using the I2CRS bit.

The user can choose to poll the I2CI bit or enable the interrupt. In the case of the interrupt, the PC counter will vector to 003BH at the end of each complete byte. For the first byte when the user gets to the I2CI ISR, the 7-bit address and the R/$\overline{W}$ bit will appear in the I2CDAT SFR.

The I2CTX bit contains the R/$\overline{W}$ bit sent from the master. If I2CTX is set then the master would like to receive a byte. Hence the slave will transmit data by writing to the I2CDAT register. If I2CTX is cleared, the master would like to transmit a byte. Hence the slave will receive a serial byte. The software can interrogate the state of I2CTX to determine whether it should write to or read from I2CDAT.

Once the ADuC834 has received a valid address, hardware will hold SCLOCK low until the I2CI bit is cleared by the software. This allows the master to wait for the slave to be ready before transmitting the clocks for the next byte.

The I2CI interrupt bit will be set every time a complete data byte is received or transmitted provided it is followed by a valid ACK. If the byte is followed by a NACK, an interrupt is NOT generated. The ADuC834 will continue to issue interrupts for each complete data byte transferred until a STOP condition is received or the interface is reset.

When a STOP condition is received, the interface will reset to a state where it is waiting to be addressed (idle). Similarly, if the interface receives a NACK at the end of a sequence, it also returns to the default idle state. The I2CRS bit can be used to reset the I²C interface. This bit can be used to force the interface back to the default idle state.

It should be noted that there is no way (in hardware) to distinguish between an interrupt generated by a received START + valid address and an interrupt generated by a received data byte. User software must be used to distinguish between these interrupts.

DUAL DATA POINTER

The ADuC834 incorporates two data pointers. The second data pointer is a shadow data pointer and is selected via the data pointer control SFR (DPCON). DPCON also includes features such as automatic hardware post-increment and post-decrement, as well as automatic data pointer toggle. DPCON is described in Table XXIII.

DPCON	**Data Pointer Control SFR**
SFR Address	A7H
Power-On Default Value	00H
Bit Addressable	No

Table XXIII. DPCON SFR Bit Designations

Bit	Name	Description
7	—	Reserved for Future Use
6	DPT	Data Pointer Automatic Toggle Enable. Cleared by user to disable auto swapping of the DPTR. Set in user software to enable automatic toggling of the DPTR after each MOVX or MOVC instruction.
5	DP1m1	Shadow Data Pointer Mode
4	DP1m0	These two bits enable extra modes of the shadow data pointer operation allowing for more compact and more efficient code size and execution. m1 m0 Behavior of the Shadow Data Pointer 0 0 8052 Behavior 0 1 DPTR is post-incremented after a MOVX or a MOVC instruction 1 0 DPTR is post-decremented after a MOVX or MOVC instruction 1 1 DPTR LSB is toggled after a MOVX or MOVC instruction (This instruction can be useful for moving 8-bit blocks to/from 16-bit devices)
3	DP0m1	Main Data Pointer Mode
2	DP0m0	These two bits enable extra modes of the main data pointer operation allowing for more compact and more efficient code size and execution. m1 m0 Behavior of the Main Data Pointer 0 0 8052 Behavior 0 1 DPTR is post-incremented after a MOVX or a MOVC instruction 1 0 DPTR is post-decremented after a MOVX or MOVC instruction 1 1 DPTR LSB is toggled after a MOVX or MOVC instruction (This instruction can be useful for moving 8-bit blocks to/from 16-bit devices)
1	—	This bit is not implemented to allow the INC DPCON instruction toggle the data pointer without incrementing the rest of the SFR.
0	DPSEL	Data Pointer Select. Cleared by user to select the main data pointer. This means that the contents of the main 24-bit DPTR appears in the 3 SFRs DPL, DPH, and DPP. Set by the user to select the shadow data pointer. This means that the contents of the shadow 24-bit DPTR appears in the 3 SFRs DPL, DPH, and DPP.

NOTES

1. This is the only place where the main and shadow data pointers are distinguished. Everywhere else in this data sheet, wherever the DPTR is mentioned, operation on the active DPTR is implied.
2. Only MOVC/MOVX @DPTR instructions are relevant above. MOVC/MOVX PC/@Ri instructions will not cause the DPTR to automatically post increment/decrement, and so on. To illustrate the operation of DPCON, the following code will copy 256 bytes of code memory at Address D000h into XRAM starting from Address 0000h. the code uses 16 bytes and 2054 cycles. To perform this on a standard 8051 requires approximately 33 bytes and 7172 cycles (depending on how it is implemented).

```
MOV  DPTR,#0        ; Main DPTR = 0
MOV  DPCON,#55h     ; Select shadow DPTR
                    ; DPTR1 increment mode,
                    ; DPTR0 increment mode
                    ; DPTR auto toggling ON
MOV  DPTR,#0D000h   ; Shadow DPTR = D000h
```

```
MOVELOOP:
    CLR  A
    MOVC A,@A+DPTR      ; Get data
                        ; Post Inc DPTR
                        ; Swap to Main DPTR (Data)
    MOVX @DPTR,A        ; Put ACC in XRAM
                        ; Increment main DPTR
                        ; Swap to Shad DPTR (Code)
    MOV  A, DPL
    JNZ  MOVELOOP
```

8052 COMPATIBLE ON-CHIP PERIPHERALS

This section gives a brief overview of the various secondary peripheral circuits that are also available to the user on-chip. These remaining functions are mostly 8052 compatible (with a few additional features) and are controlled via standard 8052 SFR bit definitions.

Parallel I/O

The ADuC834 uses four input/output ports to exchange data with external devices. In addition to performing general-purpose I/O, some ports are capable of external memory operations while others are multiplexed with alternate functions 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 0

Port 0 is an 8-bit open-drain bidirectional I/O port that is directly controlled via the Port 0 SFR. Port 0 is also the multiplexed low order address and databus during accesses to external program or data memory.

Figure 36 shows a typical bit latch and I/O buffer for a Port 0 port pin. The bit latch (one bit in the port's SFR) is represented as a Type D flip-flop, which will clock in a value from the internal bus in response to a "write to latch" signal from the CPU. The Q output of the flip-flop is placed on the internal bus in response to a "read latch" signal from the CPU. The level of the port pin itself is placed on the internal bus in response to a "read pin" signal from the CPU. Some instructions that read a port activate the "read latch" signal, and others activate the "read pin" signal. See the following Read-Modify-Write Instructions section for more details.

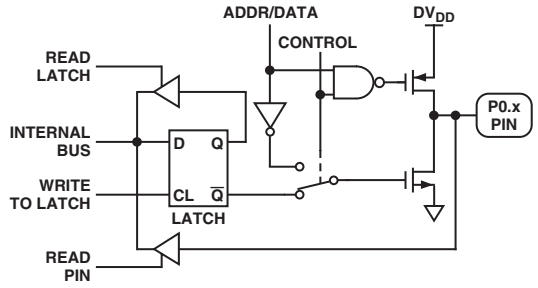


Figure 36. Port 0 Bit Latch and I/O Buffer

As shown in Figure 36, the output drivers of Port 0 pins are switchable to an internal ADDR and ADDR/DATA bus by an internal CONTROL signal for use in external memory accesses. During external memory accesses, the P0 SFR gets 1s written to it (i.e., all of its bit latches become 1). When accessing external memory, the CONTROL signal in Figure 36 goes high, enabling push-pull operation of the output pin from the internal address or databus (ADDR/DATA line). Therefore, no external pull-ups are required on Port 0 in order for it to access external memory.

In general-purpose I/O port mode, Port 0 pins that have 1s written to them via the Port 0 SFR will be configured as open-drain and will therefore float. In this state, Port 0 pins can be used as high impedance inputs. This is represented in Figure 36 by the NAND gate whose output remains high as long as the CONTROL signal is low, thereby disabling the top FET. External pull-up resistors are therefore required when Port 0 pins are used as general-purpose outputs. Port 0 pins with 0s written to them will drive a logic low output voltage (V_{OL}) and will be capable of sinking 1.6 mA.

Port 1

Port 1 is also an 8-bit port directly controlled via the P1 SFR. The Port 1 pins are divided into two distinct pin groupings P1.0 to P1.1 and P1.2 to P1.7.

P1.0 and P1.1

P1.0 and P1.1 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 (V_{OL}) 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 XXIV. The timer 2 alternate functions of P1.0 and P1.1 can only be activated if the corresponding bit latch in the P1 SFR contains a 1. Otherwise, the port pin is stuck at 0. In the case of the PWM outputs at P1.0 and P1.1, the PWM outputs will overwrite anything written to P1.0 or P1.1.

Table XXIV. P1.0 and P1.1 Alternate Pin Functions

Pin	Alternate Function
P1.0	T2 (Timer/Counter 2 External Input) PWM0 (PWM0 output at this pin)
P1.1	T2EX (Timer/Counter 2 Capture/Reload Trigger) PWM1 (PWM1 output at this pin)

Figure 37 shows a typical bit latch and I/O buffer for a P1.0 or P1.1 port pin. No external memory access is required from either of these pins although internal pull-ups are present.

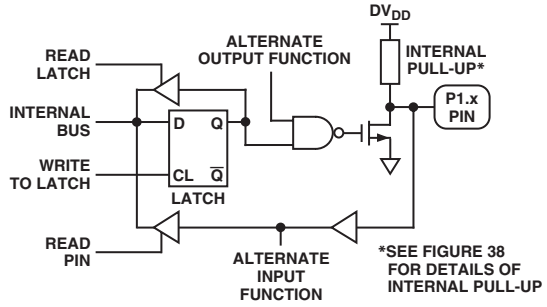


Figure 37. P1.0 and P1.1 Bit Latch and I/O Buffer

The internal pull-up consists of active circuitry as shown in Figure 38. Whenever a P1.0 or P1.1 bit latch transitions from low to high, Q1 in Figure 38 turns on for 2 oscillator periods to quickly pull the pin to a logic high state. Once there, the weaker Q3 turns on, thereby latching the pin to a logic high. If the pin is momentarily pulled low externally, Q3 will turn off, but the very weak Q2 will continue to source some current into the pin, attempting to restore it to a logic high.

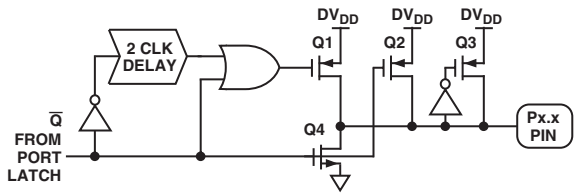


Figure 38. Internal Pull-Up Configuration

P1.2 to P1.7

The remaining Port 1 pins (P1.2–P1.7) can only be configured as analog input (ADC) 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. Figure 39 illustrates this function. Note that there are no output drivers for Port 1 pins, and they therefore cannot be used as outputs.

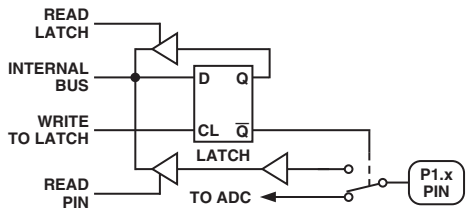


Figure 39. P1.2 to P1.7 Bit Latch and I/O Buffer

Port 2

Port 2 is a bidirectional port with internal pull-up resistors directly controlled via the P2 SFR. Port 2 also emits the high order address bytes during fetches from external program memory and middle and high order address bytes during accesses to the 24-bit external data memory space.

As shown in Figure 40, the output drivers of Ports 2 are switchable to an internal ADDR bus by an internal CONTROL signal for use in external memory accesses (as for Port 0). In external memory addressing mode (CONTROL = 1), the port pins feature push/pull operation controlled by the internal address bus (ADDR line). However unlike the P0 SFR during external memory accesses, the P2 SFR remains unchanged.

In general-purpose I/O port mode, Port 2 pins that have 1s written to them are pulled high by the internal pull-ups (Figure 38) and, in that state, they can be used as inputs. As inputs, Port 2 pins being pulled externally low will source current because of the internal pull-up resistors. Port 2 pins with 0s written to them will drive a logic low output voltage (V_{OL}) and will be capable of sinking 1.6 mA.

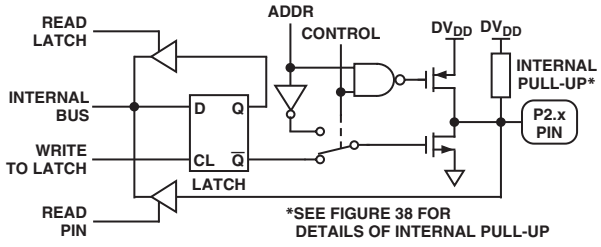


Figure 40. Port 2 Bit Latch and I/O Buffer

Port 3

Port 3 is a bidirectional port with internal pull-ups directly controlled via the P3 SFR.

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 with 0s written to them will drive a logic low output voltage (V_{OL}) and will be capable of sinking 1.6 mA.

Port 3 pins also have various secondary functions described in Table XXV. The alternate functions of Port 3 pins can only be activated if the corresponding bit latch in the P3 SFR contains a 1. Otherwise, the port pin is stuck at 0.

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

Port 3 pins have the same bit latch and I/O buffer configurations as the P1.0 and P1.1 as shown in Figure 41. The internal pull-up configuration is also defined by that in Figure 38.

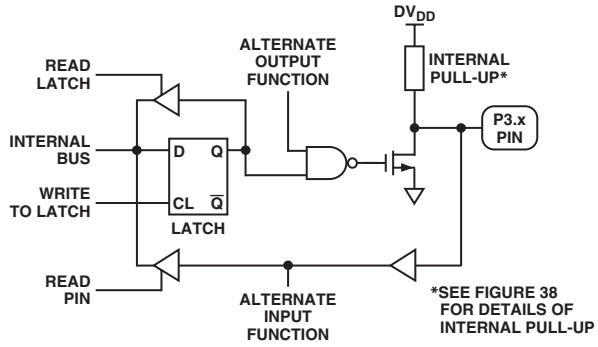


Figure 41. Port 3 Bit Latch and I/O Buffer

Additional Digital I/O

In addition to the port pins, the dedicated SPI/I^2C pins (SCLOCK and SDATA/MOSI) also feature both input and output functions. Their equivalent I/O architectures are illustrated in Figure 42 and Figure 44, respectively, for SPI operation and in Figure 43 and Figure 45 for I^2C operation.

Notice that in I^2C mode (SPE = 0), the strong pull-up FET (Q1) is disabled leaving only a weak pull-up (Q2) present. By contrast, in SPI mode (SPE = 1), the strong pull-up FET (Q1) is controlled directly by SPI hardware, giving the pin push/pull capability.

In I^2C mode (SPE = 0), two pull-down FETs (Q3 and Q4) operate in parallel in order to provide an extra 60% or 70% of current sinking capability. In SPI mode, however, (SPE = 1), only one of the pull-down FETs (Q3) operates on each pin resulting in sink capabilities identical to that of Port 0 and Port 2 pins.

On the input path of SCLOCK, notice that a Schmitt trigger conditions the signal going to the SPI hardware to prevent false triggers (double triggers) on slow incoming edges. For incoming signals from the SCLOCK and SDATA pins going to I^2C hardware, a filter conditions the signals in order to reject glitches of up to 50 ns in duration.

Notice also that direct access to the SCLOCK and SDATA/MOSI pins is afforded through the SFR interface in I²C master mode. Therefore, if you are not using the SPI or I²C functions, you can use these two pins to give additional high current digital outputs.

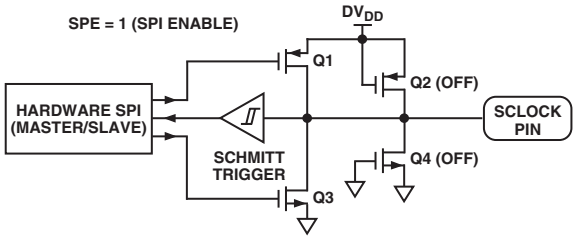


Figure 42. SCLOCK Pin I/O Functional Equivalent in SPI Mode

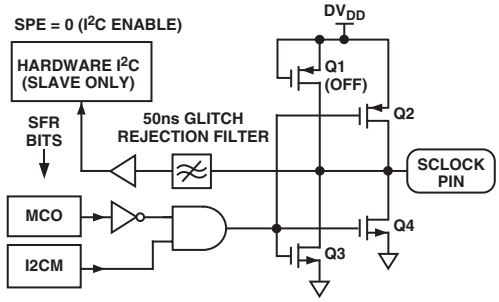


Figure 43. SCLOCK Pin I/O Functional Equivalent in I²C Mode

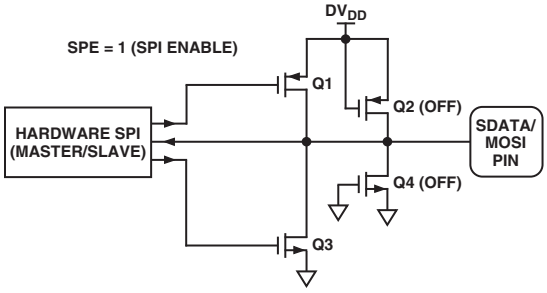


Figure 44. SDATA/MOSI Pin I/O Functional Equivalent in SPI Mode

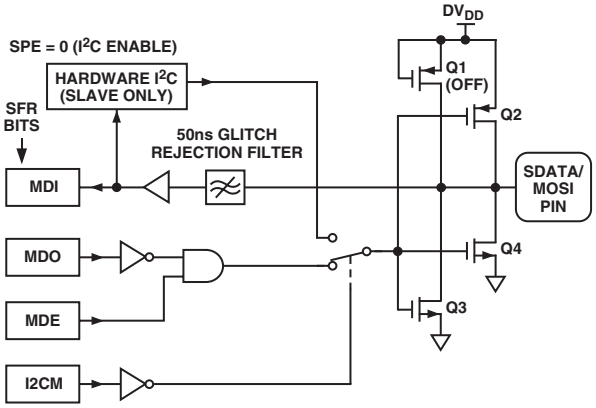


Figure 45. SDATA/MOSI Pin I/O Functional Equivalent in I²C Mode

As shown in Figure 46, the MISO pin in SPI master/slave operation offers the exact same pull-up and pull-down configuration as the MOSI pin in SPI slave/master operation.

The $\overline{SS}$ pin has a weak internal pull-up permanently enabled to prevent the $\overline{SS}$ input from floating. This pull-up can be easily overdriven by an external device to drive the $\overline{SS}$ pin low.

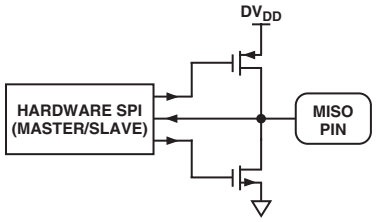


Figure 46. MISO Pin I/O Functional Equivalent

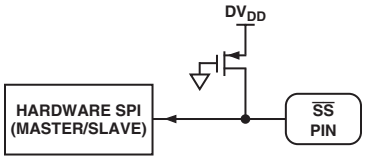


Figure 47. $\overline{SS}$ Pin I/O Functional Equivalent

Read-Modify-Write Instructions

Some 8051 instructions that read a port read the latch and others read the pin. The instructions that read the latch rather than the pins are the ones that read a value, possibly change it, and then rewrite it to the latch. These are called "read-modify-write" instructions. Listed below are the read-modify-write instructions. When the destination operand is a port, or a port bit, these instructions read the latch rather than the pin.

```
ANL           (Logical AND, e.g., ANL P1, A)
ORL           (Logical OR, e.g., ORL P2, A)
XRL           (Logical EX-OR, e.g., XRL P3, A)
JBC           (Jump If Bit = 1 and Clear Bit,
              e.g., JBC P1.1, LABEL
CPL           (Complement Bit, e.g., CPL P3.0)
INC           (Increment, e.g., INC P2)
DEC           (Decrement, e.g., DEC P2)
DJNZ          (Decrement and Jump IFf Not Zero,
              e.g.,DJNZ P3, LABEL)
MOV PX.Y, C*  (Move Carry to Bit Y of Port X)
CLR PX.Y*     (Clear Bit Y of Port X)
SETB PX.Y*    (Set Bit Y of Port X)
```

The reason that read-modify-write instructions are directed to the latch rather than the pin is to avoid a possible misinterpretation of the voltage level of a pin. For example, a port pin might be used to drive the base of a transistor. When a 1 is written to the bit, the transistor is turned on. If the CPU then reads the same port bit at the pin rather then the latch, it will read the base voltage of the transistor and interpret it as a Logic 0. Reading the latch rather than the pin will return the correct value of 1.

*These instruction read the port byte (all 8 bits), modify the addressed bit and then write the new byte back to the latch.

TIMERS/COUNTERS

The ADuC834 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 it can be viewed 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 the timers is achieved via three main SFRs. TMOD and TCON control the configuration of Timers 0 and 1 while T2CON configures Timer 2.

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

Table XXVI. 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 that 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 that 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

Table XXVII. 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 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).

*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.

Timer/Counter 0 and 1 Data Registers
Both timer 0 and timer 1 consist of two 8-bit registers. These can be used as independent registers or combined to be a single 16-bit register, depending on the timer mode configuration.

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

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

TIMER/COUNTER 0 AND 1 OPERATING MODES

The following paragraphs describe the operating modes for Timer/Counters 0 and 1. Unless otherwise noted, it should be assumed that these modes of operation are the same for 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 48 shows Mode 0 operation.

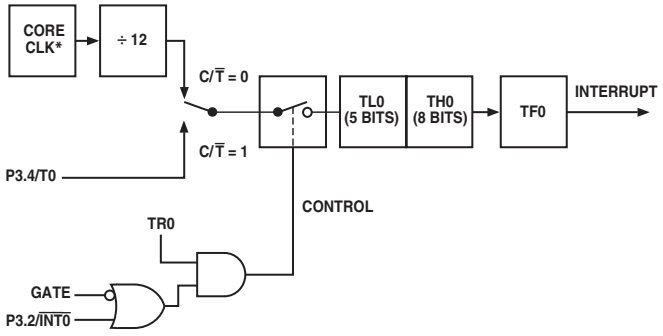


Figure 48. Timer/Counter 0, Mode 0

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. 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 49.

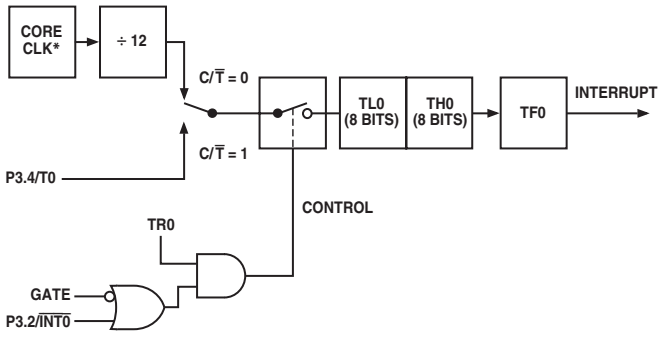


Figure 49. Timer/Counter 0, Mode 1

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 50. Overflow from TL0 not only sets TF0, but also reloads TL0 with the contents of TH0, which are preset by software. The reload leaves TH0 unchanged.

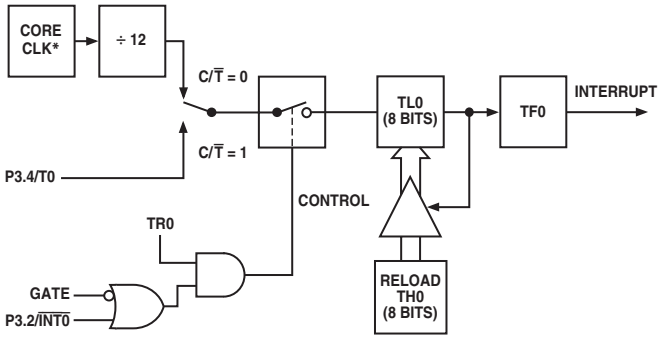


Figure 50. Timer/Counter 0, Mode 2

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 51. TL0 uses the timer 0 control bits: C/$\overline{T}$, 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.

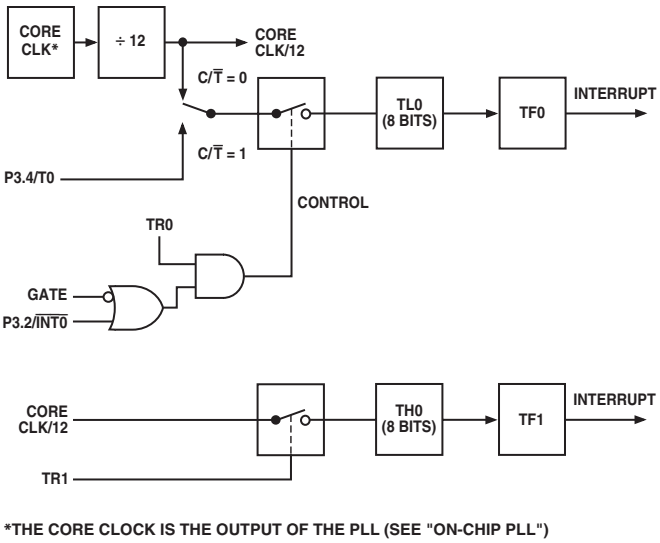


Figure 51. Timer/Counter 0, Mode 3

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 XXIX.

Table XXVIII. Timer 2 Operating Modes

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, selected by bit EXEN2 in T2CON. If EXEN2 = 0, 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, 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 52.

16-Bit Capture Mode

In the Capture Mode, there are again two options, selected by bit EXEN2 in T2CON. If EXEN2 = 0, Timer 2 is a 16-bit timer or counter that, upon overflowing, sets bit TF2, the Timer 2 overflow bit, which can be used to generate an interrupt. If EXEN2 = 1, Timer 2 still performs the above, but a l-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; EXF2, like TF2, can generate an interrupt. The Capture Mode is illustrated in Figure 53.

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. Therefore Timer 2 interrupts will not occur so they do not have to be disabled. However, in this mode, the EXF2 flag 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.

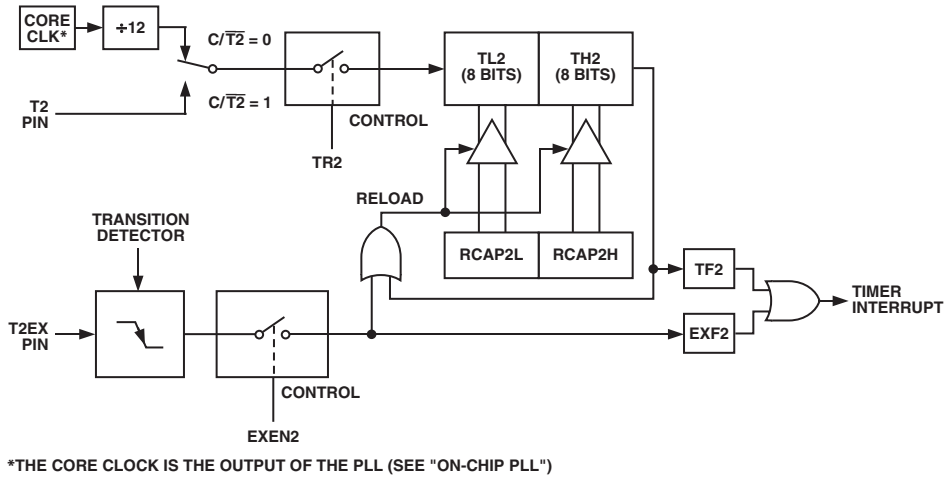


Figure 52. Timer/Counter 2, 16-Bit Autoreload Mode

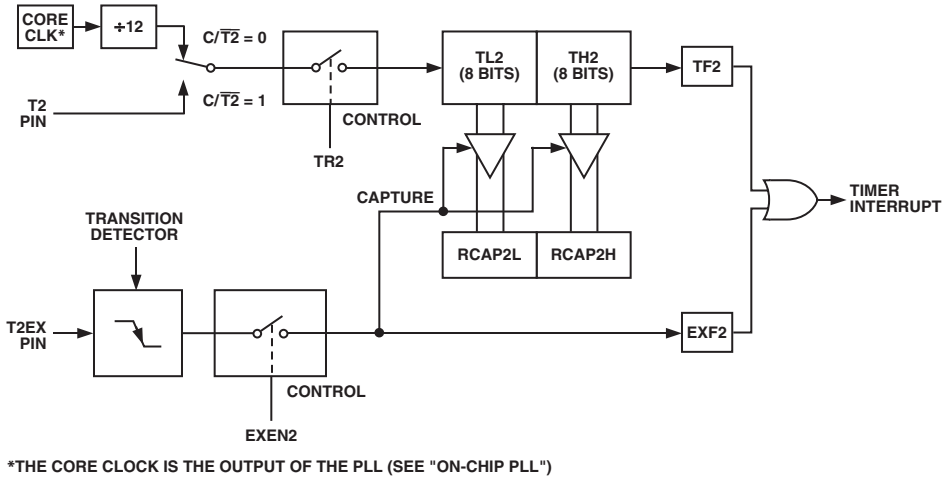


Figure 53. Timer/Counter 2, 16-Bit Capture Mode

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

Table XXIX. 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 = CDH, CCH, respectively.

RCAP2H and RCAP2L

Timer 2, Capture/Reload byte and low byte.
SFR Address = CBH, CAH, respectively.

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 comprises the following registers:

SBUF

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

SCON	UART Serial Port Control Registers
SFR Address	98H
Power-On Default Value	00H
Bit Addressable	Yes

Table XXX. 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 (f_{CORE}/12) 0 1 Mode 1: 8-bit UART, variable baud rate 1 0 Mode 2: 9-bit UART, fixed baud rate (f_{CORE}/64) or (f_{CORE}/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.

UART OPERATING MODES

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 8 bits are transmitted with the least-significant bit (LSB) first, as shown in Figure 54.

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.

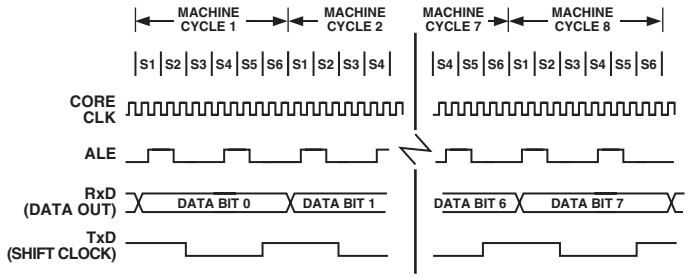


Figure 54. UART Serial Port Transmission, Mode 0

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 can be set by Timer 1 or Timer 2 (or both). Alternatively, a dedicated baud rate generator, Timer 3, is provided on-chip to generate high speed, very accurate baud rates.

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 55.

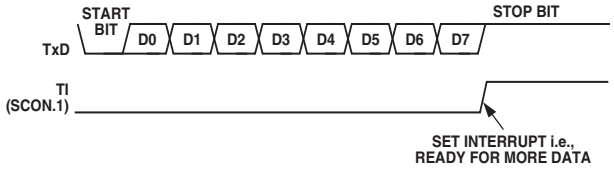


Figure 55. UART Serial Port Transmission, Mode 0

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:

$$Mode\ 0\ Baud\ Rate = \frac{f_{CORE*}}{12}$$

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

$$Mode\ 2\ Baud\ Rate = \frac{f_{CORE*} \times 2^{SMOD}}{64}$$

Mode 1 and 3 Baud Rate Generation
Traditionally, 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). On the ADuC834, however, the baud rate can also be generated via a separate baud rate generator to achieve higher baud rates and allow all three to be used for other functions.

*f_{CORE} refers to the output of the PLL as described in the "On-Chip PLL" section.

BAUD RATE GENERATION USING TIMER 1 AND TIMER 2

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:

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

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

$$Mode\ 1\ and\ Mode\ 3\ Baud\ Rate = \frac{2^{SMOD} \times f_{CORE}}{32 \times 12\left(256 - TH1\right)}$$

A very low baud rate can also be achieved with Timer 1 by leaving the Timer 1 interrupt enabled, configuring the timer to run as a 16-bit timer (high nibble of TMOD = 0100 binary), and using the Timer 1 interrupt to do a 16-bit software reload. Table XXXI shows some commonly-used baud rates and how they might be calculated from a core clock frequency of 1.5728 MHz and 12.58 MHz using Timer 1. Generally speaking, a 5% error is tolerable using asynchronous (start/stop) communications.

Table XXXI. Commonly Used Baud Rates, Timer 1

Ideal Baud	Core CLK	SMOD Value	TH1-Reload Value	Actual Baud	% Error
9600	12.58	1	–7 (F9H)	9362	2.5
1600	12.58	1	–27 (E5H)	1627	1.1
1200	12.58	1	–55 (C9H)	1192	0.7
1200	1.57	1	–7 (F9H)	1170	2.5

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.

$$Mode\ 1\ and\ Mode\ 3\ Baud\ Rate = \left(1/16\right) \times \left(Timer\ 2\ Overflow\ Rate\right)$$

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. Thus, it increments six times faster than Timer 1, and therefore baud rates six times faster are possible. Because Timer 2 has a 16-bit autoreload capability, very low baud rates are still possible.

Timer 2 is selected as the baud rate generator by setting the TCLK 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 56. In this case, the baud rate is given by the formula:

$$Mode\ 1\ and\ Mode\ 3\ Baud\ Rate = \frac{f_{CORE}}{32 \times \left(65536 - RCAP2H/L\right)}$$

Table XXXII shows some commonly used baud rates and how they might be calculated from a core clock frequency of 1.5728 MHz and 12.5829 MHz using Timer 2.

Table XXXII. Commonly Used Baud Rates, Timer 2

Ideal Baud	Core CLK	RCAP2H Value	RCAP2L Value	Actual Baud	% Error
19200	12.58	–1 (FFH)	–20 (ECH)	19661	2.4
9600	12.58	–1 (FFH)	–41 (D7H)	9591	0.1
1600	12.58	–1 (FFH)	–164 (5CH)	2398	0.1
1200	12.58	–2 (FEH)	–72 (B8H)	1199	0.1
9600	1.57	–1 (FFH)	–5 (FBH)	9830	2.4
1600	1.57	–1 (FFH)	–20 (ECH)	1658	2.4
1200	1.57	–1 (FFH)	–41 (D7H)	1199	0.1

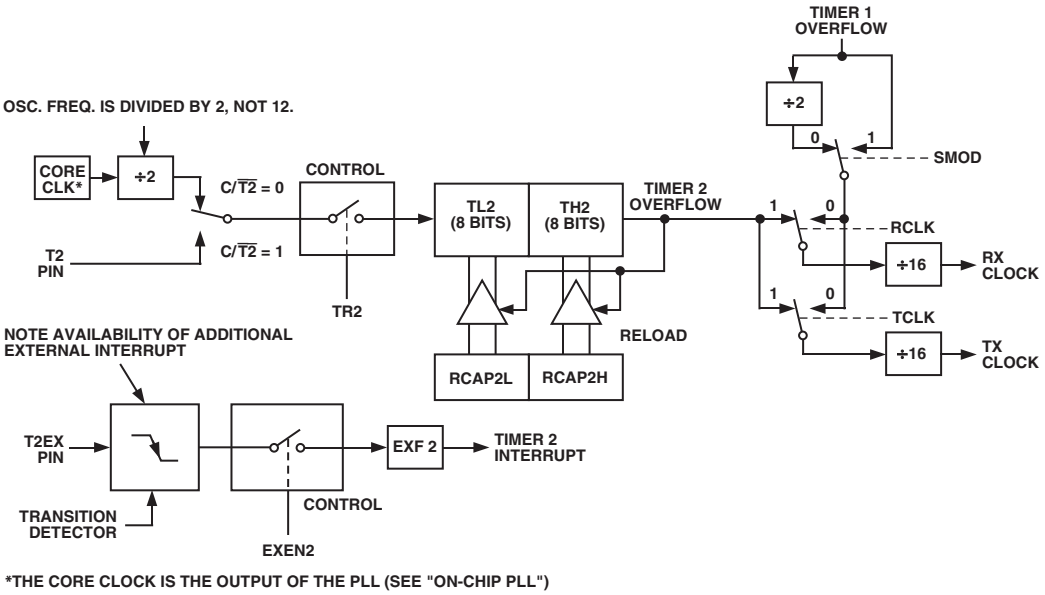


Figure 56. Timer 2, UART Baud Rates

BAUD RATE GENERATION USING TIMER 3

The high integer dividers in a UART block means that high speed baud rates are not always possible using some particular crystals. e.g., using a 12 MHz crystal, a baud rate of 115200 is not possible. To address this problem the ADuC834 has added a dedicated baud rate timer (Timer 3) specifically for generating highly accurate baud rates.

Timer 3 can be used instead of Timer 1 or Timer 2 for generating very accurate high speed UART baud rates including 115200 and 230400. Timer 3 also allows a much wider range of baud rates to be obtained. In fact, every desired bit rate from 12 bits to 393216 bits can be generated to within an error of ±0.8%. Timer 3 also frees up the other three timers allowing them to be used for different applications. A block diagram of Timer 3 is shown in Figure 57.

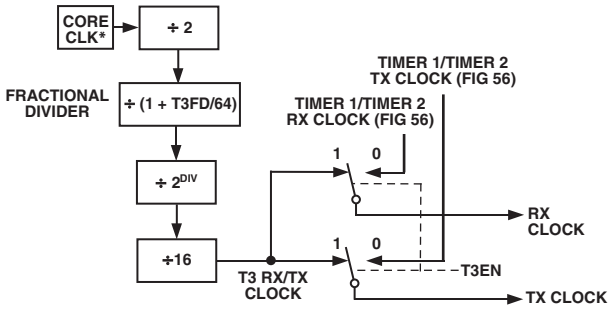


Figure 57. Timer 3, UART Baud Rates

Two SFRs (T3CON and T3FD) are used to control Timer 3. T3CON is the baud rate control SFR, allowing Timer 3 to be used to set up the UART baud rate, and setting up the binary divider (DIV).

Table XXXIII. T3CON SFR Bit Designations

Bit	Name	Description			
7	T3EN	Set to enable Timer 3 to generate the baud rate. When set PCON.7, T2CON.4 and T2CON.5 are ignored. Cleared to let the baud rate be generated as per a standard 8052.			
6	—	Reserved for future use			
5	—	Reserved for future use			
4	—	Reserved for future use			
3	—	Reserved for future use			
2	DIV2	Binary Divider Factor			
1	DIV1	DIV2	DIV1	DIV0	Bin Divider
0	DIV0	0	0	0	1
		0	0	1	2
		0	1	0	4
		0	1	1	8
		1	0	0	16
		1	0	1	32
		1	1	0	64
		1	1	1	128

The appropriate value to write to the DIV2-1-0 bits can be calculated using the following formula where f_{CORE} is the output of the PLL as described in the "On-Chip PLL" description.

Note: The *DIV* value must be rounded down.

$$DIV = \frac{\log\left(\frac{f_{CORE}}{32 \times Baud\ Rate}\right)}{\log(2)}$$

T3FD is the fractional divider ratio required to achieve the required baud rate. We can calculate the appropriate value for *T3FD* using the following formula.

Note: *T3FD* should be rounded to the nearest integer.

$$T3FD = \frac{2 \times f_{CORE}}{2^{DIV} \times Baud\ Rate} - 64$$

Once the values for *DIV* and *T3FD* are calculated, the actual baud rate can be calculated using the following formula:

$$Actual\ Baud\ Rate = \frac{2 \times f_{CORE}}{2^{DIV} \times (T3FD + 64)}$$

For a baud rate of 115200 while operating from the maximum core frequency (CD = 0) we have:

$$DIV = \log(12582912/32 \times 115200) / \log 2 = 1.77 = 1$$

$$T3FD = (2 \times 12.582912) / (2^1 \times 115200) - 64 = 45.22 = 2Dh$$

Therefore, the actual baud rate is 115439 bits.

Table XXXIV. Commonly Used Baud Rates Using Timer 3

Ideal Baud	CD	DIV	T3CON	T3FD	% Error
230400	0	0	80H	2DH	0.2
115200	0	1	81H	2DH	0.2
115200	1	0	80H	2DH	0.2
57600	0	2	82H	2DH	0.2
57600	1	1	81H	2DH	0.2
57600	2	0	80H	2DH	0.2
38400	0	3	83H	12H	0.1
38400	1	2	82H	12H	0.1
38400	2	1	81H	12H	0.1
38400	3	0	80H	12H	0.1
19200	0	4	84H	12H	0.1
19200	1	3	83H	12H	0.1
19200	2	2	82H	12H	0.1
19200	3	1	81H	12H	0.1
19200	4	0	80H	12H	0.1
9600	0	5	85H	12H	0.1
9600	1	4	84H	12H	0.1
9600	2	3	83H	12H	0.1
9600	3	2	82H	12H	0.1
9600	4	1	81H	12H	0.1
9600	5	0	80H	12H	0.1
38400	0	3	83H	12H	0.1

INTERRUPT SYSTEM

The ADuC834 provides a total of 11 interrupt sources with two priority levels. The control and configuration of the interrupt system is carried out through three interrupt-related SFRs. These are the IE (Interrupt Enable) Register, the IP (Interrupt Priority Register) and the IEIP2 (Secondary Interrupt Enable/Priority SFR) Registers. Their bit definitions are given in the Tables XXXV– XXXVII.

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

Table XXXV. 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

Table XXXVI. 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

Table XXXVII. 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 used to determine which interrupt is serviced first. The polling sequence is shown in Table XXXVIII.

Table XXXVIII. Priority within an Interrupt Level

Source	Priority	Description
PSMI	1 (Highest)	Power Supply Monitor Interrupt
WDS	2	Watchdog Interrupt
IE0	3	External Interrupt 0
RDY0/RDY1	4	ADC Interrupt
TF0	5	Timer/Counter 0 Interrupt
IE1	6	External Interrupt 1
TF1	7	Timer/Counter 1 Interrupt
ISPI/I2CI	8	SPI Interrupt
RI + TI	9	Serial Interrupt
TF2 + EXF2	10	Timer/Counter 2 Interrupt
TII	11 (Lowest)	Time Interval Counter Interrupt

Interrupt Vectors

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

Table XXXIX. Interrupt Vector Addresses

Source	Vector Address
IE0	0003H
TF0	000BH
IE1	0013H
TF1	001BH
RI + TI	0023H
TF2 + EXF2	002BH
RDY0/RDY1 (ADC)	0033H
ISPI/I2CI	003BH
PSMI	0043H
TII	0053H
WDS (WDIR = 1)*	005BH

*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 ($\overline{EA}$) 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.

ADuC834 HARDWARE DESIGN CONSIDERATIONS

This section outlines some of the key hardware design considerations that must be addressed when integrating the ADuC834 into any hardware system.

External Memory Interface

In addition to its internal program and data memories, the ADuC834 can access up to 64 Kbytes of external program memory (ROM/PROM/and so on) and up to 16 Mbytes of external data memory (SRAM).

To select from which code space (internal or external program memory) to begin executing code, tie the $\overline{EA}$ (external access) pin high or low, respectively. When $\overline{EA}$ is high (pulled up to V_{DD}), user program execution will start at Address 0 in the internal 62 Kbytes Flash/EE code space. When $\overline{EA}$ is low (tied to ground) user program execution will start at Address 0 in the external code space. When executing from internal code space, accesses to the program space above F7FFH (62 Kbytes) will be read as NOP instructions.

Note that a second very important function of the $\overline{EA}$ pin is described in the Single Pin Emulation Mode section.

External program memory (if used) must be connected to the ADuC834 as illustrated in Figure 58. Sixteen I/O lines (Ports 0 and 2) are dedicated to bus functions during external program memory fetches. Port 0 (P0) serves as a multiplexed address/databus. It emits the low byte of the program counter (PCL) as an address, and then goes into a high impedance input state awaiting the arrival of the code byte from the program memory. During the time that the low byte of the program counter is valid on P0, the signal ALE (Address Latch Enable) clocks this byte into an external address latch. Meanwhile, Port 2 (P2) emits the high byte of the program counter (PCH), and $\overline{PSEN}$ strobes the EPROM and the code byte is read into the ADuC834.

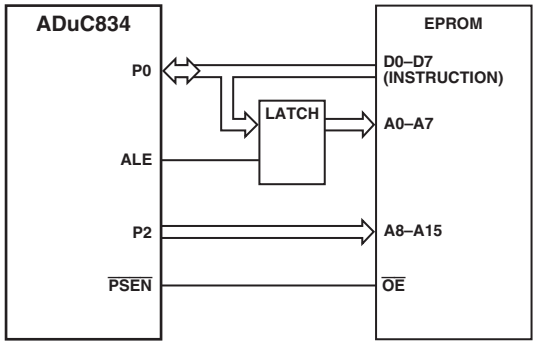


Figure 58. External Program Memory Interface

Note that program memory addresses are always 16 bits wide, even in cases where the actual amount of program memory used is less than 64 Kbytes. External program execution sacrifices two of the 8-bit ports (P0 and P2) to the function of addressing the program memory. While executing from external program memory, Ports 0 and 2 can be used simultaneously for read/write access to external data memory, but not for general-purpose I/O.

Though both external program memory and external data memory are accessed using some of the same pins, the two are completely independent of each other from a software point of view. For example, the chip can read/write external data memory while executing from external program memory.

Figure 59 shows a hardware configuration for accessing up to 64 Kbytes of external data memory. This interface is standard to any 8051 compatible MCU.

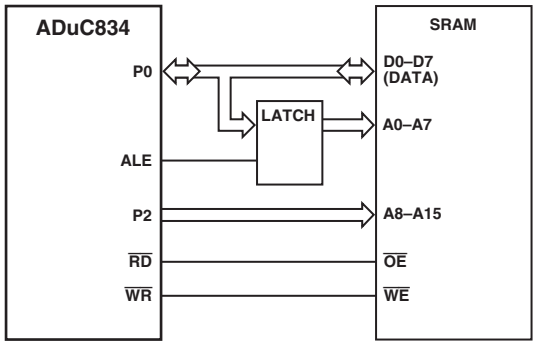


Figure 59. External Data Memory Interface (64 Kbytes Address Space)

If access to more than 64 Kbytes of RAM is desired, a feature unique to the MicroConverter allows addressing up to 16 Mbytes of external RAM simply by adding an additional latch as illustrated in Figure 60.

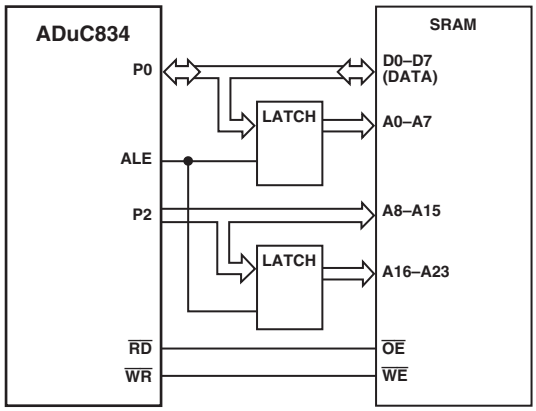


Figure 60. External Data Memory Interface (16 Mbytes Address Space)

In either implementation, Port 0 (P0) serves as a multiplexed address/databus. It emits the low byte of the data pointer (DPL) as an address, which is latched by ALE prior to data being placed on the bus by the ADuC834 (write operation) or the external data memory (read operation). Port 2 (P2) provides the data pointer page byte (DPP) to be latched by ALE, followed by the data pointer high byte (DPH). If no latch is connected to P2, DPP is ignored by the SRAM, and the 8051 standard of 64 Kbyte external data memory access is maintained.

Detailed timing diagrams of external program and data memory read and write access can be found in the Timing Specification sections of this data sheet.

Power Supplies

The ADuC834's operational power supply voltage range is 2.7 V to 5.25 V. Although the guaranteed data sheet specifications are given only for power supplies within 2.7 V to 3.6 V or 5% of the nominal 5 V level, the chip will function equally well at any power supply level between 2.7 V and 5.25 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 DV_{DD} line. In this mode, the part can also operate with split supplies; that is, using different voltage supply levels for each supply. For example, this means that the system can be designed to operate with a DV_{DD} voltage level of 3 V while the AV_{DD} level can be at 5 V, or vice-versa if required. A typical split supply configuration is shown in Figure 61.

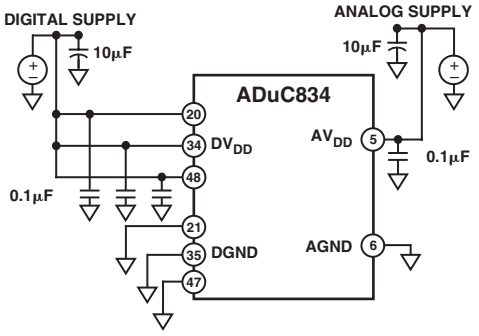


Figure 61. External Dual Supply Connections

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 62. In this configuration, other analog circuitry (such as op amps, voltage reference, and so on) can be powered from the AV_{DD} supply line as well.

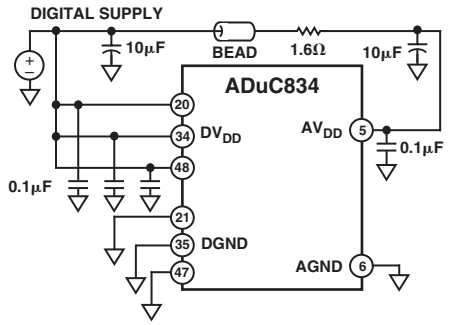


Figure 62. External Single Supply Connections

Notice that in both Figure 61 and Figure 62 a large value (10 µF) reservoir capacitor sits on DV_{DD} and a separate 10 µF capacitor sits on AV_{DD}. Also, local decoupling capacitors (0.1 µF) are located at each V_{DD} pin of the chip. As per standard design practice, be sure to include all of these capacitors and ensure the smaller capacitors are closest to each V_{DD} pin with lead 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 ADuC834 should be referenced to the same system ground reference point.

Power-On Reset Operation

An internal POR (Power-On Reset) is implemented on the ADuC834. For DV_{DD} below 2.45 V, the internal POR will hold the ADuC834 in reset. As DV_{DD} rises above 2.45 V, an internal timer will time out for typically 128 ms before the part is released from reset. The user must ensure that the power supply has reached a stable 2.7 V minimum level by this time. Likewise on power-down, the internal POR will hold the ADuC834 in reset until the power supply has dropped below 1 V. Figure 63 illustrates the operation of the internal POR in detail.

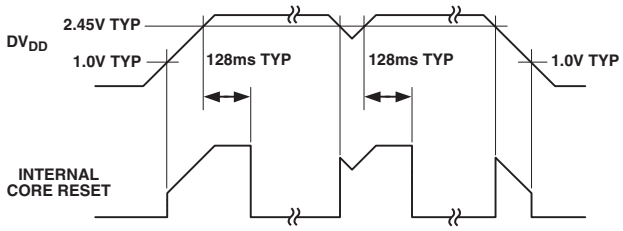


Figure 63. Internal Power-on-Reset Operation

Power Consumption

The DV_{DD} power supply current consumption is specified in normal, idle, and power-down modes. The AV_{DD} power supply current is specified with the analog peripherals disabled. The normal mode power consumption represents the current drawn from DV_{DD} by the digital core. The other on-chip peripherals (watchdog timer, power supply monitor, and so on) consume negligible current and are therefore lumped in with the normal operating current here. Of course, the user must add any currents sourced by the parallel and serial I/O pins, and those sourced by the DAC in order to determine the total current needed at the ADuC834's DV_{DD} and AV_{DD} supply pins. Also, current drawn from the DV_{DD} 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, and ALE and $\overline{\text{PSEN}}$ outputs go high 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) while ALE and $\overline{\text{PSEN}}$ outputs are held low. During full Power-Down mode with the oscillator and wake-up timer running, the ADuC834 typically consumes a total of 15 μA. There are five ways of terminating Power-Down mode:

Asserting the RESET Pin (Pin 15)
Returns to Normal Mode. All registers are set to their reset default value and program execution starts at the reset vector once the RESET pin is deasserted.

Cycling Power
All registers are set to their default state and program execution starts at the reset vector approximately 128 ms later.

Time Interval Counter (TIC) Interrupt
If the OSC_PD bit in the PLLCON SFR is clear, the 32 kHz oscillator will remain powered up even in Power-Down mode. If the Time Interval Counter (Wakeup/RTC timer) is enabled, a TIC interrupt will wake the ADuC834 up from Power-Down mode. The CPU services the TIC interrupt. The RETI at the end of the TIC ISR will return the core to the instruction after that which enabled power-down.

SPI Interrupt
If the SERIPD Bit in the PCON SFR is set, then an SPI interrupt, if enabled, will wake the ADuC834 up from Power-Down mode. The CPU services the SPI interrupt. The RETI at the end of the ISR will return the core to the instruction after that which enabled power-down.

$\overline{\text{INT0}}$ Interrupt
If the INT0PD bit in the PCON SFR is set, an external interrupt 0, if enabled, will wake up the ADuC834 from power-down. The CPU services the SPI interrupt. The RETI at the end of the ISR will return the core to the instruction after that which enabled power-down.

Wake-Up from Power-Down Latency
Even with the 32 kHz crystal enabled during power-down, the PLL will take some time to lock after a wake-up from power-down. Typically, the PLL will take about 1 ms to lock. During this time, code will execute but not at the specified frequency. Some operations require an accurate clock, for example UART communications, to achieve specified 50/60 Hz rejection from the ADCs. The following code may be used to wait for the PLL to lock:

```
WAITFORLOCK:
    MOV     A, PLLCON
    JNB     ACC.6, WAITFORLOCK
```

If the crystal has been powered down during power-down, there is an additional delay associated with the startup of the crystal oscillator before the PLL can lock. 32 kHz crystals are inherently slow to oscillate, typically taking about 150 ms. Once again, during this time before lock, code will execute but the exact frequency of the clock cannot be guaranteed. Again for any timing sensitive operations, it is recommended to wait for lock using the lock bit in PLLCON as shown above.

Grounding and Board Layout Recommendations
As with all high resolution data converters, special attention must be paid to grounding and PC board layout of ADuC834-based designs in order to achieve optimum performance from the ADCs and DAC.

Although the ADuC834 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 ADuC834, as illustrated in the simplified example of Figure 64a. 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 ADuC834 since a ground loop would result. In these cases, tie the ADuC834's AGND and DGND Pins all to the analog ground plane, as illustrated in Figure 64b. 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 ADuC834 can then be placed between the digital and analog sections, as illustrated in Figure 64c.

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 64b 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 64c. 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.

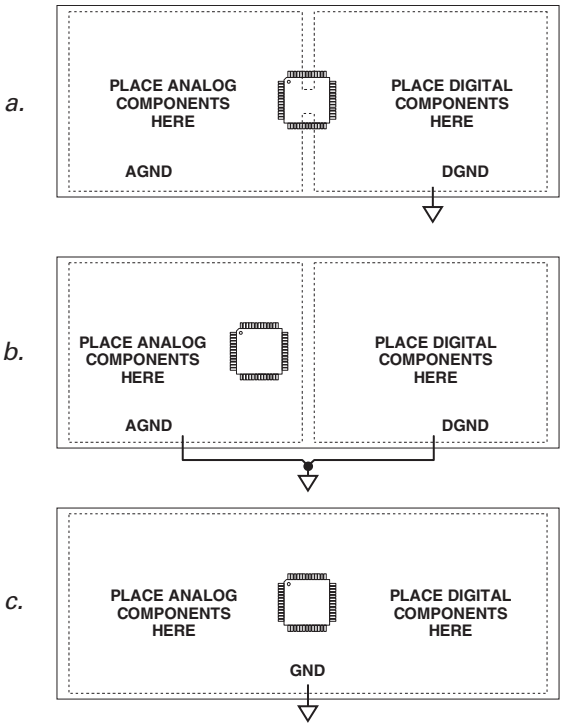


Figure 64. System Grounding Schemes

If the user plans to connect fast logic signals (rise/fall time < 5 ns) to any of the ADuC834's digital inputs, add a series resistor to each relevant line to keep rise and fall times longer than 5 ns at the ADuC834 input pins. A value of 100 Ω or 200 Ω is usually sufficient to prevent high speed signals from coupling capacitively into the ADuC834 and affecting the accuracy of ADC conversions.

ADuC834 System Self-Identification

In some hardware designs, it may be an advantage for the software running on the ADuC834 target to identify the host MicroConverter. For example, code running on the ADuC834 may also be used with the ADuC824 or the ADuC816, and is required to operate differently.

The CHIPID SFR is a read-only register located at SFR address C2H. The upper nibble of this SFR designates the MicroConverter within the Σ-Δ ADC family. User software can read this SFR to identify the host MicroConverter and thus execute slightly different code if required. The CHIPID SFR reads as follows for the Σ-Δ ADC family of MicroConverter products.

ADuC836	CHIPID = 3xH
ADuC834	CHIPID = 2xH
ADuC824	CHIPID = 0xH
ADuC816	CHIPID = 1xH

Clock Oscillator

As described earlier, the core clock frequency for the ADuC834 is generated from an on-chip PLL that locks onto a multiple (384 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 (32 and 33) as shown in Figure 65.

As shown in the typical external crystal connection diagram in Figure 65, 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.

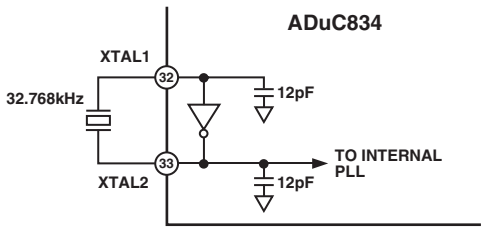


Figure 65. External Parallel Resonant Crystal Connections Other Hardware Considerations

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.

OTHER HARDWARE CONSIDERATIONS

In-Circuit Serial Download Access

Nearly all ADuC834 designs will want to take advantage of the in-circuit reprogrammability of the chip. This is accomplished by a connection to the ADuC834's UART, which requires an external RS-232 chip for level translation if downloading code from a PC. Basic configuration of an RS-232 connection is illustrated in Figure 66 with a simple ADM3202-based circuit. If users would rather not include an RS-232 chip onto the target board, refer to the application note *uC006–A 4-Wire UART-to-PC Interface* available at www.analog.com/microconverter, for a simple (and zero-cost-per-board) method of gaining in-circuit serial download access to the ADuC834.

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-down resistor that can be jumpered onto the $\overline{\text{PSEN}}$ pin, as shown in Figure 66. To get the ADuC834 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. With the jumper removed, the device will power-on in Normal mode (and run the program) whenever power is cycled or RESET is toggled.

Note that $\overline{\text{PSEN}}$ is normally an output (as described in the External Memory Interface section) and it is sampled as an input only on the falling edge of RESET (i.e., at power-up or upon an external manual reset). Note also that if any external circuitry unintentionally pulls $\overline{\text{PSEN}}$ low during power-up or reset events, it could cause the chip to enter Download Mode and therefore fail to begin user code execution as it should. To prevent this, ensure that no external signals are capable of pulling the $\overline{\text{PSEN}}$ pin low, except for the external $\overline{\text{PSEN}}$ jumper itself.

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 ADuC834 device, (unlike "ROM monitor" type debuggers) and therefore no external memory is needed to enable in-system debug sessions.

Single-Pin Emulation Mode

Also built into the ADuC834 is a dedicated controller for single-pin in-circuit emulation (ICE) using standard production ADuC834 devices. In this mode, emulation access is gained by connection to a single pin, the $\overline{\text{EA}}$ pin. Normally, this pin is hard-wired either high or low to select execution from internal or external program memory space, as described earlier. To enable single-pin emulation mode, however, users will need to pull the $\overline{\text{EA}}$ pin high through a 1 kΩ resistor as shown in Figure 66. The emulator will then connect to the 2-pin header also shown in Figure 66. 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. As represented in Figure 66, 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).

Typical System Configuration

A typical ADuC834 configuration is shown in Figure 66. It summarizes some of the hardware considerations discussed in the previous paragraphs.

Figure 66 also includes connections for a typical analog measurement application of the ADuC834, namely an interface to an RTD (Resistive Temperature Device). The arrangement shown is commonly referred to as a 4-wire RTD configuration.

Here, the on-chip excitation current sources are enabled to excite the sensor. The excitation current flows directly through the RTD generating a voltage across the RTD proportional to its resistance. This differential voltage is routed directly to the positive and negative inputs of the primary ADC (AIN1, AIN2 respectively). The same current that excited the RTD also flows through a series resistance R_{REF} generating a ratiometric voltage reference V_{REF}. The ratiometric voltage reference ensures that variations in the excitation current do not affect the measurement system as the input voltage from the RTD and reference voltage across R_{REF} vary ratiometrically with the excitation current. Resistor R_{REF} must, however, have a low temperature coefficient to avoid errors in the reference voltage over temperature. R_{REF} must also be large enough to generate at least a 1 V voltage reference.

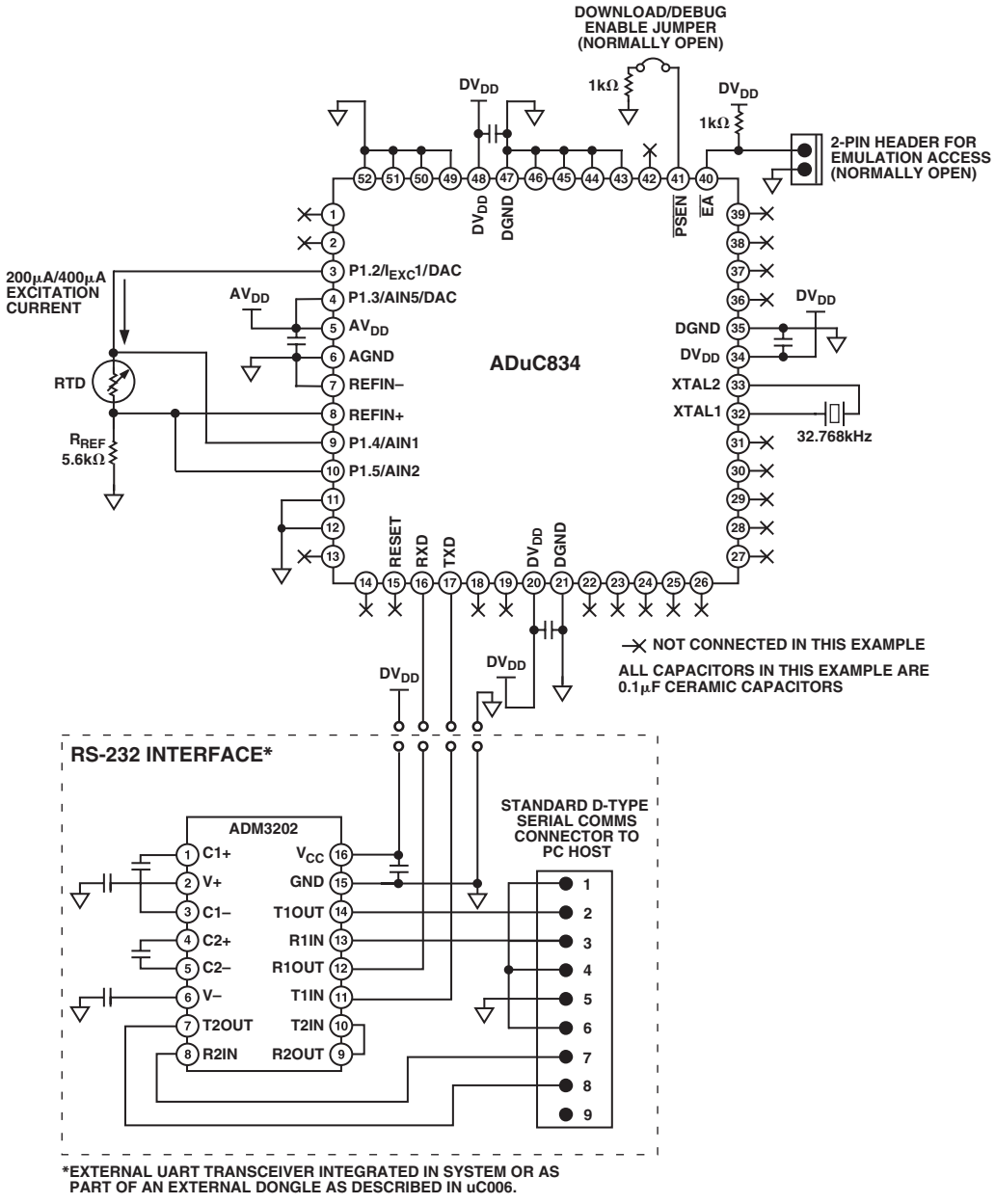


Figure 66. Typical System Configuration

QUICKSTART DEVELOPMENT SYSTEM

The QuickStart Development System is a full featured, low cost development tool suite supporting the ADuC834. The system consists of the following PC-based (Windows® compatible) hardware and software development tools.

Hardware:	ADuC834 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/Emulator:	Serial Port/Single Pin Debugger/Emulator with Assembly and C Source debug
Miscellaneous Other:	CD-ROM Documentation and Two Additional Prototype Devices

Figures 67 shows the typical components of a QuickStart Development System while Figure 68 shows a typical debug session. A brief description of some of the software tools' components in the QuickStart Development System is given below.

Figure 67. Components of the QuickStart Development System

Download—In Circuit 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.

Debugger/Emulator—In-Circuit Debugger/Emulator

The Debugger/Emulator is a Windows application that allows the user to debug code execution on silicon using the MicroConverter UART serial port or via a single pin to provide non intrusive debug. The debugger provides access to all on-chip peripherals during a typical debug session, including single-step and multiple break-point code execution control. C source and assembly level debug are both possible with the emulator.

ADSIM—Windows Simulator

The Simulator is a Windows application that fully simulates 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.

Figure 68. Typical Debug Session

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	69
t_{CKL}	XTAL1 Width Low		6.26		μs	69
t_{CKH}	XTAL1 Width High		6.26		μs	69
t_{CKR}	XTAL1 Rise Time		9		μs	69
t_{CKF}	XTAL1 Fall Time		9		μs	69
$1/t_{CORE}$	ADuC834 Core Clock Frequency[4]	0.098		12.58	MHz	
t_{CORE}	ADuC834 Core Clock Period[5]		0.636		μs	
t_{CYC}	ADuC834 Machine Cycle Time[6]	0.95	7.6	122.45	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 70.
[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 70.
[3]C_{LOAD} for Port0, ALE, $\overline{PSEN}$ outputs = 100 pF; C_{LOAD} for all other outputs = 80 pF unless otherwise noted.
[4]ADuC834 internal PLL locks onto a multiple (384 times) the external crystal frequency of 32.768 kHz to provide a stable 12.583 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 1.57 MHz.
[6]ADuC834 Machine Cycle Time is nominally defined as 12/Core_Clk.

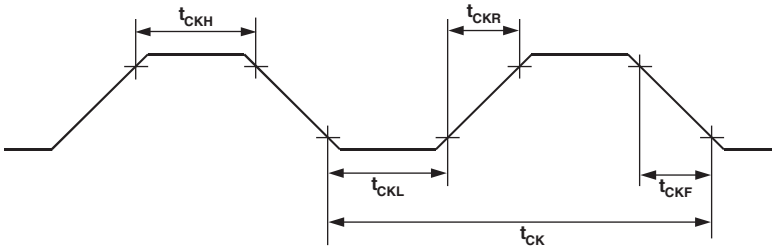


Figure 69. XTAL1 Input

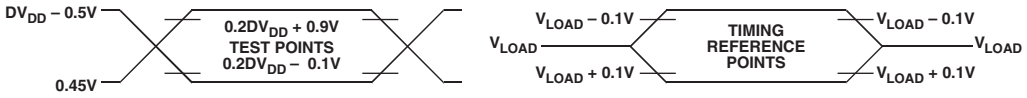


Figure 70. Timing Waveform Characteristics

Parameter		12.58 MHz Core_Clk Min	12.58 MHz Core_Clk Max	Variable Core_Clk Min	Variable Core_Clk Max	Unit	Figure
EXTERNAL PROGRAM MEMORY							
t_{LHLL}	ALE Pulsewidth	119		$2t_{CORE} - 40$		ns	71
t_{AVLL}	Address Valid to ALE Low	39		$t_{CORE} - 40$		ns	71
t_{LLAX}	Address Hold after ALE Low	49		$t_{CORE} - 30$		ns	71
t_{LLIV}	ALE Low to Valid Instruction In		218		$4t_{CORE} - 100$	ns	71
t_{LLPL}	ALE Low to $\overline{PSEN}$ Low	49		$t_{CORE} - 30$		ns	71
t_{PLPH}	$\overline{PSEN}$ Pulsewidth	193		$3t_{CORE} - 45$		ns	71
t_{PLIV}	$\overline{PSEN}$ Low to Valid Instruction In		133		$3t_{CORE} - 105$	ns	71
t_{PXIX}	Input Instruction Hold after $\overline{PSEN}$	0		0		ns	71
t_{PXIZ}	Input Instruction Float after $\overline{PSEN}$		54		$t_{CORE} - 25$	ns	71
t_{AVIV}	Address to Valid Instruction In		292		$5t_{CORE} - 105$	ns	71
t_{PLAZ}	$\overline{PSEN}$ Low to Address Float		25		25	ns	71
t_{PHAX}	Address Hold after $\overline{PSEN}$ High	0		0		ns	71

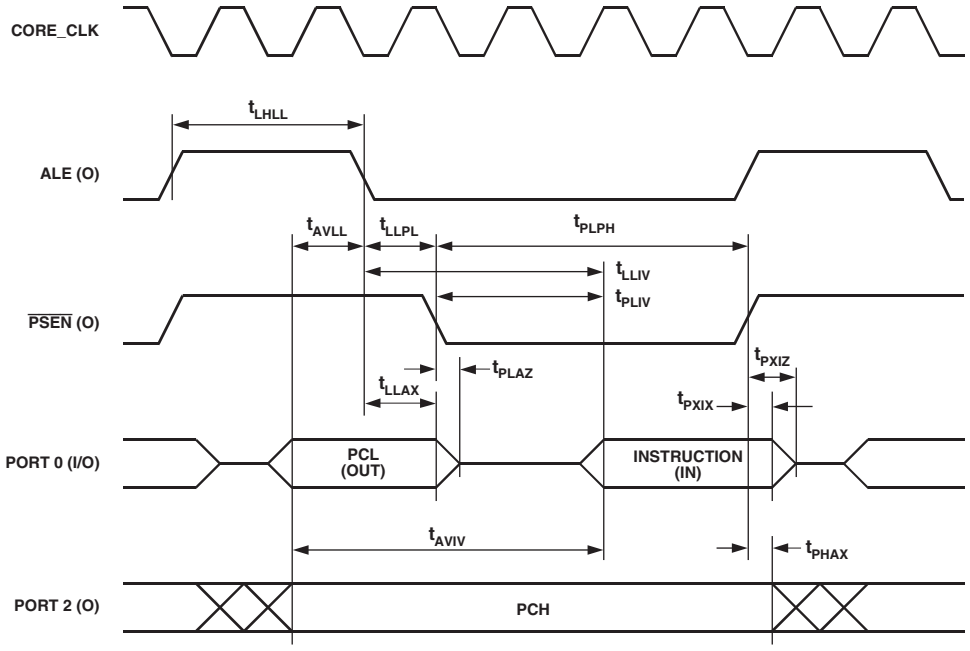


Figure 71. External Program Memory Read Cycle

Parameter		12.58 MHz Core_Clk Min	Max	Variable Core_Clk Min	Max	Unit	Figure
EXTERNAL DATA MEMORY READ CYCLE							
t_{RLRH}	$\overline{RD}$ Pulsewidth	377		$6t_{CORE} - 100$		ns	72
t_{AVLL}	Address Valid after ALE Low	39		$t_{CORE} - 40$		ns	72
t_{LLAX}	Address Hold after ALE Low	44		$t_{CORE} - 35$		ns	72
t_{RLDV}	$\overline{RD}$ Low to Valid Data In		232		$5t_{CORE} - 165$	ns	72
t_{RHDX}	Data and Address Hold after $\overline{RD}$	0		0		ns	72
t_{RHDZ}	Data Float after $\overline{RD}$		89		$2t_{CORE} - 70$	ns	72
t_{LLDV}	ALE Low to Valid Data In		486		$8t_{CORE} - 150$	ns	72
t_{AVDV}	Address to Valid Data In		550		$9t_{CORE} - 165$	ns	72
t_{LLWL}	ALE Low to $\overline{RD}$ Low	188	288	$3t_{CORE} - 50$	$3t_{CORE} + 50$	ns	72
t_{AVWL}	Address Valid to $\overline{RD}$ Low	188		$4t_{CORE} - 130$		ns	72
t_{RLAZ}	$\overline{RD}$ Low to Address Float		0		0	ns	72
t_{WHLH}	$\overline{RD}$ High to ALE High	39	119	$t_{CORE} - 40$	$t_{CORE} + 40$	ns	72

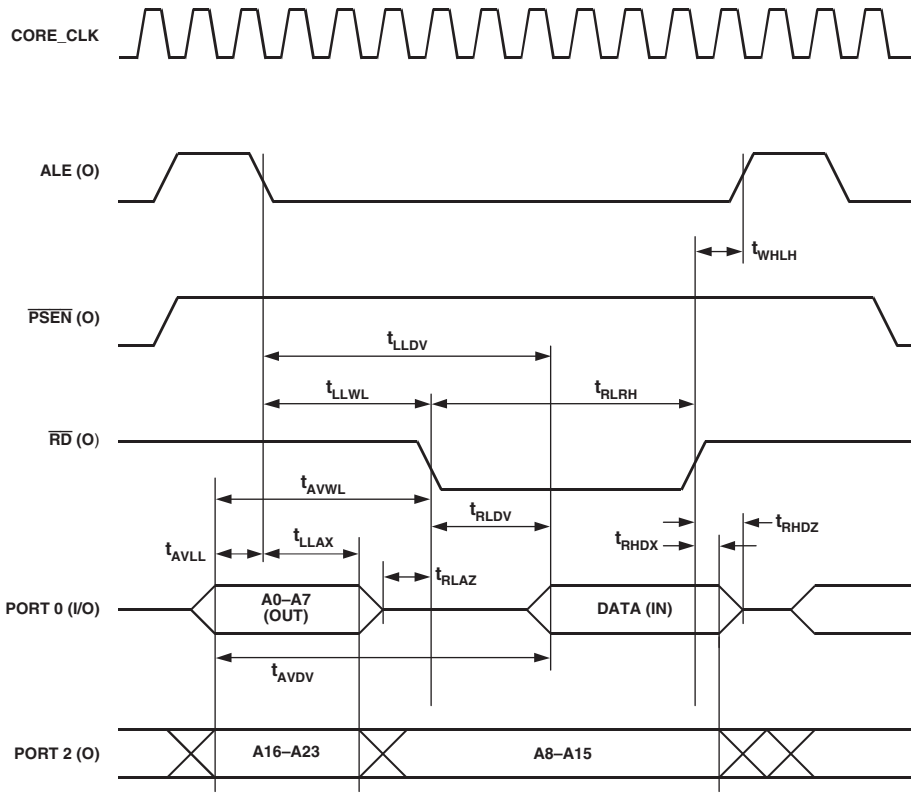


Figure 72. External Data Memory Read Cycle

Parameter		12.58 MHz Core_Clk Min	Max	Variable Core_Clk Min	Max	Unit	Figure
EXTERNAL DATA MEMORY WRITE CYCLE							
t_{WLWH}	$\overline{WR}$ Pulsewidth	377		$6t_{CORE} - 100$		ns	73
t_{AVLL}	Address Valid after ALE Low	39		$t_{CORE} - 40$		ns	73
t_{LLAX}	Address Hold after ALE Low	44		$t_{CORE} - 35$		ns	73
t_{LLWL}	ALE Low to $\overline{WR}$ Low	188	288	$3t_{CORE} - 50$	$3t_{CORE} + 50$	ns	73
t_{AVWL}	Address Valid to $\overline{WR}$ Low	188		$4t_{CORE} - 130$		ns	73
t_{QVWX}	Data Valid to $\overline{WR}$ Transition	29		$t_{CORE} - 50$		ns	73
t_{QVWH}	Data Setup before $\overline{WR}$	406		$7t_{CORE} - 150$		ns	73
t_{WHQX}	Data and Address Hold after $\overline{WR}$	29		$t_{CORE} - 50$		ns	73
t_{WHLH}	$\overline{WR}$ High to ALE High	39	119	$t_{CORE} - 40$	$t_{CORE} + 40$	ns	73

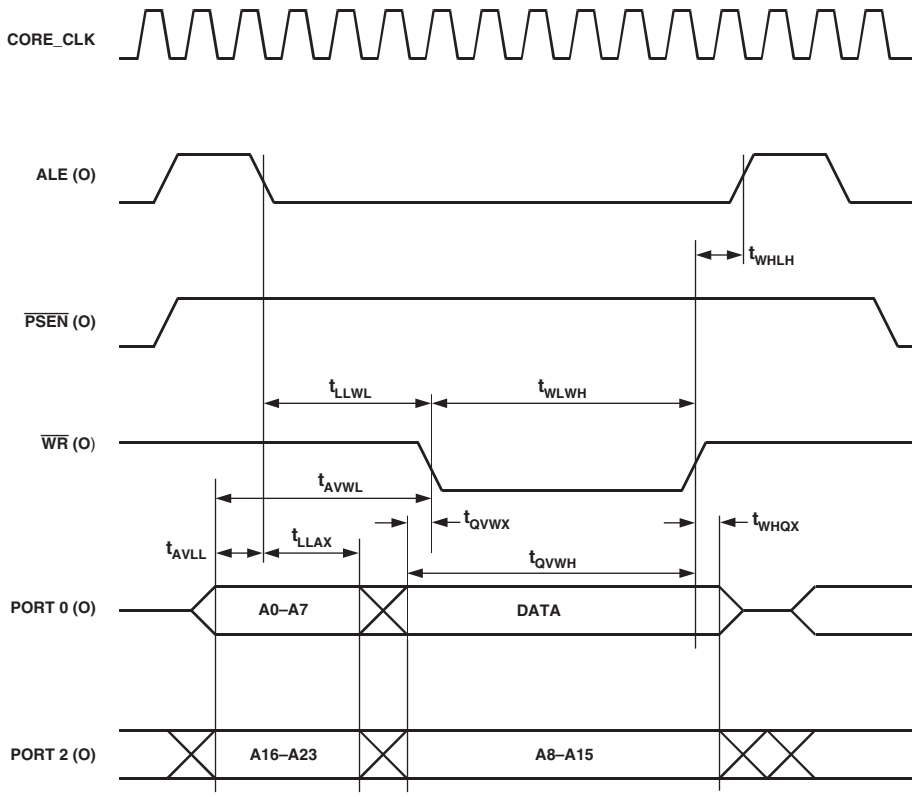


Figure 73. External Data Memory Write Cycle

Parameter		12.58 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		0.95			$12t_{CORE}$		μs	74
t_{QVXH}	Output Data Setup to Clock	662			$10t_{CORE} - 133$			ns	74
t_{DVXH}	Input Data Setup to Clock	292			$2t_{CORE} + 133$			ns	74
t_{XHDX}	Input Data Hold after Clock	0			0			ns	74
t_{XHQX}	Output Data Hold after Clock	42			$2t_{CORE} - 117$			ns	74

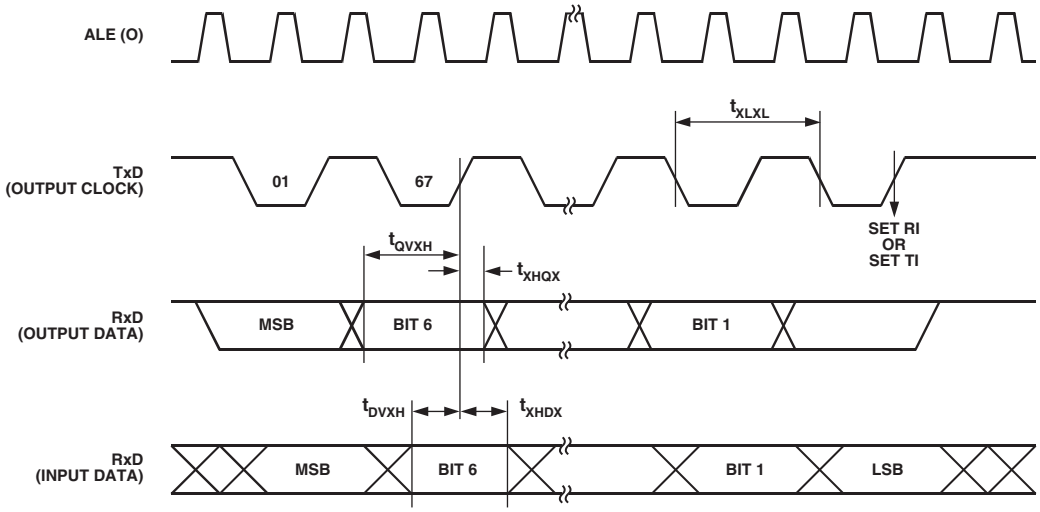


Figure 74. UART Timing in Shift Register Mode

Parameter		Min	Typ	Max	Unit	Figure
SPI MASTER MODE TIMING (CPHA = 1)						
t_{SL}	SCLOCK Low Pulsewidth*		630		ns	75
t_{SH}	SCLOCK High Pulsewidth*		630		ns	75
t_{DAV}	Data Output Valid after SCLOCK Edge			50	ns	75
t_{DSU}	Data Input Setup Time before SCLOCK Edge	100			ns	75
t_{DHD}	Data Input Hold Time after SCLOCK Edge	100			ns	75
t_{DF}	Data Output Fall Time		10	25	ns	75
t_{DR}	Data Output Rise Time		10	25	ns	75
t_{SR}	SCLOCK Rise Time		10	25	ns	75
t_{SF}	SCLOCK Fall Time		10	25	ns	75

*Characterized under the following conditions:
Core clock divider bits CD2, CD1, and CD0 in PLLCON SFR set to 0, 1, and 1, respectively, i.e., core clock frequency = 1.57 MHz, and SPI bit-rate selection bits SPR1 and SPR0 in SPICON SFR set to 0 and 0, respectively.

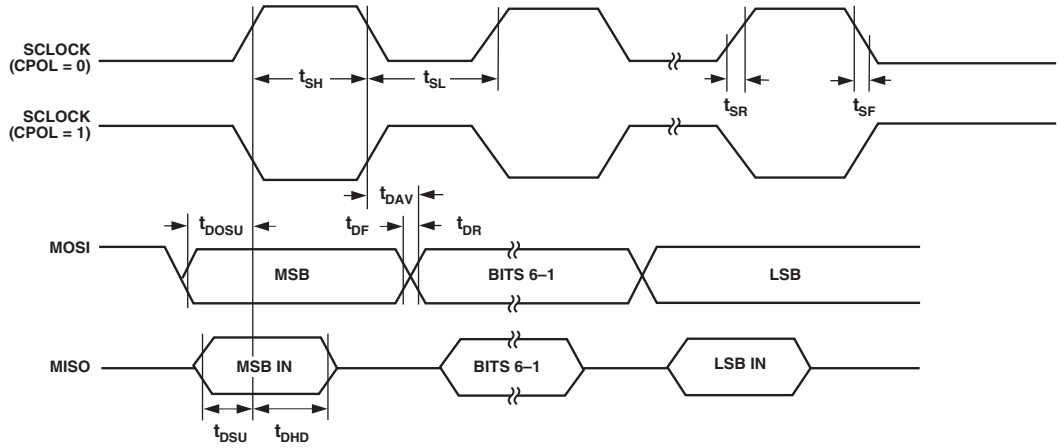


Figure 75. SPI Master Mode Timing (CPHA = 1)

Parameter		Min	Typ	Max	Unit	Figure
SPI MASTER MODE TIMING (CPHA = 0)						
t_{SL}	SCLOCK Low Pulsewidth*		630		ns	76
t_{SH}	SCLOCK High Pulsewidth*		630		ns	76
t_{DAV}	Data Output Valid after SCLOCK Edge			50	ns	76
t_{DOSU}	Data Output Setup before SCLOCK Edge			150	ns	76
t_{DSU}	Data Input Setup Time before SCLOCK Edge	100			ns	76
t_{DHD}	Data Input Hold Time after SCLOCK Edge	100			ns	76
t_{DF}	Data Output Fall Time		10	25	ns	76
t_{DR}	Data Output Rise Time		10	25	ns	76
t_{SR}	SCLOCK Rise Time		10	25	ns	76
t_{SF}	SCLOCK Fall Time		10	25	ns	76

*Characterized under the following conditions:
a. Core clock divider bits CD2, CD1 and CD0 in PLLCON SFR set to 0, 1, and 1 respectively, i.e., core clock frequency = 1.57 MHz, and
b. SPI bit-rate selection bits SPR1 and SPR0 in SPICON SFR set to 0 and 0 respectively.

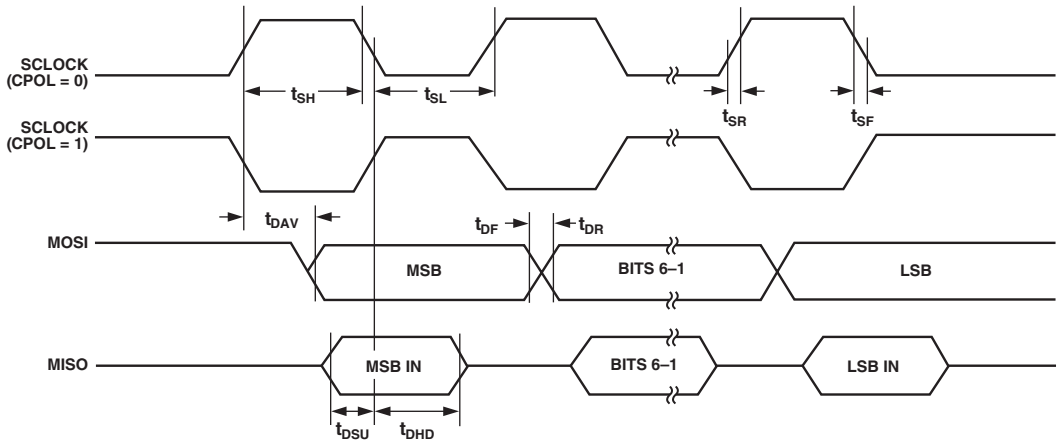


Figure 76. SPI Master Mode Timing (CPHA = 0)

Parameter		Min	Typ	Max	Unit	Figure
SPI SLAVE MODE TIMING (CPHA = 1)						
t_{SS}	$\overline{SS}$ to SCLOCK Edge	0			ns	77
t_{SL}	SCLOCK Low Pulsewidth		330		ns	77
t_{SH}	SCLOCK High Pulsewidth		330		ns	77
t_{DAV}	Data Output Valid after SCLOCK Edge			50	ns	77
t_{DSU}	Data Input Setup Time before SCLOCK Edge	100			ns	77
t_{DHD}	Data Input Hold Time after SCLOCK Edge	100			ns	77
t_{DF}	Data Output Fall Time		10	25	ns	77
t_{DR}	Data Output Rise Time		10	25	ns	77
t_{SR}	SCLOCK Rise Time		10	25	ns	75
t_{SF}	SCLOCK Fall Time		10	25	ns	77
t_{SFS}	$\overline{SS}$ High after SCLOCK Edge	0			ns	77

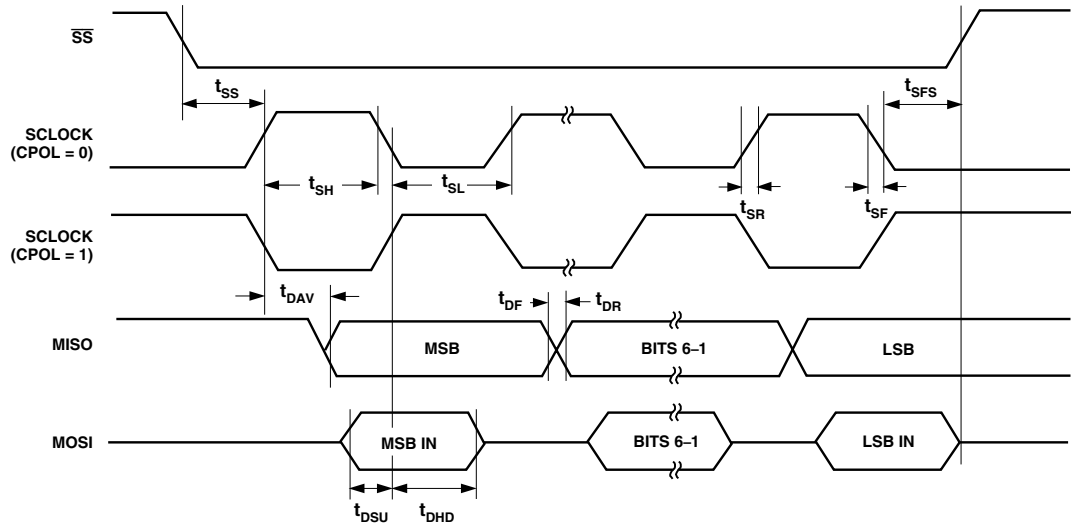


Figure 77. SPI Slave Mode Timing (CPHA = 1)

Parameter		Min	Typ	Max	Unit	Figure
SPI SLAVE MODE TIMING (CPHA = 0)						
t_{SS}	$\overline{SS}$ to SCLOCK Edge	0			ns	78
t_{SL}	SCLOCK Low Pulsewidth		330		ns	78
t_{SH}	SCLOCK High Pulsewidth		330		ns	78
t_{DAV}	Data Output Valid after SCLOCK Edge			50	ns	78
t_{DSU}	Data Input Setup Time before SCLOCK Edge	100			ns	78
t_{DHD}	Data Input Hold Time after SCLOCK Edge	100			ns	78
t_{DF}	Data Output Fall Time		10	25	ns	78
t_{DR}	Data Output Rise Time		10	25	ns	78
t_{SR}	SCLOCK Rise Time		10	25	ns	78
t_{SF}	SCLOCK Fall Time		10	25	ns	78
t_{SSR}	$\overline{SS}$ to SCLOCK Edge			50	ns	78
t_{DOSS}	Data Output Valid after $\overline{SS}$ Edge			20	ns	78
t_{SFS}	$\overline{SS}$ High after SCLOCK Edge	0			ns	78

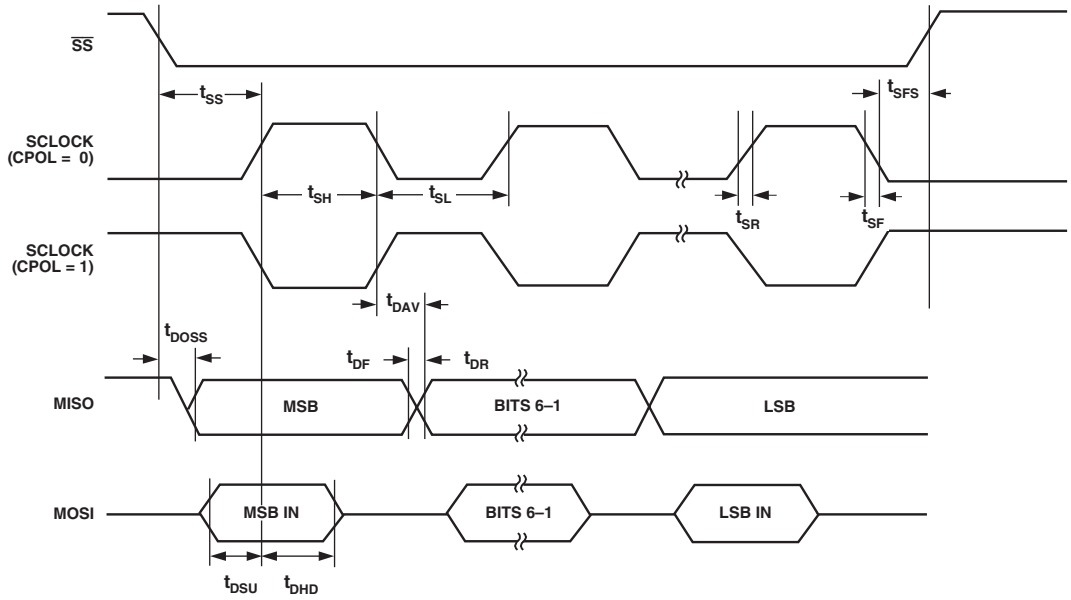


Figure 78. SPI Slave Mode Timing (CPHA = 0)

Parameter		Min	Max	Unit	Figure
I^2C-SERIAL INTERFACE TIMING					
t_L	SCLOCK Low Pulsewidth	4.7		μs	79
t_H	SCLOCK High Pulsewidth	4.0		μs	79
t_{SHD}	Start Condition Hold Time	0.6		μs	79
t_{DSU}	Data Setup Time	100		ns	79
t_{DHD}	Data Hold Time		0.9	μs	79
t_{RSU}	Setup Time for Repeated Start	0.6		μs	79
t_{PSU}	Stop Condition Setup Time	0.6		μs	79
t_{BUF}	Bus Free Time between a STOP Condition and a START Condition	1.3		μs	79
t_R	Rise Time of Both SCLOCK and SDATA		300	ns	79
t_F	Fall Time of Both SCLOCK and SDATA		300	ns	79
t_{SUP}*	Pulsewidth of Spike Suppressed		50	ns	79

*Input filtering on both the SCLOCK and SDATA inputs surpresses noise spikes less than 50 ns.

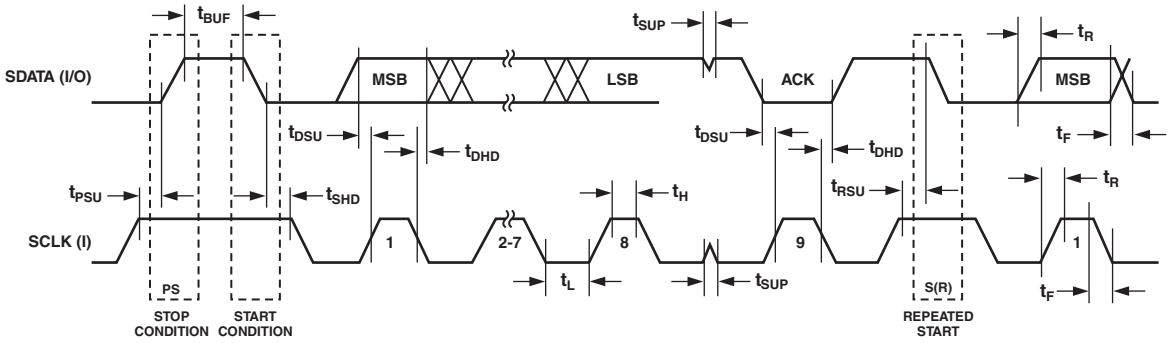
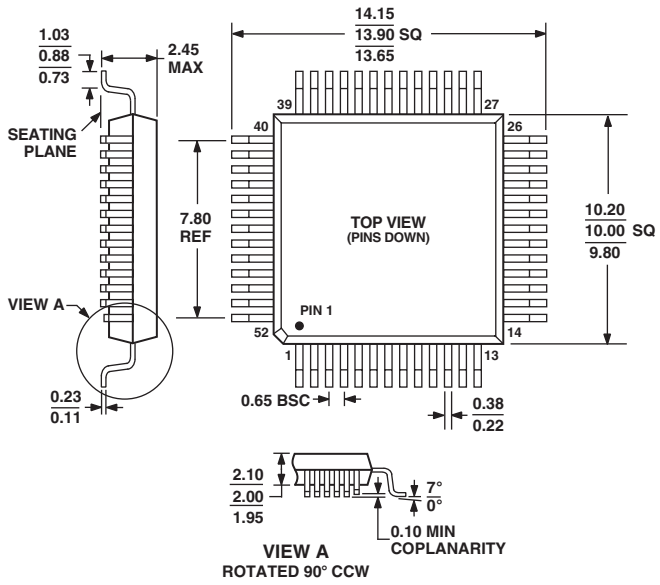


Figure 79. I^2C Compatible Interface Timing

OUTLINE DIMENSIONS

52-Lead Metric Quad Flat Package [MQFP]
(S-52)

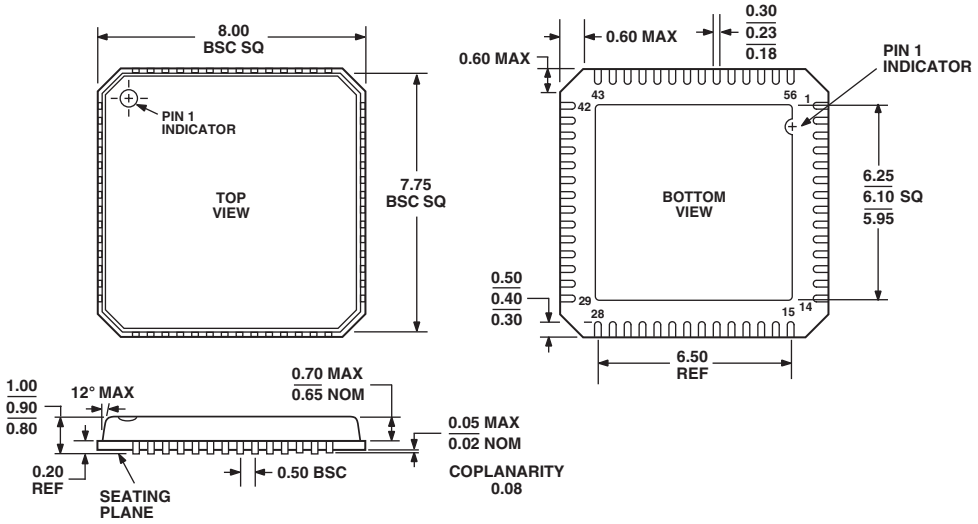
Dimensions shown in millimeters



COMPLIANT TO JEDEC STANDARDS MO-022-AC-1

56-Lead Lead Frame Chip Scale Package [LFCSP]
8 × 8 mm Body
(CP-56)

Dimensions shown in millimeters



COMPLIANT TO JEDEC STANDARDS MO-220-VLLD-2

C02942–0–4/03(A)

Revision History

Location	Page
4/03—Data Sheet changed from REV. 0 to REV. A.	
Updated OUTLINE DIMENSIONS	80