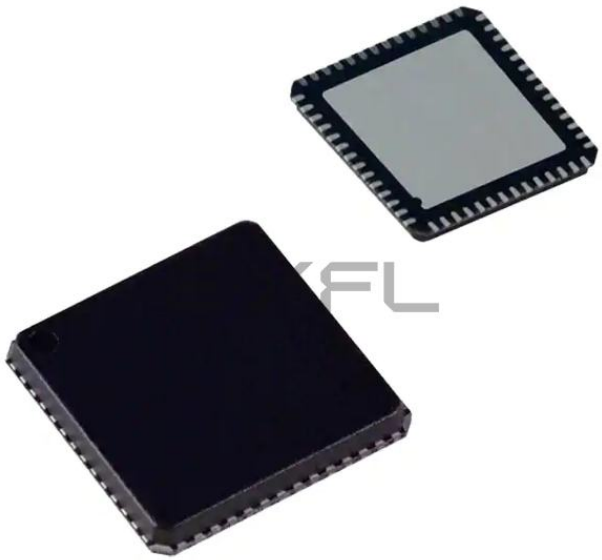

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	8052
Core Size	8-Bit
Speed	12.58MHz
Connectivity	EBI/EMI, I²C, SPI, UART/USART
Peripherals	POR, PSM, PWM, Temp Sensor, WDT
Number of I/O	34
Program Memory Size	62KB (62K x 8)
Program Memory Type	FLASH
EEPROM Size	4K x 8
RAM Size	2.25K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.25V
Data Converters	A/D 3x16b, 4x24b; D/A 1x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	56-VFQFN Exposed Pad, CSP
Supplier Device Package	56-LFCSP-VQ (8x8)
Purchase URL	https://www.e-xfl.com/product-detail/analog-devices/aduc834bcpz-reel

TABLE OF CONTENTS

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

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.

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.

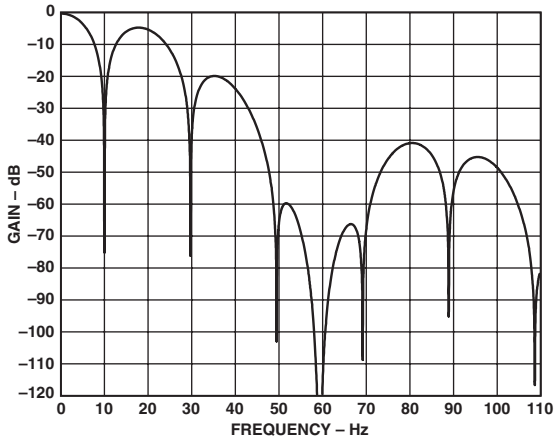

Figure 11. Filter Response, SF = 69 dec

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.

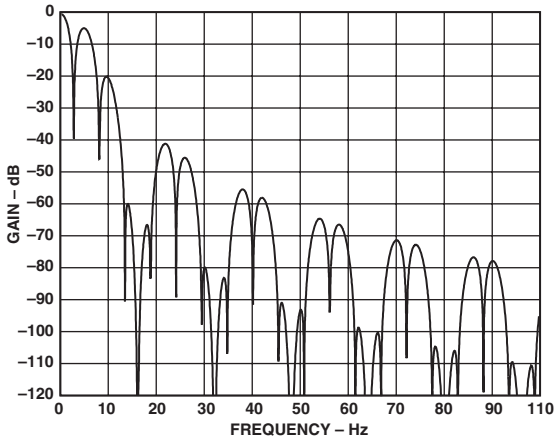

Figure 12. Filter Response, SF = 255 dec

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.

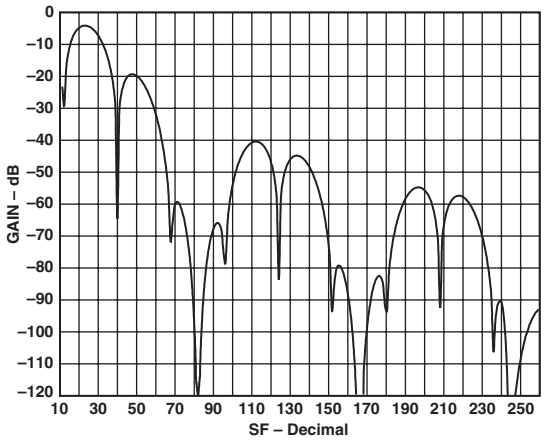

Figure 13. 50 Hz Normal Mode Rejection vs. SF

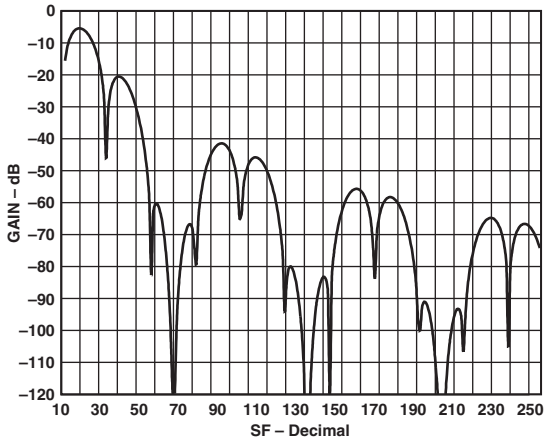

Figure 14. 60 Hz Normal Mode Rejection vs. SF

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.

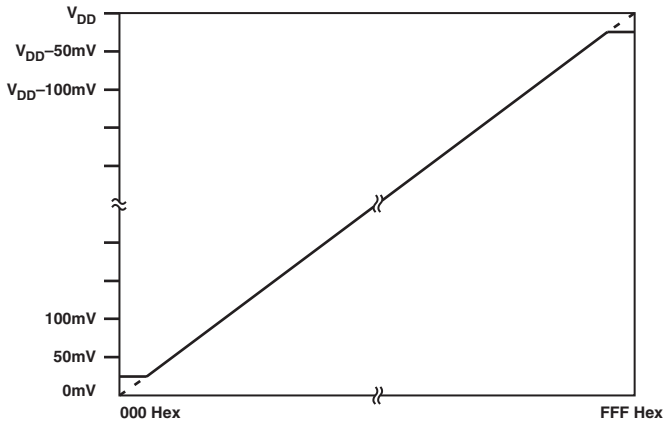

Figure 22. Endpoint Nonlinearities Due to Amplifier Saturation

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}.

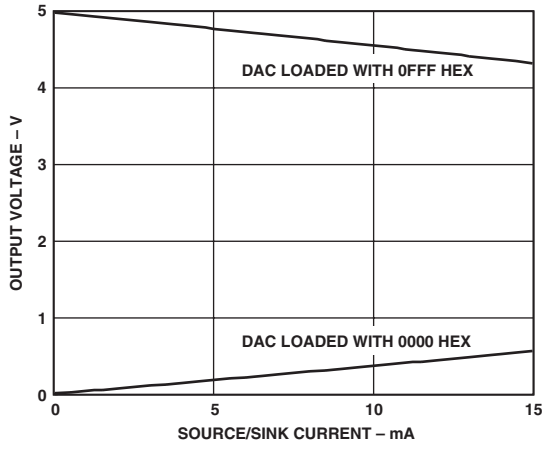

Figure 23. Source and Sink Current Capability with $V_{REF} = AV_{DD} = 5$ V

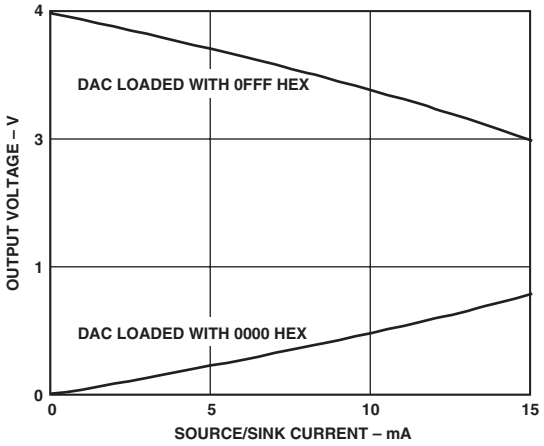

Figure 24. Source and Sink Current Capability with $V_{REF} = V_{DD} = 3$ V

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.

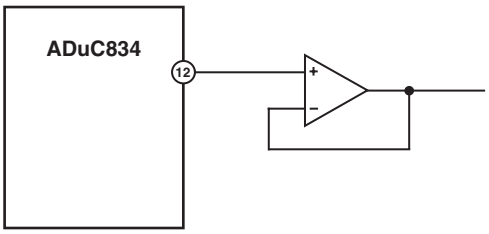

Figure 25. Buffering the DAC Output

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.

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.

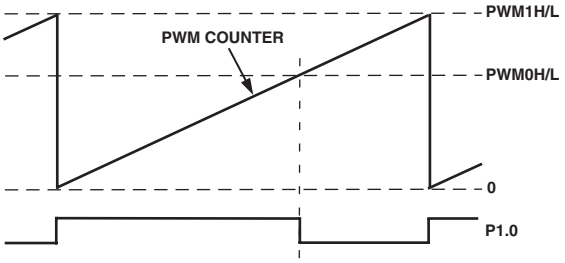

Figure 27. PWM in Mode 1

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.

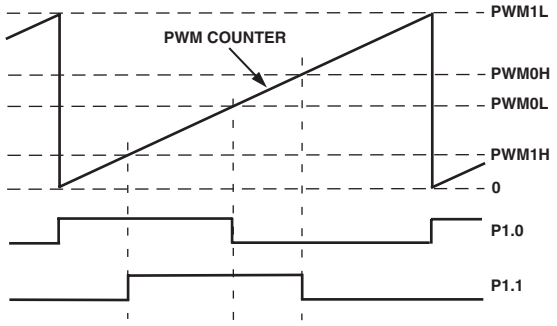

Figure 28. PWM Mode 2

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).

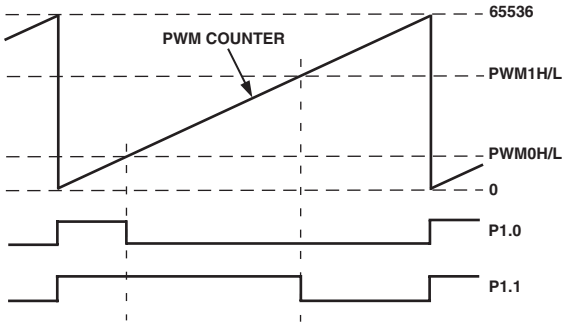

Figure 29. PWM Mode 3

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

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 timesout, 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.

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.

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.

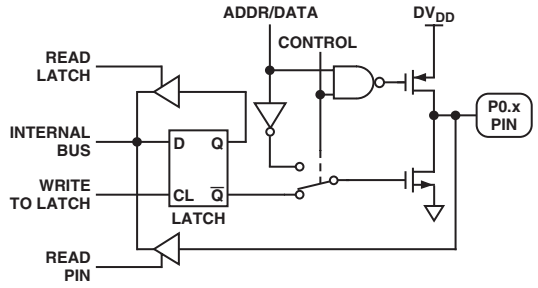

Figure 36. Port 0 Bit Latch and I/O Buffer

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.

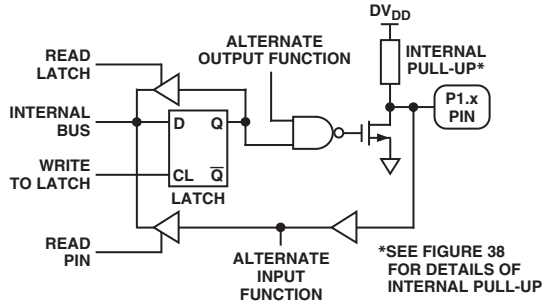

Figure 37. P1.0 and P1.1 Bit Latch and I/O Buffer

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.

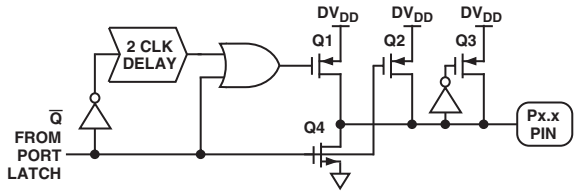

Figure 38. Internal Pull-Up Configuration

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.

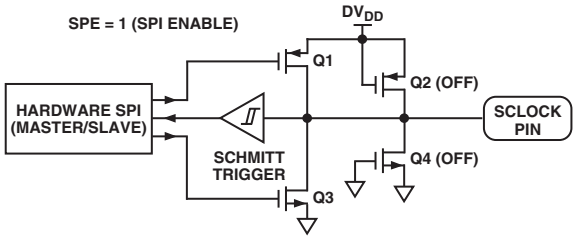

Figure 42. SCLOCK Pin I/O Functional Equivalent in SPI Mode

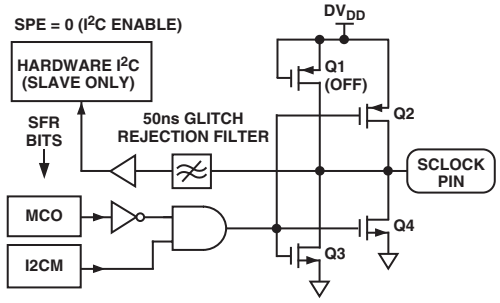

Figure 43. SCLOCK Pin I/O Functional Equivalent in I²C Mode

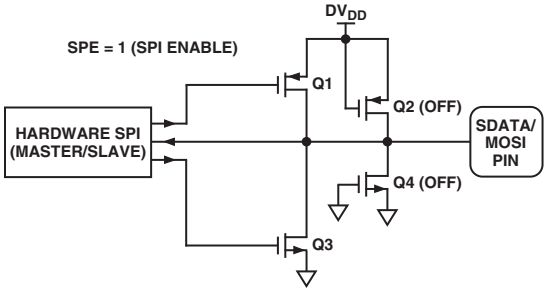

Figure 44. SDATA/MOSI Pin I/O Functional Equivalent in SPI Mode

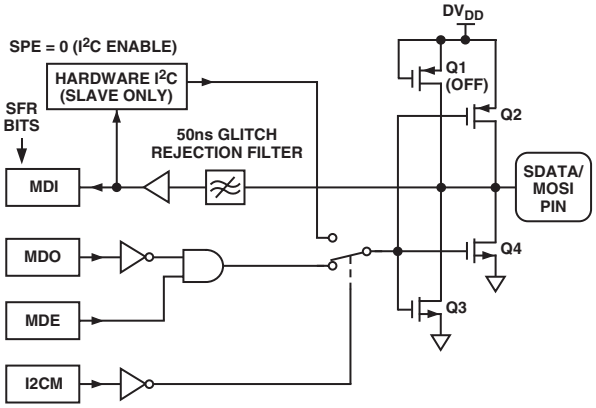

Figure 45. SDATA/MOSI Pin I/O Functional Equivalent in I²C Mode

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.

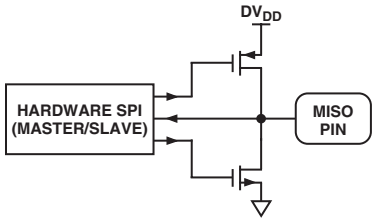

Figure 46. MISO Pin I/O Functional Equivalent

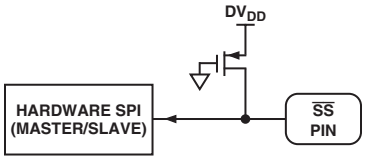

Figure 47. $\overline{SS}$ Pin I/O Functional Equivalent

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.

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

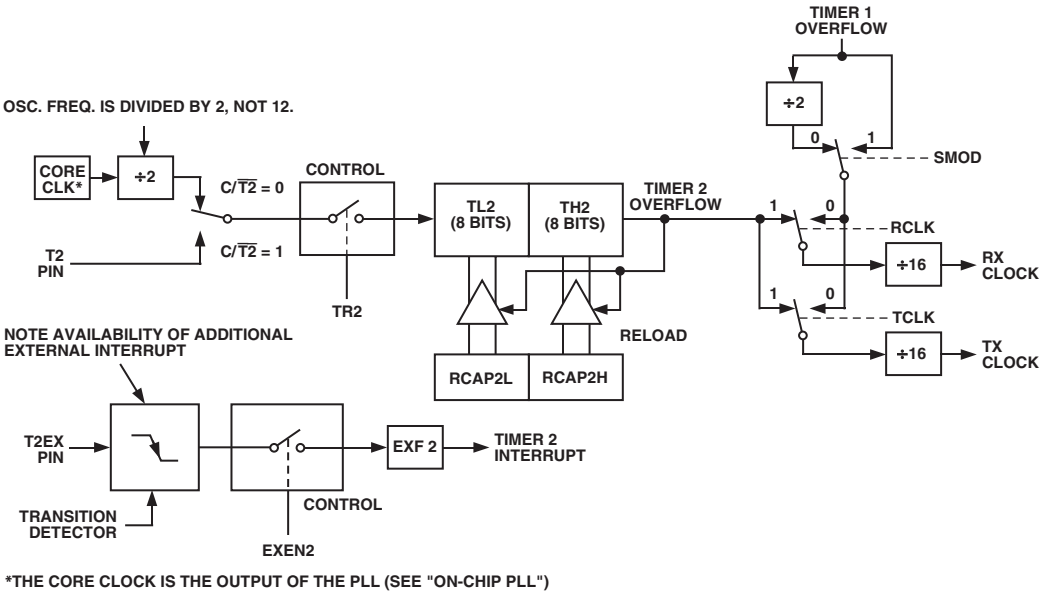

Figure 56. Timer 2, UART Baud Rates

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.

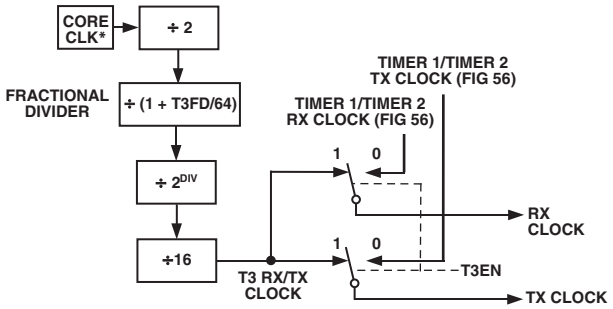

Figure 57. Timer 3, UART Baud Rates

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

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.

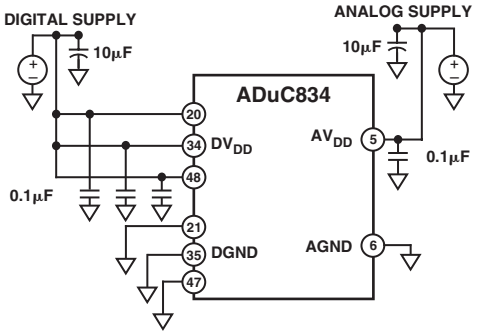

Figure 61. External Dual Supply Connections

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.

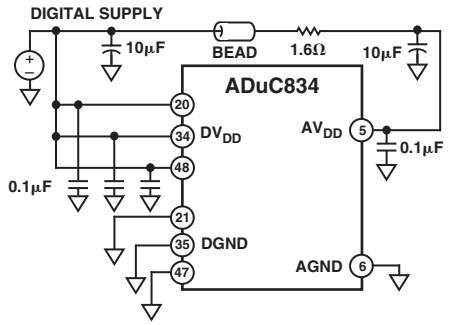

Figure 62. External Single Supply Connections

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.

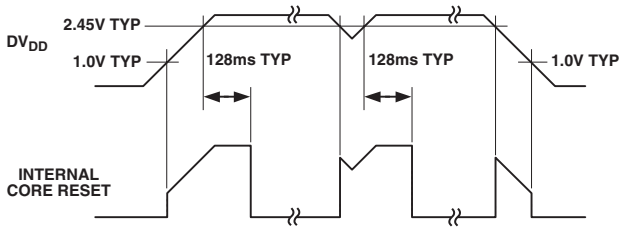

Figure 63. Internal Power-on-Reset Operation

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.

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).

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

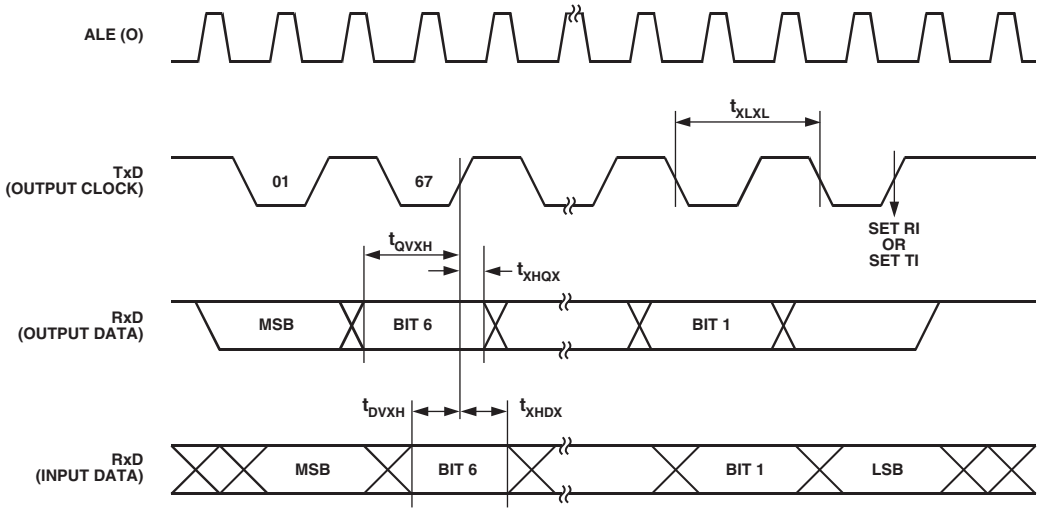

Figure 74. UART Timing in Shift Register Mode

Parameter		Min	Typ	Max	Unit	Figure
SPI 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

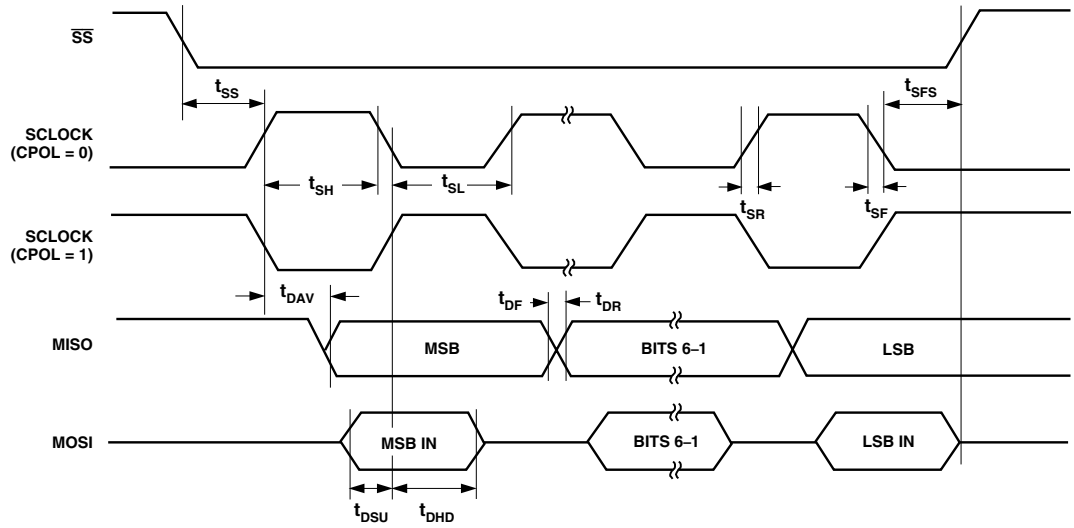

Figure 77. SPI Slave Mode Timing (CPHA = 1)