



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	Active
Core Processor	8052
Core Size	8-Bit
Speed	12.58MHz
Connectivity	I <sup>2</sup> C, 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 ~ 3.6V
Data Converters	A/D 10x24b; D/A 1x12b, 2x16b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	52-QFP
Supplier Device Package	52-MQFP (10x10)
Purchase URL	<a href="https://www.e-xfl.com/product-detail/analog-devices/aduc847bsz62-3">https://www.e-xfl.com/product-detail/analog-devices/aduc847bsz62-3</a>

Hardware Design Considerations .....	89	Other Hardware Considerations.....	92
External Memory Interface.....	89	QuickStart Development System .....	96
Power Supplies.....	89	QuickStart-PLUS Development System .....	96
Power-On Reset Operation.....	90	Timing Specifications .....	97
Power Consumption.....	90	Outline Dimensions.....	106
Power-Saving Modes .....	90	Ordering Guide .....	107
Grounding and Board Layout Recommendations .....	91		

## REVISION HISTORY

### 5/2016—Rev. C to Rev. D

Changed uC004 to AN-1074 .....	Throughout
Updated Outline Dimensions.....	108
Changes to Ordering Guide.....	109

### 12/2012—Rev. B to Rev. C

Changes to Figure 3 and Table 3 .....	11
Changes to Burnout Current Sources Section.....	32
Change to ADCMODE (ADC Mode Register) Section.....	42
Changes to Mode 4 (Dual NRZ 16-Bit $\Sigma$ - $\Delta$ DAC) Section .....	58
Change to Hardware Slave Mode Section .....	63
Updated Outline Dimensions.....	104
Changes to Ordering Guide.....	105

### 2/2005—Rev. A to Rev. B

Changes to Figure 1.....	1
Changes to the Burnout Current Sources Section .....	32
Changes to the Excitation Currents Section.....	36
Changes to Table 30 .....	47
Changes to the Flash/EE Memory on the ADuC845, ADuC847, ADuC848 Section .....	48
Changes to Figure 39 .....	57
Changes to On-Chip PLL (PLLCON) Section .....	60
Added 3 V Part Section Heading .....	88
Added 5 V Part Section .....	88
Changes to Figure 70 .....	91
Changes to Figure 71 .....	93

### 6/2004—Rev. 0 to Rev. A

Changes to Figure 5 .....	17
Changes to Figure 6 .....	18
Changes to Figure 7 .....	19
Changes to Table 5 .....	24
Changes to Table 24 .....	41
Changes to Table 25 .....	43
Changes to Table 26 .....	44
Changes to Table 27 .....	45
Changes to User Download Mode Section.....	50
Added Figure 51 and Renumbered Subsequent Figures.....	50
Edits to the DACH/DACL Data Registers Section.....	53
Changes to Table 34 .....	56
Added SPIDAT: SPI Data Register Section .....	65
Changes to Table 42 .....	67
Changes to Table 43 .....	68
Changes to Table 44 .....	69
Changes to Table 45 .....	71
Changes to Table 50 .....	75
Changes to Timer/Counter 0 and 1 Data Registers Section.....	76
Changes to Table 54 .....	80
Added the SBUF—UART Serial Port Data Register Section .....	80
Addition to the Timer 3 Generated Baud Rates Section .....	83
Added Table 57 and Renumbered Subsequent Tables .....	84
Changes to Table 61 .....	86

### 4/2004—Revision 0: Initial Version

Pin No.		Mnemonic	Type <sup>1</sup>	Description
52-MQFP	56-LFCSP			
43 to 46, 49 to 52	46 to 49, 52 to 55	P0.0 to P0.7	I/O	These pins are part of Port 0, 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 is required on P0 outputs to force a valid logic high level externally. Port 0 is also the multiplexed low-order address and data bus during accesses to external data memory. In this application, Port 0 uses strong internal pull-ups when emitting 1s.
	EP	EPAD		Exposed Pad. For the LFCSP, the exposed paddle must be left unconnected.

<sup>1</sup> I = input, S = supply, I/O means input/output, and O = output.

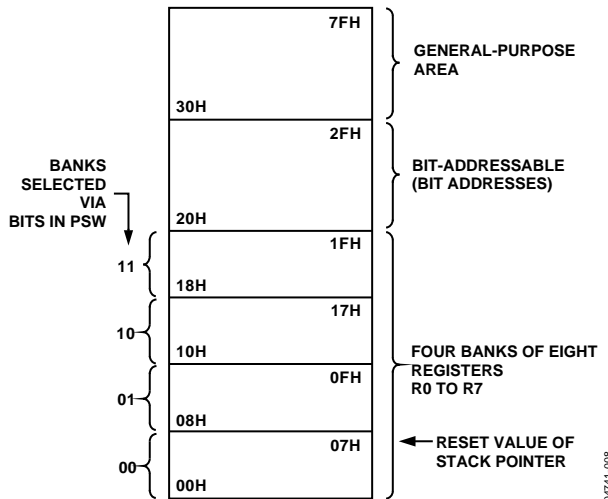


Figure 8. Lower 128 Bytes of Internal Data Memory

### Internal XRAM

The ADuC845, ADuC847, and ADuC848 contain 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 CFG84x.0 (Table 7) bit is set; otherwise, access to the external data memory occurs just like a standard 8051.

Even with the CFG84x.0 bit set, access to the external (off chip), XRAM occurs once the 24-bit DPTR is greater than 0007FFH.

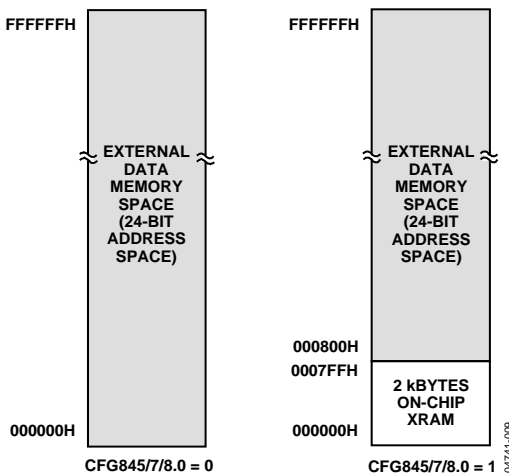


Figure 9. Internal and External XRAM

When enabled and when accessing the internal XRAM, the P0 and P2 port pin operations, as well as the  $\overline{RD}$  and  $\overline{WR}$  strobes, do not operate as a standard 8051 MOVX instruction. This allows the user to use these port pins as standard I/O. The internal XRAM can be configured as part of the extended 11-bit stack pointer. By default, the stack operates exactly like an 8052 in that it rolls over from FFH to 00H in the general-purpose RAM. On the ADuC845, ADuC847, and ADuC848, however, it

is possible (by setting CFG845.7/ADuC847.7/ADuC848.7) to enable the 11-bit extended stack pointer. In this case, the stack rolls over from FFH in RAM to 0100H in XRAM.

The 11-bit stack pointer is visible in the SPH and SP SFRs. The SP SFR is located at 81H as with a standard 8052. The SPH SFR is located at B7H. The 3 LSBs of the SPH SFR contain the 3 extra bits necessary to extend the 8-bit stack pointer in the SP SFR into an 11-bit stack pointer.

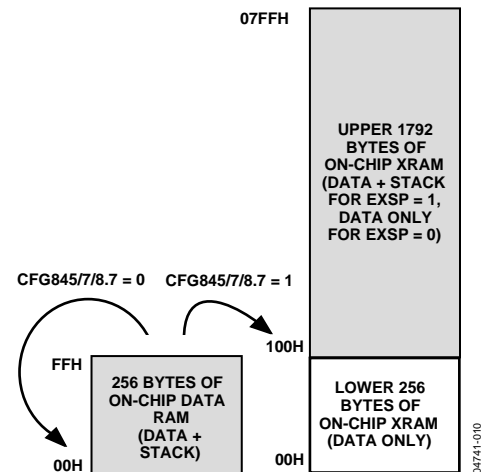


Figure 10. Extended Stack Pointer Operation

### External Data Memory (External XRAM)

There is no support for external program memory access to the devices. However, just like a standard 8051-compatible core, the ADuC845/ADuC847/ADuC848 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 devices, however, can access up to 16 Mbytes of external data memory. This is an enhancement of the 64 kbytes of external data memory space available on a standard 8051-compatible core. See the Hardware Design Considerations section for details.

When accessing external RAM, the EWAIT register might need to be programmed to give extra machine cycles to the MOVX operation. This is to account for differing external RAM access speeds.

### EWAIT SFR

SFR Address: 9FH  
Power-On Default: 00H  
Bit Addressable: No

This special function register (SFR), when programmed, dictates the number of wait states for the MOVX instruction. The value can vary between 0H and 7H. The MOVX instruction increases by one machine cycle ( $4 + n$ , where  $n$  = EWAIT number in decimal) for every increase in the EWAIT value.

**ADC Noise Performance with Chop Enabled ( $\overline{CHOP} = 0$ )**

Table 10, Table 11, Table 12, and Table 13 show the output rms noise and output peak-to-peak resolution in bits (rounded to the nearest 0.5 LSB) for some typical output update rates for the [ADuC845](#), [ADuC847](#), and [ADuC848](#). The numbers are typical and are generated at a differential input voltage of 0 V and a common-mode voltage of 2.5 V. The output update rate is selected via the SF7 to SF0 bits in the SF filter register. It is important to note that the peak-to-peak resolution figures represent the resolution for which there is no code flicker within a 6-sigma limit.

The output noise comes from two sources. The first source is the electrical noise in the semiconductor devices (device noise) used in the implementation of the modulator. The second source is quantization noise, which is added when the analog input is converted to the digital domain. The device noise is at a low level and is independent of frequency. The quantization noise starts at an even lower level but rises rapidly with increasing frequency to become the dominant noise source.

The numbers in the tables are given for the bipolar input ranges. For the unipolar ranges, the rms noise numbers are in the same range as the bipolar figures, but the peak-to-peak resolution is based on half the signal range, which effectively means losing 1 bit of resolution.

**Table 10. [ADuC845](#) and [ADuC847](#) Typical Output RMS Noise ( $\mu\text{V}$ ) vs. Input Range and Update Rate with Chop Enabled**

SF Word	Data Update Rate (Hz)	Input Range							
		$\pm 20\text{ mV}$	$\pm 40\text{ mV}$	$\pm 80\text{ mV}$	$\pm 160\text{ mV}$	$\pm 320\text{ mV}$	$\pm 640\text{ mV}$	$\pm 1.28\text{ V}$	$\pm 2.56\text{ V}$
13	105.03	1.75	1.30	1.65	1.5	2.1	3.1	7.15	13.3
23	59.36	1.25	0.95	1.08	0.94	1.0	1.87	3.24	7.1
27	50.56	1.0	1.0	0.85	0.85	1.13	1.56	2.9	3.6
69	19.79	0.63	0.68	0.52	0.7	0.61	1.1	1.3	2.75
255	5.35	0.31	0.38	0.34	0.32	0.4	0.45	0.68	1.22

**Table 11. [ADuC845](#) and [ADuC847](#) Typical Peak-to-Peak Resolution (Bits) vs. Input Range and Update Rate with Chop Enabled**

SF Word	Data Update Rate (Hz)	Input Range							
		$\pm 20\text{ mV}$	$\pm 40\text{ mV}$	$\pm 80\text{ mV}$	$\pm 160\text{ mV}$	$\pm 320\text{ mV}$	$\pm 640\text{ mV}$	$\pm 1.28\text{ V}$	$\pm 2.56\text{ V}$
13	105.03	12	13	14	15	15.5	16	16	16
23	59.36	12	13.5	14.5	15.5	16.5	16.5	17	16.5
27	50.56	12.5	13.5	15	16	16.5	17	17	17.5
69	19.79	13	14	15.5	16	17.5	17.5	18	18
255	5.35	14.5	15	16	17	18	18.5	19	19.5

**Table 12. [ADuC848](#) Typical Output Noise ( $\mu\text{V}$ ) vs. Input Range and Update Rate with Chop Enabled**

SF Word	Data Update Rate (Hz)	Input Range							
		$\pm 20\text{ mV}$	$\pm 40\text{ mV}$	$\pm 80\text{ mV}$	$\pm 160\text{ mV}$	$\pm 320\text{ mV}$	$\pm 640\text{ mV}$	$\pm 1.28\text{ V}$	$\pm 2.56\text{ V}$
13	105.03	1.75	1.30	1.65	1.5	2.1	3.1	7.15	13.3
23	59.36	1.25	0.95	1.08	0.94	1.0	1.87	3.24	7.1
27	50.56	1.0	1.0	0.85	0.85	1.13	1.56	2.9	3.6
69	19.79	0.63	0.68	0.52	0.7	0.61	1.1	1.3	2.75
255	5.35	0.31	0.38	0.34	0.32	0.4	0.45	0.68	1.22

**Table 13. [ADuC848](#) Typical Peak-to-Peak Resolution (Bits) vs. Input Range and Update Rate with Chop Enabled**

SF Word	Data Update Rate (Hz)	Input Range							
		$\pm 20\text{ mV}$	$\pm 40\text{ mV}$	$\pm 80\text{ mV}$	$\pm 160\text{ mV}$	$\pm 320\text{ mV}$	$\pm 640\text{ mV}$	$\pm 1.28\text{ V}$	$\pm 2.56\text{ V}$
13	105.03	12	13	14	15	15.5	16	16	16
23	59.36	12	13.5	14.5	15.5	16	16	17	16
27	50.56	12.5	13.5	15	16	16	16	16	16
69	19.79	13	14	15.5	16	16	16	16	16
255	5.35	14.5	15	16	16	16	16	16	16

**SF (ADC SINC FILTER CONTROL REGISTER)**

The SF register is used to configure the decimation factor for the ADC, and therefore, has a direct influence on the ADC throughput rate.

SFR Address: D4H  
 Power-On Default: 45H  
 Bit Addressable: No

**Table 28. Sinc Filter SFR Bit Designations**

SF.7	SF.6	SF.5	SF.4	SF.3	SF.2	SF.1	SF.0
0	1	0	0	0	1	0	1

The bits in this register set the decimation factor of the ADC. This has a direct bearing on the throughput rate of the ADC along with the chop setting. The equations used to determine the ADC throughput rate are

$$F_{adc} (\text{Chop On}) = \frac{1}{3 \times 8 \times SF_{word}} \times 32.768 \text{ kHz}$$

where  $SF_{word}$  is in decimal.

$$F_{adc} (\text{Chop Off}) = \frac{1}{8 \times SF_{word}} \times 32.768 \text{ kHz}$$

where  $SF_{word}$  is in decimal.

**Table 29. SF SFR Bit Examples**  
**Chop Enabled (ADCMODE.3 = 0)**

SF (Decimal)	SF (Hexadecimal)	F <sub>adc</sub> (Hz)	T <sub>adc</sub> (ms)	T <sub>settle</sub> (ms)
13 <sup>1</sup>	0D	105.3	9.52	19.04
69	45	19.79	50.53	101.1
82	52	16.65	60.06	120.1
255	FF	5.35	186.77	373.54

**Chop Disabled (ADCMODE.3 = 1)**

SF (Decimal)	SF (Hexadecimal)	F <sub>adc</sub> (Hz)	T <sub>adc</sub> (ms)	T <sub>settle</sub> (ms)
3	03	1365.3	0.73	2.2
69	45	59.36	16.84	50.52
82	52	49.95	20.02	60.06
255	FF	16.06	62.25	186.8

<sup>1</sup> With chop enabled, if an SF word smaller than 13 is written to this SF register, the filter automatically defaults to 13.

During ADC calibration, the user-programmed value of SF word is used. The SF word does not default to the maximum setting (255) as it did on previous MicroConverter® products. However, for optimum calibration results, it is recommended that the maximum SF word be set.

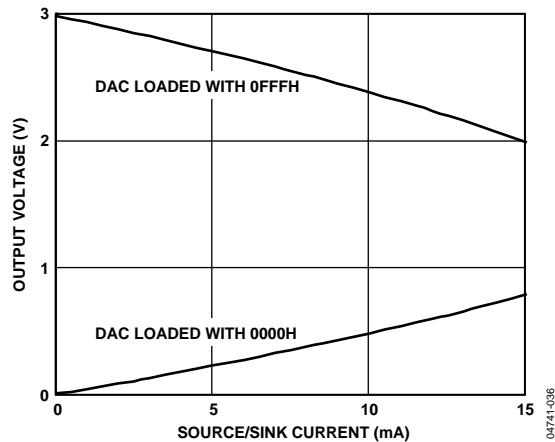


Figure 36. Source and Sink Current Capability with  $V_{REF} = AV_{DD} = 3\text{ V}$

For larger loads, the current drive capability may not be sufficient. To increase the source and sink current capability of the DAC, an external buffer should be added as shown in Figure 37.

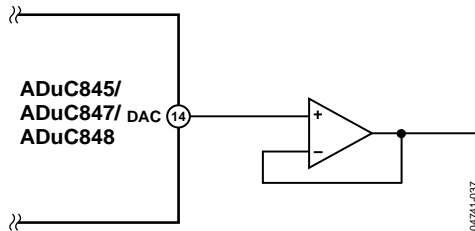


Figure 37. Buffering the DAC Output

The internal 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 it remains inactive until enabled in software. This means that if a zero output is desired during power-on or power-down transient conditions, a pull-down resistor must be added to each DAC output. Assuming that this resistor is in place, the DAC output remains at ground potential whenever the DAC is disabled.

## PULSE-WIDTH MODULATOR (PWM)

The ADuC845/ADuC847/ADuC848 has a highly flexible PWM offering programmable resolution and an input clock. The PWM can be configured in six different modes of operation. Two of these modes allow the PWM to be configured as a  $\Sigma$ - $\Delta$  DAC with up to 16 bits of resolution. A block diagram of the PWM is shown in Figure 38.

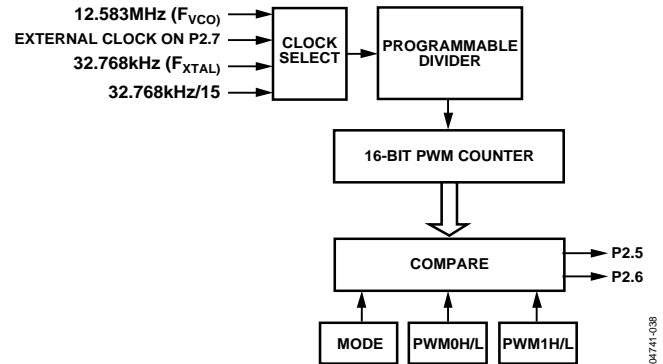


Figure 38. PWM Block Diagram

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

PWMCON (as described in Table 34) 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 P2.5 and P2.6.

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.

**Mode 5 (Dual 8-Bit PWM)**

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

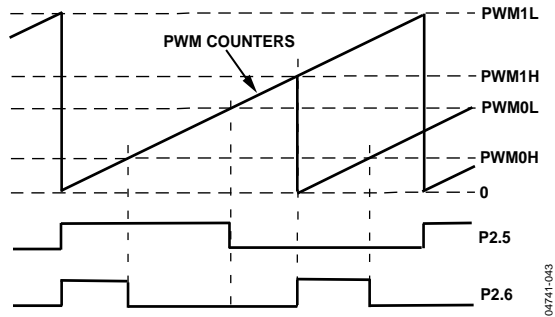


Figure 43. PWM Mode 5

**Mode 6 (Dual RZ 16-Bit  $\Sigma$ - $\Delta$  DAC)**

Mode 6 provides a high speed PWM output similar to that of a  $\Sigma$ - $\Delta$  DAC. Mode 6 operates very similarly to Mode 4; however, the key difference is that Mode 6 provides return to zero (RZ)  $\Sigma$ - $\Delta$  DAC output. Mode 4 provides non-return-to-zero  $\Sigma$ - $\Delta$  DAC outputs. RZ mode ensures that any difference in the rise and fall times does not affect the  $\Sigma$ - $\Delta$  DAC INL. However, RZ mode halves the dynamic range of the  $\Sigma$ - $\Delta$  DAC outputs from 0 V– to  $AV_{DD}$  down to 0 V to  $AV_{DD}/2$ . 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), typically P2.6 is low for three full clocks ( $3 \times 80$  ns), high for one-half a clock (40 ns), and then low again for one-half a clock (40 ns) before repeating itself. Over every 65536 clocks, the PWM compromises 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  $\Sigma$ - $\Delta$  DAC output can occur at 3 kHz. Similarly, writing 00000001 to the 8 LSBs gives an 8-bit accurate  $\Sigma$ - $\Delta$  DAC output at 49 kHz.

The output resolution is set by the PWM1L and PWM1H SFRs for the P2.5 and P2.6 outputs, respectively. PWM0L and PWM0H set the duty cycles of the PWM outputs at P2.5 and P2.6, respectively. Both PWMs have the same clock source and clock divider.

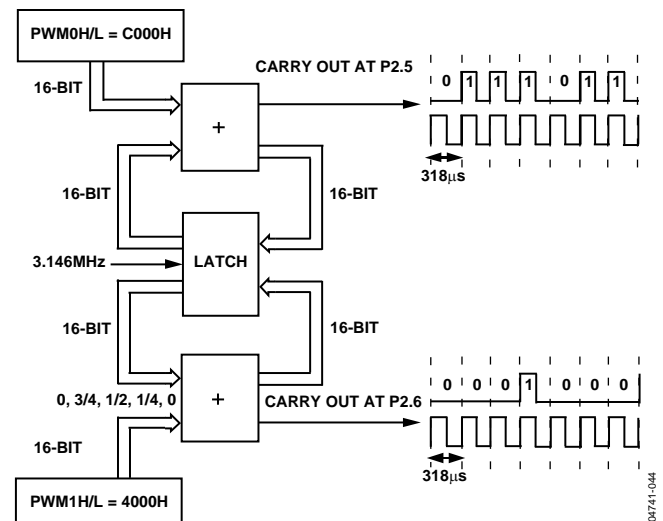


Figure 44. PWM Mode 6

**Mode 7**

In Mode 7, the PWM is disabled, allowing P2.5 and P2.6 to be used as normal.



## POWER SUPPLY MONITOR

The power supply monitor, once enabled, monitors the  $DV_{DD}$  and  $AV_{DD}$  supplies on the devices. It indicates when any of the supply pins drop below one of four user-selectable voltage trip points from 2.63 V to 4.63 V. For correct operation of the power supply monitor function,  $AV_{DD}$  must be equal to or greater than 2.63 V. Monitor function is controlled via the PSMCON SFR. If enabled via the IEIP2 SFR, the monitor interrupts the core by using the PSMI bit in the PSMCON SFR. This bit is not cleared until the failing power supply returns above the trip point for at least 250 ms.

The monitor function allows the user to save working registers to avoid possible data loss due to the low supply condition, and also ensures that normal code execution does not resume until a

safe supply level is well established. The supply monitor is also protected against spurious glitches triggering the interrupt circuit.

The 5 V part has an internal POR trip level of 4.63 V, which means that there are no usable  $DV_{DD}$  PSM trip levels on the 5 V part. The 3 V part has a POR trip level of 2.63 V following a reset and initialization sequence, allowing all relevant PSM trip points to be used.

### PSMCON—Power Supply Monitor Control Register

SFR Address: DFH  
Power-On Default: DEH  
Bit Addressable: No

**Table 43. PSMCON SFR Bit Designations**

Bit No.	Name	Description															
7	CMPD	$DV_{DD}$ Comparator Bit. This read-only bit directly reflects the state of the $DV_{DD}$ comparator. Read 1 indicates that the $DV_{DD}$ supply is above its selected trip point. Read 0 indicates that the $DV_{DD}$ supply is below its selected trip point.															
6	CMPA	$AV_{DD}$ Comparator Bit. This read-only bit directly reflects the state of the $AV_{DD}$ comparator. Read 1 indicates that the $AV_{DD}$ supply is above its selected trip point. Read 0 indicates that the $AV_{DD}$ supply is below its selected trip point.															
5	PSMI	Power Supply Monitor Interrupt Bit. Set high by the MicroConverter if either CMPA or CMPD is low, indicating low analog or digital supply. The PSMI bit can be used to interrupt the processor. Once CMPD and/or CMPA returns (and remains) high, a 250 ms counter is started. When this counter times out, the PSMI interrupt is cleared. PSMI can also be written by the user. However, if either comparator output is low, it is not possible for the user to clear PSMI.															
4, 3	TPD1, TPD0	$DV_{DD}$ Trip Point Selection Bits. A 5 V part has no valid PSM trip points. If the $DV_{DD}$ supply falls below the 4.63 V point, the device resets (POR). For a 3 V part, all relevant PSM trip points are valid. The 3 V POR trip point is 2.63 V (fixed). These bits select the $DV_{DD}$ trip point voltage as follows:															
		<table> <tr> <th>TPD1</th><th>TPD0</th><th>Selected <math>DV_{DD}</math> Trip Point (V)</th></tr> <tr> <td>0</td><td>0</td><td>4.63</td></tr> <tr> <td>0</td><td>1</td><td>3.08</td></tr> <tr> <td>1</td><td>0</td><td>2.93</td></tr> <tr> <td>1</td><td>1</td><td>2.63</td></tr> </table>	TPD1	TPD0	Selected $DV_{DD}$ Trip Point (V)	0	0	4.63	0	1	3.08	1	0	2.93	1	1	2.63
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, 1	TPA1, TPA0	$AV_{DD}$ Trip Point Selection Bits. These bits select the $AV_{DD}$ trip point voltage as follows:															
		<table> <tr> <th>TPA1</th><th>TPA0</th><th>Selected <math>AV_{DD}</math> Trip Point (V)</th></tr> <tr> <td>0</td><td>0</td><td>4.63</td></tr> <tr> <td>0</td><td>1</td><td>3.08</td></tr> <tr> <td>1</td><td>0</td><td>2.93</td></tr> <tr> <td>1</td><td>1</td><td>2.63</td></tr> </table>	TPA1	TPA0	Selected $AV_{DD}$ Trip Point (V)	0	0	4.63	0	1	3.08	1	0	2.93	1	1	2.63
TPA1	TPA0	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.															

## TIME INTERVAL COUNTER (TIC)

A TIC is provided on-chip for counting longer intervals than the standard 8051-compatible timers can count. The TIC is capable of timeout intervals ranging from 1/128 second to 255 hours. Also, this counter is clocked by the external 32.768 kHz crystal rather than by the core clock, and it can 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. Note that instructions to the TIC SFRs are also clocked at 32.768 kHz, so sufficient time must be allowed in user code for these instructions to execute.

Six SFRs are associated with the time interval counter, TIMECON being its control register. Depending on the configuration of the IT0 and IT1 bits in TIMECON, the selected time counter register overflow clocks the interval counter. When this counter is equal to the time interval value loaded in the INTVAL SFR, the TII bit (TIMECON.2) is set and generates an interrupt, if enabled. If the device is in power-down mode, again with TIC interrupt enabled, the TII bit wakes up the device and resumes code execution by vectoring directly to the TIC interrupt service vector address at 0053H. The TIC-related SFRs are described in Table 45. Note also that the time based SFRs can be written initially with the current time; the TIC can then be controlled and accessed by user software. In effect, this facilitates the implementation of a real-time clock. A basic block diagram of the TIC is shown in Figure 47.

Because the TIC is clocked directly from a 32 kHz external crystal on the devices, instructions that access the TIC registers are also clocked at 32 kHz (not at the core frequency). The user must ensure that sufficient time is given for these instructions to execute.

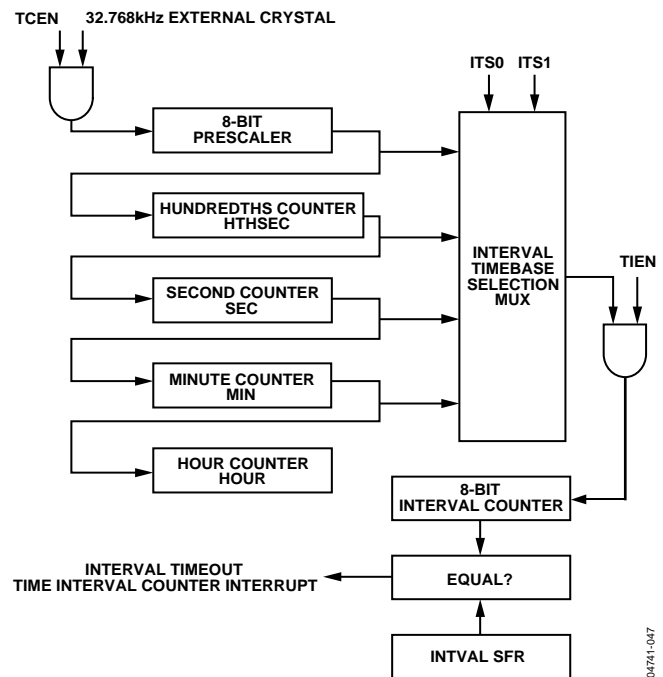


Figure 47. TIC Simplified Block Diagram

**INTVAL—User Timer Interval Select Register**

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

**HTHSEC—Hundredths of Seconds Time Register**

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

**SEC—Seconds Time Register**

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

**MIN—Minutes Time Register**

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

**HOUR—Hours Time Register**

Function:	This register is incremented in 1-hour intervals once TCEN in TIMECON is active. The HOUR SFR counts from 0 to 23 before rolling over to 0.
SFR Address:	A5H
Power-On Default:	00H
Bit Addressable:	No
Valid Value:	0 to 23 decimal

To enable the TIC as a real-time clock, the HOUR, MIN, SEC, and HTHSEC registers can be loaded with the current time. Once the TCEN bit is high, the TIC starts. To use the TIC as a time interval counter, select the count interval—hundredths of seconds, seconds, minutes, and hours via the ITS0 and ITS1 bits in the TIMECON SFR. Load the count required into the INTVAL SFR.

Note that INTVAL is only an 8-bit register, so user software must take into account any intervals longer than are possible with 8 bits. Therefore, to count an interval of 20 seconds, use the following procedure:

```
MOV TIMECON, #0D0H ;Enable 24Hour mode, count seconds, Clear TCEN.  
MOV INTVAL, #14H ;Load INTVAL with required count interval...in this case 14H = 20  
MOV TIMECON, #0D3H ;Start TIC counting and enable the 8bit INTVAL counter.
```

### Timer/Counter 0 and 1 Operating Modes

This section describes the operating modes for Timer/Counters 0 and 1. Unless otherwise noted, these modes of operation are the same for both Timer 0 and Timer 1.

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

Mode 0 configures an 8-bit timer/counter. Figure 52 shows Mode 0 operation. Note that the divide-by-12 prescaler is not present on the single-cycle core.

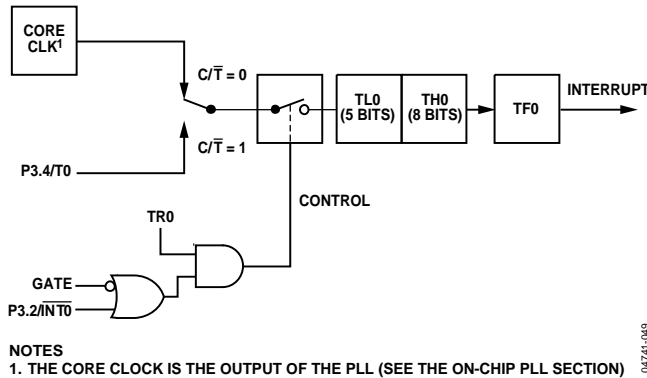


Figure 52. Timer/Counter 0, Mode 0

In this mode, the timer register is configured as a 13-bit register. As the count rolls over from all 1s to all 0s, it sets the timer overflow flag, TF0. TF0 can then be used to request an interrupt. The counted input is enabled to the timer when  $TR0 = 1$  and either  $Gate = 0$  or  $\overline{INT0} = 1$ . Setting  $Gate = 1$  allows the timer to be controlled by external input  $\overline{INT0}$  to facilitate pulse-width measurements.  $TR0$  is a control bit in the special function register TCON;  $Gate$  is in TMOD. The 13-bit register consists of all 8 bits of TH0 and the lower 5 bits of TL0. The upper 3 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 Mode 1 timer register runs with all 16 bits. Mode 1 is shown in Figure 53.

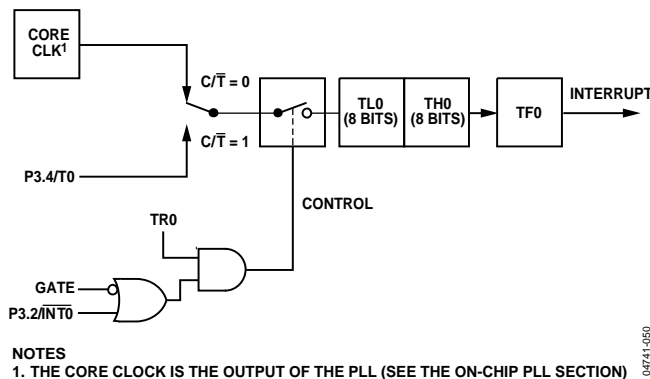


Figure 53. Timer/Counter 0, Mode 1

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

Mode 2 configures the timer register as an 8-bit counter (TL0) with automatic reload as shown in Figure 54. Overflow from TL0 not only sets TF0, but also reloads TL0 with the contents of TH0, which is preset by software. The reload leaves TH0 unchanged.

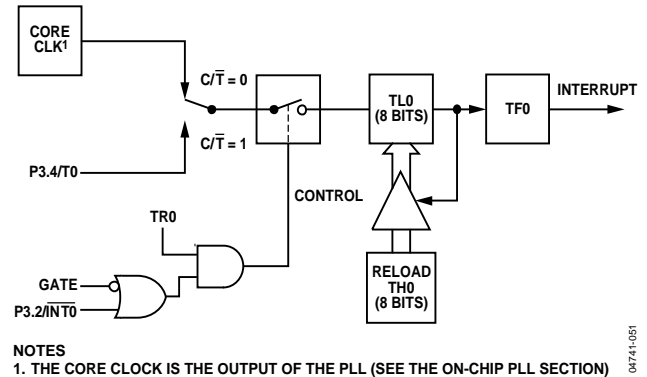


Figure 54. Timer/Counter 0, Mode 2

#### 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 55. 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. Therefore, TH0 then 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 it 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.

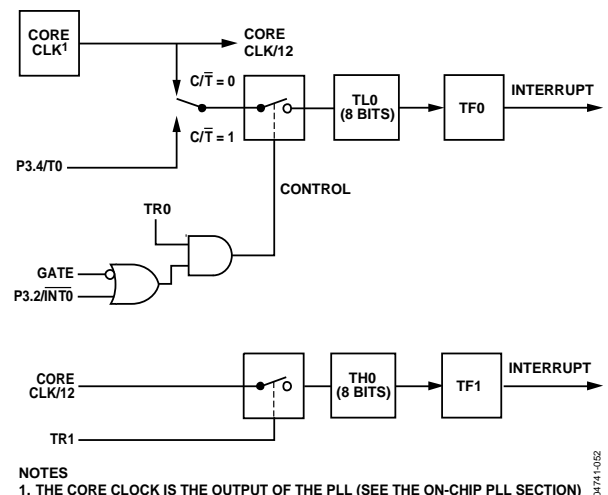


Figure 55. Timer/Counter 0, Mode 3

**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.

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 as shown in Figure 58.

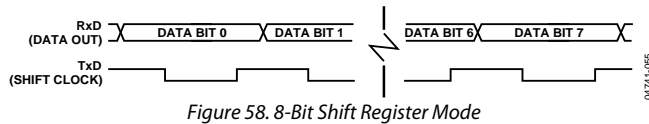


Figure 58. 8-Bit Shift Register Mode

**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 are received on RxD. The baud rate is set by the Timer 1 or Timer 2 overflow rate, or a combination of the two (one for transmission and the other for reception).

Transmission is initiated by writing to SBUF. The write to SBUF signal also loads a 1 (stop bit) into the 9th 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 59.

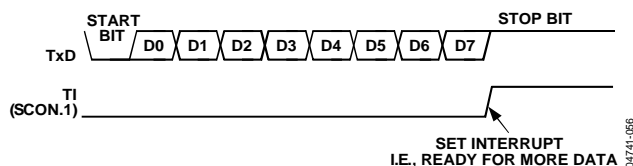


Figure 59. 8-Bit Variable Baud Rate

Reception is initiated when a 1-to-0 transition is detected on RxD. Assuming that a valid start bit is detected, character reception continues. The start bit is skipped and the 8 data bits are clocked into the serial port shift register. When all 8 bits have been clocked in, the following events occur:

- The 8 bits in the receive shift register are latched into SBUF.
- The 9th bit (stop bit) is clocked into RB8 in SCON.
- The receiver interrupt flag (RI) is set.

All of the following conditions must be met at the time the final shift pulse is generated:

- RI = 0
- Either SM2 = 0 or SM2 = 1
- Received stop bit = 1

If any 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), 8 data bits, a programmable 9th bit, and a stop bit (1). The 9th 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 8 data bits must be written into SBUF. The ninth bit must be written to TB8 in SCON. When transmission is initiated, the 8 data bits (from SBUF) are loaded into the transmit shift register (LSB first). The contents of TB8 are loaded into the 9th bit position of the transmit shift register. The transmission starts at the next valid baud rate clock. The TI flag is set as soon as the stop bit appears on TxD.

Reception for Mode 2 is similar to that of Mode 1. The 8 data bytes are input at RxD (LSB first) and loaded onto the receive shift register. When all 8 bits have been clocked in, the following events occur:

- The 8 bits in the receive shift register are latched into SBUF.
- The 9th data bit is latched into RB8 in SCON.
- The receiver interrupt flag (RI) is set.

All of the following conditions must be met at the time the final shift pulse is generated:

- RI = 0
- Either SM2 = 0 or SM2 = 1
- Received stop bit = 1

If any 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 when 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:

$$\text{Mode 0 Baud Rate} = \left( \frac{\text{Core Clock Frequency}}{12} \right)$$

#### 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 1/32 of the core clock. If SMOD = 1, the baud rate is 1/16 of the core clock:

$$\text{Mode 2 Baud Rate} = \frac{2^{\text{SMOD}}}{32} \times \text{Core Clock Frequency}$$

#### Modes 1 and 3 Baud Rate Generation

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

#### Timer 1 Generated Baud Rates

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

$$\text{Modes 1 and 3 Baud Rate} = \frac{2^{\text{SMOD}}}{32} \times \text{Timer 1 Overflow Rate}$$

The Timer 1 interrupt should be disabled in this application. The timer itself can be configured for either timer or counter operation, and in any of its three running modes. In the most typical application, it is configured for timer operation in autoreload mode (high nibble of TMOD = 0010 binary). In that case, the baud rate is given by the formula

$$\text{Modes 1 and 3 Baud Rate} = \frac{2^{\text{SMOD}}}{32} \times \frac{\text{Core Clock Frequency}}{(256 - \text{TH1})}$$

#### Timer 2 Generated Baud Rates

Baud rates can also be generated by 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 or received. Because Timer 2 has a 16-bit autoreload mode, a wider range of baud rates is possible.

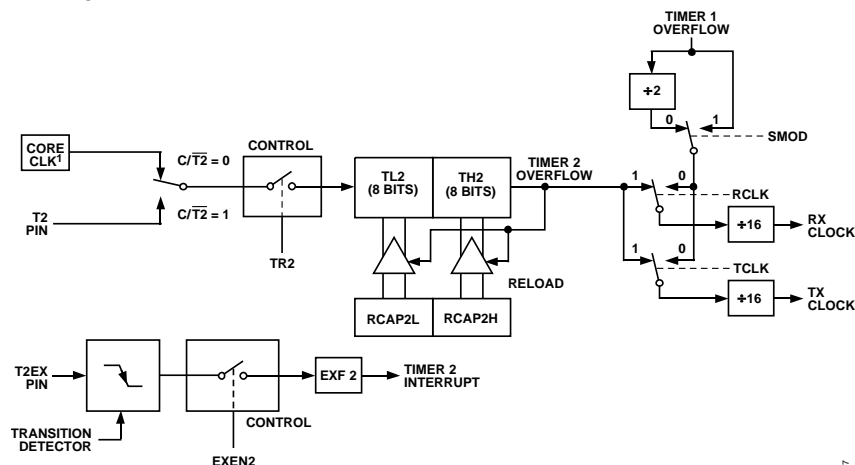
$$\text{Modes 1 and 3 Baud Rate} = \frac{1}{16} \times \text{Timer 2 Overflow Rate}$$

Therefore, when Timer 2 is used to generate baud rates, the timer increments every two clock cycles rather than every core machine cycle as before. It increments six times faster than Timer 1, and, therefore, baud rates six times faster are possible. Because Timer 2 has 16-bit autoreload capability, very low baud rates are still possible.

Timer 2 is selected as the baud rate generator by setting the 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 60.

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

$$\text{Modes 1 and 3 Baud Rate} = \frac{\text{Core Clock Frequency}}{(16 \times [65536 - (\text{RCAP2H} : \text{RCAP2L})])}$$



NOTES  
1. THE CORE CLOCK IS THE OUTPUT OF THE PLL (SEE THE ON-CHIP PLL SECTION)

Figure 60. Timer 2, UART Baud Rates

### Timer 3 Generated Baud Rates

The high integer dividers in a UART block mean that high speed baud rates are not always possible. Also, generating baud rates requires the exclusive use of a timer, rendering it unusable for other applications when the UART is required. To address this problem, the ADuC845/ADuC847/ADuC848 have 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 bps to 393216 bps can be generated to within an error of  $\pm 0.8\%$ . 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 61.

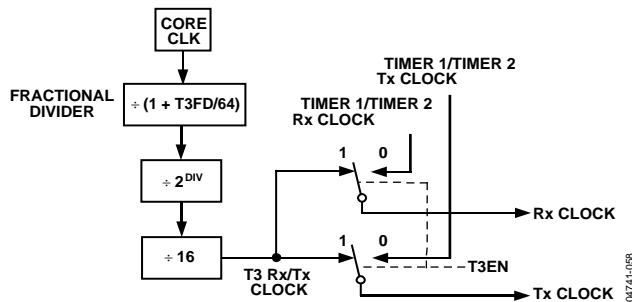


Figure 61. Timer 3, UART Baud Rate

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 to set up the binary divider (DIV).

The appropriate value to write to the DIV2-1-0 bits can be calculated using the following formula where  $f_{\text{CORE}}$  is defined in PLLCON SFR. Note that the DIV value must be rounded down.

$$DIV = \frac{\log\left(\frac{\text{Core Clock Frequency}}{16 \times \text{Baud Rate}}\right)}{\log(2)}$$

T3FD is the fractional divider ratio required to achieve the required baud rate. The appropriate value for T3FD can be calculated with the following formula:

$$T3FD = \frac{2 \times \text{Core Clock Frequency}}{2^{DIV-1} \times \text{Baud Rate}} - 64$$

Note that T3FD should be rounded to the nearest integer. Once the values for DIV and T3FD are calculated, the actual baud rate can be calculated with the following formula:

$$\text{Actual Baud Rate} = \frac{2 \times \text{Core Clock Frequency}}{2^{DIV-1} \times (T3FD + 64)}$$

For example, to get a baud rate of 9600 while operating at a core clock frequency of 1.5725 MHz, that is, CD = 3,

$$DIV = \log(1572500 / (16 \times 9600)) / \log 2 = 3.35 = 3$$

Note that the DIV result is rounded down.

$$T3FD = (2 \times 1572500) / (2^{3-1} \times 9600) - 64 = 18 = 12H$$

Therefore, the actual baud rate is 9588 bps, which gives an error of 0.12%.

The T3CON and T3FD registers are used to control Timer 3.

#### T3CON – Timer 3 Control Register

SFR Address: 9EH  
Power-On Default: 00H  
Bit Addressable: No

as op amps and voltage reference) can be powered from the  $AV_{DD}$  supply line as well.

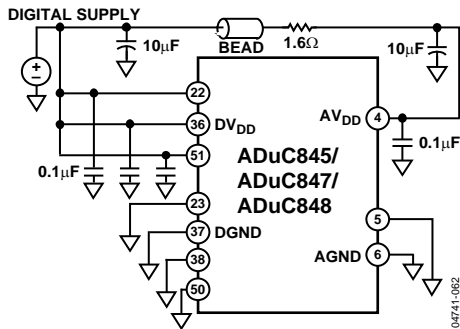


Figure 65. External Single-Supply Connections (56-Lead LFCSP Pin Numbering)

Notice that in both Figure 64 and Figure 65 a large value (10  $\mu$ F) reservoir capacitor sits on  $DV_{DD}$  and a separate 10  $\mu$ F capacitor sits on  $AV_{DD}$ . Also, local decoupling capacitors (0.1  $\mu$ F) are located at each  $V_{DD}$  pin of the chip. As per standard design practice, be sure to include all of these capacitors and ensure that the smaller capacitors are closer than the 10  $\mu$ F capacitors 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, note that, at all times, the analog and digital ground pins on the device must be referenced to the same system ground reference point. It is recommended that the LFCSP paddle be soldered to ensure mechanical stability but be floated with respect to system  $V_{DD}$ S or grounds.

## POWER-ON RESET OPERATION

An internal power-on reset (POR) is implemented on the [ADuC845/ADuC847/ADuC848](#).

### 3 V Part

For  $DV_{DD}$  below 2.63 V, the internal POR holds the device in reset. As  $DV_{DD}$  rises above 2.63 V, an internal timer times out for typically 128 ms before the device is released from reset. The user must ensure that the power supply has at least reached a stable 2.7 V minimum level by this time. Likewise on power-down, the internal POR holds the device in reset until the power supply drops below 1 V. Figure 66 illustrates the operation of the internal POR.

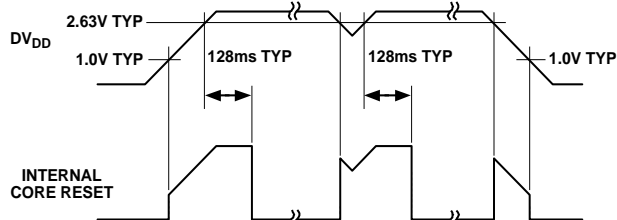


Figure 66. 3 V Part POR operation

### 5 V Part

For  $DV_{DD}$  below 4.5 V, the internal POR holds the device in reset. As  $DV_{DD}$  rises above 4.5 V, an internal timer times out for approximately 128 ms before the device is released from reset. The user must ensure that the power supply has reached a stable 4.75 V minimum level by this time. Likewise on power-down, the internal POR holds the device in reset until the power supply drops below 1 V. Figure 67 illustrates this operation.

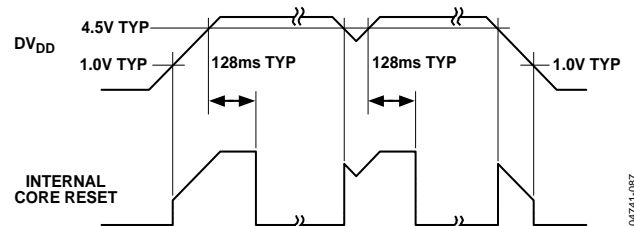


Figure 67. 5 V Part POR Operation

## POWER CONSUMPTION

The  $DV_{DD}$  power supply current consumption is specified in normal 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 (such as the watchdog timer and power supply monitor) consume negligible current and are therefore included with the normal operating current. The user must add any currents sourced by the parallel and serial I/O pins, and those sourced by the DAC to determine the total current needed at the [ADuC845/ADuC847/ADuC848](#)  $DV_{DD}$  and  $AV_{DD}$  supply pins. Also, current drawn from the  $DV_{DD}$  supply increases by approximately 5 mA during Flash/EE erase and program cycles.

## POWER-SAVING MODES

Setting the power-down mode bit,  $PCON.1$ , in the  $PCON$  SFR described in Table 6, allows the chip to be switched from normal mode into full power-down mode.

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, driven directly from the oscillator, can also be enabled during power-down. However, all other on-chip peripherals 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  $PSEN$  outputs are held low. There are five ways to terminate power-down mode:

- **Asserting the RESET Pin**

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 de-asserted.



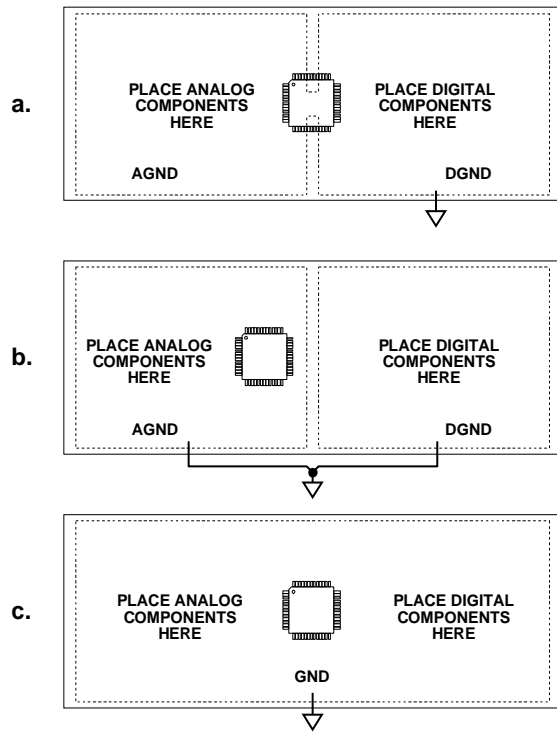


Figure 68. System Grounding Schemes

If the user plans to connect fast logic signals (rise/fall time < 5 ns) to any of the digital inputs of the ADuC845/ADuC847/ADuC848 add a series resistor to each relevant line to keep rise and fall times longer than 5 ns at the input pins of the device. A value of 100  $\Omega$  or 200  $\Omega$  is usually sufficient to prevent high speed signals from coupling capacitively into the device and affecting the accuracy of ADC conversions.

When using the LFCSP package, it is recommended that the paddle underneath the chip be soldered to the board to provide maximum mechanical stability. However, it is recommended that this paddle not be grounded but left floating. All results and specifications contained in this data sheet are taken or recorded with the paddle floating.

### System Self-Identification

In some hardware designs, it may be advantageous for the software to be able to identify the host MicroConverter.

The CHIPID SFR is a read-only register located at SFR address C2H. The upper nibble of this SFR designates the MicroConverter within the  $\Sigma$ - $\Delta$  ADC family. User software can read this SFR to identify the host MicroConverter and therefore execute slightly different code if required. The CHIPID SFR reads as follows for the  $\Sigma$ - $\Delta$  ADC family of MicroConverter products. Note that the ADuC845/ADuC847/ADuC848 are treated as one device as far as the CHIPID is concerned.

Table 63. CHIPID Values for  $\Sigma$ - $\Delta$  MicroConverter Products

Device	CHIPID
ADuC816	1xH
ADuC824	0xH
ADuC836	3xH
ADuC834	2xH
ADuC845/ADuC847/ADuC848	AxH

### Clock Oscillator

As described earlier, the core clock frequency for the ADuC845/ADuC847/ADuC848 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 as shown in Figure 69.

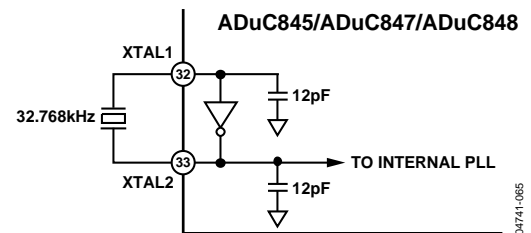


Figure 69. Crystal Connectivity to ADuC845/ADuC847/ADuC848

As shown in the typical external crystal connection diagram in Figure 69, two internal 12 pF capacitors are provided on-chip. These are connected internally, directly to the XTAL1 and XTAL2 pins. The total input capacitance at both pins is detailed in the Specifications table. Note that the total capacitance required for a particular crystal must be in accordance with the crystal manufacturer. However, in most cases, no additional external capacitance is required above that already supplied on-chip.

## OTHER HARDWARE CONSIDERATIONS

### In-Circuit Serial Download Access

Nearly all ADuC845/ADuC847/ADuC848 designs can take advantage of the in-circuit reprogrammability of the chip. This is accomplished by a connection to the UART of the devices, which requires an external RS-232 chip for level translation if downloading code from a PC. Basic configuration of an RS-232 connection is shown in Figure 70 with a simple ADM3202-based circuit. If users would rather not include an RS-232 chip on the target board, refer to the uC006 Application Note, A 4-Wire UART-to-PC Interface, for a simple (and zero-cost-per-board) method of gaining in-circuit serial download access to the device.

## TIMING SPECIFICATIONS

AC inputs during testing are driven at  $DV_{DD} - 0.5$  V for Logic 1 and 0.45 V for Logic 0. Timing measurements are made at  $V_{IH}$  min for Logic 1 and  $V_{IL}$  max for Logic 0 as shown in Figure 72.

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 72.

$C_{LOAD}$  for all outputs = 80 pF, unless otherwise noted.

$AV_{DD} = 2.7$  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.

**Table 64. CLOCK INPUT (External Clock Driven XTAL1) Parameter**

		32.768 kHz External Crystal			Unit
		Min	Typ	Max	
$t_{CK}$	XTAL1 Period		30.52		$\mu$ s
$t_{CKL}$	XTAL1 Width Low		6.26		$\mu$ s
$t_{CKH}$	XTAL1 Width High		6.26		$\mu$ s
$t_{CKR}$	XTAL1 Rise Time		9		ns
$t_{CKF}$	XTAL1 Fall Time		9		ns
$1/t_{CORE}$	Core Clock Frequency <sup>1</sup>	0.098	1.57	12.58	MHz
$t_{CORE}$	Core Clock Period <sup>2</sup>		0.636		$\mu$ s
$t_{CYC}$	Machine Cycle Time <sup>3</sup>	10.2	0.636	0.08	$\mu$ s

<sup>1</sup> ADuC845/ADuC847/ADuC848 internal PLL locks onto a multiple (512 times) of the 32.768 kHz external crystal frequency to provide a stable 12.58 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.

<sup>2</sup> This number is measured at the default Core\_Clk operating frequency of 1.57 MHz.

<sup>3</sup> ADuC845/ADuC847/ADuC848 machine cycle time is nominally defined as  $1/\text{Core\_Clk}$ .

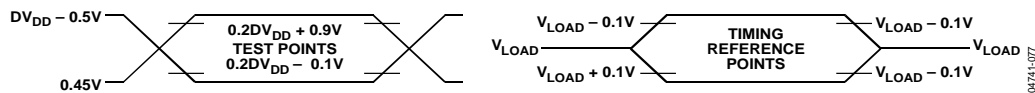
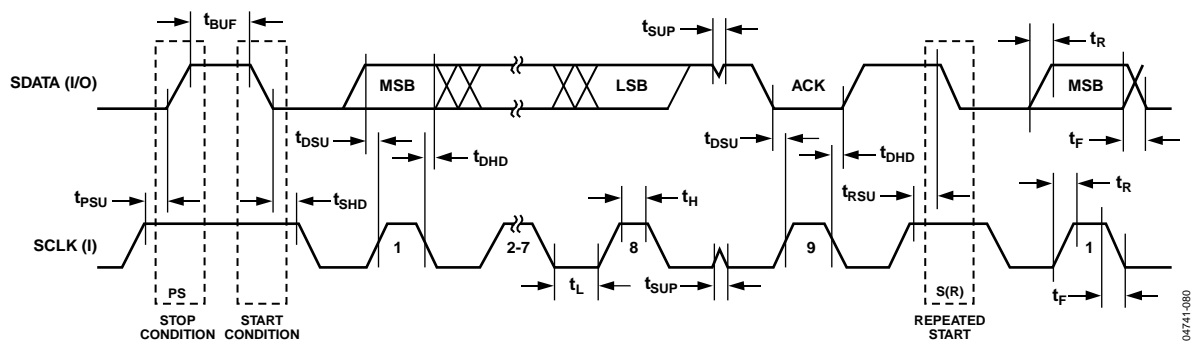


Figure 72. Timing Waveform Characteristics

Figure 75. I<sup>2</sup>C-Compatible Interface Timing

04741-080

Table 72. UART TIMING (SHIFT REGISTER MODE) Parameter

		12.58 MHz Core_Clk			Variable Core_Clk			Unit
		Min	Typ	Max	Min	Typ	Max	
TXLXL	Serial Port Clock Cycle Time		954			12t <sub>core</sub>		ns
TQVXH	Output Data Setup to Clock	662						ns
TDVXH	Input Data Setup to Clock	292						ns
TXHDX	Input Data Hold After Clock	0						ns
TXHQX	Output Data Hold After Clock	22						ns

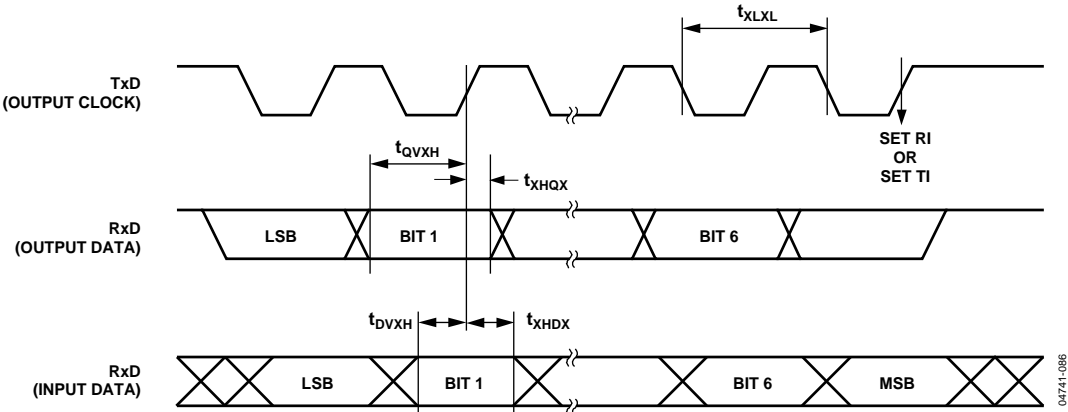


Figure 80. UART Timing in Shift Register Mode

## NOTES