**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 | I²C, SPI, UART/USART |
| Peripherals | POR, PSM, PWM, Temp Sensor, WDT |
| Number of I/O | 34 |
| Program Memory Size | 8KB (8K x 8) |
| Program Memory Type | FLASH |
| EEPROM Size | 4K x 8 |
| RAM Size | 2.25K x 8 |
| Voltage - Supply (Vcc/Vdd) | 4.75V ~ 5.25V |
| 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 | https://www.e-xfl.com/product-detail/analog-devices/aduc845bsz8-5 |

# 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. Input buffer on for primary ADC, unless otherwise noted. Core speed = 1.57 MHz (default CD = 3), unless otherwise noted.

**Table 1.**

| Parameter | Min | Typ | Max | Unit | Test Conditions/Comments |
|---|---|---|---|---|---|
| PRIMARY ADC | | | | | |
| Conversion Rate | 5.4 | | 105 | Hz | Chop on (ADCMODE.3 = 0) |
| | 16.06 | | 1365 | Hz | Chop off (ADCMODE.3 = 1) |
| No Missing Codes[2] | 24 | | | Bits | ≤26.7 Hz update rate with chop enabled |
| | 24 | | | Bits | ≤80.3 Hz update rate with chop disabled |
| Resolution (ADuC845/ADuC847) | See Table 11 and Table 15 | | | | |
| Resolution (ADuC848) | See Table 13 and Table 17 | | | | |
| Output Noise (ADuC845/ADuC847) | See Table 10 and Table 14 | | | µV (rms) | Output noise varies with selected update rates, gain range, and chop status. |
| Output Noise (ADuC848) | See Table 12 and Table 16 | | | µV (rms) | Output noise varies with selected update rates, gain range, and chop status. |
| Integral Nonlinearity | | | ±15 | ppm of FSR | 1 $LSB_{16}$ |
| Offset Error[3] | | ±3 | | µV | Chop on |
| | | | | | Chop off, offset error is in the order of the noise for the programmed gain and update rate following a calibration. |
| Offset Error Drift vs. Temperature[2] | | ±10 | | nV/°C | Chop on (ADCMODE.3 = 0) |
| | | ±200 | | nV/°C | Chop off (ADCMODE.3 = 1) |
| Full-Scale Error[4] | | | | | |
| ADuC845/ADuC847 | | ±10 | | µV | ±20 mV to ±2.56 V |
| ADuC848 | | ±10 | | µV | ±20 mV to ±640 mV |
| | | ±0.5 | | $LSB_{16}$ | ±1.28 V to ±2.56 V |
| Gain Error Drift vs. Temperature[4] | | ±0.5 | | ppm/°C | |
| Power Supply Rejection | | | | | |
| | 80 | | | dB | AIN = 1 V, ±2.56 V, chop enabled |
| | | 113 | | dB | AIN = 7.8 mV, ±20 mV, chop enabled |
| | | 80 | | dB | AIN = 1 V, ±2.56 V, chop disabled[2] |
| PRIMARY ADC ANALOG INPUTS | | | | | |
| Differential Input Voltage Ranges[5, 6] | | | | | Gain = 1 to 128 |
| Bipolar Mode (ADC0CON1.5 = 0) | | ±1.024 × $V_{REF}$/GAIN | | V | $V_{REF}$ = REFIN(+) – REFIN(–) or REFIN2(+) – REFIN2(–) (or Int 1.25 $V_{REF}$) |
| Unipolar Mode (ADC0CON1.5 = 1) | | 0 – 1.024 × $V_{REF}$/GAIN | | V | $V_{REF}$ = REFIN(+) – REFIN(–) or REFIN2(+) – REFIN2(–) (or Int 1.25 $V_{REF}$) |
| ADC Range Matching | | ±2 | | µV | AIN = 18 mV, chop enabled |
| Common-Mode Rejection DC | | | | | Chop enabled, chop disabled |
| On AIN | 95 | | | dB | AIN = 7.8 mV, range = ±20 mV |
| | | 113 | | dB | AIN = 1 V, range = ±2.56 V |
| Common-Mode Rejection 50 Hz/60 Hz[2] | | | | | 50 Hz/60 Hz ± 1 Hz, 16.6 Hz and 50 Hz update rate, chop enabled, REJ60 enabled |
| On AIN | 95 | | | dB | AIN = 7.8 mV, range = ±20 mV |
| | 90 | | | dB | AIN = 1 V, range = ±2.56 V |

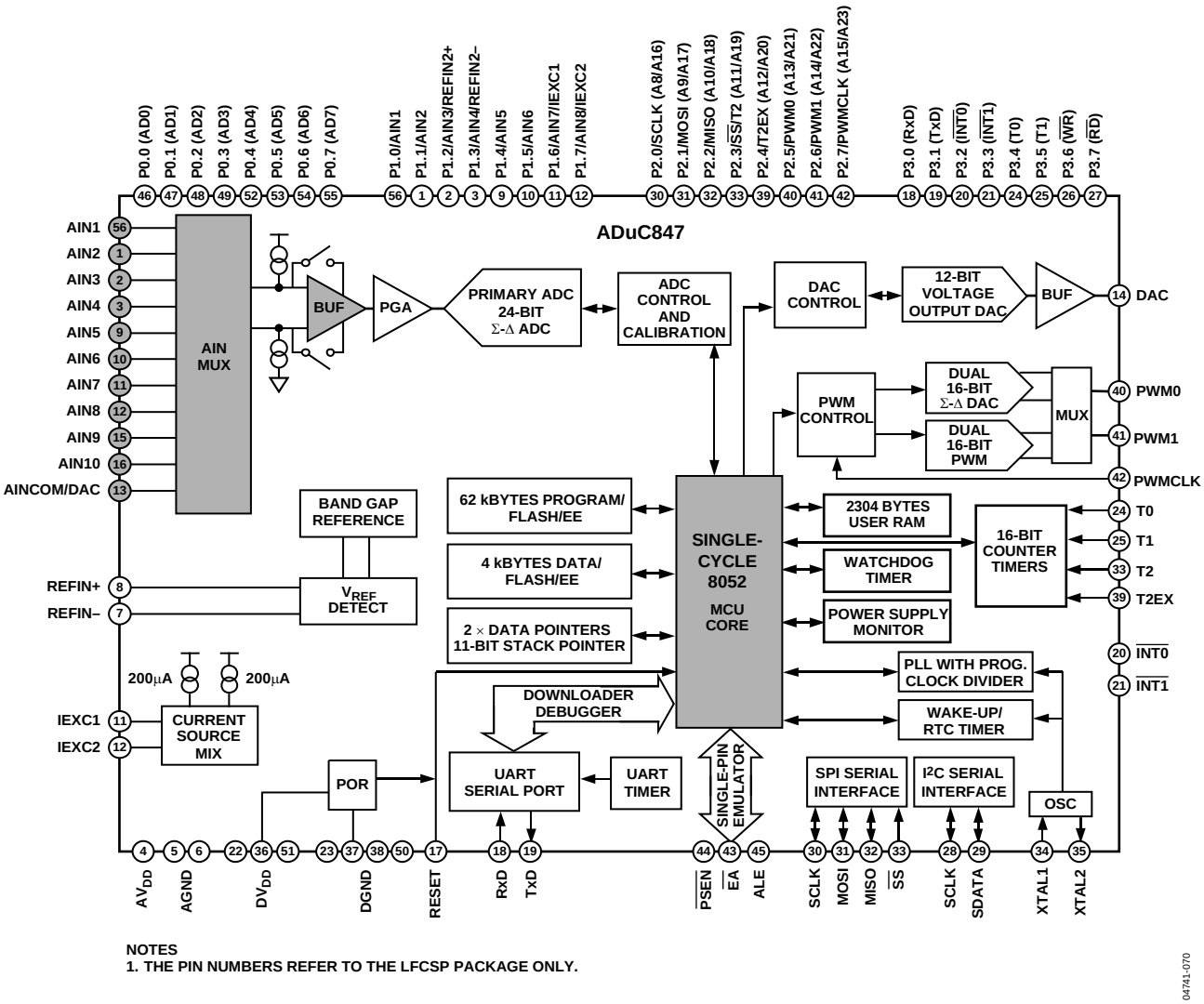


*Figure 5. Detailed Block Diagram of the ADuC847*

| Mnemonic | Description | Bytes | Cycles[1] |
|---|---|---|---|
| RLC A | Rotate A left through carry | 1 | 1 |
| RR A | Rotate A right | 1 | 1 |
| RRC A | Rotate A right through carry | 1 | 1 |
| **Data Transfer** | | | |
| MOV A,Rn | Move register to A | 1 | 1 |
| MOV A,@Ri | Move indirect memory to A | 1 | 2 |
| MOV Rn,A | Move A to register | 1 | 1 |
| MOV @Ri,A | Move A to indirect memory | 1 | 2 |
| MOV A,dir | Move direct byte to A | 2 | 2 |
| MOV A,#data | Move immediate to A | 2 | 2 |
| MOV Rn,#data | Move register to immediate | 2 | 2 |
| MOV dir,A | Move A to direct byte | 2 | 2 |
| MOV Rn, dir | Move register to direct byte | 2 | 2 |
| MOV dir, Rn | Move direct to register | 2 | 2 |
| MOV @Ri,#data | Move immediate to indirect memory | 2 | 2 |
| MOV dir,@Ri | Move indirect to direct memory | 2 | 2 |
| MOV @Ri,dir | Move direct to indirect memory | 2 | 2 |
| MOV dir,dir | Move direct byte to direct byte | 3 | 3 |
| MOV dir,#data | Move immediate to direct byte | 3 | 3 |
| MOV DPTR,#data | Move immediate to data pointer | 3 | 3 |
| MOVC A,@A+DPTR | Move code byte relative DPTR to A | 1 | 4 |
| MOVC A,@A+PC | Move code byte relative PC to A | 1 | 4 |
| MOVX[2] A,@Ri | Move external (A8) data to A | 1 | 4 |
| MOVX[2] A,@DPTR | Move external (A16) data to A | 1 | 4 |
| MOVX[2] @Ri,A | Move A to external data (A8) | 1 | 4 |
| MOVX[2] @DPTR,A | Move A to external data (A16) | 1 | 4 |
| PUSH dir | Push direct byte onto stack | 2 | 2 |
| POP dir | Pop direct byte from stack | 2 | 2 |
| XCH A,Rn | Exchange A and register | 1 | 1 |
| XCH A,@Ri | Exchange A and indirect memory | 1 | 2 |
| XCHD A,@Ri | Exchange A and indirect memory nibble | 1 | 2 |
| XCH A,dir | Exchange A and direct byte | 2 | 2 |
| **Boolean** | | | |
| CLR C | Clear carry | 1 | 1 |
| CLR bit | Clear direct bit | 2 | 2 |
| SETB C | Set carry | 1 | 1 |
| SETB bit | Set direct bit | 2 | 2 |
| CPL C | Complement carry | 1 | 1 |
| CPL bit | Complement direct bit | 2 | 2 |
| ANL C,bit | AND direct bit and carry | 2 | 2 |
| ANL C,/bit | AND direct bit inverse to carry | 2 | 2 |
| ORL C,bit | OR direct bit and carry | 2 | 2 |
| ORL C,/bit | OR direct bit inverse to carry | 2 | 2 |
| MOV C,bit | Move direct bit to carry | 2 | 2 |
| MOV bit,C | Move carry to direct bit | 2 | 2 |
| **Branching** | | | |
| JMP @A+DPTR | Jump indirect relative to DPTR | 1 | 3 |
| RET | Return from subroutine | 1 | 4 |
| RETI | Return from interrupt | 1 | 4 |
| ACALL addr11 | Absolute jump to subroutine | 2 | 3 |
| AJMP addr11 | Absolute jump unconditional | 2 | 3 |

### *Signal Chain Overview (Chop Enabled, $\overline{CHOP}$ = 0)*

With the $\overline{CHOP}$ bit = 0 (see the ADCMODE SFR bit designations in Table 24), the chopping scheme is enabled. This is the default condition and gives optimum performance in terms of offset errors and drift performance. With chop enabled, the available output rates vary from 5.35 Hz to 105 Hz (SF = 255 and 13, respectively). A typical block diagram of the ADC input channel with chop enabled is shown in Figure 12.

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. 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 ADCs.

The ADC filter is a low-pass Sinc$^3$ or (sinx/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 Sinc filter word loaded into the filter (SF) register (see Table 28). The complete signal chain is chopped, resulting in excellent dc offset and offset drift specifications and is extremely beneficial in applications where drift, noise rejection, and optimum EMI rejection are important.

With chop enabled, the ADC repeatedly reverses its inputs. The decimated digital output words from the Sinc$^3$ filter, therefore, have a positive offset and a negative offset term included. As a result, a final summing stage is included so that each output word from the filter is summed and averaged with the previous filter output to produce a new valid output result to be written to the ADC data register. Programming the Sinc$^3$ decimation factor is restricted to an 8-bit register called SF (see Table 28), the actual decimation factor is the register value times 8. Therefore, the decimated output rate from the Sinc$^3$ filter (and the ADC conversion rate) is

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

where:
$f_{ADC}$ is the ADC conversion rate.
*SF* is the decimal equivalent of the word loaded to the filter register.
$f_{MOD}$ is the modulator sampling rate of 32.768 kHz.

The chop rate of the channel is half the output data rate:

$$f_{CHOP} = \frac{1}{2 \times f_{ADC}}$$

As shown in the block diagram (Figure 12), the Sinc$^3$ filter outputs alternately contain $+V_{OS}$ and $-V_{OS}$, where $V_{OS}$ is the respective channel offset.

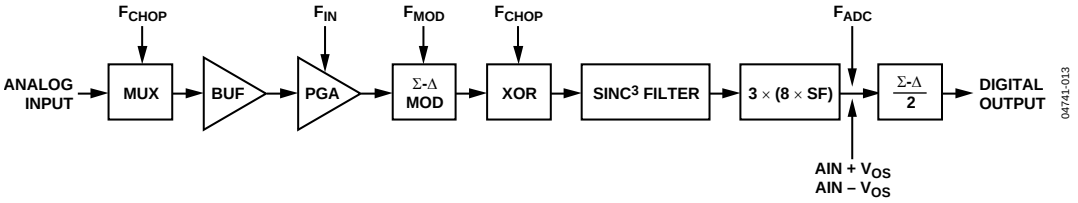


*Figure 12. Block Diagram of the ADC Input Channel with Chop Enabled*

that the internal calibration procedure for full-scale calibration automatically selects the reference in voltage at PGA = 1.

Therefore, the full-scale endpoint calibration automatically subtracts the offset calibration error, it is advisable to perform an offset calibration at the same gain range as that used for full-scale calibration. There is no penalty to the full-scale calibration in redoing the zero-scale calibration at the required PGA range because the full-scale calibration has very good matching at all the PGA ranges.

This procedure also applies when chop is disabled.

Note that for internal calibration to be effective, the AIN− pin should be held at a steady voltage, within the allowable common-mode range to keep it from floating during calibration.

### *System Calibration Example*

With chop enabled, a system zero-scale or offset calibration should never be required. However, if a full-scale or gain calibration is required for any reason, use the following typical procedure for doing so.

1. Apply a differential voltage of 0 V to the selected analog inputs (AIN+ to AIN−) that are held at a common-mode voltage.

   Perform a system zero-scale or offset calibration by setting the MD2...0 bits in the ADCMODE register to 110B.

2. Apply a full-scale differential voltage across the ADC inputs again at the same common-mode voltage.

   Perform a system full-scale or gain calibration by setting the MD2...0 bits in the ADCMODE register to 111B.

Perform a system calibration at the required PGA range to be used since the ADC scales to the differential voltages that are applied to the ADC during the calibration routines.

In bipolar mode, the zero-scale calibration determines the mid-scale point of the ADC (800000H) or 0 V.

## PROGRAMMABLE GAIN AMPLIFIER

The primary ADC incorporates an on-chip programmable gain amplifier (PGA). The PGA can be programmed through eight different ranges, which are programmed via the range bits (RN0 to RN2) in the ADC0CON1 register. With an external 2.5 V reference applied, the unipolar ranges are 0 mV to 20 mV, 0 mV to 40 mV, 0 mV to 80 mV, 0 mV to 160 mV, 0 mV to 320 mV, 0 mV to 640 mV, 0 V to 1.28 V and 0 V to 2.56 V, while in bipolar mode the ranges are ±20 mV, ±40 mV, ±80 mV, ±160 mV, ±320 mV, ±64 0 mV, ±1.28 V, and ±2.56 V. These ranges should appear on the input to the on-chip PGA. The ADC range-matching specification of 2 µV (typical with chop enabled) means that calibration need only be carried out on a single range and need not be repeated when the ADC range is changed. This is a significant advantage compared to similar mixed-signal solutions available on the market. The auxiliary (ADuC845 only) ADC does not incorporate a PGA, and the gain is fixed at 0 V to 2.50 V in unipolar mode, and ±2.50 V in bipolar mode.

## BIPOLAR/UNIPOLAR CONFIGURATION

The analog inputs of the ADuC845/ADuC847/ADuC848 can accept either unipolar or bipolar input voltage ranges. Bipolar input ranges do not imply that the device can handle negative voltages with respect to system AGND, but rather with respect to the negative reference input. Unipolar and bipolar signals on the AIN(+) input on the ADC are referenced to the voltage on the respective AIN(−) input. AIN(+) and AIN(−) refer to the signals seen by the ADC.

For example, if AIN(−) is biased to 2.5 V (tied to the external reference voltage) and the ADC is configured for a unipolar analog input range of 0 mV to >20 mV, the input voltage range on AIN(+) is 2.5 V to 2.52 V. On the other hand, if AIN(−) is biased to 2.5 V (again the external reference voltage) and the ADC is configured for a bipolar analog input range of ±1.28 V, the analog input range on the AIN(+) is 1.22 V to 3.78 V, that is, 2.5 V ± 1.28 V.

The modes of operation for the ADC are fully differential mode or pseudo differential mode. In fully differential mode, AIN1 to AIN2 are one differential pair, and AIN3 to AIN4 are another pair (AIN5 to AIN6, AIN7 to AIN8, and AIN9 to AIN10 are the others). In differential mode, all AIN(−) pin names imply the negative analog input of the selected differential pair, that is, AIN2, AIN4, AIN6, AIN8, AIN10. The term AIN(+) implies the positive input of the selected differential pair, that is, AIN1, AIN3, AIN5, AIN7, AIN9. In pseudo differential mode, each analog input is paired with the AINCOM pin, which can be biased up or tied to AGND. In this mode, the AIN(−) implies AINCOM, and AIN(+) implies any one of the ten analog input channels.

The configuration of the inputs (unipolar vs. bipolar) is shown in Figure 17.

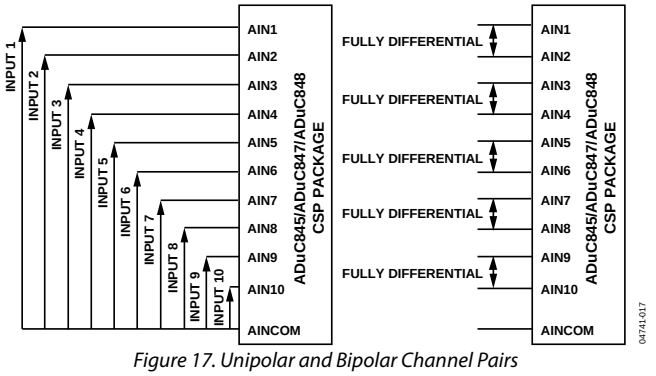


*Figure 17. Unipolar and Bipolar Channel Pairs*

# FUNCTIONAL DESCRIPTION

## ADC SFR INTERFACE

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

**Table 22. ADC SFR Interface**

| Name | Description |
|---|---|
| ADCSTAT | ADC Status Register. Holds the general status of the primary and auxiliary (ADuC845 only) ADCs. |
| ADCMODE | ADC Mode Register. Controls the general modes of operation for primary and auxiliary (ADuC845 only) ADCs. |
| ADC0CON1 | Primary ADC Control Register 1. Controls the specific configuration of the primary ADC. |
| ADC0CON2 | Primary ADC Control Register 2. Controls the specific configuration of the primary ADC. |
| ADC1CON | Auxiliary ADC Control Register. Controls the specific configuration of the auxiliary ADC. ADuC845 only. |
| SF | Sinc Filter Register. Configures the decimation factor for the Sinc$^3$ filter and, therefore, the primary and auxiliary (ADuC845 only) ADC update rates. |
| ICON | Current Source Control Register. Allows user control of the various on-chip current source options. |
| ADC0L/M/H | Primary ADC 24-bit (16-bit on the ADuC848) conversion result is held in these three 8-bit registers. ADC0L is not available on the ADuC848. |
| ADC1L/M/H | Auxiliary ADC 24-bit conversion result is held in these two 8-bit registers. ADuC845 only. |
| OF0L/M/H | Primary ADC 24-bit offset calibration coefficient is held in these three 8-bit registers. OF0L is not available on the ADuC848. |
| OF1L/H | Auxiliary ADC 16-bit offset calibration coefficient is held in these two 8-bit registers. ADuC845 only. |
| GN0L/M/H | Primary ADC 24-bit gain calibration coefficient is held in these three 8-bit registers. GN0L is not available on the ADuC848. |
| GN1L/H | Auxiliary ADC 16-bit gain calibration coefficient is held in these two 8-bit registers. ADuC845 only. |

## USER DOWNLOAD MODE (ULOAD)

Figure 28 shows that it is possible to use the 62 kbytes of Flash/EE program memory available to the user as one single block of memory. In this mode, all the Flash/EE memory is read-only to user code.

However, most of the Flash/EE program memory can also be written to during run time simply by entering ULOAD mode. In ULOAD mode, the lower 56 kbytes of program memory can be erased and reprogrammed by the user software as shown in Figure 30. ULOAD mode can be used to upgrade the code in the field via any user-defined download protocol. By configuring the SPI port on the ADuC845/ADuC847/ADuC848 as a slave, it is possible to completely reprogram the 56 kbytes of Flash/EE program memory in under 5 s (see the AN-1074 Application Note).

Alternatively, ULOAD mode can be used to save data to the 56 kbytes of Flash/EE memory. This can be extremely useful in data logging applications where the devices can provide up to 60 kbytes of data memory on-chip (4 kbytes of dedicated Flash/EE data memory also exist).

The upper 6 kbytes of the 62 kbytes of Flash/EE program memory (8 kbytes on the 32-kbyte parts) are programmable only via serial download or parallel programming. This means that this space appears as read-only to user code; therefore, it cannot be accidentally erased or reprogrammed by erroneous code execution, making it very suitable to use the 6 kbytes as a bootloader. A bootload enable option exists in the Windows® serial downloader (WSD) to "Always RUN from E000H after Reset." If using a bootloader, this option is recommended to ensure that the bootloader always executes correct code after reset.

Programming the Flash/EE program memory via ULOAD mode is described in the Flash/EE Memory Control SFR section of ECON and also in the AN-1074 Application Note.

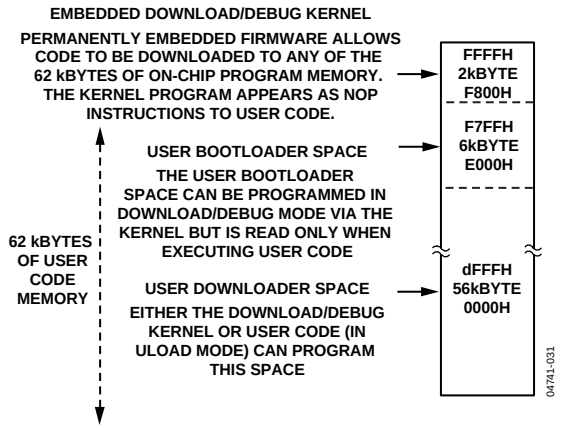


*Figure 30. Flash/EE Program Memory Map in ULOAD Mode (62-kbyte Part)*

The 32-kbyte memory parts have the user bootload space starting at 6000H. The memory mapping is shown in Figure 31.

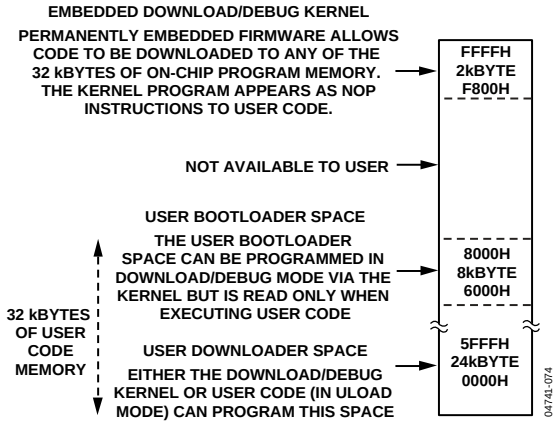


*Figure 31. Flash/EE Program Memory Map in ULOAD Mode (32-kbyte Part)*

ULOAD mode is not available on the 8-kbyte Flash/EE program memory parts.

### *Flash/EE Program Memory Security*

The ADuC845/ADuC847/ADuC848 facilitate three modes of Flash/EE program memory security: the lock, secure, and serial safe modes. These modes can be independently activated, restricting access to the internal code space. They can be enabled as part of serial download protocol, as described in the AN-1074 Application Note, or via parallel programming.

### *Lock Mode*

This mode locks the code memory, disabling parallel programming of the program memory. However, reading the memory in parallel mode and reading the memory via a MOVC command from external memory are still allowed. This mode is deactivated by initiating an ERASE CODE AND DATA command in serial download or parallel programming modes.

### *Secure Mode*

This mode locks the code memory, disabling parallel programming of the program memory. Reading/verifying the memory in parallel mode and reading the internal memory via a MOVC command from external memory are also disabled. This mode is deactivated by initiating an ERASE CODE AND DATA command in serial download or parallel programming modes.

### *Serial Safe Mode*

This mode disables serial download capability on the device. If serial safe mode is activated and an attempt is made to reset the device into serial download mode, that is, RESET asserted (pulled high) and de-asserted (pulled low) with $\overline{\text{PSEN}}$ low, the device interprets the serial download reset as a normal reset only. It therefore does not enter serial download mode, but executes only a normal reset sequence. Serial safe mode can be disabled only by initiating an ERASE CODE AND DATA command in parallel programming mode.

**Table 36. PWM0L: PWM Pulse Width Low Byte**

| PWM0L.7 | PWM0L.6 | PWM0L.5 | PWM0L.4 | PWM0L.3 | PWM0L.2 | PWM0L.1 | PWM0L.0 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

### *PWM Cycle Width High Byte (PWM1H)*

SFR Address: B4H
Power-On Default: 00H
Bit Addressable: No

**Table 37. PWM1H: PWM Cycle Width High Byte**

| PWM1H.7 | PWM1H.6 | PWM1H.5 | PWM1H.4 | PWM1H.3 | PWM1H.2 | PWM1H.1 | PWM1H.0 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

### *PWM Cycle Width Low Byte (PWM1L)*

SFR Address: B3H
Power-On Default: 00H
Bit Addressable: No

**Table 38. PWM1L: PWM Cycle Width Low Byte**

| PWM1L.7 | PWM1L.6 | PWM1L.5 | PWM1L.4 | PWM1L.3 | PWM1L.2 | PWM1L.1 | PWM1L.0 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

### *Mode 0*

In Mode 0, the PWM is disabled, allowing P2.5 and P2.6 to be used as normal digital I/Os.

### *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. For example, 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 39.

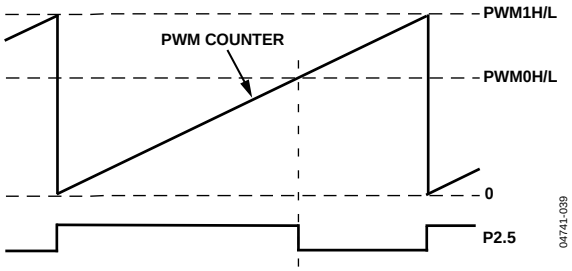


*Figure 39. PWM in Mode 1*

### *Mode 2 (Twin 8-Bit PWM)*

In Mode 2, the duty cycle and the resolution of the PWM outputs are programmable. The maximum resolution of the PWM output is 8 bits.

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

The outputs of the PWM at P2.5 and P2.6 are shown in Figure 40. As can be seen, the output of PWM0 (P2.5) goes low when the PWM counter equals PWM0L. The output of PWM1 (P2.6) 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.

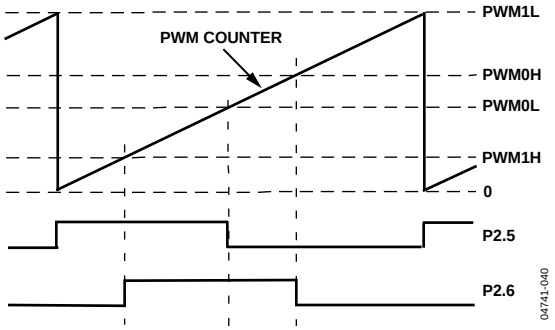


*Figure 40. PWM Mode 2*

### *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 P2.5 and P2.6 are independently programmable.

As shown in Figure 41, while the PWM counter is less than PWM0H/L, the output of PWM0 (P2.5) is high. Once the PWM counter equals PWM0H/L, PWM0 (P2.5) 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 (P2.6) is high. Once the PWM counter equals PWM1H/L, PWM1 (P2.6) goes low and remains low until the PWM counter rolls over.

In this mode, both PWM outputs are synchronized, that is, once the PWM counter rolls over to 0, both PWM0 (P2.5) and PWM1 (P2.6) go high.

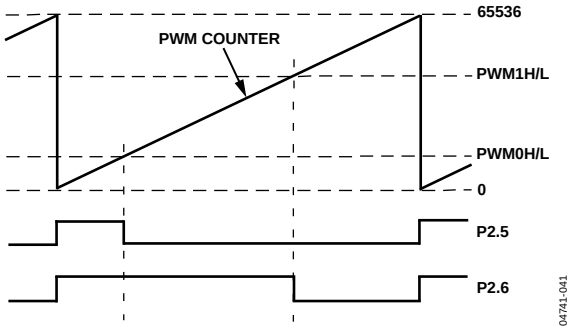


*Figure 41. PWM Mode 3*

### *Mode 4 (Dual NRZ 16-Bit Σ-Δ DAC)*

Mode 4 provides a high speed PWM output similar to that of a Σ-Δ DAC. Typically, this mode is used with the PWM clock equal to 12.58 MHz.

In this mode, P2.5 and P2.6 are updated every PWM clock (80 ns in the case of 12.58 MHz). Over any 65536 cycles (16-bit PWM), PWM0 (P2.5) is high for PWM0H/L cycles and low for (65536 – PWM0H/L) cycles. Similarly, PWM1 (P2.6) is high for PWM1H/L cycles and low for (65536 – PWM1H/L) cycles.

If PWM1H is set to 4010H (slightly above one-quarter of FS), typically P2.6 is low for three clocks and high for one clock (each clock is approximately 80 ns). Over every 65536 clocks, the PWM compromises for the fact that the output should be slightly above one-quarter of full scale, by having a high cycle followed by only two low cycles.

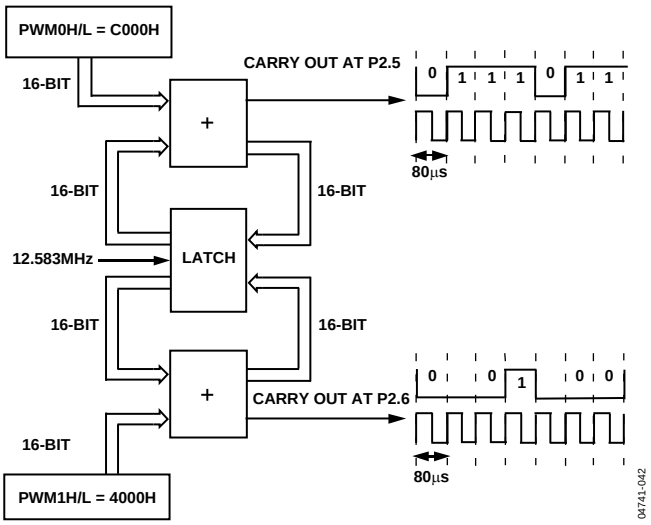


*Figure 42. PWM Mode 4*

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

## ON-CHIP PLL (PLLCON)

The ADuC845/ADuC847/ADuC848 are intended for use with a 32.768 kHz watch crystal. A PLL locks onto a multiple (384) of this to provide a stable 12.582912 MHz clock for the system. The core can operate at this frequency or at binary submultiples of it to allow power saving when maximum core performance is not required. The default core clock is the PLL clock divided by 8 or 1.572864 MHz. The ADC clocks are also derived from the PLL clock, with the modulator rate being the same as the crystal oscillator frequency. The control register for the PLL is called PLLCON and is described as follows.

The 5 V parts can be set to a maximum core frequency of 12.58 MHz (CD2...0 = 000) while at 3 V, the maximum core clock rate is 6.29 MHz (CD2...0 = 001). The CD bits should not be set to 000b on the 3 V parts.

The 3 V parts are limited to a core clock speed of 6.29 MHz (CD = 1).

### PLLCON PLL Control Register

SFR Address: D7H
Power-On Default: 53H
Bit Addressable: No

**Table 39. PLLCON PLL Control Register**

| Bit No. | Name | Description |
|---|---|---|
| 7 | OSC_PD | Oscillator Power-Down Bit.<br>If low, the 32 kHz crystal oscillator continues running in power-down mode.<br>If high, the 32.768 kHz oscillator is powered down.<br>When this bit is low, the seconds counter continues to count in power-down mode and can interrupt the CPU to exit power-down. The oscillator is always enabled in normal mode. |
| 6 | LOCK | PLL Lock Bit. This is a read-only bit.<br>Set automatically at power-on to indicate that the PLL loop is correctly tracking the crystal clock. After power-down, this bit can be polled to wait for the PLL to lock.<br>Cleared automatically at power-on to indicate that the PLL is not correctly tracking the crystal clock. This might be due to the absence of a crystal clock or an external crystal at power-on. In this mode, the PLL output can be 12.58 MHz ± 20%. After the device wakes up from power-down, user code can poll this bit to wait for the PLL to lock. If LOCK = 0, the PLL is not locked. |
| 5 | ––– | Not Implemented. Write Don't Care. |
| 4 | LTEA | EA Status. Read-only bit. Reading this bit returns the state of the external $\overline{EA}$ pin latched at reset or power-on. |
| 3 | FINT | Fast Interrupt Response Bit.<br>Set by the user to enable the response to any interrupt to be executed at the fastest core clock frequency.<br>Cleared by the user to disable the fast interrupt response feature.<br>This function must not be used on 3 V parts. |
| 2, 1, 0 | CD2, CD1, CD0 | CPU (Core Clock) Divider Bits. This number determines the frequency at which the core operates.<br>CD2 CD1 CD0 Core Clock Frequency (MHz)<br>0 0 0 12.582912. Not a valid selection on 3 V parts.<br>0 0 1 6.291456 (Maximum core clock rate allowed on the 3 V parts)<br>0 1 0 3.145728<br>0 1 1 1.572864 (Default core frequency)<br>1 0 0 0.786432<br>1 0 1 0.393216<br>1 1 0 0.196608<br>1 1 1 0.098304<br>On 3 V parts (ADuC84xBCPxx-3 or ADuC84xBSxx-3), the CD settings can be only CD = 1; CD = 0 is not a valid selection. If CD = 0 is selected on a 3 V part by writing to PLLCON, the instruction is ignored, and the previous CD value is retained.<br>The Fast Interrupt bit (FINT) must not be used on 3 V parts since it automatically sets the CD bits to 0, which is not a valid setting. |

**SPICON—SPI Control Register**

SFR Address: F8H
Power-On Default: 05H
Bit Addressable: Yes

**Table 41. SPICON SFR Bit Designations**

| Bit No. | Name | Description |
|---|---|---|
| 7 | ISPI | SPI Interrupt Bit.<br>Set by the MicroConverter at the end of each SPI transfer.<br>Cleared directly by user code or indirectly by reading the SPIDAT SFR. |
| 6 | WCOL | Write Collision Error Bit.<br>Set by the MicroConverter if SPIDAT is written to while an SPI transfer is in progress.<br>Cleared by user code. |
| 5 | SPE | SPI Interface Enable Bit.<br>Set by user code to enable SPI functionality.<br>Cleared by user code to enable standard Port 2 functionality. |
| 4 | SPIM | SPI Master/Slave Mode Select Bit.<br>Set by user code to enable master mode operation (SCLOCK is an output).<br>Cleared by user code to enable slave mode operation (SCLOCK is an input). |
| 3 | CPOL[1] | Clock Polarity Bit.<br>Set by user code to enable SCLOCK idle high.<br>Cleared by user code to enable SCLOCK idle low. |
| 2 | CPHA[1] | Clock Phase Select Bit.<br>Set by user code if the leading SCLOCK edge is to transmit data.<br>Cleared by user code if the trailing SCLOCK edge is to transmit data. |
| 1, 0 | SPR1, SPR0 | SPI Bit-Rate Bits.<br>SPR1 SPR0 Selected Bit Rate<br>0 0 $f_{core}/2$<br>0 1 $f_{core}/4$<br>1 0 $f_{core}/8$<br>1 1 $f_{core}/16$ |

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

Note that both SPI and I²C use the same ISR (Vector Address 3BH); therefore, when using SPI and I²C simultaneously, it is necessary to check the interfaces following an interrupt to determine which one caused the interrupt.

***SPIDAT: SPI Data Register***

SFR Address: 7FH
Power-On Default: 00H
Bit Addressable: No

## DUAL DATA POINTERS

The devices incorporate two data pointers. The second data pointer is a shadow data pointer and is selected via the data pointer control SFR (DPCON). DPCON features automatic hardware post-increment and post-decrement as well as an automatic data pointer toggle.

**DPCON—Data Pointer Control SFR**

SFR Address: A7H
Power-On Default: 00H
Bit Addressable: No

**Table 42. DPCON SFR Bit Designations**

| Bit No. | Name | Description |
|---|---|---|
| 7 | ---- | Not Implemented. Write Don't Care. |
| 6 | DPT | Data Pointer Automatic Toggle Enable.<br>Cleared by the user to disable autoswapping of the DPTR.<br>Set in user software to enable automatic toggling of the DPTR after each MOVX or MOVC instruction. |
| 5, 4 | DP1m1, DP1m0 | Shadow Data Pointer Mode. These bits enable extra modes of the shadow data pointer operation, allowing more compact and more efficient code size and execution.<br>DP1m1 DP1m0 Behavior of the Shadow Data Pointer<br>0 0 8052 behavior.<br>0 1 DPTR is post-incremented after a MOVX or a MOVC instruction.<br>1 0 DPTR is post-decremented after a MOVX or MOVC instruction.<br>1 1 DPTR LSB is toggled after a MOVX or MOVC instruction. (This instruction can be useful for moving 8-bit blocks to/from 16-bit devices.) |
| 3, 2 | DP0m1, DP0m0 | Main Data Pointer Mode. These bits enable extra modes of the main data pointer operation, allowing more compact and more efficient code size and execution.<br>DP0m1 DP0m0 Behavior of the Main Data Pointer<br>0 0 8052 behavior.<br>0 1 DPTR is post-incremented after a MOVX or a MOVC instruction.<br>1 0 DPTR is post-decremented after a MOVX or MOVC instruction.<br>1 1 DPTR LSB is toggled after a MOVX or MOVC instruction. (This instruction is useful for moving 8-bit blocks to/from 16-bit devices.) |
| 1 | ---- | Not Implemented. Write Don't Care. |
| 0 | DPSEL | Data Pointer Select.<br>Cleared by the user to select the main data pointer. This means that the contents of this 24-bit register are placed into the DPL, DPH, and DPP SFRs.<br>Set by the user to select the shadow data pointer. This means that the contents of a separate 24-bit register appear in the DPL, DPH, and DPP SFRs. |

Note the following:

- The Dual Data Pointer section is the only place in which main and shadow data pointers are distinguished. Whenever the DPTR is mentioned elsewhere in this data sheet, active DPTR is implied.
- Only the MOVC/MOVX @DPTR instructions automatically post-increment and post-decrement the DPTR. Other MOVC/MOVX instructions, such as MOVC PC or MOVC @Ri, do not cause the DPTR to automatically post-increment and post-decrement.

To illustrate the operation of DPCON, the following code copies 256 bytes of code memory at Address D000H into XRAM, starting from Address 0000H.

```
        MOV DPTR,#0       ;Main DPTR = 0
        MOV DPCON,#55H    ;Select shadow DPTR
                          ;DPTR1 increment mode
                          ;DPTR0 increment mode
                          ;DPTR auto toggling ON
        MOV DPTR,#0D000H  ;DPTR = D000H
MOVELOOP: CLR A
        MOVC A,@A+DPTR    ;Get data
                          ;Post Inc DPTR
                          ;Swap to Main DPTR(Data)
         MOVX @DPTR,A     ;Put ACC in XRAM
                          ;Increment main DPTR
                          ;Swap Shadow DPTR(Code)
         MOV A, DPL
         JNZ MOVELOOP
```

### *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.

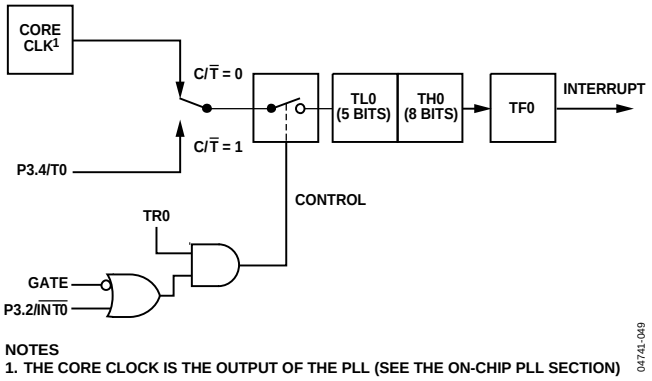


*Figure 52. Timer/Counter 0, Mode 0*

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.

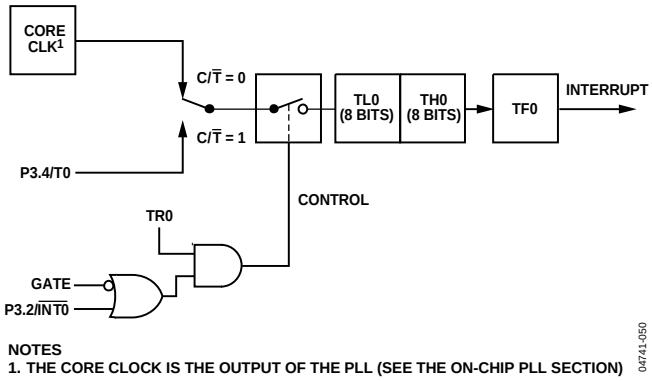


*Figure 53. Timer/Counter 0, Mode 1*

#### 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.

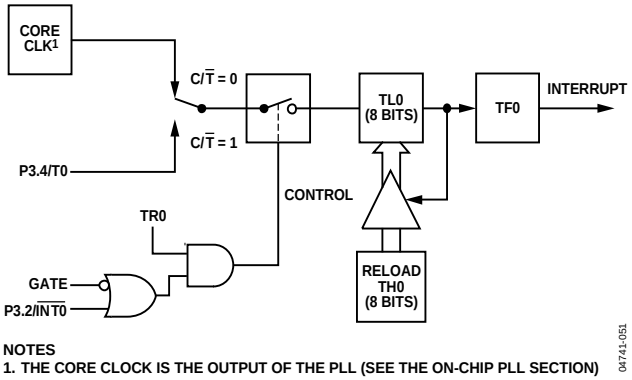


*Figure 54. Timer/Counter 0, Mode 2*

#### 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.

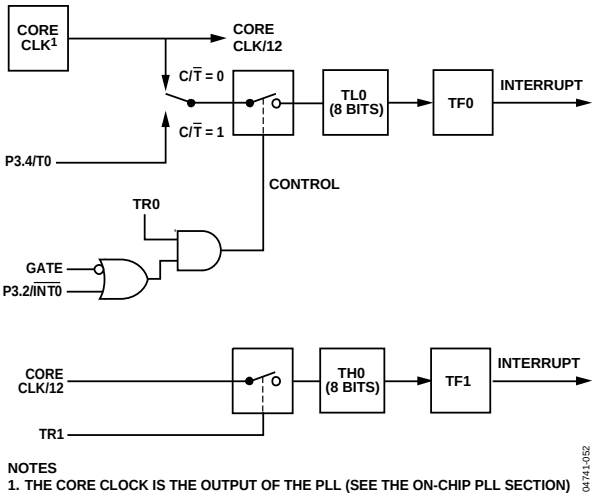


*Figure 55. Timer/Counter 0, Mode 3*

**Table 55. T3CON SFR Bit Designations**

| Bit No. | Name | Description | | | |
|---|---|---|---|---|---|
| 7 | T3BAUDEN | T3UARTBAUD Enable. | | | |
| | | 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 | | Not Implemented. Write Don't Care. | | | |
| 5 | | Not Implemented. Write Don't Care. | | | |
| 4 | | Not Implemented. Write Don't Care. | | | |
| 3 | | Not Implemented. Write Don't Care. | | | |
| 2, 1, 0 | DIV2, DIV1, DIV0 | Binary Divider | | | |
| | | DIV2 | DIV1 | DIV0 | |
| | | 0 | 0 | 0 | Binary Divider 0. See Table 57. |
| | | 0 | 0 | 1 | Binary Divider 1. See Table 57. |
| | | 0 | 1 | 0 | Binary Divider 2. See Table 57. |
| | | 0 | 1 | 1 | Binary Divider 3. See Table 57. |
| | | 1 | 0 | 0 | Binary Divider 4. See Table 57. |
| | | 1 | 0 | 1 | Binary Divider 5. See Table 57. |
| | | 1 | 1 | 0 | Binary Divider 6. See Table 57. |

### *T3FD—Timer 3 Fractional Divider Register*

See Table 57 for values.

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

**Table 56. T3FD SFR Bit Designations**

| Bit No. | Name | Description |
|---|---|---|
| 7 | ---- | Not Implemented. Write Don't Care. |
| 6 | ---- | Not Implemented. Write Don't Care. |
| 5 | T3FD.5 | Timer 3 Fractional Divider Bit 5. |
| 4 | T3FD.4 | Timer 3 Fractional Divider Bit 4. |
| 3 | T3FD.3 | Timer 3 Fractional Divider Bit 3. |
| 2 | T3FD.2 | Timer 3 Fractional Divider Bit 2. |
| 1 | T3FD.1 | Timer 3 Fractional Divider Bit 1. |
| 0 | T3FD.0 | Timer 3 Fractional Divider Bit 0. |

## INTERRUPT SYSTEM

The ADuC845/ADuC847/ADuC848 provide nine interrupt sources with two priority levels. The control and configuration of the interrupt system is carried out through three interrupt-related SFRs:

**IE** Interrupt Enable Register
**IP** Interrupt Priority Register
**IEIP2** Secondary Interrupt Enable Register

### IE—Interrupt Enable Register

SFR Address: A8H
Power-On Default: 00H
Bit Addressable: Yes

**Table 58. IE SFR Bit Designations**

| Bit No. | Name | Description |
|---|---|---|
| 7 | EA | Set by the user to enable all interrupt sources.<br>Cleared by the user to disable all interrupt sources. |
| 6 | EADC | Set by the user to enable the ADC interrupt.<br>Cleared by the user to disable the ADC interrupt. |
| 5 | ET2 | Set by the user to enable the Timer 2 interrupt.<br>Cleared by the user to disable the Timer 2 interrupt. |
| 4 | ES | Set by the user to enable the UART serial port interrupt.<br>Cleared by the user to disable the UART serial port interrupt. |
| 3 | ET1 | Set by the user to enable the Timer 1 interrupt.<br>Cleared by the user to disable the Timer 1 interrupt. |
| 2 | EX1 | Set by the user to enable External Interrupt 1 ($\overline{INT0}$).<br>Cleared by the user to disable External Interrupt 1 ($\overline{INT0}$). |
| 1 | ET0 | Set by the user to enable the Timer 0 interrupt.<br>Cleared by the user to disable the Timer 0 interrupt. |
| 0 | EX0 | Set by the user to enable External Interrupt 0 ($\overline{INT0}$).<br>Cleared by the user to disable External Interrupt 0 ($\overline{INT0}$). |

### IP—Interrupt Priority Register

SFR Address: B8H
Power-On Default: 00H
Bit Addressable: Yes

**Table 59. IP SFR Bit Designations**

| Bit No. | Name | Description |
|---|---|---|
| 7 | ----- | Not Implemented. Write Don't Care. |
| 6 | PADC | ADC Interrupt Priority (1 = High; 0 = Low). |
| 5 | PT2 | Timer 2 Interrupt Priority (1 = High; 0 = Low). |
| 4 | PS | UART Serial Port Interrupt Priority (1 = High; 0 = Low). |
| 3 | PT1 | Timer 1 Interrupt Priority (1 = High; 0 = Low). |
| 2 | PX1 | $\overline{INT0}$ (External Interrupt 1) priority (1 = High; 0 = Low). |
| 1 | PT0 | Timer 0 Interrupt Priority (1 = High; 0 = Low). |
| 0 | PX0 | $\overline{INT0}$ (External Interrupt 0) Priority (1 = High; 0 = Low). |

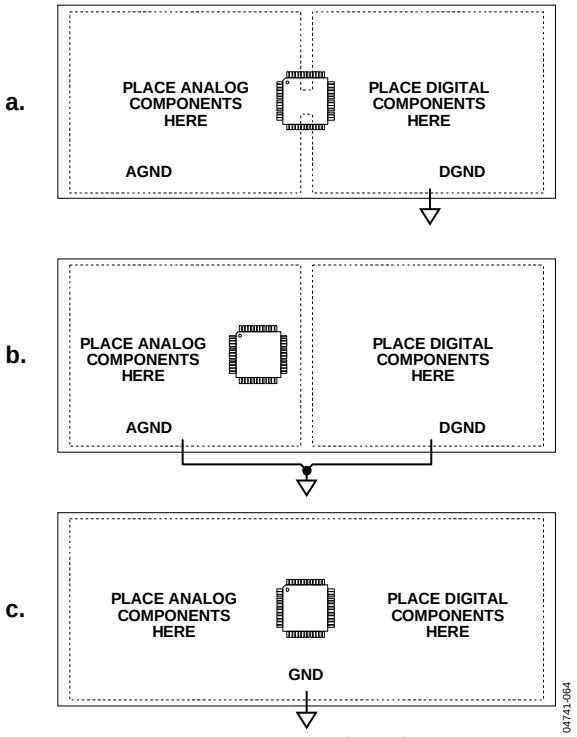


*Figure 68. System Grounding Schemes*

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 Ω or 200 Ω 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 Σ-Δ 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 Σ-Δ 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 Σ-Δ 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.

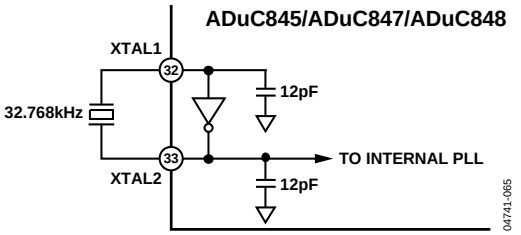


*Figure 69. Crystal Connectivity to ADuC845/ADuC847/ADuC848*

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.

**Table 68. SPI MASTER MODE TIMING (CPHA = 1) Parameter**

| | | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| $t_{SL}$ | SCLOCK Low Pulse Width[1] | | 635 | | ns |
| $t_{SH}$ | SCLOCK High Pulse Width[1] | | 635 | | ns |
| $t_{DAV}$ | Data Output Valid After SCLOCK Edge | | | 50 | ns |
| $t_{DSU}$ | Data Input Setup Time Before SCLOCK Edge | 100 | | | ns |
| $t_{DHD}$ | Data Input Hold Time After SCLOCK Edge | 100 | | | ns |
| $t_{DF}$ | Data Output Fall Time | | 10 | 25 | ns |
| $t_{DR}$ | Data Output Rise Time | | 10 | 25 | ns |
| $t_{SR}$ | SCLOCK Rise Time | | 10 | 25 | ns |
| $t_{SF}$ | SCLOCK Fall Time | | 10 | 25 | ns |

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

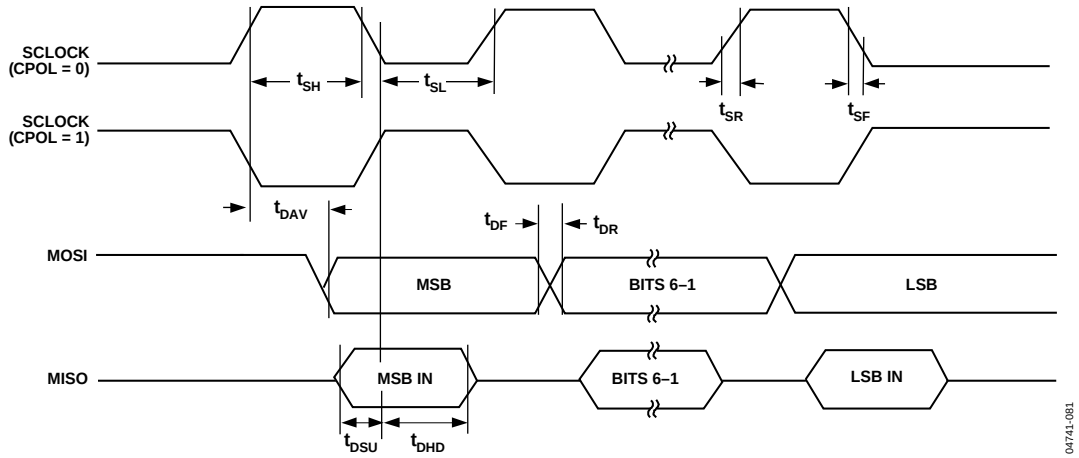


*Figure 76. SPI Master Mode Timing (CHPA = 1)*

**Table 69. SPI MASTER MODE TIMING (CPHA = 0) Parameter**

| | | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| $t_{SL}$ | SCLOCK Low Pulse Width[1] | | 635 | | ns |
| $t_{SH}$ | SCLOCK High Pulse Width[1] | | 635 | | ns |
| $t_{DAV}$ | Data Output Valid After SCLOCK Edge | | | 50 | ns |
| $t_{DOSU}$ | Data Output Setup Before SCLOCK Edge | | | 150 | ns |
| $t_{DSU}$ | Data Input Setup Time Before SCLOCK Edge | 100 | | | ns |
| $t_{DHD}$ | Data Input Hold Time After SCLOCK Edge | 100 | | | ns |
| $t_{DF}$ | Data Output Fall Time | | 10 | 25 | ns |
| $t_{DR}$ | Data Output Rise Time | | 10 | 25 | ns |
| $t_{SR}$ | SCLOCK Rise Time | | 10 | 25 | ns |
| $t_{SF}$ | SCLOCK Fall Time | | 10 | 25 | ns |

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

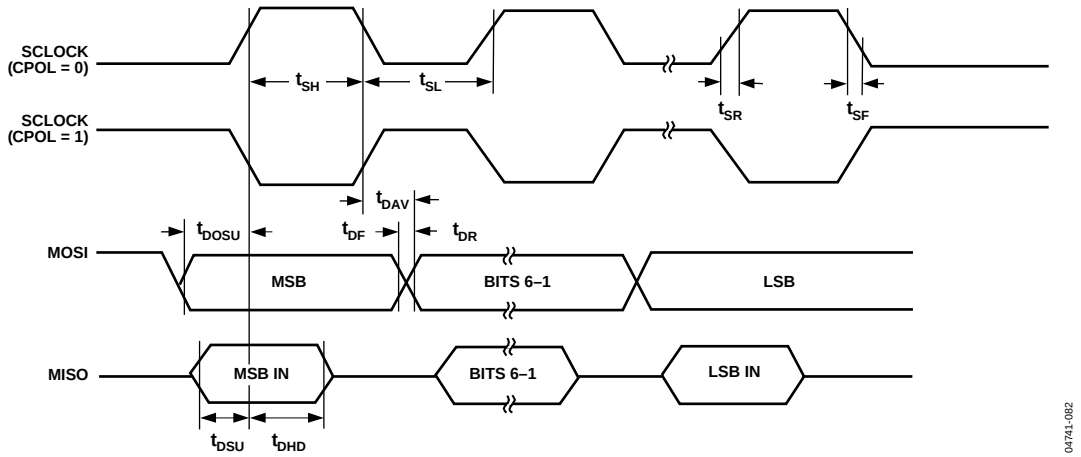


*Figure 77. SPI Master Mode Timing (CHPA = 0)*

**Table 70. SPI SLAVE MODE TIMING (CPHA = 1) Parameter**

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

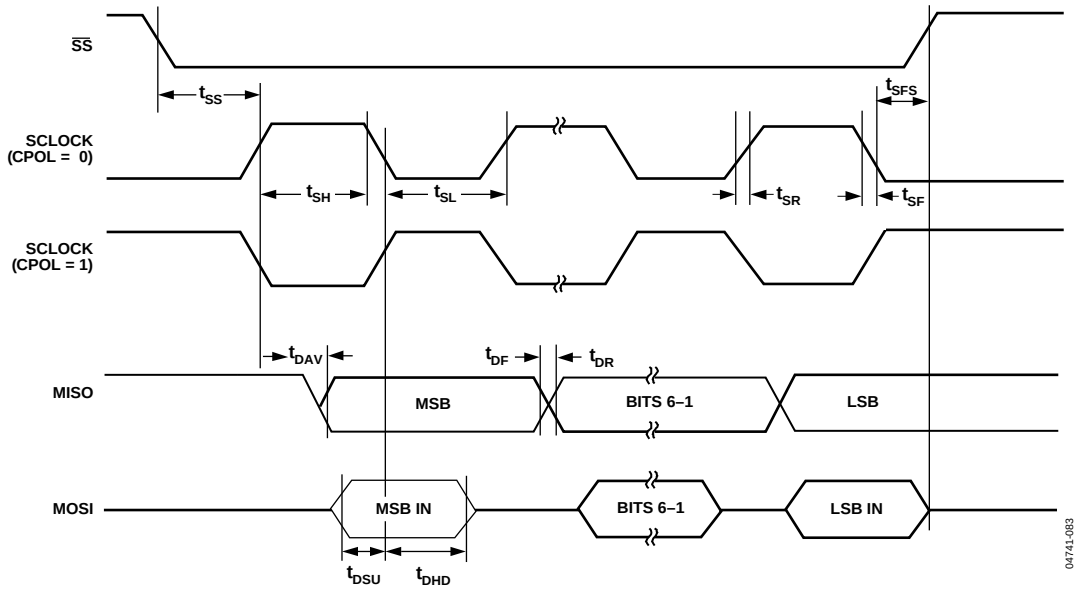


*Figure 78. SPI Slave Mode Timing (CHPA = 1)*

## ORDERING GUIDE

| Model[1, 2, 3] | Temperature Range | Package Description | Package Option |
|---|---|---|---|
| ADuC845BSZ62-5 | −40°C to +125°C | 52-Lead MQFP, Lead Free, 62-kbyte, 5 V | S-52-2 |
| ADuC845BSZ62-5-RL | −40°C to +125°C | 52-Lead MQFP, Lead Free, 62-kbyte, 5 V | S-52-2 |
| ADuC845BSZ62-3 | −40°C to +125°C | 52-Lead MQFP, Lead Free, 62-kbyte, 3 V | S-52-2 |
| ADuC845BSZ8-5 | −40°C to +125°C | 52-Lead MQFP, Lead Free, 8-kbyte, 5 V | S-52-2 |
| ADuC845BSZ8-5-RL | −40°C to +125°C | 52-Lead MQFP, Lead Free, 8-kbyte, 5 V | S-52-2 |
| ADuC845BSZ8-3 | −40°C to +125°C | 52-Lead MQFP, Lead Free, 8-kbyte, 3 V | S-52-2 |
| ADuC845BCPZ62-5 | −40°C to +85°C | 56-Lead LFCSP, Lead Free, 62-kbyte, 5 V | CP-56-11 |
| ADuC845BCPZ62-3 | −40°C to +85°C | 56-Lead LFCSP, Lead Free, 62-kbyte, 3 V | CP-56-11 |
| ADuC845BCPZ8-5 | −40°C to +85°C | 56-Lead LFCSP, Lead Free, 8-kbyte, 5 V | CP-56-11 |
| ADuC845BCPZ8-3 | −40°C to +85°C | 56-Lead LFCSP, Lead Free, 8-kbyte, 3 V | CP-56-11 |
| ADuC847BSZ62-5 | −40°C to +125°C | 52-Lead MQFP, Lead Free, 62-kbyte, 5 V | S-52-2 |
| ADuC847BSZ62-3 | −40°C to +125°C | 52-Lead MQFP, Lead Free, 62-kbyte, 3 V | S-52-2 |
| ADuC847BSZ32-5 | −40°C to +125°C | 52-Lead MQFP, Lead Free, 32-kbyte, 5 V | S-52-2 |
| ADuC847BSZ32-3 | −40°C to +125°C | 52-Lead MQFP, Lead Free, 32-kbyte, 3 V | S-52-2 |
| ADuC847BSZ8-5 | −40°C to +125°C | 52-Lead MQFP, Lead Free, 8-kbyte, 5 V | S-52-2 |
| ADuC847BSZ8-3 | −40°C to +125°C | 52-Lead MQFP, Lead Free, 8-kbyte, 3 V | S-52-2 |
| ADuC847BCPZ62-5 | −40°C to +85°C | 56-Lead LFCSP, Lead Free, 62-kbyte, 5 V | CP-56-11 |
| ADuC847BCPZ62-3 | −40°C to +85°C | 56-Lead LFCSP, Lead Free, 62-kbyte, 3 V | CP-56-11 |
| ADuC847BCPZ8-5 | −40°C to +85°C | 56-Lead LFCSP, Lead Free, 8-kbyte, 5 V | CP-56-11 |
| ADuC847BCPZ8-3 | −40°C to +85°C | 56-Lead LFCSP, Lead Free, 8-kbyte, 3 V | CP-56-11 |
| ADuC848BSZ62-5 | −40°C to +125°C | 52-Lead MQFP, Lead Free, 62-kbyte, 5 V | S-52-2 |
| ADuC848BSZ62-3 | −40°C to +125°C | 52-Lead MQFP, Lead Free, 62-kbyte, 3 V | S-52-2 |
| ADuC848BSZ32-5 | −40°C to +125°C | 52-Lead MQFP, Lead Free, 32-kbyte, 5 V | S-52-2 |
| ADuC848BSZ32-3 | −40°C to +125°C | 52-Lead MQFP, Lead Free, 32-kbyte, 3 V | S-52-2 |
| ADuC848BSZ8-5 | −40°C to +125°C | 52-Lead MQFP, Lead Free, 8-kbyte, 5 V | S-52-2 |
| ADuC848BSZ8-3 | −40°C to +125°C | 52-Lead MQFP, Lead Free, 8-kbyte, 3 V | S-52-2 |
| ADuC848BCPZ62-5 | −40°C to +85°C | 56-Lead LFCSP, Lead Free, 62-kbyte, 5 V | CP-56-11 |
| ADuC848BCPZ62-3 | −40°C to +85°C | 56-Lead LFCSP, Lead Free, 62-kbyte, 3 V | CP-56-11 |
| ADuC848BCPZ8-5 | −40°C to +85°C | 56-Lead LFCSP, Lead Free, 8-kbyte, 5 V | CP-56-11 |
| ADuC848BCPZ8-3 | −40°C to +85°C | 56-Lead LFCSP, Lead Free, 8-kbyte, 3 V | CP-56-11 |
| EVAL-ADuC845QSZ | | QuickStart Development System | |
| EVAL-ADuC845QSPZ | | QuickStart-PLUS Development System | |
| EVAL-ADuC847QSZ | | QuickStart Development System | |
| EVAL-ADUC-CABLE1Z | | ADuC Serial Downloader Cable for UART | |

[1] The -3 and -5 in the Model column indicate the $DV_{DD}$ operating voltage.
[2] Z = RoHS Compliant Part.
[3] The QuickStart Plus system can only be ordered directly from Accutron. It can be purchased from the website http://www.accutron.com.