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-rl

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 Σ-Δ 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

Parameter	Min	Typ	Max	Unit	Test Conditions/Comments
AUXILIARY ADC ANALOG INPUTS (ADuC845 ONLY)					
Differential Input Voltage Ranges[5,6]					
Bipolar Mode (ADC1CON.5 = 0)		$\pm V_{REF}$		V	REFIN = REFIN(+) − REFIN(−) (or Int 1.25 V_{REF})
Unipolar Mode (ADC1CON.5 = 1)		0 − V_{REF}		V	REFIN = REFIN(+) − REFIN(−) (or Int 1.25 V_{REF})
Average Analog Input Current		125		nA/V	
Analog Input Current Drift		±2		pA/V/°C	
Absolute AIN/AINCOM Voltage Limits[2,7]	A_{GND} − 0.03		AV_{DD} + 0.03	V	
Normal Mode Rejection 50 Hz/60 Hz[2]					
On AIN and REFIN	75			dB	50 Hz/60 Hz ± 1 Hz, 16.6 Hz Fadc, SF = 52H, chop on, REJ60 on
	100			dB	50 Hz ± 1 Hz, 16.6 Hz Fadc, SF = 52H, chop on
	67			dB	50 Hz/60 Hz ± 1 Hz, 50 Hz Fadc, SF = 52H, chop off, REJ60 on
	100			dB	50 Hz ± 1 Hz, 50 Hz Fadc, SF = 52H, chop off
ADC SYSTEM CALIBRATION					
Full-Scale Calibration Limit			+1.05 × FS	V	
Zero-Scale Calibration Limit	−1.05 × FS			V	
Input Span	0.8 × FS		2.1 × FS	V	
DAC					
Voltage Range		0 − V_{REF}		V	DACCON.2 = 0
		0 − AV_{DD}		V	DACCON.2 = 1
Resistive Load		10		kΩ	From DAC output to AGND
Capactive Load		100		pF	From DAC output to AGND
Output Impedance		0.5		Ω	
I_{SINK}		50		µA	
DC Specifications[8]					
Resolution	12			Bits	
Relative Accuracy		±3		LSB	
Differential Nonlinearity			−1	LSB	Guaranteed 12-bit monotonic
Offset Error			±50	mV	
Gain Error			±1	%	AV_{DD} range
		±1		%	V_{REF} range
AC Specifications[2,8]					
Voltage Output Settling Time		15		µs	Settling time to 1 LSB of final value
Digital-to-Analog Glitch Energy		10		nVs	1 LSB change at major carry
INTERNAL REFERENCE					
ADC Reference					Chop enabled
Reference Voltage	1.25 − 1%	1.25	1.25 + 1%	V	Initial tolerance @ 25°C, V_{DD} = 5 V
Power Supply Rejection		45		dB	
Reference Tempco		100		ppm/°C	
DAC Reference					
Reference Voltage	2.5 − 1%	2.5	2.5 + 1%	±1% V	Initial tolerance @ 25°C, V_{DD} = 5 V
Power Supply Rejection		50		dB	
Reference Tempco		±100		ppm/°C	
TEMPERATURE SENSOR (ADuC845 ONLY)					
Accuracy		±2		°C	
Thermal Impedance		90		°C/W	MQFP
		52		°C/W	LFCSP

Pin No.				
52-MQFP	56-LFCSP	Mnemonic	Type[1]	Description
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.

[1] I = input, S = supply, I/O means input/output, and O = output.

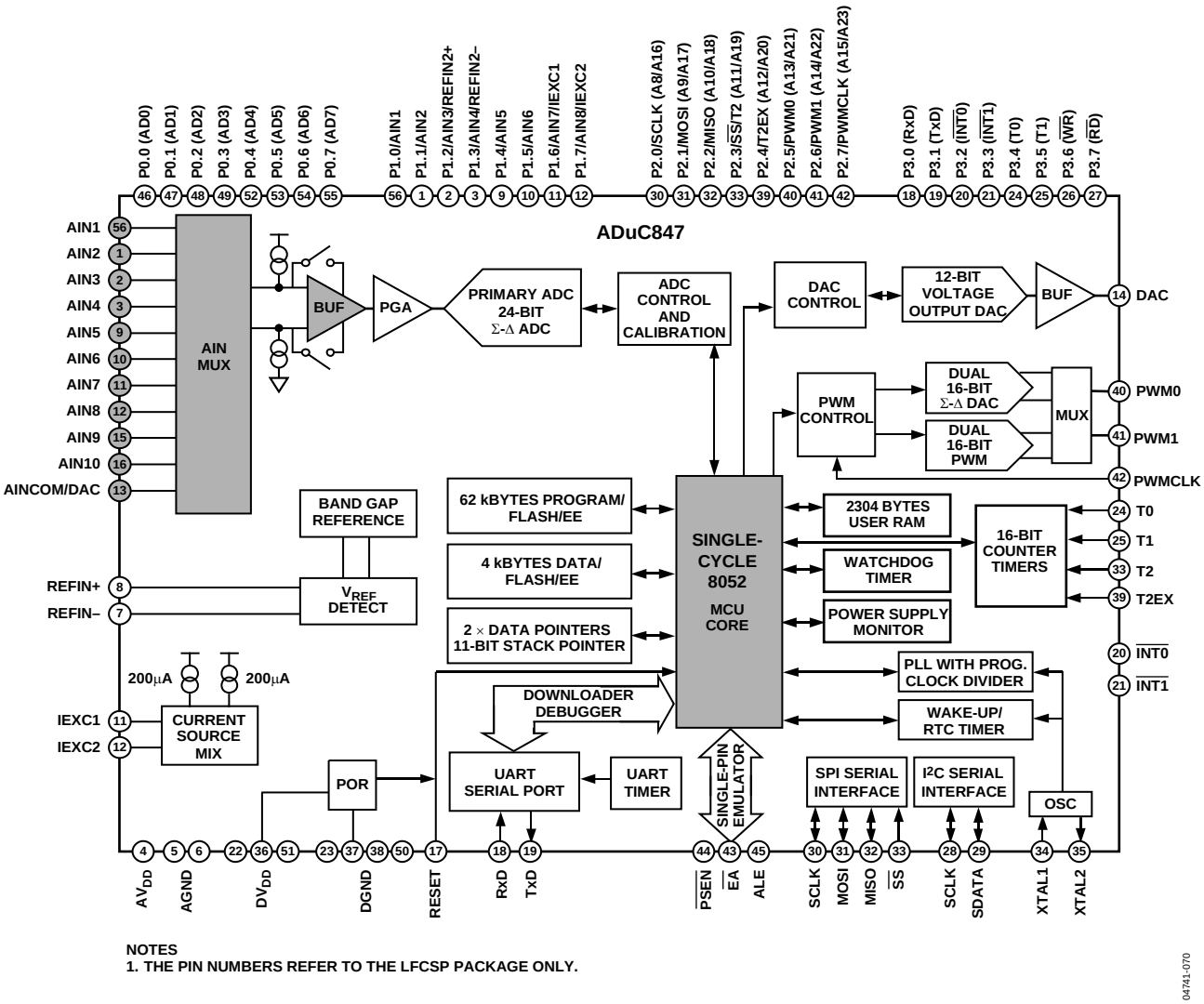

Figure 5. Detailed Block Diagram of the ADuC847

Mnemonic	Description	Bytes	Cycles[1]
SJMP rel	Short jump (relative address)	2	3
JC rel	Jump on carry = 1	2	3
JNC rel	Jump on carry = 0	2	3
JZ rel	Jump on accumulator = 0	2	3
JNZ rel	Jump on accumulator ! = 0	2	3
DJNZ Rn,rel	Decrement register, JNZ relative	2	3
LJMP	Long jump unconditional	3	4
LCALL[3] addr16	Long jump to subroutine	3	4
JB bit,rel	Jump on direct bit = 1	3	4
JNB bit,rel	Jump on direct bit = 0	3	4
JBC bit,rel	Jump on direct bit = 1 and clear	3	4
CJNE A,dir,rel	Compare A, direct JNE relative	3	4
CJNE A,#data,rel	Compare A, immediate JNE relative	3	4
CJNE Rn,#data,rel	Compare register, immediate JNE relative	3	4
CJNE @Ri,#data,rel	Compare indirect, immediate JNE relative	3	4
DJNZ dir,rel	Decrement direct byte, JNZ relative	3	4
Miscellaneous			
NOP	No operation	1	1

[1] One cycle is one clock.
[2] MOVX instructions are four cycles when they have 0 wait state. Cycles of MOVX instructions are 4 + *n* cycles when they have *n* wait states as programmed via EWAIT.
[3] LCALL instructions are three cycles when the LCALL instruction comes from an interrupt.

MEMORY ORGANIZATION

The ADuC845, ADuC847, and ADuC848 contain four memory blocks:

- 62 kbytes/32 kbytes/8 kbytes of on-chip Flash/EE program memory
- 4 kbytes of on-chip Flash/EE data memory
- 256 bytes of general-purpose RAM
- 2 kbytes of internal XRAM

Flash/EE Program Memory

The devices provide up to 62 kbytes of Flash/EE program memory to run user code. All further references to Flash/EE program memory assume the 62-kbyte option.

When $\overline{EA}$ is pulled high externally during a power cycle or a hardware reset, the devices default to code execution from their internal 62 kbytes of Flash/EE program memory. The devices do not support the rollover from internal code space to external code space. No external code space is available on the devices. Permanently embedded firmware allows code to be serially downloaded to the 62 kbytes of internal code space via the UART serial port while the device is in-circuit. No external hardware is required.

During run time, 56 kbytes of the 62-kbyte program memory can be reprogrammed. This means that the code space can be upgraded in the field by using a user-defined protocol running on the devices, or it can be used as a data memory. For details, see the Nonvolatile Flash/EE Memory Overview section.

Flash/EE Data Memory

The user has 4 kbytes of Flash/EE data memory available that can be accessed indirectly by using a group of registers mapped into the special function register (SFR) space. For details, see the Nonvolatile Flash/EE Memory Overview section.

General-Purpose RAM

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

The lower 128 bytes of internal data memory are mapped as shown in Figure 8. The lowest 32 bytes are grouped into four banks of eight registers addressed as R0 to R7. The next 16 bytes (128 bits), locations 20H to 2FH above the register banks, form a block of directly addressable bit locations at Bit Addresses 00H to 7FH. The stack can be located anywhere in the internal memory address space, and the stack depth can be expanded up to 2048 bytes.

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

This offset is removed by performing a running average of 2. This average by 2 means that the settling time to any change in programming of the ADC is twice the normal conversion time, while an asynchronous step change on the analog input is not fully reflected until the third subsequent output. See Figure 13.

$$t_{SETTLE} = \frac{2}{f_{ADC}} = 2 \times t_{ADC}$$

The allowable range for SF (chop enabled) is 13 to 255 with a default of 69 (45H). The corresponding conversion rates, rms and peak-to-peak noise performances are shown in Table 10, Table 11, Table 12, and Table 13. The numbers are typical and generated at a differential input voltage of 0 V and a common-mode voltage of 2.5 V. Note that the conversion time increases by 0.732 ms for each increment in SF.

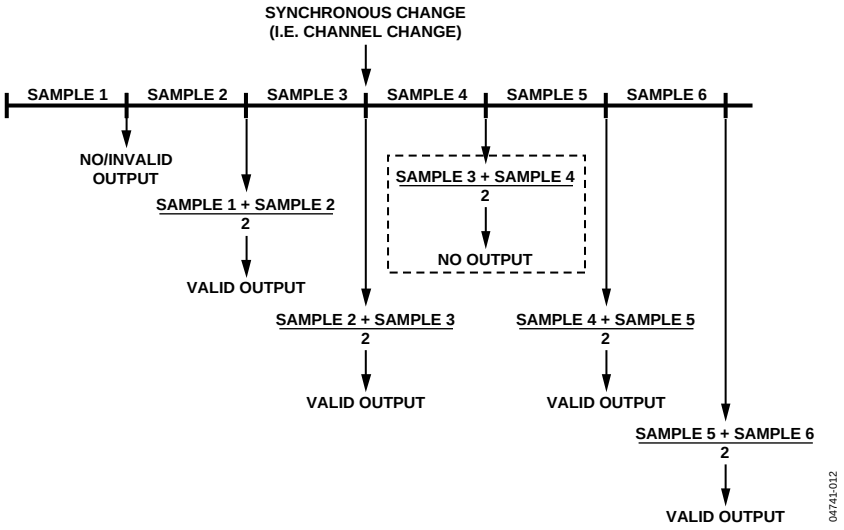

Figure 13. ADC Settling Time Following a Synchronous Change with Chop Enabled

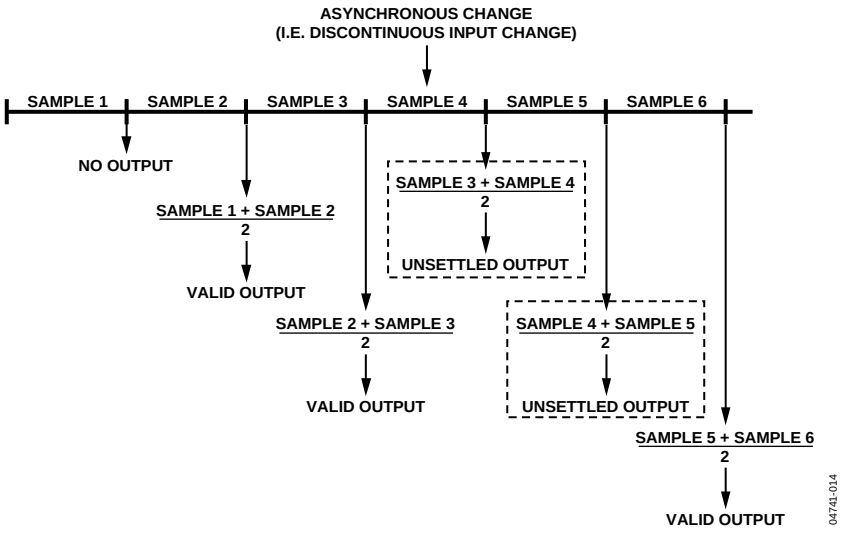

Figure 14. ADC Settling Time Following an Asynchronous Change with Chop Enabled

(see Table 30). These burnout current sources are also available only with buffering enabled via the BUF0/BUF1 bits in the ADC0CON1 SFR. Once the burnout currents are turned on, a current flows in the external transducer circuit, and a measurement of the input voltage on the analog input channel can be taken. When the resulting voltage measured is full scale, the transducer has gone open circuit. When the voltage measured is 0 V, this indicates that the transducer has gone short circuit. The current sources work over the normal absolute input voltage range specifications.

REFERENCE DETECT CIRCUIT

The main and auxiliary (ADuC845 only) ADCs can be configured to allow the use of the internal band gap reference or an external reference that is applied to the REFIN± pins by means of the XREF0/1 bit in the Control Registers AD0CON2 and AD1CON (ADuC845 only). A reference detection circuit is provided to detect whether a valid voltage is applied to the REFIN± pins. This feature arose in connection with strain-gage sensors in weigh scales where the reference and signal are provided via a cable from the remote sensor. It is desirable to detect whether the cable is disconnected. If either of the pins is floating or if the applied voltage is below a specified threshold, a flag (NOXREF) is set in the ADC status register (ADCSTAT), conversion results are clamped, and calibration registers are not updated if a calibration is in progress.

Note that the reference detect does not look at REFIN2± pins.

If, during either an offset or gain calibration, the NOEXREF bit becomes active, indicating an incorrect V_{REF}, updating the relevant calibration register is inhibited to avoid loading incorrect data into these registers, and the appropriate bits in ADCSTAT (ERR0 or ERR1) are set. If the user needs to verify that a valid reference is in place every time a calibration is performed, the status of the ERR0 and ERR1 bits should be checked at the end of every calibration cycle.

SINC FILTER REGISTER (SF)

The number entered into the SF register sets the decimation factor of the Sinc^3 filter for the ADC. See Table 28 and Table 29.

The range of operation of the SF word depends on whether ADC chop is on or off. With chop disabled, the minimum SF word is 3 and the maximum is 255. This gives an ADC throughput rate from 16.06 Hz to 1.365 kHz. With chop enabled, the minimum SF word is 13 (all values lower than 13 are clamped to 13) and the maximum is 255. This gives an ADC throughput rate of 5.4 Hz to 105 Hz. See the f_{ADC} equation in the ADC description preceding section.

An additional feature of the Sinc^3 filter is a second notch filter positioned in the frequency response at 60 Hz. This gives simultaneous 60 Hz rejection to whatever notch is defined by the SF filter. This 60 Hz filter is enabled via the REJ60 bit in the ADCMODE register (ADCMODE.6). The notch is valid only for SF words ≥ 68; otherwise, ADC errors occur, and, the notch is best used with an SF word of 82d giving simultaneous 50 Hz and 60 Hz rejection. This function is useful only with an ADC clock (modulator rate) of 32.768 kHz. During calibration, the current (user-written) value of the SF register is used.

Σ-Δ MODULATOR

A Σ-Δ ADC usually consists of two main blocks, an analog modulator, and a digital filter. For the ADuC845/ADuC847/ADuC848, the analog modulator consists of a difference amplifier, an integrator block, a comparator, and a feedback DAC as shown in Figure 16.

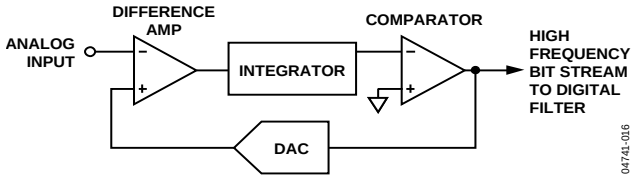

Figure 16. Σ-Δ Modulator Simplified Block Diagram

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

DIGITAL FILTER

The output of the ∑-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 device.

The ADuC845/ADuC847/ADuC848 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 SF (Sinc filter) SFR as listed in Table 28 and Table 29.

Figure 22, Figure 23, Figure 24, and Figure 25 show the frequency response of the ADC, yielding an overall output rate of 16.6 Hz with chop enabled and 50 Hz with chop disabled. Also detailed in these plots is the effect of the fixed 60 Hz drop-in notch filter

ADC1CON (AUXILIARY ADC CONTROL REGISTER) (ADUC845 ONLY)

ADC1CON is used to configure the auxiliary ADC for reference, channel selection, and unipolar or bipolar coding. The auxiliary ADC is available only on the ADuC845.

SFR Address: D3H
Power-On Default: 00H
Bit Addressable: No

Table 27. ADC1CON SFR Bit Designations

Bit No.	Name	Description
7	–––	Not Implemented. Write Don't Care.
6	AXREF	Auxiliary (ADuC845 only) ADC External Reference Bit. Set by the user to enable the auxiliary ADC to use the external reference via REFIN±. Cleared by the user to enable the auxiliary ADC to use the internal band gap reference. Auxiliary ADC cannot use the REFIN2± reference inputs.
5	AUNI	Auxiliary (ADuC845 only) ADC Unipolar Bit. Set by the user to enable unipolar coding, that is, zero input results in 000000H output. Cleared by the user to enable bipolar coding, zero input results in 800000H output.
4	–––	Not Implemented. Write Don't Care.
3, 2, 1, 0	ACH3, ACH2, ACH1, ACH0	Auxiliary ADC Channel Select Bits. Written by the user to select the auxiliary ADC channel.

ACH3	ACH2	ACH1	ACH0	Selected Auxiliary ADC Input Range (V_{REF} = 2.5 V).
0	0	0	0	AIN1–AINCOM
0	0	0	1	AIN2–AINCOM
0	0	1	0	AIN3–AINCOM
0	0	1	1	AIN4–AINCOM
0	1	0	0	AIN5–AINCOM
0	1	0	1	AIN6–AINCOM
0	1	1	0	AIN7–AINCOM
0	1	1	1	AIN8–AINCOM
1	0	0	0	AIN9–AINCOM (not a valid selection on the MQFP package)
1	0	0	1	AIN10–AINCOM (not a valid selection on the MQFP package)
1	0	1	0	AIN1–AIN2
1	0	1	1	AIN3–AIN4
1	1	0	0	AIN5–AIN6
1	1	0	1	AIN7–AIN8
1	1	1	0	Temperature Sensor[1]
1	1	1	1	AINCOM–AINCOM

[1] Note the following about the temperature sensor:
When the temperature sensor is selected, user code must select the internal reference via the AXREF bit and clear the AUNI bit (ADC1CON.5) to select bipolar coding.
Chop mode must be enabled for correct temperature sensor operation.
The temperature sensor is factory calibrated to yield conversion results 800000H at 0°C (ADC chop on).
A +1°C change in temperature results in a +1 LSB change in the ADC1H register ADC conversion result
The temperature sensor is not available on the ADuC847 or ADuC848.

NONVOLATILE FLASH/EE MEMORY OVERVIEW

The ADuC845/ADuC847/ADuC848 incorporate Flash/EE memory technology on-chip to provide the user with nonvolatile, in-circuit reprogrammable code and data memory space.

Like EEPROM, flash memory can be programmed in-system at the byte level, although it must first be erased, in page blocks. Thus, flash memory is often and more correctly referred to as Flash/EE memory.

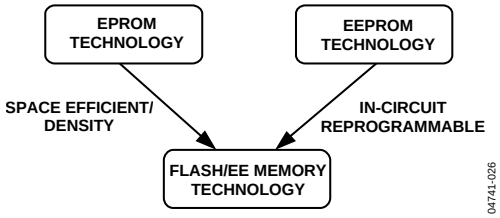

Figure 26. Flash/EE Memory Development

Overall, Flash/EE memory represents a step closer to the ideal memory device that includes nonvolatility, in-circuit programmability, high density, and low cost. The Flash/EE memory technology allows the user to update program code space in-circuit, without needing to replace onetime programmable (OTP) devices at remote operating nodes.

Flash/EE Memory on the ADuC845, ADuC847, ADuC848

The ADuC845/ADuC847/ADuC848 provide two arrays of Flash/EE memory for user applications—up to 62 kbytes of Flash/EE program space and 4 kbytes of Flash/EE data memory space. Also, 8-kbyte and 32-kbyte program memory options are available. All examples and references in this datasheet use the 62-kbyte option; however, similar protocols and procedures are applicable to the 32-kbyte and 8-kbyte options unless otherwise noted, provided that the difference in memory size is taken into account.

The 62 kbytes Flash/EE code space are provided on-chip to facilitate code execution without any external discrete ROM device requirements. The program memory can be programmed in-circuit, using the serial download mode provided, using conventional third party memory programmers, or via any user-defined protocol in user download (ULOAD) mode.

The 4-kbyte Flash/EE data memory space can be used as a general-purpose, nonvolatile scratchpad area. User access to this area is via a group of seven SFRs. This space can be programmed at a byte level, although it must first be erased in 4-byte pages.

All the following sections use the 62-kbyte program space as an example when referring to program and ULOAD mode. For the 64-kbyte part, the ULOAD area takes up the top 6 kbytes of the program space, that is, from 56 kbytes to 62 kbytes. For the 32-kbyte part, the ULOAD space moves to the top 8 kbytes of the on-chip program memory, that is., from 24 kbytes to 32 kbytes.

No ULOAD mode is available on the 8-kbyte part since the bootload area on the 8-kbyte part is 8 kbytes long, so no usable user program space remains. The kernel still resides in the protected area from 62 kbytes to 64 kbytes.

Flash/EE Memory Reliability

The Flash/EE program and data memory arrays on the ADuC845/ADuC847/ADuC848 are fully qualified for two key Flash/EE memory characteristics: Flash/EE memory cycling endurance and Flash/EE memory data retention.

Endurance quantifies the ability of the Flash/EE memory to be cycled through many program, read, and erase cycles. In real terms, a single endurance cycle is composed of four independent, sequential events:

1. Initial page erase sequence
2. Read/verify sequence
3. Byte program sequence
4. Second read/verify sequence

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

As indicated in the Specifications table, the ADuC845/ADuC847/ ADuC848 Flash/EE memory endurance qualification has been carried out in accordance with JEDEC Specification A117 over the industrial temperature range of –40°C, +25°C, +85°C, and +125°C. (The LFCSP package is qualified to +85°C only.) The results allow the specification of a minimum endurance figure over supply and temperature of 100,000 cycles, with an endurance figure of 700,000 cycles being typical of operation at 25°C.

Retention is the ability of the Flash/EE memory to retain its programmed data over time. Again, the devices have been qualified in accordance with the formal JEDEC Retention Lifetime Specification (A117) at a specific junction temperature (T_J = 55°C). As part of this qualification procedure, the Flash/EE memory is cycled to its specified endurance limit described previously, before data retention is characterized. This means that the Flash/EE memory is guaranteed to retain its data for its full specified retention lifetime every time the Flash/EE memory is reprogrammed. It should also be noted that retention lifetime, based on an activation energy of 0.6 eV, derates with T_J as shown in Figure 27.

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.

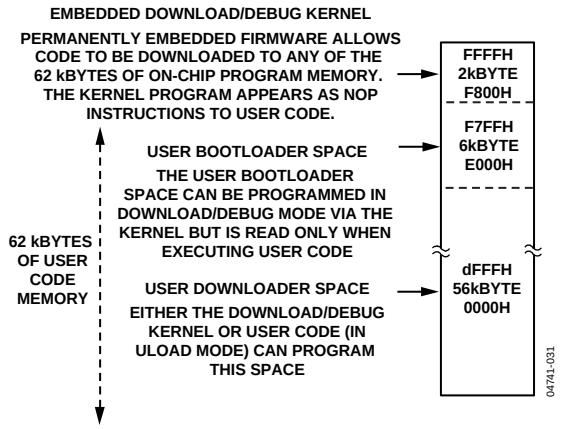

Figure 30. Flash/EE Program Memory Map in ULOAD Mode (62-kbyte Part)

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

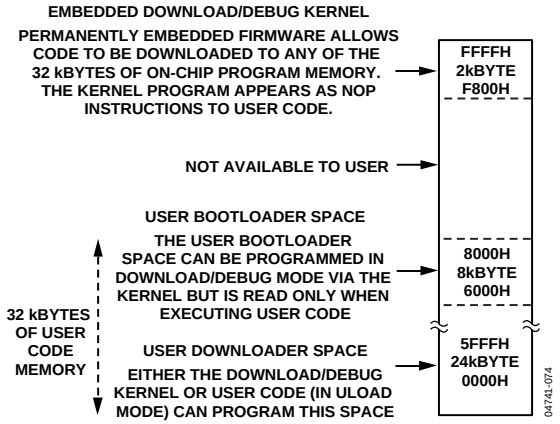

Figure 31. Flash/EE Program Memory Map in ULOAD Mode (32-kbyte Part)

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.

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. If low, the 32 kHz crystal oscillator continues running in power-down mode. If high, the 32.768 kHz oscillator is powered down. 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. 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. 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. Set by the user to enable the response to any interrupt to be executed at the fastest core clock frequency. Cleared by the user to disable the fast interrupt response feature. 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. CD2 CD1 CD0 Core Clock Frequency (MHz) 0 0 0 12.582912. Not a valid selection on 3 V parts. 0 0 1 6.291456 (Maximum core clock rate allowed on the 3 V parts) 0 1 0 3.145728 0 1 1 1.572864 (Default core frequency) 1 0 0 0.786432 1 0 1 0.393216 1 1 0 0.196608 1 1 1 0.098304 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. 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.

Hardware Slave Mode

After reset, the ADuC845/ADuC847/ADuC848 default to hardware slave mode. Slave mode is enabled by clearing the I2CM bit in I2CCON. The devices have a full hardware slave. In slave mode, the I²C address is stored in the I2CADD register. Data received or to be transmitted is stored in the I2CDAT register.

Once enabled in I²C slave mode, the slave controller waits for a start condition. If the parts detect a valid start condition, followed by a valid address, followed by the R/W bit, then the I2CI interrupt bit is automatically set by hardware. The I²C peripheral generates a core interrupt only if the user has pre-configured the I²C interrupt enable bit in the IEIP2 SFR as well as the global interrupt bit, EA, in the IE SFR. Therefore,

```
MOV IEIP2, #01h    ;Enable I²C Interrupt
SETB EA
```

An autoclear of the I2CI bit is implemented on the devices so that this bit is cleared automatically upon read or write access to the I2CDAT SFR.

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

If for any reason the user tries to clear the interrupt more than once, that is, access the data SFR more than once per interrupt, the I²C controller stops. The interface then must be reset by using the I2CRS bit.

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

The I2CTX bit contains the R/W bit sent from the master. If I2CTX is set, the master is ready to receive a byte; therefore the slave transmits data by writing to the I2CDAT register. If I2CTX is cleared, the master is ready to transmit a byte; therefore the slave receives a serial byte. Software can interrogate the state of I2CTX to determine whether it should write to or read from I2CDAT.

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

The I2CI interrupt bit is set every time a complete data byte is received or transmitted, provided that it is followed by a valid ACK. If the byte is followed by a NACK, an interrupt is not generated.

The device continues to issue interrupts for each complete data byte transferred until a stop condition is received or the interface is reset.

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

SPI SERIAL INTERFACE

The ADuC845/ADuC847/ADuC848 integrate a complete hardware serial peripheral interface (SPI) interface on-chip. SPI is an industry-standard synchronous serial interface that allows 8 bits of data to be synchronously transmitted and received simultaneously, that is, full duplex. Note that the SPI pins are multiplexed with the Port 2 pins, P2.0, P2.1, P2.2, and P2.3. These pins have SPI functionality only if SPE is set. Otherwise, with SPE cleared, standard Port 2 functionality is maintained. SPI can be configured for master or slave operation and typically consists of Pins SCLOCK, MISO, MOSI, and $\overline{SS}$.

SCLOCK (Serial Clock I/O Pin)

Pin 28 (MQFP Package), Pin 30 (LFCSP Package)
The master clock (SCLOCK) is used to synchronize the data transmitted and received through the MOSI and MISO data lines.

A single data bit is transmitted and received in each SCLOCK period. Therefore, a byte is transmitted/received after eight SCLOCK periods. The SCLOCK pin is configured as an output in master mode and as an input in slave mode. In master mode, the bit rate, polarity, and phase of the clock are controlled by the CPOL, CPHA, SPR0, and SPR1 bits in the SPICON SFR (see Table 41). In slave mode, the SPICON register must be configured with the same phase and polarity (CPHA and CPOL) as the master. The data is transmitted on one edge of the SCLOCK signal and sampled on the other.

MISO (Master In, Slave Out Pin)

Pin 30 (MQFP Package), Pin 32 (LFCSP Package)
The MISO pin is configured as an input line in master mode and an output line in slave mode. The MISO line on the master (data in) should be connected to the MISO line in the slave device (data out). The data is transferred as byte-wide (8-bit) serial data, MSB first.

MOSI (Master Out, Slave In Pin)

Pin 29 (MQFP Package), Pin31 (LFCSP Package)
The MOSI pin is configured as an output line in master mode and an input line in slave mode. The MOSI line on the master (data out) should be connected to the MOSI line in the slave device (data in). The data is transferred as byte-wide (8-bit) serial data, MSB first.

TIMECON—TIC Control Register

SFR Address: A1H
Power-On Default: 00H
Bit Addressable: No

Table 45. TIMECON SFR Bit Designations

Bit No.	Name	Description
7	----	Not Implemented. Write Don't Care.
6	TFH	Twenty-Four Hour Select Bit. Set by the user to enable the hour counter to count from 0 to 23. Cleared by the user to enable the hour counter to count from 0 to 255.
5, 4	ITS1, ITS0	Interval Timebase Selection Bits. ITS1 ITS0 Interval Timebase 0 0 1/128 Second 0 1 Seconds 1 0 Minutes 1 1 Hours
3	ST1	Single Time Interval Bit. Set by the user to generate a single interval timeout. If set, a timeout clears the TIEN bit. Cleared by the user to allow the interval counter to be automatically reloaded and start counting again at each interval timeout.
2	TII	TIC Interrupt Bit. Set when the 8-bit interval counter matches the value in the INTVAL SFR. Cleared by user software.
1	TIEN	Time Interval Enable Bit. Set by the user to enable the 8-bit time interval counter. Cleared by the user to disable the interval counter.
0	TCEN	Time Clock Enable Bit. Set by the user to enable the time clock to the time interval counters. Cleared by the user to disable the clock to the time interval counters and reset the time interval SFRs to the last value written to them by the user. The time registers (HTHSEC, SEC, MIN, and HOUR) can be written while TCEN is low.

8052-COMPATIBLE ON-CHIP PERIPHERALS

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

Parallel I/O

The ADuC845/ADuC847/ADuC848 use four input/output ports to exchange data with external devices. In addition to performing general-purpose I/O, some are capable of external memory operations, while others are multiplexed with alternate functions for the peripheral functions available on-chip. In general, when a peripheral is enabled, that pin cannot be used as a general-purpose I/O pin.

Port 0

Port 0 is an 8-bit open-drain bidirectional I/O port that is directly controlled via the Port 0 SFR (80H). Port 0 is also the multiplexed low-order address and data bus during accesses to external data memory.

Figure 48 shows a typical bit latch and I/O buffer for a Port 0 pin. The bit latch (one bit in the SFRof the port) is represented as a Type D flip-flop, which clocks in a value from the internal bus in response to a write to latch signal from the CPU. The Q output of the flip-flop is placed on the internal bus in response to a read latch signal from the CPU. The level of the port pin itself is placed on the internal bus in response to a read pin signal from the CPU. Some instructions that read a port activate the read latch signal, and others activate the read pin signal. See the Read-Modify-Write Instructions section for details.

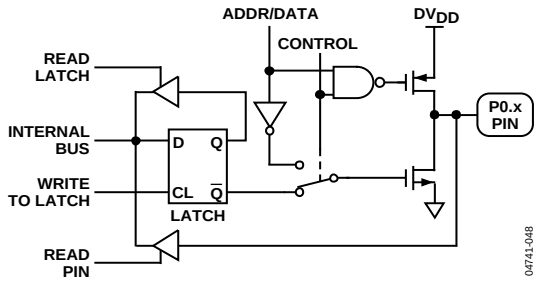

Figure 48. Port 0 Bit Latch and I/O Buffer

As shown in Figure 48, the output drivers of Port 0 pins are switchable to an internal ADDR and ADDR/DATA bus by an internal control signal for use in external memory accesses. During external memory accesses, the P0 SFR has 1s written to it; therefore, all its bit latches become 1. When accessing external memory, the control signal in Figure 48 goes high, enabling push-pull operation of the output pin from the internal address or data bus (ADDR/DATA line). Therefore, no external pull-ups are required on Port 0 for it to access external memory.

In general-purpose I/O port mode, Port 0 pins that have 1s written to them via the Port 0 SFR are configured as open-drain and, therefore, float. In this state, Port 0 pins can be used as high impedance inputs. This is represented in Figure 48 by the NAND gate whose output remains high as long as the control signal is low, thereby disabling the top FET. External pull-up resistors are, therefore, required when Port 0 pins are used as general-purpose outputs. Port 0 pins with 0s written to them drive a logic low output voltage (V_{OL}) and are capable of sinking 1.6 mA.

Port 1

Port 1 is also an 8-bit port directly controlled via the P1 SFR (90H). Port 1 digital output capability is not supported on this device. Port 1 pins can be configured as digital inputs or analog inputs. By (power-on) default, these pins are configured as analog inputs, that is, 1 is written to the corresponding Port 1 register bit. To configure any of these pins as digital inputs, the user should write a 0 to these port bits to configure the corresponding pin as a high impedance digital input. These pins also have various secondary functions aside from their analog input capability, as described in Table 46.

Table 46. Port 1 Alternate Functions

Pin No.	Alternate Function
P1.2	REFIN2+ (second reference input, postive)
P1.3	REFIN2− (second reference input, negative)
P1.6	IEXC1 (200 µA excitation current source)
P1.7	IEXC2 (200 µA excitation current source)

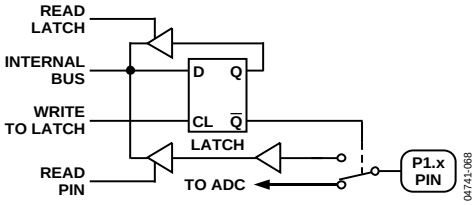

Figure 49. Port 1 Bit Latch and I/O Buffer

Port 2

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

In general-purpose I/O port mode, Port 2 pins that have 1s written to them are pulled high by the internal pull-ups as shown in Figure 50 and, in that state, can be used as inputs. As inputs, Port 2 pins pulled externally low source current because of the internal pull-up resistors. Port 2 pins with 0s written to them drive a logic low output voltage (V_{OL}) and are capable of sinking 1.6 mA.

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.

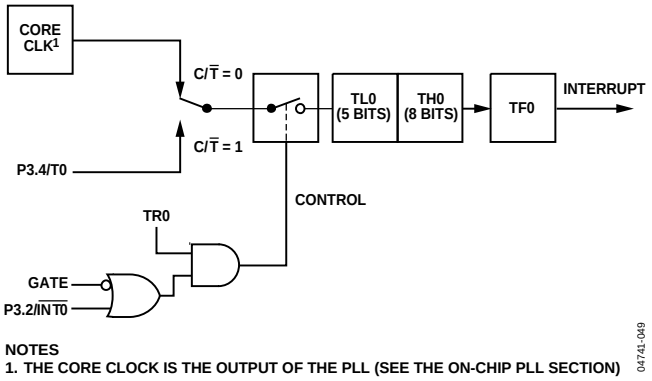

Figure 52. Timer/Counter 0, Mode 0

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.

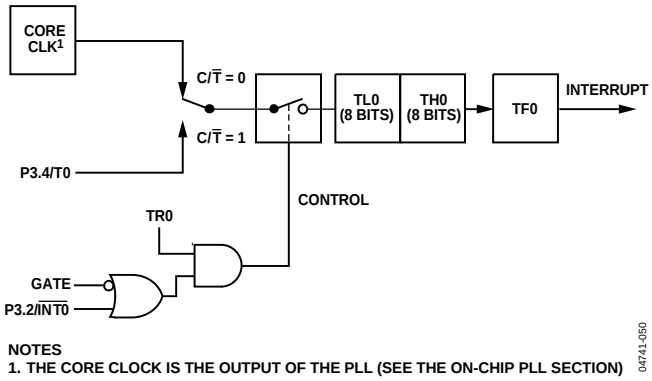

Figure 53. Timer/Counter 0, Mode 1

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.

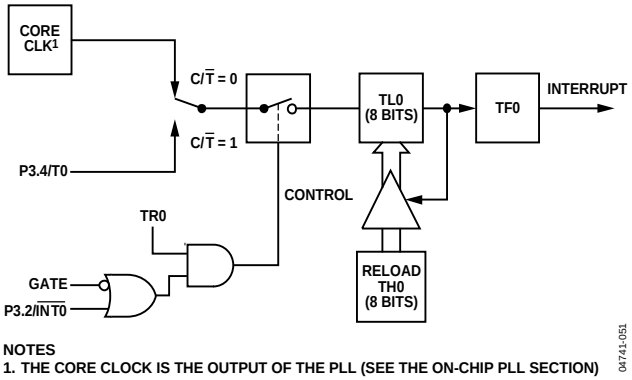

Figure 54. Timer/Counter 0, Mode 2

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.

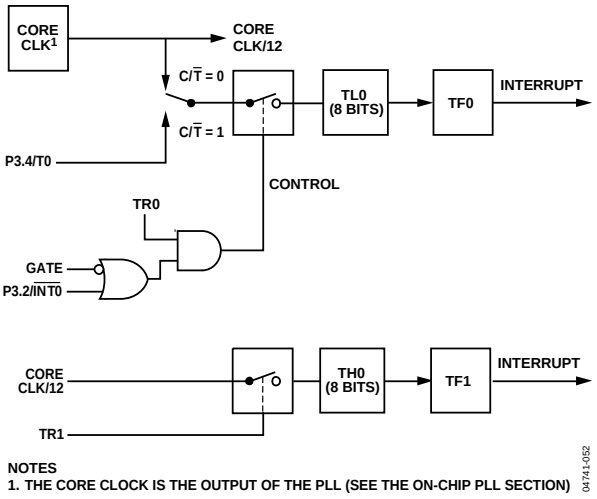

Figure 55. Timer/Counter 0, Mode 3

T2CON—Timer/Counter 2 Control Register

SFR Address: C8H
Power-On Default: 00H
Bit Addressable: Yes

Table 52. T2CON SFR Bit Designations

Bit No.	Name	Description
7	TF2	Timer 2 Overflow Flag. Set by hardware on a Timer 2 overflow. TF2 cannot be set when either RCLK = 1 or TCLK = 1. Cleared by user software.
6	EXF2	Timer 2 External Flag. Set by hardware when either a capture or reload is caused by a negative transition on T2EX and EXEN2 = 1. Cleared by user software.
5	RCLK	Receive Clock Enable Bit. Set by the user to enable the serial port to use Timer 2 overflow pulses for its receive clock in serial port Modes 1 and 3. Cleared by the user to enable Timer 1 overflow to be used for the receive clock.
4	TCLK	Transmit Clock Enable Bit. Set by the user to enable the serial port to use Timer 2 overflow pulses for its transmit clock in serial port Modes 1 and 3. Cleared by the user to enable Timer 1 overflow to be used for the transmit clock.
3	EXEN2	Timer 2 External Enable Flag. Set by the user to enable a capture or reload to occur as a result of a negative transition on T2EX if Timer 2 is not being used to clock the serial port. Cleared by the user for Timer 2 to ignore events at T2EX.
2	TR2	Timer 2 Start/Stop Control Bit. Set by the user to start Timer 2. Cleared by the user to stop Timer 2.
1	CNT2	Timer 2 Timer or Counter Function Select Bit. Set by the user to select the counter function (input from external T2 pin). Cleared by the user to select the timer function (input from on-chip core clock).
0	CAP2	Timer 2 Capture/Reload Select Bit. Set by the user to enable captures on negative transitions at T2EX if EXEN2 = 1. Cleared by the user to enable autoreloads with Timer 2 overflows or negative transitions at T2EX when EXEN2 = 1. When either RCLK = 1 or TCLK = 1, this bit is ignored and the timer is forced to autoreload on Timer 2 overflow.

Timer/Counter 2 Data Registers

Timer/Counter 2 also has two pairs of 8-bit data registers associated with it. These are used as both timer data registers and as timer capture/reload registers.

TH2 and TL2—Timer 2 data high byte and low byte.
SFR Address: CDH and CCH respectively.
Power-On Default: 00H and 00H, respectively.

RCAP2H and RCAP2L—Timer 2 capture/reload byte and low byte.
SFR Address: CBH and CAH, respectively.
Power-On Default: 00H and 00H, respectively.

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 ±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.

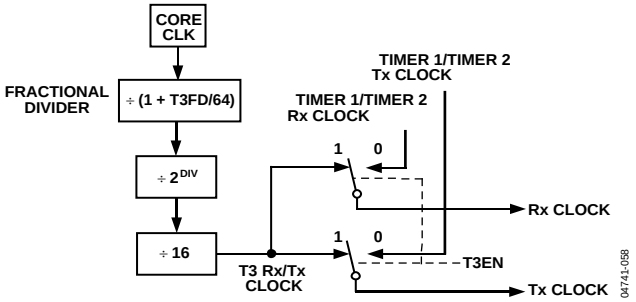

Figure 61. Timer 3, UART Baud Rate

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_{CORE} is defined in PLLCON SFR. Note that the DIV value must be rounded down.

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

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:

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

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:

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

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

QuickStart DEVELOPMENT SYSTEM

The QuickStart Development System is an entry-level, low cost development tool suite supporting the ADuC8xx MicroConverter product family. The system consists of the following PC-based (Windows®-compatible) hardware and software development tools:

Hardware:	Evaluation board and serial port programming cable.
Software:	Serial download software.
Miscellaneous:	CD-ROM documentation and prototype evaluation board.

A brief description of some of the software tools and components in the QuickStart system follows.

Download—In-Circuit Serial Downloader

The serial downloader is a Windows application that allows the user to serially download an assembled program (Intel® hexa-decimal format file) to the on-chip program flash memory via the serial COM port on a standard PC. The AN-1074 Application Note details this serial download protocol.

ASPIRE—IDE

The ASPIRE® integrated development environment is a Windows application that allows the user to compile, edit, and debug code in the same environment. The ASPIRE software allows users to debug code execution on silicon using the MicroConverter UART serial port. The debugger provides access to all on-chip peripherals during a typical debug session as well as single-step, animate (automatic single stepping), and break-point code execution control.

Note that the ASPIRE IDE is also included as part of the QuickStart-PLUS system. As part of the QuickStart-PLUS system the ASPIRE IDE also supports mixed level and C source debugging. This is not available in the QuickStart system where the program is limited to assembly only.

QuickStart-PLUS DEVELOPMENT SYSTEM

The QuickStart-PLUS development system offers users enhanced nonintrusive debug and emulation tools. The system consists of the following PC-based (Windows-compatible) hardware and software development tools:

Hardware:	Prototype Board, Accutron NonIntrusive Single-Pin Emulator.
Software:	ASPIRE Integrated Development Environment. Features full C and Assembly emulation using the Accutron single-pin emulator.
Miscellaneous:	CD-ROM documentation.

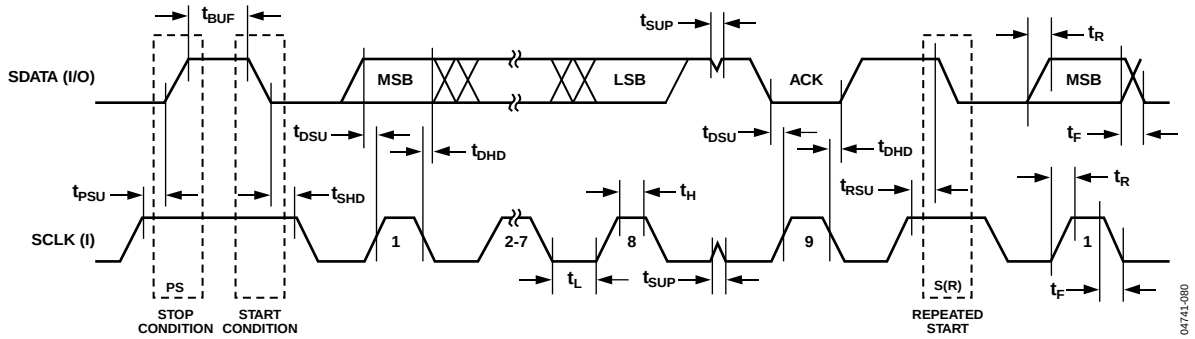

Figure 75. I²C-Compatible Interface Timing

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

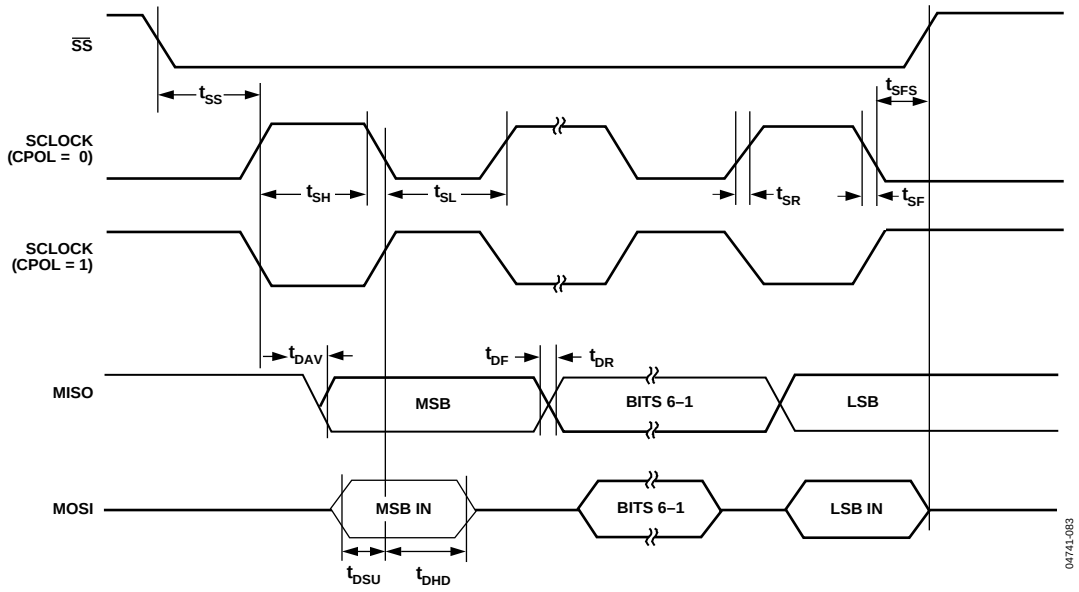

Figure 78. SPI Slave Mode Timing (CHPA = 1)