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	32KB (32K 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/aduc847bsz32-5

TABLE OF CONTENTS

ABOSOLUTE MAXIMUM RATINGS

T_A = 25°C, unless otherwise noted.

Table 2.

Parameter	Rating
AV_{DD} to AGND	–0.3 V to +7 V
AV_{DD} to DGND	–0.3 V to +7 V
DV_{DD} to DGND	–0.3 V to +7 V
DV_{DD} to DGND	–0.3 V to +7 V
AGND to DGND[1]	–0.3 V to +0.3 V
AV_{DD} to DV_{DD}	–2 V to +5 V
Analog Input Voltage to AGND[2]	–0.3 V to AV_{DD} + 0.3 V
Reference Input Voltage to AGND	–0.3 V to AV_{DD} + 0.3 V
AIN/REFIN Current (Indefinite)	30 mA
Digital Input Voltage to DGND	–0.3 V to DV_{DD} + 0.3 V
Digital Output Voltage to DGND	–0.3 V to DV_{DD} + 0.3 V
Operating Temperature Range	–40°C to +125°C
Storage Temperature Range	–65°C to +150°C
Junction Temperature	150°C
θ_{JA} Thermal Impedance (MQFP)	90°C/W
θ_{JA} Thermal Impedance (LFCSP)	52°C/W
Lead Temperature, Soldering	
Vapor Phase (60 sec)	215°C
Infrared (15 sec)	220°C

[1] AGND and DGND are shorted internally on the ADuC845, ADuC847, and ADuC848.
[2] Applies to the P1.0 to P1.7 pins operating in analog or digital input modes.

Stresses at or above those listed under Absolute Maximum Ratings may cause permanent damage to the product. This is a stress rating only; functional operation of the product at these or any other conditions above those indicated in the operational section of this specification is not implied. Operation beyond the maximum operating conditions for extended periods may affect product reliability.

ESD CAUTION

ESD (electrostatic discharge) sensitive device. Charged devices and circuit boards can discharge without detection. Although this product features patented or proprietary protection circuitry, damage may occur on devices subjected to high energy ESD. Therefore, proper ESD precautions should be taken to avoid performance degradation or loss of functionality.

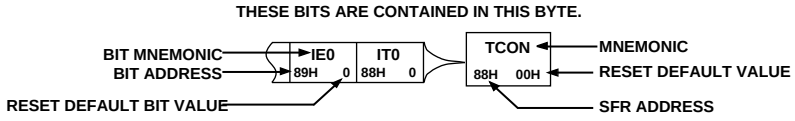
COMPLETE SFR MAP

ISPI FFH 0	WCOL FEH 0	SPE FDH 0	SPIM FCH 0	CPOL FBH 0	CPHA FAH 1	SPR1 F9H 0	SPR0 F8H 0	BITS
F7H 0	F6H 0	F5H 0	F4H 0	F3H 0	F2H 0	F1H 0	F0H 0	BITS
MDO EFH 0	MDE EEH 0	MCO EDH 0	MDI ECH 0	I2CM EBH 0	I2CRS EAH 0	I2CTX E9H 0	I2CI E8H 0	BITS
E7H 0	E6H 0	E5H 0	E4H 0	E3H 0	E2H 0	E1H 0	E0H 0	BITS
RDY0 DFH 0	RDY1 DEH 0	CAL DDH 0	NOXREF DCH 0	ERR0 DBH 0	ERR1 DAH 0	D9H 0	D8H 0	BITS
CY D7H 0	AC D6H 0	F0 D5H 0	RS1 D4H 0	RS0 D3H 0	OV D2H 0	FI D1H 0	P D0H 0	BITS
TF2 CFH 0	EXF2 CEH 0	RCLK CDH 0	TCLK CCH 0	EXEN2 CBH 0	TR2 CAH 0	CNT2 C9H 0	CAP2 C8H 0	BITS
PRE3 C7H 0	PRE2 C6H 0	PRE1 C5H 0	PRE0 C4H 1	WDIR C3H 0	WDS C2H 0	WDE C1H 0	WDWR C0H 0	BITS
BFH 0	PADC BEH 0	PT2 BDH 0	PS BCH 0	PT1 BBH 0	PX1 BAH 0	PT0 B9H 0	PX0 B8H 0	BITS
RD B7H 1	WR B6H 1	T1 B5H 1	T0 B4H 1	INT1 B3H 1	INT0 B2H 1	TxD B1H 1	RxD B0H 1	BITS
EA AFH 0	EADC AEH 0	ET2 ADH 0	ES ACH 0	ET1 ABH 0	EX1 AAH 0	ET0 A9H 0	EX0 A8H 0	BITS
A7H 1	A6H 1	A5H 1	A4H 1	A3H 1	A2H 1	A1H 1	A0H 1	BITS
SM0 9FH 0	SM1 9EH 0	SM2 9DH 0	REN 9CH 0	TB8 9BH 0	RB8 9AH 0	TI 99H 0	RI 98H 0	BITS
97H 1	96H 1	95H 1	94H 1	93H 1	92H 1	T2EX 91H 1	T2 90H 1	BITS
TF1 8FH 0	TR1 8EH 0	TF0 8DH 0	TR0 8CH 0	IE1 8BH 0	IT1 8AH 0	IE0 89H 0	IT0 88H 0	BITS
87H 1	86H 1	85H 1	84H 1	83H 1	82H 1	81H 1	80H 1	BITS

SPICON F8H 05H	RESERVED	RESERVED	DACL FBH 00H	DACH FCH 00H	DACCON FDH 00H	RESERVED	RESERVED
B F0H 00H	RESERVED	I2CADD1 F2H 7FH	NOT USED	RESERVED	RESERVED	RESERVED	SPIDAT F7H 00H
I2CCON E8H 00H	GN0L[2] E9H xxH	GN0M[2] EAH xxH	GN0H[2] EBH xxH	GN1L[2] ADuC845 ONLY ECH xxH	GN1H[2] ADuC845 ONLY EDH xxH	RESERVED	RESERVED
ACC E0H 00H	OF0L E1H xxH	OF0M E2H xxH	OF0H E3H xxH	OF1L ADuC845 ONLY E4H xxH	OF1H ADuC845 ONLY E5H xxH	ADC0CON2 E6H 00H	RESERVED
ADCSTAT D8H 00H	ADC0L NOT AVAILABLE ON ADuC848 D9H 00H	ADC0M DAH 00H	ADC0H DBH 00H	ADC1M ADuC845 ONLY DCH 00H	ADC1H ADuC845 ONLY DDH 00H	ADC1L ADuC845 ONLY DEH 00H	PSMCON DFH DEH
PSW D0H 00H	ADCMODE D1H 08H	ADC0CON1 D2H 07H	ADC1CON ADuC845 ONLY D3H 00H	SF D4H 45H	ICON D5H 00H	RESERVED	PLLCON D7H 53H
T2CON C8H 00H	RESERVED	RCAP2L CAH 00H	RCAP2H CBH 00H	TL2 CCH 00H	TH2 CDH 00H	RESERVED	RESERVED
WDCON C0H 10H	RESERVED	CHIPID C2H A0H	RESERVED	RESERVED	RESERVED	EDARL C6H 00H	EDARH C7H 00H
IP B8H 00H	ECON B9H 00H	RESERVED	RESERVED	EDATA1 BCH 00H	EDATA2 BDH 00H	EDATA3 BEH 00H	EDATA4 BFH 00H
P3 B0H FFH	PWM0L B1H 00H	PWM0H B2H 00H	PWM1L B3H 00H	PWM1H B4H 00H	RESERVED	RESERVED	SPH B7H 00H
IE A8H 00H	IEIP2 A9H A 0H	RESERVED	RESERVED	RESERVED	RESERVED	PWMCON AEH 00H	CFG845/7/8 AFH 00H
P2 A0H FFH	TIMECON A1H 00H	HTHSEC[1] A2H 00H	SEC[1] A3H 00H	MIN[1] A4H 00H	HOUR[1] A5H 00H	INTVAL A6H 00H	DPCON A7H 00H
SCON 98H 00H	SBUF 99H 00H	I2CDAT 9AH 00H	I2CADD 9BH 55H	RESERVED	T3FD 9DH 00H	T3CON 9EH 00H	EWAIT 9FH 00H
P1 90H FFH	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED
TCON 88H 00H	TMOD 89H 00H	TL0 8AH 00H	TL1 8BH 00H	TH0 8CH 00H	TH1 8DH 00H	RESERVED	RESERVED
P0 80H FFH	SP 81H 07H	DPL 82H 00H	DPH 83H 00H	DPP 84H 00H	RESERVED	RESERVED	PCON 87H 00H

[1] THESE SFRs MAINTAIN THEIR PRE-RESET VALUES AFTER A RESET IF TIMECON.0 = 1.
[2] CALIBRATION COEFFICIENTS ARE PRECONFIGURED ON POWER-UP TO FACTORY CALIBRATED VALUES.

SFR MAP KEY:



SFR NOTE:
SFRs WHOSE ADDRESSES END IN 0H OR 8H ARE BIT ADDRESSABLE.

04741-073

Figure 7. Complete SFR Map for the ADuC845, ADuC847, and ADuC848

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.

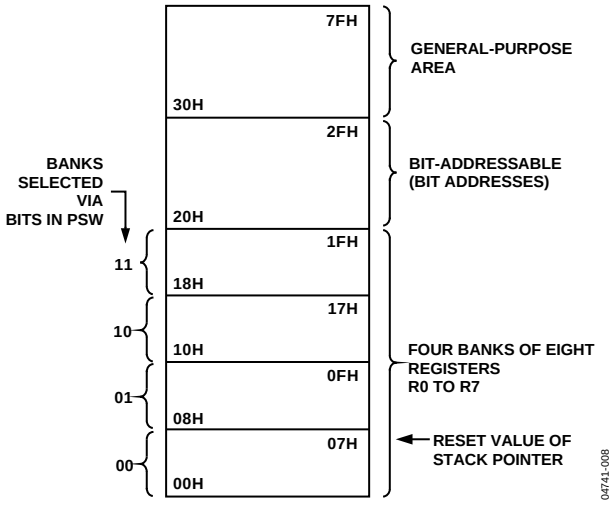


Figure 8. Lower 128 Bytes of Internal Data Memory

Internal XRAM

The ADuC845, ADuC847, and ADuC848 contain 2 kbytes of on-chip extended data memory. This memory, although on-chip, is accessed via the MOVX instruction. The 2 kbytes of internal XRAM are mapped into the bottom 2 kbytes of the external address space if the CFG84x.0 (Table 7) bit is set; otherwise, access to the external data memory occurs just like a standard 8051.

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

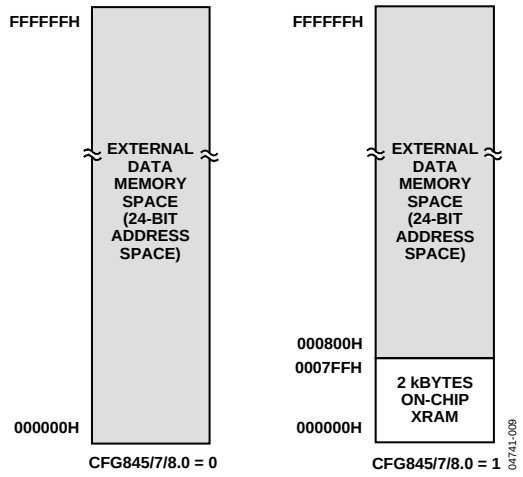


Figure 9. Internal and External XRAM

When enabled and when accessing the internal XRAM, the P0 and P2 port pin operations, as well as the $\overline{RD}$ and $\overline{WR}$ strobes, do not operate as a standard 8051 MOVX instruction. This allows the user to use these port pins as standard I/O. The internal XRAM can be configured as part of the extended 11-bit stack pointer. By default, the stack operates exactly like an 8052 in that it rolls over from FFH to 00H in the general-purpose RAM. On the ADuC845, ADuC847, and ADuC848, however, it is possible (by setting CFG845.7/ADuC847.7/ADuC848.7) to enable the 11-bit extended stack pointer. In this case, the stack rolls over from FFH in RAM to 0100H in XRAM.

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

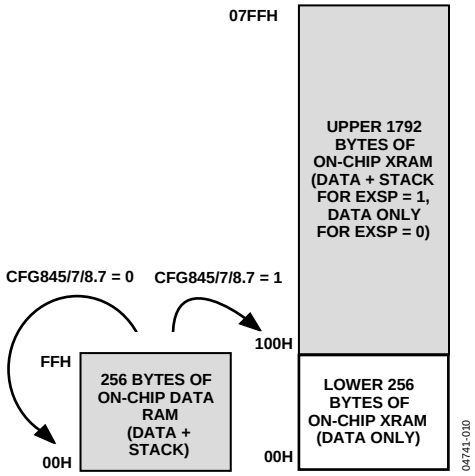


Figure 10. Extended Stack Pointer Operation

External Data Memory (External XRAM)

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

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

EWAIT SFR

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

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

ADC Noise Performance with Chop Disabled ($\overline{CHOP}$ = 1)

Table 14 through Table 17 show the output rms noise and output peak-to-peak resolution in bits (rounded to the nearest 0.5 LSB) for some typical output update rates. The numbers are typical and are generated at a differential input voltage of 0 V and a common-mode voltage of 2.5 V. The output update rate is selected via the SF7 to SF0 bits in the SF filter register. Note that the peak-to-peak resolution figures represent the resolution for which there is no code flicker within a 6-sigma limit.

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

The numbers in the tables are given for the bipolar input ranges. For the unipolar ranges, the rms noise numbers are the same as the bipolar range, but the peak-to-peak resolution is based on half the signal range, which effectively means losing 1 bit of resolution. Typically, the performance of the ADC with chop disabled shows a 0.5 LSB degradation over the performance with chop enabled.

Table 14. ADuC845 and ADuC847 Typical Output RMS Noise (μV) vs. Input Range and Update Rate with Chop Disabled

SF Word	Data Update Rate (Hz)	Input Range							
		±20 mV	±40 mV	±80 mV	±160 mV	±320 mV	±640 mV	±1.28 V	±2.56 V
3	1365.33	30.64	24.5	56.18	100.47	248.39	468.65	774.36	1739.5
13	315.08	2.07	1.95	2.28	3.24	8.22	13.9	20.98	49.26
68	59.36	0.85	0.79	1.01	0.99	0.79	1.29	2.3	3.7
82	49.95	0.83	0.77	0.85	0.77	0.91	1.12	1.59	3.2
255	16.06	0.52	0.58	0.59	0.48	0.52	0.57	1.16	1.68

Table 15. ADuC845 and ADuC847 Typical Peak-to-Peak Resolution (Bits) vs. Input Range and Update Rate with Chop Disabled

SF Word	Data Update Rate (Hz)	Input Range							
		±20 mV	±40 mV	±80 mV	±160 mV	±320 mV	±640 mV	±1.28 V	±2.56 V
3	1365.33	7.5	9	9	9	9	9	9	9
13	315.08	11.5	12.5	13.5	14	13.5	14	14	14
68	59.36	13	14	14.5	15.5	17	17	17.5	18
82	49.95	13	14	15	16	16.5	17.5	18	18
255	16.06	13.5	14.5	15.5	16.5	17.5	18.5	18.5	19

Table 16. ADuC848 Typical Output RMS Noise (μV) vs. Input Range and Update Rate with Chop Disabled

SF Word	Data Update Rate (Hz)	Input Range							
		±20 mV	±40 mV	±80 mV	±160 mV	±320 mV	±640 mV	±1.28 V	±2.56 V
3	1365.33	30.64	24.5	56.18	100.47	248.39	468.65	774.36	1739.5
13	315.08	2.07	1.95	2.28	3.24	8.22	13.9	20.98	49.26
69	59.36	0.85	0.79	1.01	0.99	0.79	1.29	2.3	3.7
82	49.95	0.83	0.77	0.85	0.77	0.91	1.12	1.59	3.2
255	16.06	0.52	0.58	0.59	0.48	0.52	0.57	1.16	1.68

Table 17. ADuC848 Typical Peak-to-Peak Resolution (Bits) vs. Input Range and Update Rate with Chop Disabled

SF Word	Data Update Rate (Hz)	Input Range							
		±20 mV	±40 mV	±80 mV	±160 mV	±320mV	±640mV	±1.28 V	±2.56 V
3	1365.33	7.5	9	9	9	9	9	9	9
13	315.08	11.5	12.5	13.5	14	13.5	14	14	14
68	59.36	13	14	14.5	15.5	16	16	16	16
82	49.95	13	14	15	16	16	16	16	16
255	16.06	13.5	14.5	15.5	16	16	16	16	16

ADCMODE (ADC MODE REGISTER)

Used to control the operational mode of both ADCs.

SFR Address: D1H
Power-On Default: 08H
Bit Addressable: No

Table 24. ADCMODE SFR Bit Designations

Bit No.	Name	Description
7	---	Not Implemented. Write Don't Care.
6	REJ60	Automatic 60 Hz Notch Select Bit. Setting this bit places a notch in the frequency response at 60 Hz, allowing simultaneous 50 Hz and 60 Hz rejection at an SF word of 82 decimal. This 60 Hz notch can be set only if SF ≥68 decimal, that is, the regular filter notch must be ≤60 Hz. This second notch is placed at 60 Hz only if the device clock is at 32.768 kHz.
5	ADC0EN	Primary ADC Enable. Set by the user to enable the primary ADC and place it in the mode selected in MD2–MD0. Cleared by the user to place the primary ADC into power-down mode.
4	ADC1EN (ADuC845 only)	Auxiliary (ADuC845 only) ADC Enable. Set by the user to enable the auxiliary (ADuC845 only) ADC and place it in the mode selected in MD2–MD0. Cleared by the user to place the auxiliary (ADuC845 only) ADC in power-down mode.
3	$\overline{\text{CHOP}}$	Chop Mode Disable. Set by the user to disable chop mode on both the primary and auxiliary (ADuC845 only) ADC allowing a three times higher ADC data throughput. SF values as low as 3 are allowed with this bit set, giving up to 1.3 kHz ADC update rates. Cleared by the user to enable chop mode on both the primary and auxiliary (ADuC845 only) ADC.
2, 1, 0	MD2, MD1, MD0	Primary and Auxiliary (ADuC845 only) ADC Mode Bits. These bits select the operational mode of the enabled ADC as follows: MD2 MD1 MD0 0 0 0 ADC Power-Down Mode (Power-On Default). 0 0 1 Idle Mode. In idle mode, the ADC filter and modulator are held in a reset state although the modulator clocks are still provided. 0 1 0 Single Conversion Mode. In single conversion mode, a single conversion is performed on the enabled ADC. Upon completion of a conversion, the ADC data registers (ADC0H/M/L and/or ADC1H/M/L (ADuC845 only)) are updated. The relevant flags in the ADCSTAT SFR are written, and power-down is re-entered with the MD2–MD0 accordingly being written to 000. Note that ADC0L is not available on the ADuC848. 0 1 1 Continuous Conversion. In continuous conversion mode, the ADC data registers are regularly updated at the selected update rate (see the Sinc Filter SFR Bit Designations in Table 28). 1 0 0 Internal Zero-Scale Calibration. Internal short automatically connected to the enabled ADC input(s). 1 0 1 Internal Full-Scale Calibration. Internal or external REFIN± or REFIN2± V_{REF} (as determined by XREF bits in ADC0CON2 and/or AXREF (ADuC845 only) in ADC1CON (ADuC845 only) is automatically connected to the enabled ADC input(s) for this calibration. 1 1 0 System Zero-Scale Calibration. User should connect system zero-scale input to the enabled ADC input(s) as selected by CH3–CH0 and ACH3–ACH0 bits in the ADC0CON2 and ADC1CON (ADuC845 only) registers. 1 1 1 System Full-Scale Calibration. User should connect system full-scale input to the enabled ADC input(s) as selected by CH3–CH0 and ACH3–ACH0 bits in the ADC0CON2 and ADC1CON (ADuC845 only) registers.

Notes on the ADCMODE Register

Any change to the MD bits immediately resets both ADCs (auxiliary ADC only applicable to the ADuC845). A write to the MD2–MD0 bits with no change in contents is also treated as a reset. (See the exception to this in the third note of this section.)

If ADC1CON1 and ADC1CON2 are written when ADC0EN = 1, or if ADC0EN is changed from 0 to 1, both ADCs are also immediately reset. In other words, the primary ADC is given priority over the auxiliary ADC and any change requested on the primary ADC is immediately responded to. Only applicable to the ADuC845.

On the other hand, if ADC1CON is written to or if ADC1EN is changed from 0 to 1, only the auxiliary ADC is reset. For example, if the primary ADC is continuously converting when the auxiliary ADC change or enable occurs, the primary ADC continues undisturbed. Rather than allow the auxiliary ADC to operate with a phase difference from the primary ADC, the auxiliary ADC falls into step with the outputs of the primary ADC. The result is that the first conversion time for the auxiliary ADC is delayed by up to three outputs while the auxiliary ADC update rate is synchronized to the primary ADC. Only applicable to ADuC845. If the ADC1CON write occurs after the primary ADC has completed its operation, the auxiliary ADC can respond immediately without having to fall into step with the primary ADCs output cycle.

If the devices are powered down via the PD bit in the PCON register, the current ADCMODE bits are preserved, that is, they are not reset to default state. Upon a subsequent resumption of normal operating mode, the ADCs restarts the selected operation defined by the ADCMODE register.

Once ADCMODE has been written with a calibration mode, the RDY0/1 (ADuC845 only) bits (ADCSTAT) are reset and the calibration commences. On completion, the appropriate calibration registers are written, the relevant bits in ADCSTAT are written, and the MD2–MD0 bits are reset to 000B to indicate that the ADC is back in power-down mode.

Any calibration request of the auxiliary ADC while the temperature sensor is selected fails to complete. Although the RDY1 bit is set at the end of the calibration cycle, no update of the calibration SFRs takes place, and the ERR1 bit is set. ADuC845 only.

Calibrations performed at maximum SF (see Table 28) value (slowest ADC throughput rate) help to ensure optimum calibration.

The duration of a calibration cycle is 2/Fadc for chop-on mode and 4/Fadc for chop-off mode.

SF (ADC SINC FILTER CONTROL REGISTER)

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

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

Table 28. Sinc Filter SFR Bit Designations

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

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

$$Fadc\ (Chop\ On) = \frac{1}{3 \times 8 \times SFword} \times 32.768\ \text{kHz}$$

where *SFword* is in decimal.

$$Fadc\ (Chop\ Off) = \frac{1}{8 \times SFword} \times 32.768\ \text{kHz}$$

where *SFword* is in decimal.

Table 29. SF SFR Bit Examples

Chop Enabled (ADCMODE.3 = 0)

SF (Decimal)	SF (Hexadecimal)	Fadc (Hz)	Tadc (ms)	Tsettle (ms)
13[1]	0D	105.3	9.52	19.04
69	45	19.79	50.53	101.1
82	52	16.65	60.06	120.1
255	FF	5.35	186.77	373.54

Chop Disabled (ADCMODE.3 = 1)

SF (Decimal)	SF (Hexadecimal)	Fadc (Hz)	Tadc (ms)	Tsettle (ms)
3	03	1365.3	0.73	2.2
69	45	59.36	16.84	50.52
82	52	49.95	20.02	60.06
255	FF	16.06	62.25	186.8

[1] With chop enabled, if an SF word smaller than 13 is written to this SF register, the filter automatically defaults to 13.

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

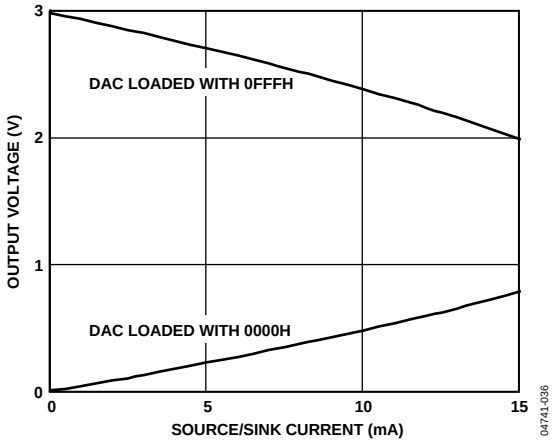


Figure 36. Source and Sink Current Capability with V_{REF} = AV_{DD} = 3 V

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

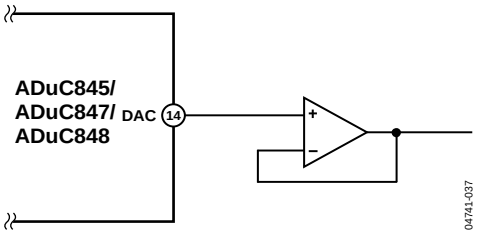


Figure 37. Buffering the DAC Output

The internal DAC output buffer also features a high impedance disable function. In the chip's default power-on state, the DAC is disabled and its output is in a high impedance state (or three-state) where it remains inactive until enabled in software. This means that if a zero output is desired during power-on or power-down transient conditions, a pull-down resistor must be added to each DAC output. Assuming that this resistor is in place, the DAC output remains at ground potential whenever the DAC is disabled.

PULSE-WIDTH MODULATOR (PWM)

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

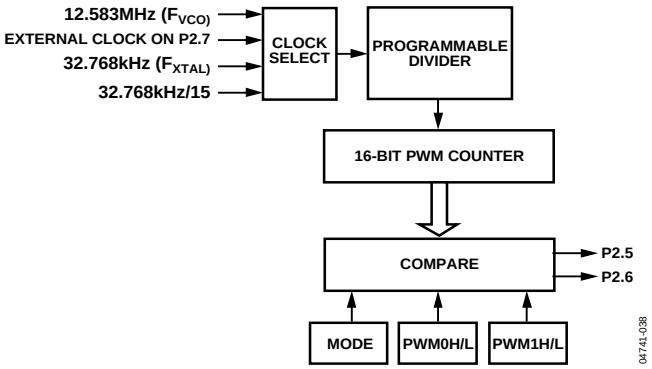


Figure 38. PWM Block Diagram

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

PWMCON (as described in Table 34) controls the different modes of operation of the PWM as well as the PWM clock frequency. PWM0H/L and PWM1H/L are the data registers that determine the duty cycles of the PWM outputs at P2.5 and P2.6.

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

I²C SERIAL INTERFACE

The ADuC845/ADuC847/ADuC848 support a fully licensed I²C serial interface. The I²C interface is implemented as a full hardware slave and software master. SDATA (Pin 27 on the MQFP package and Pin 29 on the LFCSP package) is the data I/O pin. SCLK (Pin 26 on the MQFP package and Pin 28 on the LFCSP package) is the serial interface clock for the SPI interface. The I²C interface on the devices is fully independent of all other pin/function multiplexing. The I²C interface incorporated on the ADuC845/ADuC847/ADuC848 also includes a second address register (I2CADD1) at SFR Address F2H with a default power-on value of 7FH. The I²C interface is always available to the user and is not multiplexed with any other I/O functionality on the chip. This means that the I²C and SPI interfaces can be used at the same time.

Note that when using the I²C and SPI interfaces simultaneously, they both use the same interrupt routine (Vector Address 3BH). When an interrupt occurs from one of these, it is necessary to interrogate each interface to see which one has triggered the ISR request.

The four SFRs that are used to control the I²C interface are described next.

I2CCON—I²C Control Register

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

Table 40. I2CCON SFR Bit Designations

Bit No.	Name	Description
7	MDO	I²C Software Master Data Output Bit (master mode only). This data bit is used to implement a master I²C transmitter interface in software. Data written to this bit is output on the SDATA pin if the data output enable bit (MDE) is set.
6	MDE	I²C Software Output Enable Bit (master mode only). Set by the user to enable the SDATA pin as an output (Tx). Cleared by the user to enable the SDATA pin as an input (Rx).
5	MCO	I²C Software Master Clock Output Bit (master mode only). This bit is used to implement the SCLK for a master I²C transmitter in software. Data written to this bit is output on the SCLK pin.
4	MDI	I²C Software Master Data Input Bit (master mode only). This data bit is used to implement a master I²C receiver interface in software. Data on the SDATA pin is latched into this bit on an SCLK transition if the data output enable (MDE) bit is 0.
3	I2CM	I²C Master/Slave Mode Bit. Set by the user to enable I²C software master mode. Cleared by the user to enable I²C hardware slave mode.
2	I2CRS	I²C Reset Bit (slave mode only). Set by the user to reset the I²C interface. Cleared by the user code for normal I²C operation.
1	I2CTX	I²C Direction Transfer Bit (slave mode only). Set by the MicroConverter if the I²C interface is transmitting. Cleared by the MicroConverter if the I²C interface is receiving.
0	I2CI	I²C Interrupt Bit (slave mode only). Set by the MicroConverter after a byte has been transmitted or received. Cleared by the MicroConverter when the user code reads the I2CDAT SFR. I2CI should not be cleared by user code.

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. Cleared by the user to disable autoswapping of the DPTR. 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. DP1m1 DP1m0 Behavior of the Shadow Data Pointer 0 0 8052 behavior. 0 1 DPTR is post-incremented after a MOVX or a MOVC instruction. 1 0 DPTR is post-decremented after a MOVX or MOVC instruction. 1 1 DPTR LSB is toggled after a MOVX or MOVC instruction. (This instruction can be useful for moving 8-bit blocks to/from 16-bit devices.)
3, 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. DP0m1 DP0m0 Behavior of the Main Data Pointer 0 0 8052 behavior. 0 1 DPTR is post-incremented after a MOVX or a MOVC instruction. 1 0 DPTR is post-decremented after a MOVX or MOVC instruction. 1 1 DPTR LSB is toggled after a MOVX or MOVC instruction. (This instruction is useful for moving 8-bit blocks to/from 16-bit devices.)
1	----	Not Implemented. Write Don't Care.
0	DPSEL	Data Pointer Select. 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. 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
```

TIME INTERVAL COUNTER (TIC)

A TIC is provided on-chip for counting longer intervals than the standard 8051-compatible timers can count. The TIC is capable of timeout intervals ranging from 1/128 second to 255 hours. Also, this counter is clocked by the external 32.768 kHz crystal rather than by the core clock, and it can remain active in power-down mode and time long power-down intervals. This has obvious applications for remote battery-powered sensors where regular widely spaced readings are required. Note that instructions to the TIC SFRs are also clocked at 32.768 kHz, so sufficient time must be allowed in user code for these instructions to execute.

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

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

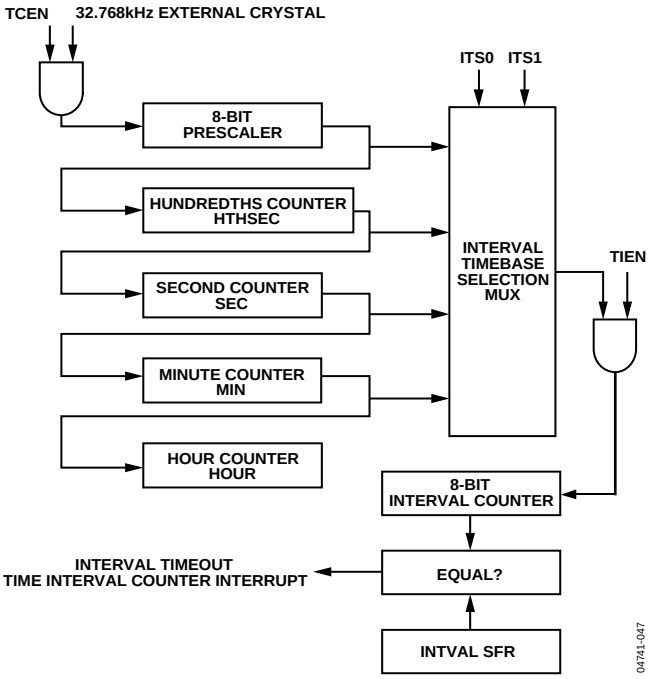


Figure 47. TIC Simplified Block Diagram

Mode 0 (8-Bit Shift Register Mode)

Mode 0 is selected by clearing both the SM0 and SM1 bits in the SFR SCON. Serial data enters and exits through RxD. TxD outputs the shift clock. Eight data bits are transmitted or received. Transmission is initiated by any instruction that writes to SBUF. The data is shifted out of the RxD line. The 8 bits are transmitted with the least significant bit (LSB) first.

Reception is initiated when the receive enable bit (REN) is 1 and the receive interrupt bit (RI) is 0. When RI is cleared, the data is clocked into the RxD line, and the clock pulses are output from the TxD line as shown in Figure 58.

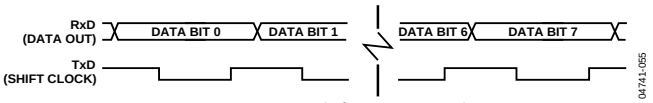


Figure 58. 8-Bit Shift Register Mode

Mode 1 (8-Bit UART, Variable Baud Rate)

Mode 1 is selected by clearing SM0 and setting SM1. Each data byte (LSB first) is preceded by a start bit (0) and followed by a stop bit (1). Therefore, 10 bits are transmitted on TxD or are received on RxD. The baud rate is set by the Timer 1 or Timer 2 overflow rate, or a combination of the two (one for transmission and the other for reception).

Transmission is initiated by writing to SBUF. The write to SBUF signal also loads a 1 (stop bit) into the 9th bit position of the transmit shift register. The data is output bit-by-bit until the stop bit appears on TxD and the transmit interrupt flag (TI) is automatically set as shown in Figure 59.

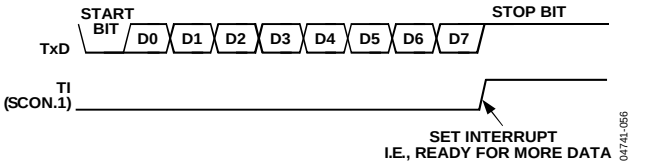


Figure 59. 8-Bit Variable Baud Rate

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

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

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

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

If any of these conditions is *not* met, the received frame is irretrievably lost, and RI is not set.

Mode 2 (9-Bit UART with Fixed Baud Rate)

Mode 2 is selected by setting SM0 and clearing SM1. In this mode, the UART operates in 9-bit mode with a fixed baud rate. The baud rate is fixed at Core_Clk/64 by default, although by setting the SMOD bit in PCON, the frequency can be doubled to Core_Clk/32. Eleven bits are transmitted or received: a start bit (0), 8 data bits, a programmable 9th bit, and a stop bit (1). The 9th bit is most often used as a parity bit, although it can be used for anything, including a ninth data bit if required.

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

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

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

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

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

If any of these conditions is not met, the received frame is irretrievably lost, and RI is not set.

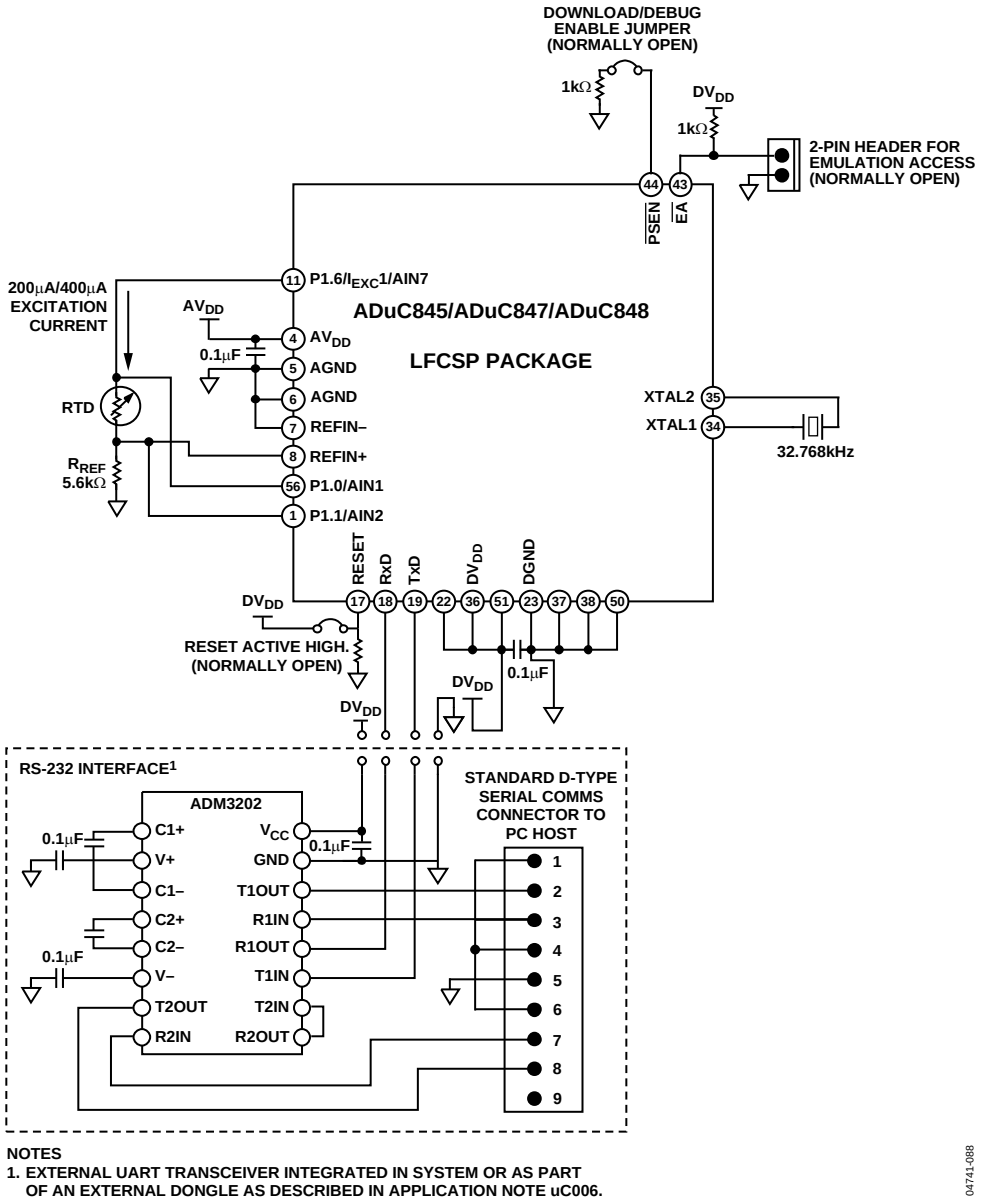


Figure 70. UART Connectivity in Typical System

In addition to the basic UART connections, users also need a way to trigger the chip into download mode. This is accomplished via a 1 kΩ pull-down resistor that can be jumpered onto the $\overline{\text{PSEN}}$ pin, as shown in Figure 70. To get the devices into download mode, connect this jumper and power-cycle the device (or manually reset the device, if a manual reset button is available), and it is ready to receive a new program serially. With the jumper removed, the device powers on in normal mode (and runs the program) whenever power is cycled or RESET is toggled. Note that $\overline{\text{PSEN}}$ is normally an output and that it is sampled as an input only on the falling edge of RESET, that is, at power-on or upon an external manual reset. Note also that if any external circuitry unintentionally pulls $\overline{\text{PSEN}}$ low during power-on or reset events, it may cause the chip to enter download mode and fail to begin user code execution. To prevent this, ensure that no external signals are capable of pulling the $\overline{\text{PSEN}}$ pin low, except for the external $\overline{\text{PSEN}}$ jumper itself or the method of download entry in use during a reset or power-cycle condition.

Embedded Serial Port Debugger

From a hardware perspective, entry to serial port debug mode is identical to the serial download entry sequence described previously. In fact, both serial download and serial port debug modes are essentially one mode of operation used in two different ways.

The serial port debugger is fully contained on the device, unlike ROM monitor type debuggers, and, therefore, no external memory is needed to enable in-system debug sessions.

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.

TIMING SPECIFICATIONS

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

For timing purposes, a port pin is no longer floating when a 100 mV change from load voltage occurs. A port pin begins to float when a 100 mV change from the loaded V_{OH}/V_{OL} level occurs as shown in Figure 72.

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

AV_{DD} = 2.7 V to 3.6 V or 4.75 V to 5.25 V, DV_{DD} = 2.7 V to 3.6 V or 4.75 V to 5.25 V; all specifications T_{MIN} to T_{MAX}, unless otherwise noted.

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

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

[1] ADuC845/ADuC847/ADuC848 internal PLL locks onto a multiple (512 times) of the 32.768 kHz external crystal frequency to provide a stable 12.58 MHz internal clock for the system. The core can operate at this frequency or at a binary submultiple called Core_Clk, selected via the PLLCON SFR.
[2] This number is measured at the default Core_Clk operating frequency of 1.57 MHz.
[3] ADuC845/ADuC847/ADuC848 machine cycle time is nominally defined as 1/Core_Clk.

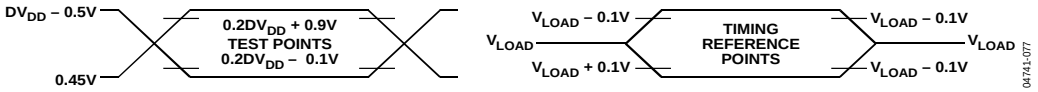


Figure 72. Timing Waveform Characteristics

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.

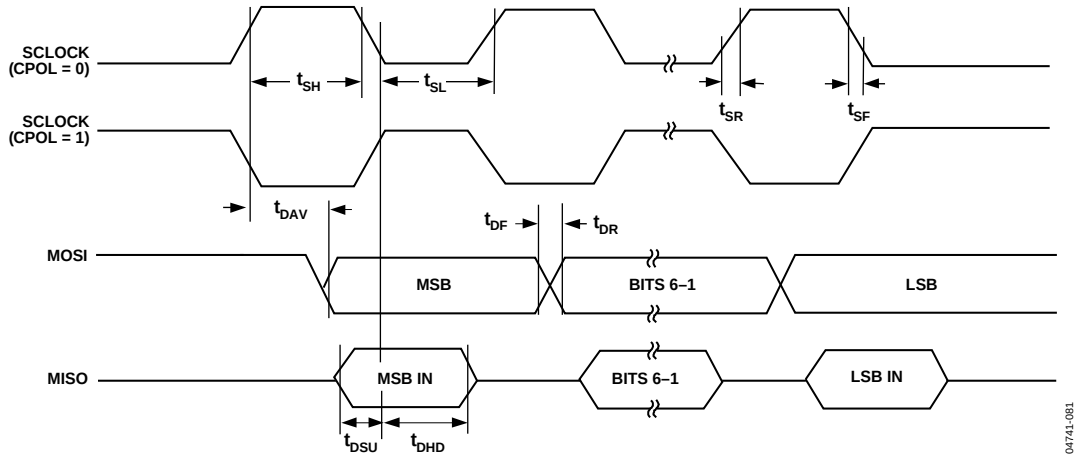


Figure 76. SPI Master Mode Timing (CHPA = 1)

Table 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

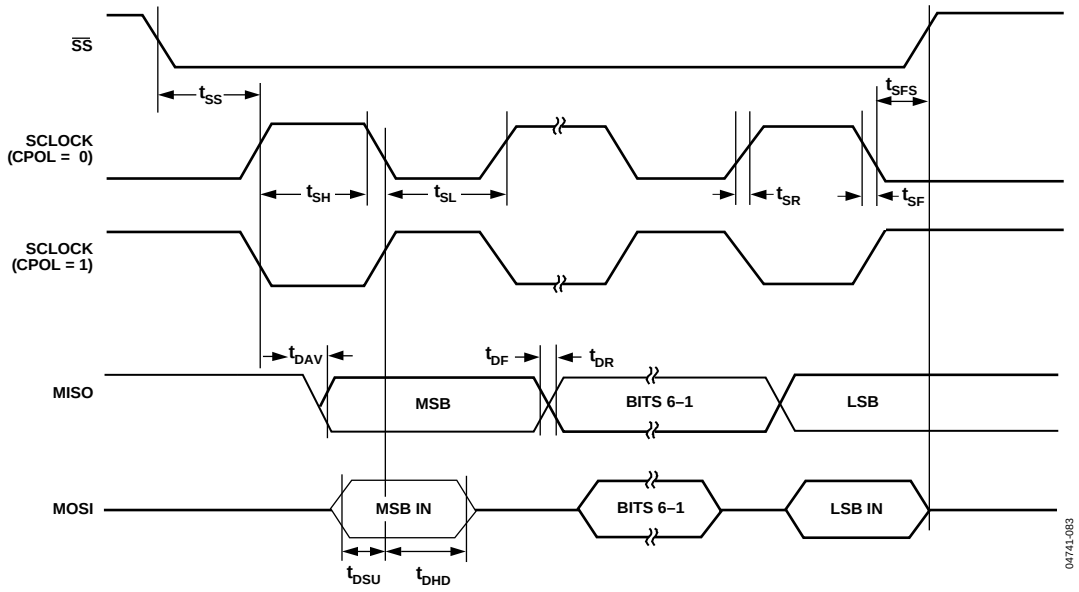


Figure 78. SPI Slave Mode Timing (CHPA = 1)

NOTES