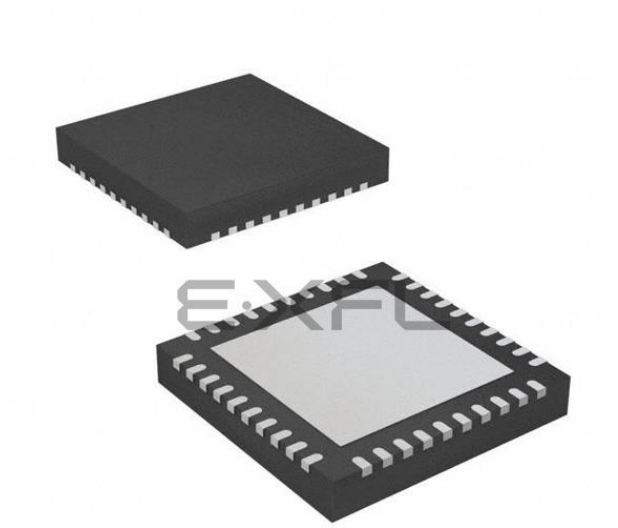

Welcome to E-XFL.COM

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	8051
Core Size	8-Bit
Speed	25MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	DMA, LCD, POR, PWM, WDT
Number of I/O	34
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	8.25K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.8V
Data Converters	A/D 16x10b/12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	40-VFQFN Exposed Pad
Supplier Device Package	40-QFN (6x6)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f967-b-gm

Table of Contents

1. System Overview 22
1.1. CIP-51™ Microcontroller Core 28
1.1.1. Fully 8051 Compatible 28
1.1.2. Improved Throughput 28
1.1.3. Additional Features 28
1.2. Port Input/Output 29
1.3. Serial Ports 30
1.4. Programmable Counter Array 30
1.5. SAR ADC with 16-bit Auto-Averaging Accumulator and Autonomous Low Power Burst Mode 31
1.6. Programmable Current Reference (IREF0) 32
1.7. Comparators 32
2. Ordering Information 34
3. Pinout and Package Definitions 35
3.1. DQFN-76 Package Specifications 46
3.1.1. Package Drawing 46
3.1.2. Land Pattern 47
3.1.3. Soldering Guidelines 48
3.2. QFN-40 Package Specifications 50
3.3. TQFP-80 Package Specifications 52
3.3.1. Soldering Guidelines 55
4. Electrical Characteristics 56
4.1. Absolute Maximum Specifications 56
4.2. Electrical Characteristics 57
5. SAR ADC with 16-bit Auto-Averaging Accumulator and Autonomous Low Power Burst Mode 78
5.1. Output Code Formatting 78
5.2. Modes of Operation 80
5.2.1. Starting a Conversion 80
5.2.2. Tracking Modes 80
5.2.3. Burst Mode 82
5.2.4. Settling Time Requirements 83
5.2.5. Gain Setting 83
5.3. 8-Bit Mode 84
5.4. 12-Bit Mode 84
5.5. Low Power Mode 85
5.6. Programmable Window Detector 91
5.6.1. Window Detector In Single-Ended Mode 93
5.6.2. ADC0 Specifications 94
5.7. ADC0 Analog Multiplexer 95
5.8. Temperature Sensor 97
5.8.1. Calibration 97
5.9. Voltage and Ground Reference Options 100

List of Registers

SFR Definition 5.1. ADC0CN: ADC0 Control 86
SFR Definition 5.2. ADC0CF: ADC0 Configuration 87
SFR Definition 5.3. ADC0AC: ADC0 Accumulator Configuration 88
SFR Definition 5.4. ADC0PWR: ADC0 Burst Mode Power-Up Time 89
SFR Definition 5.5. ADC0TK: ADC0 Burst Mode Track Time 90
SFR Definition 5.6. ADC0H: ADC0 Data Word High Byte 91
SFR Definition 5.7. ADC0L: ADC0 Data Word Low Byte 91
SFR Definition 5.8. ADC0GTH: ADC0 Greater-Than High Byte 92
SFR Definition 5.9. ADC0GTL: ADC0 Greater-Than Low Byte 92
SFR Definition 5.10. ADC0LTH: ADC0 Less-Than High Byte 93
SFR Definition 5.11. ADC0LTL: ADC0 Less-Than Low Byte 93
SFR Definition 5.12. ADC0MX: ADC0 Input Channel Select 96
SFR Definition 5.13. TOFFH: Temperature Sensor Offset High Byte 99
SFR Definition 5.14. TOFFL: Temperature Sensor Offset Low Byte 99
SFR Definition 5.15. REF0CN: Voltage Reference Control 102
SFR Definition 6.1. IREF0CN: Current Reference Control 103
SFR Definition 6.2. IREF0CF: Current Reference Configuration 104
SFR Definition 7.1. CPT0CN: Comparator 0 Control 108
SFR Definition 7.2. CPT0MD: Comparator 0 Mode Selection 109
SFR Definition 7.3. CPT1CN: Comparator 1 Control 110
SFR Definition 7.4. CPT1MD: Comparator 1 Mode Selection 111
SFR Definition 7.5. CPT0MX: Comparator0 Input Channel Select 113
SFR Definition 7.6. CPT1MX: Comparator1 Input Channel Select 114
SFR Definition 8.1. DPL: Data Pointer Low Byte 121
SFR Definition 8.2. DPH: Data Pointer High Byte 121
SFR Definition 8.3. SP: Stack Pointer 122
SFR Definition 8.4. ACC: Accumulator 122
SFR Definition 8.5. B: B Register 122
SFR Definition 8.6. PSW: Program Status Word 123
SFR Definition 9.1. PSBANK: Program Space Bank Select 127
SFR Definition 10.1. EMI0CN: External Memory Interface Control 132
SFR Definition 10.2. EMI0CF: External Memory Configuration 133
SFR Definition 10.3. EMI0TC: External Memory Timing Control 138
SFR Definition 11.1. DMA0EN: DMA0 Channel Enable 150
SFR Definition 11.2. DMA0INT: DMA0 Full-Length Interrupt 151
SFR Definition 11.3. DMA0MINT: DMA0 Mid-Point Interrupt 152
SFR Definition 11.4. DMA0BUSY: DMA0 Busy 153
SFR Definition 11.5. DMA0SEL: DMA0 Channel Select for Configuration 154
SFR Definition 11.6. DMA0NMD: DMA Channel Mode 155
SFR Definition 11.7. DMA0NCF: DMA Channel Configuration 156
SFR Definition 11.8. DMA0NBAH: Memory Base Address High Byte 157
SFR Definition 11.9. DMA0NBAL: Memory Base Address Low Byte 157
SFR Definition 11.10. DMA0NAOH: Memory Address Offset High Byte 158

Table 3.1. Pin Definitions for the C8051F96x (Continued)

Name	Pin Numbers			Type	Description
	DQFN76	TQFP80	QFN40		
P4.2 LCD10	B2	5		D I/O or A In A O	Port 4.2. See Port I/O Section for a complete description. LCD Segment Pin 10
P4.3 LCD11	B1	3		D I/O or A In A O	Port 4.3. See Port I/O Section for a complete description. LCD Segment Pin 11
P4.4 LCD12	D1	80		D I/O or A In A O	Port 4.4. See Port I/O Section for a complete description. LCD Segment Pin 12
P4.5 LCD13	B28	77		D I/O or A In A O	Port 4.5. See Port I/O Section for a complete description. LCD Segment Pin 13
P4.6 LCD14	B27	75		D I/O or A In A O	Port 4.6. See Port I/O Section for a complete description. LCD Segment Pin 14
P4.7 LCD15	B26	73		D I/O or A In A O	Port 4.7. See Port I/O Section for a complete description. LCD Segment Pin 15
P5.0 LCD16	B25	71		D I/O or A In A O	Port 5.0. See Port I/O Section for a complete description. LCD Segment Pin 16
P5.1 LCD17	B24	69		D I/O or A In A O	Port 5.1. See Port I/O Section for a complete description. LCD Segment Pin 17

Table 4.4. Digital Supply Current with DC-DC Converter Disabled (Continued)
–40 to +85 °C, 25 MHz system clock unless otherwise specified.

Parameter	Condition	Min	Typ	Max	Unit
Digital Supply Current—Sleep Mode (LCD disabled, RTC enabled)					
Digital Supply Current (Sleep Mode, SmaRTClock running, 32.768 kHz crystal)	1.8 V, T = 25 °C 3.0 V, T = 25 °C 3.6 V, T = 25 °C 1.8 V, T = 85 °C 3.0 V, T = 85 °C 3.6 V, T = 85 °C (includes SmaRTClock oscillator and V_{BAT} Supply Monitor)	— — — — — —	0.35 0.55 0.60 1.56 2.38 2.79	— — — — — —	µA
Digital Supply Current (Sleep Mode, SmaRTClock running, internal LFO)	1.8 V, T = 25 °C 3.0 V, T = 25 °C 3.6 V, T = 25 °C 1.8 V, T = 85 °C 3.0 V, T = 85 °C 3.6 V, T = 85 °C (includes SmaRTClock oscillator and V_{BAT} Supply Monitor)	— — — — — —	0.20 0.35 0.45 1.30 2.06 2.41	— — — — — —	µA
Digital Supply Current—Sleep Mode (LCD disabled, RTC disabled)					
Digital Supply Current (Sleep Mode)	1.8 V, T = 25 °C 3.0 V, T = 25 °C 3.6 V, T = 25 °C 1.8 V, T = 85 °C 3.0 V, T = 85 °C 3.6 V, T = 85 °C (includes POR supply monitor)	— — — — — —	0.05 0.08 0.12 1.2 2.2 2.4	— — 0.23 — — —	µA
Digital Supply Current (Sleep Mode, POR Supply Monitor Disabled)	1.8 V, T = 25 °C 3.0 V, T = 25 °C 3.6 V, T = 25 °C 1.8 V, T = 85 °C 3.0 V, T = 85 °C 3.6 V, T = 85 °C	— — — — — —	0.01 0.02 0.06 1.1 2.1 2.3	— — — — — —	µA

Notes:

1. Active Current measure using typical code loop - Digital Supply Current depends upon the particular code being executed. Digital Supply Current depends on the particular code being executed. The values in this table are obtained with the CPU executing a mix of instructions in two loops: djnz R1, $, followed by a loop that accesses an SFR, and moves data around using the CPU (between accumulator and b-register). The supply current will vary slightly based on the physical location of this code in flash. As described in the Flash Memory chapter, it is best to align the jump addresses with a flash word address (byte location /4), to minimize flash accesses and power consumption.
2. Includes oscillator and regulator supply current.
3. Based on device characterization data; Not production tested.
4. Measured with one-shot enabled.
5. Low-Power Idle mode current measured with CLKMODE = 0x04, PCON = 0x01, and PCLKEN = 0x0F.
6. Using SmaRTClock osillator with external 32.768 kHz CMOS clock. Does not include crystal bias current.
7. Low-Power Idle mode current measured with CLKMODE = 0x04, PCON = 0x01, and PCLKEN = 0x00.

Table 4.12. ADC0 Electrical Characteristics (Continued)

V_{BAT} = 1.8 to 3.8 V, VREF = 1.65 V (REFSL[1:0] = 11), –40 to +85 °C unless otherwise specified.

Parameter	Conditions	Min	Typ	Max	Units
Analog Inputs					
ADC Input Voltage Range	Single Ended (AIN+ – GND)	0	—	V_{REF}	V
Absolute Pin Voltage with respect to GND	Single Ended	0	—	V_{BAT}	V
Sampling Capacitance	1x Gain 0.5x Gain	—	16 13	—	pF
Input Multiplexer Impedance		—	5	—	kΩ
Power Specifications					
Power Supply Current (V_{BAT} supplied to ADC0)	Normal Power Mode: Conversion Mode (300 ksps) Tracking Mode (0 ksps) Low Power Mode: Conversion Mode (150 ksps) Tracking Mode (0 ksps)	 — — — —	 650 740 370 400	 — — — —	µA
Power Supply Rejection	Internal High Speed VREF External VREF	— —	67 74	— —	dB

1. INL and DNL specifications for 12-bit mode do not include the first or last four ADC codes.
2. The maximum code in 12-bit mode is 0xFFFC. The Full Scale Error is referenced from the maximum code.
3. Performance in 8-bit mode is similar to 10-bit mode.

Table 4.13. Temperature Sensor Electrical Characteristics

V_{BAT} = 1.8 to 3.8 V, –40 to +85 °C unless otherwise specified.

Parameter	Conditions	Min	Typ	Max	Units
Linearity		—	±1	—	°C
Slope		—	3.40	—	mV/°C
Slope Error*		—	40	—	µV/°C
Offset	Temp = 25 °C	—	1025	—	mV
Offset Error*	Temp = 25 °C	—	18	—	mV
Temperature Sensor Turn-On Time		—	1.7	—	µs
Supply Current		—	35	—	µA

***Note:** Represents one standard deviation from the mean.

5.3. 8-Bit Mode

Setting the ADC08BE bit in register ADC0CF to 1 will put the ADC in 8-bit mode.In 8-bit mode, only the 8 MSBs of data are converted, allowing the conversion to be completed in two fewer SAR clock cycles than a 10-bit conversion. This can result in an overall lower power consumption since the system can spend more time in a low power mode. The two LSBs of a conversion are always 00 in this mode, and the ADC0L register will always read back 0x00.

5.4. 12-Bit Mode

C8051F96x devices have an enhanced SAR converter that provides 12-bit resolution while retaining the 10- and 8-bit operating modes of the other devices in the family. When configured for 12-bit conversions, the ADC performs four 10-bit conversions using four different reference voltages and combines the results into a single 12-bit value. Unlike simple averaging techniques, this method provides true 12-bit resolution of ac or dc input signals without depending on noise to provide dithering. The converter also employs a hardware Dynamic Element Matching algorithm that reconfigures the largest elements of the internal DAC for each of the four 10-bit conversions to cancel the any matching errors, enabling the converter to achieve 12-bit linearity performance to go along with its 12-bit resolution. For best performance, the Low Power Oscillator should be selected as the system clock source while taking 12-bit ADC measurements.

The 12-bit mode is enabled by setting the AD012BE bit (ADC0AC.7) to logic 1 and configuring Burst Mode for four conversions as described in Section 5.2.3. The conversion can be initiated using any of the methods described in Section 5.2.1, and the 12-bit result will appear in the ADC0H and ADC0L registers. Since the 12-bit result is formed from a combination of four 10-bit results, the maximum output value is 4 x (1023) = 4092, rather than the max value of (2^12 – 1) = 4095 that is produced by a traditional 12-bit converter. To further increase resolution, the burst mode repeat value may be configured to any multiple of four conversions. For example, if a repeat value of 16 is selected, the ADC0 output will be a 14-bit number (sum of four 12-bit numbers) with 13 effective bits of resolution.

Figure 5.9 shows the typical temperature sensor error assuming a 1-point calibration at 25 °C. **Parameters that affect ADC measurement, in particular the voltage reference value, will also affect temperature measurement.**

A single-point offset measurement of the temperature sensor is performed on each device during production test. The measurement is performed at 25 °C ±5 °C, using the ADC with the internal high speed reference buffer selected as the Voltage Reference. The direct ADC result of the measurement is stored in the SFR registers TOFFH and TOFFL, shown in SFR Definition 5.13 and SFR Definition 5.14.

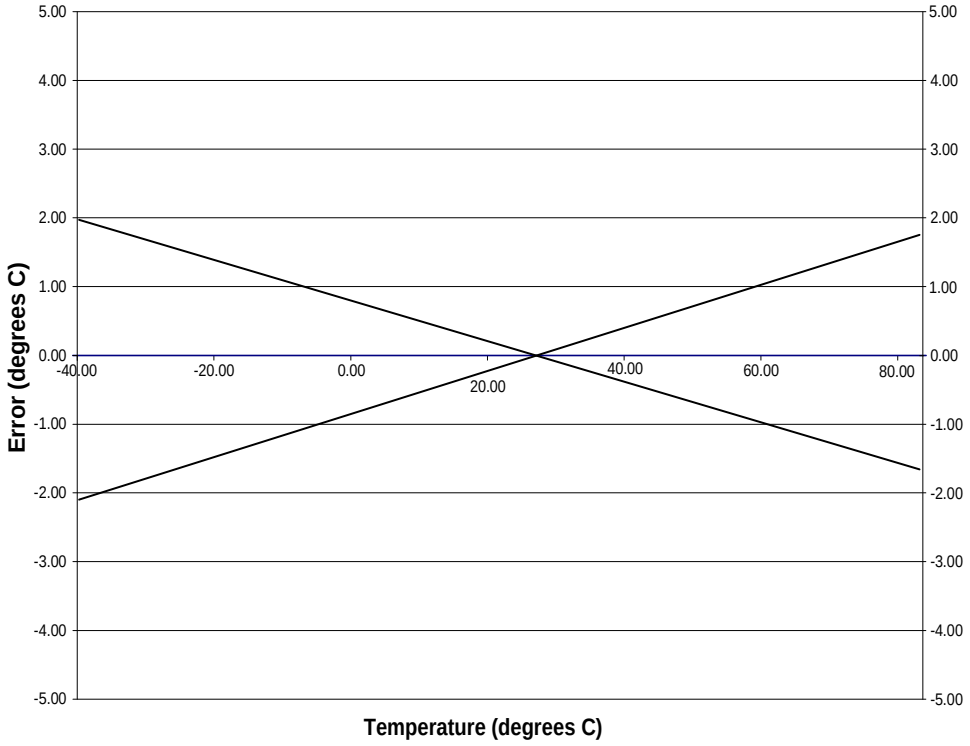

Figure 5.9. Temperature Sensor Error with 1-Point Calibration (V_{REF} = 1.68 V)

SFR Definition 5.15. REF0CN: Voltage Reference Control

Bit	7	6	5	4	3	2	1	0
Name			REFGND	REFSL		TEMPE		
Type	R	R	R/W	R/W	R/W	R/W	R	R
Reset	0	0	0	1	1	0	0	0

SFR Page = 0x0; SFR Address = 0xD1

Bit	Name	Function
7:6	Unused	Read = 00b; Write = Don't Care.
5	REFGND	**Analog Ground Reference.** Selects the ADC0 ground reference. 0: The ADC0 ground reference is the GND pin. 1: The ADC0 ground reference is the P0.1/AGND pin.
4:3	REFSL	**Voltage Reference Select.** Selects the ADC0 voltage reference. 00: The ADC0 voltage reference is the P0.0/VREF pin. 01: The ADC0 voltage reference is the VDD pin. 10: The ADC0 voltage reference is the internal 1.8 V digital supply voltage. 11: The ADC0 voltage reference is the internal 1.65 V high speed voltage reference.
2	TEMPE	**Temperature Sensor Enable.** Enables/Disables the internal temperature sensor. 0: Temperature Sensor Disabled. 1: Temperature Sensor Enabled.
1:0	Unused	Read = 00b; Write = Don't Care.

5.14. Voltage Reference Electrical Specifications

See Table 4.14 on page 72 for detailed Voltage Reference Electrical Specifications.

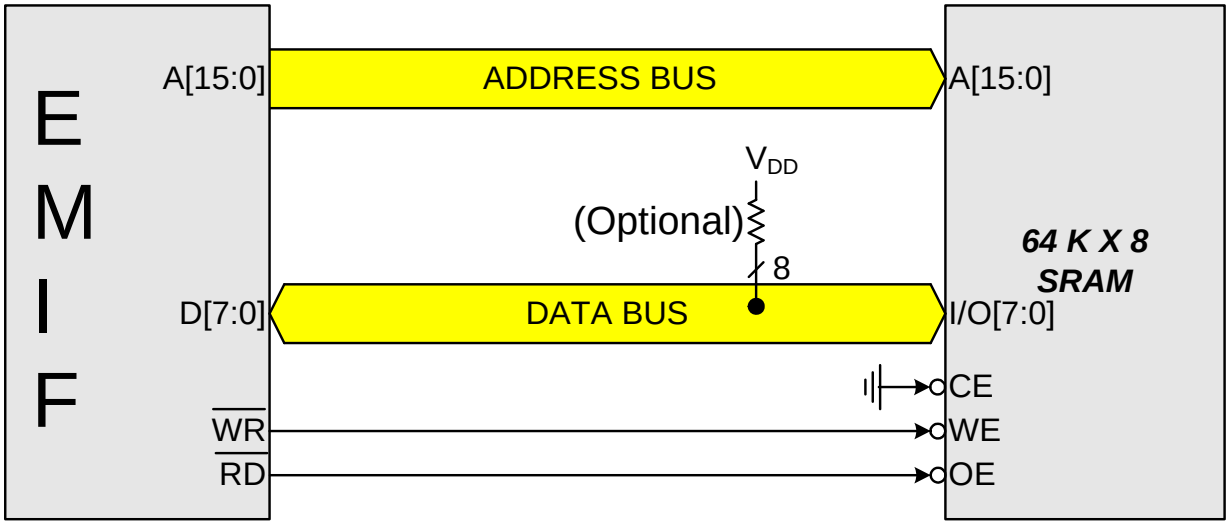

Figure 10.2. Non-multiplexed Configuration Example

10.5. Memory Mode Selection

The external data memory space can be configured in one of four modes, shown in Figure 10.3, based on the EMIF Mode bits in the EMI0CF register (SFR Definition 10.2). These modes are summarized below. More information about the different modes can be found in Section “10.6. Timing” on page 137.

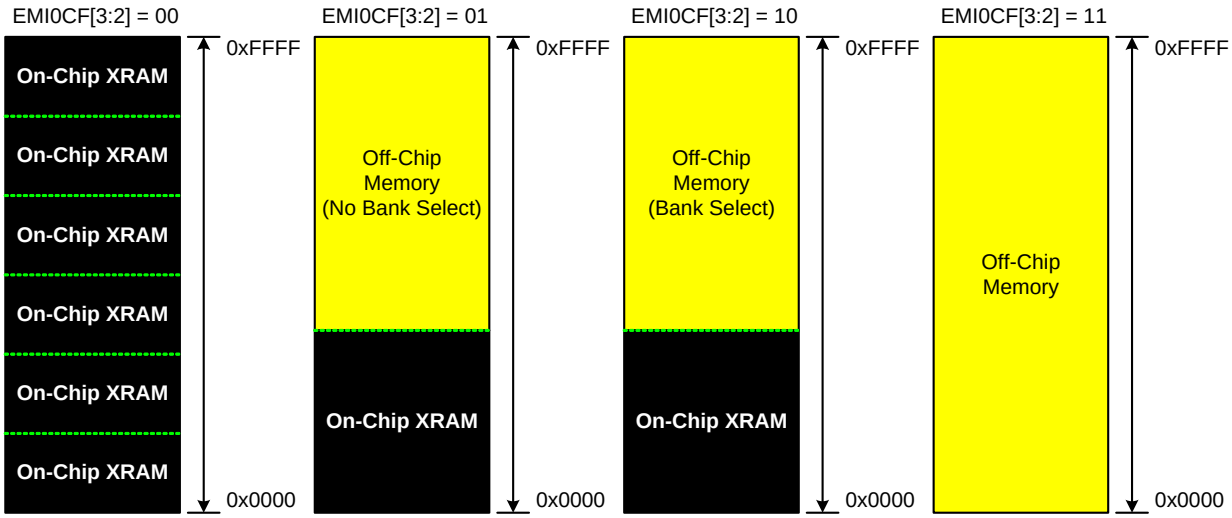

Figure 10.3. EMIF Operating Modes

Data transfer size DMA0NSZH:L defines the maximum number of bytes for the DMA0 transfer of the selected channel. If the address offset reaches data transfer size, the full-length interrupt flag bit CHn_INT (DMA0INT) of the selected channel will be asserted. Similarly, the mid-point interrupt flag bit CHn_MINT is set when the address offset is equal to half of data transfer size if the transfer size is an even number or when the address offset is equal to half of the transfer size plus one if the transfer size is an odd number. Interrupt flags must be cleared by software so that the next DMA0 data transfer can proceed.

The DMA0 subsystem permits data transfer between SFR registers and XRAM. The DMA0 subsystem executes its task based on settings of a channel's control and memory interface configuration SFRs. When data is copied from XRAM to SFR registers, it takes two cycles for DMA0 to read from XRAM and the SFR write occurs in the second cycle. If more than one byte is involved, a pipeline is used. When data is copied from SFR registers to XRAM, the DMA0 only requires one cycle for one byte transaction.

The selected DMA0 channel for a peripheral should be enabled through the enable bits CHn_EN (DMA0EN.n) to allow the DMA0 to transfer the data. When the DMA0 is transferring data on a channel, the busy status bit of the channel CHn_BUSY (DMA0BUSY.n) is set. During the transaction, writes to DMA0NSZH:L, DMA0NBAH:L, and DMA0NAOH:L are disabled.

Each peripheral is responsible for asserting the peripheral transfer requests necessary to service the particular peripheral. Some peripherals may have a complex state machine to manage the peripheral requests. Please refer to the DMA enabled peripheral chapters for additional information (AES0, CRC1, ENC0 and SPI1).

Besides reporting transaction status of a channel, DMA0BUSY can be used to force a DMA0 transfer on an already configured channel by setting the CHn_BUSY bit (DMA0BUSY.n).

The DMA0NMD sfr has a wrap bit that supports address offset wrapping. The size register DMA0NSZ sets the transfer size. Normally the address offset starts at zero and increases until it reaches size minus one. At this point the transfer is complete and the interrupt bit will be set. When the wrap bit is set, the address offset will automatically be reset to zero and transfers will continue as long as the peripheral keeps requesting data.

The wrap feature can be used to support key wrapping for the AES0 module. Normally the same key is used over and over with additional data blocks. So the wrap bit should be set when using the XRAM to AES0KIN request. This feature supports multiple-block encryption operations.

11.2. DMA0 Arbitration

11.2.1. DMA0 Memory Access Arbitration

If both DMA0 and CPU attempt to access SFR register or XRAM at the same time, the CPU pre-empts the DMA0 module. DMA0 will be stalled until CPU completes its bus activity.

11.2.2. DMA0 Channel Arbitration

Multiple DMA0 channels can request transfer simultaneously, but only one DMA0 channel will be granted the bus to transfer the data. Channel 0 has the highest priority. DMA0 channels are serviced based on their priority. A higher priority channel is serviced first. Channel arbitration occurs at the end of the data transfer granularity (transaction boundary) of the DMA. When there is a DMA0 request at the transaction boundary from higher priority channel, lower priority ones will be stalled until the highest priority one completes its transaction. So, for 16-bit transfers, the transaction boundary is at every 2 bytes.

11.3. DMA0 Operation in Low Power Modes

DMA0 remains functional in normal active, low power active, idle, low power idle modes but not in sleep or suspend mode. CPU will wait for DMA0 to complete all pending requests before it enters sleep mode. When the system wakes up from suspend or sleep mode to normal active mode, pending DMA0 interrupts will be serviced according to priority of channels. DMA0 stalls when CPU is in debug mode.

SFR Definition 11.2. DMA0INT: DMA0 Full-Length Interrupt

Bit	7	6	5	4	3	2	1	0
Name		CH6_INT	CH5_INT	CH4_INT	CH3_INT	CH2_INT	CH1_INT	CH0_INT
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Page = 0x2; SFR Address = 0xD3

Bit	Name	Function
7	Unused	Read = 0b, Write = Don't Care
6	CH6_INT	**Channel 6 Full-Length Interrupt Flag.[1]** 0: Full-length interrupt has not occured on channel 6. 1: Full-length interrupt has not occured on channel 6.
5	CH5_INT	0: Full-length interrupt has not occured on channel 5. 1: Full-length interrupt has not occured on channel 5.
4	CH4_INT	**Channel 4 Full-Length Interrupt Flag.[1]** 0: Full-length interrupt has not occured on channel 4. 1: Full-length interrupt has not occured on channel 4.
3	CH3_INT	**Channel 3 Full-Length Interrupt Flag.[1]** 0: Full-length interrupt has not occured on channel 3. 1: Full-length interrupt has not occured on channel 3.
2	CH2_INT	**Channel 2 Full-Length Interrupt Flag.[1]** 0: Full-length interrupt has not occured on channel 2. 1: Full-length interrupt has not occured on channel 2.
1	CH1_INT	**Channel 1 Full-Length Interrupt Flag.[1]** 0: Full-length interrupt has not occured on channel 1. 1: Full-length interrupt has not occured on channel 1.
0	CH0_INT	**Channel 0 Full-Length Interrupt Flag.[1]** 0: Full-length interrupt has not occured on channel 0. 1: Full-length interrupt has not occured on channel 0.

Note: 1.Full-length interrupt flag is set when the offset address DMA0NAOH/L is equals to data transfer size DMA0NSZH/L minus 1. This flag must be cleared by software or system reset. The full-length interrupt is enabled by setting bit 7 of DMA0NCF with DMA0SEL configured for the corresponding channel.

SFR Definition 16.3. SFRNEXT: SFR Next

Bit	7	6	5	4	3	2	1	0
Name	SFRNEXT[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

;SFR Page = All Pages; SFR Address = 0x85

Bit	Name	Function
7:0	SFRNEXT[7:0]	**SFR Page Bits.** This is the value that will go to the SFR Page register upon a return from interrupt. Write: Sets the SFR Page contained in the second byte of the SFR Stack. This will cause the SFRPAGE SFR to have this SFR page value upon a return from interrupt. Read: Returns the value of the SFR page contained in the second byte of the SFR stack. SFR page context is retained upon interrupts/return from interrupts in a 3 byte SFR Page Stack: SFRPAGE is the first entry, SFRNEXT is the second, and SFRLAST is the third entry. The SFR stack bytes may be used alter the context in the SFR Page Stack, and will not cause the stack to “push” or “pop”. Only interrupts and return from interrupts cause pushes and pops of the SFR Page Stack.

SFR Definition 20.1. DC0CN: DC-DC Converter Control

Bit	7	6	5	4	3	2	1	0
Name	CLKSEL	CLKDIV[1:0]		AD0CKINV	CLKINV	SYNC	MINPW[1:0]	
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	
Reset	0	0	0	0	0	0	1	1

SFR Page = 0x0; SFR Address = 0x97

Bit	Name	Function
7	CLKSEL	**DC-DC Converter Clock Source Select.** Specifies the dc-dc converter clock source. 0: The dc-dc converter is clocked from its local oscillator. 1: The dc-dc converter is clocked from the system clock.
6:5	CLKDIV[1:0]	**DC-DC Clock Divider.** Divides the dc-dc converter clock when the system clock is selected as the clock source for dc-dc converter. Ignored all other times. 00: The dc-dc converter clock is system clock divided by 1. 01: The dc-dc converter clock is system clock divided by 2. 10: The dc-dc converter clock is system clock divided by 4. 11: The dc-dc converter clock is system clock divided by 8.
4	AD0CKINV	**ADC0 Clock Inversion (Clock Invert During Sync).** Inverts the ADC0 SAR clock derived from the dc-dc converter clock when the SYNC bit (DC0CN.3) is enabled. This bit is ignored when the SYNC bit is set to zero. 0: ADC0 SAR clock is inverted. 1: ADC0 SAR clock is not inverted.
3	CLKINV	**DC-DC Converter Clock Invert.** Inverts the system clock used as the input to the dc-dc clock divider. 0: The dc-dc converter clock is not inverted. 1: The dc-dc converter clock is inverted.
2	SYNC	**ADC0 Synchronization Enable.** When synchronization is enabled, the ADC0SC[4:0] bits in the ADC0CF register must be set to 00000b. 0: The ADC is not synchronized to the dc-dc converter. 1: The ADC is synchronized to the dc-dc converter. ADC0 tracking is performed during the longest quiet time of the dc-dc converter switching cycle and ADC0 SAR clock is also synchronized to the dc-dc converter switching cycle.
1:0	MINPW[1:0]	**DC-DC Converter Minimum Pulse Width.** Specifies the minimum pulse width. 00: Minimum pulse detection logic is disabled (no pulse skipping). 01: Minimum pulse width is 10 ns. 10: Minimum pulse width is 20 ns. 11: Minimum pulse width is 40 ns.

Internal Register Definition 24.8. CAPTUREn: SmaRTClock Timer Capture

Bit	7	6	5	4	3	2	1	0
Name	CAPTURE[31:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SmaRTClock Addresses: CAPTURE0 = 0x00; CAPTURE1 = 0x01; CAPTURE2 =0x02; CAPTURE3: 0x03.

Bit	Name	Function
7:0	CAPTURE[31:0]	**SmaRTClock Timer Capture.** These 4 registers (CAPTURE3–CAPTURE0) are used to read or set the 32-bit SmaRTClock timer. Data is transferred to or from the SmaRTClock timer when the RTC0SET or RTC0CAP bits are set.
Note: The least significant bit of the timer capture value is CAPTURE0.0.		

Internal Register Definition 24.9. ALARM0Bn: SmaRTClock Alarm 0 Match Value

Bit	7	6	5	4	3	2	1	0
Name	ALARM0[31:0]							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SmaRTClock Address: ALARM0B0 = 0x08; ALARM0B1 = 0x09; ALARM0B2 = 0x0A; ALARM0B3 = 0x0B

Bit	Name	Function
7:0	ALARM0[31:0]	**SmaRTClock Alarm 0 Programmed Value.** These 4 registers (ALARM0B3–ALARM0B0) are used to set an alarm event for the SmaRTClock timer. The SmaRTClock alarm should be disabled (ALRM0EN=0) when updating these registers.
Note: The least significant bit of the alarm programmed value is ALARM0B0.0.		

SFR Definition 25.6. PC0DCL: PC0 Debounce Configuration Low

Bit	7	6	5	4	3	2	1	0
Name	PC0DCL[7:0]							
Type	R/W							
Reset	0	0	0	0	0	1	0	0

SFR Address = 0xF9; SFR Page = 0x2

Bit	Name	Function
7:0	PC0DCL[7:0]	**Pulse Counter Debounce Low** Number of cumulative good samples seen by the integrator before recognizing the input as low. Setting PC0DCL to 0x00 will disable integrators on both PC0 and PC1. The actual value used is PC0DCL plus one. Sampling a low decrements while sampling a high increments the count. Switch bounce produces a random looking signal. The worst case would be to bounce high at each sample point and not start decrementing the integrator until the switch bounce settled. Therefore, minimum pulse width should account for twice the debounce time. For example, using a sample rate of 1 ms and a PC0DCL value of 0x09 will look for 10 cumulative lows before recognizing the input as low (1 ms x 10 = 10 ms). The minimum pulse width should be 20 ms or greater for this example. If PC0DCL has a value of 0x03 and the sample rate is 500 µs, the integrator would need to see 4 cumulative lows before recognizing the low (500 µs x 4 = 2 ms). The minimum pulse width should be 4 ms for this example.

SFR Definition 27.1. XBR0: Port I/O Crossbar Register 0

Bit	7	6	5	4	3	2	1	0
Name	CP1AE	CP1E	CP0AE	CP0E	SYSCKE	SMB0E	SPI0E	URT0E
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Page = 0x0; SFR Address = 0xE1

Bit	Name	Function
7	CP1AE	**Comparator1 Asynchronous Output Enable.** 0: Asynchronous CP1 output unavailable at Port pin. 1: Asynchronous CP1 output routed to Port pin.
6	CP1E	**Comparator1 Output Enable.** 0: CP1 output unavailable at Port pin. 1: CP1 output routed to Port pin.
5	CP0AE	**Comparator0 Asynchronous Output Enable.** 0: Asynchronous CP0 output unavailable at Port pin. 1: Asynchronous CP0 output routed to Port pin.
4	CP0E	**Comparator0 Output Enable.** 0: CP1 output unavailable at Port pin. 1: CP1 output routed to Port pin.
3	SYSCKE	**$\overline{\text{SYSCLK}}$ Output Enable.** 0: $\overline{\text{SYSCLK}}$ output unavailable at Port pin. 1: $\overline{\text{SYSCLK}}$ output routed to Port pin.
2	SMB0E	**SMBus I/O Enable.** 0: SMBus I/O unavailable at Port pin. 1: SDA and SCL routed to Port pins.
1	SPI0E	**SPI0 I/O Enable** 0: SPI0 I/O unavailable at Port pin. 1: SCK, MISO, and MOSI (for SPI0) routed to Port pins. NSS (for SPI0) routed to Port pin only if SPI0 is configured to 4-wire mode.
0	URT0E	**UART0 Output Enable.** 0: UART I/O unavailable at Port pin. 1: TX0 and RX0 routed to Port pins P0.4 and P0.5.
Note: SPI0 can be assigned either 3 or 4 Port I/O pins.		

SFR Definition 27.24. P3MDIN: Port3 Input Mode

Bit	7	6	5	4	3	2	1	0
Name	P3MDIN[7:0]							
Type	R/W							
Reset	1	1	1	1	1	1	1	1

SFR Page = 0xF; SFR Address = 0xF1

Bit	Name	Function
7:0	P3MDIN[3:0]	**Analog Configuration Bits for P3.7–P3.0 (respectively).** Port pins configured for analog mode have their weak pullup and digital receiver disabled. The digital driver is not explicitly disabled. 0: Corresponding P3.n pin is configured for analog mode. 1: Corresponding P3.n pin is not configured for analog mode.

SFR Definition 27.25. P3MDOUT: Port3 Output Mode

Bit	7	6	5	4	3	2	1	0
Name	P3MDOUT[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Page = 0xF; SFR Address = 0xB1

Bit	Name	Function
7:0	P3MDOUT[7:0]	**Output Configuration Bits for P3.7–P3.0 (respectively).** These bits control the digital driver even when the corresponding bit in register P3MDIN is logic 0. 0: Corresponding P3.n Output is open-drain. 1: Corresponding P3.n Output is push-pull.

28.4.4. Data Register

The SMBus Data register SMB0DAT holds a byte of serial data to be transmitted or one that has just been received. Software may safely read or write to the data register when the SI flag is set. Software should not attempt to access the SMB0DAT register when the SMBus is enabled and the SI flag is cleared to logic 0, as the interface may be in the process of shifting a byte of data into or out of the register.

Data in SMB0DAT is always shifted out MSB first. After a byte has been received, the first bit of received data is located at the MSB of SMB0DAT. While data is being shifted out, data on the bus is simultaneously being shifted in. SMB0DAT always contains the last data byte present on the bus. In the event of lost arbitration, the transition from master transmitter to slave receiver is made with the correct data or address in SMB0DAT.

SFR Definition 28.5. SMB0DAT: SMBus Data

Bit	7	6	5	4	3	2	1	0
Name	SMB0DAT[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Page = 0x0; SFR Address = 0xC2

Bit	Name	Function
7:0	SMB0DAT[7:0]	**SMBus Data.** The SMB0DAT register contains a byte of data to be transmitted on the SMBus serial interface or a byte that has just been received on the SMBus serial interface. The CPU can read from or write to this register whenever the SI serial interrupt flag (SMB0CN.0) is set to logic 1. The serial data in the register remains stable as long as the SI flag is set. When the SI flag is not set, the system may be in the process of shifting data in/out and the CPU should not attempt to access this register.

28.5. SMBus Transfer Modes

The SMBus interface may be configured to operate as master and/or slave. At any particular time, it will be operating in one of the following four modes: Master Transmitter, Master Receiver, Slave Transmitter, or Slave Receiver. The SMBus interface enters Master Mode any time a START is generated, and remains in Master Mode until it loses an arbitration or generates a STOP. An SMBus interrupt is generated at the end of all SMBus byte frames. Note that the position of the ACK interrupt when operating as a receiver depends on whether hardware ACK generation is enabled. As a receiver, the interrupt for an ACK occurs **before** the ACK with hardware ACK generation disabled, and **after** the ACK when hardware ACK generation is enabled. As a transmitter, interrupts occur **after** the ACK, regardless of whether hardware ACK generation is enabled or not.

28.5.1. Write Sequence (Master)

During a write sequence, an SMBus master writes data to a slave device. The master in this transfer will be a transmitter during the address byte, and a transmitter during all data bytes. The SMBus interface generates the START condition and transmits the first byte containing the address of the target slave and the data direction bit. In this case the data direction bit (R/W) will be logic 0 (WRITE). The master then transmits one or more bytes of serial data. After each byte is transmitted, an acknowledge bit is generated by the slave. The transfer is ended when the STO bit is set and a STOP is generated. Note that the interface will switch to Master Receiver Mode if SMB0DAT is not written following a Master Transmitter interrupt.

30.1. Signal Descriptions

The four signals used by SPI0 (MOSI, MISO, SCK, NSS) are described below.

30.1.1. Master Out, Slave In (MOSI)

The master-out, slave-in (MOSI) signal is an output from a master device and an input to slave devices. It is used to serially transfer data from the master to the slave. This signal is an output when SPI0 is operating as a master and an input when SPI0 is operating as a slave. Data is transferred most-significant bit first. When configured as a master, MOSI is driven by the MSB of the shift register in both 3- and 4-wire mode.

30.1.2. Master In, Slave Out (MISO)

The master-in, slave-out (MISO) signal is an output from a slave device and an input to the master device. It is used to serially transfer data from the slave to the master. This signal is an input when SPI0 is operating as a master and an output when SPI0 is operating as a slave. Data is transferred most-significant bit first. The MISO pin is placed in a high-impedance state when the SPI module is disabled and when the SPI operates in 4-wire mode as a slave that is not selected. When acting as a slave in 3-wire mode, MISO is always driven by the MSB of the shift register.

30.1.3. Serial Clock (SCK)

The serial clock (SCK) signal is an output from the master device and an input to slave devices. It is used to synchronize the transfer of data between the master and slave on the MOSI and MISO lines. SPI0 generates this signal when operating as a master. The SCK signal is ignored by a SPI slave when the slave is not selected (NSS = 1) in 4-wire slave mode.

30.1.4. Slave Select (NSS)

The function of the slave-select (NSS) signal is dependent on the setting of the NSSMD1 and NSSMD0 bits in the SPI0CN register. There are three possible modes that can be selected with these bits:

1. NSSMD[1:0] = 00: 3-Wire Master or 3-Wire Slave Mode: SPI0 operates in 3-wire mode, and NSS is disabled. When operating as a slave device, SPI0 is always selected in 3-wire mode. Since no select signal is present, SPI0 must be the only slave on the bus in 3-wire mode. This is intended for point-to-point communication between a master and one slave.
2. NSSMD[1:0] = 01: 4-Wire Slave or Multi-Master Mode: SPI0 operates in 4-wire mode, and NSS is enabled as an input. When operating as a slave, NSS selects the SPI0 device. When operating as a master, a 1-to-0 transition of the NSS signal disables the master function of SPI0 so that multiple master devices can be used on the same SPI bus.
3. NSSMD[1:0] = 1x: 4-Wire Master Mode: SPI0 operates in 4-wire mode, and NSS is enabled as an output. The setting of NSSMD0 determines what logic level the NSS pin will output. This configuration should only be used when operating SPI0 as a master device.

See Figure 30.2, Figure 30.3, and Figure 30.4 for typical connection diagrams of the various operational modes. **Note that the setting of NSSMD bits affects the pinout of the device.** When in 3-wire master or 3-wire slave mode, the NSS pin will not be mapped by the crossbar. In all other modes, the NSS signal will be mapped to a pin on the device. See Section “27. Port Input/Output” on page 351 for general purpose port I/O and crossbar information.

30.2. SPI0 Master Mode Operation

A SPI master device initiates all data transfers on a SPI bus. SPI0 is placed in master mode by setting the Master Enable flag (MSTEN, SPI0CN.6). Writing a byte of data to the SPI0 data register (SPI0DAT) when in master mode writes to the transmit buffer. If the SPI shift register is empty, the byte in the transmit buffer is moved to the shift register, and a data transfer begins. The SPI0 master immediately shifts out the data serially on the MOSI line while providing the serial clock on SCK. The SPIF (SPI0CN.7) flag is set to logic 1 at the end of the transfer. If interrupts are enabled, an interrupt request is generated when the SPIF flag

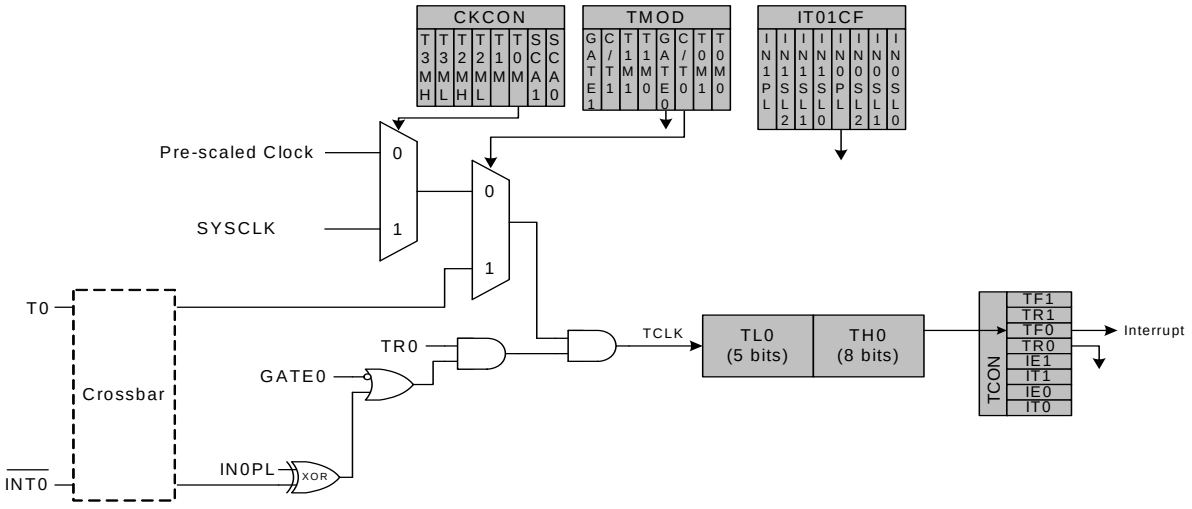

Figure 32.1. T0 Mode 0 Block Diagram

32.1.2. Mode 1: 16-bit Counter/Timer

Mode 1 operation is the same as Mode 0, except that the counter/timer registers use all 16 bits. The counter/timers are enabled and configured in Mode 1 in the same manner as for Mode 0.

32.1.3. Mode 2: 8-bit Counter/Timer with Auto-Reload

Mode 2 configures Timer 0 and Timer 1 to operate as 8-bit counter/timers with automatic reload of the start value. TL0 holds the count and TH0 holds the reload value. When the counter in TL0 overflows from all ones to 0x00, the timer overflow flag TF0 (TCON.5) is set and the counter in TL0 is reloaded from TH0. If Timer 0 interrupts are enabled, an interrupt will occur when the TF0 flag is set. The reload value in TH0 is not changed. TL0 must be initialized to the desired value before enabling the timer for the first count to be correct. When in Mode 2, Timer 1 operates identically to Timer 0.

Both counter/timers are enabled and configured in Mode 2 in the same manner as Mode 0. Setting the TR0 bit (TCON.4) enables the timer when either GATE0 (TMOD.3) is logic 0 or when the input signal INT0 is active as defined by bit IN0PL in register IT01CF (see Section “17.6. External Interrupts INT0 and INT1” on page 242 for details on the external input signals INT0 and INT1).