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, POR, PWM, WDT
Number of I/O	57
Program Memory Size	128KB (128K 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	80-TQFP
Supplier Device Package	80-TQFP (12x12)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f962-b-gqr

List of Registers

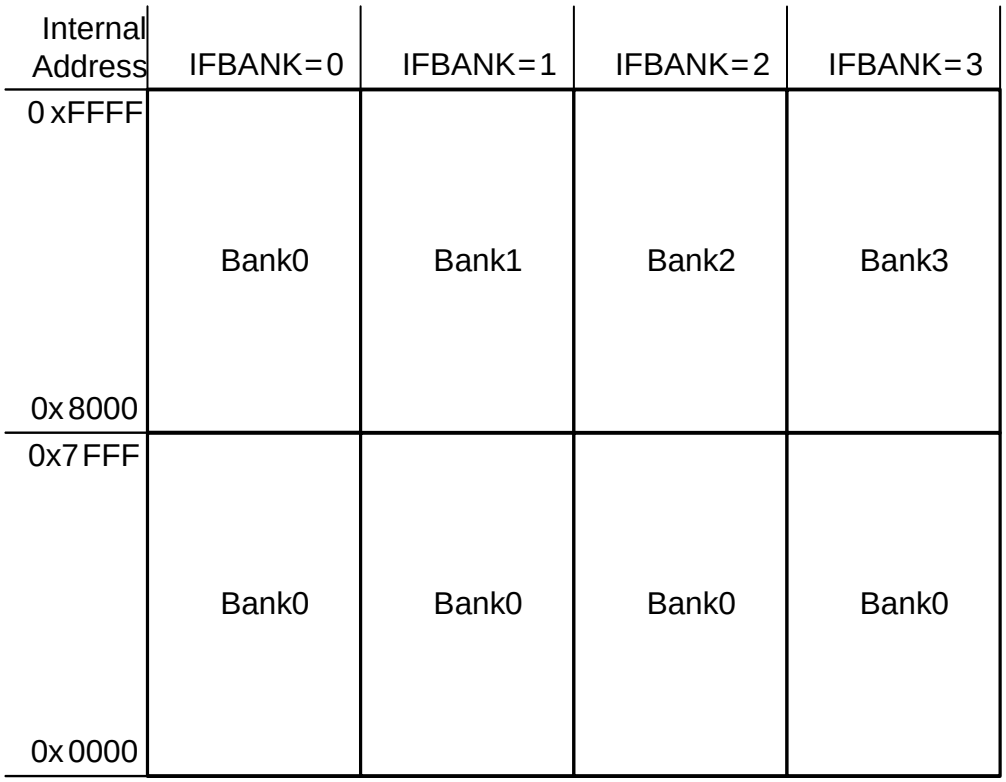

Figure 9.3. Address Memory Map for Instruction Fetches

SFR Definition 9.1. PSBANK: Program Space Bank Select

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

SFR Page = All Pages; SFR Address = 0x84

Bit	Name	Function
7:6	Reserved	Read = 00b, Must Write = 00b.
5:4	COBANK[1:0]	**Constant Operations Bank Select.** These bits select which flash bank is targeted during constant operations (MOVC and flash MOVX) involving address 0x8000 to 0xFFFF. 00: Constant Operations Target Bank 0 (note that Bank 0 is also mapped between 0x0000 to 0x7FFF). 01: Constant operations target Bank 1. 10: Constant operations target Bank 2. 11: Constant operations target Bank 3.
3:2	Reserved	Read = 00b, Must Write = 00b.
1:0	IFBANK[1:0]	**Instruction Fetch Operations Bank Select.** These bits select which flash bank is used for instruction fetches involving address 0x8000 to 0xFFFF. These bits can only be changed from code in Bank 0. 00: Instructions fetch from Bank 0 (note that Bank 0 is also mapped between 0x0000 to 0x7FFF). 01: Instructions fetch from Bank 1. 10: Instructions fetch from Bank 2. 11: Instructions fetch from Bank 3.

Note:
1. COBANK[1:0] and IFBANK[1:0] should not be set to (10b) or (11b) on the C8051F964/5/6/7/8/9 devices.
2. On devices with 64 kB of flash or less, keep PSBANK at its default setting of 0x11.

9.1.1. MOVX Instruction and Program Memory

The MOVX instruction in an 8051 device is typically used to access external data memory. On the C8051F96x devices, the MOVX instruction is normally used to read and write on-chip XRAM, but can be re-configured to write and erase on-chip flash memory space. MOVC instructions are always used to read flash memory, while MOVX write instructions are used to erase and write flash. This flash access feature provides a mechanism for the C8051F96x to update program code and use the program memory space for non-volatile data storage. Refer to Section “18. Flash Memory” on page 244 for further details.

9.2. Data Memory

The C8051F96x device family includes 8448 bytes (C8051F960/1/2/3/4/5/6/7) or 4352 bytes (C8051F968/9) of RAM data memory. 256 bytes of this memory is mapped into the internal RAM space of the 8051. 8192 or 4096 bytes of this memory is on-chip “external” memory. The data memory map is shown in Figure 9.1 for reference.

9.2.1. Internal RAM

SFR Definition 10.2. EMI0CF: External Memory Configuration

Bit	7	6	5	4	3	2	1	0
Name				EMD2	EMD[1:0]		EALE[1:0]	
Type	R/W							
Reset	0	0	0	0	0	0	1	1

SFR Page = 0x0; SFR Address = 0xAB

Bit	Name	Function
7:5	Unused	Read = 000b; Write = Don't Care.
4	EMD2	**EMIF Multiplex Mode Select Bit.** 0: EMIF operates in multiplexed address/data mode 1: EMIF operates in non-multiplexed mode (separate address and data pins)
3:2	EMD[1:0]	**EMIF Operating Mode Select Bits.** 00: Internal Only: MOVX accesses on-chip XRAM only. All effective addresses alias to on-chip memory space 01: Split Mode without Bank Select: Accesses below the 8 kB boundary are directed on-chip. Accesses above the 8 kB boundary are directed off-chip. 8-bit off-chip MOVX operations use current contents of the Address high port latches to resolve the upper address byte. To access off chip space, EMI0CN must be set to a page that is not contained in the on-chip address space. 10: Split Mode with Bank Select: Accesses below the 8 kB boundary are directed on-chip. Accesses above the 8 kB boundary are directed off-chip. 8-bit off-chip MOVX operations uses the contents of EMI0CN to determine the high-byte of the address. 11: External Only: MOVX accesses off-chip XRAM only. On-chip XRAM is not visible to the CPU.
1:0	EALE[1:0]	**ALE Pulse-Width Select Bits.** These bits only have an effect when EMD2 = 0. 00: ALE high and ALE low pulse width = 1 SYSCLK cycle. 01: ALE high and ALE low pulse width = 2 SYSCLK cycles. 10: ALE high and ALE low pulse width = 3 SYSCLK cycles. 11: ALE high and ALE low pulse width = 4 SYSCLK cycles.

12. Cyclic Redundancy Check Unit (CRC0)

C8051F96x devices include a cyclic redundancy check unit (CRC0) that can perform a CRC using a 16-bit or 32-bit polynomial. CRC0 accepts a stream of 8-bit data written to the CRC0IN register. CRC0 posts the 16-bit or 32-bit result to an internal register. The internal result register may be accessed indirectly using the CRC0PNT bits and CRC0DAT register, as shown in Figure 12.1. CRC0 also has a bit reverse register for quick data manipulation.

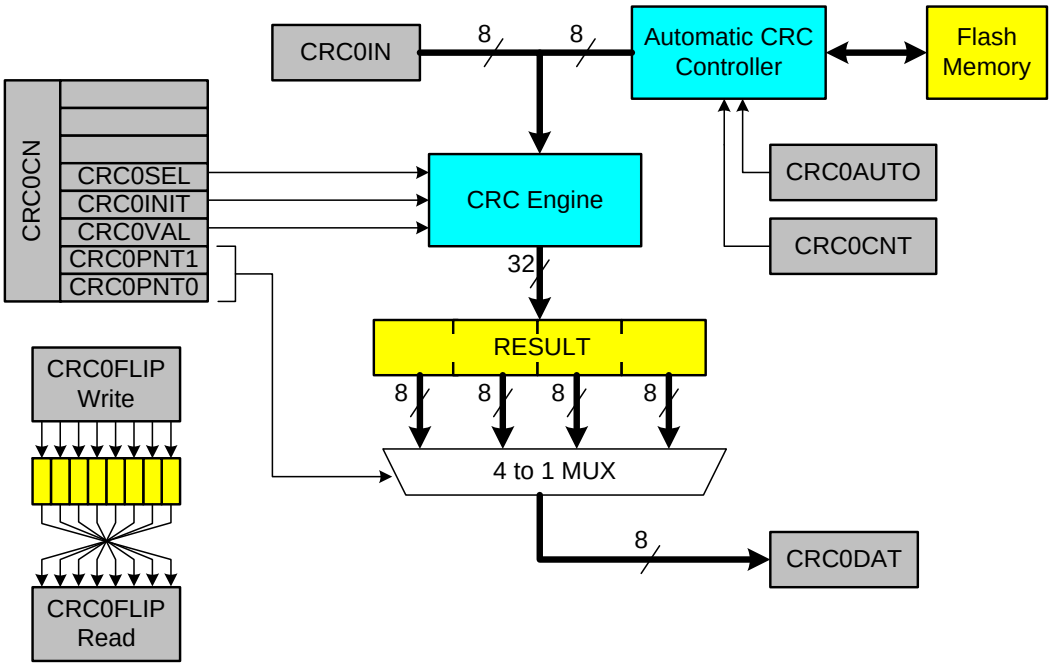

Figure 12.1. CRC0 Block Diagram

12.1. 16-bit CRC Algorithm

The C8051F96x CRC unit calculates the 16-bit CRC MSB-first, using a poly of 0x1021. The following describes the 16-bit CRC algorithm performed by the hardware:

1. XOR the most-significant byte of the current CRC result with the input byte. If this is the first iteration of the CRC unit, the current CRC result will be the set initial value (0x0000 or 0xFFFF).
   a. If the MSB of the CRC result is set, left-shift the CRC result, and then XOR the CRC result with the polynomial (0x1021).
   b. If the MSB of the CRC result is not set, left-shift the CRC result.
2. Repeat at Step 2a for the number of input bits (8).

SFR Definition 12.4. CRC0AUTO: CRC0 Automatic Control

Bit	7	6	5	4	3	2	1	0
Name	AUTOEN	CRCDONE	CRC0ST[5:0]					
Type	R/W							R/W
Reset	0	1	0	0	0	0	0	0

SFR Page = 0xF; SFR Address = 0x96

Bit	Name	Function
7	AUTOEN	**Automatic CRC Calculation Enable.** When AUTOEN is set to 1, any write to CRC0CN will initiate an automatic CRC starting at flash sector CRC0ST and continuing for CRC0CNT sectors.
6	CRCDONE	**CRCDONE Automatic CRC Calculation Complete.** Set to 0 when a CRC calculation is in progress. Note that code execution is stopped during a CRC calculation, therefore reads from firmware will always return 1.
5:0	CRC0ST[5:0]	**Automatic CRC Calculation Starting Flash Sector.** These bits specify the flash sector to start the automatic CRC calculation. The starting address of the first flash sector included in the automatic CRC calculation is CRC0ST x 1024. For 128 kB devices, pages 32–63 access the upper code bank as selected by the IFBANK bits in the PSBANK SFR.

SFR Definition 12.5. CRC0CNT: CRC0 Automatic Flash Sector Count

Bit	7	6	5	4	3	2	1	0
Name			CRC0CNT[5:0]					
Type	R/W							R/W
Reset	0	0	0	0	0	0	0	0

SFR Page = 0xF; SFR Address = 0x97

Bit	Name	Function
7:6	Unused	Read = 00b; Write = Don’t Care.
5:0	CRC0CNT[5:0]	**Automatic CRC Calculation Flash Sector Count.** These bits specify the number of flash sectors to include in an automatic CRC calculation. The starting address of the last flash sector included in the automatic CRC calculation is (CRC0ST+CRC0CNT) x 1024. The last page should not exceed page 63. Setting both CRC0ST and CRC0CNT to 0 will perform a CRC over the 64kB banked memory space.

SFR Definition 13.2. CRC1IN: CRC1 Data IN

Bit	7	6	5	4	3	2	1	0
Name	CRC1IN[7:0]							
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

SFR Page = 0x2; SFR Address = 0xB9; Not Bit-Addressable

Bit	Name	Function
7:0	CRC1IN[7:0]	**CRC1Data IN.** CRC Data should be sequentially written, one byte at a time, to the CRC1IN Data input SFR. When the CRC1 module is used with the DMA, the DMA will write directly to this SFR.

SFR Definition 13.3. CRC1POLL: CRC1 Polynomial LSB

Bit	7	6	5	4	3	2	1	0
Name	CRC1POLL[7:0]							
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

SFR Page = 0x2; SFR Address = 0xBC; Not Bit-Addressable

Bit	Name	Function
7:0	CRC1POLL[7:0]	**CRC1 Polynomial LSB.**

SFR Definition 13.4. CRC1POLH: CRC1 Polynomial MSB

Bit	7	6	5	4	3	2	1	0
Name	CRC1POLH[7:0]							
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

SFR Page = 0x2; SFR Address = 0xBD; Not Bit-Addressable

Bit	Name	Function
7:0	CRC1POLH[7:0]	**CRC1 Polynomial MSB.**

- Disable the DMA by writing 0x00 to DMA0EN
- Increment counter and repeat all steps for additional blocks

14.6.6.1. CTR Encryption using SFRs

- First Configure AES Module for CTR Block Cipher Mode Encryption
  - Reset AES module by writing 0x00 to AES0BCFG.
  - Configure the AES Module data flow for XOR on output data by writing 0x02 to the AES0DCFG sfr.
  - Write key size to bits 1 and 0 of the AES0BCFG.
  - Configure the AES core for encryption by setting bit 2 of AES0BCFG.
  - Enable the AES core by setting bit 3 of AES0BCFG.
- Repeat alternating write sequence 16 times
  - Write plaintext byte to AES0BIN.
  - Write counter byte to AES0XIN
  - Write encryption key byte to AES0KIN.
- Write remaining encryption key bytes to AES0KIN for 192-bit and 256-bit decryption only.
- Wait on AES done interrupt or poll bit 5 of AES0BCFG.
- Read 16 encrypted bytes from the AES0YOUT sfr.

If encrypting multiple blocks, increment the counter and repeat this process. It is not necessary reconfigure the AES module for each block.

SFR Definition 15.2. ENC0L: ENC0 Data Low Byte

Bit	7	6	5	4	3	2	1	0
Name	ENC0L[7:0]							
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

SFR Page = 0x2; SFR Address = 0xC2; Bit-Addressable

Bit	Name	Function
7:0	ENC0L[7:0]	**ENC0 Data Low Byte.**

SFR Definition 15.3. ENC0M: ENC0 Data Middle Byte

Bit	7	6	5	4	3	2	1	0
Name	ENC0M[7:0]							
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

SFR Page = 0x2; SFR Address = 0xC3; Bit-Addressable

Bit	Name	Function
7:0	ENC0M[7:0]	**ENC0 Data Middle Byte.**

SFR Definition 15.4. ENC0H: ENC0 Data High Byte

Bit	7	6	5	4	3	2	1	0
Name	ENC0H[7:0]							
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

SFR Page = 0x2; SFR Address = 0xC4; Bit-Addressable

Bit	Name	Function
7:0	ENC0H[7:0]	**ENC0 Data High Byte.**

18.6. Minimizing Flash Read Current

The flash memory in the C8051F96x devices is responsible for a substantial portion of the total digital supply current when the device is executing code. Below are suggestions to minimize flash read current.

1. Use idle, low power idle, suspend, or sleep modes while waiting for an interrupt, rather than polling the interrupt flag. Idle mode and low power idle mode is particularly well-suited for use in implementing short pauses, since the wake-up time is no more than three system clock cycles. See the Power Management chapter for details on the various low-power operating modes.
2. The flash memory is organized in 4-byte words starting with a byte with address ending in 00b and ending with a byte with address ending in 11b. A 4-byte pre-fetch buffer is used to read 4 bytes of flash in a single read operation. Short loops that straddle word boundaries or have an instruction byte with address ending in 11b should be avoided when possible. If a loop executes in 20 or more clock cycles, any resulting increase in operating current due to mis-alignment will be negligible.
3. To minimize the power consumption of small loops, it is best to locate them such that the number of 4-byte words to be fetched from flash is minimized. Consider a 2-byte, 3-cycle loop (e.g., SJMP $, or while(1);). The flash read current of such a loop will be minimized if both address bytes are contained in the first 3 bytes of a single 4-byte word. Such a loop should be manually located at an address ending in 00b or the number of bytes in the loop should be increased (by padding with NOP instructions) in order to minimize flash read current.

SFR Definition 20.3. DC0MD: DC-DC Converter Mode

Bit	7	6	5	4	3	2	1	0
Name	Reserved	ILIMIT			FORBYP	AUTOBYP	Reserved	DC0EN
Type	R/W	R/W			R/W	R/W	R/W	R/W
Reset	0	1	0	1	0	0	0	0

SFR Page = 0x2; SFR Address = 0xB3

Bit	Name	Function
7	Reserved	Read = 0b; Must write 0b.
6:4	ILIMIT	**Peak Current Limit Threshold.** 000: Reserved 001: Peak Inductor current is limited to 200 mA 010: Peak Inductor current is limited to 300 mA 011: Peak Inductor current is limited to 400 mA 100: Peak Inductor current is limited to 500 mA 101: Peak Inductor current is limited to 600 mA 110: Reserved 111: Reserved
3	FORBYP	**Enable Forced Bypass Mode.** 0: Forced bypass mode is disabled. 1: Forced bypass mode is enabled.
2	AUTOBYP	**Enable Automatic Bypass Mode.** 0: Automatic Bypass mode is disabled. 1: Automatic bypass mode is enabled.
1	Reserved	Read = 1b; Must write 1b.
0	DC0EN	**DC-DC Converter Enable.** 0: DC-DC converter is disabled. 1: DC-DC converter is enabled.

26.3. LCD Contrast Adjustment

The LCD Bias voltages which determine the LCD contrast are generated using the VBAT supply voltage or the on-chip charge pump. There are four contrast control modes to accomodate a wide variety of applications and supply voltages. The target contrast voltage is programmable in 60 mV steps from 1.9 to 3.72 V. The LCD contrast voltage is controlled by the LCD0CNTRST register and the contrast control mode is selected by setting the appropriate bits in the LCD0MSCN, LCD0MSCF, LCD0PWR, and LCD0VBMCN registers.

Note: An external 10 µF decoupling capacitor is required on the VLCD pin to create a charge reservoir at the output of the charge pump.

Table 26.1. Bit Configurations to select Contrast Control Modes

Mode	LCD0MSCN.2	LCD0MSCF.0	LCD0PWR.3	LCD0VBMCN.7
1	0	1	0	0
2	0	1	1	1
3	1*	0	1	1
4	1*	0	0	1
*** May be set to 0 to support increased load currents.**				

26.3.1. Contrast Control Mode 1 (Bypass Mode)

In Contrast Control Mode 1, the contrast control circuitry is disabled and the VLCD voltage follows the VBAT supply voltage, as shown in Figure 26.3. This mode is useful in systems where the VBAT voltage always remains constant and will provide the lowest LCD power consumption. Bypass Mode is selected using the following procedure:

1. Clear Bit 2 of the LCD0MSCN register to 0b (LCD0MSCN &= ~0x04)
2. Set Bit 0 of the LCD0MSCF register to 1b (LCD0MSCF |= 0x01)
3. Clear Bit 3 of the LCD0PWR register to 0b (LCD0PWR &= ~0x08)
4. Clear Bit 7 of the LCD0VBMCN register to 0b (LCD0VBMCN &= ~0x80)

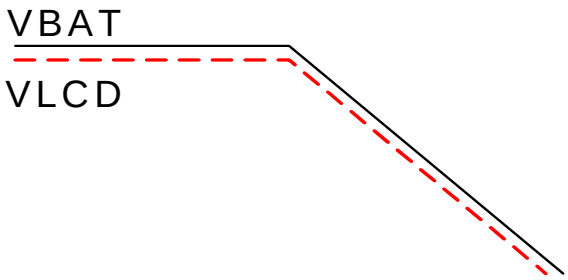

Figure 26.3. Contrast Control Mode 1

SFR Definition 26.14. LCD0CHPCF: LCD0 Charge Pump Configuration

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

SFR Page = 0x2; SFR Address = 0xAD

Bit	Name	Function
7:0	Reserved	Must write 0x60.

SFR Definition 26.15. LCD0CHPMD: LCD0 Charge Pump Mode

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

SFR Page = 0x2; SFR Address = 0xAE

Bit	Name	Function
7:0	Reserved	Must write 0xE9.

SFR Definition 26.16. LCD0BUFCN: LCD0 Buffer Control

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

SFR Page = 0xF; SFR Address = 0x9C

Bit	Name	Function
7:0	Reserved	Must write 0x44.

SFR Definition 27.16. P1MDOUT: Port1 Output Mode

Bit	7	6	5	4	3	2	1	0
Name	P1MDOUT[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Page = 0x0; SFR Address = 0xA5

Bit	Name	Function
7:0	P1MDOUT[7:0]	**Output Configuration Bits for P1.7–P1.0 (respectively).** These bits control the digital driver even when the corresponding bit in register P1MDIN is logic 0. 0: Corresponding P1.n Output is open-drain. 1: Corresponding P1.n Output is push-pull.

SFR Definition 27.17. P1DRV: Port1 Drive Strength

Bit	7	6	5	4	3	2	1	0
Name	P1DRV[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Page = 0xF; SFR Address = 0xA5

Bit	Name	Function
7:0	P1DRV[7:0]	**Drive Strength Configuration Bits for P1.7–P1.0 (respectively).** Configures digital I/O Port cells to high or low output drive strength. 0: Corresponding P1.n Output has low output drive strength. 1: Corresponding P1.n Output has high output drive strength.

Table 28.6. SMBus Status Decoding With Hardware ACK Generation Enabled (EHACK = 1)

Mode	Values Read				Current SMbus State	Typical Response Options	Values to Write			Next Status Vector Expected
	Status Vector	ACKRQ	ARBLOST	ACK			STA	STO	ACK	
Master Transmitter	1110	0	0	X	A master START was generated.	Load slave address + R/W into SMB0DAT.	0	0	X	1100
	1100	0	0	0	A master data or address byte was transmitted; NACK received.	Set STA to restart transfer.	1	0	X	1110
						Abort transfer.	0	1	X	-
		0	0	1	A master data or address byte was transmitted; ACK received.	Load next data byte into SMB0DAT.	0	0	X	1100
						End transfer with STOP.	0	1	X	-
						End transfer with STOP and start another transfer.	1	1	X	-
						Send repeated START.	1	0	X	1110
						Switch to Master Receiver Mode (clear SI without writing new data to SMB0DAT). Set ACK for initial data byte.	0	0	1	1000
Master Receiver	1000	0	0	1	A master data byte was received; ACK sent.	Set ACK for next data byte; Read SMB0DAT.	0	0	1	1000
						Set NACK to indicate next data byte as the last data byte; Read SMB0DAT.	0	0	0	1000
						Initiate repeated START.	1	0	0	1110
						Switch to Master Transmitter Mode (write to SMB0DAT before clearing SI).	0	0	X	1100
		0	0	0	A master data byte was received; NACK sent (last byte).	Read SMB0DAT; send STOP.	0	1	0	-
						Read SMB0DAT; Send STOP followed by START.	1	1	0	1110
						Initiate repeated START.	1	0	0	1110
						Switch to Master Transmitter Mode (write to SMB0DAT before clearing SI).	0	0	X	1100

SFR Definition 29.1. SCON0: Serial Port 0 Control

Bit	7	6	5	4	3	2	1	0
Name	S0MODE		MCE0	REN0	TB80	RB80	TI0	RI0
Type	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	1	0	0	0	0	0	0

SFR Page = 0x0; SFR Address = 0x98; Bit-Addressable

Bit	Name	Function
7	S0MODE	**Serial Port 0 Operation Mode.** Selects the UART0 Operation Mode. 0: 8-bit UART with Variable Baud Rate. 1: 9-bit UART with Variable Baud Rate.
6	Unused	Read = 1b. Write = Don't Care.
5	MCE0	**Multiprocessor Communication Enable.** **For Mode 0 (8-bit UART): Checks for valid stop bit.** 0: Logic level of stop bit is ignored. 1: RI0 will only be activated if stop bit is logic level 1. **For Mode 1 (9-bit UART): Multiprocessor Communications Enable.** 0: Logic level of ninth bit is ignored. 1: RI0 is set and an interrupt is generated only when the ninth bit is logic 1.
4	REN0	**Receive Enable.** 0: UART0 reception disabled. 1: UART0 reception enabled.
3	TB80	**Ninth Transmission Bit.** The logic level of this bit will be sent as the ninth transmission bit in 9-bit UART Mode (Mode 1). Unused in 8-bit mode (Mode 0).
2	RB80	**Ninth Receive Bit.** RB80 is assigned the value of the STOP bit in Mode 0; it is assigned the value of the 9th data bit in Mode 1.
1	TI0	**Transmit Interrupt Flag.** Set by hardware when a byte of data has been transmitted by UART0 (after the 8th bit in 8-bit UART Mode, or at the beginning of the STOP bit in 9-bit UART Mode). When the UART0 interrupt is enabled, setting this bit causes the CPU to vector to the UART0 interrupt service routine. This bit must be cleared manually by software.
0	RI0	**Receive Interrupt Flag.** Set to 1 by hardware when a byte of data has been received by UART0 (set at the STOP bit sampling time). When the UART0 interrupt is enabled, setting this bit to 1 causes the CPU to vector to the UART0 interrupt service routine. This bit must be cleared manually by software.

31.9. Master Mode Unidirectional Data Transfer

A unidirectional SPI master mode DMA transfer will transfer a specified number of bytes out on the MOSI pin. The MOSI data must be stored in XRAM before initiating the DMA transfers. The SPI1DAT-to-XRAM peripheral request is not used. Since the DMA does not read the SPI1DAT SFR, the SPI will discard the MISO data.

A unidirectional transfer only requires one DMA channel to transfer XRAM data to the SPI1DAT SFR. The DMA interrupt will indicate the completion of the data transfer to the SPI1DAT SFR. When the interrupt occurs, the DMA has written all of the data to the SPI1DAT SFR, but the SPI has not transmitted the last byte. Firmware may poll on the SPIBSY bit to determine when the SPI has transmitted the last byte. Firmware should not deassert the NSS pin until after the SPI has transmitted the last byte.

To initiate a master mode unidirectional data transfer:

1. Configure the SPI1 SFRs normally for Master mode.
   a. Enable Master mode by setting bit 6 in SPI1CFG.
   b. Configure the clock polarity CKPOL and clock phase CKPHA as desired in SPI1CFG.
   c. Configure SPI1CKR for the desired SPI clock rate.
   d. Configure the desired 4-wire master or 3-wire master mode in SPI1CN.
   e. Enable the SPI by setting bit 0 of SPI1CN.
2. Configure the desired DMA channel for the XRAM-to-SPI1DAT transfer.
   a. Disable the desired DMA channel by clearing the corresponding bit in DMA0EN.
   b. Select the desired DMA channel by writing to DMA0SEL.
   c. Configure the selected DMA channel to use the XRAM-to-SPI1DAT XRAM peripheral request by writing 0x03 to DMA0NCF.
   d. Enable DMA interrupts for the desired channel by setting bit 7 of DMA0NCF.
   e. Write 0 to DMA0NMD to disable wrapping.
   f. Write the address for the first byte of master output (MOSI) data to DMA0NBAH:L.
   g. Write the size of the SPI transfer in bytes to DMA0NSZH:L.
   h. Clear the address offset SFRs CMA0A0H:L.
   i. Enable the interrupt on the desired channel by setting the corresponding bit in DMA0INT.
   j. Enable DMA interrupts by setting bit 5 of EIE2.
3. Clear the interrupt bit in DMA0INT for the desired channel.
4. Enable the desired channel by setting the corresponding bit in the DMA0EN SFR to initiate the SPI transfer operation.
5. Wait on the DMA interrupt.
6. Clear the DMA enables in the DMA0EN SFR.
7. Clear the DMA interrupts in the DMA0INT SFR.
8. If desired, wait on the SPIBSY bit in SPI1CFG for the last byte transfer to complete.

32.2.2. 8-bit Timers with Auto-Reload

When T2SPLIT is set, Timer 2 operates as two 8-bit timers (TMR2H and TMR2L). Both 8-bit timers operate in auto-reload mode as shown in Figure 32.5. TMR2RLL holds the reload value for TMR2L; TMR2RLH holds the reload value for TMR2H. The TR2 bit in TMR2CN handles the run control for TMR2H. TMR2L is always running when configured for 8-bit Mode.

Each 8-bit timer may be configured to use SYSCLK, SYSCLK divided by 12, SmaRTClock divided by 8 or Comparator 0 output. The Timer 2 Clock Select bits (T2MH and T2ML in CKCON) select either SYSCLK or the clock defined by the Timer 2 External Clock Select bits (T2XCLK[1:0] in TMR2CN), as follows:

T2MH	T2XCLK[1:0]	TMR2H Clock Source
0	00	SYSCLK / 12
0	01	SmaRTClock / 8
0	10	Reserved
0	11	Comparator 0
1	X	SYSCLK

T2ML	T2XCLK[1:0]	TMR2L Clock Source
0	00	SYSCLK / 12
0	01	SmaRTClock / 8
0	10	Reserved
0	11	Comparator 0
1	X	SYSCLK

The TF2H bit is set when TMR2H overflows from 0xFF to 0x00; the TF2L bit is set when TMR2L overflows from 0xFF to 0x00. When Timer 2 interrupts are enabled (IE.5), an interrupt is generated each time TMR2H overflows. If Timer 2 interrupts are enabled and TF2LEN (TMR2CN.5) is set, an interrupt is generated each time either TMR2L or TMR2H overflows. When TF2LEN is enabled, software must check the TF2H and TF2L flags to determine the source of the Timer 2 interrupt. The TF2H and TF2L interrupt flags are not cleared by hardware and must be manually cleared by software.

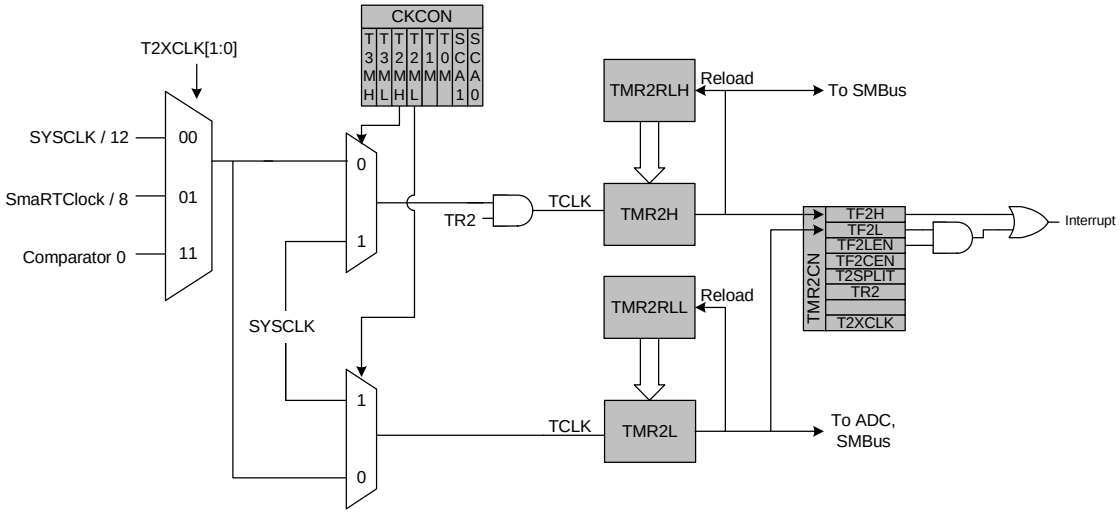

Figure 32.5. Timer 2 8-Bit Mode Block Diagram

32.2.3. Comparator 0/SmaRTClock Capture Mode

The Capture Mode in Timer 2 allows either Comparator 0 or the SmaRTClock period to be measured against the system clock or the system clock divided by 12. Comparator 0 and the SmaRTClock period can also be compared against each other. Timer 2 Capture Mode is enabled by setting TF2CEN to 1. Timer 2 should be in 16-bit auto-reload mode when using Capture Mode.

SFR Definition 32.11. TMR2L: Timer 2 Low Byte

Bit	7	6	5	4	3	2	1	0
Name	TMR2L[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Page = 0x0; SFR Address = 0xCC

Bit	Name	Function
7:0	TMR2L[7:0]	**Timer 2 Low Byte.** In 16-bit mode, the TMR2L register contains the low byte of the 16-bit Timer 2. In 8-bit mode, TMR2L contains the 8-bit low byte timer value.

SFR Definition 32.12. TMR2H Timer 2 High Byte

Bit	7	6	5	4	3	2	1	0
Name	TMR2H[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Page = 0x0; SFR Address = 0xCD

Bit	Name	Function
7:0	TMR2H[7:0]	**Timer 2 High Byte.** In 16-bit mode, the TMR2H register contains the high byte of the 16-bit Timer 2. In 8-bit mode, TMR2H contains the 8-bit high byte timer value.