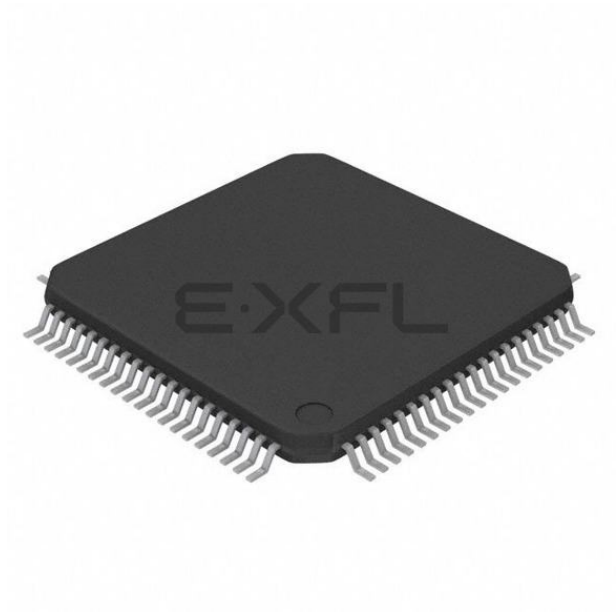

Welcome to E-XFL.COM

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	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/c8051f960-b-gq

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.

SFR Definition 7.4. CPT1MD: Comparator 1 Mode Selection

Bit	7	6	5	4	3	2	1	0
Name			CP1RIE	CP1FIE			CP1MD[1:0]	
Type	R/W	R	R/W	R/W	R	R	R/W	
Reset	1	0	0	0	0	0	1	0

SFR Page = 0x0; SFR Address = 0x9C

Bit	Name	Function
7	Reserved	Read = 1b, Must Write 1b.
6	Unused	**Unused.** Read = 0b, Write = don't care.
5	CP1RIE	**Comparator1 Rising-Edge Interrupt Enable.** 0: Comparator1 Rising-edge interrupt disabled. 1: Comparator1 Rising-edge interrupt enabled.
4	CP1FIE	**Comparator1 Falling-Edge Interrupt Enable.** 0: Comparator1 Falling-edge interrupt disabled. 1: Comparator1 Falling-edge interrupt enabled.
3:2	Unused	Read = 00b, Write = don't care.
1:0	CP1MD[1:0]	**Comparator1 Mode Select** These bits affect the response time and power consumption for Comparator1. 00: Mode 0 (Fastest Response Time, Highest Power Consumption) 01: Mode 1 10: Mode 2 11: Mode 3 (Slowest Response Time, Lowest Power Consumption)

11. Direct Memory Access (DMA0)

An on-chip direct memory access (DMA0) is included on the C8051F96x devices. The DMA0 subsystem allows autonomous variable-length data transfers between XRAM and peripheral SFR registers without CPU intervention. During DMA0 operation, the CPU is free to perform some other tasks. In order to save total system power consumption, the CPU and flash can be powered down. DMA0 improves the system performance and efficiency with high data throughput peripherals.

DMA0 contains seven independent channels, common control registers, and a DMA0 Engine (see Figure 11.1). Each channel includes a register that assigns a peripheral to the channel, a channel control register, and a set of SFRs that include XRAM address information and SFR address information used by the channel during a data transfer. The DMA0 architecture is described in detail in Section 11.1.

The DMA0 in C8051F96x devices supports four peripherals: AES0, ENC0, CRC1, and SPI1. Peripherals with DMA0 capability should be configured to work with the DMA0 through their own registers. The DMA0 provides up to seven channels, and each channel can be configured for one of nine possible data transfer functions:

- XRAM to ENC0L/M/H
- ENC0L/M/H sfrs to XRAM
- XRAM to CRC1IN sfr
- XRAM to SPI1DAT sfr
- SPI1DAT sfr to XRAM
- XRAM to AES0KIN sfr
- XRAM to AES0BIN sfr
- XRAM to AES0XIN sfr
- AES0YOUT sfr to XRAM

The DMA0 subsystem signals the MCU through a set of interrupt service routine flags. Interrupts can be generated when the DMA0 transfers half of the data length or full data length on any channel.

SFR Definition 11.6. DMA0NMD: DMA Channel Mode

Bit	7	6	5	4	3	2	1	0
Name								WRAP
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 = 0xD6

Bit	Name	Function
7:1	reserved	Read = 0, Write = 0
0	WRAP	**Wrap Enable.** Setting this bit will enable wrapping. The DMA0NSZ register sets the transfer size. Normally the DMA0AO value starts at zero in increases to the DMANSZ minus one. At this point the transfer is complete and the interrupt bit will be set. If the WRAP bit is set, the DMA0NAO will be reset to zero.
Note: This sfr is a DMA channel indirect register. Select the desired channel first using the DMA0SEL sfr.		

12.2. 32-bit CRC Algorithm

The C8051F41x CRC unit calculates the 32-bit CRC using a poly of 0x04C11DB7. The CRC-32 algorithm is "reflected", meaning that all of the input bytes and the final 32-bit output are bit-reversed in the processing engine. The following is a description of a simplified CRC algorithm that produces results identical to the hardware:

Step 1. XOR the least-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 (0x00000000 or 0xFFFFFFFF).
Step 2. Right-shift the CRC result.
Step 3. If the LSB of the CRC result is set, XOR the CRC result with the reflected polynomial (0xEDB88320).
Step 4. Repeat at Step 2 for the number of input bits (8).

For example, the 32-bit 'F41x CRC algorithm can be described by the following code:

```
unsigned long UpdateCRC (unsigned long CRC_acc, unsigned char CRC_input)
{
   unsigned char i; // loop counter
   #define POLY 0xEDB88320 // bit-reversed version of the poly 0x04C11DB7
   // Create the CRC "dividend" for polynomial arithmetic (binary arithmetic
   // with no carries)

   CRC_acc = CRC_acc ^ CRC_input;

   // "Divide" the poly into the dividend using CRC XOR subtraction
   // CRC_acc holds the "remainder" of each divide
   //
   // Only complete this division for 8 bits since input is 1 byte
   for (i = 0; i < 8; i++)
   {
      // Check if the MSB is set (if MSB is 1, then the POLY can "divide"
      // into the "dividend")
      if ((CRC_acc & 0x00000001) == 0x00000001)
      {
         // if so, shift the CRC value, and XOR "subtract" the poly
         CRC_acc = CRC_acc >> 1;
         CRC_acc ^= POLY;
      }
      else
      {
         // if not, just shift the CRC value
         CRC_acc = CRC_acc >> 1;
      }
   }
   // Return the final remainder (CRC value)
   return CRC_acc;
}
```

The following table lists several input values and the associated outputs using the 32-bit 'F41x CRC algorithm (an initial value of 0xFFFFFFFF is used):

The key and data to be encrypted should be stored as an array with the first byte to be encrypted at the lowest address. The value of the big endian bit of the DMACF0 sfr does not matter. The AES block uses only one byte transfers, so there is no particular endianness associated with a one byte transfer.

The dummy data can be zeros or any value. The encrypted data is discarded, so the value of the dummy data does not mater.

It is not strictly required to use DMA channels 0, 1, and 2. Any three DMA channels may be used. The internal state machine of the AES module will send the peripheral requests in the required order.

If the other DMA channels are going to be used concurrently with encryption, then only the bits corresponding to the encryption channels should be manipulated in DM0AEN and DMA0NT sfrs.

14.2.2. Key Inversion using SFRs

Normally, the AES block is used with the DMA. This provides the best performance and lowest power consumption. However, it is also possible to use the DMA with direct SFR access. The steps are documented in the datasheet for completeness.

Steps to generate the Decryption Key from Encryption Key using SFR.

- First configure the AES block for Key inversion:
  - Reset AES module by writing 0x00 to AES0BCFG.
  - Configure the AES Module data flow for inverse key generation by writing 0x04 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.
- Write the dummy data alternating with Key data:
  - Write the first dummy byte to AES0BIN
  - Write the first key byte to AES0KIN
  - Repeat until all dummy data bytes are written
- If using 192-bit and 256-bit key, write remaining key bytes to AES0KIN:
- Wait on AES done interrupt or poll bit 5 of AES0BCFG
- Read first byte of the decryption key from the AES0YOUT sfr

SFR Definition 16.2. SFRPAGE: SFR Page

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

SFR Page = All Pages; SFR Address = 0xA7

Bit	Name	Function
7:0	SFRPAGE[7:0]	**SFR Page Bits.** Represents the SFR Page the C8051 core uses when reading or modifying SFRs. Write: Sets the SFR Page. Read: Byte is the SFR page the C8051 core is using. When enabled in the SFR Page Control Register (SFR0CN), the C8051 core will automatically switch to the SFR Page that contains the SFRs of the corresponding peripheral/function that caused the interrupt, and return to the previous SFR page upon return from interrupt (unless SFR Stack was altered before a returning from the interrupt). SFRPAGE is the top byte of the SFR Page Stack, and push/pop events of this stack are caused by interrupts (and not by reading/writing to the SFRPAGE register)

Table 16.3. Special Function Registers (Continued)

SFRs are listed in alphabetical order. All undefined SFR locations are reserved

Register	Address	SFR Page	Description	Page
PC0INT0	0xFB	0x2	PC0 Interrupt 0	332
PC0INT1	0xFC	0x2	PC0 Interrupt 1	333
PC0MD	0xD9	0x2	PC0 Mode	321
PC0PCF	0xD7	0x2	PC0 Pull-up Configuration	322
PC0STAT	0xC1	0x2	PC0 Status	324
PC0TH	0xE4	0x2	PC0 Threshold	323
PCA0CN	0xD8	All Pages	PCA0 Control	480
PCA0CPH0	0xFC	0x0	PCA0 Capture 0 High	485
PCA0CPH1	0xEA	0x0	PCA0 Capture 1 High	485
PCA0CPH2	0xEC	0x0	PCA0 Capture 2 High	485
PCA0CPH3	0xEE	0x0	PCA0 Capture 3 High	485
PCA0CPH4	0xFE	0x0	PCA0 Capture 4 High	485
PCA0CPH5	0xD3	0x0	PCA0 Capture 5 High	485
PCA0CPL0	0xFB	0x0	PCA0 Capture 0 Low	485
PCA0CPL1	0xE9	0x0	PCA0 Capture 1 Low	485
PCA0CPL2	0xEB	0x0	PCA0 Capture 2 Low	485
PCA0CPL3	0xED	0x0	PCA0 Capture 3 Low	485
PCA0CPL4	0xFD	0x0	PCA0 Capture 4 Low	485
PCA0CPL5	0xD2	0x0	PCA0 Capture 5 Low	485
PCA0CPM0	0xDA	0x0	PCA0 Module 0 Mode Register	483
PCA0CPM1	0xDB	0x0	PCA0 Module 1 Mode Register	483
PCA0CPM2	0xDC	0x0	PCA0 Module 2 Mode Register	483
PCA0CPM3	0xDD	0x0	PCA0 Module 3 Mode Register	483
PCA0CPM4	0xDE	0x0	PCA0 Module 4 Mode Register	483
PCA0CPM5	0xCE	0x0	PCA0 Module 5 Mode Register	483
PCA0H		0x0	PCA0 Counter High	484
PCA0L	0xF9	0x0	PCA0 Counter Low	484
PCA0MD	0xD9	0x0	PCA0 Mode	481
PCA0PWM	0xDF	0x0	PCA0 PWM Configuration	482
PCLKACT	0xF5	0xF	Peripheral Clock Enable Active Mode	260
PCLKEN	0xFE	0xF	Peripheral Clock Enables (LP Idle)	261
PCON	0x87	All Pages	Power Control	268
PMU0CF	0xB5	0x0	PMU0 Configuration 0	265
PMU0FL	0xB6	0x0	PMU0 flag	266

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.

Important Notes:

- The Power-on Reset (POR) delay is not incurred after a supply monitor reset. See Section “4. Electrical Characteristics” on page 56 for complete electrical characteristics of the active mode supply monitors.
- Software should take care not to inadvertently disable the supply monitor as a reset source when writing to RSTSRC to enable other reset sources or to trigger a software reset. All writes to RSTSRC should explicitly set PORSF to 1 to keep the supply monitor enabled as a reset source.
- The supply monitor must be enabled before selecting it as a reset source**.** Selecting the supply monitor as a reset source before it has stabilized may generate a system reset. In systems where this reset would be undesirable, a delay should be introduced between enabling the supply monitor and selecting it as a reset source. See Section “4. Electrical Characteristics” on page 56 for minimum supply monitor turn-on time. **No delay should be introduced in systems where software contains routines that erase or write Flash memory.** The procedure for enabling the V_{DD} supply monitor and selecting it as a reset source is shown below:
  1. Enable the Supply Monitor (VDMEN bit in VDM0CN = 1).
  2. Wait for the Supply Monitor to stabilize (optional).
  3. Select the Supply Monitor as a reset source (PORSF bit in RSTSRC = 1).

SFR Definition 25.20. PC0INT0: PC0 Interrupt 0

Bit	7	6	5	4	3	2	1	0
Name	CMP1F	CMP1EN	CMP0F	CMP0EN	OVRF	OVREN	DIRCHGF	DIRCHGEN
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 Address = 0xFB; SFR Page = 0x2

Bit	Name	Function
7	CMP1F	**Comparator 1 Flag** 0: Counter 1 did not match comparator 1 value. 1: Counter 1 matched comparator 1 value.
6	CMP1EN	**Comparator 1 Interrupt/Wake-up Source Enable** 0:CMP1F not enabled as interrupt or wake-up source. 1:CMP1F enabled as interrupt or wake-up source.
5	CMP0F	**Comparator 0 Flag** 0: Counter 0 did not match comparator 0 value. 1: Counter 0 matched comparator 0 value.
4	CMP0EN	**Comparator 0 Interrupt/Wake-up Source Enable** 0:CMP0F not enabled as interrupt or wake-up source. 1:CMP0F enabled as interrupt or wake-up source.
3	OVRF	**Counter Overflow Flag** 1:Neither of the counters has overflowed. 1:One of the counters has overflowed.
2	OVREN	**Counter Overflow Interrupt/Wake-up Source Enable** 0:OVRF not enabled as interrupt or wake-up source. 1:OVRF enabled as interrupt or wake-up source.
1	DIRCHGF	**Direction Change Flag** Direction changed for quadrature mode only. 0:No change in direction detected. 1:Direction Change detected.
0	DIRCHGEN	**Direction Change Interrupt/Wake-up Source Enable** 0:DIRCHGF not enabled as interrupt or wake-up source. 1:DIRCHGF enabled as interrupt or wake-up source.

26.3.4. Contrast Control Mode 4 (Auto-Bypass Mode)

In Contrast Control Mode 4, behavior is identical to Constant Contrast Mode as long as VBAT is greater than the VBAT monitor threshold voltage. When VBAT drops below the programmed threshold, the device automatically enters bypass mode powering VLCD directly from VBAT. The charge pump is always disabled in this mode. Auto-Bypass Mode is selected using the following procedure:

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

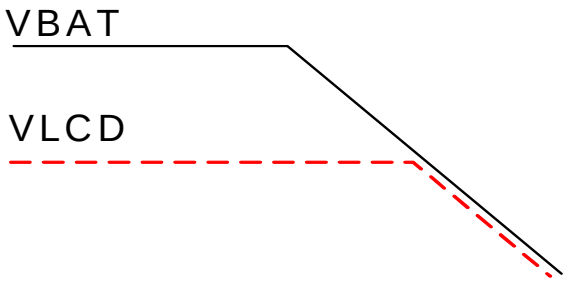

Figure 26.6. Contrast Control Mode 4

SFR Definition 27.18. P2: Port2

Bit	7	6	5	4	3	2	1	0
Name	P2[7:0]							
Type	R/W							
Reset	1	1	1	1	1	1	1	1

SFR Page = All Pages; SFR Address = 0xA0; Bit-Addressable

Bit	Name	Description	Write	Read
7:0	P2[7:0]	**Port 2 Data.** Sets the Port latch logic value or reads the Port pin logic state in Port cells configured for digital I/O.	0: Set output latch to logic LOW. 1: Set output latch to logic HIGH.	0: P2.n Port pin is logic LOW. 1: P2.n Port pin is logic HIGH.

SFR Definition 27.19. P2SKIP: Port2 Skip

Bit	7	6	5	4	3	2	1	0
Name	P2SKIP[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Page = 0x0; SFR Address = 0xD6

Bit	Name	Function
7:0	P2SKIP[7:0]	**Port 1 Crossbar Skip Enable Bits.** These bits select Port 2 pins to be skipped by the Crossbar Decoder. Port pins used for analog, special functions or GPIO should be skipped by the Crossbar. 0: Corresponding P2.n pin is not skipped by the Crossbar. 1: Corresponding P2.n pin is skipped by the Crossbar.

Data transmission begins when an instruction writes a data byte to the SBUF0 register. The TI0 Transmit Interrupt Flag (SCON0.1) is set at the end of the transmission (the beginning of the stop-bit time). Data reception can begin any time after the REN0 Receive Enable bit (SCON0.4) is set to 1. After the stop bit is received, the data byte will be loaded into the SBUF0 receive register if the following conditions are met: (1) RI0 must be logic 0, and (2) if MCE0 is logic 1, the 9th bit must be logic 1 (when MCE0 is logic 0, the state of the ninth data bit is unimportant). If these conditions are met, the eight bits of data are stored in SBUF0, the ninth bit is stored in RB80, and the RI0 flag is set to 1. If the above conditions are not met, SBUF0 and RB80 will not be loaded and the RI0 flag will not be set to 1. A UART0 interrupt will occur if enabled when either TI0 or RI0 is set to 1.

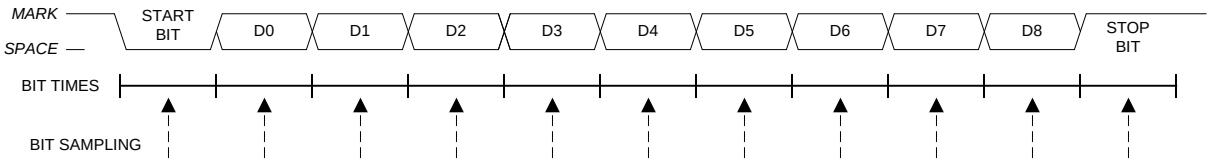

Figure 29.5. 9-Bit UART Timing Diagram

29.3. Multiprocessor Communications

9-Bit UART mode supports multiprocessor communication between a master processor and one or more slave processors by special use of the ninth data bit. When a master processor wants to transmit to one or more slaves, it first sends an address byte to select the target(s). An address byte differs from a data byte in that its ninth bit is logic 1; in a data byte, the ninth bit is always set to logic 0.

Setting the MCE0 bit (SCON0.5) of a slave processor configures its UART such that when a stop bit is received, the UART will generate an interrupt only if the ninth bit is logic 1 (RB80 = 1) signifying an address byte has been received. In the UART interrupt handler, software will compare the received address with the slave's own assigned 8-bit address. If the addresses match, the slave will clear its MCE0 bit to enable interrupts on the reception of the following data byte(s). Slaves that weren't addressed leave their MCE0 bits set and do not generate interrupts on the reception of the following data bytes, thereby ignoring the data. Once the entire message is received, the addressed slave resets its MCE0 bit to ignore all transmissions until it receives the next address byte.

Multiple addresses can be assigned to a single slave and/or a single address can be assigned to multiple slaves, thereby enabling "broadcast" transmissions to more than one slave simultaneously. The master processor can be configured to receive all transmissions or a protocol can be implemented such that the master/slave role is temporarily reversed to enable half-duplex transmission between the original master and slave(s).

SFR Definition 31.2. SPI1CN: SPI1 Control

Bit	7	6	5	4	3	2	1	0
Name	SPIF	WCOL	MODF	RXOVRN	NSSMD[1:0]		TXBMT	SPIEN
Type	R/W	R/W	R/W	R/W	R/W		R	R/W
Reset	0	0	0	0	0	1	1	0

SFR Page = 0x0; SFR Address = 0xB0; Bit-Addressable

Bit	Name	Function
7	SPIF	**SPI1 Interrupt Flag.** This bit is set to logic 1 by hardware at the end of a data transfer. If SPI interrupts are enabled, an interrupt will be generated. This bit is not automatically cleared by hardware, and must be cleared by software.
6	WCOL	**Write Collision Flag.** This bit is set to logic 1 if a write to SPI1DAT is attempted when TXBMT is 0. When this occurs, the write to SPI1DAT will be ignored, and the transmit buffer will not be written. If SPI interrupts are enabled, an interrupt will be generated. This bit is not automatically cleared by hardware, and must be cleared by software.
5	MODF	**Mode Fault Flag.** This bit is set to logic 1 by hardware when a master mode collision is detected (NSS is low, MSTEN = 1, and NSSMD[1:0] = 01). If SPI interrupts are enabled, an interrupt will be generated. This bit is not automatically cleared by hardware, and must be cleared by software.
4	RXOVRN	**Receive Overrun Flag (valid in slave mode only).** This bit is set to logic 1 by hardware when the receive buffer still holds unread data from a previous transfer and the last bit of the current transfer is shifted into the SPI1 shift register. If SPI interrupts are enabled, an interrupt will be generated. This bit is not automatically cleared by hardware, and must be cleared by software.
3:2	NSSMD[1:0]	**Slave Select Mode.** Selects between the following NSS operation modes: (See Section 31.2 and Section 31.3). 00: 3-Wire Slave or 3-Wire Master Mode. NSS signal is not routed to a port pin. 01: 4-Wire Slave or Multi-Master Mode (Default). NSS is an input to the device. 1x: 4-Wire Single-Master Mode. NSS signal is mapped as an output from the device and will assume the value of NSSMD0.
1	TXBMT	**Transmit Buffer Empty.** This bit will be set to logic 0 when new data has been written to the transmit buffer. When data in the transmit buffer is transferred to the SPI shift register, this bit will be set to logic 1, indicating that it is safe to write a new byte to the transmit buffer.
0	SPIEN	**SPI1 Enable.** 0: SPI disabled. 1: SPI enabled.

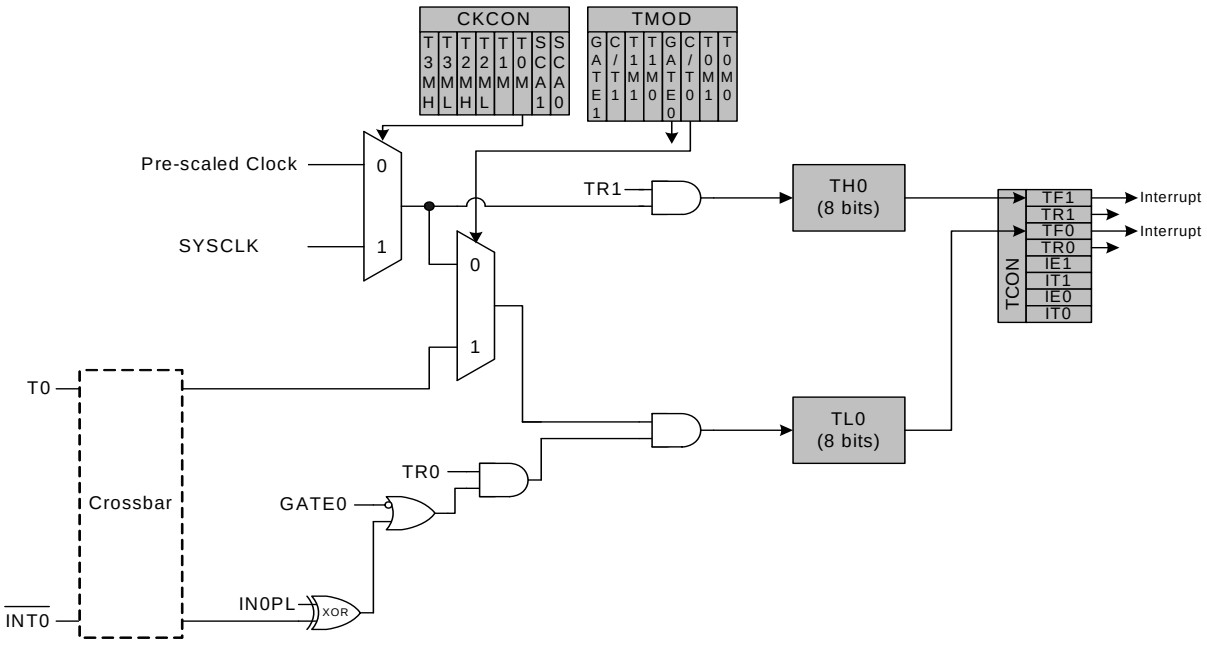

Figure 32.3. T0 Mode 3 Block Diagram