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	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	16
Program Memory Size	2KB (2K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2.2V ~ 3.6V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	24-SSOP (0.154", 3.90mm Width)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f855-c-gur

2.1. Power

2.1.1. LDO

The C8051F85x/86x devices include an internal regulator to regulate the supply voltage down the core operating voltage of 1.8 V. This LDO consumes little power, but can be shut down in the power-saving Stop mode.

2.1.2. Voltage Supply Monitor (VMON0)

The C8051F85x/86x devices include a voltage supply monitor which allows devices to function in known, safe operating condition without the need for external hardware.

The supply monitor module includes the following features:

- Holds the device in reset if the main VDD supply drops below the VDD Reset threshold.

2.1.3. Device Power Modes

The C8051F85x/86x devices feature three low power modes in addition to normal operating mode, allowing the designer to save power when the core is not in use. All power modes are detailed in Table 2.1.

Table 2.1. C8051F85x/86x Power Modes

Mode	Description	Mode Entrance	Mode Exit
Normal	Core and peripherals operating at full speed		
Idle	■ Core halted ■ Peripherals operate at full speed	Set IDLE bit in PCON	Any enabled interrupt or reset source
Stop	■ All clocks stopped ■ Core LDO and (optionally) comparators still running ■ Pins retain state	Clear STOPCF in REG0MD and Set STOP bit in PCON	Device reset
Shutdown	■ All clocks stopped ■ Core LDO and all analog circuits shut down ■ Pins retain state	Set STOPCF in REG0MD and Set STOP bit in PCON	Device reset

In addition, the user may choose to lower the clock speed in Normal and Idle modes to save power when the CPU requirements allow for lower speed.

Table 6.2. QFN-20 Landing Diagram Dimensions

Symbol	Millimeters		Symbol	Millimeters	
	Min	Max		Min	Max
D	2.71 REF		GE	2.10	—
D2	1.60	1.80	W	—	0.34
e	0.50 BSC		X	—	0.28
E	2.71 REF		Y	0.61 REF	
E2	1.60	1.80	ZE	—	3.31
f	2.53 BSC		ZD	—	3.31
GD	2.10	—			

Notes: General

1. All dimensions shown are in millimeters (mm) unless otherwise noted.
2. Dimensioning and Tolerancing is per the ANSI Y14.5M-1994 specification.
3. This Land Pattern Design is based on IPC-SM-782 guidelines.
4. All dimensions shown are at Maximum Material Condition (MMC). Least Material Condition (LMC) is calculated based on a Fabrication Allowance of 0.05 mm.

Notes: Solder Mask Design

1. All metal pads are to be non-solder mask defined (NSMD). Clearance between the solder mask and the metal pad is to be 60 µm minimum, all the way around the pad.

Notes: Stencil Design

1. A stainless steel, laser-cut and electro-polished stencil with trapezoidal walls should be used to assure good solder paste release.
2. The stencil thickness should be 0.125 mm (5 mils).
3. The ratio of stencil aperture to land pad size should be 1:1 for the perimeter pads.
4. A 1.45 x 1.45 mm square aperture should be used for the center pad. This provides approximately 70% solder paste coverage on the pad, which is optimum to assure correct component stand-off.

Notes: Card Assembly

1. A No-Clean, Type-3 solder paste is recommended.
2. The recommended card reflow profile is per the JEDEC/IPC J-STD-020 specification for Small Body Components.

Table 9.2. Special Function Registers (Continued)

Register	Address	Register Description	Page
SPI0DAT	0xA3	SPI0 Data	232
TCON	0x88	Timer 0/1 Control	271
TH0	0x8C	Timer 0 High Byte	275
TH1	0x8D	Timer 1 High Byte	276
TL0	0x8A	Timer 0 Low Byte	273
TL1	0x8B	Timer 1 Low Byte	274
TMOD	0x89	Timer 0/1 Mode	272
TMR2CN	0xC8	Timer 2 Control	277
TMR2H	0xCD	Timer 2 High Byte	282
TMR2L	0xCC	Timer 2 Low Byte	281
TMR2RLH	0xCB	Timer 2 Reload High Byte	280
TMR2RLL	0xCA	Timer 2 Reload Low Byte	279
TMR3CN	0x91	Timer 3 Control	283
TMR3H	0x95	Timer 3 High Byte	288
TMR3L	0x94	Timer 3 Low Byte	287
TMR3RLH	0x93	Timer 3 Reload High Byte	286
TMR3RLL	0x92	Timer 3 Reload Low Byte	285
VDM0CN	0xFF	Supply Monitor Control	216
WDTCN	0x97	Watchdog Timer Control	300
XBR0	0xE1	Port I/O Crossbar 0	193
XBR1	0xE2	Port I/O Crossbar 1	194
XBR2	0xE3	Port I/O Crossbar 2	195

11.1. Device Identification Registers

Register 11.1. DEVICEID: Device Identification

Bit	7	6	5	4	3	2	1	0
Name	DEVICEID							
Type	R							
Reset	0	0	1	1	0	0	0	0
SFR Address: 0xB5								

Table 11.2. DEVICEID Register Bit Descriptions

Bit	Name	Function
7:0	DEVICEID	**Device ID.** This read-only register returns the 8-bit device ID: 0x30 (C8051F85x/86x).

Register 11.2. DERIVID: Derivative Identification

Bit	7	6	5	4	3	2	1	0
Name	DERIVID							
Type	R							
Reset	X	X	X	X	X	X	X	X
SFR Address: 0xAD								

Table 11.3. DERIVID Register Bit Descriptions

Bit	Name	Function
7:0	DERIVID	**Derivative ID.** This read-only register returns the 8-bit derivative ID, which can be used by firmware to identify which device in the product family the code is executing on. The '{R}' tag in the part numbers below indicates the device revision letter in the ordering code. 0xD0: C8051F850-{R}-GU 0xD1: C8051F851-{R}-GU 0xD2: C8051F852-{R}-GU 0xD3: C8051F853-{R}-GU 0xD4: C8051F854-{R}-GU 0xD5: C8051F855-{R}-GU 0xE0: C8051F860-{R}-GS 0xE1: C8051F861-{R}-GS 0xE2: C8051F862-{R}-GS 0xE3: C8051F863-{R}-GS 0xE4: C8051F864-{R}-GS 0xE5: C8051F865-{R}-GS 0xF0: C8051F850-{R}-GM 0xF1: C8051F851-{R}-GM 0xF2: C8051F852-{R}-GM 0xF3: C8051F853-{R}-GM 0xF4: C8051F854-{R}-GM 0xF5: C8051F855-{R}-GM

12. Interrupts

The C8051F85x/86x includes an extended interrupt system supporting multiple interrupt sources with two priority levels. The allocation of interrupt sources between on-chip peripherals and external input pins varies according to the specific version of the device. Each interrupt source has one or more associated interrupt-pending flag(s) located in an SFR. When a peripheral or external source meets a valid interrupt condition, the associated interrupt-pending flag is set to logic 1.

If interrupts are enabled for the source, an interrupt request is generated when the interrupt-pending flag is set. As soon as execution of the current instruction is complete, the CPU generates an LCALL to a predetermined address to begin execution of an interrupt service routine (ISR). Each ISR must end with an RETI instruction, which returns program execution to the next instruction that would have been executed if the interrupt request had not occurred. If interrupts are not enabled, the interrupt-pending flag is ignored by the hardware and program execution continues as normal. The interrupt-pending flag is set to logic 1 regardless of the interrupt's enable/disable state.

Each interrupt source can be individually enabled or disabled through the use of an associated interrupt enable bit in an SFR (IE and EIE1). However, interrupts must first be globally enabled by setting the EA bit in the IE register to logic 1 before the individual interrupt enables are recognized. Setting the EA bit to logic 0 disables all interrupt sources regardless of the individual interrupt-enable settings.

Some interrupt-pending flags are automatically cleared by the hardware when the CPU vectors to the ISR. However, most are not cleared by the hardware and must be cleared by software before returning from the ISR. If an interrupt-pending flag remains set after the CPU completes the return-from-interrupt (RETI) instruction, a new interrupt request will be generated immediately and the CPU will re-enter the ISR after the completion of the next instruction.

12.1. MCU Interrupt Sources and Vectors

The C8051F85x/86x MCUs support interrupt sources for each peripheral on the device. Software can simulate an interrupt by setting any interrupt-pending flag to logic 1. If interrupts are enabled for the flag, an interrupt request will be generated and the CPU will vector to the ISR address associated with the interrupt-pending flag. MCU interrupt sources, associated vector addresses, priority order and control bits are summarized in Table 12.1. Refer to the datasheet section associated with a particular on-chip peripheral for information regarding valid interrupt conditions for the peripheral and the behavior of its interrupt-pending flag(s).

12.1.1. Interrupt Priorities

Each interrupt source can be individually programmed to one of two priority levels: low or high. A low priority interrupt service routine can be preempted by a high priority interrupt. A high priority interrupt cannot be preempted. Each interrupt has an associated interrupt priority bit in an SFR (IP or EIP1) used to configure its priority level. Low priority is the default. If two interrupts are recognized simultaneously, the interrupt with the higher priority is serviced first. If both interrupts have the same priority level, a fixed priority order is used to arbitrate, given in Table 12.1.

12.1.2. Interrupt Latency

Interrupt response time depends on the state of the CPU when the interrupt occurs. Pending interrupts are sampled and priority decoded each system clock cycle. Therefore, the fastest possible response time is 5 system clock cycles: 1 clock cycle to detect the interrupt and 4 clock cycles to complete the LCALL to the ISR. If an interrupt is pending when a RETI is executed, a single instruction is executed before an LCALL is made to service the pending interrupt. Therefore, the maximum response time for an interrupt (when no other interrupt is currently being serviced or the new interrupt is of greater priority) occurs when the CPU is performing an RETI instruction followed by a DIV as the next instruction. In this case, the response time is 18 system clock cycles: 1 clock cycle to detect the interrupt, 5 clock cycles to execute the RETI, 8 clock

14.1. ADC0 Analog Multiplexer

ADC0 on C8051F85x/86x has an analog multiplexer capable of selecting any pin on ports P0 and P1 (up to 16 total), the on-chip temperature sensor, the internal regulated supply, the VDD supply, or GND. ADC0 input channels are selected using the ADC0MX register.

Table 14.1. ADC0 Input Multiplexer Channels

ADC0MX setting	Signal Name	QSOP24 Pin Name	QFN20 Pin Name	SOIC16 Pin Name
00000	ADC0.0	P0.0	P0.0	P0.0
00001	ADC0.1	P0.1	P0.1	P0.1
00010	ADC0.2	P0.2	P0.2	P0.2
00011	ADC0.3	P0.3	P0.3	P0.3
00100	ADC0.4	P0.4	P0.4	P0.4
00101	ADC0.5	P0.5	P0.5	P0.5
00110	ADC0.6	P0.6	P0.6	P0.6
00111	ADC0.7	P0.7	P0.7	P0.7
01000	ADC0.8	P1.0	P1.0	P1.0
01001	ADC0.9	P1.1	P1.1	P1.1
01010	ADC0.10	P1.2	P1.2	P1.2
01011	ADC0.11	P1.3	P1.3	P1.3
01100	ADC0.12	P1.4	P1.4	Reserved
01101	ADC0.13	P1.5	P1.5	Reserved
01110	ADC0.14	P1.6	P1.6	Reserved
01111	ADC0.15	P1.7	Reserved	Reserved
10000	Temp Sensor	Internal Temperature Sensor		
10001	LDO	Internal 1.8 V LDO Output		
10010	VDD	VDD Supply Pin		
10011	GND	GND Supply Pin		
10100-11111	None	No connection		

Register 14.10. ADC0GTL: ADC0 Greater-Than Low Byte

Bit	7	6	5	4	3	2	1	0
Name	ADC0GTL							
Type	RW							
Reset	1	1	1	1	1	1	1	1
SFR Address: 0xC3								

Table 14.13. ADC0GTL Register Bit Descriptions

Bit	Name	Function
7:0	ADC0GTL	**Greater-Than Low Byte.** Least Significant Byte of the 16-bit Greater-Than window compare register.
Note: In 8-bit mode, this register should be set to 0x00.		

Register 14.12. ADC0LTL: ADC0 Less-Than Low Byte

Bit	7	6	5	4	3	2	1	0
Name	ADC0LTL							
Type	RW							
Reset	0	0	0	0	0	0	0	0
SFR Address: 0xC5								

Table 14.15. ADC0LTL Register Bit Descriptions

Bit	Name	Function
7:0	ADC0LTL	**Less-Than Low Byte.** Least Significant Byte of the 16-bit Less-Than window compare register.
Note: In 8-bit mode, this register should be set to 0x00.		

Register 14.14. REF0CN: Voltage Reference Control

Bit	7	6	5	4	3	2	1	0
Name	IREFLVL	Reserved	GNDSL	REFSL		TEMPE	Reserved	
Type	RW	R	RW	RW		RW	R	
Reset	0	0	0	1	1	0	0	0
SFR Address: 0xD1								

Table 14.17. REF0CN Register Bit Descriptions

Bit	Name	Function
7	IREFLVL	**Internal Voltage Reference Level.** Sets the voltage level for the internal reference source. 0: The internal reference operates at 1.65 V nominal. 1: The internal reference operates at 2.4 V nominal.
6	Reserved	Must write reset value.
5	GNDSL	**Analog Ground Reference.** Selects the ADC0 ground reference. 0: The ADC0 ground reference is the GND pin. 1: The ADC0 ground reference is the AGND pin.
4:3	REFSL	**Voltage Reference Select.** Selects the ADC0 voltage reference. 00: The ADC0 voltage reference is the 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 voltage reference.
2	TEMPE	**Temperature Sensor Enable.** Enables/Disables the internal temperature sensor. 0: Temperature Sensor Disabled. 1: Temperature Sensor Enabled.
1:0	Reserved	Must write reset value.

Table 17.1. CMP0 Positive Input Multiplexer Channels

CMXP Setting in Register CPT0MX	Signal Name	QSOP24 Pin Name	QFN20 Pin Name	SOIC16 Pin Name
0000	CP0P.0	P0.0	P0.0	P0.0
0001	CP0P.1	P0.1	P0.1	P0.1
0010	CP0P.2	P0.2	P0.2	P0.2
0011	CP0P.3	P0.3	P0.3	P0.3
0100	CP0P.4	P0.4	P0.4	P0.4
0101	CP0P.5	P0.5	P0.5	P0.5
0110	CP0P.6	P0.6	P0.6	Reserved
0111	CP0P.7	P0.7	P0.7	Reserved
1000	LDO	Internal 1.8 V LDO Output		
1001-1111	None	No connection		

Table 17.2. CMP0 Negative Input Multiplexer Channels

CMXN Setting in Register CPT0MX	Signal Name	QSOP24 Pin Name	QFN20 Pin Name	SOIC16 Pin Name
0000	CP0N.0	P0.0	P0.0	P0.0
0001	CP0N.1	P0.1	P0.1	P0.1
0010	CP0N.2	P0.2	P0.2	P0.2
0011	CP0N.3	P0.3	P0.3	P0.3
0100	CP0N.4	P0.4	P0.4	P0.4
0101	CP0N.5	P0.5	P0.5	P0.5
0110	CP0N.6	P0.6	P0.6	Reserved
0111	CP0N.7	P0.7	P0.7	Reserved
1000	GND	GND		
1001-1111	None	No connection		

20.3.3. Software Timer (Compare) Mode

In Software Timer mode, the PCA counter/timer value is compared to the module's 16-bit capture/compare register (PCA0CPHn and PCA0CPLn). When a match occurs, the Capture/Compare Flag (CCFn) in PCA0CN is set to logic 1. An interrupt request is generated if the CCFn interrupt for that module is enabled. The CCFn bit is not automatically cleared by hardware when the CPU vectors to the interrupt service routine, and must be cleared by software. Setting the ECOMn and MATn bits in the PCA0CPMn register enables Software Timer mode.

Important Note About Capture/Compare Registers: When writing a 16-bit value to the PCA0 Capture/Compare registers, the low byte should always be written first. Writing to PCA0CPLn clears the ECOMn bit to 0; writing to PCA0CPHn sets ECOMn to 1.

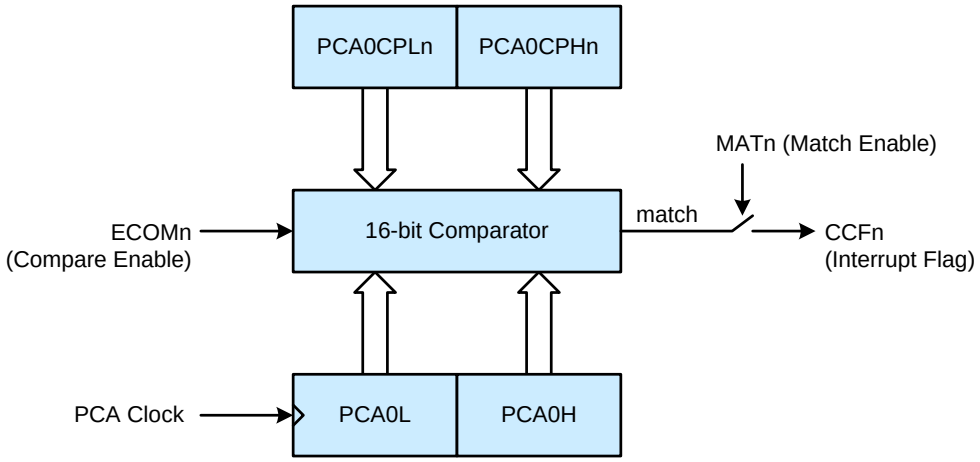

Figure 20.3. PCA Software Timer Mode Diagram

Register 20.17. PCA0CPH2: PCA Capture Module High Byte

Bit	7	6	5	4	3	2	1	0
Name	PCA0CPH2							
Type	RW							
Reset	0	0	0	0	0	0	0	0
SFR Address: 0xEC								

Table 20.19. PCA0CPH2 Register Bit Descriptions

Bit	Name	Function
7:0	PCA0CPH2	**PCA Capture Module High Byte.** The PCA0CPH2 register holds the high byte (MSB) of the 16-bit capture module.This register address also allows access to the high byte of the corresponding PCA channels auto-reload value for 9 to 11-bit PWM mode. The ARSEL bit in register PCA0PWM controls which register is accessed.
Note: A write to this register will set the modules ECOM bit to a 1.		

Note: The SPI can be operated in either 3-wire or 4-wire modes, pending the state of the NSSMD1–NSSMD0 bits in register SPI0CN. According to the SPI mode, the NSS signal may or may not be routed to a port pin. The order in which SMBus pins are assigned is defined by the SWAP bit in the SMB0TC register.

22.11. Supply Monitor Control Registers

Register 22.2. VDM0CN: Supply Monitor Control

Bit	7	6	5	4	3	2	1	0
Name	VDMEN	VDDSTAT	Reserved					
Type	RW	R	R					
Reset	X	X	X	X	X	X	X	X
SFR Address: 0xFF								

Table 22.2. VDM0CN Register Bit Descriptions

Bit	Name	Function
7	VDMEN	**Supply Monitor Enable.** This bit turns the supply monitor circuit on/off. The supply monitor cannot generate system resets until it is also selected as a reset source in register RSTSRC. 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. 0: Supply Monitor Disabled. 1: Supply Monitor Enabled.
6	VDDSTAT	**Supply Status.** This bit indicates the current power supply status (supply monitor output). 0: V_{DD} is at or below the supply monitor threshold. 1: V_{DD} is above the supply monitor threshold.
5:0	Reserved	Must write reset value.

24. System Management Bus / I^{2}C (SMBus0)

The SMBus I/O interface is a two-wire, bi-directional serial bus. The SMBus is compliant with the System Management Bus Specification, version 1.1, and compatible with the I^{2}C serial bus.

Reads and writes to the SMBus by the system controller are byte oriented with the SMBus interface autonomously controlling the serial transfer of the data. Data can be transferred at up to 1/20th of the system clock as a master or slave (this can be faster than allowed by the SMBus specification, depending on the system clock used). A method of extending the clock-low duration is available to accommodate devices with different speed capabilities on the same bus.

The SMBus may operate as a master and/or slave, and may function on a bus with multiple masters. The SMBus provides control of SDA (serial data), SCL (serial clock) generation and synchronization, arbitration logic, and START/STOP control and generation. The SMBus peripherals can be fully driven by software (i.e., software accepts/rejects slave addresses, and generates ACKs), or hardware slave address recognition and automatic ACK generation can be enabled to minimize software overhead. A block diagram of the SMBus0 peripheral is shown in Figure 24.1.

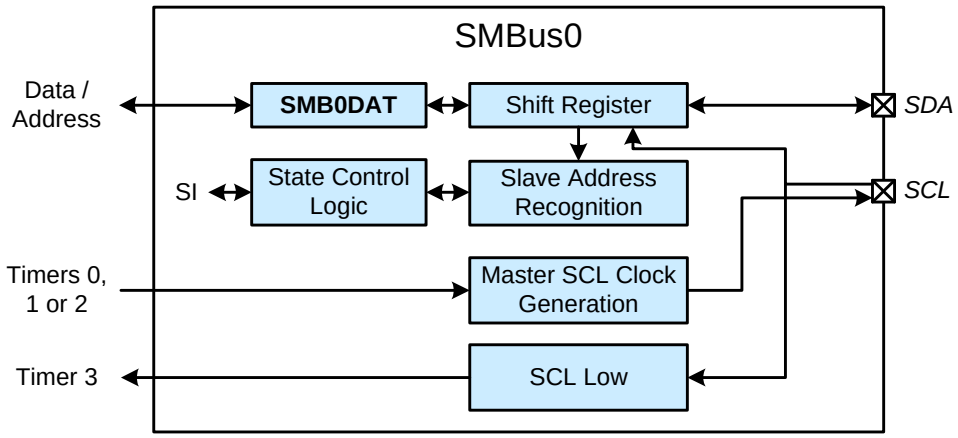

Figure 24.1. SMBus0 Block Diagram

Register 24.6. SMB0DAT: SMBus0 Data

Bit	7	6	5	4	3	2	1	0
Name	SMB0DAT							
Type	RW							
Reset	0	0	0	0	0	0	0	0
SFR Address: 0xC2								

Table 24.12. SMB0DAT Register Bit Descriptions

Bit	Name	Function
7:0	SMB0DAT	**SMBus0 Data.** The SMB0DAT register contains a byte of data to be transmitted on the SMBus0 serial interface or a byte that has just been received on the SMBus0 serial interface. The CPU can safely read from or write to this register whenever the SI serial interrupt flag 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.

25.1.1. Mode 0: 13-bit Counter/Timer

Timer 0 and Timer 1 operate as 13-bit counter/timers in Mode 0. The following describes the configuration and operation of Timer 0. However, both timers operate identically, and Timer 1 is configured in the same manner as described for Timer 0.

The TH0 register holds the eight MSBs of the 13-bit counter/timer. TL0 holds the five LSBs in bit positions TL0.4–TL0.0. The three upper bits of TL0 (TL0.7–TL0.5) are indeterminate and should be masked out or ignored when reading. As the 13-bit timer register increments and overflows from 0x1FFF (all ones) to 0x0000, the timer overflow flag TF0 in TCON is set and an interrupt will occur if Timer 0 interrupts are enabled.

The CT0 bit in the TMOD register selects the counter/timer's clock source. When CT0 is set to logic 1, high-to-low transitions at the selected Timer 0 input pin (T0) increment the timer register. Clearing CT selects the clock defined by the T0M bit in register CKCON. When T0M is set, Timer 0 is clocked by the system clock. When T0M is cleared, Timer 0 is clocked by the source selected by the Clock Scale bits in CKCON.

Setting the TR0 bit enables the timer when either GATE0 in the TMOD register is logic 0 or the input signal $\overline{INT0}$ is active as defined by bit IN0PL in register IT01CF. Setting GATE0 to 1 allows the timer to be controlled by the external input signal $\overline{INT0}$, facilitating pulse width measurements.

TR0	GATE0	$\overline{INT0}$	Counter/Timer
0	X	X	Disabled
1	0	X	Enabled
1	1	0	Disabled
1	1	1	Enabled
Note: X = Don't Care			

Setting TR0 does not force the timer to reset. The timer registers should be loaded with the desired initial value before the timer is enabled.

TL1 and TH1 form the 13-bit register for Timer 1 in the same manner as described above for TL0 and TH0. Timer 1 is configured and controlled using the relevant TCON and TMOD bits just as with Timer 0. The input signal $\overline{INT1}$ is used with Timer 1; the /INT1 polarity is defined by bit IN1PL in register IT01CF.

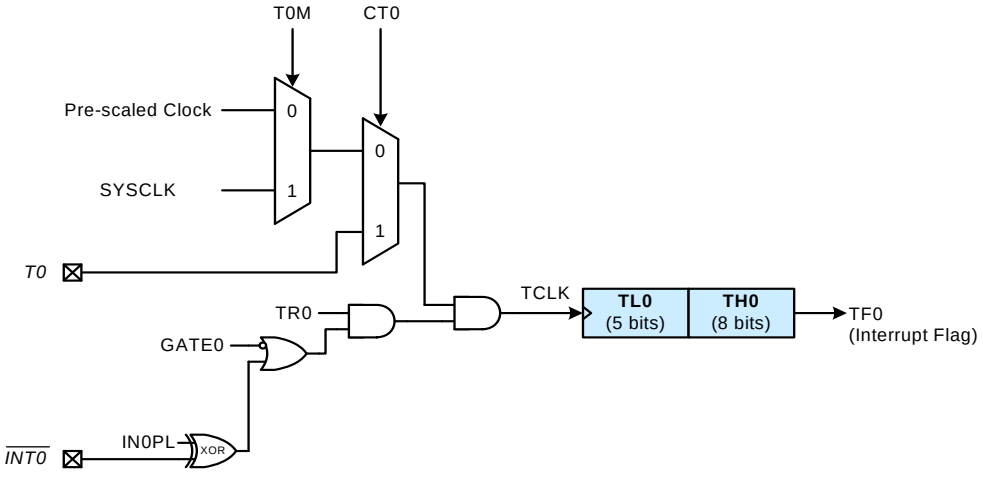

Figure 25.1. T0 Mode 0 Block Diagram

26.2. Operational Modes

UART0 provides standard asynchronous, full duplex communication. The UART mode (8-bit or 9-bit) is selected by the S0MODE bit in register SCON.

26.2.1. 8-Bit UART

8-Bit UART mode uses a total of 10 bits per data byte: one start bit, eight data bits (LSB first), and one stop bit. Data are transmitted LSB first from the TX pin and received at the RX pin. On receive, the eight data bits are stored in SBUF0 and the stop bit goes into RB8 in the SCON register.

Data transmission begins when software writes a data byte to the SBUF0 register. The TI Transmit Interrupt Flag is set at the end of the transmission (the beginning of the stop-bit time). Data reception can begin any time after the REN Receive Enable bit is set to logic 1. After the stop bit is received, the data byte will be loaded into the SBUF0 receive register if the following conditions are met: RI must be logic 0, and if MCE is logic 1, the stop bit must be logic 1. In the event of a receive data overrun, the first received 8 bits are latched into the SBUF0 receive register and the following overrun data bits are lost.

If these conditions are met, the eight bits of data is stored in SBUF0, the stop bit is stored in RB8 and the RI flag is set. If these conditions are not met, SBUF0 and RB8 will not be loaded and the RI flag will not be set. An interrupt will occur if enabled when either TI or RI is set.

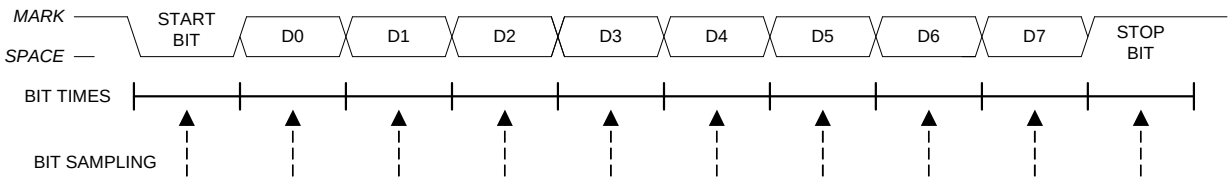

Figure 26.3. 8-Bit UART Timing Diagram

28.2. Temperature Sensor Offset and Slope

The temperature sensor slope and offset characteristics of Revision B devices are different than the slope and offset characteristics of Revision C devices. The differences are:

Table 28.1. Temperature Sensor Revision Differences

Parameter	Symbol	Test Condition	Min	Typ	Max	Unit
Revision B						
Offset	V_{OFF}	T_A = 0 °C	—	713	—	mV
Slope	M		—	2.67	—	mV/°C
Revision C						
Offset	V_{OFF}	T_A = 0 °C	—	757	—	mV
Slope	M		—	2.85	—	mV/°C

Firmware that uses the slope and offset of the temperature sensor to calculate the temperature from the sensor ADC reading can detect the revision of the device by reading the REVID register and adjust the slope and offset calculations based on the result. A REVID value of 0x01 indicates a Revision B device, and a REVID value of 0x02 indicates a Revision C device.

28.3. Flash Endurance

The flash endurance, or number of times the flash may be written and erased, on some Revision B devices may be lower than expected. Table 1.4 specifies a minimum Endurance (Write/Erase Cycles) as 20000, but some Revision B devices may support a minimum of ~5000 cycles.

28.4. Latch-Up Performance

Pulling the device pins below ground and drawing significant current (~3.5 mA) can cause a Power-On Reset event with Revision B devices. Some pins, like P0.0 and P0.1, are more susceptible to this behavior than others. This behavior is outside normal operating parameters and would typically be seen during latch-up or ESD performance testing.

28.5. Unique Identifier

Revision B devices do not implement the unique identifier described in “Device Identification and Unique Identifier” on page 68.