

Welcome to [E-XFL.COM](https://www.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	Not For New Designs
Core Processor	8051
Core Size	8-Bit
Speed	25MHz
Connectivity	SMBus (2-Wire/I ² C), SPI, UART/USART
Peripherals	Cap Sense, POR, PWM, Temp Sensor, WDT
Number of I/O	17
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 16x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	20-VFQFN Exposed Pad
Supplier Device Package	20-QFN (4x4)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f813-gm

3. Pin Definitions

Table 3.1. Pin Definitions for the C8051F80x-83x

Name	Pin QSOP-24	Pin QFN-20	Pin SOIC-16	Type	Description
GND	5	2	4		Ground. This ground connection is required. The center pad may optionally be connected to ground as well on the QFN-20 packages.
V _{DD}	6	3	5		Power Supply Voltage.
RST/ C2CK	7	4	6	D I/O D I/O	Device Reset. Open-drain output of internal POR or V _{DD} monitor. An external source can initiate a system reset by driving this pin low for at least 10 μ s. Clock signal for the C2 Debug Interface.
P2.0/ C2D	8	5	7	D I/O D I/O	Bi-directional data signal for the C2 Debug Interface. Shared with P2.0 on 20-pin packaging and P2.4 on 24-pin packaging. Bi-directional data signal for the C2 Debug Interface. Shared with P2.0 on 20-pin packaging and P2.4 on 24-pin packaging.
P0.0/ VREF	4	1	3	D I/O or A In A In	Port 0.0. External VREF input.
P0.1	3	20	2	D I/O or A In	Port 0.1.
P0.2/ XTAL1	2	19	1	D I/O or A In A In	Port 0.2. External Clock Input. This pin is the external oscillator return for a crystal or resonator.
P0.3/ XTAL2	23	18	16	D I/O or A In A I/O or D In	Port 0.3. External Clock Output. For an external crystal or resonator, this pin is the excitation driver. This pin is the external clock input for CMOS, capacitor, or RC oscillator configurations.
P0.4	22	17	15	D I/O or A In	Port 0.4.

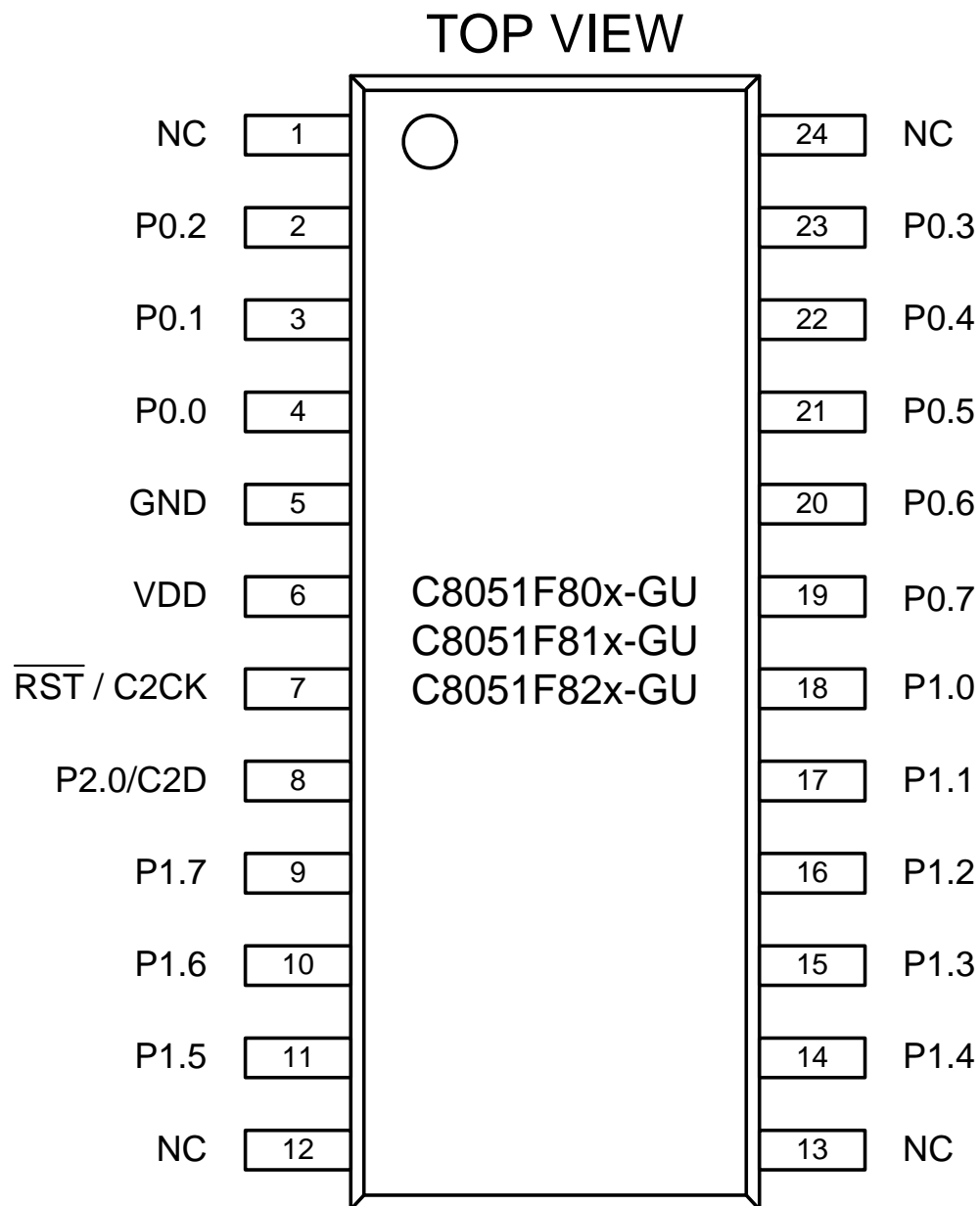


Figure 3.2. QSOP-24 Pinout Diagram (Top View)

C8051F80x-83x

Table 7.10. Power Management Electrical Characteristics

$V_{DD} = 1.8$ to 3.6 V; $T_A = -40$ to $+85$ °C unless otherwise specified. Use factory-calibrated settings.

Parameter	Conditions	Min	Typ	Max	Units
Idle Mode Wake-Up Time		2	—	3	SYSCLKs
Suspend Mode Wake-up Time		—	500	—	ns

Table 7.11. Temperature Sensor Electrical Characteristics

$V_{DD} = 3.0$ V, -40 to $+85$ °C unless otherwise specified.

Parameter	Conditions	Min	Typ	Max	Units
Linearity		—	1	—	°C
Slope		—	2.43	—	mV/°C
Slope Error*		—	±45	—	µV/°C
Offset	Temp = 0 °C	—	873	—	mV
Offset Error*	Temp = 0 °C	—	14.5	—	mV

*Note: Represents one standard deviation from the mean.

Table 7.12. Voltage Reference Electrical Characteristics

$V_{DD} = 1.8$ to 3.6 V; -40 to $+85$ °C unless otherwise specified.

Parameter	Conditions	Min	Typ	Max	Units
Internal High Speed Reference (REFSL[1:0] = 11)					
Output Voltage	25 °C ambient	1.55	1.65	1.75	V
Turn-on Time		—	—	1.7	µs
Supply Current		—	180	—	µA
External Reference (REF0E = 0)					
Input Voltage Range		0	—	V_{DD}	
Input Current	Sample Rate = 500 ksps; VREF = 3.0 V	—	7	—	µA

C8051F80x-83x

SFR Definition 8.9. ADC0MX: AMUX0 Channel Select

Bit	7	6	5	4	3	2	1	0
Name				AMX0P[3:0]				
Type	R	R	R	R/W				
Reset	0	0	0	1	1	1	1	1

SFR Address = 0xBB

Bit	Name	Function																																																																		
7:5	Unused	Read = 000b; Write = Don't Care.																																																																		
4:0	AMX0P[4:0]	AMUX0 Positive Input Selection. <table> <tr> <th></th><th>20-Pin and 24-Pin Devices</th><th>16-Pin Devices</th></tr> <tr><td>00000:</td><td>P0.0</td><td>P0.0</td></tr> <tr><td>00001:</td><td>P0.1</td><td>P0.1</td></tr> <tr><td>00010:</td><td>P0.2</td><td>P0.2</td></tr> <tr><td>00011:</td><td>P0.3</td><td>P0.3</td></tr> <tr><td>00100:</td><td>P0.4</td><td>P0.4</td></tr> <tr><td>00101:</td><td>P0.5</td><td>P0.5</td></tr> <tr><td>00110:</td><td>P0.6</td><td>P0.6</td></tr> <tr><td>00111:</td><td>P0.7</td><td>P0.7</td></tr> <tr><td>01000:</td><td>P1.0</td><td>P1.0</td></tr> <tr><td>01001:</td><td>P1.1</td><td>P1.1</td></tr> <tr><td>01010:</td><td>P1.2</td><td>P1.2</td></tr> <tr><td>01011:</td><td>P1.3</td><td>P1.3</td></tr> <tr><td>01100:</td><td>P1.4</td><td>Reserved.</td></tr> <tr><td>01101:</td><td>P1.5</td><td>Reserved.</td></tr> <tr><td>01110:</td><td>P1.6</td><td>Reserved.</td></tr> <tr><td>01111:</td><td>P1.7</td><td>Reserved.</td></tr> <tr><td>10000:</td><td>Temp Sensor</td><td>Temp Sensor</td></tr> <tr><td>10001:</td><td>VREG Output</td><td>VREG Output</td></tr> <tr><td>10010:</td><td>VDD</td><td>VDD</td></tr> <tr><td>10011:</td><td>GND</td><td>GND</td></tr> <tr><td>10100 – 11111:</td><td colspan="2">no input selected</td></tr> </table>		20-Pin and 24-Pin Devices	16-Pin Devices	00000:	P0.0	P0.0	00001:	P0.1	P0.1	00010:	P0.2	P0.2	00011:	P0.3	P0.3	00100:	P0.4	P0.4	00101:	P0.5	P0.5	00110:	P0.6	P0.6	00111:	P0.7	P0.7	01000:	P1.0	P1.0	01001:	P1.1	P1.1	01010:	P1.2	P1.2	01011:	P1.3	P1.3	01100:	P1.4	Reserved.	01101:	P1.5	Reserved.	01110:	P1.6	Reserved.	01111:	P1.7	Reserved.	10000:	Temp Sensor	Temp Sensor	10001:	VREG Output	VREG Output	10010:	VDD	VDD	10011:	GND	GND	10100 – 11111:	no input selected	
	20-Pin and 24-Pin Devices	16-Pin Devices																																																																		
00000:	P0.0	P0.0																																																																		
00001:	P0.1	P0.1																																																																		
00010:	P0.2	P0.2																																																																		
00011:	P0.3	P0.3																																																																		
00100:	P0.4	P0.4																																																																		
00101:	P0.5	P0.5																																																																		
00110:	P0.6	P0.6																																																																		
00111:	P0.7	P0.7																																																																		
01000:	P1.0	P1.0																																																																		
01001:	P1.1	P1.1																																																																		
01010:	P1.2	P1.2																																																																		
01011:	P1.3	P1.3																																																																		
01100:	P1.4	Reserved.																																																																		
01101:	P1.5	Reserved.																																																																		
01110:	P1.6	Reserved.																																																																		
01111:	P1.7	Reserved.																																																																		
10000:	Temp Sensor	Temp Sensor																																																																		
10001:	VREG Output	VREG Output																																																																		
10010:	VDD	VDD																																																																		
10011:	GND	GND																																																																		
10100 – 11111:	no input selected																																																																			

10.1. External Voltage References

To use an external voltage reference, REFSL[1:0] should be set to 00. Bypass capacitors should be added as recommended by the manufacturer of the external voltage reference.

10.2. Internal Voltage Reference Options

A 1.65 V high-speed reference is included on-chip. The high speed internal reference is selected by setting REFSL[1:0] to 11. When selected, the high speed internal reference will be automatically enabled on an as-needed basis by ADC0.

For applications with a non-varying power supply voltage, using the power supply as the voltage reference can provide ADC0 with added dynamic range at the cost of reduced power supply noise rejection. To use the 1.8 to 3.6 V power supply voltage (V_{DD}) or the 1.8 V regulated digital supply voltage as the reference source, REFSL[1:0] should be set to 01 or 10, respectively.

10.3. Analog Ground Reference

To prevent ground noise generated by switching digital logic from affecting sensitive analog measurements, a separate analog ground reference option is available. When enabled, the ground reference for ADC0 is taken from the P0.1/AGND pin. Any external sensors sampled by ADC0 should be referenced to the P0.1/AGND pin. The separate analog ground reference option is enabled by setting REFGND to 1. Note that when using this option, P0.1/AGND must be connected to the same potential as GND.

10.4. Temperature Sensor Enable

The TEMPE bit in register REF0CN enables the temperature sensor. While disabled, the temperature sensor defaults to a high impedance state and any ADC0 measurements performed on the sensor result in meaningless data.

13. Capacitive Sense (CS0)

The Capacitive Sense subsystem included on the C8051F800/1/3/4/6/7/9, C8051F810/2/3/5/6/8/9, C8051F821/2/4/5/7/8, C8051F830/1/3/4 uses a capacitance-to-digital circuit to determine the capacitance on a port pin. The module can take measurements from different port pins using the module's analog multiplexer. The multiplexer supports up to 16 channels. See SFR Definition 13.9. "CS0MX: Capacitive Sense Mux Channel Select" on page 81 for channel availability for specific part numbers. The module is enabled only when the CS0EN bit (CS0CN) is set to 1. Otherwise the module is in a low-power shutdown state. The module can be configured to take measurements on one port pin or a group of port pins, using auto-scan. An accumulator can be configured to accumulate multiple conversions on an input channel. Interrupts can be generated when CS0 completes a conversion or when the measured value crosses a threshold defined in CS0THH:L.

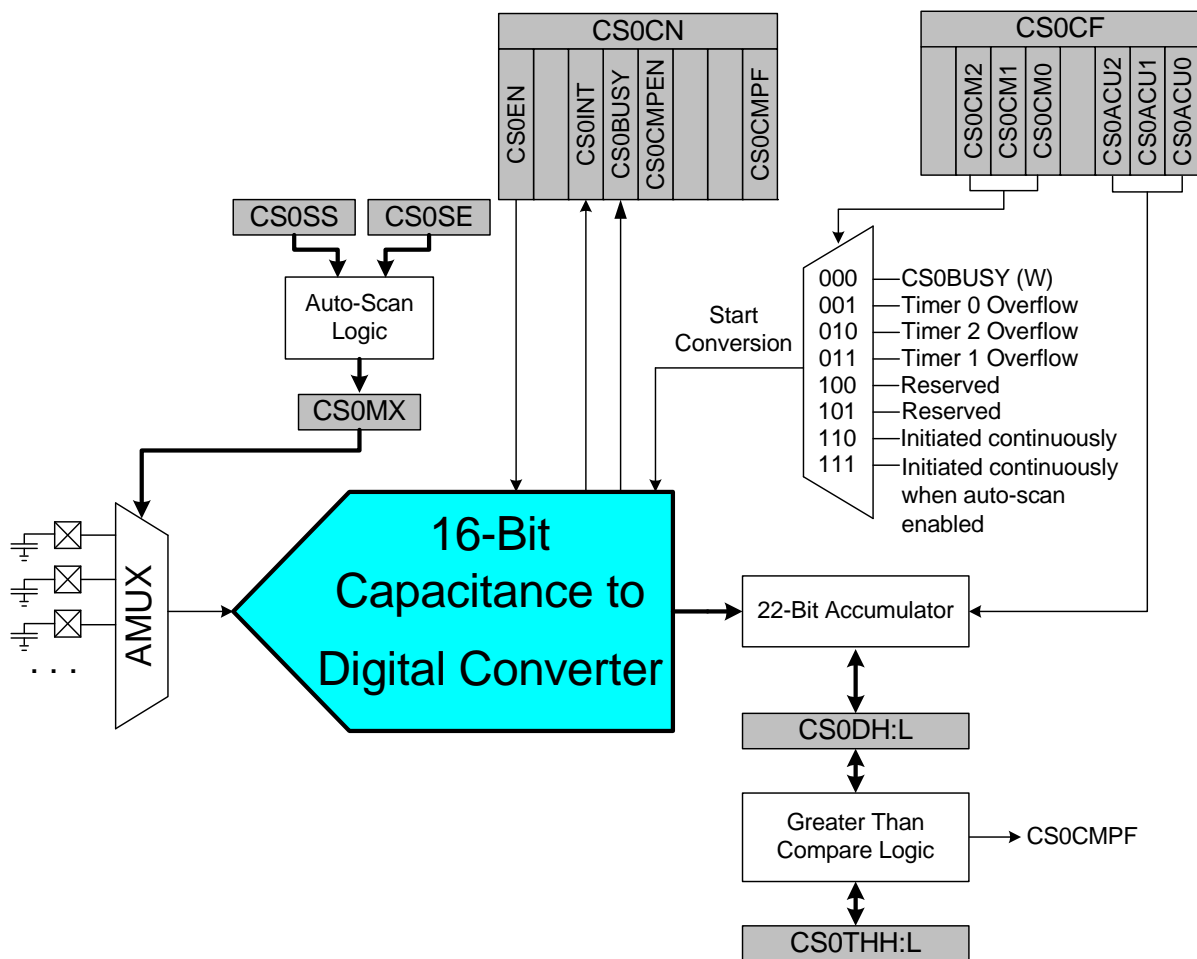


Figure 13.1. CS0 Block Diagram

13.6. Capacitive Sense Multiplexer

The input multiplexer can be controlled through two methods. The CS0MX register can be written to through firmware, or the register can be configured automatically using the modules auto-scan functionality (see “13.3. Automatic Scanning”).

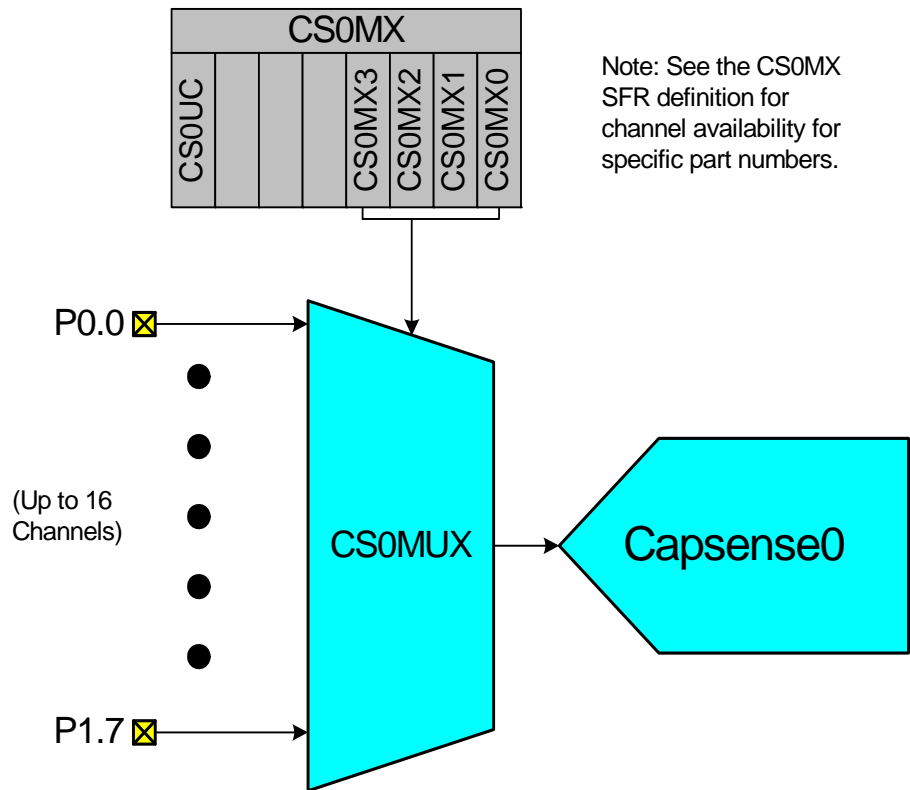


Figure 13.3. CS0 Multiplexer Block Diagram

C8051F80x-83x

SFR Definition 18.1. IE: Interrupt Enable

Bit	7	6	5	4	3	2	1	0
Name	EA	ESPI0	ET2	ES0	ET1	EX1	ET0	EX0
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 = 0xA8; Bit-Addressable

Bit	Name	Function
7	EA	Enable All Interrupts. Globally enables/disables all interrupts. It overrides individual interrupt mask settings. 0: Disable all interrupt sources. 1: Enable each interrupt according to its individual mask setting.
6	ESPI0	Enable Serial Peripheral Interface (SPI0) Interrupt. This bit sets the masking of the SPI0 interrupts. 0: Disable all SPI0 interrupts. 1: Enable interrupt requests generated by SPI0.
5	ET2	Enable Timer 2 Interrupt. This bit sets the masking of the Timer 2 interrupt. 0: Disable Timer 2 interrupt. 1: Enable interrupt requests generated by the TF2L or TF2H flags.
4	ES0	Enable UART0 Interrupt. This bit sets the masking of the UART0 interrupt. 0: Disable UART0 interrupt. 1: Enable UART0 interrupt.
3	ET1	Enable Timer 1 Interrupt. This bit sets the masking of the Timer 1 interrupt. 0: Disable all Timer 1 interrupt. 1: Enable interrupt requests generated by the TF1 flag.
2	EX1	Enable External Interrupt 1. This bit sets the masking of External Interrupt 1. 0: Disable external interrupt 1. 1: Enable interrupt requests generated by the $\overline{\text{INT1}}$ input.
1	ET0	Enable Timer 0 Interrupt. This bit sets the masking of the Timer 0 interrupt. 0: Disable all Timer 0 interrupt. 1: Enable interrupt requests generated by the TF0 flag.
0	EX0	Enable External Interrupt 0. This bit sets the masking of External Interrupt 0. 0: Disable external interrupt 0. 1: Enable interrupt requests generated by the $\overline{\text{INT0}}$ input.

8. Restore previous interrupt state.

Steps 4–6 must be repeated for each 512-byte page to be erased.

Note: Flash security settings may prevent erasure of some Flash pages, such as the reserved area and the page containing the lock bytes. For a summary of Flash security settings and restrictions affecting Flash erase operations, please see Section “19.3. Security Options” on page 114.

19.1.3. Flash Write Procedure

A write to Flash memory can clear bits to logic 0 but cannot set them; only an erase operation can set bits to logic 1 in Flash. **A byte location to be programmed should be erased before a new value is written.**

The recommended procedure for writing a single byte in Flash is as follows:

1. Save current interrupt state and disable interrupts.
2. Ensure that the Flash byte has been erased (has a value of 0xFF).
3. Set the PSWE bit (register PSCTL).
4. Clear the PSEE bit (register PSCTL).
5. Write the first key code to FLKEY: 0xA5.
6. Write the second key code to FLKEY: 0xF1.
7. Using the MOVX instruction, write a single data byte to the desired location within the 512-byte sector.
8. Clear the PSWE bit.
9. Restore previous interrupt state.

Steps 5–7 must be repeated for each byte to be written.

Note: Flash security settings may prevent writes to some areas of Flash, such as the reserved area. For a summary of Flash security settings and restrictions affecting Flash write operations, please see Section “19.3. Security Options” on page 114.

19.2. Non-volatile Data Storage

The Flash memory can be used for non-volatile data storage as well as program code. This allows data such as calibration coefficients to be calculated and stored at run time. Data is written using the MOVX write instruction and read using the MOVC instruction.

Note: MOVX read instructions always target XRAM.

19.3. Security Options

The CIP-51 provides security options to protect the Flash memory from inadvertent modification by software as well as to prevent the viewing of proprietary program code and constants. The Program Store Write Enable (bit PSWE in register PSCTL) and the Program Store Erase Enable (bit PSEE in register PSCTL) bits protect the Flash memory from accidental modification by software. PSWE must be explicitly set to 1 before software can modify the Flash memory; both PSWE and PSEE must be set to 1 before software can erase Flash memory. Additional security features prevent proprietary program code and data constants from being read or altered across the C2 interface.

A Security Lock Byte located at the last byte of Flash user space offers protection of the Flash program memory from access (reads, writes, and erases) by unprotected code or the C2 interface. The Flash security mechanism allows the user to lock all Flash pages, starting at page 0, by writing a non-0xFF value to the lock byte. **Note that writing a non-0xFF value to the lock byte will lock all pages of FLASH from reads, writes, and erases, including the page containing the lock byte.**

The level of Flash security depends on the Flash access method. The three Flash access methods that can be restricted are reads, writes, and erases from the C2 debug interface, user firmware executing on unlocked pages, and user firmware executing on locked pages. Table 19.1 summarizes the Flash security

20. Power Management Modes

The C8051F80x-83x devices have three software programmable power management modes: Idle, Stop, and Suspend. Idle mode and Stop mode are part of the standard 8051 architecture, while Suspend mode is an enhanced power-saving mode implemented by the high-speed oscillator peripheral.

Idle mode halts the CPU while leaving the peripherals and clocks active. In Stop mode, the CPU is halted, all interrupts and timers (except the Missing Clock Detector) are inactive, and the internal oscillator is stopped (analog peripherals remain in their selected states; the external oscillator is not affected). Suspend mode is similar to Stop mode in that the internal oscillator and CPU are halted, but the device can wake on events such as a Port Mismatch, Comparator low output, or a Timer 3 overflow. Since clocks are running in Idle mode, power consumption is dependent upon the system clock frequency and the number of peripherals left in active mode before entering Idle. Stop mode and Suspend mode consume the least power because the majority of the device is shut down with no clocks active. SFR Definition 20.1 describes the Power Control Register (PCON) used to control the C8051F80x-83x's Stop and Idle power management modes. Suspend mode is controlled by the SUSPEND bit in the OSCICN register (SFR Definition 22.3).

Although the C8051F80x-83x has Idle, Stop, and Suspend modes available, more control over the device power can be achieved by enabling/disabling individual peripherals as needed. Each analog peripheral can be disabled when not in use and placed in low power mode. Digital peripherals, such as timers or serial buses, draw little power when they are not in use. Turning off oscillators lowers power consumption considerably, at the expense of reduced functionality.

20.1. Idle Mode

Setting the Idle Mode Select bit (PCON.0) causes the hardware to halt the CPU and enter Idle mode as soon as the instruction that sets the bit completes execution. All internal registers and memory maintain their original data. All analog and digital peripherals can remain active during Idle mode.

Idle mode is terminated when an enabled interrupt is asserted or a reset occurs. The assertion of an enabled interrupt will cause the Idle Mode Selection bit (PCON.0) to be cleared and the CPU to resume operation. The pending interrupt will be serviced and the next instruction to be executed after the return from interrupt (RETI) will be the instruction immediately following the one that set the Idle Mode Select bit. If Idle mode is terminated by an internal or external reset, the CIP-51 performs a normal reset sequence and begins program execution at address 0x0000.

Note: If the instruction following the write of the IDLE bit is a single-byte instruction and an interrupt occurs during the execution phase of the instruction that sets the IDLE bit, the CPU may not wake from Idle mode when a future interrupt occurs. Therefore, instructions that set the IDLE bit should be followed by an instruction that has two or more opcode bytes, for example:

```
// in 'C':
PCON |= 0x01;           // set IDLE bit
PCON = PCON;           // ... followed by a 3-cycle dummy instruction

; in assembly:
ORL PCON, #01h          ; set IDLE bit
MOV PCON, PCON          ; ... followed by a 3-cycle dummy instruction
```

If enabled, the Watchdog Timer (WDT) will eventually cause an internal watchdog reset and thereby terminate the Idle mode. This feature protects the system from an unintended permanent shutdown in the event of an inadvertent write to the PCON register. If this behavior is not desired, the WDT may be disabled by software prior to entering the Idle mode if the WDT was initially configured to allow this operation. This provides the opportunity for additional power savings, allowing the system to remain in the Idle mode indefinitely, waiting for an external stimulus to wake up the system. Refer to Section "29.4. Watchdog Timer Mode" on page 236 for more information on the use and configuration of the WDT.

21. Reset Sources

Reset circuitry allows the controller to be easily placed in a predefined default condition. On entry to this reset state, the following occur:

- CIP-51 halts program execution
- Special Function Registers (SFRs) are initialized to their defined reset values
- External Port pins are forced to a known state
- Interrupts and timers are disabled.

All SFRs are reset to the predefined values noted in the SFR detailed descriptions. The contents of internal data memory are unaffected during a reset; any previously stored data is preserved. However, since the stack pointer SFR is reset, the stack is effectively lost, even though the data on the stack is not altered.

The Port I/O latches are reset to 0xFF (all logic ones) in open-drain mode. Weak pullups are enabled during and after the reset. For V_{DD} Monitor and power-on resets, the RST pin is driven low until the device exits the reset state.

On exit from the reset state, the program counter (PC) is reset, and the system clock defaults to the internal oscillator. The Watchdog Timer is enabled with the system clock divided by 12 as its clock source. Program execution begins at location 0x0000.

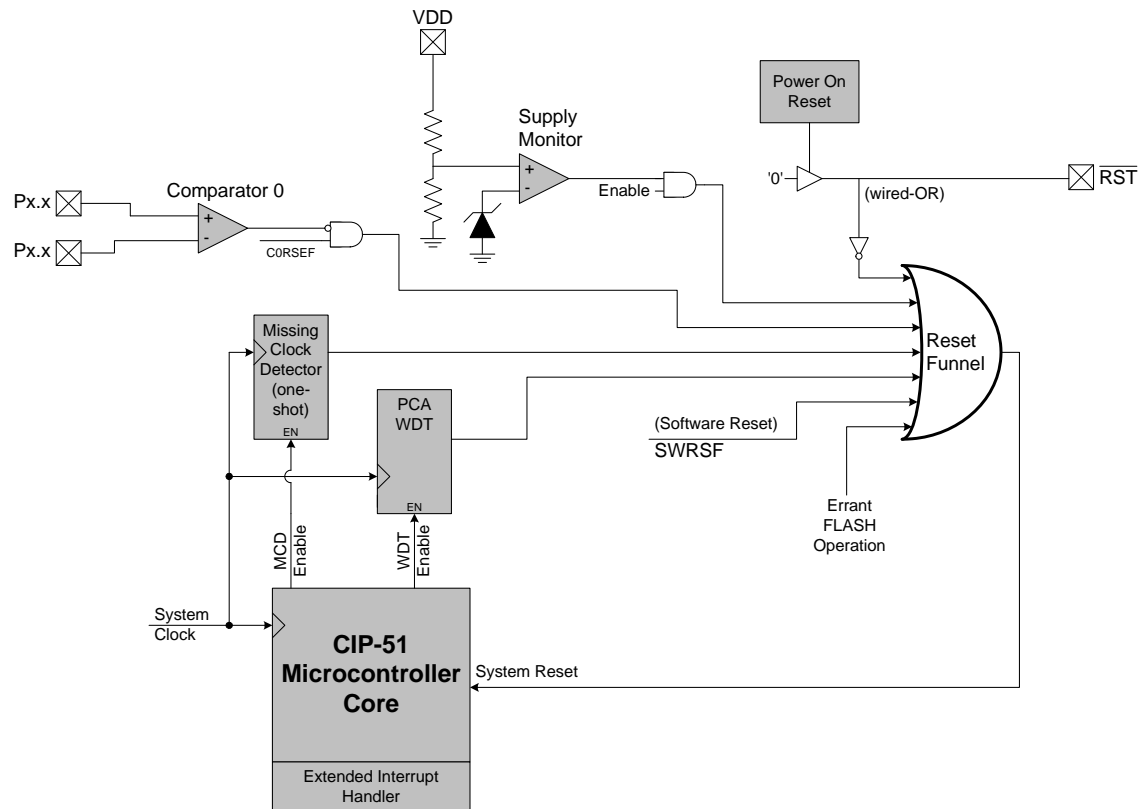


Figure 21.1. Reset Sources

21.2. Power-Fail Reset / V_{DD} Monitor

When a power-down transition or power irregularity causes V_{DD} to drop below V_{RST} , the power supply monitor will drive the \overline{RST} pin low and hold the CIP-51 in a reset state (see Figure 21.2). When V_{DD} returns to a level above V_{RST} , the CIP-51 will be released from the reset state. Even though internal data memory contents are not altered by the power-fail reset, it is impossible to determine if V_{DD} dropped below the level required for data retention. If the PORSF flag reads 1, the data may no longer be valid. The V_{DD} monitor is enabled and selected as a reset source after power-on resets. Its defined state (enabled/disabled) is not altered by any other reset source. For example, if the V_{DD} monitor is disabled by code and a software reset is performed, the V_{DD} monitor will still be disabled after the reset.

Important Note: If the V_{DD} monitor is being turned on from a disabled state, it should be enabled before it is selected as a reset source. Selecting the V_{DD} monitor as a reset source before it is enabled and stabilized may cause a system reset. In some applications, this reset may be undesirable. If this is not desirable in the application, a delay should be introduced between enabling the monitor and selecting it as a reset source. The procedure for enabling the V_{DD} monitor and configuring it as a reset source from a disabled state is shown below:

1. Enable the V_{DD} monitor (VDMEN bit in VDM0CN = 1).
2. If necessary, wait for the V_{DD} monitor to stabilize.
3. Select the V_{DD} monitor as a reset source (PORSF bit in RSTSRC = 1).

See Figure 21.2 for V_{DD} monitor timing; note that the power-on-reset delay is not incurred after a V_{DD} monitor reset. See Section “7. Electrical Characteristics” on page 39 for complete electrical characteristics of the V_{DD} monitor.

22.3. External Oscillator Drive Circuit

The external oscillator circuit may drive an external crystal, ceramic resonator, capacitor, or RC network. A CMOS clock may also provide a clock input. For a crystal or ceramic resonator configuration, the crystal/resonator must be wired across the XTAL1 and XTAL2 pins as shown in Option 1 of Figure 22.1. A 10 M Ω resistor also must be wired across the XTAL2 and XTAL1 pins for the crystal/resonator configuration. In RC, capacitor, or CMOS clock configuration, the clock source should be wired to the XTAL2 pin as shown in Option 2, 3, or 4 of Figure 22.1. The type of external oscillator must be selected in the OSCXCN register, and the frequency control bits (XFCN) must be selected appropriately (see SFR Definition 22.4).

Important Note on External Oscillator Usage: Port pins must be configured when using the external oscillator circuit. When the external oscillator drive circuit is enabled in crystal/resonator mode, Port pins P0.2 and P0.3 are used as XTAL1 and XTAL2 respectively. When the external oscillator drive circuit is enabled in capacitor, RC, or CMOS clock mode, Port pin P0.3 is used as XTAL2. The Port I/O Crossbar should be configured to skip the Port pins used by the oscillator circuit; see Section “23.3. Priority Crossbar Decoder” on page 143 for Crossbar configuration. Additionally, when using the external oscillator circuit in crystal/resonator, capacitor, or RC mode, the associated Port pins should be configured as **analog inputs**. In CMOS clock mode, the associated pin should be configured as a **digital input**. See Section “23.4. Port I/O Initialization” on page 147 for details on Port input mode selection.

23.1. Port I/O Modes of Operation

Port pins P0.0–P1.7 use the Port I/O cell shown in Figure 23.2. Each Port I/O cell can be configured by software for analog I/O or digital I/O using the PnMDIN and PnMDOUT registers. Port pin P2.0 can be configured by software for digital I/O using the P2MDOUT register. On reset, all Port I/O cells default to a high impedance state with weak pull-ups enabled. Until the crossbar is enabled (XBARE = 1), both the high and low port I/O drive circuits are explicitly disabled on all crossbar pins.

23.1.1. Port Pins Configured for Analog I/O

Any pins to be used as Comparator or ADC input, Capacitive Sense input, external oscillator input/output, VREF output, or AGND connection should be configured for analog I/O (PnMDIN.n = 0, Pn.n = 1). When a pin is configured for analog I/O, its weak pullup, digital driver, and digital receiver are disabled. To prevent the low port I/O drive circuit from pulling the pin low, a '1' should be written to the corresponding port latch (Pn.n = 1). Port pins configured for analog I/O will always read back a value of 0 regardless of the actual voltage on the pin.

Configuring pins as analog I/O saves power and isolates the Port pin from digital interference. Port pins configured as digital I/O may still be used by analog peripherals; however, this practice is not recommended and may result in measurement errors.

23.1.2. Port Pins Configured For Digital I/O

Any pins to be used by digital peripherals (UART, SPI, SMBus, etc.), external digital event capture functions, or as GPIO should be configured as digital I/O (PnMDIN.n = 1). For digital I/O pins, one of two output modes (push-pull or open-drain) must be selected using the PnMDOUT registers.

Push-pull outputs (PnMDOUT.n = 1) drive the Port pad to the VDD or GND supply rails based on the output logic value of the Port pin. Open-drain outputs have the high side driver disabled; therefore, they only drive the Port pad to GND when the output logic value is 0 and become high impedance inputs (both high and low drivers turned off) when the output logic value is 1.

When a digital I/O cell is placed in the high impedance state, a weak pull-up transistor pulls the Port pad to the VDD supply voltage to ensure the digital input is at a defined logic state. Weak pull-ups are disabled when the I/O cell is driven to GND to minimize power consumption and may be globally disabled by setting WEAKPUD to 1. The user should ensure that digital I/O are always internally or externally pulled or driven to a valid logic state to minimize power consumption. Port pins configured for digital I/O always read back the logic state of the Port pad, regardless of the output logic value of the Port pin.

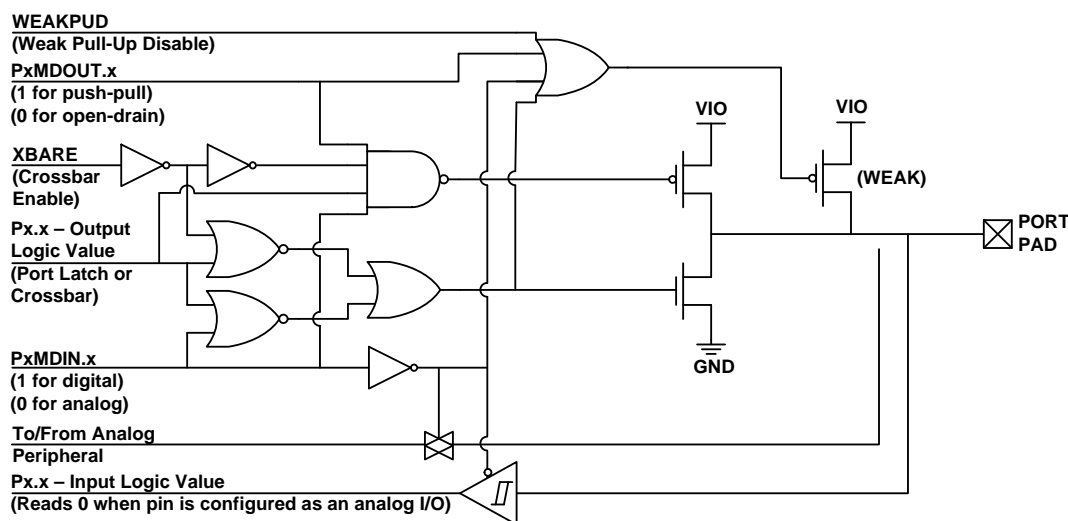


Figure 23.2. Port I/O Cell Block Diagram

23.4. Port I/O Initialization

Port I/O initialization consists of the following steps:

1. Select the input mode (analog or digital) for all Port pins, using the Port Input Mode register (PnMDIN). **If the pin is in analog mode, a '1' must also be written to the corresponding Port Latch (Pn).**
2. Select the output mode (open-drain or push-pull) for all Port pins, using the Port Output Mode register (PnMDOUT).
3. Select any pins to be skipped by the I/O Crossbar using the Port Skip registers (PnSKIP).
4. Assign Port pins to desired peripherals (XBR0, XBR1).
5. Enable the Crossbar (XBARE = 1).

All Port pins must be configured as either analog or digital inputs. When a pin is configured as an analog input, its weak pullup, digital driver, and digital receiver are disabled. This process saves power and reduces noise on the analog input. Pins configured as digital inputs may still be used by analog peripherals; however this practice is not recommended.

Additionally, all analog input pins should be configured to be skipped by the Crossbar (accomplished by setting the associated bits in PnSKIP). Port input mode is set in the PnMDIN register, where a 1 indicates a digital input, and a 0 indicates an analog input. All port pins in analog mode must have a '1' set in the corresponding Port Latch register. All pins default to digital inputs on reset. See SFR Definition 23.8 and SFR Definition 23.12 for the PnMDIN register details.

The output driver characteristics of the I/O pins are defined using the Port Output Mode registers (PnMDOUT). Each Port Output driver can be configured as either open drain or push-pull. This selection is required even for the digital resources selected in the XBRn registers, and is not automatic. The only exception to this is the SMBus (SDA, SCL) pins, which are configured as open-drain regardless of the PnMDOUT settings. When the WEAKPUD bit in XBR1 is 0, a weak pullup is enabled for all Port I/O configured as open-drain. WEAKPUD does not affect the push-pull Port I/O. Furthermore, the weak pullup is turned off on an output that is driving a 0 to avoid unnecessary power dissipation.

Registers XBR0 and XBR1 must be loaded with the appropriate values to select the digital I/O functions required by the design. Setting the XBARE bit in XBR1 to 1 enables the Crossbar. Until the Crossbar is enabled, the external pins remain as standard Port I/O (in input mode), regardless of the XBRn Register settings. For given XBRn Register settings, one can determine the I/O pin-out using the Priority Decode Table; as an alternative, the Configuration Wizard utility will determine the Port I/O pin-assignments based on the XBRn Register settings.

The Crossbar must be enabled to use Port pins as standard Port I/O in output mode. Port output drivers are disabled while the Crossbar is disabled.

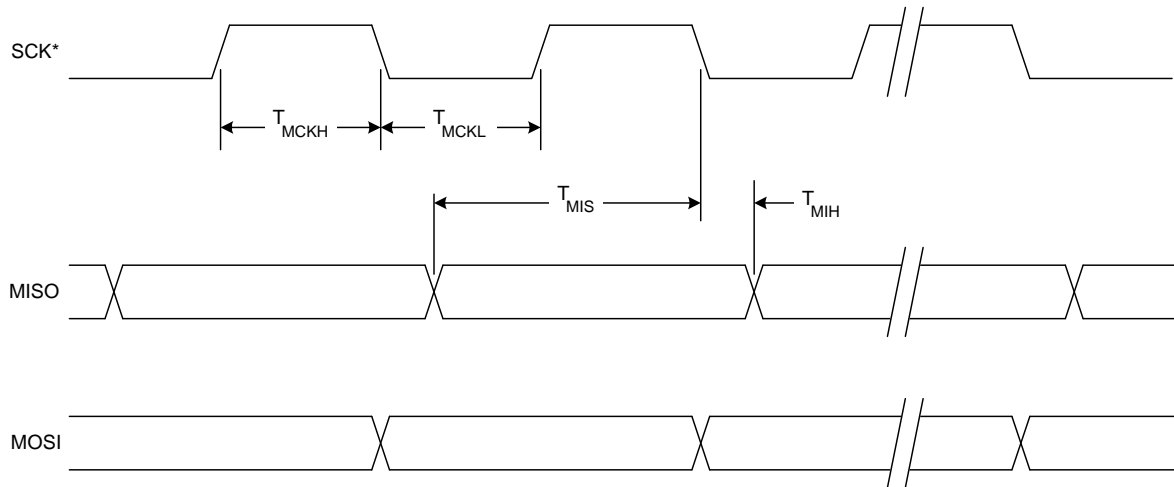
C8051F80x-83x

SFR Definition 23.2. XBR1: Port I/O Crossbar Register 1

Bit	7	6	5	4	3	2	1	0
Name	WEAKPUD	XBARE	T1E	T0E	ECIE		PCA0ME[1:0]	
Type	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W
Reset	0	0	0	0	0	0	0	0

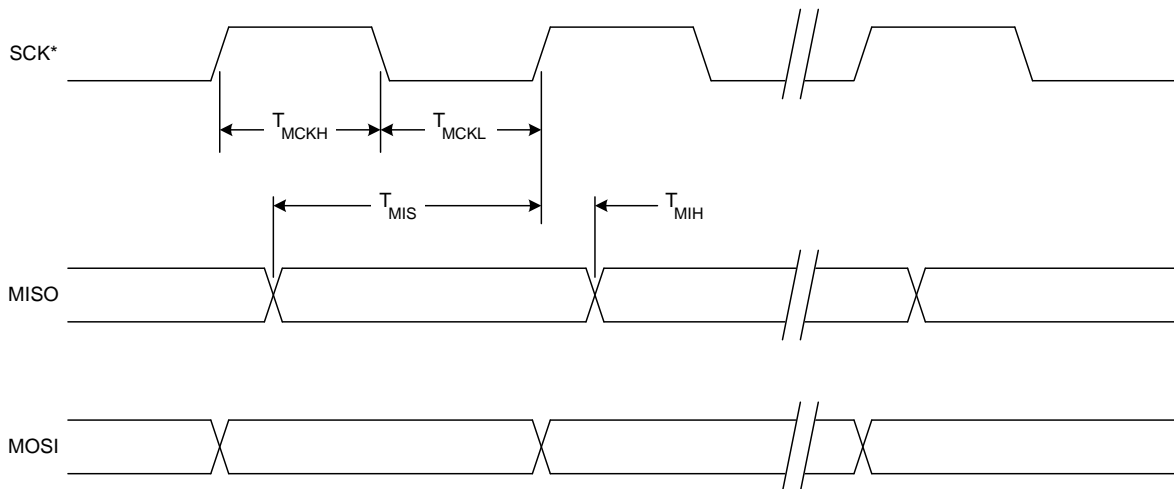
SFR Address = 0xE2

Bit	Name	Function
7	WEAKPUD	Port I/O Weak Pullup Disable. 0: Weak Pullups enabled (except for Ports whose I/O are configured for analog mode). 1: Weak Pullups disabled.
6	XBARE	Crossbar Enable. 0: Crossbar disabled. 1: Crossbar enabled.
5	T1E	T1 Enable. 0: T1 unavailable at Port pin. 1: T1 routed to Port pin.
4	T0E	T0 Enable. 0: T0 unavailable at Port pin. 1: T0 routed to Port pin.
3	ECIE	PCA0 External Counter Input Enable. 0: ECI unavailable at Port pin. 1: ECI routed to Port pin.
2	Unused	Read = 0b; Write = Don't Care.
1:0	PCA0ME[1:0]	PCA Module I/O Enable Bits. 00: All PCA I/O unavailable at Port pins. 01: CEX0 routed to Port pin. 10: CEX0, CEX1 routed to Port pins. 11: CEX0, CEX1, CEX2 routed to Port pins.



* SCK is shown for CKPOL = 0. SCK is the opposite polarity for CKPOL = 1.

Figure 25.8. SPI Master Timing (CKPHA = 0)



* SCK is shown for CKPOL = 0. SCK is the opposite polarity for CKPOL = 1.

Figure 25.9. SPI Master Timing (CKPHA = 1)

Table 26.5. SMBus Status Decoding With Hardware ACK Generation Disabled (EHACK = 0)
(Continued)

Mode	Values Read				Current SMBus State	Typical Response Options	Values to Write			Next Status Vector Expected
	Status Vector	ACKRQ	ARBLOST	ACK			STA	STO	ACK	
Slave Transmitter	0100	0	0	0	A slave byte was transmitted; NACK received.	No action required (expecting STOP condition).	0	0	X	0001
		0	0	1	A slave byte was transmitted; ACK received.	Load SMB0DAT with next data byte to transmit.	0	0	X	0100
		0	1	X	A Slave byte was transmitted; error detected.	No action required (expecting Master to end transfer).	0	0	X	0001
	0101	0	X	X	An illegal STOP or bus error was detected while a Slave Transmission was in progress.	Clear STO.	0	0	X	—
Slave Receiver	0010	1	0	X	A slave address + R/W was received; ACK requested.	If Write, Acknowledge received address	0	0	1	0000
						If Read, Load SMB0DAT with data byte; ACK received address	0	0	1	0100
						NACK received address.	0	0	0	—
	0010	1	1	X	Lost arbitration as master; slave address + R/W received; ACK requested.	If Write, Acknowledge received address	0	0	1	0000
						If Read, Load SMB0DAT with data byte; ACK received address	0	0	1	0100
						NACK received address.	0	0	0	—
						Reschedule failed transfer; NACK received address.	1	0	0	1110
	0001	0	0	X	A STOP was detected while addressed as a Slave Transmitter or Slave Receiver.	Clear STO.	0	0	X	—
						Lost arbitration while attempting a STOP.	0	0	0	—
	0000	1	0	X	A slave byte was received; ACK requested.	Acknowledge received byte; Read SMB0DAT.	0	0	1	0000
						NACK received byte.	0	0	0	—
Bus Error Condition	0010	0	1	X	Lost arbitration while attempting a repeated START.	Abort failed transfer.	0	0	X	—
						Reschedule failed transfer.	1	0	X	1110
	0001	0	1	X	Lost arbitration due to a detected STOP.	Abort failed transfer.	0	0	X	—
						Reschedule failed transfer.	1	0	X	1110
	0000	1	1	X	Lost arbitration while transmitting a data byte as master.	Abort failed transfer.	0	0	0	—
						Reschedule failed transfer.	1	0	0	1110

27.2. Operational Modes

UART0 provides standard asynchronous, full duplex communication. The UART mode (8-bit or 9-bit) is selected by the S0MODE bit (SCON0.7). Typical UART connection options are shown in Figure 27.3.

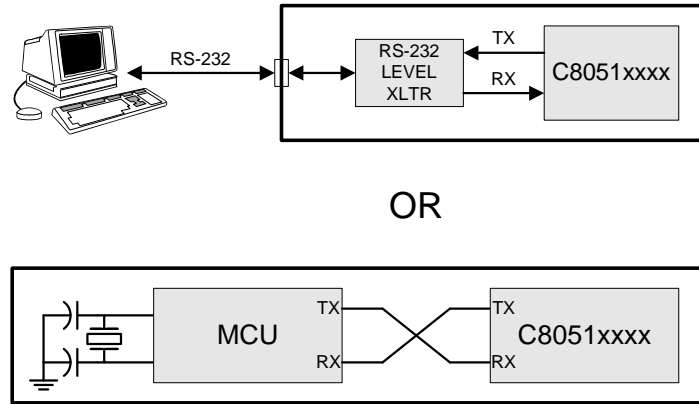


Figure 27.3. UART Interconnect Diagram

27.2.1. 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 TX0 pin and received at the RX0 pin. On receive, the eight data bits are stored in SBUF0 and the stop bit goes into RB80 (SCON0.2).

Data transmission begins when software 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 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: RI0 must be logic 0, and if MCE0 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 RB80 and the RI0 flag is set. If these conditions are not met, SBUF0 and RB80 will not be loaded and the RI0 flag will not be set. An interrupt will occur if enabled when either TI0 or RI0 is set.

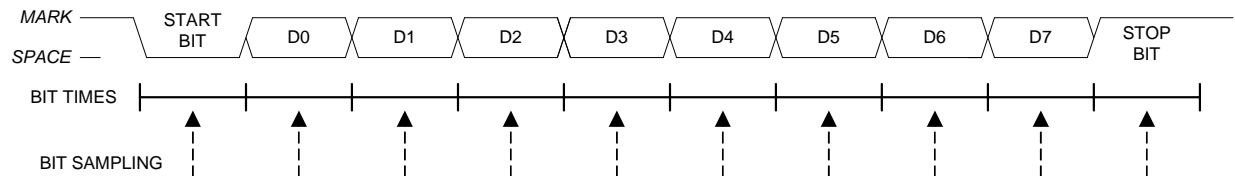


Figure 27.4. 8-Bit UART Timing Diagram

29.3.4. Frequency Output Mode

Frequency Output Mode produces a programmable-frequency square wave on the module's associated CEXn pin. The capture/compare module high byte holds the number of PCA clocks to count before the output is toggled. The frequency of the square wave is then defined by Equation 29.1.

$$F_{CEXn} = \frac{F_{PCA}}{2 \times PCA0CPHn}$$

Note: A value of 0x00 in the PCA0CPHn register is equal to 256 for this equation.

Equation 29.1. Square Wave Frequency Output

Where F_{PCA} is the frequency of the clock selected by the CPS2–0 bits in the PCA mode register, PCA0MD. The lower byte of the capture/compare module is compared to the PCA counter low byte; on a match, CEXn is toggled and the offset held in the high byte is added to the matched value in PCA0CPLn. Frequency Output Mode is enabled by setting the ECOMn, TOGn, and PWMn bits in the PCA0CPMn register. The MATn bit should normally be set to 0 in this mode. If the MATn bit is set to 1, the CCFn flag for the channel will be set when the 16-bit PCA0 counter and the 16-bit capture/compare register for the channel are equal.

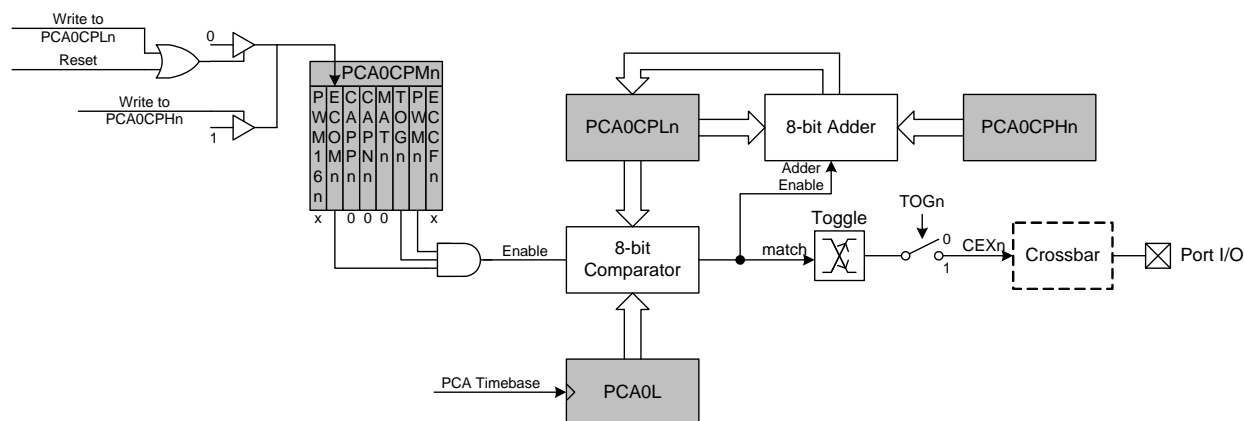


Figure 29.7. PCA Frequency Output Mode

29.3.5. 8-bit through 15-bit Pulse Width Modulator Modes

Each module can be used independently to generate a pulse width modulated (PWM) output on its associated CEXn pin. The frequency of the output is dependent on the timebase for the PCA counter/timer, and the setting of the PWM cycle length (8, 9, 10, 11, 12, 13, 14, or 15-bits). For backwards-compatibility with the 8-bit PWM mode available on other devices, the 8-bit PWM mode operates slightly different than 9, 10, 11, 12, 13, 14, and 15-bit PWM modes. **It is important to note that all channels configured for 8-bit through 15-bit PWM mode will use the same cycle length.** For example, it is not possible to configure one channel for 8-bit PWM mode and another for 11-bit mode. However, other PCA channels can be configured to Pin Capture, High-Speed Output, Software Timer, Frequency Output, or 16-bit PWM mode independently.