

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, WDT
Number of I/O	17
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	20-VQFN Exposed Pad
Supplier Device Package	20-QFN (4x4)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f807-gmr

Table of Contents

1. System Overview	15
2. Ordering Information	25
3. Pin Definitions.....	28
4. QFN-20 Package Specifications	33
5. QSOP-24 Package Specifications	35
6. SOIC-16 Package Specifications	37
7. Electrical Characteristics	39
7.1. Absolute Maximum Specifications	39
7.2. Electrical Characteristics	40
8. 10-Bit ADC (ADC0).....	46
8.1. Output Code Formatting	47
8.2. 8-Bit Mode	47
8.3. Modes of Operation	47
8.3.1. Starting a Conversion.....	47
8.3.2. Tracking Modes.....	48
8.3.3. Settling Time Requirements.....	49
8.4. Programmable Window Detector.....	53
8.4.1. Window Detector Example.....	55
8.5. ADC0 Analog Multiplexer	56
9. Temperature Sensor	58
9.1. Calibration	58
10. Voltage and Ground Reference Options.....	60
10.1. External Voltage References.....	61
10.2. Internal Voltage Reference Options	61
10.3. Analog Ground Reference.....	61
10.4. Temperature Sensor Enable	61
11. Voltage Regulator (REG0).....	63
12. Comparator0.....	65
12.1. Comparator Multiplexer	69
13. Capacitive Sense (CS0).....	71
13.1. Configuring Port Pins as Capacitive Sense Inputs	72
13.2. Capacitive Sense Start-Of-Conversion Sources	72
13.3. Automatic Scanning.....	72
13.4. CS0 Comparator.....	73
13.5. CS0 Conversion Accumulator	74
13.6. Capacitive Sense Multiplexer	80
14. CIP-51 Microcontroller.....	82
14.1. Instruction Set.....	83
14.1.1. Instruction and CPU Timing	83
14.2. CIP-51 Register Descriptions	88
15. Memory Organization	92
15.1. Program Memory.....	93
15.1.1. MOVX Instruction and Program Memory	93

C8051F80x-83x

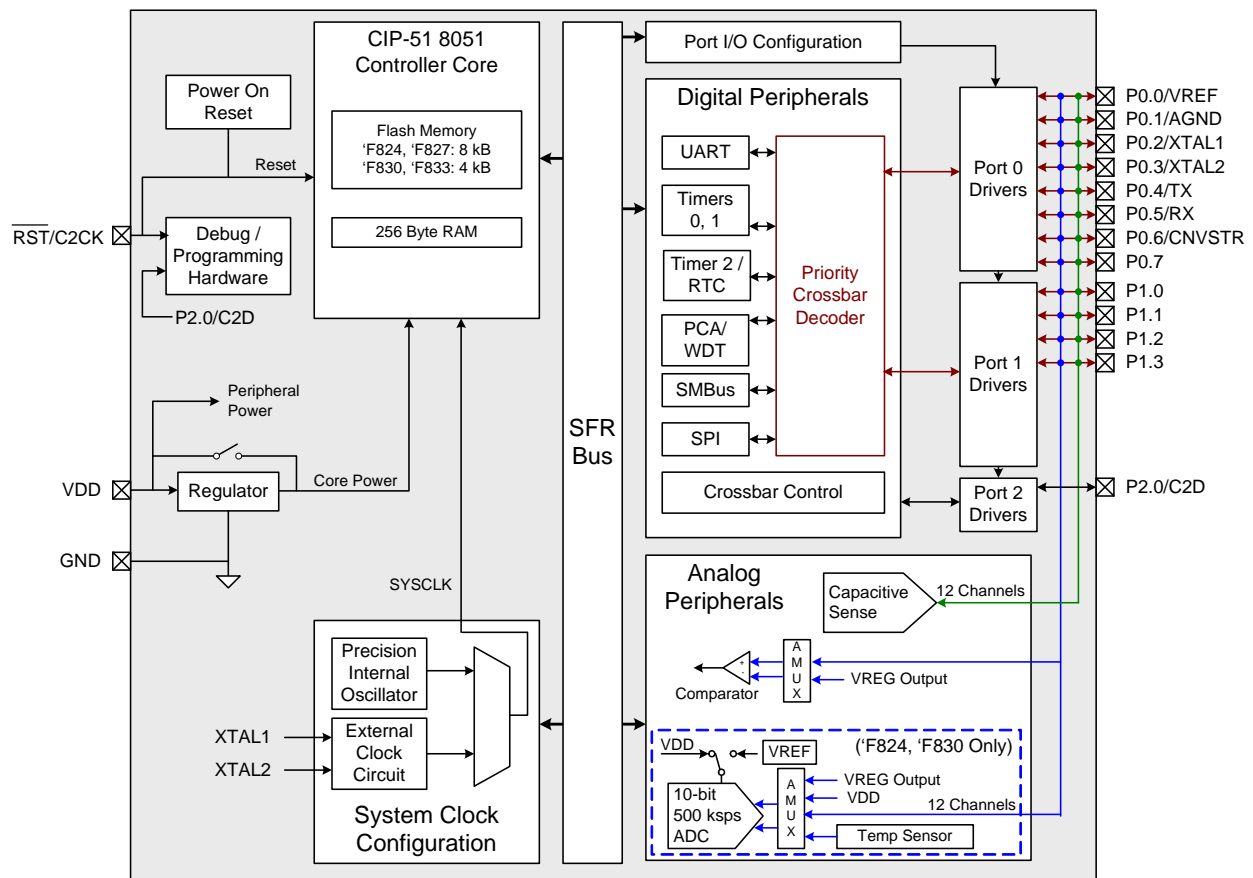


Figure 1.7. C8051F824, C8051F827, C8051F830, C8051F833 Block Diagram

SFR Definition 8.1. ADC0CF: ADC0 Configuration

Bit	7	6	5	4	3	2	1	0
Name	AD0SC[4:0]					AD0LJST	AD08BE	AMP0GN0
Type	R/W					R/W	R/W	R/W
Reset	1	1	1	1	1	0	0	1

SFR Address = 0xBC

Bit	Name	Function
7:3	AD0SC[4:0]	ADC0 SAR Conversion Clock Period Bits. SAR Conversion clock is derived from system clock by the following equation, where <i>AD0SC</i> refers to the 5-bit value held in bits AD0SC4–0. SAR Conversion clock requirements are given in the ADC specification table. $AD0SC = \frac{SYSCLK}{CLK_{SAR}} - 1$
2	AD0LJST	ADC0 Left Justify Select. 0: Data in ADC0H:ADC0L registers are right-justified. 1: Data in ADC0H:ADC0L registers are left-justified. Note: The AD0LJST bit is only valid for 10-bit mode (AD08BE = 0).
1	AD08BE	8-Bit Mode Enable. 0: ADC operates in 10-bit mode (normal). 1: ADC operates in 8-bit mode. Note: When AD08BE is set to 1, the AD0LJST bit is ignored.
0	AMP0GN0	ADC Gain Control Bit. 0: Gain = 0.5 1: Gain = 1

8.4.1. Window Detector Example

Figure 8.4 shows two example window comparisons for right-justified data, with $ADC0LTH:ADC0LTL = 0x0080$ (128d) and $ADC0GTH:ADC0GTL = 0x0040$ (64d). The input voltage can range from 0 to $VREF \times (1023/1024)$ with respect to GND, and is represented by a 10-bit unsigned integer value. In the left example, an $AD0WINT$ interrupt will be generated if the $ADC0$ conversion word ($ADC0H:ADC0L$) is within the range defined by $ADC0GTH:ADC0GTL$ and $ADC0LTH:ADC0LTL$ (if $0x0040 < ADC0H:ADC0L < 0x0080$). In the right example, an $AD0WINT$ interrupt will be generated if the $ADC0$ conversion word is outside of the range defined by the $ADC0GT$ and $ADC0LT$ registers (if $ADC0H:ADC0L < 0x0040$ or $ADC0H:ADC0L > 0x0080$). Figure 8.5 shows an example using left-justified data with the same comparison values.

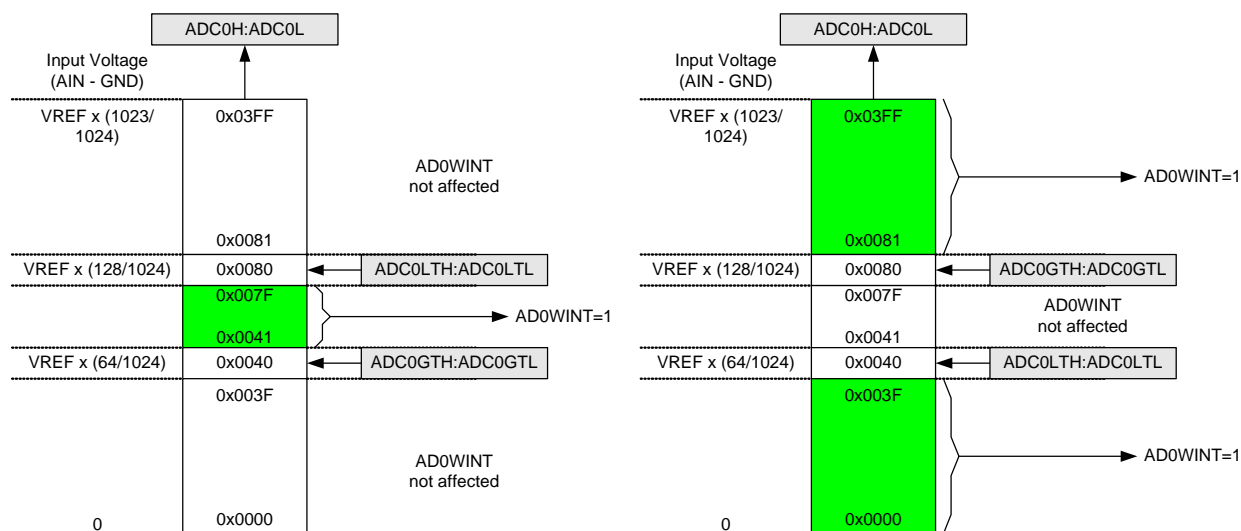


Figure 8.4. ADC Window Compare Example: Right-Justified Data

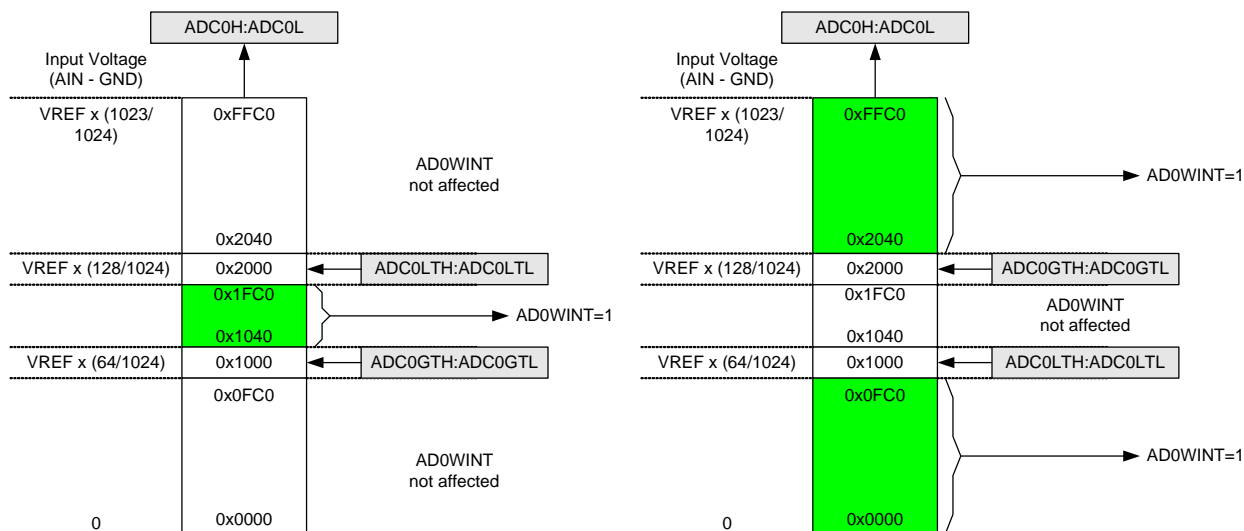


Figure 8.5. ADC Window Compare Example: Left-Justified Data

11. Voltage Regulator (REG0)

C8051F80x-83x devices include an internal voltage regulator (REG0) to regulate the internal core supply to 1.8 V from a V_{DD} supply of 1.8 to 3.6 V. A power-saving mode is built into the regulator to help reduce current consumption in low-power applications. This mode is accessed through the REG0CN register (SFR Definition 11.1). Electrical characteristics for the on-chip regulator are specified in Table 7.5 on page 41

Under default conditions, when the device enters STOP mode the internal regulator will remain on. This allows any enabled reset source to generate a reset for the device and bring the device out of STOP mode. For additional power savings, the STOPCF bit can be used to shut down the regulator and the internal power network of the device when the part enters STOP mode. When STOPCF is set to 1, the RST pin or a full power cycle of the device are the only methods of generating a reset.

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.

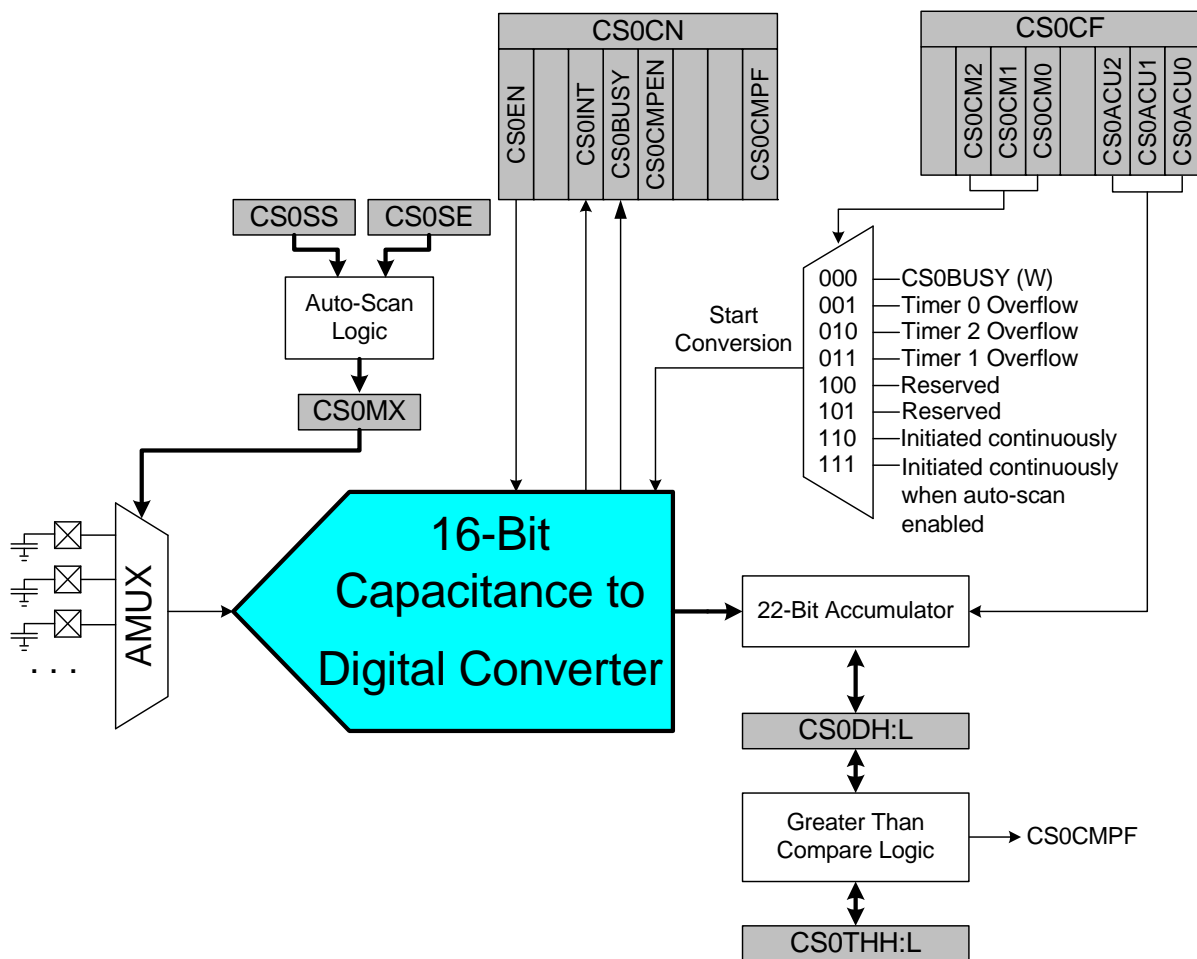


Figure 13.1. CS0 Block Diagram

features of the C8051F80x-83x devices.

Table 19.1. Flash Security Summary

Action	C2 Debug Interface	User Firmware executing from:	
		an unlocked page	a locked page
Read, Write or Erase unlocked pages (except page with Lock Byte)	Permitted	Permitted	Permitted
Read, Write or Erase locked pages (except page with Lock Byte)	Not Permitted	FEDR	Permitted
Read or Write page containing Lock Byte (if no pages are locked)	Permitted	Permitted	Permitted
Read or Write page containing Lock Byte (if any page is locked)	Not Permitted	FEDR	Permitted
Read contents of Lock Byte (if no pages are locked)	Permitted	Permitted	Permitted
Read contents of Lock Byte (if any page is locked)	Not Permitted	FEDR	Permitted
Erase page containing Lock Byte (if no pages are locked)	Permitted	FEDR	FEDR
Erase page containing Lock Byte—Unlock all pages (if any page is locked)	Only by C2DE	FEDR	FEDR
Lock additional pages (change 1s to 0s in the Lock Byte)	Not Permitted	FEDR	FEDR
Unlock individual pages (change 0s to 1s in the Lock Byte)	Not Permitted	FEDR	FEDR
Read, Write or Erase Reserved Area	Not Permitted	FEDR	FEDR
<p>C2DE—C2 Device Erase (Erases all Flash pages including the page containing the Lock Byte) FEDR—Not permitted; Causes Flash Error Device Reset (FERROR bit in RSTSRC is 1 after reset)</p> <ul style="list-style-type: none"> ■ All prohibited operations that are performed via the C2 interface are ignored (do not cause device reset). ■ Locking any Flash page also locks the page containing the Lock Byte. ■ Once written to, the Lock Byte cannot be modified except by performing a C2 Device Erase. ■ If user code writes to the Lock Byte, the Lock does not take effect until the next device reset. 			

19.4. Flash Write and Erase Guidelines

Any system which contains routines which write or erase Flash memory from software involves some risk that the write or erase routines will execute unintentionally if the CPU is operating outside its specified operating range of VDD, system clock frequency, or temperature. This accidental execution of Flash modifying code can result in alteration of Flash memory contents causing a system failure that is only recoverable by re-Flashing the code in the device.

To help prevent the accidental modification of Flash by firmware, the VDD Monitor must be enabled and enabled as a reset source on C8051F80x-83x devices for the Flash to be successfully modified. **If either the VDD Monitor or the VDD Monitor reset source is not enabled, a Flash Error Device Reset will be generated when the firmware attempts to modify the Flash.**

21.1. Power-On Reset

During power-up, the device is held in a reset state and the $\overline{\text{RST}}$ pin is driven low until V_{DD} settles above V_{RST} . A delay occurs before the device is released from reset; the delay decreases as the V_{DD} ramp time increases (V_{DD} ramp time is defined as how fast V_{DD} ramps from 0 V to V_{RST}). Figure 21.2. plots the power-on and V_{DD} monitor reset timing. The maximum V_{DD} ramp time is 1 ms; slower ramp times may cause the device to be released from reset before V_{DD} reaches the V_{RST} level. For ramp times less than 1 ms, the power-on reset delay (T_{PORDelay}) is typically less than 10 ms.

On exit from a power-on reset, the PORSF flag (RSTSRC.1) is set by hardware to logic 1. When PORSF is set, all of the other reset flags in the RSTSRC Register are indeterminate (PORSF is cleared by all other resets). Since all resets cause program execution to begin at the same location (0x0000) software can read the PORSF flag to determine if a power-up was the cause of reset. The content of internal data memory should be assumed to be undefined after a power-on reset. The V_{DD} monitor is enabled and selected as a reset source following a power-on reset.

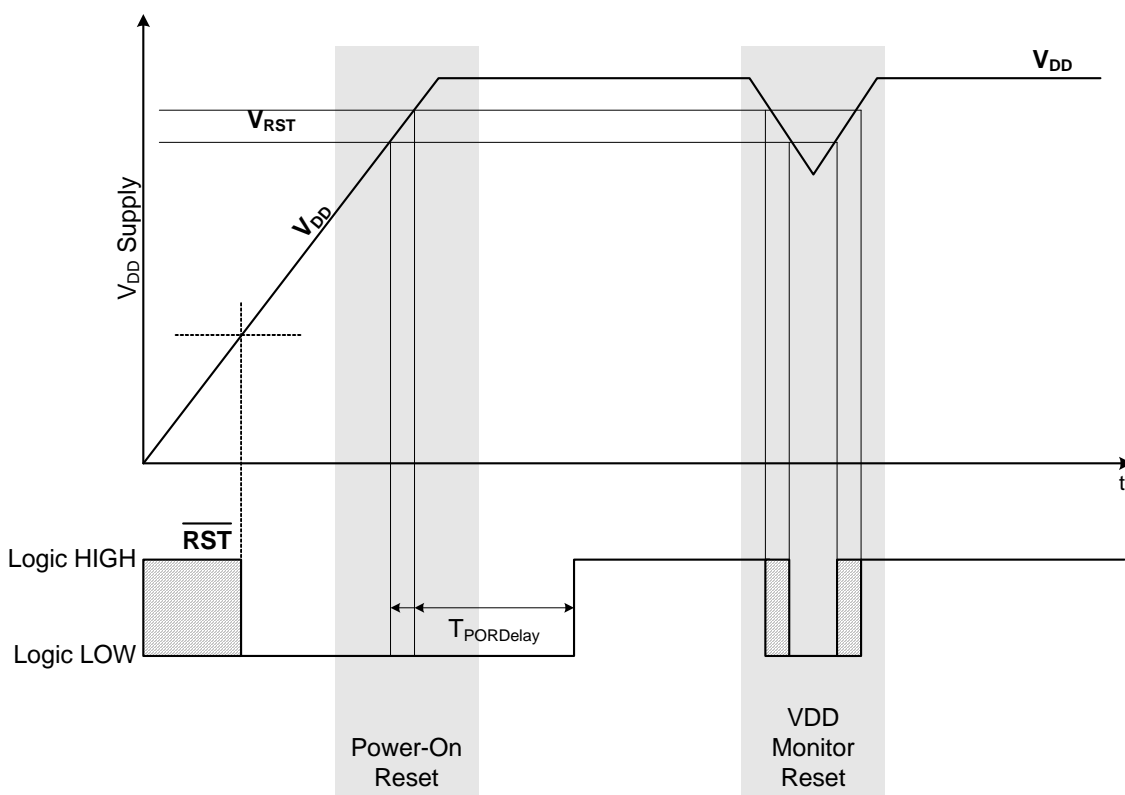


Figure 21.2. Power-On and V_{DD} Monitor Reset Timing

SFR Definition 22.3. OSCICN: Internal H-F Oscillator Control

Bit	7	6	5	4	3	2	1	0
Name	IOSCEN	IFRDY	SUSPEND	STSYNC	SSE		IFCN[1:0]	
Type	R/W	R	R/W	R	R/W	R	R/W	
Reset	1	1	0	0	0	0	0	0

SFR Address = 0xB2

Bit	Name	Function
7	IOSCEN	Internal H-F Oscillator Enable Bit. 0: Internal H-F Oscillator Disabled. 1: Internal H-F Oscillator Enabled.
6	IFRDY	Internal H-F Oscillator Frequency Ready Flag. 0: Internal H-F Oscillator is not running at programmed frequency. 1: Internal H-F Oscillator is running at programmed frequency.
5	SUSPEND	Internal Oscillator Suspend Enable Bit. Setting this bit to logic 1 places the internal oscillator in SUSPEND mode. The internal oscillator resumes operation when one of the SUSPEND mode awakening events occurs.
4	STSYNC	Suspend Timer Synchronization Bit. This bit is used to indicate when it is safe to read and write the registers associated with the suspend wake-up timer. If a suspend wake-up source other than Timer 2 has brought the oscillator out of suspend mode, it make take up to three timer clocks before the timer can be read or written. 0: Timer 2 registers can be read safely. 1: Timer 2 register reads and writes should not be performed.
3	SSE	Spread Spectrum Enable. Spread spectrum enable bit. 0: Spread Spectrum clock dithering disabled. 1: Spread Spectrum clock dithering enabled.
2	Unused	Read = 0b; Write = Don't Care
1:0	IFCN[1:0]	Internal H-F Oscillator Frequency Divider Control Bits. 00: SYSCLK derived from Internal H-F Oscillator divided by 8. 01: SYSCLK derived from Internal H-F Oscillator divided by 4. 10: SYSCLK derived from Internal H-F Oscillator divided by 2. 11: SYSCLK derived from Internal H-F Oscillator divided by 1.

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.

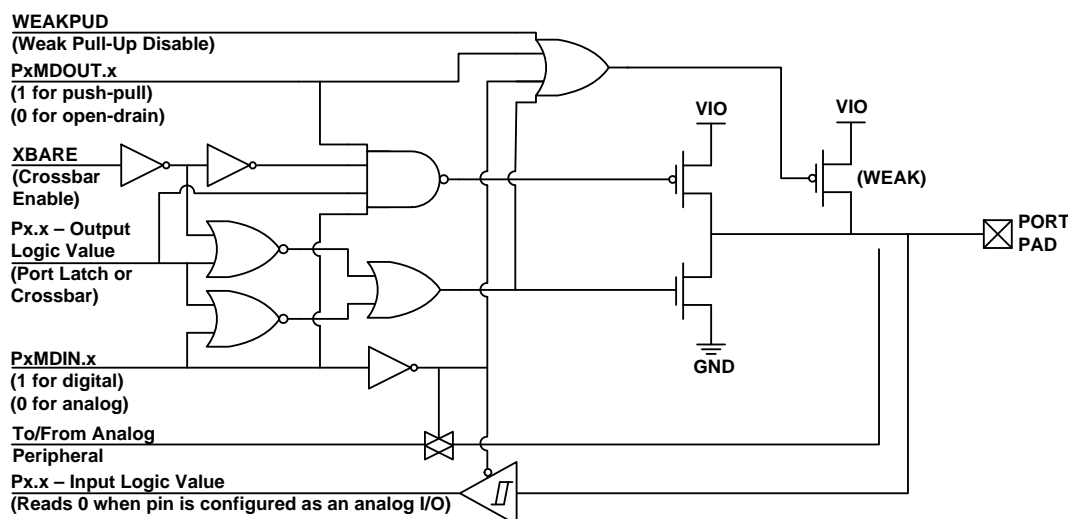


Figure 23.2. Port I/O Cell Block Diagram

C8051F80x-83x

SFR Definition 23.3. P0MASK: Port 0 Mask Register

Bit	7	6	5	4	3	2	1	0
Name	P0MASK[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xFE

Bit	Name	Function
7:0	P0MASK[7:0]	Port 0 Mask Value. Selects P0 pins to be compared to the corresponding bits in P0MAT. 0: P0.n pin logic value is ignored and cannot cause a Port Mismatch event. 1: P0.n pin logic value is compared to P0MAT.n.

SFR Definition 23.4. P0MAT: Port 0 Match Register

Bit	7	6	5	4	3	2	1	0
Name	P0MAT[7:0]							
Type	R/W							
Reset	1	1	1	1	1	1	1	1

SFR Address = 0xFD

Bit	Name	Function
7:0	P0MAT[7:0]	Port 0 Match Value. Match comparison value used on Port 0 for bits in P0MASK which are set to 1. 0: P0.n pin logic value is compared with logic LOW. 1: P0.n pin logic value is compared with logic HIGH.

SFR Definition 23.8. P0MDIN: Port 0 Input Mode

Bit	7	6	5	4	3	2	1	0
Name	P0MDIN[7:0]							
Type	R/W							
Reset	1	1	1	1	1	1	1	1

SFR Address = 0xF1

Bit	Name	Function
7:0	P0MDIN[7:0]	Analog Configuration Bits for P0.7–P0.0 (respectively). Port pins configured for analog mode have their weak pullup, digital driver, and digital receiver disabled. In order for the P0.n pin to be in analog mode, there MUST be a '1' in the Port Latch register corresponding to that pin. 0: Corresponding P0.n pin is configured for analog mode. 1: Corresponding P0.n pin is not configured for analog mode.

SFR Definition 23.9. P0MDOUT: Port 0 Output Mode

Bit	7	6	5	4	3	2	1	0
Name	P0MDOUT[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xA4

Bit	Name	Function
7:0	P0MDOUT[7:0]	Output Configuration Bits for P0.7–P0.0 (respectively). These bits are ignored if the corresponding bit in register P0MDIN is logic 0. 0: Corresponding P0.n Output is open-drain. 1: Corresponding P0.n Output is push-pull.

C8051F80x-83x

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

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).	0	0	X	1000
Master Receiver	1000	1	0	X	A master data byte was received; ACK requested.	Acknowledge received byte; Read SMB0DAT.	0	0	1	1000
						Send NACK to indicate last byte, and send STOP.	0	1	0	—
						Send NACK to indicate last byte, and send STOP followed by START.	1	1	0	1110
						Send ACK followed by repeated START.	1	0	1	1110
						Send NACK to indicate last byte, and send repeated START.	1	0	0	1110
						Send ACK and switch to Master Transmitter Mode (write to SMB0DAT before clearing SI).	0	0	1	1100
						Send NACK and switch to Master Transmitter Mode (write to SMB0DAT before clearing SI).	0	0	0	1100

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.

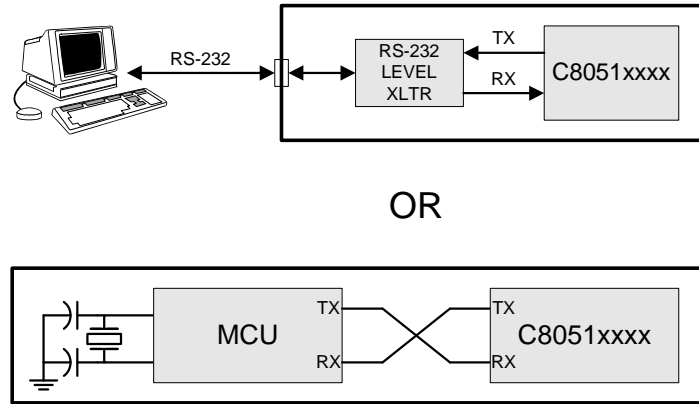


Figure 27.3. UART Interconnect Diagram

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.

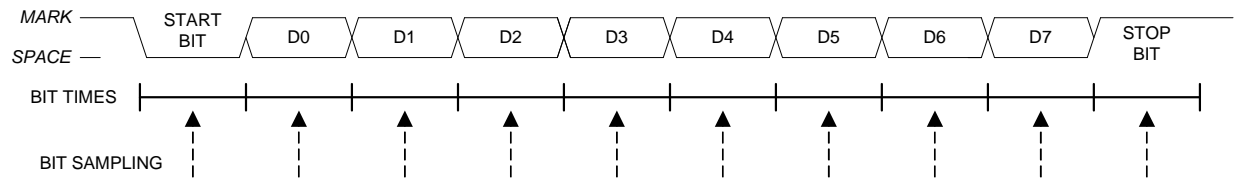


Figure 27.4. 8-Bit UART Timing Diagram

28.2. Timer 2

Timer 2 is a 16-bit timer formed by two 8-bit SFRs: TMR2L (low byte) and TMR2H (high byte). Timer 2 may operate in 16-bit auto-reload mode or (split) 8-bit auto-reload mode. The T2SPLIT bit (TMR2CN.3) defines the Timer 2 operation mode. Timer 2 can also be used in capture mode to capture rising edges of the Comparator 0 output.

Timer 2 may be clocked by the system clock, the system clock divided by 12, or the external oscillator source divided by 8. The external clock mode is ideal for real-time clock (RTC) functionality, where the internal oscillator drives the system clock while Timer 2 (and/or the PCA) is clocked by an external oscillator source. The external oscillator source divided by 8 is synchronized with the system clock when in all operating modes except suspend. When the internal oscillator is placed in suspend mode, the external clock/8 signal can directly drive the timer. This allows the use of an external clock to wake up the device from suspend mode. The timer will continue to run in suspend mode and count up. When the timer overflow occurs, the device will wake from suspend mode, and begin executing code again. The timer value may be set prior to entering suspend, to overflow in the desired amount of time (number of clocks) to wake the device. If a wake-up source other than the timer wakes the device from suspend mode, it may take up to three timer clocks before the timer registers can be read or written. During this time, the STSYNC bit in register OSCICN will be set to 1, to indicate that it is not safe to read or write the timer registers.

28.2.1. 16-bit Timer with Auto-Reload

When T2SPLIT (TMR2CN.3) is zero, Timer 2 operates as a 16-bit timer with auto-reload. Timer 2 can be clocked by SYSCLK, SYSCLK divided by 12, or the external oscillator clock source divided by 8. As the 16-bit timer register increments and overflows from 0xFFFF to 0x0000, the 16-bit value in the Timer 2 reload registers (TMR2RLH and TMR2RLL) is loaded into the Timer 2 register as shown in Figure 28.4, and the Timer 2 High Byte Overflow Flag (TMR2CN.7) is set. If Timer 2 interrupts are enabled (if IE.5 is set), an interrupt will be generated on each Timer 2 overflow. Additionally, if Timer 2 interrupts are enabled and the TF2LEN bit is set (TMR2CN.5), an interrupt will be generated each time the lower 8 bits (TMR2L) overflow from 0xFF to 0x00.

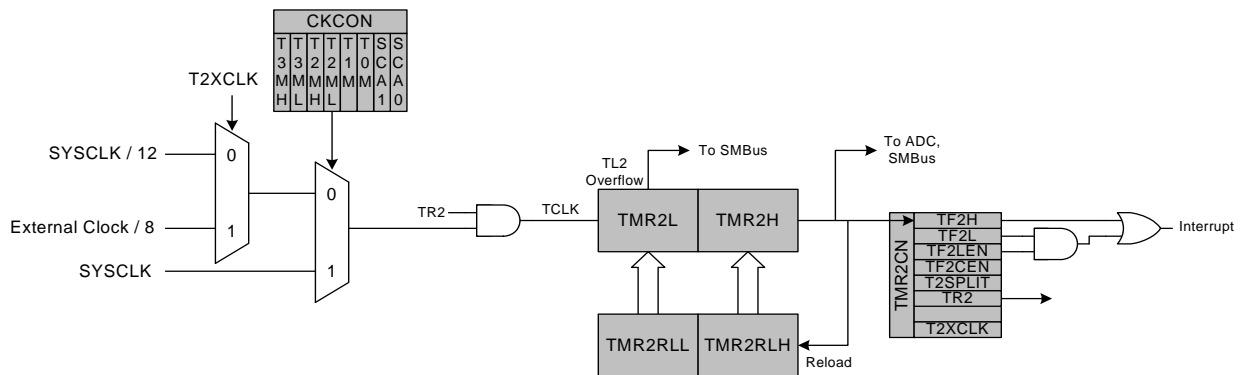


Figure 28.4. Timer 2 16-Bit Mode Block Diagram

29.3. Capture/Compare Modules

Each module can be configured to operate independently in one of six operation modes: edge-triggered capture, software timer, high-speed output, frequency output, 8-bit through 15-bit pulse width modulator, or 16-bit pulse width modulator. Each module has Special Function Registers (SFRs) associated with it in the CIP-51 system controller. These registers are used to exchange data with a module and configure the module's mode of operation. Table 29.2 summarizes the bit settings in the PCA0CPMn and PCA0PWM registers used to select the PCA capture/compare module's operating mode. Note that all modules set to use 8-bit through 15-bit PWM mode must use the same cycle length (8–15 bits). Setting the ECCFn bit in a PCA0CPMn register enables the module's CCFn interrupt.

Table 29.2. PCA0CPM and PCA0PWM Bit Settings for PCA Capture/Compare Modules^{1,2,3,4,5,6}

Operational Mode Bit Number	PCA0CPMn								PCA0PWM					
	7	6	5	4	3	2	1	0	7	6	5	4	3	2–0
Capture triggered by positive edge on CEXn	X	X	1	0	0	0	0	A	0	X	B	X	X	XXX
Capture triggered by negative edge on CEXn	X	X	0	1	0	0	0	A	0	X	B	X	X	XXX
Capture triggered by any transition on CEXn	X	X	1	1	0	0	0	A	0	X	B	X	X	XXX
Software Timer	X	C	0	0	1	0	0	A	0	X	B	X	X	XXX
High Speed Output	X	C	0	0	1	1	0	A	0	X	B	X	X	XXX
Frequency Output	X	C	0	0	0	1	1	A	0	X	B	X	X	XXX
8-Bit Pulse Width Modulator ⁷	0	C	0	0	E	0	1	A	0	X	B	X	X	000
9-Bit Pulse Width Modulator ⁷	0	C	0	0	E	0	1	A	D	X	B	X	X	001
10-Bit Pulse Width Modulator ⁷	0	C	0	0	E	0	1	A	D	X	B	X	X	010
11-Bit Pulse Width Modulator ⁷	0	C	0	0	E	0	1	A	D	X	B	X	X	011
12-Bit Pulse Width Modulator ⁷	0	C	0	0	E	0	1	A	D	X	B	X	X	100
13-Bit Pulse Width Modulator ⁷	0	C	0	0	E	0	1	A	D	X	B	X	X	101
14-Bit Pulse Width Modulator ⁷	0	C	0	0	E	0	1	A	D	X	B	X	X	110
15-Bit Pulse Width Modulator ⁷	0	C	0	0	E	0	1	A	D	X	B	X	X	111
16-Bit Pulse Width Modulator	1	C	0	0	E	0	1	A	0	X	B	X	0	XXX
16-Bit Pulse Width Modulator with Auto-Reload	1	C	0	0	E	0	1	A	D	X	B	X	1	XXX

Notes:

1. X = Don't Care (no functional difference for individual module if 1 or 0).
2. A = Enable interrupts for this module (PCA interrupt triggered on CCFn set to 1).
3. B = Enable 8th through 15th bit overflow interrupt (Depends on setting of CLSEL[2:0]).
4. C = When set to 0, the digital comparator is off. For high speed and frequency output modes, the associated pin will not toggle. In any of the PWM modes, this generates a 0% duty cycle (output = 0).
5. D = Selects whether the Capture/Compare register (0) or the Auto-Reload register (1) for the associated channel is accessed via addresses PCA0CPHn and PCA0CPLn.
6. E = When set, a match event will cause the CCFn flag for the associated channel to be set.
7. All modules set to 8-bit through 15-bit PWM mode use the same cycle length setting.

29.3.6. 16-Bit Pulse Width Modulator Mode

A PCA module may be operated in 16-Bit PWM mode. 16-bit PWM mode is independent of the other (8-bit through 15-bit) PWM modes. In this mode, the 16-bit capture/compare module defines the number of PCA clocks for the low time of the PWM signal. When the PCA counter matches the module contents, the output on CEXn is asserted high; when the 16-bit counter overflows, CEXn is asserted low. 16-Bit PWM Mode is enabled by setting the ECOMn, PWMn, and PWM16n bits in the PCA0CPMn register.

The duty cycle of the PWM output signal can be varied by writing to an “Auto-Reload” Register, which is dual-mapped into the PCA0CPHn and PCA0CPLn register locations. The auto-reload registers are accessed (read or written) when the bit ARSEL in PCA0PWM is set to 1. The capture/compare registers are accessed when ARSEL is set to 0. This synchronous update feature allows software to asynchronously write a new PWM high time, which will then take effect on the following PWM period.

For backwards-compatibility with the 16-bit PWM mode available on other devices, the PWM duty cycle can also be changed without using the “Auto-Reload” register. To output a varying duty cycle without using the “Auto-Reload” register, new value writes should be synchronized with PCA CCFn match interrupts. Match interrupts should be enabled ($ECCFn = 1$ AND $MATn = 1$) to help synchronize the capture/compare register writes. If the MATn bit is set to 1, the CCFn flag for the module will be set each time a 16-bit comparator match (rising edge) occurs. The CF flag in PCA0CN can be used to detect the overflow (falling edge). The duty cycle for 16-Bit PWM Mode is given by Equation 29.4.

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.

$$\text{Duty Cycle} = \frac{(65536 - PCA0CPn)}{65536}$$

Equation 29.4. 16-Bit PWM Duty Cycle

Using Equation 29.4, the largest duty cycle is 100% (PCA0CPn = 0), and the smallest duty cycle is 0.0015% (PCA0CPn = 0xFFFF). A 0% duty cycle may be generated by clearing the ECOMn bit to 0.

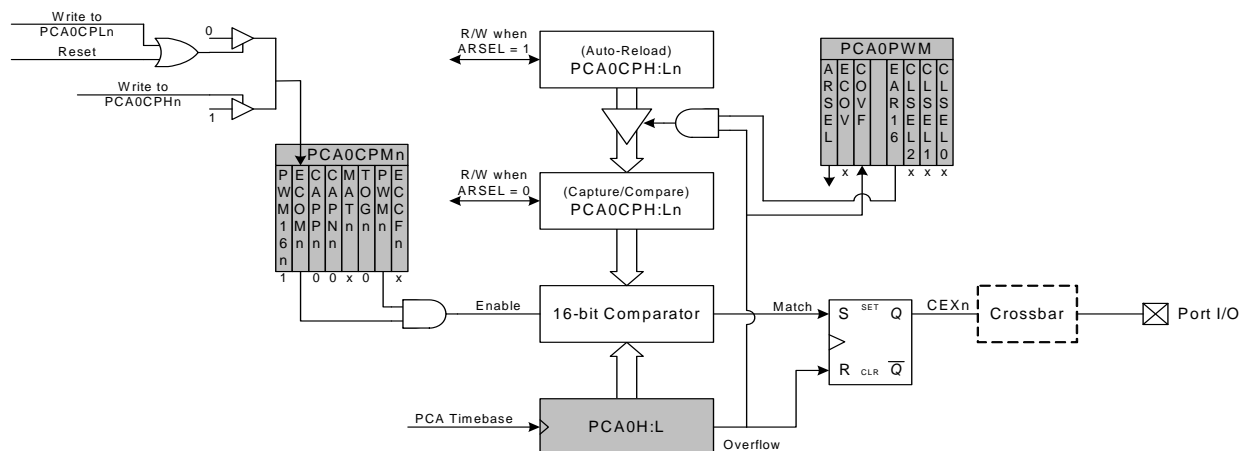


Figure 29.10. PCA 16-Bit PWM Mode

C8051F80x-83x

SFR Definition 29.1. PCA0CN: PCA0 Control

Bit	7	6	5	4	3	2	1	0
Name	CF	CR				CCF2	CCF1	CCF0
Type	R/W	R/W	R	R	R	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD8; Bit-Addressable

Bit	Name	Function
7	CF	PCA Counter/Timer Overflow Flag. Set by hardware when the PCA Counter/Timer overflows from 0xFFFF to 0x0000. When the Counter/Timer Overflow (CF) interrupt is enabled, setting this bit causes the CPU to vector to the PCA interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software.
6	CR	PCA Counter/Timer Run Control. This bit enables/disables the PCA Counter/Timer. 0: PCA Counter/Timer disabled. 1: PCA Counter/Timer enabled.
5:3	Unused	Read = 000b, Write = Don't care.
2	CCF2	PCA Module 2 Capture/Compare Flag. This bit is set by hardware when a match or capture occurs. When the CCF2 interrupt is enabled, setting this bit causes the CPU to vector to the PCA interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software.
1	CCF1	PCA Module 1 Capture/Compare Flag. This bit is set by hardware when a match or capture occurs. When the CCF1 interrupt is enabled, setting this bit causes the CPU to vector to the PCA interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software.
0	CCF0	PCA Module 0 Capture/Compare Flag. This bit is set by hardware when a match or capture occurs. When the CCF0 interrupt is enabled, setting this bit causes the CPU to vector to the PCA interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software.