

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/c8051f812-gm

C8051F80x-83x

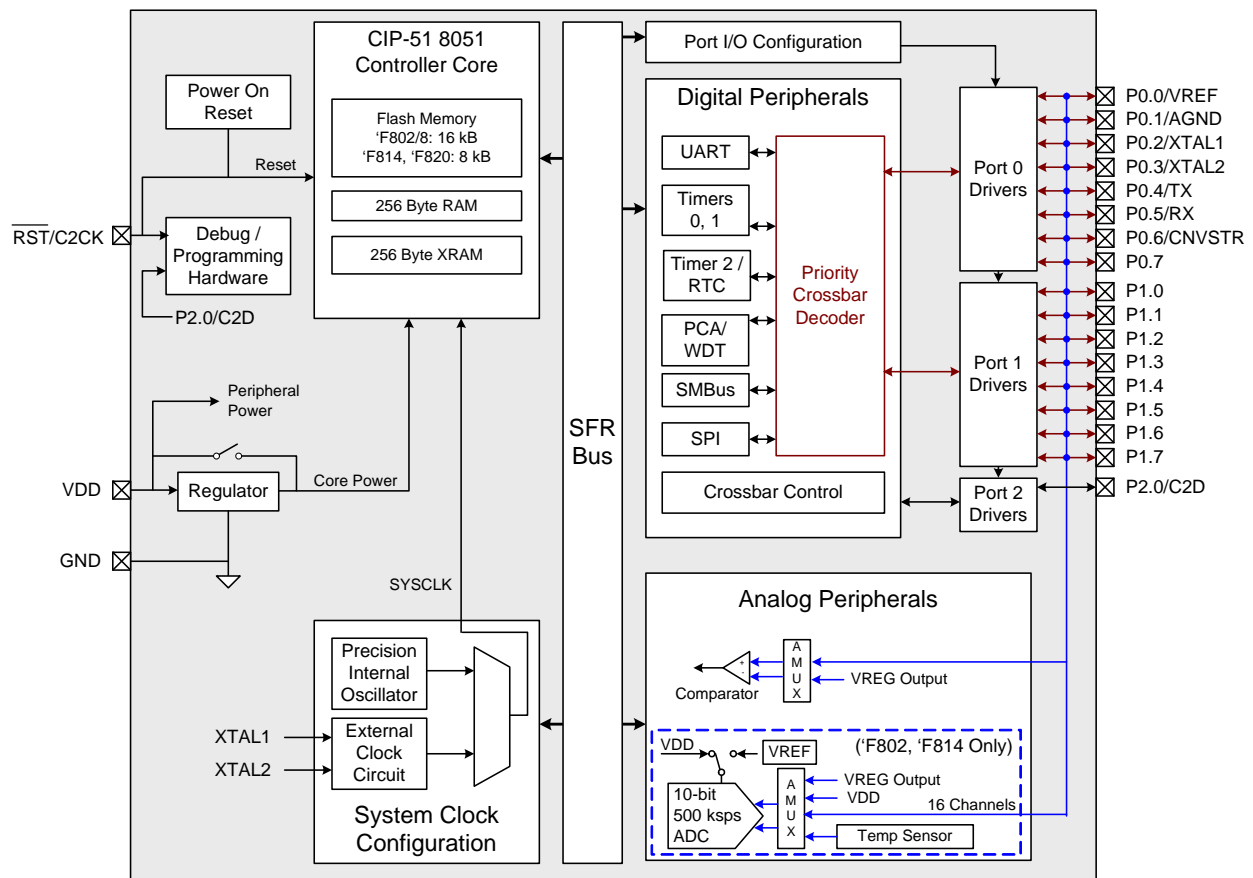


Figure 1.3. C8051F802, C8051F808, C8051F814, C8051F820 Block Diagram

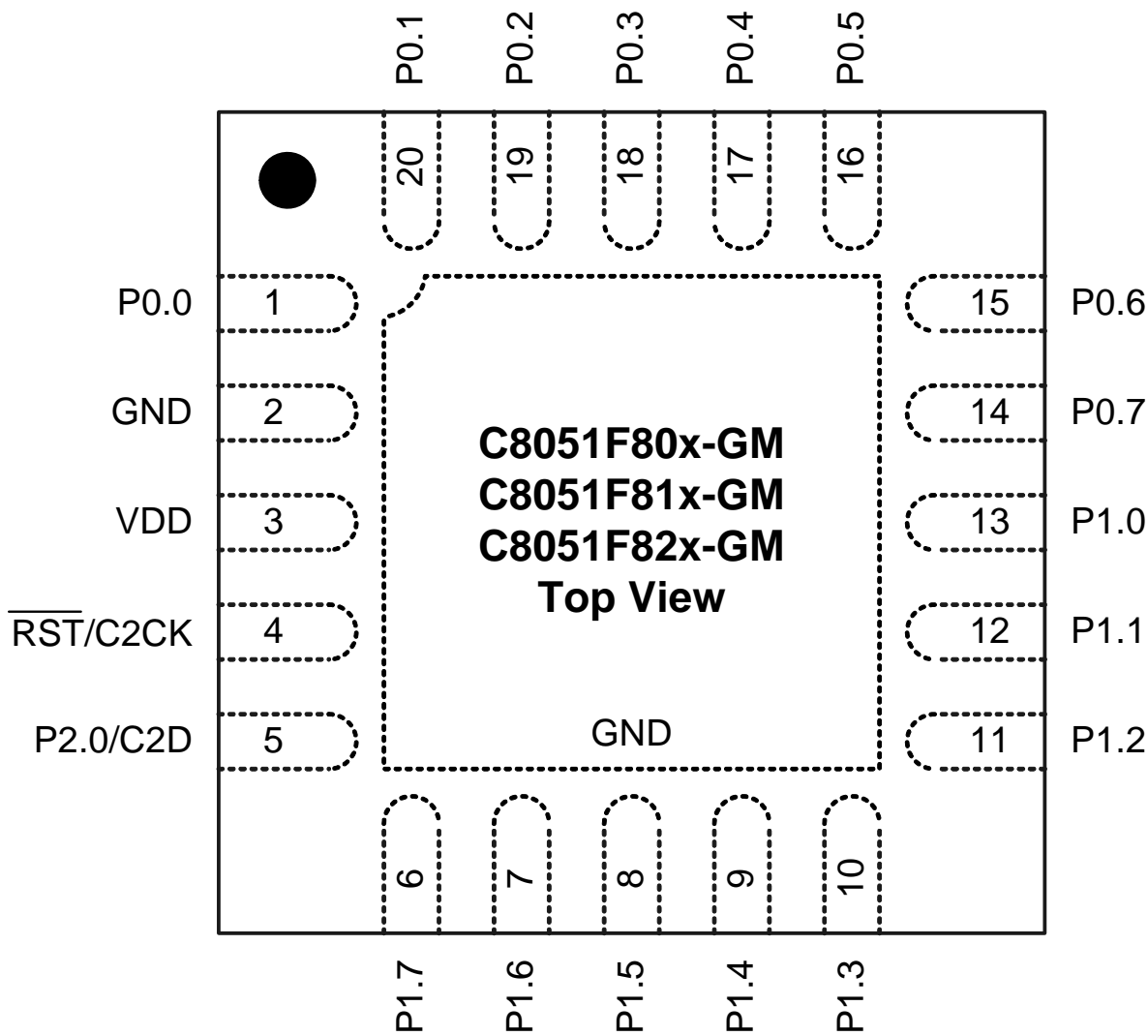


Figure 3.1. QFN-20 Pinout Diagram (Top View)

6. SOIC-16 Package Specifications

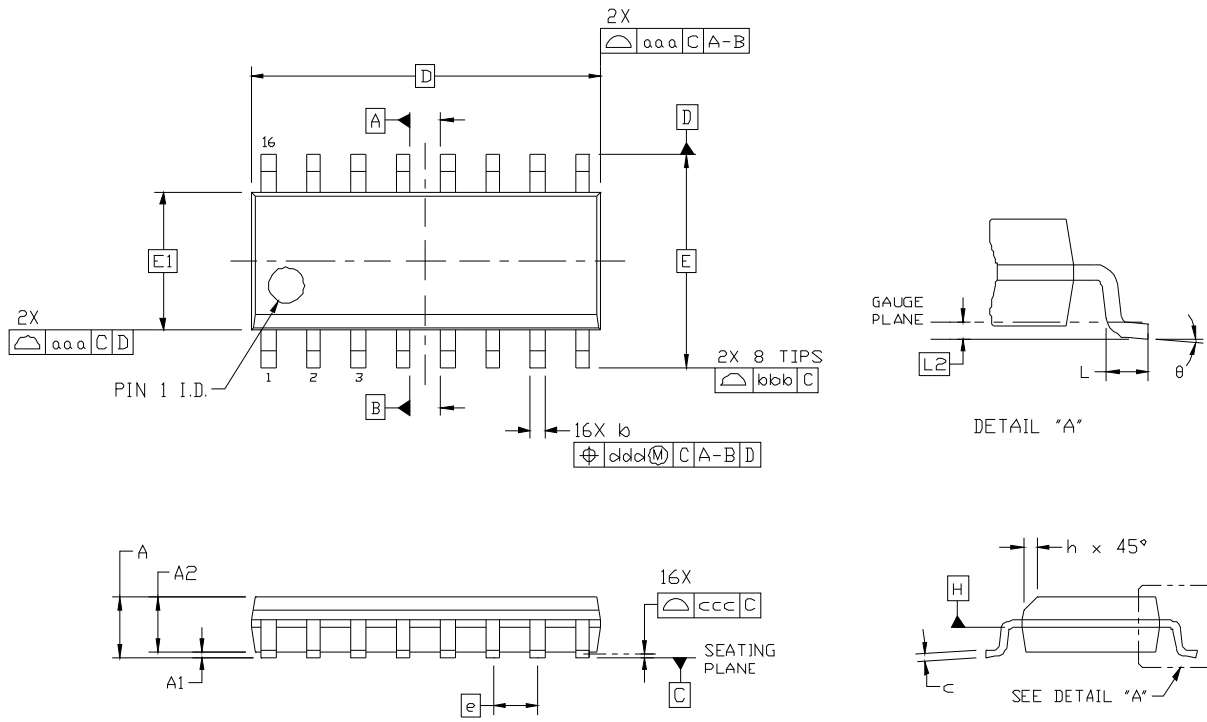


Figure 6.1. SOIC-16 Package Drawing

Table 6.1. SOIC-16 Package Dimensions

Dimension	Min	Nom	Max	Dimension	Min	Nom	Max
A	—		1.75	L	0.40		1.27
A1	0.10		0.25	L2	0.25 BSC		
A2	1.25		—	h	0.25		0.50
b	0.31		0.51	θ	0°		8°
c	0.17		0.25	aaa	0.10		
D	9.90 BSC			bbb	0.20		
E	6.00 BSC			ccc	0.10		
E1	3.90 BSC			ddd	0.25		
e	1.27 BSC						

Notes:

1. All dimensions shown are in millimeters (mm) unless otherwise noted.
2. Dimensioning and Tolerancing per ANSI Y14.5M-1994.
3. This drawing conforms to the JEDEC Solid State Outline MS-012, Variation AC.
4. Recommended card reflow profile is per the JEDEC/IPC J-STD-020 specification for Small Body Components.

8.3.3. Settling Time Requirements

A minimum tracking time is required before each conversion to ensure that an accurate conversion is performed. This tracking time is determined by any series impedance, including the AMUX0 resistance, the ADC0 sampling capacitance, and the accuracy required for the conversion. In delayed tracking mode, three SAR clocks are used for tracking at the start of every conversion. For many applications, these three SAR clocks will meet the minimum tracking time requirements.

Figure 8.3 shows the equivalent ADC0 input circuit. The required ADC0 settling time for a given settling accuracy (SA) may be approximated by Equation 8.1. See Table 7.9 for ADC0 minimum settling time requirements as well as the mux impedance and sampling capacitor values.

$$t = \ln\left(\frac{2^n}{SA}\right) \times R_{TOTAL} C_{SAMPLE}$$

Equation 8.1. ADC0 Settling Time Requirements

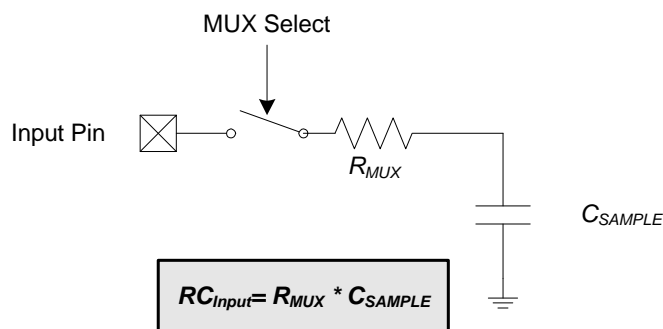
Where:

SA is the settling accuracy, given as a fraction of an LSB (for example, 0.25 to settle within 1/4 LSB)

t is the required settling time in seconds

R_{TOTAL} is the sum of the AMUX0 resistance and any external source resistance.

n is the ADC resolution in bits (10).



Note: See electrical specification tables for R_{MUX} and C_{SAMPLE} parameters.

Figure 8.3. ADC0 Equivalent Input Circuits

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.

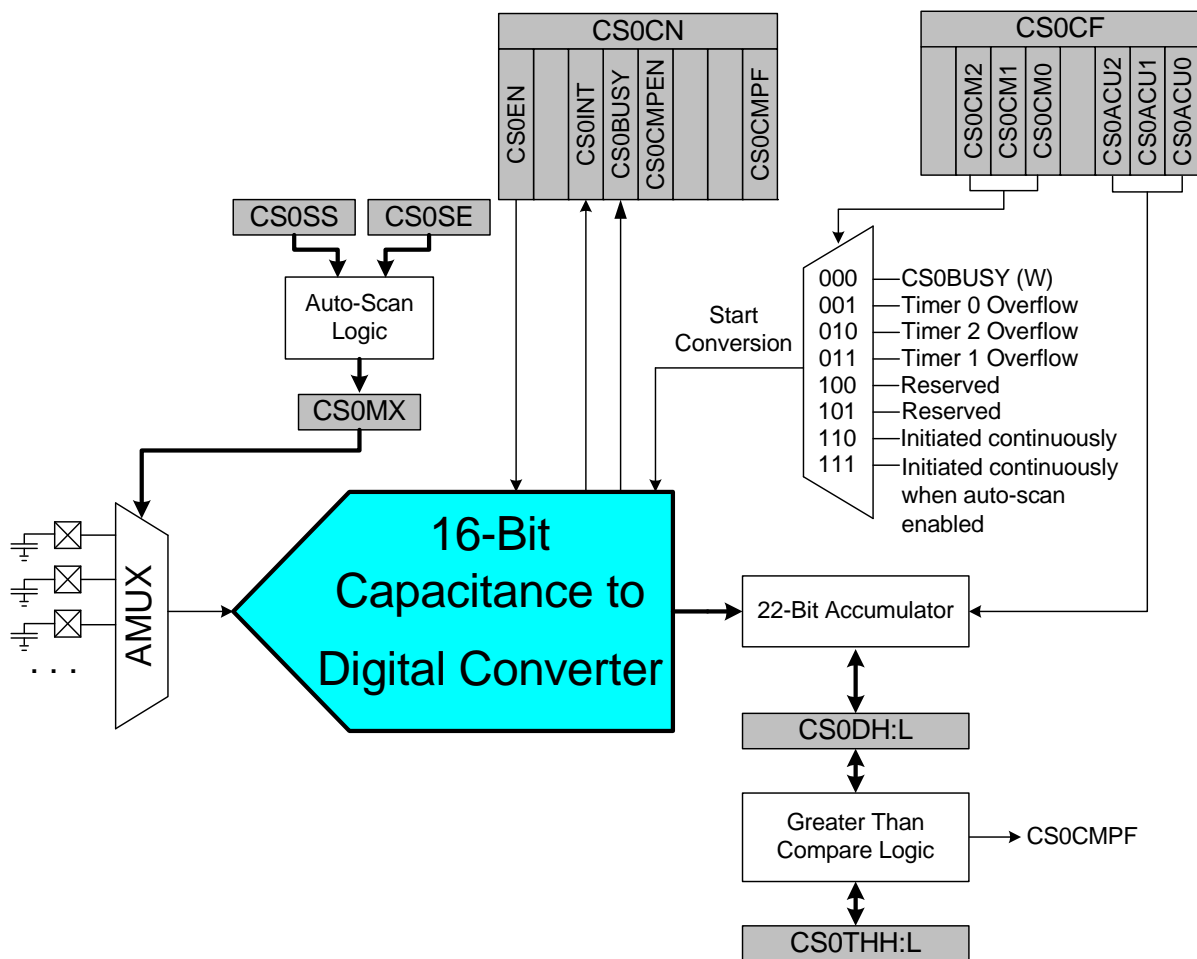


Figure 13.1. CS0 Block Diagram

Table 17.2. Special Function Registers (Continued)

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

Register	Address	Description	Page
P1MAT	0xED	P1 Match	152
P1MDIN	0xF2	Port 1 Input Mode Configuration	156
P1MDOUT	0xA5	Port 1 Output Mode Configuration	156
P1SKIP	0xD5	Port 1 Skip	157
P2	0xA0	Port 2 Latch	157
P2MDOUT	0xA6	Port 2 Output Mode Configuration	158
PCA0CN	0xD8	PCA Control	238
PCA0CPH0	0xFC	PCA Capture 0 High	243
PCA0CPH1	0xEA	PCA Capture 1 High	243
PCA0CPH2	0xEC	PCA Capture 2 High	243
PCA0CPL0	0xFB	PCA Capture 0 Low	243
PCA0CPL1	0xE9	PCA Capture 1 Low	243
PCA0CPL2	0xEB	PCA Capture 2 Low	243
PCA0CPM0	0xDA	PCA Module 0 Mode Register	241
PCA0CPM1	0xDB	PCA Module 1 Mode Register	241
PCA0CPM2	0xDC	PCA Module 2 Mode Register	241
PCA0H	0xFA	PCA Counter High	242
PCA0L	0xF9	PCA Counter Low	242
PCA0MD	0xD9	PCA Mode	239
PCA0PWM	0xF7	PCA PWM Configuration	240
PCON	0x87	Power Control	122
PSCTL	0x8F	Program Store R/W Control	118
PSW	0xD0	Program Status Word	91
REF0CN	0xD1	Voltage Reference Control	62
REG0CN	0xC9	Voltage Regulator Control	64
REVID	0xB6	Revision ID	96
RSTSRC	0xEF	Reset Source Configuration/Status	128

22.3.1. External Crystal Example

If a crystal or ceramic resonator is used as an external oscillator source for the MCU, the circuit should be configured as shown in Figure 22.1, Option 1. The External Oscillator Frequency Control value (XFCN) should be chosen from the Crystal column of the table in SFR Definition 22.4 (OSCXCN register). For example, an 11.0592 MHz crystal requires an XFCN setting of 111b and a 32.768 kHz Watch Crystal requires an XFCN setting of 001b. After an external 32.768 kHz oscillator is stabilized, the XFCN setting can be switched to 000 to save power. It is recommended to enable the missing clock detector before switching the system clock to any external oscillator source.

When the crystal oscillator is first enabled, the oscillator amplitude detection circuit requires a settling time to achieve proper bias. Introducing a delay of 1 ms between enabling the oscillator and checking the XTLVLD bit will prevent a premature switch to the external oscillator as the system clock. Switching to the external oscillator before the crystal oscillator has stabilized can result in unpredictable behavior. The recommended procedure is as follows:

1. Force XTAL1 and XTAL2 to a low state. This involves enabling the Crossbar and writing 0 to the port pins associated with XTAL1 and XTAL2.
2. Configure XTAL1 and XTAL2 as analog inputs.
3. Enable the external oscillator.
4. Wait at least 1 ms.
5. Poll for XTLVLD = 1.
6. If desired, enable the Missing Clock Detector.
7. Switch the system clock to the external oscillator.

Important Note on External Crystals: Crystal oscillator circuits are quite sensitive to PCB layout. The crystal should be placed as close as possible to the XTAL pins on the device. The traces should be as short as possible and shielded with ground plane from any other traces which could introduce noise or interference.

The capacitors shown in the external crystal configuration provide the load capacitance required by the crystal for correct oscillation. These capacitors are "in series" as seen by the crystal and "in parallel" with the stray capacitance of the XTAL1 and XTAL2 pins.

Note: The desired load capacitance depends upon the crystal and the manufacturer. Please refer to the crystal data sheet when completing these calculations.

For example, a tuning-fork crystal of 32.768 kHz with a recommended load capacitance of 12.5 pF should use the configuration shown in Figure 22.1, Option 1. The total value of the capacitors and the stray capacitance of the XTAL pins should equal 25 pF. With a stray capacitance of 3 pF per pin, the 22 pF capacitors yield an equivalent capacitance of 12.5 pF across the crystal, as shown in Figure 22.2.

22.3.3. External Capacitor Example

If a capacitor is used as an external oscillator for the MCU, the circuit should be configured as shown in Figure 22.1, Option 3. The capacitor should be no greater than 100 pF; however for very small capacitors, the total capacitance may be dominated by parasitic capacitance in the PCB layout. To determine the required External Oscillator Frequency Control value (XFCN) in the OSCXCN Register, select the capacitor to be used and find the frequency of oscillation according to Equation 22.2, where f = the frequency of oscillation in MHz, C = the capacitor value in pF, and V_{DD} = the MCU power supply in volts.

Equation 22.2. C Mode Oscillator Frequency

$$f = (KF) / (R \times V_{DD})$$

For example: Assume $V_{DD} = 3.0$ V and $f = 150$ kHz:

$$f = KF / (C \times V_{DD})$$

$$0.150 \text{ MHz} = KF / (C \times 3.0)$$

Since the frequency of roughly 150 kHz is desired, select the K Factor from the table in SFR Definition 22.4 (OSCXCN) as $KF = 22$:

$$0.150 \text{ MHz} = 22 / (C \times 3.0)$$

$$C \times 3.0 = 22 / 0.150 \text{ MHz}$$

$$C = 146.6 / 3.0 \text{ pF} = 48.8 \text{ pF}$$

Therefore, the XFCN value to use in this example is 011b and $C = 50$ pF.

C8051F80x-83x

Each Port has a corresponding PnSKIP register which allows its individual Port pins to be assigned to digital functions or skipped by the Crossbar. All Port pins used for analog functions or GPIO should have their PnSKIP bit set to 1.

The Port input mode of the I/O pins is defined using the Port Input Mode registers (PnMDIN). Each Port cell can be configured for analog or digital I/O. This selection is required even for the digital resources selected in the XBRn registers, and is not automatic. The only exception to this is P2.0, which can only be used for digital I/O.

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.

SFR Definition 23.7. P0: Port 0

Bit	7	6	5	4	3	2	1	0
Name	P0[7:0]							
Type	R/W							
Reset	1	1	1	1	1	1	1	1

SFR Address = 0x80; Bit-Addressable

Bit	Name	Description	Write	Read
7:0	P0[7:0]	Port 0 Data. Sets the Port latch logic value or reads the Port pin logic state in Port cells configured for digital I/O.	0: Set output latch to logic LOW. 1: Set output latch to logic HIGH.	0: P0.n Port pin is logic LOW. 1: P0.n Port pin is logic HIGH.

C8051F80x-83x

SFR Definition 23.14. P1SKIP: Port 1 Skip

Bit	7	6	5	4	3	2	1	0
Name	P1SKIP[7:0]							
Type	R/W							
Reset	0*	0*	0*	0*	0	0	0	0

SFR Address = 0xD5

Bit	Name	Function
7:0	P1SKIP[7:0]	Port 1 Crossbar Skip Enable Bits. These bits select Port 1 pins to be skipped by the Crossbar Decoder. Port pins used for analog, special functions or GPIO should be skipped by the Crossbar. 0: Corresponding P1.n pin is not skipped by the Crossbar. 1: Corresponding P1.n pin is skipped by the Crossbar. Note: P1.4–P1.7 are not available on 16-pin packages, with the reset value of 1111b for P1SKIP[7:4].

SFR Definition 23.15. P2: Port 2

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

SFR Address = 0xA0; Bit-Addressable

Bit	Name	Description	Write	Read
7:1	Unused	Unused.	Don't Care	0000000b
0	P2[0]	Port 2 Data. Sets the Port latch logic value or reads the Port pin logic state in Port cells configured for digital I/O.	0: Set output latch to logic LOW. 1: Set output latch to logic HIGH.	0: P2.0 Port pin is logic LOW. 1: P2.0 Port pin is logic HIGH.

Table 25.1. SPI Slave Timing Parameters

Parameter	Description	Min	Max	Units
Master Mode Timing (See Figure 25.8 and Figure 25.9)				
T_{MCKH}	SCK High Time	$1 \times T_{SYSCLK}$	—	ns
T_{MCKL}	SCK Low Time	$1 \times T_{SYSCLK}$	—	ns
T_{MIS}	MISO Valid to SCK Shift Edge	$1 \times T_{SYSCLK} + 20$	—	ns
T_{MIH}	SCK Shift Edge to MISO Change	0	—	ns
Slave Mode Timing (See Figure 25.10 and Figure 25.11)				
T_{SE}	NSS Falling to First SCK Edge	$2 \times T_{SYSCLK}$	—	ns
T_{SD}	Last SCK Edge to NSS Rising	$2 \times T_{SYSCLK}$	—	ns
T_{SEZ}	NSS Falling to MISO Valid	—	$4 \times T_{SYSCLK}$	ns
T_{SDZ}	NSS Rising to MISO High-Z	—	$4 \times T_{SYSCLK}$	ns
T_{CKH}	SCK High Time	$5 \times T_{SYSCLK}$	—	ns
T_{CKL}	SCK Low Time	$5 \times T_{SYSCLK}$	—	ns
T_{SIS}	MOSI Valid to SCK Sample Edge	$2 \times T_{SYSCLK}$	—	ns
T_{SIH}	SCK Sample Edge to MOSI Change	$2 \times T_{SYSCLK}$	—	ns
T_{SOH}	SCK Shift Edge to MISO Change	—	$4 \times T_{SYSCLK}$	ns
T_{SLH}	Last SCK Edge to MISO Change (CKPHA = 1 ONLY)	$6 \times T_{SYSCLK}$	$8 \times T_{SYSCLK}$	ns
Note: T_{SYSCLK} is equal to one period of the device system clock (SYSCLK).				

overflow after 25 ms (and SMBTOE set), the Timer 3 interrupt service routine can be used to reset (disable and re-enable) the SMBus in the event of an SCL low timeout.

26.3.5. SCL High (SMBus Free) Timeout

The SMBus specification stipulates that if the SCL and SDA lines remain high for more than 50 μ s, the bus is designated as free. When the SMBFTE bit in SMB0CF is set, the bus will be considered free if SCL and SDA remain high for more than 10 SMBus clock source periods (as defined by the timer configured for the SMBus clock source). If the SMBus is waiting to generate a Master START, the START will be generated following this timeout. A clock source is required for free timeout detection, even in a slave-only implementation.

26.4. Using the SMBus

The SMBus can operate in both Master and Slave modes. The interface provides timing and shifting control for serial transfers; higher level protocol is determined by user software. The SMBus interface provides the following application-independent features:

- Byte-wise serial data transfers
- Clock signal generation on SCL (Master Mode only) and SDA data synchronization
- Timeout/bus error recognition, as defined by the SMB0CF configuration register
- START/STOP timing, detection, and generation
- Bus arbitration
- Interrupt generation
- Status information
- Optional hardware recognition of slave address and automatic acknowledgement of address/data

SMBus interrupts are generated for each data byte or slave address that is transferred. When hardware acknowledgement is disabled, the point at which the interrupt is generated depends on whether the hardware is acting as a data transmitter or receiver. When a transmitter (i.e., sending address/data, receiving an ACK), this interrupt is generated after the ACK cycle so that software may read the received ACK value; when receiving data (i.e., receiving address/data, sending an ACK), this interrupt is generated before the ACK cycle so that software may define the outgoing ACK value. If hardware acknowledgement is enabled, these interrupts are always generated after the ACK cycle. See Section 26.5 for more details on transmission sequences.

Interrupts are also generated to indicate the beginning of a transfer when a master (START generated), or the end of a transfer when a slave (STOP detected). Software should read the SMB0CN (SMBus Control register) to find the cause of the SMBus interrupt. The SMB0CN register is described in Section 26.4.2; Table 26.5 provides a quick SMB0CN decoding reference.

26.4.1. SMBus Configuration Register

The SMBus Configuration register (SMB0CF) is used to enable the SMBus Master and/or Slave modes, select the SMBus clock source, and select the SMBus timing and timeout options. When the ENSMB bit is set, the SMBus is enabled for all master and slave events. Slave events may be disabled by setting the INH bit. With slave events inhibited, the SMBus interface will still monitor the SCL and SDA pins; however, the interface will NACK all received addresses and will not generate any slave interrupts. When the INH bit is set, all slave events will be inhibited following the next START (interrupts will continue for the duration of the current transfer).

imum setup and hold times for the two EXTHOLD settings. Setup and hold time extensions are typically necessary when SYSCLK is above 10 MHz.

Table 26.2. Minimum SDA Setup and Hold Times

EXTHOLD	Minimum SDA Setup Time	Minimum SDA Hold Time
0	$T_{low} - 4$ system clocks or 1 system clock + s/w delay *	3 system clocks
1	11 system clocks	12 system clocks
Note: Setup Time for ACK bit transmissions and the MSB of all data transfers. When using software acknowledgement, the s/w delay occurs between the time SMB0DAT or ACK is written and when SI is cleared. Note that if SI is cleared in the same write that defines the outgoing ACK value, s/w delay is zero.		

With the SMBTOE bit set, Timer 3 should be configured to overflow after 25 ms in order to detect SCL low timeouts (see Section “26.3.4. SCL Low Timeout” on page 182). The SMBus interface will force Timer 3 to reload while SCL is high, and allow Timer 3 to count when SCL is low. The Timer 3 interrupt service routine should be used to reset SMBus communication by disabling and re-enabling the SMBus.

SMBus Free Timeout detection can be enabled by setting the SMBFTE bit. When this bit is set, the bus will be considered free if SDA and SCL remain high for more than 10 SMBus clock source periods (see Figure 26.4).

SFR Definition 26.2. SMB0CN: SMBus Control

Bit	7	6	5	4	3	2	1	0
Name	MASTER	TXMODE	STA	STO	ACKRQ	ARBLOST	ACK	SI
Type	R	R	R/W	R/W	R	R	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xC0; Bit-Addressable

Bit	Name	Description	Read	Write
7	MASTER	SMBus Master/Slave Indicator. This read-only bit indicates when the SMBus is operating as a master.	0: SMBus operating in slave mode. 1: SMBus operating in master mode.	N/A
6	TXMODE	SMBus Transmit Mode Indicator. This read-only bit indicates when the SMBus is operating as a transmitter.	0: SMBus in Receiver Mode. 1: SMBus in Transmitter Mode.	N/A
5	STA	SMBus Start Flag.	0: No Start or repeated Start detected. 1: Start or repeated Start detected.	0: No Start generated. 1: When Configured as a Master, initiates a START or repeated START.
4	STO	SMBus Stop Flag.	0: No Stop condition detected. 1: Stop condition detected (if in Slave Mode) or pending (if in Master Mode).	0: No STOP condition is transmitted. 1: When configured as a Master, causes a STOP condition to be transmitted after the next ACK cycle. Cleared by Hardware.
3	ACKRQ	SMBus Acknowledge Request.	0: No Ack requested 1: ACK requested	N/A
2	ARBLOST	SMBus Arbitration Lost Indicator.	0: No arbitration error. 1: Arbitration Lost	N/A
1	ACK	SMBus Acknowledge.	0: NACK received. 1: ACK received.	0: Send NACK 1: Send ACK
0	SI	SMBus Interrupt Flag. This bit is set by hardware under the conditions listed in Table 15.3. SI must be cleared by software. While SI is set, SCL is held low and the SMBus is stalled.	0: No interrupt pending 1: Interrupt Pending	0: Clear interrupt, and initiate next state machine event. 1: Force interrupt.

C8051F80x-83x

SFR Definition 26.3. SMB0ADR: SMBus Slave Address

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

SFR Address = 0xD7

Bit	Name	Function
7:1	SLV[6:0]	SMBus Hardware Slave Address. Defines the SMBus Slave Address(es) for automatic hardware acknowledgement. Only address bits which have a 1 in the corresponding bit position in SLVM[6:0] are checked against the incoming address. This allows multiple addresses to be recognized.
0	GC	General Call Address Enable. When hardware address recognition is enabled (EHACK = 1), this bit will determine whether the General Call Address (0x00) is also recognized by hardware. 0: General Call Address is ignored. 1: General Call Address is recognized.

SFR Definition 26.4. SMB0ADM: SMBus Slave Address Mask

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

SFR Address = 0xD6

Bit	Name	Function
7:1	SLVM[6:0]	SMBus Slave Address Mask. Defines which bits of register SMB0ADR are compared with an incoming address byte, and which bits are ignored. Any bit set to 1 in SLVM[6:0] enables comparisons with the corresponding bit in SLV[6:0]. Bits set to 0 are ignored (can be either 0 or 1 in the incoming address).
0	EHACK	Hardware Acknowledge Enable. Enables hardware acknowledgement of slave address and received data bytes. 0: Firmware must manually acknowledge all incoming address and data bytes. 1: Automatic Slave Address Recognition and Hardware Acknowledge is Enabled.

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	—
	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

C8051F80x-83x

SFR Definition 28.1. CKCON: Clock Control

Bit	7	6	5	4	3	2	1	0
Name			T2MH	T2ML	T1M	T0M	SCA[1:0]	
Type	R	R	R/W	R/W	R/W	R/W	R/W	
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x8E

Bit	Name	Function
7:6	Unused	Read = 0b; Write = Don't care
5	T2MH	Timer 2 High Byte Clock Select. Selects the clock supplied to the Timer 2 high byte (split 8-bit timer mode only). 0: Timer 2 high byte uses the clock defined by the T2XCLK bit in TMR2CN. 1: Timer 2 high byte uses the system clock.
4	T2ML	Timer 2 Low Byte Clock Select. Selects the clock supplied to Timer 2. If Timer 2 is configured in split 8-bit timer mode, this bit selects the clock supplied to the lower 8-bit timer. 0: Timer 2 low byte uses the clock defined by the T2XCLK bit in TMR2CN. 1: Timer 2 low byte uses the system clock.
3	T1	Timer 1 Clock Select. Selects the clock source supplied to Timer 1. Ignored when C/T1 is set to 1. 0: Timer 1 uses the clock defined by the prescale bits SCA[1:0]. 1: Timer 1 uses the system clock.
2	T0	Timer 0 Clock Select. Selects the clock source supplied to Timer 0. Ignored when C/T0 is set to 1. 0: Counter/Timer 0 uses the clock defined by the prescale bits SCA[1:0]. 1: Counter/Timer 0 uses the system clock.
1:0	SCA[1:0]	Timer 0/1 Prescale Bits. These bits control the Timer 0/1 Clock Prescaler: 00: System clock divided by 12 01: System clock divided by 4 10: System clock divided by 48 11: External clock divided by 8 (synchronized with the system clock)

C8051F80x-83x

28.2.2. 8-bit Timers with Auto-Reload

When T2SPLIT is set, Timer 2 operates as two 8-bit timers (TMR2H and TMR2L). Both 8-bit timers operate in auto-reload mode as shown in Figure 28.5. TMR2RLL holds the reload value for TMR2L; TMR2RLH holds the reload value for TMR2H. The TR2 bit in TMR2CN handles the run control for TMR2H. TMR2L is always running when configured for 8-bit Mode. Timer 2 can also be used in capture mode to capture rising edges of the Comparator 0 output.

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

T2MH	T2XCLK	TMR2H Clock Source
0	0	SYSCLK / 12
0	1	External Clock / 8
1	X	SYSCLK

T2ML	T2XCLK	TMR2L Clock Source
0	0	SYSCLK / 12
0	1	External Clock / 8
1	X	SYSCLK

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

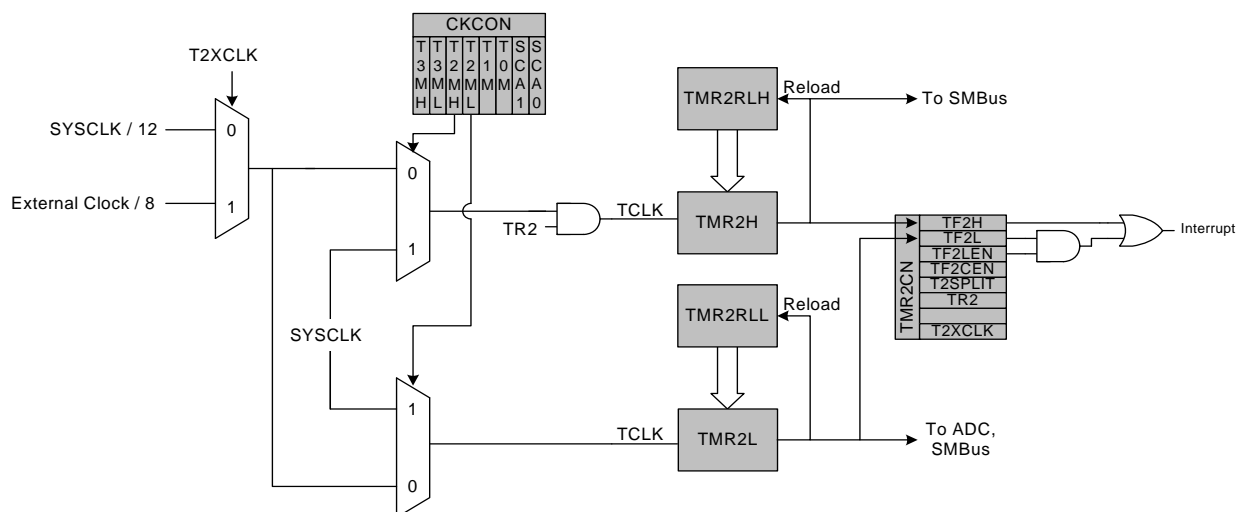


Figure 28.5. Timer 2 8-Bit Mode Block Diagram

set is then given (in PCA clocks) by Equation 29.5, where PCA0L is the value of the PCA0L register at the time of the update.

$$Offset = (256 \times PCA0CPL2) + (256 - PCA0L)$$

Equation 29.5. Watchdog Timer Offset in PCA Clocks

The WDT reset is generated when PCA0L overflows while there is a match between PCA0CPH2 and PCA0H. Software may force a WDT reset by writing a 1 to the CCF2 flag (PCA0CN.2) while the WDT is enabled.

29.4.2. Watchdog Timer Usage

To configure the WDT, perform the following tasks:

1. Disable the WDT by writing a 0 to the WDTE bit.
2. Select the desired PCA clock source (with the CPS2–CPS0 bits).
3. Load PCA0CPL2 with the desired WDT update offset value.
4. Configure the PCA Idle mode (set CIDL if the WDT should be suspended while the CPU is in Idle mode).
5. Enable the WDT by setting the WDTE bit to 1.
6. Reset the WDT timer by writing to PCA0CPH2.

The PCA clock source and Idle mode select cannot be changed while the WDT is enabled. The watchdog timer is enabled by setting the WDTE or WDLCK bits in the PCA0MD register. When WDLCK is set, the WDT cannot be disabled until the next system reset. If WDLCK is not set, the WDT is disabled by clearing the WDTE bit.

The WDT is enabled following any reset. The PCA0 counter clock defaults to the system clock divided by 12, PCA0L defaults to 0x00, and PCA0CPL2 defaults to 0x00. Using Equation 29.5, this results in a WDT timeout interval of 256 PCA clock cycles, or 3072 system clock cycles. Table 29.3 lists some example timeout intervals for typical system clocks.

Table 29.3. Watchdog Timer Timeout Intervals¹

System Clock (Hz)	PCA0CPL2	Timeout Interval (ms)
24,500,000	255	32.1
24,500,000	128	16.2
24,500,000	32	4.1
3,062,500 ²	255	257
3,062,500 ²	128	129.5
3,062,500 ²	32	33.1
32,000	255	24576
32,000	128	12384
32,000	32	3168
Notes: <ol style="list-style-type: none"> 1. Assumes SYSCLK/12 as the PCA clock source, and a PCA0L value of 0x00 at the update time. 2. Internal SYSCLK reset frequency = Internal Oscillator divided by 8. 		

29.5. Register Descriptions for PCA0

Following are detailed descriptions of the special function registers related to the operation of the PCA.

C8051F80x-83x

SFR Definition 29.3. PCA0PWM: PCA0 PWM Configuration

Bit	7	6	5	4	3	2	1	0
Name	ARSEL	ECOV	COVF		EAR16	CLSEL[1:0]		
Type	R/W	R/W	R/W	R	R/W	R/W		
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xF7

Bit	Name	Function			
7	ARSEL	Auto-Reload Register Select. This bit selects whether to read and write the normal PCA capture/compare registers (PCA0CPn), or the Auto-Reload registers at the same SFR addresses. This function is used to define the reload value for 9-bit through 15-bit PWM mode and 16-bit PWM mode. In all other modes, the Auto-Reload registers have no function. 0: Read/Write Capture/Compare Registers at PCA0CPHn and PCA0CPLn. 1: Read/Write Auto-Reload Registers at PCA0CPHn and PCA0CPLn.			
6	ECOV	Cycle Overflow Interrupt Enable. This bit sets the masking of the Cycle Overflow Flag (COVF) interrupt. 0: COVF will not generate PCA interrupts. 1: A PCA interrupt will be generated when COVF is set.			
5	COVF	Cycle Overflow Flag. This bit indicates an overflow of the nth bit (n= 9 through 15) of the main PCA counter (PCA0). The specific bit used for this flag depends on the setting of the CLSEL bits. The bit can be set by hardware or software, but must be cleared by software. 0: No overflow has occurred since the last time this bit was cleared. 1: An overflow has occurred since the last time this bit was cleared.			
4	Unused	Read = 0b; Write = Don't care.			
3	EAR16	16-Bit PWM Auto-Reload Enable. This bit controls the Auto-Reload feature in 16-bit PWM mode, which loads the PCA0CPn capture/compare registers with the values from the Auto-Reload registers at the same SFR addresses on an overflow of the PCA counter (PCA0). This setting affects all PCA channels that are configured to use 16-bit PWM mode. 0: 16-bit PWM mode Auto-Reload is disabled. This default setting is backwards-compatible with the 16-bit PWM mode available on other devices. 1: 16-bit PWM mode Auto-Reload is enabled.			
2:0	CLSEL[2:0]	Cycle Length Select. When 16-bit PWM mode is not selected, these bits select the length of the PWM cycle, from 8 to 15 bits. This affects all channels configured for PWM which are not using 16-bit PWM mode. These bits are ignored for individual channels configured to 16-bit PWM mode.			
		<table> <tr> <td>000: 8 bits. 001: 9 bits. 010: 10 bits.</td> <td>011: 11 bits. 100: 12 bits. 101: 13 bits.</td> <td>110: 14 bits. 111: 15 bits.</td> </tr> </table>	000: 8 bits. 001: 9 bits. 010: 10 bits.	011: 11 bits. 100: 12 bits. 101: 13 bits.	110: 14 bits. 111: 15 bits.
000: 8 bits. 001: 9 bits. 010: 10 bits.	011: 11 bits. 100: 12 bits. 101: 13 bits.	110: 14 bits. 111: 15 bits.			