E·XFL



Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "<u>Embedded -</u> <u>Microcontrollers</u>"

Details

Product Status	Last Time Buy
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	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	A/D 16x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	24-SSOP (0.154", 3.90mm Width)
Supplier Device Package	24-QSOP
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f800-gur

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong



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





6. SOIC-16 Package Specifications

Figure 6.1. SOIC-16 Package Drawing

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°
С	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	
е		1.27 BSC					

Table 6.1. SOIC-16 Package Dimensions

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.

 Recommended card reflow profile is per the JEDEC/IPC J-STD-020 specification for Small Body Components.



7.2. Electrical Characteristics

Table 7.2. Global Electrical Characteristics

-40 to +85 °C, 25 MHz system clock unless otherwise specified.

Parameter	Conditions	Min	Тур	Max	Units
Supply Voltage		1.8	3.0	3.6	V
Digital Supply Current with CPU Active (Normal Mode ¹)	$\begin{split} V_{DD} &= 1.8 \text{ V}, \text{ Clock} = 25 \text{ MHz} \\ V_{DD} &= 1.8 \text{ V}, \text{ Clock} = 1 \text{ MHz} \\ V_{DD} &= 1.8 \text{ V}, \text{ Clock} = 32 \text{ kHz} \\ V_{DD} &= 3.0 \text{ V}, \text{ Clock} = 25 \text{ MHz} \\ V_{DD} &= 3.0 \text{ V}, \text{ Clock} = 1 \text{ MHz} \\ V_{DD} &= 3.0 \text{ V}, \text{ Clock} = 32 \text{ kHz} \end{split}$		4.6 1.2 135 5.5 1.3 150	6.0 — 6.5 —	mA mA μA mA μA
Digital Supply Current with CPU Inactive (Idle Mode ¹)	$\begin{split} V_{DD} &= 1.8 \text{ V, Clock} = 25 \text{ MHz} \\ V_{DD} &= 1.8 \text{ V, Clock} = 1 \text{ MHz} \\ V_{DD} &= 1.8 \text{ V, Clock} = 32 \text{ kHz} \\ V_{DD} &= 3.0 \text{ V, Clock} = 25 \text{ MHz} \\ V_{DD} &= 3.0 \text{ V, Clock} = 1 \text{ MHz} \\ V_{DD} &= 3.0 \text{ V, Clock} = 32 \text{ kHz} \end{split}$		2 190 2.3 335 115	2.6 — 2.8 —	mA μA μA mA μA
Digital Supply Current (shutdown)	Oscillator not running (stop mode), Internal Regulator Off, 25 °C		0.5	2	μA
	Oscillator not running (stop or suspend mode), Internal Regulator On, 25 °C	_	105	140	μA
Digital Supply RAM Data Retention Voltage			1.3	_	V
Specified Operating Tempera- ture Range		-40	_	+85	°C
SYSCLK (system clock frequency)	See Note 2	0	_	25	MHz
Tsysl (SYSCLK low time)		18	_	—	ns
Tsysh (SYSCLK high time)		18	-	_	ns
Notes:					

1. Includes bias current for internal voltage regulator.

2. SYSCLK must be at least 32 kHz to enable debugging.



SFR Definition 8.9. ADC0MX: AMUX0 Channel Select

Bit	7	6	5	4	3	2	1	0
Name				AMX0P[3:0]				
Туре	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	Read = 000b; Write = Don't Care.			
4:0	AMX0P[4:0]	AMUX0 Positive In	AMUX0 Positive Input Selection.			
			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			



14. CIP-51 Microcontroller

The MCU system controller core is the CIP-51 microcontroller. The CIP-51 is fully compatible with the MCS-51[™] instruction set; standard 803x/805x assemblers and compilers can be used to develop software. The MCU family has a superset of all the peripherals included with a standard 8051. The CIP-51 also includes on-chip debug hardware (see description in Section 30), and interfaces directly with the analog and digital subsystems providing a complete data acquisition or control-system solution in a single integrated circuit.

The CIP-51 Microcontroller core implements the standard 8051 organization and peripherals as well as additional custom peripherals and functions to extend its capability (see Figure 14.1 for a block diagram). The CIP-51 includes the following features:

- Fully Compatible with MCS-51 Instruction Set
- 25 MIPS Peak Throughput with 25 MHz Clock
- 0 to 25 MHz Clock Frequency
- Extended Interrupt Handler

- Reset Input
- Power Management Modes
- On-chip Debug Logic
- Program and Data Memory Security

Performance

The CIP-51 employs a pipelined architecture that greatly increases its instruction throughput over the standard 8051 architecture. In a standard 8051, all instructions except for MUL and DIV take 12 or 24 system clock cycles to execute, and usually have a maximum system clock of 12 MHz. By contrast, the CIP-51 core executes 70% of its instructions in one or two system clock cycles, with no instructions taking more than eight system clock cycles.



Figure 14.1. CIP-51 Block Diagram



Mnemonic	Description	Bytes	Clock Cycles
XRL direct, #data	Exclusive-OR immediate to direct byte	3	3
CLR A	Clear A	1	1
CPL A	Complement A	1	1
RL A	Rotate A left	1	1
RLC A	Rotate A left through Carry	1	1
RR A	Rotate A right	1	1
RRC A	Rotate A right through Carry	1	1
SWAP A	Swap nibbles of A	1	1
Data Transfer			
MOV A, Rn	Move Register to A	1	1
MOV A, direct	Move direct byte to A	2	2
MOV A, @Ri	Move indirect RAM to A	1	2
MOV A, #data	Move immediate to A	2	2
MOV Rn, A	Move A to Register	1	1
MOV Rn, direct	Move direct byte to Register	2	2
MOV Rn, #data	Move immediate to Register	2	2
MOV direct, A	Move A to direct byte	2	2
MOV direct, Rn	Move Register to direct byte	2	2
MOV direct, direct	Move direct byte to direct byte	3	3
MOV direct, @Ri	Move indirect RAM to direct byte	2	2
MOV direct, #data	Move immediate to direct byte	3	3
MOV @Ri, A	Move A to indirect RAM	1	2
MOV @Ri, direct	Move direct byte to indirect RAM	2	2
MOV @Ri, #data	Move immediate to indirect RAM	2	2
MOV DPTR, #data16	Load DPTR with 16-bit constant	3	3
MOVC A, @A+DPTR	Move code byte relative DPTR to A	1	3
MOVC A, @A+PC	Move code byte relative PC to A	1	3
MOVX A, @Ri	Move external data (8-bit address) to A	1	3
MOVX @Ri, A	Move A to external data (8-bit address)	1	3
MOVX A, @DPTR	Move external data (16-bit address) to A	1	3
MOVX @DPTR, A	Move A to external data (16-bit address)	1	3
PUSH direct	Push direct byte onto stack	2	2
POP direct	Pop direct byte from stack	2	2
XCH A, Rn	Exchange Register with A	1	1
XCH A, direct	Exchange direct byte with A	2	2
XCH A, @Ri	Exchange indirect RAM with A	1	2
XCHD A, @Ri	Exchange low nibble of indirect RAM with A	1	2
Boolean Manipulation			
CLR C	Clear Carry	1	1
CLR bit	Clear direct bit	2	2
SETB C	Set Carry	1	1
SETB bit	Set direct bit	2	2
CPL C	Complement Carry	1	1
CPL bit	Complement direct bit	2	2

 Table 14.1. CIP-51 Instruction Set Summary (Continued)



Mnemonic	Description	Bytes	Clock Cycles
ANL C, bit	AND direct bit to Carry	2	2
ANL C, /bit	AND complement of direct bit to Carry	2	2
ORL C, bit	OR direct bit to carry	2	2
ORL C, /bit	OR complement of direct bit to Carry	2	2
MOV C, bit	Move direct bit to Carry	2	2
MOV bit, C	Move Carry to direct bit	2	2
JC rel	Jump if Carry is set	2	2/3
JNC rel	Jump if Carry is not set	2	2/3
JB bit, rel	Jump if direct bit is set	3	3/4
JNB bit, rel	Jump if direct bit is not set	3	3/4
JBC bit, rel	Jump if direct bit is set and clear bit	3	3/4
Program Branching	·		
ACALL addr11	Absolute subroutine call	2	3
LCALL addr16	Long subroutine call	3	4
RET	Return from subroutine	1	5
RETI	Return from interrupt	1	5
AJMP addr11	Absolute jump	2	3
LJMP addr16	Long jump	3	4
SJMP rel	Short jump (relative address)	2	3
JMP @A+DPTR	Jump indirect relative to DPTR	1	3
JZ rel	Jump if A equals zero	2	2/3
JNZ rel	Jump if A does not equal zero	2	2/3
CJNE A, direct, rel	Compare direct byte to A and jump if not equal	3	4/5
CJNE A, #data, rel	Compare immediate to A and jump if not equal	3	3/4
CJNE Rn, #data, rel	Compare immediate to Register and jump if not equal	3	3/4
CJNE @Ri, #data, rel	Compare immediate to indirect and jump if not equal	3	4/5
DJNZ Rn, rel	Decrement Register and jump if not zero	2	2/3
DJNZ direct, rel	Decrement direct byte and jump if not zero	3	3/4
NOP	No operation	1	1

Table 14.1.	CIP-51	Instruction Se	t Summar	v	(Continued)
	0		c Ouman	<i>,</i> ,	(0011111404)



18.3. INTO and INT1 External Interrupts

The INTO and INT1 external interrupt sources are configurable as active high or low, edge or level sensitive. The INOPL (INTO Polarity) and IN1PL (INT1 Polarity) bits in the IT01CF register select active high or active low; the IT0 and IT1 bits in TCON (Section "28.1. Timer 0 and Timer 1" on page 211) select level or edge sensitive. The table below lists the possible configurations.

IT0	IN0PL	INT0 Interrupt
1	0	Active low, edge sensitive
1	1	Active high, edge sensitive
0	0	Active low, level sensitive
0	1	Active high, level sensitive

IT1	IN1PL	INT1 Interrupt
1	0	Active low, edge sensitive
1	1	Active high, edge sensitive
0	0	Active low, level sensitive
0	1	Active high, level sensitive

INT0 and INT1 are assigned to Port pins as defined in the IT01CF register (see SFR Definition 18.7). Note that INT0 and INT0 Port pin assignments are independent of any Crossbar assignments. INT0 and INT1 will monitor their assigned Port pins without disturbing the peripheral that was assigned the Port pin via the Crossbar. To assign a Port pin only to INT0 and/or INT1, configure the Crossbar to skip the selected pin(s). This is accomplished by setting the associated bit in register XBR0 (see Section "23.3. Priority Crossbar Decoder" on page 143 for complete details on configuring the Crossbar).

IE0 (TCON.1) and IE1 (TCON.3) serve as the interrupt-pending flags for the INT0 and INT1 external interrupts, respectively. If an INT0 or INT1 external interrupt is configured as edge-sensitive, the corresponding interrupt-pending flag is automatically cleared by the hardware when the CPU vectors to the ISR. When configured as level sensitive, the interrupt-pending flag remains logic 1 while the input is active as defined by the corresponding polarity bit (IN0PL or IN1PL); the flag remains logic 0 while the input is inactive. The external interrupt source must hold the input active until the interrupt request is recognized. It must then deactivate the interrupt request before execution of the ISR completes or another interrupt request will be generated.



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	I [1:0]
Туре	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 inter- nal 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: SYSULK derived from Internal H-F Oscillator divided by 2.
		11. STSCER derived from memai H-F Oscillator divided by 1.



Digital Function	Potentially Assignable Port Pins	SFR(s) used for Assignment					
UART0, SPI0, SMBus, SYSCLK, PCA0 (CEX0-2 and ECI), T0, or T1.	Any Port pin available for assignment by the Crossbar. This includes P0.0 - P1.7 ² pins which have their PnSKIP bit set to 0. ¹	XBR0, XBR1					
Any pin used for GPIO	P0.0-P2.0 ²	PnSKIP					
 Notes: 1. The Crossbar will always assign UART0 pins to P0.4 and P0.5. 2. Port pins P1.4–P1.7 are not available on the 16-pin packages. 							

Table 23.2. Port I/O Assignment for Digital Functions

23.2.3. Assigning Port I/O Pins to External Digital Event Capture Functions

External digital event capture functions can be used to trigger an interrupt or wake the device from a low power mode when a transition occurs on a digital I/O pin. The digital event capture functions do not require dedicated pins and will function on both GPIO pins (PnSKIP = 1) and pins in use by the Crossbar (PnSKIP = 0). External digital event capture functions cannot be used on pins configured for analog I/O. Table 23.3 shows all available external digital event capture functions.

Table 23.3. Port I/O Assignment for External Digital Event Capture Functions

Digital Function	Potentially Assignable Port Pins	SFR(s) used for Assignment				
External Interrupt 0	P0.0–P0.7	IT01CF				
External Interrupt 1	P0.0–P0.7	IT01CF				
Port Match	P0.0–P1.7 [*]	POMASK, POMAT P1MASK, P1MAT				
Note: Port pins P1.4–P1.7 are not available on the 16-pin packages.						



SFR Definition 23.10. P0SKIP: Port 0 Skip

Bit	7	6	5	4	3	2	1	0	
Name	P0SKIP[7:0]								
Туре	R/W								
Reset	0	0	0	0	0	0	0	0	

SFR Address = 0xD4

Bit	Name	Function
7:0	P0SKIP[7:0]	Port 0 Crossbar Skip Enable Bits.
		 These bits select Port 0 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 P0.n pin is not skipped by the Crossbar. 1: Corresponding P0.n pin is skipped by the Crossbar.

SFR Definition 23.11. P1: Port 1

Bit	7	6	5	4	3	2	1	0	
Name	P1[7:0]								
Туре	R/W								
Reset	1	1	1	1	1	1	1	1	

SFR Address = 0x90; Bit-Addressable

Bit	Name	Description	Write	Read
7:0	P1[7:0]	Port 1 Data. Sets the Port latch logic value or reads the Port pin logic state in Port cells con- figured for digital I/O. Note: P1.4–P1.7 are not available on 16-pin packages.	0: Set output latch to logic LOW. 1: Set output latch to logic HIGH.	0: P1.n Port pin is logic LOW. 1: P1.n Port pin is logic HIGH.



SFR Definition 24.4. CRC0AUTO: CRC Automatic Control

Bit	7	6	5	4	3	2	1	0	
Name	AUTOEN	CRCCPT	Reserved	CRC0ST[4:0]					
Туре	R/W								
Reset	0	1	0	0	0	0	0	0	

SFR Address = 0xD2

Bit	Name	Function
7	AUTOEN	Automatic CRC Calculation Enable.
		When AUTOEN is set to 1, any write to CRC0CN will initiate an automatic CRC starting at Flash sector CRC0ST and continuing for CRC0CNT sectors.
6	CRCCPT	Automatic CRC Calculation Complete.
		Set to 0 when a CRC calculation is in progress. Code execution is stopped during a CRC calculation, therefore reads from firmware will always return 1.
5	Reserved	Must write 0.
4:0	CRC0ST[4:0]	Automatic CRC Calculation Starting Flash Sector.
		These bits specify the Flash sector to start the automatic CRC calculation. The starting address of the first Flash sector included in the automatic CRC calculation is CRC0ST x 512.

SFR Definition 24.5. CRC0CNT: CRC Automatic Flash Sector Count

Bit	7	6	5	4	3	2	1	0		
Name				CRC0CNT[5:0]						
Туре	R	R		R/W						
Reset	0	0	0	0	0	0	0	0		

SFR Address = 0xD3

Bit	Name	Function
7:6	Unused	Read = 00b; Write = Don't Care.
5:0	CRC0CNT[5:0]	Automatic CRC Calculation Flash Sector Count.
		These bits specify the number of Flash sectors to include when performing an automatic CRC calculation. The base address of the last flash sector included in the automatic CRC calculation is equal to (CRC0ST + CRC0CNT) x 512.



26. SMBus

The SMBus I/O interface is a two-wire, bi-directional serial bus. The SMBus is compliant with the System Management Bus Specification, version 1.1, and compatible with the I²C serial bus. Reads and writes to the interface by the system controller are byte oriented with the SMBus interface autonomously controlling the serial transfer of the data. Data can be transferred at up to 1/20th of the system clock as a master or slave (this can be faster than allowed by the SMBus specification, depending on the system clock used). A method of extending the clock-low duration is available to accommodate devices with different speed capabilities on the same bus.

The SMBus interface may operate as a master and/or slave, and may function on a bus with multiple masters. The SMBus provides control of SDA (serial data), SCL (serial clock) generation and synchronization, arbitration logic, and START/STOP control and generation. The SMBus peripheral can be fully driven by software (i.e., software accepts/rejects slave addresses, and generates ACKs), or hardware slave address recognition and automatic ACK generation can be enabled to minimize software overhead. A block diagram of the SMBus peripheral and the associated SFRs is shown in Figure 26.1.



Figure 26.1. SMBus Block Diagram



All transactions are initiated by a master, with one or more addressed slave devices as the target. The master generates the START condition and then transmits the slave address and direction bit. If the transaction is a WRITE operation from the master to the slave, the master transmits the data a byte at a time waiting for an ACK from the slave at the end of each byte. For READ operations, the slave transmits the data waiting for an ACK from the master at the end of each byte. At the end of the data transfer, the master generates a STOP condition to terminate the transaction and free the bus. Figure 26.3 illustrates a typical SMBus transaction.



Figure 26.3. SMBus Transaction

26.3.1. Transmitter Vs. Receiver

On the SMBus communications interface, a device is the "transmitter" when it is sending an address or data byte to another device on the bus. A device is a "receiver" when an address or data byte is being sent to it from another device on the bus. The transmitter controls the SDA line during the address or data byte. After each byte of address or data information is sent by the transmitter, the receiver sends an ACK or NACK bit during the ACK phase of the transfer, during which time the receiver controls the SDA line.

26.3.2. Arbitration

A master may start a transfer only if the bus is free. The bus is free after a STOP condition or after the SCL and SDA lines remain high for a specified time (see Section "26.3.5. SCL High (SMBus Free) Timeout" on page 183). In the event that two or more devices attempt to begin a transfer at the same time, an arbitration scheme is employed to force one master to give up the bus. The master devices continue transmitting until one attempts a HIGH while the other transmits a LOW. Since the bus is open-drain, the bus will be pulled LOW. The master attempting the HIGH will detect a LOW SDA and lose the arbitration. The winning master continues its transmission without interruption; the losing master becomes a slave and receives the rest of the transfer if addressed. This arbitration scheme is non-destructive: one device always wins, and no data is lost.

26.3.3. Clock Low Extension

SMBus provides a clock synchronization mechanism, similar to I2C, which allows devices with different speed capabilities to coexist on the bus. A clock-low extension is used during a transfer in order to allow slower slave devices to communicate with faster masters. The slave may temporarily hold the SCL line LOW to extend the clock low period, effectively decreasing the serial clock frequency.

26.3.4. SCL Low Timeout

If the SCL line is held low by a slave device on the bus, no further communication is possible. Furthermore, the master cannot force the SCL line high to correct the error condition. To solve this problem, the SMBus protocol specifies that devices participating in a transfer must detect any clock cycle held low longer than 25 ms as a "timeout" condition. Devices that have detected the timeout condition must reset the communication no later than 10 ms after detecting the timeout condition.

When the SMBTOE bit in SMB0CF is set, Timer 3 is used to detect SCL low timeouts. Timer 3 is forced to reload when SCL is high, and allowed to count when SCL is low. With Timer 3 enabled and configured to



SMBCS1	SMBCS0	SMBus Clock Source		
0	0	Timer 0 Overflow		
0	1	Timer 1 Overflow		
1 0		Timer 2 High Byte Overflow		
1 1		Timer 2 Low Byte Overflow		

Table 26.1. SMBus Clock Source Selection

The SMBCS1–0 bits select the SMBus clock source, which is used only when operating as a master or when the Free Timeout detection is enabled. When operating as a master, overflows from the selected source determine the absolute minimum SCL low and high times as defined in Equation 26.1. Note that the selected clock source may be shared by other peripherals so long as the timer is left running at all times. For example, Timer 1 overflows may generate the SMBus and UART baud rates simultaneously. Timer configuration is covered in Section "28. Timers" on page 209.

$$T_{HighMin} = T_{LowMin} = \frac{1}{f_{ClockSourceOverflow}}$$

Equation 26.1. Minimum SCL High and Low Times

The selected clock source should be configured to establish the minimum SCL High and Low times as per Equation 26.1. When the interface is operating as a master (and SCL is not driven or extended by any other devices on the bus), the typical SMBus bit rate is approximated by Equation 26.2.

$$BitRate = \frac{f_{ClockSourceOverflow}}{3}$$

Equation 26.2. Typical SMBus Bit Rate

Figure 26.4 shows the typical SCL generation described by Equation 26.2. Notice that T_{HIGH} is typically twice as large as T_{LOW} . The actual SCL output may vary due to other devices on the bus (SCL may be extended low by slower slave devices, or driven low by contending master devices). The bit rate when operating as a master will never exceed the limits defined by equation Equation 26.1.



Figure 26.4. Typical SMBus SCL Generation

Setting the EXTHOLD bit extends the minimum setup and hold times for the SDA line. The minimum SDA setup time defines the absolute minimum time that SDA is stable before SCL transitions from low-to-high. The minimum SDA hold time defines the absolute minimum time that the current SDA value remains stable after SCL transitions from high-to-low. EXTHOLD should be set so that the minimum setup and hold times meet the SMBus Specification requirements of 250 ns and 300 ns, respectively. Table 26.2 shows the minimum setup.



26.5.3. Write Sequence (Slave)

During a write sequence, an SMBus master writes data to a slave device. The slave in this transfer will be a receiver during the address byte, and a receiver during all data bytes. When slave events are enabled (INH = 0), the interface enters Slave Receiver Mode when a START followed by a slave address and direction bit (WRITE in this case) is received. If hardware ACK generation is disabled, upon entering Slave Receiver Mode, an interrupt is generated and the ACKRQ bit is set. The software must respond to the received slave address with an ACK, or ignore the received slave address with a NACK. If hardware ACK generation is enabled, the hardware will apply the ACK for a slave address which matches the criteria set up by SMB0ADR and SMB0ADM. The interrupt will occur after the ACK cycle.

If the received slave address is ignored (by software or hardware), slave interrupts will be inhibited until the next START is detected. If the received slave address is acknowledged, zero or more data bytes are received.

If hardware ACK generation is disabled, the ACKRQ is set to 1 and an interrupt is generated after each received byte. Software must write the ACK bit at that time to ACK or NACK the received byte.

With hardware ACK generation enabled, the SMBus hardware will automatically generate the ACK/NACK, and then post the interrupt. It is important to note that the appropriate ACK or NACK value should be set up by the software prior to receiving the byte when hardware ACK generation is enabled.

The interface exits Slave Receiver Mode after receiving a STOP. Note that the interface will switch to Slave Transmitter Mode if SMB0DAT is written while an active Slave Receiver. Figure 26.7 shows a typical slave write sequence. Two received data bytes are shown, though any number of bytes may be received. Notice that the "data byte transferred" interrupts occur at different places in the sequence, depending on whether hardware ACK generation is enabled. The interrupt occurs **before** the ACK with hardware ACK generation disabled, and **after** the ACK when hardware ACK generation is enabled.



Figure 26.7. Typical Slave Write Sequence



			Fre	quency: 24.5 M	IHz				
	Target Baud Rate (bps)	Baud Rate % Error	Oscillator Divide Factor	Timer Clock Source	SCA1–SCA0 (pre-scale select) ¹	T1M ¹	Timer 1 Reload Value (hex)		
	230400	-0.32%	106	SYSCLK	XX ²	1	0xCB		
E	115200	-0.32%	212	SYSCLK	XX	1	0x96		
ror Sc	57600	0.15%	426	SYSCLK	XX	1	0x2B		
Υ Ψ	28800	-0.32%	848	SYSCLK/4	01	0	0x96		
ц я	14400	0.15%	1704	SYSCLK/12	00	0	0xB9		
YS	9600	-0.32%	2544	SYSCLK/12	00	0	0x96		
– v	2400	-0.32%	10176	SYSCLK/48	10	0	0x96		
	1200	0.15%	20448	SYSCLK/48	10	0	0x2B		
Notes: 1. 2.	 Notes: 1. SCA1–SCA0 and T1M bit definitions can be found in Section 28.1. 2. X = Don't care. 								

Table 27.1. Timer Settings for Standard Baud RatesUsing The Internal 24.5 MHz Oscillator

Table 27.2. Timer Settings for Standard Baud RatesUsing an External 22.1184 MHz Oscillator

	Frequency: 22.1184 MHz								
	Target Baud Rate (bps)	Baud Rate % Error	Oscillator Divide Factor	Timer Clock Source	SCA1–SCA0 (pre-scale select) ¹	T1M ¹	Timer 1 Reload Value (hex)		
	230400	0.00%	96	SYSCLK	XXZ	1	0xD0		
εĸ	115200	0.00%	192	SYSCLK	XX	1	0xA0		
ror Dsc	57600	0.00%	384	SYSCLK	XX	1	0x40		
K f al C	28800	0.00%	768	SYSCLK / 12	00	0	0xE0		
CL	14400	0.00%	1536	SYSCLK / 12	00	0	0xC0		
ΥS xte	9600	0.00%	2304	SYSCLK / 12	00	0	0xA0		
ŚШ	2400	0.00%	9216	SYSCLK / 48	10	0	0xA0		
	1200	0.00%	18432	SYSCLK / 48	10	0	0x40		
۲. ۲	230400	0.00%	96	EXTCLK / 8	11	0	0xFA		
ror Sc	115200	0.00%	192	EXTCLK / 8	11	0	0xF4		
K f I O	57600	0.00%	384	EXTCLK / 8	11	0	0xE8		
Y SCLI Iterna	28800	0.00%	768	EXTCLK / 8	11	0	0xD0		
	14400	0.00%	1536	EXTCLK / 8	11	0	0xA0		
s =	9600	0.00%	2304	EXTCLK / 8	11	0	0x70		
Mateau		-							

Notes:

1. SCA1–SCA0 and T1M bit definitions can be found in Section 28.1.

2. X = Don't care.



29.4. Watchdog Timer Mode

A programmable watchdog timer (WDT) function is available through the PCA Module 2. The WDT is used to generate a reset if the time between writes to the WDT update register (PCA0CPH2) exceed a specified limit. The WDT can be configured and enabled/disabled as needed by software.

With the WDTE bit set in the PCA0MD register, Module 2 operates as a watchdog timer (WDT). The Module 2 high byte is compared to the PCA counter high byte; the Module 2 low byte holds the offset to be used when WDT updates are performed. The Watchdog Timer is enabled on reset. Writes to some PCA registers are restricted while the Watchdog Timer is enabled. The WDT will generate a reset shortly after code begins execution. To avoid this reset, the WDT should be explicitly disabled (and optionally re-configured and re-enabled if it is used in the system).

29.4.1. Watchdog Timer Operation

While the WDT is enabled:

- PCA counter is forced on.
- Writes to PCA0L and PCA0H are not allowed.
- PCA clock source bits (CPS2–CPS0) are frozen.
- PCA Idle control bit (CIDL) is frozen.
- Module 2 is forced into software timer mode.
- Writes to the Module 2 mode register (PCA0CPM2) are disabled.

While the WDT is enabled, writes to the CR bit will not change the PCA counter state; the counter will run until the WDT is disabled. The PCA counter run control bit (CR) will read zero if the WDT is enabled but user software has not enabled the PCA counter. If a match occurs between PCA0CPH2 and PCA0H while the WDT is enabled, a reset will be generated. To prevent a WDT reset, the WDT may be updated with a write of any value to PCA0CPH2. Upon a PCA0CPH2 write, PCA0H plus the offset held in PCA0CPL2 is loaded into PCA0CPH2 (See Figure 29.11).



Figure 29.11. PCA Module 2 with Watchdog Timer Enabled

The 8-bit offset held in PCA0CPH2 is compared to the upper byte of the 16-bit PCA counter. This offset value is the number of PCA0L overflows before a reset. Up to 256 PCA clocks may pass before the first PCA0L overflow occurs, depending on the value of the PCA0L when the update is performed. The total off-



SFR Definition 29.1. PCA0CN: PCA0 Control

Bit	7	6	5	4	3	2	1	0
Name	CF	CR				CCF2	CCF1	CCF0
Туре	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 rou- tine. 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 rou- tine. 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 rou- tine. This bit is not automatically cleared by hardware and must be cleared by software.



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]	
Туре	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 Ena	able.			
		This bit sets the masking of the	e Cycle Overflow Flag (COV	F) interrupt.		
		0: COVF will not generate PCA	A interrupts.			
5	COVE					
5	COVI	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 to16-bit PWM mode.				
		000: 8 bits. 001: 9 bits. 010: 10 bits.	011: 11 bits. 100: 12 bits. 101: 13 bits.	110: 14 bits. 111: 15 bits.		

