# 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/I²C), SPI, UART/USART
Peripherals	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/c8051f802-gur

Email: info@E-XFL.COM

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

## **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





Figure 1.1. C8051F800, C8051F806, C8051F812, C8051F818 Block Diagram





Figure 4.2. QFN-20 Recommended PCB Land Pattern

Dimension	Min	Мах		
C1	3.70			
C2	3.	70		
E	0.:	50		
X1	0.20	0.30		

Dimension	Min	Max
X2	2.15	2.25
Y1	0.90	1.00
Y2	2.15	2.25

#### Notes:

General

- 1. All dimensions shown are in millimeters (mm) unless otherwise noted.
- 2. Dimensioning and Tolerancing is per the ANSI Y14.5M-1994 specification.
- 3. This Land Pattern Design is based on the IPC-7351 guidelines.

#### Solder Mask Design

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

#### Stencil Design

- 5. A stainless steel, laser-cut and electro-polished stencil with trapezoidal walls should be used to assure good solder paste release.
- 6. The stencil thickness should be 0.125 mm (5 mils).
- 7. The ratio of stencil aperture to land pad size should be 1:1 for all perimeter pins.
- **8.** A 2x2 array of 0.95 mm openings on a 1.1 mm pitch should be used for the center pad to assure the proper paste volume.

#### Card Assembly

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



#### 8.3.2. Tracking Modes

The AD0TM bit in register ADC0CN enables "delayed conversions", and will delay the actual conversion start by three SAR clock cycles, during which time the ADC will continue to track the input. If AD0TM is left at logic 0, a conversion will begin immediately, without the extra tracking time. For internal start-of-conversion sources, the ADC will track anytime it is not performing a conversion. When the CNVSTR signal is used to initiate conversions, ADC0 will track either when AD0TM is logic 1, or when AD0TM is logic 0 and CNVSTR is held low. See Figure 8.2 for track and convert timing details. Delayed conversion mode is useful when AMUX settings are frequently changed, due to the settling time requirements described in Section "8.3.3. Settling Time Requirements" on page 49.





Figure 8.2. 10-Bit ADC Track and Conversion Example Timing





## 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.



## 20. Power Management Modes

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

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

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

### 20.1. Idle Mode

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

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

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

// in `C': PCON  = 0x01; PCON = PCON;	<pre>// set IDLE bit // followed by a 3-cycle dummy instruction</pre>
; in assembly: ORL PCON, #01h MOV PCON, PCON	; set IDLE bit ; followed by a 3-cycle dummy instruction

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



## 20.2. Stop Mode

Setting the Stop Mode Select bit (PCON.1) causes the controller core to enter Stop mode as soon as the instruction that sets the bit completes execution. In Stop mode the internal oscillator, CPU, and all digital peripherals are stopped; the state of the external oscillator circuit is not affected. Each analog peripheral (including the external oscillator circuit) may be shut down individually prior to entering Stop Mode. Stop mode can only be terminated by an internal or external reset. On reset, the device performs the normal reset sequence and begins program execution at address 0x0000.

If enabled, the Missing Clock Detector will cause an internal reset and thereby terminate the Stop mode. The Missing Clock Detector should be disabled if the CPU is to be put to in STOP mode for longer than the MCD timeout of 100  $\mu$ s.

## 20.3. Suspend Mode

Suspend mode allows a system running from the internal oscillator to go to a very low power state similar to Stop mode, but the processor can be awakened by certain events without requiring a reset of the device. Setting the SUSPEND bit (OSCICN.5) causes the hardware to halt the CPU and the high-frequency internal oscillator, and go into Suspend mode as soon as the instruction that sets the bit completes execution. All internal registers and memory maintain their original data. Most digital peripherals are not active in Suspend mode. The exception to this is the Port Match feature and Timer 3, when it is run from an external oscillator source.

The clock divider bits CLKDIV[2:0] in register CLKSEL must be set to "divide by 1" when entering suspend mode.

Suspend mode can be terminated by five types of events, a port match (described in Section "23.5. Port Match" on page 150), a Timer 2 overflow (described in Section "28.2. Timer 2" on page 219), a comparator low output (if enabled), a capacitive sense greater-than comparator event, or a device reset event. In order to run Timer 3 in suspend mode, the timer must be configured to clock from the external clock source. When suspend mode is terminated, the device will continue execution on the instruction following the one that set the SUSPEND bit. If the wake event (port match or Timer 2 overflow) was configured to generate an interrupt, the interrupt will be serviced upon waking the device. If suspend mode is terminated by an internal or external reset, the CIP-51 performs a normal reset sequence and begins program execution at address 0x0000.

**Note:** The device will still enter suspend mode if a wake source is "pending", and the device will not wake on such pending sources. It is important to ensure that the intended wake source will trigger after the device enters suspend mode. For example, if a CS0 conversion completes and the interrupt fires before the device is in suspend mode, that interrupt cannot trigger the wake event. Because port match events are level-sensitive, pre-existing port match events will trigger a wake, as long as the match condition is still present when the device enters suspend.



## 21. Reset Sources

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

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

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

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

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



Figure 21.1. Reset Sources



## 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	<b>I</b> [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.



## 22.3. External Oscillator Drive Circuit

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

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



## 23.1. Port I/O Modes of Operation

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

### 23.1.1. Port Pins Configured for Analog I/O

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

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

### 23.1.2. Port Pins Configured For Digital I/O

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

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

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







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

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

SFR Address = 0xE2

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



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 that 50  $\mu$ 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).



#### 26.4.4. Data Register

The SMBus Data register SMB0DAT holds a byte of serial data to be transmitted or one that has just been received. Software may safely read or write to the data register when the SI flag is set. Software should not attempt to access the SMB0DAT register when the SMBus is enabled and the SI flag is cleared to logic 0, as the interface may be in the process of shifting a byte of data into or out of the register.

Data in SMB0DAT is always shifted out MSB first. After a byte has been received, the first bit of received data is located at the MSB of SMB0DAT. While data is being shifted out, data on the bus is simultaneously being shifted in. SMB0DAT always contains the last data byte present on the bus. In the event of lost arbitration, the transition from master transmitter to slave receiver is made with the correct data or address in SMB0DAT.

### SFR Definition 26.5. SMB0DAT: SMBus Data

Bit	7	6	5	4	3	2	1	0
Name	SMB0DAT[7:0]							
Туре	R/W							
Reset	0	0	0	0	0	0	0	0

 SFR Address = 0xC2

 Bit
 Name
 Function

 7:0
 SMB0DAT[7:0]
 SMBus Data.

 The SMB0DAT register contains a byte of data to be transmitted on the SMBus serial interface or a byte that has just been received on the SMBus serial interface. The CPU can read from or write to this register whenever the SI serial interrupt flag (SMB0CN.0) is set to logic 1. The serial data in the register remains stable as long as the SI flag is set. When the SI flag is not set, the system may be in the process of shifting data in/out and the CPU should not attempt to access this register.



		Frequency: 24.5 MHz						
	Target Baud Rate (bps)	Baud Rate % Error	Oscillator Divide Factor	Timer Clock Source	SCA1–SCA0 (pre-scale select) <sup>1</sup>	T1M <sup>1</sup>	Timer 1 Reload Value (hex)	
	230400	-0.32%	106	SYSCLK	XX <sup>2</sup>	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.	<ul> <li>Notes:</li> <li>1. SCA1–SCA0 and T1M bit definitions can be found in Section 28.1.</li> <li>2. X = Don't care.</li> </ul>							

# 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

			Frequ	uency: 22.1184	MHz		
	Target Baud Rate (bps)	Baud Rate % Error	Oscillator Divide Factor	Timer Clock Source	SCA1–SCA0 (pre-scale select) <sup>1</sup>	T1M <sup>1</sup>	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
CL	28800	0.00%	768	EXTCLK / 8	11	0	0xD0
ΥS	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.





Figure 28.2. T0 Mode 2 Block Diagram

### 28.1.4. Mode 3: Two 8-bit Counter/Timers (Timer 0 Only)

In Mode 3, Timer 0 is configured as two separate 8-bit counter/timers held in TL0 and TH0. The counter/timer in TL0 is controlled using the Timer 0 control/status bits in TCON and TMOD: TR0, C/T0, GATE0 and TF0. TL0 can use either the system clock or an external input signal as its timebase. The TH0 register is restricted to a timer function sourced by the system clock or prescaled clock. TH0 is enabled using the Timer 1 run control bit TR1. TH0 sets the Timer 1 overflow flag TF1 on overflow and thus controls the Timer 1 interrupt.

Timer 1 is inactive in Mode 3. When Timer 0 is operating in Mode 3, Timer 1 can be operated in Modes 0, 1 or 2, but cannot be clocked by external signals nor set the TF1 flag and generate an interrupt. However, the Timer 1 overflow can be used to generate baud rates or overflow conditions for other peripherals. While Timer 0 is operating in Mode 3, Timer 1 run control is handled through its mode settings. To run Timer 1 while Timer 0 is in Mode 3, set the Timer 1 Mode as 0, 1, or 2. To disable Timer 1, configure it for Mode 3.



## SFR Definition 28.3. TMOD: Timer Mode

Bit	7	6	5	4	3	2	1	0
Name	GATE1	C/T1	T1M[1:0]		GATE0	C/T0	T0M[1:0]	
Туре	R/W	R/W	R/	W	R/W	R/W	R/	W
Reset	0	0	0	0	0	0	0	0

#### SFR Address = 0x89

Bit	Name	Function
7	GATE1	Timer 1 Gate Control.
		0: Timer 1 enabled when TR1 = 1 irrespective of $\overline{INT1}$ logic level. 1: Timer 1 enabled only when TR1 = 1 AND $\overline{INT1}$ is active as defined by bit IN1PL in register IT01CF (see SFR Definition 18.7).
6	C/T1	Counter/Timer 1 Select.
		<ul><li>0: Timer: Timer 1 incremented by clock defined by T1M bit in register CKCON.</li><li>1: Counter: Timer 1 incremented by high-to-low transitions on external pin (T1).</li></ul>
5:4	T1M[1:0]	Timer 1 Mode Select.
		These bits select the Timer 1 operation mode. 00: Mode 0, 13-bit Counter/Timer 01: Mode 1, 16-bit Counter/Timer 10: Mode 2, 8-bit Counter/Timer with Auto-Reload 11: Mode 3, Timer 1 Inactive
3	GATE0	Timer 0 Gate Control.
		0: Timer 0 enabled when TR0 = 1 irrespective of $\overline{INT0}$ logic level. 1: Timer 0 enabled only when TR0 = 1 AND $\overline{INT0}$ is active as defined by bit IN0PL in register IT01CF (see SFR Definition 18.7).
2	C/T0	Counter/Timer 0 Select.
		<ul><li>0: Timer: Timer 0 incremented by clock defined by T0M bit in register CKCON.</li><li>1: Counter: Timer 0 incremented by high-to-low transitions on external pin (T0).</li></ul>
1:0	T0M[1:0]	Timer 0 Mode Select.
		These bits select the Timer 0 operation mode. 00: Mode 0, 13-bit Counter/Timer 01: Mode 1, 16-bit Counter/Timer 10: Mode 2, 8-bit Counter/Timer with Auto-Reload 11: Mode 3, Two 8-bit Counter/Timers



## 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.



NOTES:

