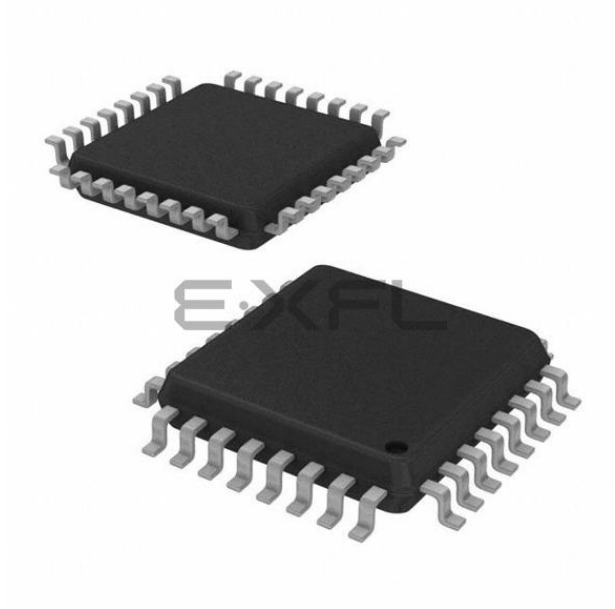

Welcome to E-XFL.COM

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 "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	8051
Core Size	8-Bit
Speed	48 MIPS
Connectivity	I²C, SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	25
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2.25K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.25V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	32-LQFP
Supplier Device Package	32-LQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f387-gqr

List of Registers

nitely, waiting for an external stimulus to wake up the system. Refer to Section “17.6. PCA Watchdog Timer Reset” on page 133 for more information on the use and configuration of the WDT.

10.2. 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.

By default, when in stop mode the internal regulator is still active. However, the regulator can be configured to shut down while in stop mode to save power. To shut down the regulator in stop mode, the STOPCF bit in register REG01CN should be set to 1 prior to setting the STOP bit (see SFR Definition 9.1). If the regulator is shut down using the STOPCF bit, only the $\overline{\text{RST}}$ pin or a full power cycle are capable of resetting the device.

10.3. Suspend Mode

Setting the SUSPEND bit (OSCICN.5) causes the hardware to halt 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. The CPU is not halted in Suspend, so code can still be executed using an oscillator other than the internal high-frequency oscillator.

Suspend mode can be terminated by resume signalling on the USB data pins, or a device reset event. When suspend mode is terminated, if the oscillator source is the internal high-frequency oscillator, the device will continue execution on the instruction following the one that set the SUSPEND bit. If the wake event 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.

SFR Definition 10.1. PCON: Power Control

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

SFR Address = 0x87; SFR Page = All Pages

Bit	Name	Function
7:2	GF[5:0]	**General Purpose Flags 5–0.** These are general purpose flags for use under software control.
1	STOP	**Stop Mode Select.** Setting this bit will place the CIP-51 in stop mode. This bit will always be read as 0. 1: CPU goes into stop mode (internal oscillator stopped).
0	IDLE	**IDLE: Idle Mode Select.** Setting this bit will place the CIP-51 in Idle mode. This bit will always be read as 0. 1: CPU goes into Idle mode. (Shuts off clock to CPU, but clock to Timers, Interrupts, Serial Ports, and Analog Peripherals are still active.)

SFR Definition 11.6. PSW: Program Status Word

Bit	7	6	5	4	3	2	1	0
Name	CY	AC	F0	RS[1:0]		OV	F1	PARITY
Type	R/W	R/W	R/W	R/W		R/W	R/W	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD0; SFR Page = All Pages; Bit-Addressable

Bit	Name	Function
7	CY	**Carry Flag.** This bit is set when the last arithmetic operation resulted in a carry (addition) or a borrow (subtraction). It is cleared to logic 0 by all other arithmetic operations.
6	AC	**Auxiliary Carry Flag.** This bit is set when the last arithmetic operation resulted in a carry into (addition) or a borrow from (subtraction) the high order nibble. It is cleared to logic 0 by all other arithmetic operations.
5	F0	**User Flag 0.** This is a bit-addressable, general purpose flag for use under software control.
4:3	RS[1:0]	**Register Bank Select.** These bits select which register bank is used during register accesses. 00: Bank 0, Addresses 0x00-0x07 01: Bank 1, Addresses 0x08-0x0F 10: Bank 2, Addresses 0x10-0x17 11: Bank 3, Addresses 0x18-0x1F
2	OV	**Overflow Flag.** This bit is set to 1 under the following circumstances: • An ADD, ADDC, or SUBB instruction causes a sign-change overflow. • A MUL instruction results in an overflow (result is greater than 255). • A DIV instruction causes a divide-by-zero condition. The OV bit is cleared to 0 by the ADD, ADDC, SUBB, MUL, and DIV instructions in all other cases.
1	F1	**User Flag 1.** This is a bit-addressable, general purpose flag for use under software control.
0	PARITY	**Parity Flag.** This bit is set to logic 1 if the sum of the eight bits in the accumulator is odd and cleared if the sum is even.

SFR Definition 14.1. EMI0CN: External Memory Interface Control

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

SFR Address = 0xAA; SFR Page = All Pages

Bit	Name	Function
7:0	PGSEL[7:0]	**XRAM Page Select Bits.** The XRAM Page Select Bits provide the high byte of the 16-bit external data memory address when using an 8-bit MOVX command, effectively selecting a 256-byte page of RAM. 0x00: 0x0000 to 0x00FF 0x01: 0x0100 to 0x01FF ... 0xFE: 0xFE00 to 0xFEFF 0xFF: 0xFF00 to 0xFFFF

SFR Definition 14.3. EMI0TC: External Memory TIming Control

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

SFR Address = 0x84; SFR Page = All Pages

Bit	Name	Function
7:6	EAS[1:0]	**EMIF Address Setup Time Bits.** 00: Address setup time = 0 SYSCLK cycles. 01: Address setup time = 1 SYSCLK cycle. 10: Address setup time = 2 SYSCLK cycles. 11: Address setup time = 3 SYSCLK cycles.
5:2	EWR[3:0]	**EMIF $\overline{WR}$ and $\overline{RD}$ Pulse-Width Control Bits.** 0000: $\overline{WR}$ and $\overline{RD}$ pulse width = 1 SYSCLK cycle. 0001: $\overline{WR}$ and $\overline{RD}$ pulse width = 2 SYSCLK cycles. 0010: $\overline{WR}$ and $\overline{RD}$ pulse width = 3 SYSCLK cycles. 0011: $\overline{WR}$ and $\overline{RD}$ pulse width = 4 SYSCLK cycles. 0100: $\overline{WR}$ and $\overline{RD}$ pulse width = 5 SYSCLK cycles. 0101: $\overline{WR}$ and $\overline{RD}$ pulse width = 6 SYSCLK cycles. 0110: $\overline{WR}$ and $\overline{RD}$ pulse width = 7 SYSCLK cycles. 0111: $\overline{WR}$ and $\overline{RD}$ pulse width = 8 SYSCLK cycles. 1000: $\overline{WR}$ and $\overline{RD}$ pulse width = 9 SYSCLK cycles. 1001: $\overline{WR}$ and $\overline{RD}$ pulse width = 10 SYSCLK cycles. 1010: $\overline{WR}$ and $\overline{RD}$ pulse width = 11 SYSCLK cycles. 1011: $\overline{WR}$ and $\overline{RD}$ pulse width = 12 SYSCLK cycles. 1100: $\overline{WR}$ and $\overline{RD}$ pulse width = 13 SYSCLK cycles. 1101: $\overline{WR}$ and $\overline{RD}$ pulse width = 14 SYSCLK cycles. 1110: $\overline{WR}$ and $\overline{RD}$ pulse width = 15 SYSCLK cycles. 1111: $\overline{WR}$ and $\overline{RD}$ pulse width = 16 SYSCLK cycles.
1:0	EAH[1:0]	**EMIF Address Hold Time Bits.** 00: Address hold time = 0 SYSCLK cycles. 01: Address hold time = 1 SYSCLK cycle. 10: Address hold time = 2 SYSCLK cycles. 11: Address hold time = 3 SYSCLK cycles.

Table 15.2. Special Function Registers

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

Register	Address	Page	Description	Page
ACC	0xE0	All Pages	Accumulator	86
ADC0CF	0xBC	All Pages	ADC0 Configuration	53
ADC0CN	0xE8	All Pages	ADC0 Control	55
ADC0GTH	0xC4	All Pages	ADC0 Greater-Than Compare High	56
ADC0GTL	0xC3	All Pages	ADC0 Greater-Than Compare Low	56
ADC0H	0xBE	All Pages	ADC0 High	54
ADC0L	0xBD	All Pages	ADC0 Low	54
ADC0LTH	0xC6	All Pages	ADC0 Less-Than Compare Word High	57
ADC0LTL	0xC5	All Pages	ADC0 Less-Than Compare Word Low	57
AMX0N	0xBA	All Pages	AMUX0 Negative Channel Select	61
AMX0P	0xBB	All Pages	AMUX0 Positive Channel Select	60
B	0xF0	All Pages	B Register	86
CKCON	0x8E	All Pages	Clock Control	264
CKCON1	0xE4	F	Clock Control 1	265
CLKMUL	0xB9	0	Clock Multiplier	147
CLKSEL	0xA9	All Pages	Clock Select	144
CPT0CN	0x9B	All Pages	Comparator0 Control	67
CPT0MD	0x9D	All Pages	Comparator0 Mode Selection	68
CPT0MX	0x9F	All Pages	Comparator0 MUX Selection	72
CPT1CN	0x9A	All Pages	Comparator1 Control	69
CPT1MD	0x9C	All Pages	Comparator1 Mode Selection	70
CPT1MX	0x9E	All Pages	Comparator1 MUX Selection	73
DPH	0x83	All Pages	Data Pointer High	85
DPL	0x82	All Pages	Data Pointer Low	85
EIE1	0xE6	All Pages	Extended Interrupt Enable 1	123
EIE2	0xE7	All Pages	Extended Interrupt Enable 2	125
EIP1	0xF6	All Pages	Extended Interrupt Priority 1	124
EIP2	0xF7	All Pages	Extended Interrupt Priority 2	126
EMI0CF	0x85	All Pages	External Memory Interface Configuration	97
EMI0CN	0xAA	All Pages	External Memory Interface Control	96
EMI0TC	0x84	All Pages	External Memory Interface Timing	103
FLKEY	0xB7	All Pages	Flash Lock and Key	140
FLSCL	0xB6	All Pages	Flash Scale	141

Accessing FLASH **from the C2 debug interface**:

1. Any unlocked page may be read, written, or erased.
2. Locked pages cannot be read, written, or erased.
3. The page containing the Lock Byte may be read, written, or erased if it is unlocked.
4. Reading the contents of the Lock Byte is always permitted.
5. Locking additional pages (changing 1s to 0s in the Lock Byte) is not permitted.
6. Unlocking FLASH pages (changing 0s to 1s in the Lock Byte) requires the C2 Device Erase command, which erases all FLASH pages including the page containing the Lock Byte and the Lock Byte itself.
7. The Reserved Area cannot be read, written, or erased.

Accessing FLASH **from user firmware** executing on an **unlocked page**:

1. Any unlocked page except the page containing the Lock Byte may be read, written, or erased.
2. Locked pages cannot be read, written, or erased.
3. The page containing the Lock Byte cannot be erased. It may be read or written only if it is unlocked.
4. Reading the contents of the Lock Byte is always permitted.
5. Locking additional pages (changing 1s to 0s in the Lock Byte) is not permitted.
6. Unlocking FLASH pages (changing 0s to 1s in the Lock Byte) is not permitted.
7. The Reserved Area cannot be read, written, or erased. Any attempt to access the reserved area, or any other locked page, will result in a FLASH Error device reset.

Accessing FLASH **from user firmware** executing on a **locked page**:

1. Any unlocked page except the page containing the Lock Byte may be read, written, or erased.
2. Any locked page except the page containing the Lock Byte may be read, written, or erased.
3. The page containing the Lock Byte cannot be erased. It may only be read or written.
4. Reading the contents of the Lock Byte is always permitted.
5. Locking additional pages (changing 1s to 0s in the Lock Byte) is not permitted.
6. Unlocking FLASH pages (changing 0s to 1s in the Lock Byte) is not permitted.
7. The Reserved Area cannot be read, written, or erased. Any attempt to access the reserved area, or any other locked page, will result in a FLASH Error device reset.

SFR Definition 19.6. OSCXCN: External Oscillator Control

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

SFR Address = 0xB1; SFR Page = All Pages

Bit	Name	Function
7	XCLKVLD	**External Oscillator Valid Flag.** Provides External Oscillator status and is valid at all times for all modes of operation except External CMOS Clock Mode and External CMOS Clock Mode with divide by 2. In these modes, XCLKVLD always returns 0. 0: External Oscillator is unused or not yet stable. 1: External Oscillator is running and stable.
6:4	XOSCMD[2:0]	**External Oscillator Mode Select.** 00x: External Oscillator circuit off. 010: External CMOS Clock Mode. 011: External CMOS Clock Mode with divide-by-2 stage. 100: RC Oscillator Mode with divide-by-2 stage. 101: Capacitor Oscillator Mode with divide-by-2 stage. 110: Crystal Oscillator Mode. 111: Crystal Oscillator Mode with divide-by-2 stage.
3	Unused	Read = 0; Write = don't care
2:0	XFCN[2:0]	**External Oscillator Frequency Control Bits.** Set according to the desired frequency for RC mode. Set according to the desired K Factor for C mode.

XFCN	Crystal Mode	RC Mode	C Mode
000	f ≤ 20 kHz	f ≤ 25 kHz	K Factor = 0.87
001	20 kHz < f ≤ 58 kHz	25 kHz < f ≤ 50 kHz	K Factor = 2.6
010	58 kHz < f ≤ 155 kHz	50 kHz < f ≤ 100 kHz	K Factor = 7.7
011	155 kHz < f ≤ 415 kHz	100 kHz < f ≤ 200 kHz	K Factor = 22
100	415 kHz < f ≤ 1.1 MHz	200 kHz < f ≤ 400 kHz	K Factor = 65
101	1.1 MHz < f ≤ 3.1 MHz	400 kHz < f ≤ 800 kHz	K Factor = 180
110	3.1 MHz < f ≤ 8.2 MHz	800 kHz < f ≤ 1.6 MHz	K Factor = 664
111	8.2 MHz < f ≤ 25 MHz	1.6 MHz < f ≤ 3.2 MHz	K Factor = 1590

SFR Definition 20.2. XBR1: Port I/O Crossbar Register 1

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

SFR Address = 0xE2; SFR Page = All Pages

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.
2:0	PCA0ME[2:0]	**PCA Module I/O Enable Bits.** 000: All PCA I/O unavailable at Port pins. 001: CEX0 routed to Port pin. 010: CEX0, CEX1 routed to Port pins. 011: CEX0, CEX1, CEX2 routed to Port pins. 100: CEX0, CEX1, CEX2, CEX3 routed to Port pins. 101: CEX0, CEX1, CEX2, CEX3 routed to Port pins. 11x: Reserved.

SFR Definition 20.3. XBR2: Port I/O Crossbar Register 2

Bit	7	6	5	4	3	2	1	0
Name							SMB1E	URT1E
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xE3; SFR Page = All Pages

Bit	Name	Function
7:2	Reserved	Must write 000000b
1	SMB1E	**SMBus1 I/O Enable.** 0: SMBus1 I/O unavailable at Port pins. 1: SMBus1 I/O routed to Port pins.
0	URT1E	**UART1 I/OEnable.** 0: UART1 I/O unavailable at Port pins. 1: UART1 TX1, RX1 routed to Port pins.

20.3. General Purpose Port I/O

Port pins that remain unassigned by the Crossbar and are not used by analog peripherals can be used for general purpose I/O. Ports 3-0 are accessed through corresponding special function registers (SFRs) that are both byte addressable and bit addressable. Port 4 (C8051F380/2/4/6 only) uses an SFR which is byte-addressable. When writing to a Port, the value written to the SFR is latched to maintain the output data value at each pin. When reading, the logic levels of the Port's input pins are returned regardless of the XBRn settings (i.e., even when the pin is assigned to another signal by the Crossbar, the Port register can always read its corresponding Port I/O pin). The exception to this is the execution of the read-modify-write instructions. The read-modify-write instructions when operating on a Port SFR are the following: ANL, ORL, XRL, JBC, CPL, INC, DEC, DJNZ and MOV, CLR or SETB, when the destination is an individual bit in a Port SFR. For these instructions, the value of the register (not the pin) is read, modified, and written back to the SFR.

USB Register Definition 21.5. CLKREC: Clock Recovery Control

Bit	7	6	5	4	3	2	1	0
Name	CRE	CRSSEN	CRLOW					
Type	R/W	R/W	R/W	R/W				
Reset	0	0	0	0	1	1	1	1

USB Register Address = 0x0F

Bit	Name	Function
7	CRE	**Clock Recovery Enable Bit.** This bit enables/disables the USB clock recovery feature. 0: Clock recovery disabled. 1: Clock recovery enabled.
6	CRSSEN	**Clock Recovery Single Step.** This bit forces the oscillator calibration into single-step mode during clock recovery. 0: Normal calibration mode. 1: Single step mode.
5	CRLOW	**Low Speed Clock Recovery Mode.** This bit must be set to 1 if clock recovery is used when operating as a Low Speed USB device. 0: Full Speed Mode. 1: Low Speed Mode.
4:0	Reserved	Must Write = 01111b.

USB Register Definition 21.8. POWER: USB0 Power

Bit	7	6	5	4	3	2	1	0
Name	ISOUD			USBINH	USBRST	RESUME	SUSMD	SUSEN
Type	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W
Reset	0	0	0	0	0	0	0	0

USB Register Address = 0x01

Bit	Name	Function
7	ISOUD	**ISO Update Bit.** This bit affects all IN Isochronous endpoints. 0: When software writes INPRDY = 1, USB0 will send the packet when the next IN token is received. 1: When software writes INPRDY = 1, USB0 will wait for a SOF token before sending the packet. If an IN token is received before a SOF token, USB0 will send a zero-length data packet.
6:5	Unused	Read = 00b. Write = don't care.
4	USBINH	**USB0 Inhibit Bit.** This bit is set to 1 following a power-on reset (POR) or an asynchronous USB0 reset. Software should clear this bit after all USB0 transceiver initialization is complete. Software cannot set this bit to 1. 0: USB0 enabled. 1: USB0 inhibited. All USB traffic is ignored.
3	USBRST	**Reset Detect.** / **Read:** 0: Reset signaling is not present. 1: Reset signaling detected on the bus. / **Write:** Writing 1 to this bit forces an asynchronous USB0 reset.
2	RESUME	**Force Resume.** Writing a 1 to this bit while in Suspend mode (SUSMD = 1) forces USB0 to generate Resume signaling on the bus (a remote wakeup event). Software should write RESUME = 0 after 10 to 15 ms to end the Resume signaling. An interrupt is generated, and hardware clears SUSMD, when software writes RESUME = 0.
1	SUSMD	**Suspend Mode.** Set to 1 by hardware when USB0 enters suspend mode. Cleared by hardware when software writes RESUME = 0 (following a remote wakeup) or reads the CMINT register after detection of Resume signaling on the bus. 0: USB0 not in suspend mode. 1: USB0 in suspend mode.
0	SUSEN	**Suspend Detection Enable.** 0: Suspend detection disabled. USB0 will ignore suspend signaling on the bus. 1: Suspend detection enabled. USB0 will enter suspend mode if it detects suspend signaling on the bus.

USB Register Definition 21.16. CMIE: USB0 Common Interrupt Enable

Bit	7	6	5	4	3	2	1	0
Name					SOFE	RSTINTE	RSUINTE	SUSINTE
Type	R	R	R	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	1	1	0

USB Register Address = 0x0B

Bit	Name	Function
7:4	Unused	Read = 0000b. Write = don't care.
3	SOFE	**Start of Frame Interrupt Enable.** 0: SOF interrupt disabled. 1: SOF interrupt enabled.
2	RSTINTE	**Reset Interrupt Enable.** 0: Reset interrupt disabled. 1: Reset interrupt enabled.
1	RSUINTE	**Resume Interrupt Enable.** 0: Resume interrupt disabled. 1: Resume interrupt enabled.
0	SUSINTE	**Suspend Interrupt Enable.** 0: Suspend interrupt disabled. 1: Suspend interrupt enabled.

21.9. The Serial Interface Engine

The Serial Interface Engine (SIE) performs all low level USB protocol tasks, interrupting the processor when data has successfully been transmitted or received. When receiving data, the SIE will interrupt the processor when a complete data packet has been received; appropriate handshaking signals are automatically generated by the SIE. When transmitting data, the SIE will interrupt the processor when a complete data packet has been transmitted and the appropriate handshake signal has been received.

The SIE will not interrupt the processor when corrupted/erroneous packets are received.

21.10. Endpoint0

Endpoint0 is managed through the USB register E0CSR (USB Register Definition 21.18). The INDEX register must be loaded with 0x00 to access the E0CSR register.

An Endpoint0 interrupt is generated when:

1. A data packet (OUT or SETUP) has been received and loaded into the Endpoint0 FIFO. The OPRDY bit (E0CSR.0) is set to 1 by hardware.
2. An IN data packet has successfully been unloaded from the Endpoint0 FIFO and transmitted to the host; INPRDY is reset to 0 by hardware.
3. An IN transaction is completed (this interrupt generated during the status stage of the transaction).
4. Hardware sets the STSTL bit (E0CSR.2) after a control transaction ended due to a protocol violation.
5. Hardware sets the SUEND bit (E0CSR.4) because a control transfer ended before firmware sets the DATAEND bit (E0CSR.3).

The E0CNT register (USB Register Definition 21.11) holds the number of received data bytes in the Endpoint0 FIFO.

Hardware will automatically detect protocol errors and send a STALL condition in response. Firmware may force a STALL condition to abort the current transfer. When a STALL condition is generated, the STSTL bit will be set to 1 and an interrupt generated. The following conditions will cause hardware to generate a STALL condition:

1. The host sends an OUT token during a OUT data phase after the DATAEND bit has been set to 1.
2. The host sends an IN token during an IN data phase after the DATAEND bit has been set to 1.
3. The host sends a packet that exceeds the maximum packet size for Endpoint0.
4. The host sends a non-zero length DATA1 packet during the status phase of an IN transaction.

Firmware sets the SDSTL bit (E0CSR.5) to 1.

21.10.1. Endpoint0 SETUP Transactions

All control transfers must begin with a SETUP packet. SETUP packets are similar to OUT packets, containing an 8-byte data field sent by the host. Any SETUP packet containing a command field of anything other than 8 bytes will be automatically rejected by USB0. An Endpoint0 interrupt is generated when the data from a SETUP packet is loaded into the Endpoint0 FIFO. Software should unload the command from the Endpoint0 FIFO, decode the command, perform any necessary tasks, and set the SOPRDY bit to indicate that it has serviced the OUT packet.

21.10.2. Endpoint0 IN Transactions

When a SETUP request is received that requires USB0 to transmit data to the host, one or more IN requests will be sent by the host. For the first IN transaction, firmware should load an IN packet into the Endpoint0 FIFO, and set the INPRDY bit (E0CSR.1). An interrupt will be generated when an IN packet is transmitted successfully. Note that no interrupt will be generated if an IN request is received before firm-

the incoming slave address. Additionally, if the GCn bit in register SMBnADR is set to 1, hardware will recognize the General Call Address (0x00). Table 22.4 shows some example parameter settings and the slave addresses that will be recognized by hardware under those conditions.

Table 22.4. Hardware Address Recognition Examples (EHACK = 1)

Hardware Slave Address SLVn[6:0]	Slave Address Mask SLVMn[6:0]	GCn bit	Slave Addresses Recognized by Hardware
0x34	0x7F	0	0x34
0x34	0x7F	1	0x34, 0x00 (General Call)
0x34	0x7E	0	0x34, 0x35
0x34	0x7E	1	0x34, 0x35, 0x00 (General Call)
0x70	0x73	0	0x70, 0x74, 0x78, 0x7C

SFR Definition 22.6. SMB0ADR: SMBus0 Slave Address

Bit	7	6	5	4	3	2	1	0
Name	SLV0[6:0]							GC0
Type	R/W							R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xCF; SFR Page = 0

Bit	Name	Function
7:1	SLV0[6:0]	**SMBus Hardware Slave Address.** Defines the SMBus0 Slave Address(es) for automatic hardware acknowledgement. Only address bits which have a 1 in the corresponding bit position in SLVM0[6:0] are checked against the incoming address. This allows multiple addresses to be recognized.
0	GC0	**General Call Address Enable.** When hardware address recognition is enabled (EHACK0 = 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 22.7. SMB0ADM: SMBus0 Slave Address Mask

Bit	7	6	5	4	3	2	1	0
Name	SLVM0[6:0]							EHACK0
Type	R/W							R/W
Reset	1	1	1	1	1	1	1	0

SFR Address = 0xCE; SFR Page = 0

Bit	Name	Function
7:1	SLVM0[6:0]	**SMBus0 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 SLVM0[6:0] enables comparisons with the corresponding bit in SLV0[6:0]. Bits set to 0 are ignored (can be either 0 or 1 in the incoming address).
0	EHACK0	**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.

SFR Definition 22.8. SMB1ADR: SMBus1 Slave Address

Bit	7	6	5	4	3	2	1	0
Name	SLV1[6:0]							GC1
Type	R/W							R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xCF; SFR Page = F

Bit	Name	Function
7:1	SLV1[6:0]	**SMBus1 Hardware Slave Address.** Defines the SMBus1 Slave Address(es) for automatic hardware acknowledgement. Only address bits which have a 1 in the corresponding bit position in SLVM1[6:0] are checked against the incoming address. This allows multiple addresses to be recognized.
0	GC1	**General Call Address Enable.** When hardware address recognition is enabled (EHACK1 = 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.

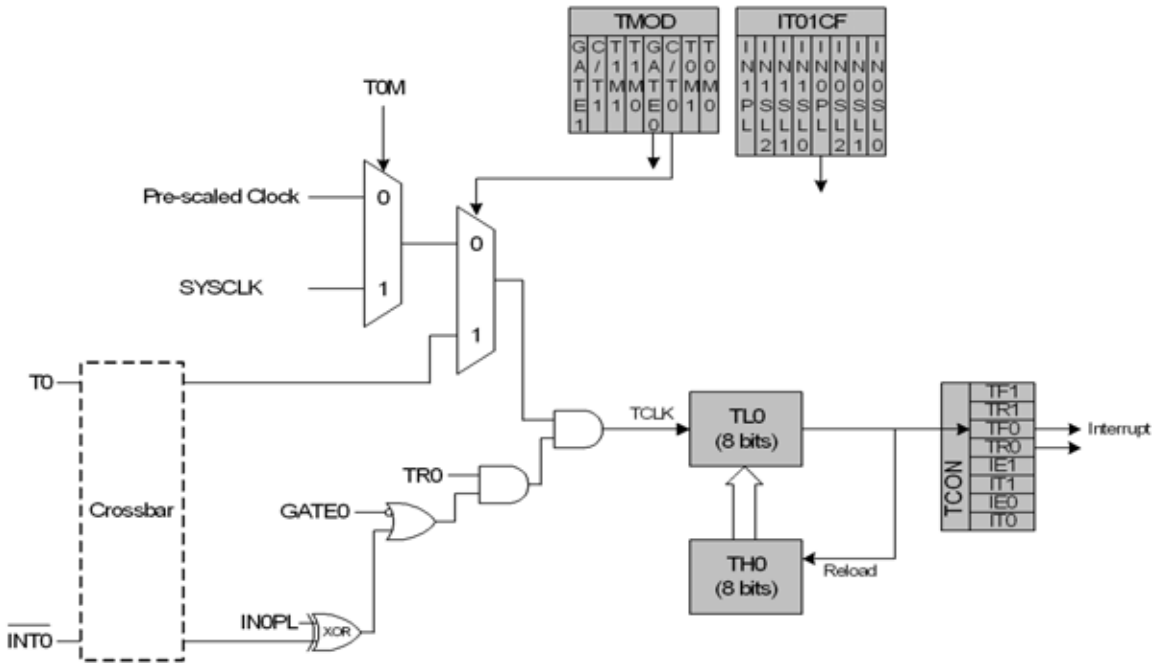

Figure 26.2. T0 Mode 2 Block Diagram

26.1.4. 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 26.14. TMR3CN: Timer 3 Control

Bit	7	6	5	4	3	2	1	0
Name	TF3H	TF3L	TF3LEN	TF3CEN	T3SPLIT	TR3	T3CSS	T3XCLK
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x91; SFR Page = 0

Bit	Name	Function
7	TF3H	**Timer 3 High Byte Overflow Flag.** Set by hardware when the Timer 3 high byte overflows from 0xFF to 0x00. In 16 bit mode, this will occur when Timer 3 overflows from 0xFFFF to 0x0000. When the Timer 3 interrupt is enabled, setting this bit causes the CPU to vector to the Timer 3 interrupt service routine. This bit is not automatically cleared by hardware.
6	TF3L	**Timer 3 Low Byte Overflow Flag.** Set by hardware when the Timer 3 low byte overflows from 0xFF to 0x00. TF3L will be set when the low byte overflows regardless of the Timer 3 mode. This bit is not automatically cleared by hardware.
5	TF3LEN	**Timer 3 Low Byte Interrupt Enable.** When set to 1, this bit enables Timer 3 Low Byte interrupts. If Timer 3 interrupts are also enabled, an interrupt will be generated when the low byte of Timer 3 overflows.
4	TF3CEN	**Timer 3 Low-Frequency Oscillator Capture Enable.** When set to 1, this bit enables Timer 3 Low-Frequency Oscillator Capture Mode. If TF3CEN is set and Timer 3 interrupts are enabled, an interrupt will be generated on a falling edge of the low-frequency oscillator output, and the current 16-bit timer value in TMR3H:TMR3L will be copied to TMR3RLH:TMR3RLL.
3	T3SPLIT	**Timer 3 Split Mode Enable.** When this bit is set, Timer 3 operates as two 8-bit timers with auto-reload.
2	TR3	**Timer 3 Run Control.** Timer 3 is enabled by setting this bit to 1. In 8-bit mode, this bit enables/disables TMR3H only; TMR3L is always enabled in split mode.
1	T3CSS	**Timer 3 Capture Source Select.** This bit selects the source of a capture event when bit T2CE is set to 1. 0: Capture source is USB SOF event. 1: Capture source is falling edge of Low-Frequency Oscillator.
0	T3XCLK	**Timer 3 External Clock Select.** This bit selects the external clock source for Timer 3. However, the Timer 3 Clock Select bits (T3MH and T3ML in register CKCON) may still be used to select between the external clock and the system clock for either timer. 0: Timer 3 clock is the system clock divided by 12. 1: Timer 3 clock is the external clock divided by 8 (synchronized with SYSCLK).

27.3.4. Frequency Output Mode

Frequency Output Mode produces a programmable-frequency square wave on the module's associated CEXn pin. The capture/compare module high byte holds the number of PCA clocks to count before the output is toggled. The frequency of the square wave is then defined by Equation 27.1.

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

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

Equation 27.1. Square Wave Frequency Output

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

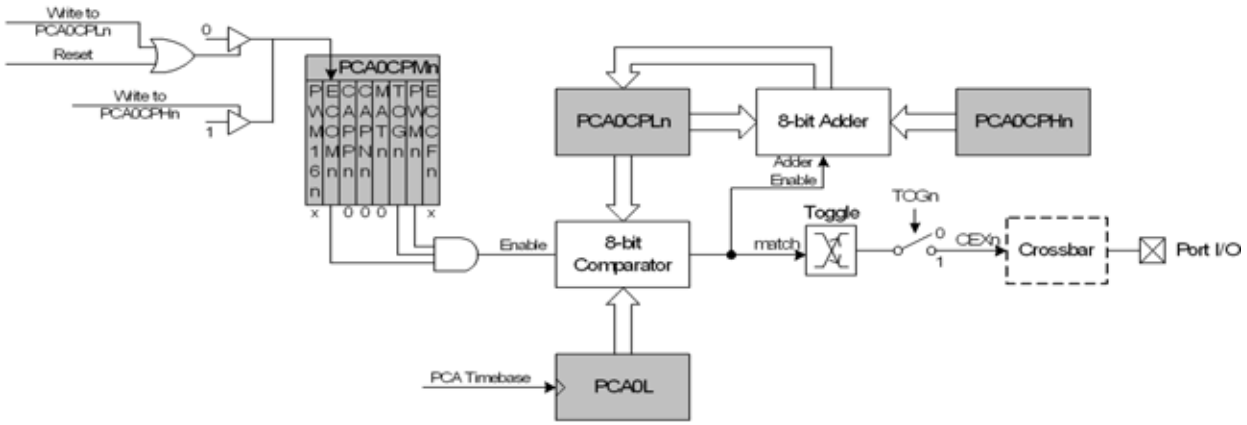

Figure 27.7. PCA Frequency Output Mode