

Welcome to [E-XFL.COM](https://www.e-xfl.com)

What is "[Embedded - Microcontrollers](#)"?



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

Applications of "[Embedded - Microcontrollers](#)"

Details

Product Status	Obsolete
Core Processor	8051
Core Size	8-Bit
Speed	25MHz
Connectivity	SMBus (2-Wire/I ² C), UART/USART
Peripherals	POR, PWM, Temp Sensor, WDT
Number of I/O	8
Program Memory Size	8KB (8K x 8)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	10-VDFN Exposed Pad
Supplier Device Package	11-QFN (3x3)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051t600-gm

18. Power Management Modes	
19. Reset Sources	
20. EPROM Memory	
Table 20.1. Security Byte Decoding	98
21. Oscillators and Clock Selection	
22. Port Input/Output	
Table 22.1. Port I/O Assignment for Analog Functions	109
Table 22.2. Port I/O Assignment for Digital Functions	109
Table 22.3. Port I/O Assignment for External Digital Event Capture Functions	110
23. SMBus	
Table 23.1. SMBus Clock Source Selection	124
Table 23.2. Minimum SDA Setup and Hold Times	125
Table 23.3. Sources for Hardware Changes to SMB0CN	129
Table 23.4. SMBus Status Decoding	135
24. UART0	
Table 24.1. Timer Settings for Standard Baud Rates Using The Internal 24.5 MHz Oscillator	144
Table 24.2. Timer Settings for Standard Baud Rates Using an External 22.1184 MHz Oscillator	144
25. Timers	
26. Programmable Counter Array	
Table 26.1. PCA Timebase Input Options	161
Table 26.2. PCA0CPM Bit Settings for PCA Capture/Compare Modules	163
Table 26.3. Watchdog Timer Timeout Intervals ¹	172
27. C2 Interface	

4. QFN-11 Package Specifications

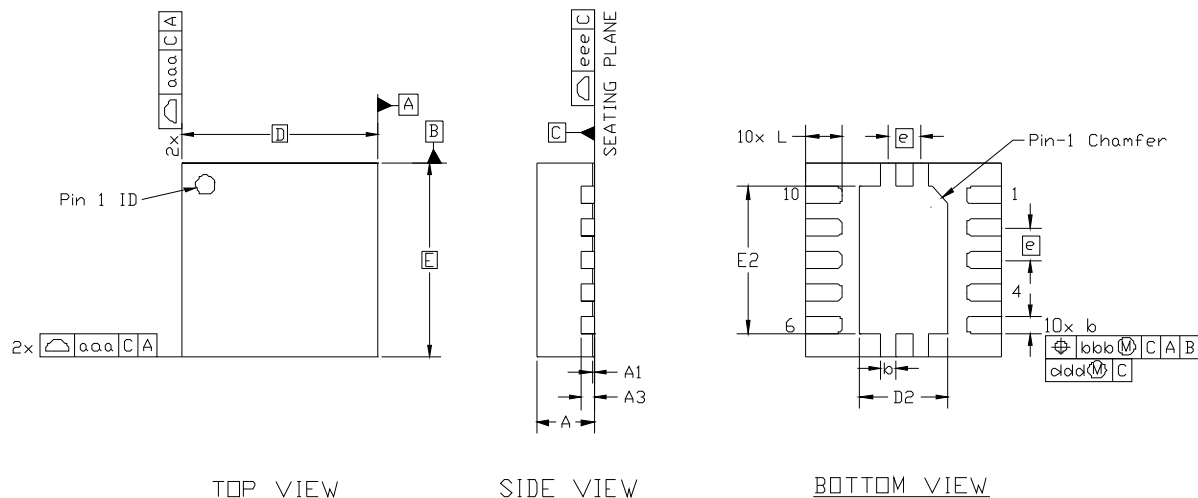


Figure 4.1. QFN-11 Package Drawing

Table 4.1. QFN-11 Package Dimensions

Dimension	Min	Nom	Max	Dimension	Min	Nom	Max
A	0.80	0.90	1.00	E	3.00 BSC		
A1	0.03	0.07	0.11	E2	2.20	2.25	2.30
A3	0.25 REF			L	0.45	0.55	0.65
b	0.18	0.25	0.30	aaa	—	—	0.15
D	3.00 BSC			bbb	—	—	0.15
D2	1.30	1.35	1.40	ddd	—	—	0.05
e	0.50 BSC			eee	—	—	0.08

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 MO-243, variation VEED except for custom features D2, E2, and L which are toleranced per supplier designation.
4. Recommended card reflow profile is per the JEDEC/IPC J-STD-020 specification for Small Body Components.

C8051T600/1/2/3/4/5/6

Table 8.2. Global Electrical Characteristics

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

Parameter	Conditions	Min	Typ	Max	Units
Tsysl (SYSCLK low time)		18	—	—	ns
Tsysh (SYSCLK high time)		18	—	—	ns
Notes: <ol style="list-style-type: none">1. Analog performance is not guaranteed when V_{DD} is below 1.8 V.2. SYSCLK must be at least 32 kHz to enable debugging.3. Supply current parameters specified with Memory Power Controller enabled.					

Table 8.3. Port I/O DC Electrical Characteristics

$V_{DD} = 1.8$ to 3.6 V, -40 to $+85$ °C unless otherwise specified.

Parameters	Conditions	Min	Typ	Max	Units
Output High Voltage	$I_{OH} = -3$ mA, Port I/O push-pull	$V_{DD} - 0.3$	—	—	V
	$I_{OH} = -10$ μ A, Port I/O push-pull	$V_{DD} - 0.1$	—	—	V
	$I_{OH} = -10$ mA, Port I/O push-pull	—	$V_{DD} - 0.5$	—	V
Output Low Voltage	$I_{OL} = 8.5$ mA	—	—	0.6	V
	$I_{OL} = 10$ μ A	—	—	0.1	V
	$I_{OL} = 25$ mA	—	$0.4 \times V_{DD}$	—	V
Input High Voltage		$0.7 \times V_{DD}$	—	—	V
Input Low Voltage		—	—	0.6	V
Input Leakage Current	Weak Pullup Off	-1	—	1	μ A
	Weak Pullup On, $V_{IN} = 0$ V	—	25	50	μ A

externally driven from -0.25 V to $(V_{DD}) + 0.25\text{ V}$ without damage or upset. The complete Comparator electrical specifications are given in Section “8. Electrical Characteristics” on page 30.

The Comparator response time may be configured in software via the CPT0MD register (see SFR Definition 13.2). Selecting a longer response time reduces the Comparator supply current.

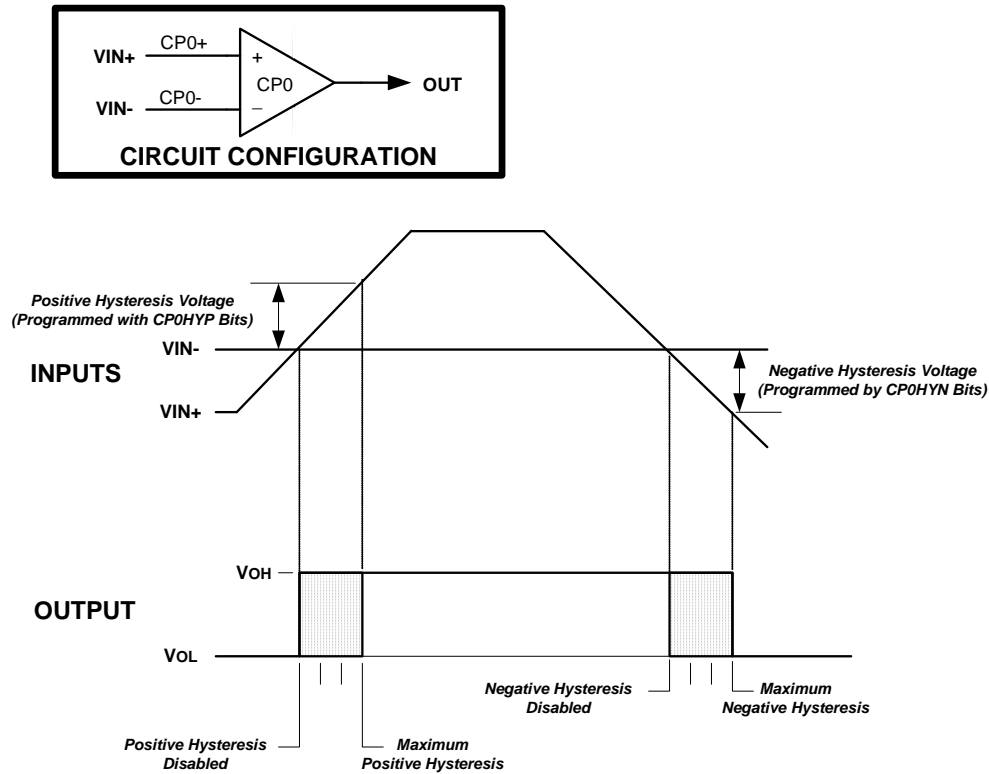


Figure 13.2. Comparator Hysteresis Plot

The Comparator hysteresis is software-programmable via its Comparator Control register CPT0CN. The user can program both the amount of hysteresis voltage (referred to as the input voltage) and the positive and negative-going symmetry of this hysteresis around the threshold voltage.

The Comparator hysteresis is programmed using Bits3–0 in the Comparator Control Register CPT0CN (shown in SFR Definition 13.1). The amount of negative hysteresis voltage is determined by the settings of the CP0HYN bits. As shown in Figure 13.2, settings of 20, 10 or 5 mV of negative hysteresis can be programmed, or negative hysteresis can be disabled. In a similar way, the amount of positive hysteresis is determined by the setting the CP0HYP bits.

Comparator interrupts can be generated on both rising-edge and falling-edge output transitions. (For Interrupt enable and priority control, see Section “17.1. MCU Interrupt Sources and Vectors” on page 81). The CP0FIF flag is set to logic 1 upon a Comparator falling-edge occurrence, and the CP0RIF flag is set to logic 1 upon the Comparator rising-edge occurrence. Once set, these bits remain set until cleared by software.

The output state of the Comparator can be obtained at any time by reading the CP0OUT bit. The Comparator is enabled by setting the CP0EN bit to logic 1, and is disabled by clearing this bit to logic 0.

SFR Definition 13.3. CPT0MX: Comparator0 MUX Selection

Bit	7	6	5	4	3	2	1	0
Name			CMX0N[1:0]				CMX0P[1:0]	
Type	R	R	R/W		R	R	R/W	
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x9F

Bit	Name	Function
7:6	Unused	Unused. Read = 00b; Write = Don't Care.
5:4	CMX0N[1:0]	Comparator0 Negative Input MUX Selection. 00: P0.1 01: P0.3 10: P0.5 11: P0.7
3:2	Unused	Unused. Read = 00b; Write = Don't Care.
1:0	CMX0P[1:0]	Comparator0 Positive Input MUX Selection. 00: P0.0 (Available only on packages with 8 I/O pins) 01: P0.2 10: P0.4 11: P0.6 (Available only on packages with 8 I/O pins)

With the CIP-51's maximum system clock at 25 MHz, it has a peak throughput of 25 MIPS. The CIP-51 has a total of 109 instructions. The table below shows the total number of instructions that require each execution time.

Clocks to Execute	1	2	2/3	3	3/4	4	4/5	5	8
Number of Instructions	26	50	5	14	7	3	1	2	1

14.1. Instruction Set

The instruction set of the CIP-51 System Controller is fully compatible with the standard MCS-51™ instruction set. Standard 8051 development tools can be used to develop software for the CIP-51. All CIP-51 instructions are the binary and functional equivalent of their MCS-51™ counterparts, including opcodes, addressing modes and effect on PSW flags. However, instruction timing is different than that of the standard 8051.

14.1.1. Instruction and CPU Timing

In many 8051 implementations, a distinction is made between machine cycles and clock cycles, with machine cycles varying from 2 to 12 clock cycles in length. However, the CIP-51 implementation is based solely on clock cycle timing. All instruction timings are specified in terms of clock cycles.

Due to the pipelined architecture of the CIP-51, most instructions execute in the same number of clock cycles as there are program bytes in the instruction. Conditional branch instructions take one less clock cycle to complete when the branch is not taken as opposed to when the branch is taken. Table 14.1 is the CIP-51 Instruction Set Summary, which includes the mnemonic, number of bytes, and number of clock cycles for each instruction.

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

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

C8051T600/1/2/3/4/5/6

SFR Definition 14.6. 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; 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: <ul style="list-style-type: none"> ■ 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.

15.2.1.1. General Purpose Registers

The lower 32 bytes of data memory, locations 0x00 through 0x1F, may be addressed as four banks of general-purpose registers. Each bank consists of eight byte-wide registers designated R0 through R7. Only one of these banks may be enabled at a time. Two bits in the program status word, RS0 (PSW.3) and RS1 (PSW.4), select the active register bank (see description of the PSW in SFR Definition 14.6). This allows fast context switching when entering subroutines and interrupt service routines. Indirect addressing modes use registers R0 and R1 as index registers.

15.2.1.2. Bit Addressable Locations

In addition to direct access to data memory organized as bytes, the sixteen data memory locations at 0x20 through 0x2F are also accessible as 128 individually addressable bits. Each bit has a bit address from 0x00 to 0x7F. Bit 0 of the byte at 0x20 has bit address 0x00 while bit 7 of the byte at 0x20 has bit address 0x07. Bit 7 of the byte at 0x2F has bit address 0x7F. A bit access is distinguished from a full byte access by the type of instruction used (bit source or destination operands as opposed to a byte source or destination).

The MCS-51™ assembly language allows an alternate notation for bit addressing of the form XX.B where XX is the byte address and B is the bit position within the byte. For example, the instruction:

```
MOV    C, 22.3h
```

moves the Boolean value at 0x13 (bit 3 of the byte at location 0x22) into the Carry flag.

15.2.1.3. Stack

A programmer's stack can be located anywhere in the internal data memory. The stack area is designated using the Stack Pointer (SP) SFR. The SP will point to the last location used. The next value pushed on the stack is placed at SP+1 and then SP is incremented. A reset initializes the stack pointer to location 0x07. Therefore, the first value pushed on the stack is placed at location 0x08, which is also the first register (R0) of register bank 1. Thus, if more than one register bank is to be used, the SP should be initialized to a location in the data memory not being used for data storage. The stack depth can extend up to the full RAM area.

C8051T600/1/2/3/4/5/6

SFR Definition 17.5. IT01CF: INT0/INT1 Configuration

Bit	7	6	5	4	3	2	1	0
Name	IN1PL	IN1SL[2:0]			IN0PL	IN0SL[2:0]		
Type	R/W	R/W			R/W	R/W		
Reset	0	0	0	0	0	0	0	1

SFR Address = 0xE4

Bit	Name	Function
7	IN1PL	$\overline{\text{INT1}}$ Polarity. 0: $\overline{\text{INT1}}$ input is active low. 1: $\overline{\text{INT1}}$ input is active high.
6:4	IN1SL[2:0]	$\overline{\text{INT1}}$ Port Pin Selection Bits. These bits select which Port pin is assigned to $\overline{\text{INT1}}$. Note that this pin assignment is independent of the Crossbar; $\overline{\text{INT1}}$ will monitor the assigned Port pin without disturbing the peripheral that has been assigned the Port pin via the Crossbar. The Crossbar will not assign the Port pin to a peripheral if it is configured to skip the selected pin. 000: Select P0.0 001: Select P0.1 010: Select P0.2 011: Select P0.3 100: Select P0.4 101: Select P0.5 110: Select P0.6 111: Select P0.7
3	IN0PL	$\overline{\text{INT0}}$ Polarity. 0: $\overline{\text{INT0}}$ input is active low. 1: $\overline{\text{INT0}}$ input is active high.
2:0	IN0SL[2:0]	$\overline{\text{INT0}}$ Port Pin Selection Bits. These bits select which Port pin is assigned to $\overline{\text{INT0}}$. Note that this pin assignment is independent of the Crossbar; $\overline{\text{INT0}}$ will monitor the assigned Port pin without disturbing the peripheral that has been assigned the Port pin via the Crossbar. The Crossbar will not assign the Port pin to a peripheral if it is configured to skip the selected pin. 000: Select P0.0 001: Select P0.1 010: Select P0.2 011: Select P0.3 100: Select P0.4 101: Select P0.5 110: Select P0.6 111: Select P0.7

C8051T600/1/2/3/4/5/6

SFR Definition 18.1. 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

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 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.)

C8051T600/1/2/3/4/5/6

SFR Definition 19.1. RSTSRC: Reset Source

Bit	7	6	5	4	3	2	1	0
Name		MEMERR	CORSEF	SWRSF	WDTRSF	MCDRSF	PORSF	PINRSF
Type	R	R	R/W	R/W	R	R/W	R/W	R
Reset	0	Varies	Varies	Varies	Varies	Varies	Varies	Varies

SFR Address = 0xEF

Bit	Name	Description	Write	Read
7	Unused	Unused.	Don't care.	0
6	MEMERR	EPROM Error Reset Flag.	N/A	Set to 1 if EPROM read/write error caused the last reset.
5	CORSEF	Comparator0 Reset Enable and Flag.	Writing a 1 enables Comparator0 as a reset source (active-low).	Set to 1 if Comparator0 caused the last reset.
4	SWRSF	Software Reset Force and Flag.	Writing a 1 forces a system reset.	Set to 1 if last reset was caused by a write to SWRSF.
3	WDTRSF	Watchdog Timer Reset Flag.	N/A	Set to 1 if Watchdog Timer overflow caused the last reset.
2	MCDRSF	Missing Clock Detector Enable and Flag.	Writing a 1 enables the Missing Clock Detector. The MCD triggers a reset if a missing clock condition is detected.	Set to 1 if Missing Clock Detector timeout caused the last reset.
1	PORSF	Power-On/V_{DD} Monitor Reset Flag, and V_{DD} monitor Reset Enable.	Writing a 1 enables the V _{DD} monitor and configures it as a reset source. Writing 1 to this bit while the V_{DD} monitor is disabled may cause a system reset.	Set to 1 any time a power-on or V _{DD} monitor reset occurs. When set to 1, all other RSTSRC flags are indeterminate.
0	PINRSF	HW Pin Reset Flag.	N/A	Set to 1 if RST pin caused the last reset.

Note: Do not use read-modify-write operations on this register

20.1.2. EPROM Read Procedure

1. Reset the device using the $\overline{\text{RST}}$ pin.
2. Wait at least 20 μs before sending the first C2 command.
3. Place the device in core reset: Write 0x04 to the DEVCTL register.
4. Write 0x00 to the EPCTL register.
5. Write the first EPROM address for reading to EPADDRH and EPADDRL.
6. Read a data byte from EPDAT. EPADDRH:L will increment by 1 after this read.
7. (Optional) Check the ERROR bit in register EPSTAT and abort the memory read operation if necessary.
8. If reading is not finished, return to Step 6 to read the next address in sequence, or return to Step 5 to select a new address.
9. Remove read mode (1st step): Write 0x40 to the EPCTL register.
10. Remove read mode (2nd step): Write 0x00 to the EPCTL register.
11. Reset the device: Write 0x02 and then 0x00 to the DEVCTL register.

20.2. Security Options

The C8051T600/1/2/3/4/5/6 devices provide security options to prevent unauthorized viewing of proprietary program code and constants. A security byte in EPROM address space can be used to lock the program memory from being read or written across the C2 interface. When read, the RDLOCK and WRLOCK bits in register EPSTAT will indicate the lock status of the location currently addressed by EPADDR. Table 20.1 shows the security byte decoding. See Section “15. Memory Organization” on page 74 for the security byte location and EPROM memory map.

Important Note: Once the security byte has been written, there are no means of unlocking the device. Locking memory from write access should be performed only after all other code has been successfully programmed to memory.

Table 20.1. Security Byte Decoding

Bits	Description
7–4	Write Lock: Clearing any of these bits to logic 0 prevents all code memory from being written across the C2 interface.
3–0	Read Lock: Clearing any of these bits to logic 0 prevents all code memory from being read across the C2 interface.

20.3. Program Memory CRC

A CRC engine is included on-chip, which provides a means of verifying EPROM contents once the device has been programmed. The CRC engine is available for EPROM verification even if the device is fully read and write locked, allowing for verification of code contents at any time.

The CRC engine is operated through the C2 debug and programming interface, and performs 16-bit CRCs on individual 256-byte blocks of program memory, or a 32-bit CRC the entire memory space. To prevent hacking and extrapolation of security-locked source code, the CRC engine will only allow CRCs to be performed on contiguous 256-byte blocks beginning on 256-byte boundaries (lowest 8-bits of address are 0x00). For example, the CRC engine can perform a CRC for locations 0x0400 through 0x04FF, but it cannot perform a CRC for locations 0x0401 through 0x0500, or on block sizes smaller or larger than 256 bytes.

20.3.1. Performing 32-bit CRCs on Full EPROM Content

A 32-bit CRC on the entire EPROM space is initiated by writing to the CRC1 byte over the C2 interface. The CRC calculation begins at address 0x0000 and ends at the end of user EPROM space. The EPBusy bit in register C2ADD will be set during the CRC operation, and cleared once the operation is complete. The 32-bit results will be available in the CRC3-0 registers. CRC3 is the MSB, and CRC0 is the LSB. The polynomial used for the 32-bit CRC calculation is 0x04C11DB7.

Note: If a 16-bit CRC has been performed since the last device reset, a device reset should be initiated before performing a 32-bit CRC operation.

20.3.2. Performing 16-bit CRCs on 256-Byte EPROM Blocks

A 16-bit CRC of individual 256-byte blocks of EPROM can be initiated by writing to the CRC0 byte over the C2 interface. The value written to CRC0 is the high byte of the beginning address for the CRC. For example, if CRC0 is written to 0x02, the CRC will be performed on the 256 bytes beginning at address 0x0200, and ending at address 0x2FF. The EPBusy bit in register C2ADD will be set during the CRC operation, and cleared once the operation is complete. The 16-bit results will be available in the CRC1-0 registers. CRC1 is the MSB, and CRC0 is the LSB. The polynomial for the 16-bit CRC calculation is 0x1021

23.4.2. SMB0CN Control Register

SMB0CN is used to control the interface and to provide status information (see SFR Definition 23.2). The higher four bits of SMB0CN (MASTER, TXMODE, STA, and STO) form a status vector that can be used to jump to service routines. MASTER indicates whether a device is the master or slave during the current transfer. TXMODE indicates whether the device is transmitting or receiving data for the current byte.

STA and STO indicate that a START and/or STOP has been detected or generated since the last SMBus interrupt. STA and STO are also used to generate START and STOP conditions when operating as a master. Writing a 1 to STA will cause the SMBus interface to enter Master Mode and generate a START when the bus becomes free (STA is not cleared by hardware after the START is generated). Writing a 1 to STO while in Master Mode will cause the interface to generate a STOP and end the current transfer after the next ACK cycle. If STO and STA are both set (while in Master Mode), a STOP followed by a START will be generated.

As a receiver, writing the ACK bit defines the outgoing ACK value; as a transmitter, reading the ACK bit indicates the value received on the last ACK cycle. ACKRQ is set each time a byte is received, indicating that an outgoing ACK value is needed. When ACKRQ is set, software should write the desired outgoing value to the ACK bit before clearing SI. A NACK will be generated if software does not write the ACK bit before clearing SI. SDA will reflect the defined ACK value immediately following a write to the ACK bit; however SCL will remain low until SI is cleared. If a received slave address is not acknowledged, further slave events will be ignored until the next START is detected.

The ARBLOST bit indicates that the interface has lost an arbitration. This may occur anytime the interface is transmitting (master or slave). A lost arbitration while operating as a slave indicates a bus error condition. ARBLOST is cleared by hardware each time SI is cleared.

The SI bit (SMBus Interrupt Flag) is set at the beginning and end of each transfer, after each byte frame, or when an arbitration is lost; see Table 23.3 for more details.

Important Note About the SI Bit: The SMBus interface is stalled while SI is set; thus SCL is held low, and the bus is stalled until software clears SI.

Table 23.3 lists all sources for hardware changes to the SMB0CN bits. Refer to Table 23.4 for SMBus status decoding using the SMB0CN register.

25.1.3. Mode 2: 8-bit Counter/Timer with Auto-Reload

Mode 2 configures Timer 0 and Timer 1 to operate as 8-bit counter/timers with automatic reload of the start value. TL0 holds the count and TH0 holds the reload value. When the counter in TL0 overflows from all ones to 0x00, the timer overflow flag TF0 in the TCON register is set and the counter in TL0 is reloaded from TH0. If Timer 0 interrupts are enabled, an interrupt will occur when the TF0 flag is set. The reload value in TH0 is not changed. TL0 must be initialized to the desired value before enabling the timer for the first count to be correct. When in Mode 2, Timer 1 operates identically to Timer 0.

Both counter/timers are enabled and configured in Mode 2 in the same manner as Mode 0. Setting the TR0 bit (TCON.4) enables the timer when either GATE0 in the TMOD register is logic 0 or when the input signal INT0 is active as defined by bit IN0PL in register IT01CF (see Section “17.3. INT0 and INT1 External Interrupt Sources” on page 87 for details on the external input signals INT0 and INT1).

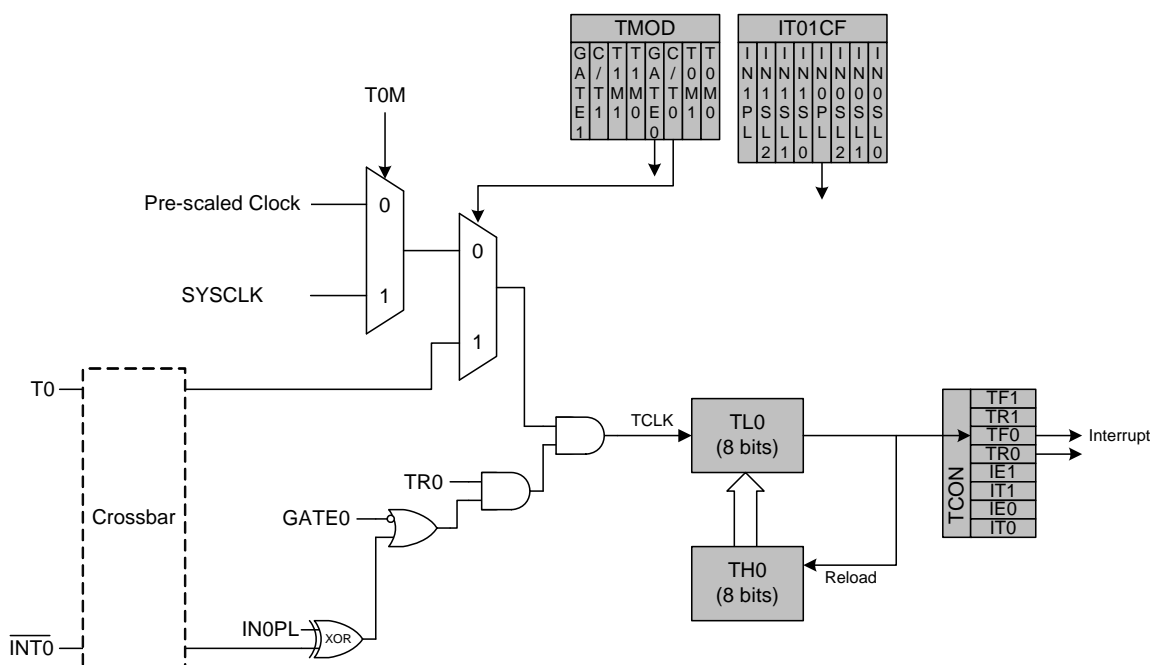


Figure 25.2. T0 Mode 2 Block Diagram

25.2.2. 8-bit Timers with Auto-Reload

When T2SPLIT is set, Timer 2 operates as two 8-bit timers (TMR2H and TMR2L). Both 8-bit timers operate in auto-reload mode as shown in Figure 25.5. TMR2RLL holds the reload value for TMR2L; TMR2RLH holds the reload value for TMR2H. The TR2 bit in TMR2CN handles the run control for TMR2H. TMR2L is always running when configured for 8-bit Mode.

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

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

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

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

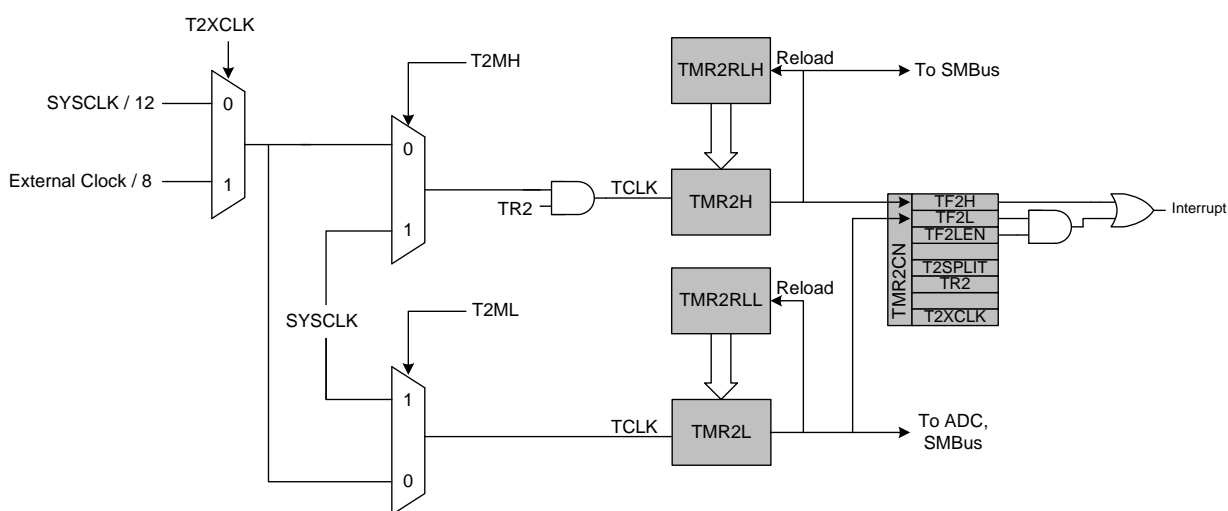


Figure 25.5. Timer 2 8-Bit Mode Block Diagram

26.3. Capture/Compare Modules

Each module can be configured to operate independently in one of six operation modes: edge-triggered capture, software timer, high-speed output, frequency output, 8-bit pulse width modulator, or 16-bit pulse width modulator. Each module has Special Function Registers (SFRs) associated with it in the CIP-51 system controller. These registers are used to exchange data with a module and configure the module's mode of operation. Table 26.2 summarizes the bit settings in the PCA0CPMn register used to select the PCA capture/compare module's operating mode. Setting the ECCFn bit in a PCA0CPMn register enables the module's CCFn interrupt.

Table 26.2. PCA0CPM Bit Settings for PCA Capture/Compare Modules

Operational Mode	Bit Number	PCA0CPMn							
		7	6	5	4	3	2	1	0
Capture triggered by positive edge on CEXn		X	X	1	0	0	0	0	A
Capture triggered by negative edge on CEXn		X	X	0	1	0	0	0	A
Capture triggered by any transition on CEXn		X	X	1	1	0	0	0	A
Software Timer		X	B	0	0	1	0	0	A
High Speed Output		X	B	0	0	1	1	0	A
Frequency Output		X	B	0	0	0	1	1	A
8-Bit Pulse Width Modulator		0	B	0	0	C	0	1	A
16-Bit Pulse Width Modulator		1	B	0	0	C	0	1	A
Notes: <ol style="list-style-type: none"> 1. X = Don't Care (no functional difference for individual module if 1 or 0). 2. A = Enable interrupts for this module (PCA interrupt triggered on CCFn set to 1). 3. B = When set to 0, the digital comparator is off. For high speed and frequency output modes, the associated pin will not toggle. In any of the PWM modes, this generates a 0% duty cycle (output = 0). 4. C = When set, a match event will cause the CCFn flag for the associated channel to be set. 									

DOCUMENT CHANGE LIST

Revision 0.5 to Revision 1.0

- Updated electrical specification tables based on test, characterization, and qualification data.
- Updated with new formatting standards.
- Corrected minor typographical errors throughout document.
- Updated wording from “OTP EPROM” to “EPROM” throughout document.
- Added information on C2 EPSTAT Register.
- Updated EPROM programming sequence.
- Added Note about 100% Tin (Sn) lead finish to ordering information table.

Updated packaging information to include JEDEC-standard drawings for package and land diagram.

Revision 1.0 to Revision 1.1

Added C8051T606 device information.

Revision 1.1 to Revision 1.2

- Updated Table 8.4 on page 34.