



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	2KB (2K 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	14-SOIC (0.154", 3.90mm Width)
Supplier Device Package	14-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051t604-gs

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

23.3.2. Arbitration.....	122
23.3.3. Clock Low Extension.....	122
23.3.4. SCL Low Timeout.....	122
23.3.5. SCL High (SMBus Free) Timeout	123
23.4. Using the SMBus.....	123
23.4.1. SMBus Configuration Register.....	123
23.4.2. SMB0CN Control Register	127
23.4.3. Data Register	130
23.5. SMBus Transfer Modes.....	131
23.5.1. Write Sequence (Master)	131
23.5.2. Read Sequence (Master)	132
23.5.3. Write Sequence (Slave)	133
23.5.4. Read Sequence (Slave).....	134
23.6. SMBus Status Decoding.....	134
24. UART0	137
24.1. Enhanced Baud Rate Generation.....	138
24.2. Operational Modes	139
24.2.1. 8-Bit UART	139
24.2.2. 9-Bit UART	140
24.3. Multiprocessor Communications	141
25. Timers	145
25.1. Timer 0 and Timer 1	147
25.1.1. Mode 0: 13-bit Counter/Timer	147
25.1.2. Mode 1: 16-bit Counter/Timer	148
25.1.3. Mode 2: 8-bit Counter/Timer with Auto-Reload.....	149
25.1.4. Mode 3: Two 8-bit Counter/Timers (Timer 0 Only).....	150
25.2. Timer 2	155
25.2.1. 16-bit Timer with Auto-Reload.....	155
25.2.2. 8-bit Timers with Auto-Reload.....	156
26. Programmable Counter Array.....	160
26.1. PCA Counter/Timer	161
26.2. PCA0 Interrupt Sources.....	162
26.3. Capture/Compare Modules	163
26.3.1. Edge-triggered Capture Mode.....	164
26.3.2. Software Timer (Compare) Mode.....	165
26.3.3. High-Speed Output Mode	166
26.3.4. Frequency Output Mode	167
26.3.5. 8-bit Pulse Width Modulator Mode	168
26.3.6. 16-Bit Pulse Width Modulator Mode.....	169
26.4. Watchdog Timer Mode	170
26.4.1. Watchdog Timer Operation	170
26.4.2. Watchdog Timer Usage	171
26.5. Register Descriptions for PCA0.....	173
27. C2 Interface	178
27.1. C2 Interface Registers.....	178

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

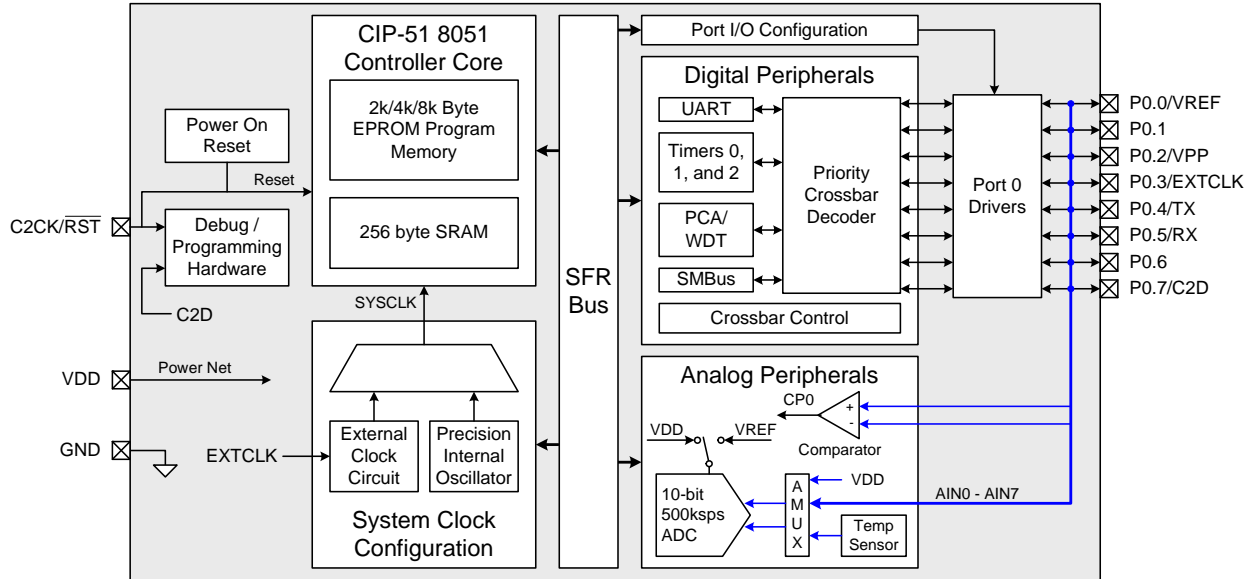


Figure 1.1. C8051T600/2/4 Block Diagram

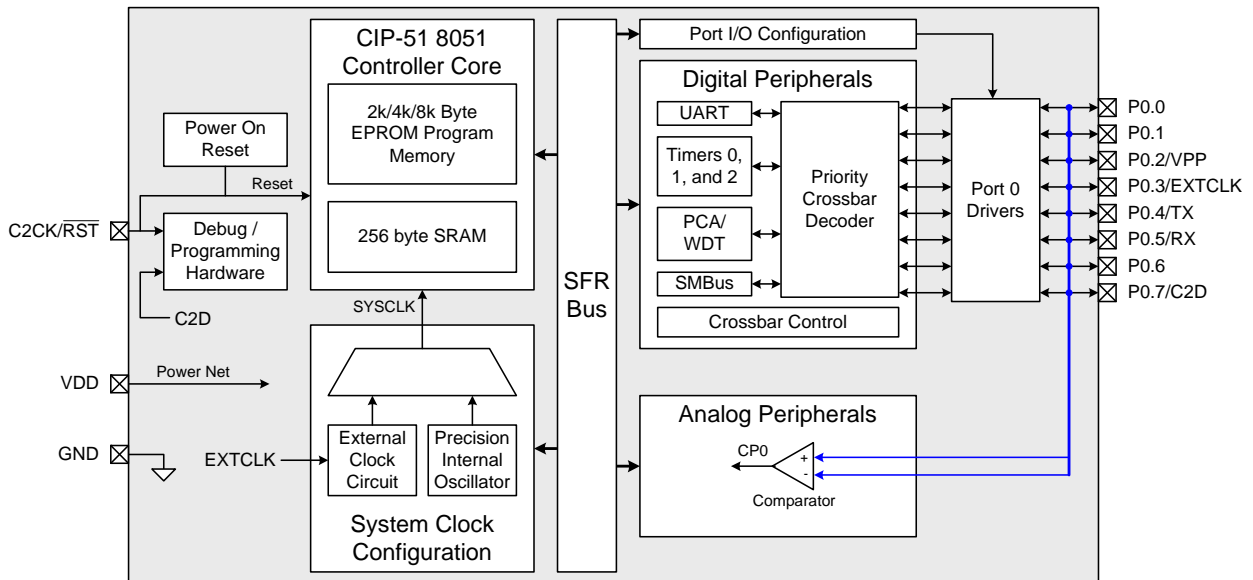


Figure 1.2. C8051T601/3/5 Block Diagram

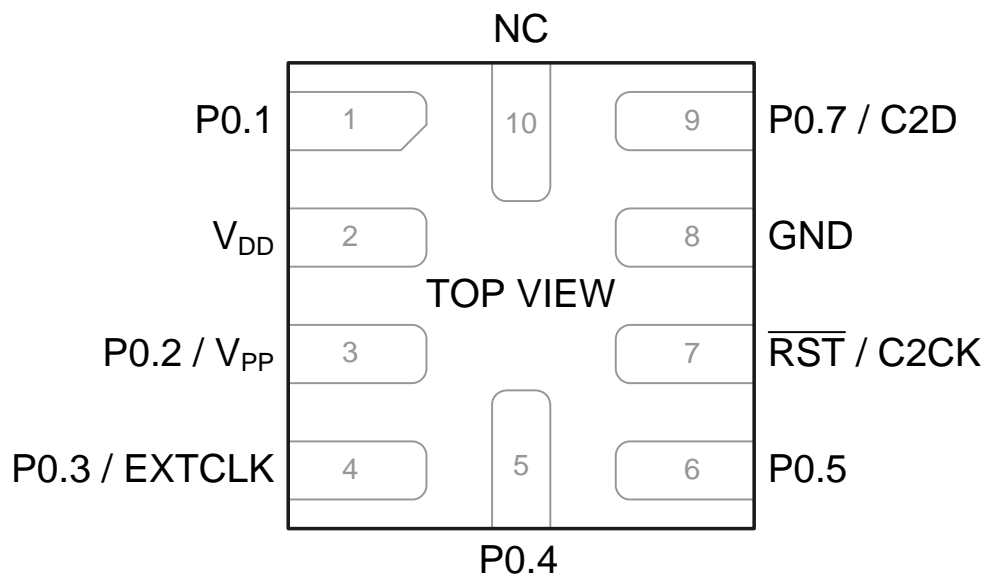


Figure 3.5. C8051T606-ZM QFN10 Pinout Diagram (Top View)

5. SOIC-14 Package Specifications

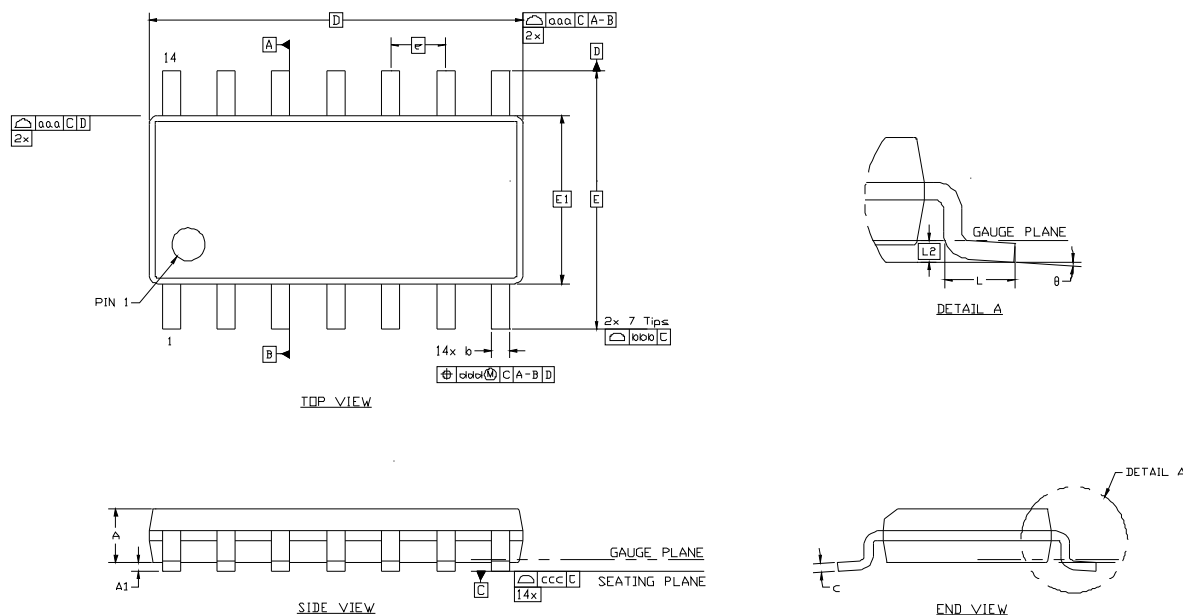


Figure 5.1. SOIC-14 Package Drawing

Table 5.1. SOIC-14 Package Dimensions

Dimension	Min	Nom	Max	Dimension	Min	Nom	Max
A	—	—	1.75	L	0.40	—	1.27
A1	0.10	—	0.25	L2	0.25 BSC		
b	0.33	—	0.51	θ	0°	—	8°
c	0.17	—	0.25	aaa	0.10		
D	8.65 BSC			bbb	0.20		
E	6.00 BSC			ccc	0.10		
E1	3.90 BSC			ddd	0.25		
e	1.27 BSC						

Notes:

1. All dimensions shown are in millimeters (mm).
2. Dimensioning and Tolerancing per ANSI Y14.5M-1994.
3. This drawing conforms to JEDEC outline MS012, variation AB.
4. Recommended card reflow profile is per the JEDEC/IPC J-STD-020 specification for Small Body Components.

12. Voltage Regulator (REG0)

C8051T600/1/2/3/4/5/6 devices include an internal voltage regulator (REG0) to regulate the internal core supply to 1.8 V from a V_{DD} supply of 1.8 to 3.6 V. Two power-saving modes are built into the regulator to help reduce current consumption in low-power applications. These modes are accessed through the REG0CN register (SFR Definition 12.1). Electrical characteristics for the on-chip regulator are specified in Table 8.5 on page 34.

If an external regulator is used to power the device, the internal regulator may be put into bypass mode using the BYPASS bit. **The internal regulator should never be placed in bypass mode unless an external 1.8 V regulator is used to supply V_{DD} . Doing so could cause permanent damage to the device.**

Under default conditions, when the device enters STOP mode the internal regulator will remain on. This allows any enabled reset source to generate a reset for the device and bring the device out of STOP mode. For additional power savings, the STOPCF bit can be used to shut down the regulator and the internal power network of the device when the part enters STOP mode. When STOPCF is set to 1, the RST pin or a full power cycle of the device are the only methods of generating a reset.

SFR Definition 13.2. CPT0MD: Comparator0 Mode Selection

Bit	7	6	5	4	3	2	1	0
Name							CP0MD[1:0]	
Type	R	R	R	R	R	R	R/W	
Reset	0	0	0	0	0	0	1	0

SFR Address = 0x9D

Bit	Name	Function
7:2	Unused	Unused. Read = 000000b, Write = Don't Care.
1:0	CP0MD[1:0]	Comparator0 Mode Select. These bits affect the response time and power consumption for Comparator0. 00: Mode 0 (Fastest Response Time, Highest Power Consumption) 01: Mode 1 10: Mode 2 11: Mode 3 (Slowest Response Time, Lowest Power Consumption)

14. CIP-51 Microcontroller

The MCU system controller core is the CIP-51 microcontroller. The CIP-51 is fully compatible with the MCS-51™ instruction set; standard 803x/805x assemblers and compilers can be used to develop software. The MCU family has a superset of all the peripherals included with a standard 8051. The CIP-51 also includes on-chip debug hardware (see description in Section 27), and interfaces directly with the analog and digital subsystems providing a complete data acquisition or control-system solution in a single integrated circuit.

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

- Fully Compatible with MCS-51 Instruction Set
- 25 MIPS Peak Throughput with 25 MHz Clock
- 0 to 25 MHz Clock Frequency
- Extended Interrupt Handler
- Reset Input
- Power Management Modes
- On-chip Debug Logic
- Program and Data Memory Security

Performance

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

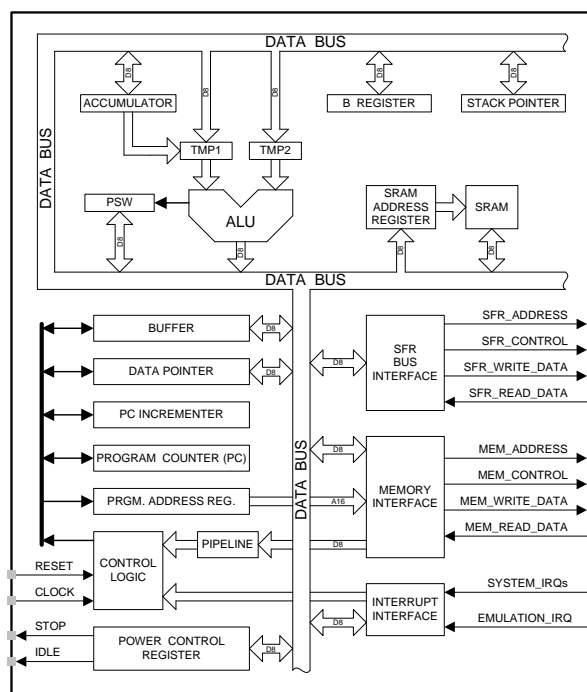


Figure 14.1. CIP-51 Block Diagram

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

Table 16.2. Special Function Registers (Continued)

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

Register	Address	Description	Page
PCA0H	0xFA	PCA Counter High	176
PCA0L	0xF9	PCA Counter Low	176
PCA0MD	0xD9	PCA Mode	174
PCON	0x87	Power Control	91
PSW	0xD0	Program Status Word	73
REF0CN	0xD1	Voltage Reference Control	56
REG0CN	0xC7	Voltage Regulator Control	58
RSTSRC	0xEF	Reset Source Configuration/Status	96
SBUF0	0x99	UART0 Data Buffer	143
SCON0	0x98	UART0 Control	142
SMB0CF	0xC1	SMBus Configuration	126
SMB0CN	0xC0	SMBus Control	128
SMB0DAT	0xC2	SMBus Data	130
SP	0x81	Stack Pointer	72
TCON	0x88	Timer/Counter Control	151
TH0	0x8C	Timer/Counter 0 High	154
TH1	0x8D	Timer/Counter 1 High	154
TL0	0x8A	Timer/Counter 0 Low	153
TL1	0x8B	Timer/Counter 1 Low	153
TMOD	0x89	Timer/Counter Mode	152
TMR2CN	0xC8	Timer/Counter 2 Control	157
TMR2H	0xCD	Timer/Counter 2 High	159
TMR2L	0xCC	Timer/Counter 2 Low	158
TMR2RLH	0xCB	Timer/Counter 2 Reload High	158
TMR2RLL	0xCA	Timer/Counter 2 Reload Low	158
TOFFH	0xA3	Temperature Sensor Offset Measurement High	54
TOFFL	0xA2	Temperature Sensor Offset Measurement Low	54
XBR0	0xE1	Port I/O Crossbar Control 0	115
XBR1	0xE2	Port I/O Crossbar Control 1	116
XBR2	0xE3	Port I/O Crossbar Control 2	117
All other SFR Locations		Reserved	

18. Power Management Modes

The C8051T600/1/2/3/4/5/6 devices have two software programmable power management modes: idle and stop. 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). 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 consumes the least power because the majority of the device is shut down with no clocks active. SFR Definition 18.1 describes the Power Control Register (PCON) used to control the C8051T600/1/2/3/4/5/6's stop and idle power management modes.

Although the C8051T600/1/2/3/4/5/6 has idle and stop 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.

18.1. 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.

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;           // set IDLE bit
PCON = PCON;           // ... followed by a 3-cycle dummy instruction

; in assembly:
ORL PCON, #01h          ; set IDLE bit
MOV PCON, PCON          ; ... 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 “19.6. PCA Watchdog Timer Reset” on page 94 for more information on the use and configuration of the WDT.

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

SFR Definition 21.2. OSCICN: Internal H-F Oscillator Control

Bit	7	6	5	4	3	2	1	0
Name				IFRDY	CLKSL	IOSCEN	IFCN[1:0]	
Type	R	R	R	R	R	R/W	R/W	
Reset	0	0	0	1	0	1	0	0

SFR Address = 0xB2

Bit	Name	Function
7:5	Unused	Unused. Read = 000b; Write = Don't Care
4	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.
3	CLKSL	System Clock Source Select Bit. 0: SYSCLK derived from the Internal Oscillator, and scaled as per the IFCN bits. 1: SYSCLK derived from the External Clock circuit.
2	IOSCEN	Internal H-F Oscillator Enable Bit. 0: Internal H-F Oscillator Disabled. 1: Internal H-F Oscillator Enabled.
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: SYSCLK derived from Internal H-F Oscillator divided by 2. 11: SYSCLK derived from Internal H-F Oscillator divided by 1.

Port	P0							
Pin	0	1	2	3	4	5	6	7
Special Function Signals	VREF			EXTCLK			CNVSTR	
TX0								
RX0								
SDA								
SCL								
CP0								
CP0A								
SYSCLK								
CEX0								
CEX1								
CEX2								
ECI								
T0								
T1								
Pin Skip Settings	0	0	0	0	0	0	0	x
	XBR0							

In this example, the crossbar is configured to assign the UART TX0 and RX0 signals, the SMBus signals, and the SYSCLK signal. Note that the SMBus signals are assigned as a pair, and there are no pins skipped using the XBR0 register.

■ These boxes represent the port pins which are used by the peripherals in this configuration.

1st TX0 is assigned to P0.4
 2nd RX0 is assigned to P0.5
 3rd SDA and SCL are assigned to P0.0 and P0.1, respectively.
 4th SYSCLK is assigned to P0.2

All unassigned pins can be used as GPIO or for other non-crossbar functions.

Figure 22.4. Priority Crossbar Decoder Example 1 - No Skipped Pins

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

SFR Definition 22.2. XBR1: Port I/O Crossbar Register 1

Bit	7	6	5	4	3	2	1	0
Name	PCA0ME[1:0]		CP0AE	CP0E	SYSCKE	SMB0E	URX0E	UTX0E
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

Bit	Name	Function
7:6	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.
5	CP0AE	Comparator0 Asynchronous Output Enable. 0: Asynchronous CP0 unavailable at Port pin. 1: Asynchronous CP0 routed to Port pin.
4	CP0E	Comparator0 Output Enable. 0: CP0 unavailable at Port pin. 1: CP0 routed to Port pin.
3	SYSCKE	/SYSCLK Output Enable. 0: /SYSCLK unavailable at Port pin. 1: /SYSCLK output routed to Port pin.
2	SMB0E	SMBus I/O Enable. 0: SMBus I/O unavailable at Port pins. 1: SMBus I/O (SDA, SCL) routed to Port pins.
1	URX0E	UART RX Input Enable. 0: UART RX unavailable at Port pin. 1: UART RX0 routed to Port pin P0.5.
0	UTX0E	UART TX Output Enable. 0: UART TX0 unavailable at Port pin. 1: UART TX0 routed to Port pin P0.4.

23.1. Supporting Documents

It is assumed the reader is familiar with or has access to the following supporting documents:

1. The I²C-Bus and How to Use It (including specifications), Philips Semiconductor.
2. The I²C-Bus Specification—Version 2.0, Philips Semiconductor.
3. System Management Bus Specification—Version 1.1, SBS Implementers Forum.

23.2. SMBus Configuration

Figure 23.2 shows a typical SMBus configuration. The SMBus specification allows any recessive voltage between 3.0 V and 5.0 V; different devices on the bus may operate at different voltage levels. The bi-directional SCL (serial clock) and SDA (serial data) lines must be connected to a positive power supply voltage through a pullup resistor or similar circuit. Every device connected to the bus must have an open-drain or open-collector output for both the SCL and SDA lines, so that both are pulled high (recessive state) when the bus is free. The maximum number of devices on the bus is limited only by the requirement that the rise and fall times on the bus not exceed 300 ns and 1000 ns, respectively.

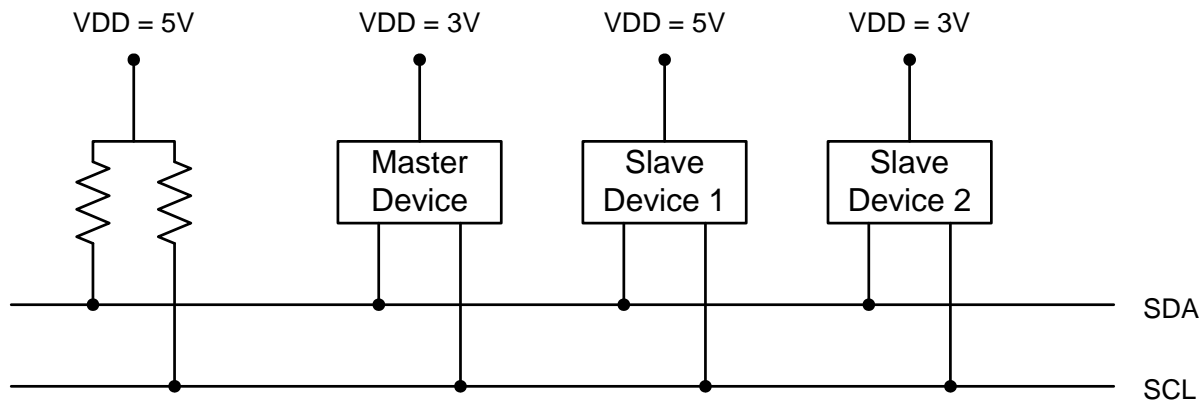


Figure 23.2. Typical SMBus Configuration

23.3. SMBus Operation

Two types of data transfers are possible: data transfers from a master transmitter to an addressed slave receiver (WRITE), and data transfers from an addressed slave transmitter to a master receiver (READ). The master device initiates both types of data transfers and provides the serial clock pulses on SCL. The SMBus interface may operate as a master or a slave, and multiple master devices on the same bus are supported. If two or more masters attempt to initiate a data transfer simultaneously, an arbitration scheme is employed with a single master always winning the arbitration. Note that it is not necessary to specify one device as the Master in a system; any device that transmits a START and a slave address becomes the master for the duration of that transfer.

A typical SMBus transaction consists of a START condition followed by an address byte (Bits7–1: 7-bit slave address; Bit0: R/W direction bit), one or more bytes of data, and a STOP condition. Bytes that are received (by a master or slave) are acknowledged (ACK) with a low SDA during a high SCL (see Figure 23.3). If the receiving device does not ACK, the transmitting device will read a NACK (not acknowledge), which is a high SDA during a high SCL.

The direction bit (R/W) occupies the least-significant bit position of the address byte. The direction bit is set to logic 1 to indicate a "READ" operation and cleared to logic 0 to indicate a "WRITE" operation.

24.2. Operational Modes

UART0 provides standard asynchronous, full duplex communication. The UART mode (8-bit or 9-bit) is selected by the S0MODE bit (SCON0.7). Typical UART connection options are shown in Figure 24.3.

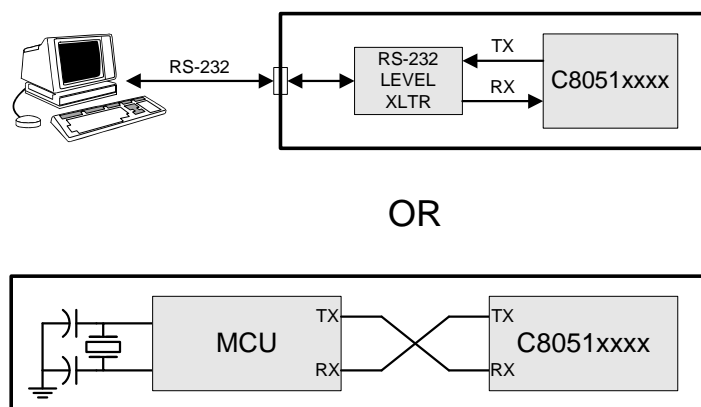


Figure 24.3. UART Interconnect Diagram

24.2.1. 8-Bit UART

8-Bit UART mode uses a total of 10 bits per data byte: one start bit, eight data bits (LSB first), and one stop bit. Data are transmitted LSB first from the TX0 pin and received at the RX0 pin. On receive, the eight data bits are stored in SBUF0 and the stop bit goes into RB80 (SCON0.2).

Data transmission begins when software writes a data byte to the SBUF0 register. The TI0 Transmit Interrupt Flag (SCON0.1) is set at the end of the transmission (the beginning of the stop-bit time). Data reception can begin any time after the REN0 Receive Enable bit (SCON0.4) is set to logic 1. After the stop bit is received, the data byte will be loaded into the SBUF0 receive register if the following conditions are met: RI0 must be logic 0, and if MCE0 is logic 1, the stop bit must be logic 1. In the event of a receive data overrun, the first received 8 bits are latched into the SBUF0 receive register and the following overrun data bits are lost.

If these conditions are met, the eight bits of data is stored in SBUF0, the stop bit is stored in RB80 and the RI0 flag is set. If these conditions are not met, SBUF0 and RB80 will not be loaded and the RI0 flag will not be set. An interrupt will occur if enabled when either TI0 or RI0 is set.

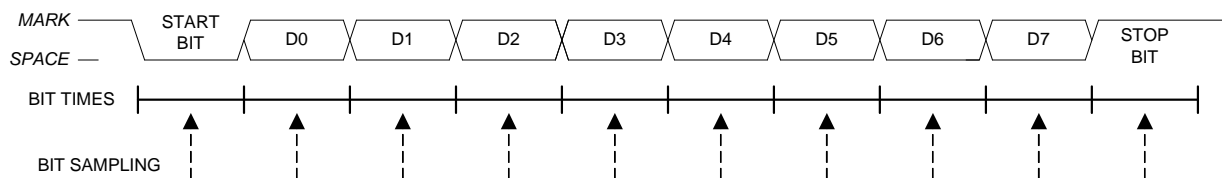


Figure 24.4. 8-Bit UART Timing Diagram

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

25.1. Timer 0 and Timer 1

Each timer is implemented as a 16-bit register accessed as two separate bytes: a low byte (TL0 or TL1) and a high byte (TH0 or TH1). The Counter/Timer Control register (TCON) is used to enable Timer 0 and Timer 1 as well as indicate status. Timer 0 interrupts can be enabled by setting the ET0 bit in the IE register (Section “17.2. Interrupt Register Descriptions” on page 82); Timer 1 interrupts can be enabled by setting the ET1 bit in the IE register (Section “17.2. Interrupt Register Descriptions” on page 82). Both counter/timers operate in one of four primary modes selected by setting the Mode Select bits T1M1–T0M0 in the Counter/Timer Mode register (TMOD). Each timer can be configured independently. Each operating mode is described below.

25.1.1. Mode 0: 13-bit Counter/Timer

Timer 0 and Timer 1 operate as 13-bit counter/timers in Mode 0. The following describes the configuration and operation of Timer 0. However, both timers operate identically, and Timer 1 is configured in the same manner as described for Timer 0.

The TH0 register holds the eight MSBs of the 13-bit counter/timer. TL0 holds the five LSBs in bit positions TL0.4–TL0.0. The three upper bits of TL0 (TL0.7–TL0.5) are indeterminate and should be masked out or ignored when reading. As the 13-bit timer register increments and overflows from 0x1FFF (all ones) to 0x0000, the timer overflow flag TF0 in TCON is set and an interrupt will occur if Timer 0 interrupts are enabled.

The C/T0 bit in the TMOD register selects the counter/timer's clock source. When C/T0 is set to logic 1, high-to-low transitions at the selected Timer 0 input pin (T0) increment the timer register (refer to Section “22.3. Priority Crossbar Decoder” on page 111 for information on selecting and configuring external I/O pins). Clearing C/T selects the clock defined by the T0M bit in register CKCON. When T0M is set, Timer 0 is clocked by the system clock. When T0M is cleared, Timer 0 is clocked by the source selected by the Clock Scale bits in CKCON (see SFR Definition 25.1).

Setting the TR0 bit (TCON.4) enables the timer when either GATE0 in the TMOD register is logic 0 or the input signal INT0 is active as defined by bit IN0PL in register IT01CF (see SFR Definition 17.5). Setting GATE0 to 1 allows the timer to be controlled by the external input signal INT0 (see Section “17.2. Interrupt Register Descriptions” on page 82), facilitating pulse width measurements

TR0	GATE0	INT0	Counter/Timer
0	X	X	Disabled
1	0	X	Enabled
1	1	0	Disabled
1	1	1	Enabled
Note: X = Don't Care			

Setting TR0 does not force the timer to reset. The timer registers should be loaded with the desired initial value before the timer is enabled.

TL1 and TH1 form the 13-bit register for Timer 1 in the same manner as described above for TL0 and TH0. Timer 1 is configured and controlled using the relevant TCON and TMOD bits just as with Timer 0. The input signal INT0 is used with Timer 1; the /INT1 polarity is defined by bit IN1PL in register IT01CF (see SFR Definition 17.5).

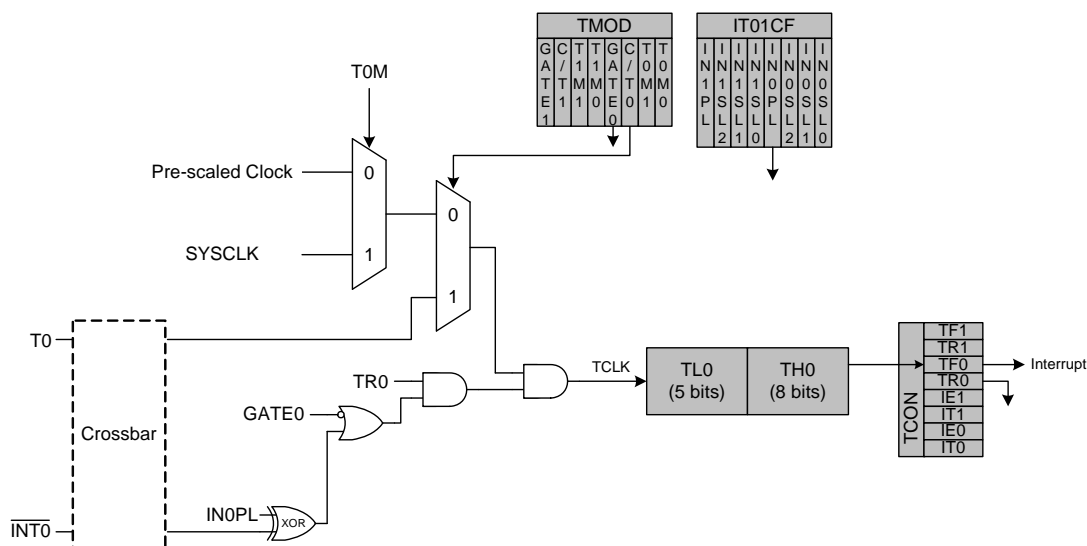


Figure 25.1. T0 Mode 0 Block Diagram

25.1.2. Mode 1: 16-bit Counter/Timer

Mode 1 operation is the same as Mode 0, except that the counter/timer registers use all 16 bits. The counter/timers are enabled and configured in Mode 1 in the same manner as for Mode 0.

SFR Definition 25.3. 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]	
Type	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 $\overline{\text{INT1}}$ logic level. 1: Timer 1 enabled only when TR1 = 1 AND $\overline{\text{INT1}}$ is active as defined by bit IN1PL in register IT01CF (see SFR Definition 17.5).
6	C/T1	Counter/Timer 1 Select. 0: Timer: Timer 1 incremented by clock defined by T1M bit in register CKCON. 1: Counter: Timer 1 incremented by high-to-low transitions on external pin (T1).
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 $\overline{\text{INT0}}$ logic level. 1: Timer 0 enabled only when TR0 = 1 AND $\overline{\text{INT0}}$ is active as defined by bit IN0PL in register IT01CF (see SFR Definition 17.5).
2	C/T0	Counter/Timer 0 Select. 0: Timer: Timer 0 incremented by clock defined by T0M bit in register CKCON. 1: Counter: Timer 0 incremented by high-to-low transitions on external pin (T0).
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 Counters/Timers

SFR Definition 25.6. TH0: Timer 0 High Byte

Bit	7	6	5	4	3	2	1	0
Name	TH0[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x8C

Bit	Name	Function
7:0	TH0[7:0]	Timer 0 High Byte. The TH0 register is the high byte of the 16-bit Timer 0.

SFR Definition 25.7. TH1: Timer 1 High Byte

Bit	7	6	5	4	3	2	1	0
Name	TH1[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x8D

Bit	Name	Function
7:0	TH1[7:0]	Timer 1 High Byte. The TH1 register is the high byte of the 16-bit Timer 1.

26.3.2. Software Timer (Compare) Mode

In Software Timer mode, the PCA counter/timer value is compared to the module's 16-bit Capture/Compare register (PCA0CPHn and PCA0CPLn). When a match occurs, the Capture/Compare Flag (CCFn) in PCA0CN is set to logic 1. An interrupt request is generated if the CCFn interrupt for that module is enabled. The CCFn bit is not automatically cleared by hardware when the CPU vectors to the interrupt service routine, and must be cleared by software. Setting the ECOMn and MATn bits in the PCA0CPMn register enables Software Timer mode.

Important Note about Capture/Compare Registers: When writing a 16-bit value to the PCA0 Capture/Compare registers, the low byte should always be written first. Writing to PCA0CPLn clears the ECOMn bit to 0; writing to PCA0CPHn sets ECOMn to 1.

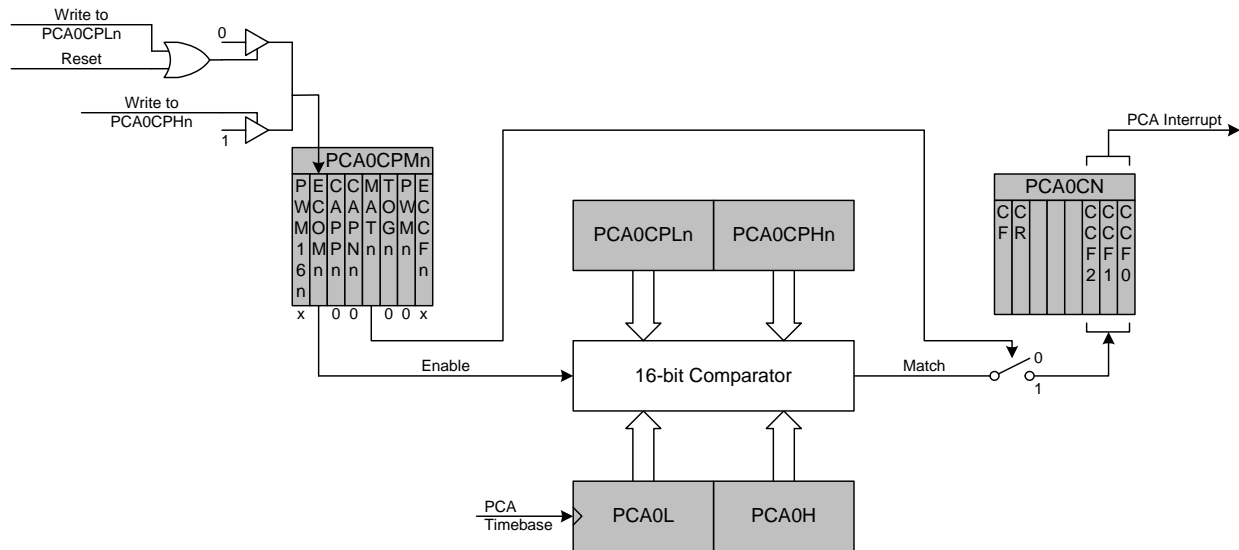


Figure 26.5. PCA Software Timer Mode Diagram