

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 <sup>2</sup> C), SPI, UART/USART
Peripherals	Cap Sense, POR, PWM, WDT
Number of I/O	39
Program Memory Size	15KB (15K x 8)
Program Memory Type	FLASH
EEPROM Size	32 x 8
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	48-VFQFN Exposed Pad
Supplier Device Package	48-QFN (7x7)
Purchase URL	<a href="https://www.e-xfl.com/product-detail/silicon-labs/c8051f705-gmr">https://www.e-xfl.com/product-detail/silicon-labs/c8051f705-gmr</a>

## List of Figures

Figure 1.1. C8051F700/1 Block Diagram .....	18
Figure 1.2. C8051F702/3 Block Diagram .....	19
Figure 1.3. C8051F704/5 Block Diagram .....	20
Figure 1.4. C8051F706/07 Block Diagram .....	21
Figure 1.5. C8051F708/09/10/11 Block Diagram .....	22
Figure 1.6. C8051F712/13/14/15 Block Diagram .....	23
Figure 1.7. C8051F716 Block Diagram .....	24
Figure 1.8. C8051F717 Block Diagram .....	25
Figure 3.1. C8051F7xx-GQ TQFP64 Pinout Diagram (Top View) .....	32
Figure 3.2. C8051F7xx-GQ QFP48 Pinout Diagram (Top View) .....	33
Figure 3.3. C8051F7xx-GM QFN48 Pinout Diagram (Top View) .....	34
Figure 3.4. C8051F716-GM QFN32 Pinout Diagram (Top View) .....	35
Figure 3.5. C8051F717-GM QFN24 Pinout Diagram (Top View) .....	36
Figure 4.1. TQFP-64 Package Drawing .....	37
Figure 4.2. TQFP-64 PCB Land Pattern .....	38
Figure 5.1. TQFP-48 Package Drawing .....	39
Figure 5.2. TQFP-48 PCB Land Pattern .....	40
Figure 6.1. QFN-48 Package Drawing .....	41
Figure 6.2. QFN-48 PCB Land Pattern .....	42
Figure 7.1. QFN-32 Package Drawing .....	43
Figure 7.2. QFN-32 Recommended PCB Land Pattern .....	44
Figure 8.1. QFN-24 Package Drawing .....	45
Figure 8.2. QFN-24 Recommended PCB Land Pattern .....	46
Figure 10.1. ADC0 Functional Block Diagram .....	55
Figure 10.2. 10-Bit ADC Track and Conversion Example Timing .....	57
Figure 10.3. ADC0 Equivalent Input Circuits .....	58
Figure 10.4. ADC Window Compare Example: Right-Justified Data .....	64
Figure 10.5. ADC Window Compare Example: Left-Justified Data .....	64
Figure 10.6. ADC0 Multiplexer Block Diagram .....	65
Figure 11.1. Temperature Sensor Transfer Function .....	67
Figure 11.2. Temperature Sensor Error with 1-Point Calibration at 0 Celsius .....	68
Figure 12.1. Voltage Reference Functional Block Diagram .....	69
Figure 14.1. Comparator0 Functional Block Diagram .....	74
Figure 14.2. Comparator Hysteresis Plot .....	75
Figure 14.3. Comparator Input Multiplexer Block Diagram .....	78
Figure 15.1. CS0 Block Diagram .....	80
Figure 15.2. Auto-Scan Example .....	83
Figure 15.3. CS0 Multiplexer Block Diagram .....	96
Figure 16.1. CIP-51 Block Diagram .....	98
Figure 17.1. C8051F70x/71x Memory Map .....	108
Figure 17.2. Flash Program Memory Map .....	109
Figure 18.1. Multiplexed Configuration Example .....	115
Figure 18.2. Non-multiplexed Configuration Example .....	116

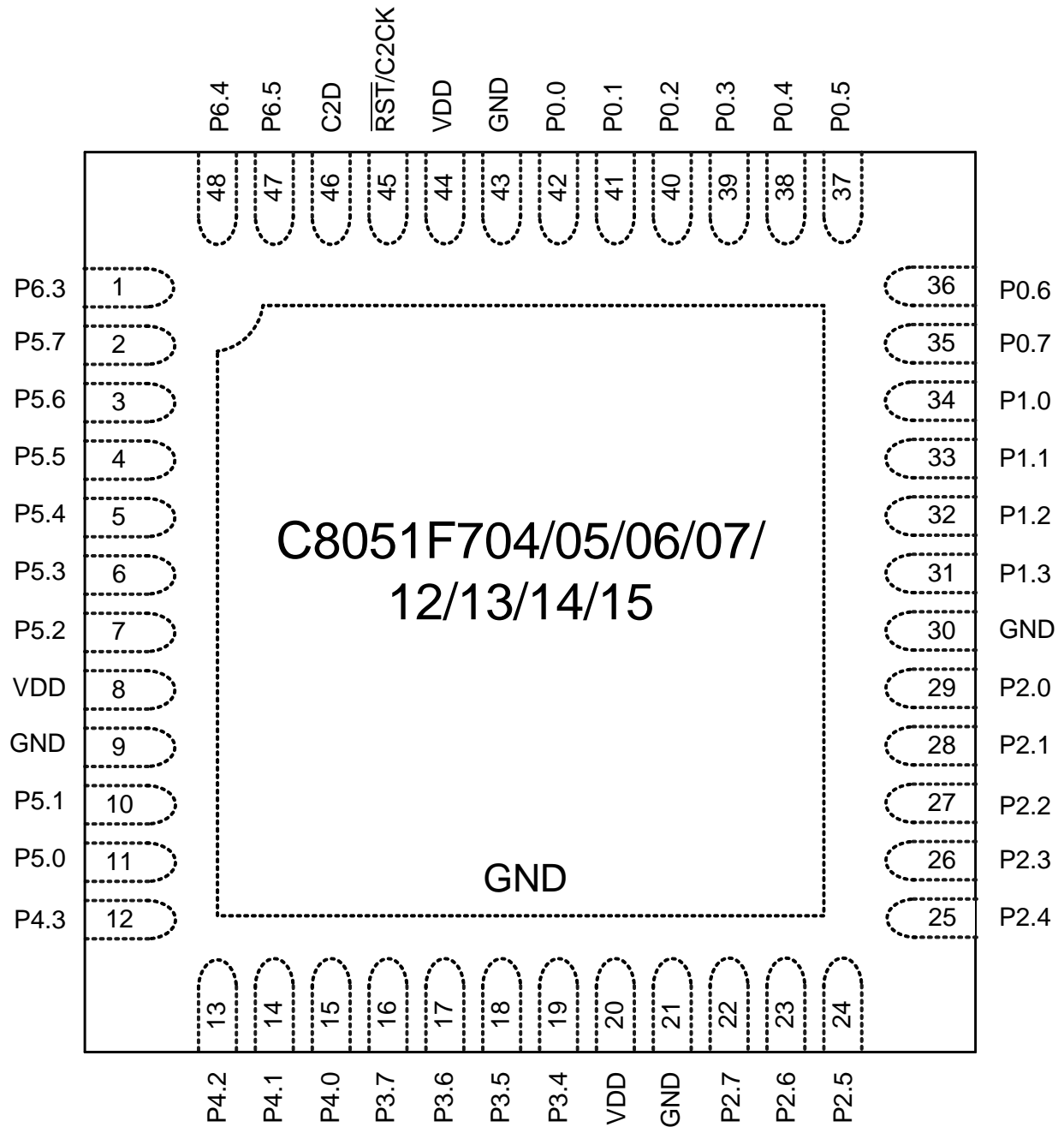
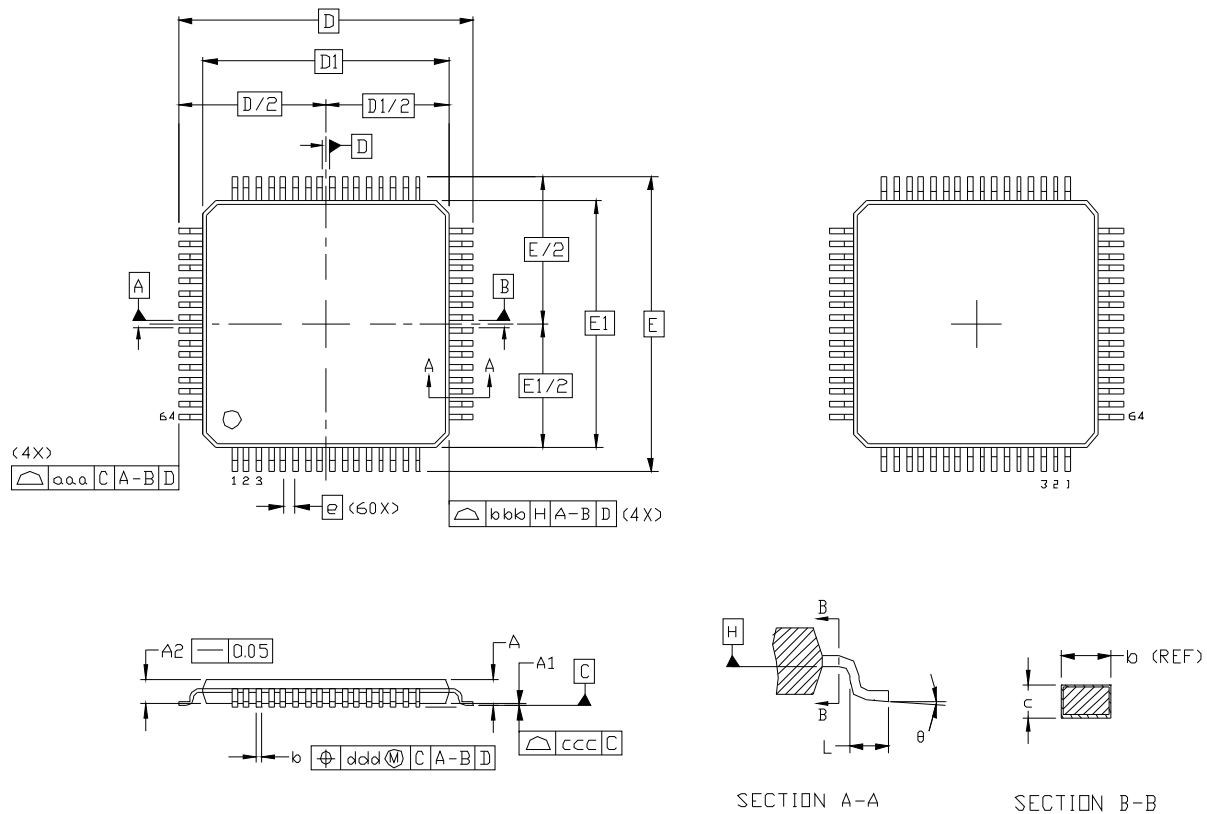


Figure 3.3. C8051F7xx-GM QFN48 Pinout Diagram (Top View)

## 4. TQFP-64 Package Specifications



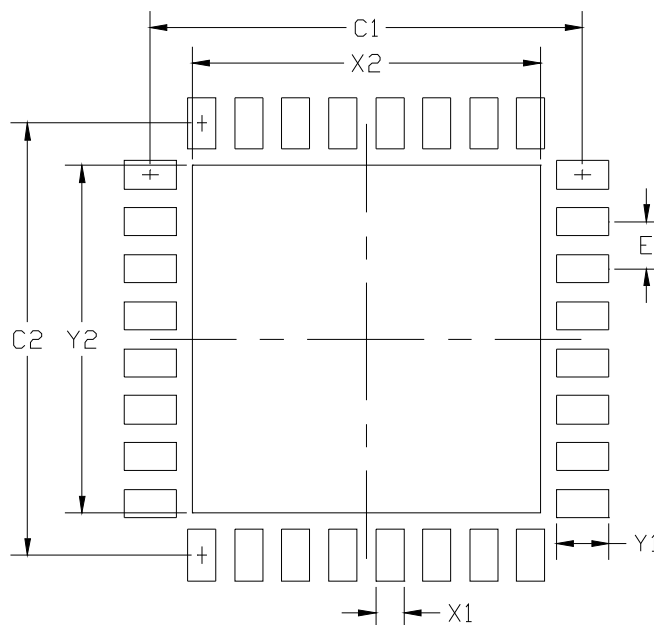
**Figure 4.1. TQFP-64 Package Drawing**

**Table 4.1. TQFP-64 Package Dimensions**

Dimension	Min	Nom	Max	Dimension	Min	Nom	Max
A	—	—	1.20	E	12.00 BSC.		
A1	0.05	—	0.15	E1	10.00 BSC.		
A2	0.95	1.00	1.05	L	0.45	0.60	0.75
b	0.17	0.22	0.27	aaa	—	—	0.20
c	0.09	—	0.20	bbb	—	—	0.20
D	12.00 BSC.			ccc	—	—	0.08
D1	10.00 BSC.			ddd	—	—	0.08
e	0.50 BSC.			θ	0°	3.5°	7°

**Notes:**

1. All dimensions shown are in millimeters (mm) unless otherwise noted.
2. Dimensioning and Tolerancing per ANSI Y14.5M-1994.
3. This package outline conforms to JEDEC MS-026, variant ACD.
4. Recommended card reflow profile is per the JEDEC/IPC J-STD-020 specification for Small Body Components.



**Figure 7.2. QFN-32 Recommended PCB Land Pattern**

**Table 7.2. QFN-32 PCB Land Pattern Dimensions**

Dimension	Min	Max	Dimension	Min	Max
C1	4.60		X2	3.60	3.70
C2	4.60		Y1	0.45	0.55
E	0.50		Y2	3.60	3.70
X1	0.20	0.30			

**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  $\mu$ m 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 3x3 array of 1.0 mm openings on a 1.25 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.

# C8051F70x/71x

## SFR Definition 13.1. REG0CN: Voltage Regulator Control

Bit	7	6	5	4	3	2	1	0
Name	STOPCF	BYPASS						
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 = 0xB9; SFR Page = F

Bit	Name	Function
7	STOPCF	<b>Stop Mode Configuration.</b> This bit configures the regulator's behavior when the device enters STOP mode. 0: Regulator is still active in STOP mode. Any enabled reset source will reset the device. 1: Regulator is shut down in STOP mode. Only the $\overline{\text{RST}}$ pin or power cycle can reset the device.
6	BYPASS	<b>Bypass Internal Regulator.</b> This bit places the regulator in bypass mode, allowing the core to run directly from the $V_{DD}$ supply pin. 0: Normal Mode—Regulator is on and regulates $V_{DD}$ down to the core voltage. 1: Bypass Mode—Regulator is in bypass mode, and the microcontroller core operates directly from the $V_{DD}$ supply voltage. <b>IMPORTANT: Bypass mode is for use with an external regulator as the supply voltage only. Never place the regulator in bypass mode when the <math>V_{DD}</math> supply voltage is greater than the specifications given in Table 9.1 on page 47. Doing so may cause permanent damage to the device.</b>
5:0	Reserved	Reserved. Must Write 000000b.

## 16. CIP-51 Microcontroller

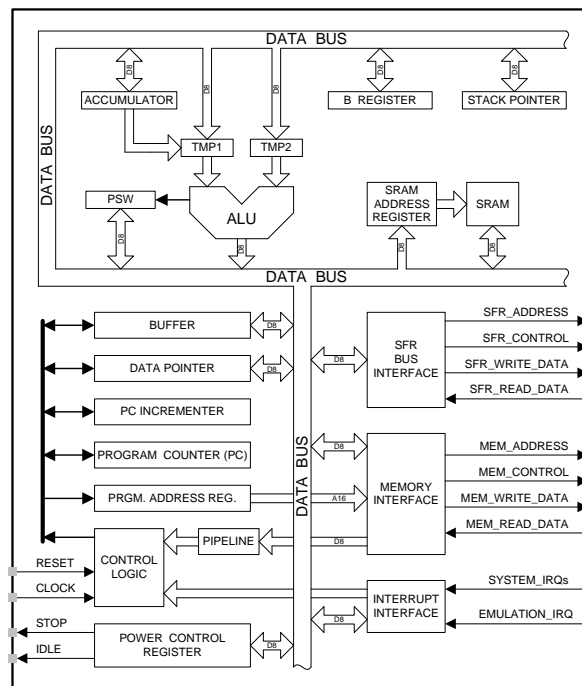
The MCU system controller core is the CIP-51 microcontroller. The CIP-51 is fully compatible with the MCS-51™ instruction set; standard 803x/805x assemblers and compilers can be used to develop software. The MCU family has a superset of all the peripherals included with a standard 8051. The CIP-51 also includes on-chip debug hardware (see description in “C2 Interface” on page 301), 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 16.1 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 16.1. CIP-51 Block Diagram**

## 18. External Data Memory Interface and On-Chip XRAM

For C8051F70x/71x devices, 256 B of RAM are included on-chip and mapped into the external data memory space (XRAM). Additionally, an External Memory Interface (EMIF) is available on the C8051F700/1/2/3/8/9 and C8051F710/1 devices, which can be used to access off-chip data memories and memory-mapped devices connected to the GPIO ports. The external memory space may be accessed using the external move instruction (MOVX) and the data pointer (DPTR), or using the MOVX indirect addressing mode using R0 or R1. If the MOVX instruction is used with an 8-bit address operand (such as @R1), then the high byte of the 16-bit address is provided by the External Memory Interface Control Register (EMI0CN, shown in SFR Definition 18.1).

**Note:** The MOVX instruction can also be used for writing to the Flash memory. See Section “22. Flash Memory” on page 148 for details. The MOVX instruction accesses XRAM by default.

### 18.1. Accessing XRAM

The XRAM memory space is accessed using the MOVX instruction. The MOVX instruction has two forms, both of which use an indirect addressing method. The first method uses the Data Pointer, DPTR, a 16-bit register which contains the effective address of the XRAM location to be read from or written to. The second method uses R0 or R1 in combination with the EMI0CN register to generate the effective XRAM address. Examples of both of these methods are given below.

#### 18.1.1. 16-Bit MOVX Example

The 16-bit form of the MOVX instruction accesses the memory location pointed to by the contents of the DPTR register. The following series of instructions reads the value of the byte at address 0x1234 into the accumulator A:

```
MOV    DPTR, #1234h      ; load DPTR with 16-bit address to read (0x1234)
MOVX   A, @DPTR          ; load contents of 0x1234 into accumulator A
```

The above example uses the 16-bit immediate MOV instruction to set the contents of DPTR. Alternately, the DPTR can be accessed through the SFR registers DPH, which contains the upper 8-bits of DPTR, and DPL, which contains the lower 8-bits of DPTR.

#### 18.1.2. 8-Bit MOVX Example

The 8-bit form of the MOVX instruction uses the contents of the EMI0CN SFR to determine the upper 8-bits of the effective address to be accessed and the contents of R0 or R1 to determine the lower 8-bits of the effective address to be accessed. The following series of instructions read the contents of the byte at address 0x1234 into the accumulator A.

```
MOV    EMI0CN, #12h      ; load high byte of address into EMI0CN
MOV    R0, #34h          ; load low byte of address into R0 (or R1)
MOVX   a, @R0            ; load contents of 0x1234 into accumulator A
```



## 18.4.2. Non-multiplexed Configuration

In Non-multiplexed mode, the Data Bus and the Address Bus pins are not shared. An example of a Non-multiplexed Configuration is shown in Figure 18.2. See Section “18.6.1. Non-Multiplexed Mode” on page 120 for more information about Non-multiplexed operation.

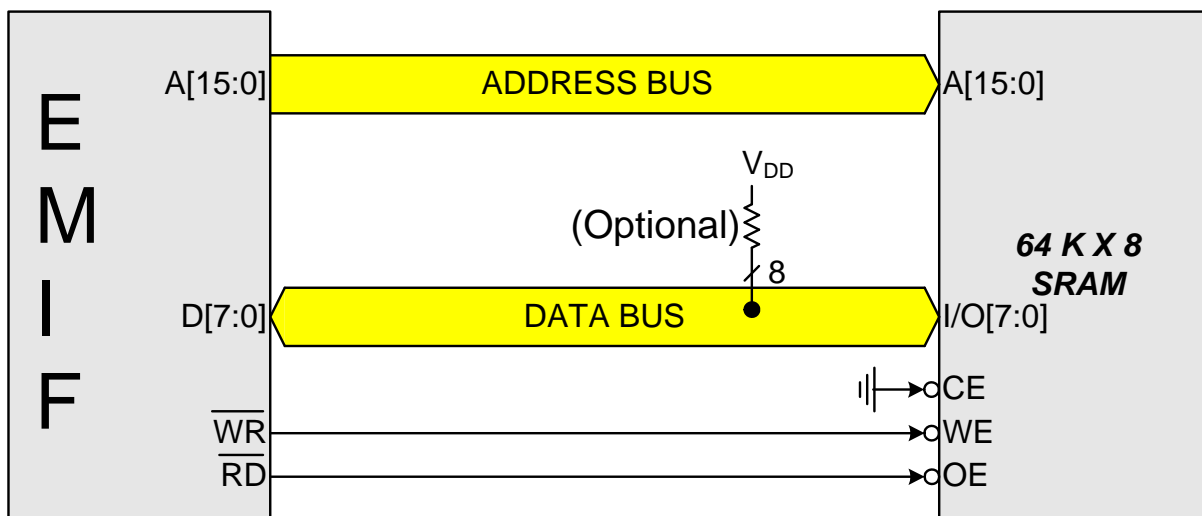


Figure 18.2. Non-multiplexed Configuration Example

**Table 21.1. Interrupt Summary**

Interrupt Source	Interrupt Vector	Priority Order	Pending Flag	Bit addressable?	Cleared by HW?	Enable Flag	Priority Control
Reset	0x0000	Top	None	N/A	N/A	Always Enabled	Always Highest
External Interrupt 0 (INT0)	0x0003	0	IE0 (TCON.1)	Y	Y	EX0 (IE.0)	PX0 (IP.0)
Timer 0 Overflow	0x000B	1	TF0 (TCON.5)	Y	Y	ET0 (IE.1)	PT0 (IP.1)
External Interrupt 1 (INT1)	0x0013	2	IE1 (TCON.3)	Y	Y	EX1 (IE.2)	PX1 (IP.2)
Timer 1 Overflow	0x001B	3	TF1 (TCON.7)	Y	Y	ET1 (IE.3)	PT1 (IP.3)
UART0	0x0023	4	RI0 (SCON0.0) TI0 (SCON0.1)	Y	N	ES0 (IE.4)	PS0 (IP.4)
Timer 2 Overflow	0x002B	5	TF2H (TMR2CN.7) TF2L (TMR2CN.6)	Y	N	ET2 (IE.5)	PT2 (IP.5)
SPI0	0x0033	6	SPIF (SPI0CN.7) WCOL (SPI0CN.6) MODF (SPI0CN.5) RXOVRN (SPI0CN.4)	Y		ESPI0 (IE.6)	PSPI0 (IP.6)
SMB0	0x003B	7	SI (SMB0CN.0)	Y	N	ESMB0 (EIE1.0)	PSMB0 (EIP1.0)
Port Match	0x0043	8	None	N/A	N/A	EMAT (EIE1.1)	PMAT (EIP1.1)
ADC0 Window Compare	0x004B	9	AD0WINT (ADC0CN.3)	Y	N	EWADC0 (EIE1.2)	PWADC0 (EIP1.2)
ADC0 Conversion Complete	0x0053	10	AD0INT (ADC0CN.5)	Y	N	EADC0 (EIE1.3)	PADC0 (EIP1.3)
Programmable Counter Array	0x005B	11	CF (PCA0CN.7) CCFn (PCA0CN.n)	Y	N	EPCA0 (EIE1.4)	PPCA0 (EIP1.4)
Comparator0	0x0063	12	CP0FIF (CPT0CN.4) CP0RIF (CPT0CN.5)	N	N	ECP0 (EIE1.5)	PCP0 (EIP1.5)
RESERVED							
Timer 3 Overflow	0x0073	14	TF3H (TMR3CN.7) TF3L (TMR3CN.6)	N	N	ET3 (EIE1.7)	PT3 (EIP1.7)
CS0 Conversion Complete	0x007B	15	CS0INT (CS0CN.5)	N	N	ECSCPT (EIE2.0)	PSCCPT (EIP2.0)
CS0 Greater Than Compare	0x0083	16	CS0CMPF (CS0CN.0)	N	N	ECSGRT (EIE2.1)	PSCGRT (EIP2.1)

## SFR Definition 21.3. EIE1: Extended Interrupt Enable 1

Bit	7	6	5	4	3	2	1	0
Name	ET3	Reserved	ECP0	EPCA0	EADC0	EWADC0	EMAT	ESMB0
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 = 0xE6; SFR Page = All Pages

Bit	Name	Function
7	ET3	<b>Enable Timer 3 Interrupt.</b> This bit sets the masking of the Timer 3 interrupt. 0: Disable Timer 3 interrupt. 1: Enable interrupt requests generated by the TF3L or TF3H flags.
6	Reserved	Must write 0.
5	ECP0	<b>Enable Comparator0 (CP0) Interrupt.</b> This bit sets the masking of the CP0 rising edge or falling edge interrupt. 0: Disable CP0 interrupts. 1: Enable interrupt requests generated by the CP0RIF and CP0FIF flags.
4	EPCA0	<b>Enable Programmable Counter Array (PCA0) Interrupt.</b> This bit sets the masking of the PCA0 interrupts. 0: Disable all PCA0 interrupts. 1: Enable interrupt requests generated by PCA0.
3	EADC0	<b>Enable ADC0 Conversion Complete Interrupt.</b> This bit sets the masking of the ADC0 Conversion Complete interrupt. 0: Disable ADC0 Conversion Complete interrupt. 1: Enable interrupt requests generated by the AD0INT flag.
2	EWADC0	<b>Enable Window Comparison ADC0 interrupt.</b> This bit sets the masking of ADC0 Window Comparison interrupt. 0: Disable ADC0 Window Comparison interrupt. 1: Enable interrupt requests generated by ADC0 Window Compare flag (AD0WINT).
1	EMAT	<b>Enable Port Match Interrupts.</b> This bit sets the masking of the Port Match event interrupt. 0: Disable all Port Match interrupts. 1: Enable interrupt requests generated by a Port Match.
0	ESMB0	<b>Enable SMBus (SMB0) Interrupt.</b> This bit sets the masking of the SMB0 interrupt. 0: Disable all SMB0 interrupts. 1: Enable interrupt requests generated by SMB0.

## SFR Definition 21.5. EIP1: Extended Interrupt Priority 1

Bit	7	6	5	4	3	2	1	0
Name	PT3	Reserved	PCP0	PPCA0	PADC0	PWADC0	PMAT	PSMB0
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 = 0xCE; SFR Page = F

Bit	Name	Function
7	PT3	<b>Timer 3 Interrupt Priority Control.</b> This bit sets the priority of the Timer 3 interrupt. 0: Timer 3 interrupt set to low priority level. 1: Timer 3 interrupt set to high priority level.
6	Reserved	Must write 0b.
5	PCP0	<b>Comparator0 (CP0) Interrupt Priority Control.</b> This bit sets the priority of the CP0 rising edge or falling edge interrupt. 0: CP0 interrupt set to low priority level. 1: CP0 interrupt set to high priority level.
4	PPCA0	<b>Programmable Counter Array (PCA0) Interrupt Priority Control.</b> This bit sets the priority of the PCA0 interrupt. 0: PCA0 interrupt set to low priority level. 1: PCA0 interrupt set to high priority level.
3	PADC0	<b>ADC0 Conversion Complete Interrupt Priority Control.</b> This bit sets the priority of the ADC0 Conversion Complete interrupt. 0: ADC0 Conversion Complete interrupt set to low priority level. 1: ADC0 Conversion Complete interrupt set to high priority level.
2	PWADC0	<b>ADC0 Window Comparator Interrupt Priority Control.</b> This bit sets the priority of the ADC0 Window interrupt. 0: ADC0 Window interrupt set to low priority level. 1: ADC0 Window interrupt set to high priority level.
1	PMAT	<b>Port Match Interrupt Priority Control.</b> This bit sets the priority of the Port Match Event interrupt. 0: Port Match interrupt set to low priority level. 1: Port Match interrupt set to high priority level.
0	PSMB0	<b>SMBus (SMB0) Interrupt Priority Control.</b> This bit sets the priority of the SMB0 interrupt. 0: SMB0 interrupt set to low priority level. 1: SMB0 interrupt set to high priority level.

## SFR Definition 22.1. PSCTL: Program Store R/W Control

Bit	7	6	5	4	3	2	1	0
Name							PSEE	PSWE
Type	R	R	R	R	R	R	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x8F; SFR Page = All Pages

Bit	Name	Function
7:2	Unused	Read = 000000b, Write = don't care.
1	PSEE	<b>Program Store Erase Enable.</b> Setting this bit (in combination with PSWE) allows an entire page of Flash program memory to be erased. If this bit is logic 1 and Flash writes are enabled (PSWE is logic 1), a write to Flash memory using the MOVX instruction will erase the entire page that contains the location addressed by the MOVX instruction. The value of the data byte written does not matter. 0: Flash program memory erasure disabled. 1: Flash program memory erasure enabled.
0	PSWE	<b>Program Store Write Enable.</b> Setting this bit allows writing a byte of data to the Flash program memory using the MOVX write instruction. The Flash location should be erased before writing data. 0: Writes to Flash program memory disabled. 1: Writes to Flash program memory enabled; the MOVX write instruction targets Flash memory.

---

**SFR Definition 23.2. EEDATA: EEPROM Byte Data**


---

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

SFR Address = 0xD1; SFR Page = All Pages

Bit	Name	Description	Write	Read
7:0	EEDATA[7:0]	<b>E2PROM Data</b> The EEDATA register is used to read bytes from the EEPROM space and write bytes to EEPROM space.	Writes byte to location stored in EEADDR.	Returns contents at location stored in EEADDR.

---

## 25.5. Comparator0 Reset

Comparator0 can be configured as a reset source by writing a 1 to the C0RSEF flag (RSTSRC.5). Comparator0 should be enabled and allowed to settle prior to writing to C0RSEF to prevent any turn-on chatter on the output from generating an unwanted reset. The Comparator0 reset is active-low: if the non-inverting input voltage (on CP0+) is less than the inverting input voltage (on CP0-), the device is put into the reset state. After a Comparator0 reset, the C0RSEF flag (RSTSRC.5) will read 1 signifying Comparator0 as the reset source; otherwise, this bit reads 0. The state of the  $\overline{\text{RST}}$  pin is unaffected by this reset.

## 25.6. Watchdog Timer Reset

The programmable Watchdog Timer (WDT) can be used to prevent software from running out of control during a system malfunction. The WDT function can be enabled or disabled by software as described in Section “26. Watchdog Timer” on page 169. If a system malfunction prevents user software from updating the WDT, a reset is generated and the WDTRSF bit (RSTSRC.3) is set to 1. The state of the  $\overline{\text{RST}}$  pin is unaffected by this reset.

## 25.7. Flash Error Reset

If a Flash read/write/erase or program read targets an illegal address, a system reset is generated. This may occur due to any of the following:

- A Flash write or erase is attempted above user code space. This occurs when PSWE is set to 1 and a MOVX write operation targets an address above address 0x3DFF.
- A Flash read is attempted above user code space. This occurs when a MOVC operation targets an address above address 0x3DFF.
- A Program read is attempted above user code space. This occurs when user code attempts to branch to an address above 0x3DFF.
- A Flash read, write or erase attempt is restricted due to a Flash security setting (see Section “22.3. Security Options” on page 149).

The FERROR bit (RSTSRC.6) is set following a Flash error reset. The state of the  $\overline{\text{RST}}$  pin is unaffected by this reset.

## 25.8. Software Reset

Software may force a reset by writing a 1 to the SWRSF bit (RSTSRC.4). The SWRSF bit will read 1 following a software forced reset. The state of the  $\overline{\text{RST}}$  pin is unaffected by this reset.

## 26. Watchdog Timer

The MCU includes a programmable Watchdog Timer (WDT) running off the system clock. A WDT overflow will force the MCU into the reset state. To prevent the reset, the WDT must be restarted by application software before overflow. If the system experiences a software or hardware malfunction preventing the software from restarting the WDT, the WDT will overflow and cause a reset.

Following a reset the WDT is automatically enabled and running with the default maximum time interval. If desired the WDT can be disabled by system software or locked on to prevent accidental disabling. Once locked, the WDT cannot be disabled until the next system reset. The state of the /RST pin is unaffected by this reset.

The WDT consists of a 21-bit timer running from the programmed system clock. The timer measures the period between specific writes to its control register. If this period exceeds the programmed limit, a WDT reset is generated. The WDT can be enabled and disabled as needed in software, or can be permanently enabled if desired. Watchdog features are controlled via the Watchdog Timer Control Register (WDTCN) shown in SFR Definition 26.1.

### 26.1. Enable/Reset WDT

The watchdog timer is both enabled and reset by writing 0xA5 to the WDTCN register. The user's application software should include periodic writes of 0xA5 to WDTCN as needed to prevent a watchdog timer overflow. The WDT is enabled and reset as a result of any system reset.

### 26.2. Disable WDT

Writing 0xDE followed by 0xAD to the WDTCN register disables the WDT. The following code segment illustrates disabling the WDT:

```
CLR EA          ; disable all interrupts
MOV WDTCN,#0DEh ; disable software watchdog timer
MOV WDTCN,#0ADh
SETB EA         ; re-enable interrupts
```

The writes of 0xDE and 0xAD must occur within 4 clock cycles of each other, or the disable operation is ignored. Interrupts should be disabled during this procedure to avoid delay between the two writes.

### 26.3. Disable WDT Lockout

Writing 0xFF to WDTCN locks out the disable feature. Once locked out, the disable operation is ignored until the next system reset. Writing 0xFF does not enable or reset the watchdog timer. Applications always intending to use the watchdog should write 0xFF to WDTCN in the initialization code.

### 26.4. Setting WDT Interval

WDTCN.[2:0] control the watchdog timeout interval. The interval is given by the following equation:

$4^{(3+WDTCN[2:0])} \times T_{sysclk}$  ; where  $T_{sysclk}$  is the system clock period.

For a 3 MHz system clock, this provides an interval range of 0.021 to 349.5 ms. WDTCN.7 must be logic 0 when setting this interval. Reading WDTCN returns the programmed interval. WDTCN.[2:0] reads 111b after a system reset.



### 28.2.2. Assigning Port I/O Pins to Digital Functions

Any Port pins not assigned to analog functions may be assigned to digital functions or used as GPIO. Most digital functions rely on the Crossbar for pin assignment; however, some digital functions bypass the Crossbar in a manner similar to the analog functions listed above. **Port pins used by these digital functions and any Port pins selected for use as GPIO should have their corresponding bit in PnSKIP set to 1.** Table 28.2 shows all available digital functions and the potential mapping of Port I/O to each digital function.

**Table 28.2. Port I/O Assignment for Digital Functions**

Digital Function	Potentially Assignable Port Pins	SFR(s) used for Assignment
UART0, SPI0, SMBus, CP0, CP0A, SYSCLK, PCA0 (CEX0-2 and ECI), T0 or T1.	Any Port pin available for assignment by the Crossbar. This includes P0.0–P2.7 pins which have their PnSKIP bit set to 0. <b>Note:</b> The Crossbar will always assign UART0 pins to P0.4 and P0.5.	XBR0, XBR1
Any pin used for GPIO	P0.0–P6.5	P0SKIP, P1SKIP, P2SKIP
External Memory Interface	P3.0–P6.2	EMI0CF

### 28.2.3. Assigning Port I/O Pins to External Event Trigger Functions

External event trigger functions can be used to trigger an interrupt or wake the device from a low power mode when a transition occurs on a digital I/O pin. The event trigger functions do not require dedicated pins and will function on both GPIO pins (PnSKIP = 1) and pins in use by the Crossbar (PnSKIP = 0). External event trigger functions cannot be used on pins configured for analog I/O. Table 28.3 shows all available external event trigger functions.

**Table 28.3. Port I/O Assignment for External Event Trigger Functions**

Event Trigger Function	Potentially Assignable Port Pins	SFR(s) used for Assignment
External Interrupt 0	P0.0–P0.7	IT01CF
External Interrupt 1	P0.0–P0.7	IT01CF
Port Match	P0.0–P1.7	P0MASK, P0MAT P1MASK, P1MAT

# C8051F70x/71x

## SFR Definition 28.15. P1SKIP: Port 1 Skip

Bit	7	6	5	4	3	2	1	0
Name	P1SKIP[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD5; SFR Page = F

Bit	Name	Function
7:0	P1SKIP[7:0]	<b>Port 1 Crossbar Skip Enable Bits.</b> These bits select Port 1 pins to be skipped by the Crossbar Decoder. Port pins used for analog, special functions or GPIO should be skipped by the Crossbar. 0: Corresponding P1.n pin is not skipped by the Crossbar. 1: Corresponding P1.n pin is skipped by the Crossbar.

## SFR Definition 28.16. P1DRV: Port 1 Drive Strength

Bit	7	6	5	4	3	2	1	0
Name	P1DRV[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xFA; SFR Page = F

Bit	Name	Function
7:0	P1DRV[7:0]	<b>Drive Strength Configuration Bits for P1.7–P1.0 (respectively).</b> Configures digital I/O Port cells to high or low output drive strength. 0: Corresponding P1.n Output has low output drive strength. 1: Corresponding P1.n Output has high output drive strength.

# C8051F70x/71x

## SFR Definition 28.31. P5MDIN: Port 5 Input Mode

Bit	7	6	5	4	3	2	1	0
Name	P5MDIN[7:0]							
Type	R/W							
Reset	1	1	1	1	1	1	1	1

SFR Address = 0xF6; SFR Page = F

Bit	Name	Function
7:0	P5MDIN[7:0]	<b>Analog Configuration Bits for P5.7–P5.0 (respectively).</b> Port pins configured for analog mode have their weak pullup, digital driver, and digital receiver disabled. 0: Corresponding P5.n pin is configured for analog mode. 1: Corresponding P5.n pin is not configured for analog mode.

## SFR Definition 28.32. P5MDOUT: Port 5 Output Mode

Bit	7	6	5	4	3	2	1	0
Name	P5MDOUT[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x9B; SFR Page = F

Bit	Name	Function
7:0	P5MDOUT[7:0]	<b>Output Configuration Bits for P5.7–P5.0 (respectively).</b> These bits are ignored if the corresponding bit in register P5MDIN is logic 0. 0: Corresponding P5.n Output is open-drain. 1: Corresponding P5.n Output is push-pull.

**SFR Definition 29.1. CRC0CN: CRC0 Control**

Bit	7	6	5	4	3	2	1	0
Name				CRC0SEL	CRC0INIT	CRC0VAL	CRC0PNT[1:0]	
Type	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 = F

Bit	Name	Function
7:5	Unused	Read = 000b; Write = Don't Care.
4	CRC0SEL	<b>CRC0 Polynomial Select Bit.</b> This bit selects the CRC0 polynomial and result length (32-bit or 16-bit). 0: CRC0 uses the 32-bit polynomial 0x04C11DB7 for calculating the CRC result. 1: CRC0 uses the 16-bit polynomial 0x1021 for calculating the CRC result.
3	CRC0INIT	<b>CRC0 Result Initialization Bit.</b> Writing a 1 to this bit initializes the entire CRC result based on CRC0VAL.
2	CRC0VAL	<b>CRC0 Set Value Initialization Bit.</b> This bit selects the set value of the CRC result. 0: CRC result is set to 0x00000000 on write of 1 to CRC0INIT. 1: CRC result is set to 0xFFFFFFFF on write of 1 to CRC0INIT.
1:0	CRC0PNT[1:0]	<b>CRC0 Result Pointer.</b> Specifies the byte of the CRC result to be read/written on the next access to CRC0DAT. The value of these bits will auto-increment upon each read or write. For CRC0SEL = 0: 00: CRC0DAT accesses bits 7–0 of the 32-bit CRC result. 01: CRC0DAT accesses bits 15–8 of the 32-bit CRC result. 10: CRC0DAT accesses bits 23–16 of the 32-bit CRC result. 11: CRC0DAT accesses bits 31–24 of the 32-bit CRC result. For CRC0SEL = 1: 00: CRC0DAT accesses bits 7–0 of the 16-bit CRC result. 01: CRC0DAT accesses bits 15–8 of the 16-bit CRC result. 10: CRC0DAT accesses bits 7–0 of the 16-bit CRC result. 11: CRC0DAT accesses bits 15–8 of the 16-bit CRC result.

### 33.1. 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 “21.2. Interrupt Register Descriptions” on page 140); Timer 1 interrupts can be enabled by setting the ET1 bit in the IE register (Section “21.2. Interrupt Register Descriptions” on page 140). 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.

#### 33.1.1. 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 “28.3. Priority Crossbar Decoder” on page 185 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 33.1).

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 21.7). Setting GATE0 to 1 allows the timer to be controlled by the external input signal INT0 (see Section “21.2. Interrupt Register Descriptions” on page 140), 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 INT1 is used with Timer 1; the INT1 polarity is defined by bit IN1PL in register IT01CF (see SFR Definition 21.7).