

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	Not For New Designs
Core Processor	8051
Core Size	8-Bit
Speed	20MHz
Connectivity	SMBus (2-Wire/I ² C), SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, Temp Sensor, WDT
Number of I/O	8
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 4x12b; D/A 2x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	32-LQFP
Supplier Device Package	32-LQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f002-gq

Figure 10.5. DPH: Data Pointer High Byte	74
Figure 10.6. PSW: Program Status Word.....	75
Figure 10.7. ACC: Accumulator.....	76
Figure 10.8. B: B Register	76
10.4. INTERRUPT HANDLER	77
Table 10.4. Interrupt Summary.....	78
Figure 10.9. IE: Interrupt Enable.....	79
Figure 10.10. IP: Interrupt Priority	80
Figure 10.11. EIE1: Extended Interrupt Enable 1	81
Figure 10.12. EIE2: Extended Interrupt Enable 2	82
Figure 10.13. EIP1: Extended Interrupt Priority 1	83
Figure 10.14. EIP2: Extended Interrupt Priority 2	84
10.5. Power Management Modes	85
Figure 10.15. PCON: Power Control Register	86
11. FLASH MEMORY.....	87
11.1. Programming The Flash Memory.....	87
Table 11.1. FLASH Memory Electrical Characteristics	87
11.2. Non-volatile Data Storage	88
11.3. Security Options	88
Figure 11.1. PSCTL: Program Store RW Control.....	88
Figure 11.2. Flash Program Memory Security Bytes	89
Figure 11.3. FLACL: Flash Access Limit (C8051F005/06/07/15/16/17 only)	90
Figure 11.4. FLSCL: Flash Memory Timing Prescaler	91
12. EXTERNAL RAM (C8051F005/06/07/15/16/17).....	92
Figure 12.1. EMI0CN: External Memory Interface Control	92
13. RESET SOURCES	93
Figure 13.1. Reset Sources Diagram	93
13.1. Power-on Reset.....	94
13.2. Software Forced Reset.....	94
Figure 13.2. VDD Monitor Timing Diagram	94
13.3. Power-fail Reset.....	94
13.4. External Reset.....	95
13.5. Missing Clock Detector Reset	95
13.6. Comparator 0 Reset	95
13.7. External CNVSTR Pin Reset.....	95
13.8. Watchdog Timer Reset	95
Figure 13.3. WDTCN: Watchdog Timer Control Register	96
Figure 13.4. RSTSRC: Reset Source Register.....	97
Table 13.1. Reset Electrical Characteristics	98
14. OSCILLATOR	99
Figure 14.1. Oscillator Diagram.....	99
Figure 14.2. OSCICN: Internal Oscillator Control Register	100
Table 14.1. Internal Oscillator Electrical Characteristics	100
Figure 14.3. OSCXCN: External Oscillator Control Register.....	101
14.1. External Crystal Example	102
14.2. External RC Example	102
14.3. External Capacitor Example	102
15. PORT INPUT/OUTPUT.....	103
15.1. Priority Cross Bar Decoder.....	103
15.2. Port I/O Initialization	103
Figure 15.1. Port I/O Functional Block Diagram	104
Figure 15.2. Port I/O Cell Block Diagram.....	104
Table 15.1. Crossbar Priority Decode	105
Figure 15.3. XBR0: Port I/O CrossBar Register 0	106

Table 18.2. Oscillator Frequencies for Standard Baud Rates	136
Figure 18.8. SBUF: Serial (UART) Data Buffer Register.....	136
Figure 18.9. SCON: Serial Port Control Register.....	137
19. TIMERS	138
19.1. Timer 0 and Timer 1	138
Figure 19.1. T0 Mode 0 Block Diagram.....	139
Figure 19.2. T0 Mode 2 Block Diagram.....	140
Figure 19.3. T0 Mode 3 Block Diagram.....	141
Figure 19.4. TCON: Timer Control Register.....	142
Figure 19.5. TMOD: Timer Mode Register.....	143
Figure 19.6. CKCON: Clock Control Register.....	144
Figure 19.7. TL0: Timer 0 Low Byte	145
Figure 19.8. TL1: Timer 1 Low Byte	145
Figure 19.9. TH0: Timer 0 High Byte	145
Figure 19.10. TH1: Timer 1 High Byte	145
19.2. Timer 2	146
Figure 19.11. T2 Mode 0 Block Diagram.....	147
Figure 19.12. T2 Mode 1 Block Diagram.....	148
Figure 19.13. T2 Mode 2 Block Diagram.....	149
Figure 19.14. T2CON: Timer 2 Control Register.....	150
Figure 19.15. RCAP2L: Timer 2 Capture Register Low Byte	151
Figure 19.16. RCAP2H: Timer 2 Capture Register High Byte	151
Figure 19.17. TL2: Timer 2 Low Byte	151
Figure 19.18. TH2: Timer 2 High Byte	151
19.3. Timer 3	152
Figure 19.19. Timer 3 Block Diagram.....	152
Figure 19.20. TMR3CN: Timer 3 Control Register	152
Figure 19.21. TMR3RLL: Timer 3 Reload Register Low Byte	153
Figure 19.22. TMR3RLH: Timer 3 Reload Register High Byte	153
Figure 19.23. TMR3L: Timer 3 Low Byte.....	153
Figure 19.24. TMR3H: Timer 3 High Byte.....	153
20. PROGRAMMABLE COUNTER ARRAY.....	154
Figure 20.1. PCA Block Diagram	154
20.1. Capture/Compare Modules	155
Table 20.1. PCA0CPM Register Settings for PCA Capture/Compare Modules	155
Figure 20.2. PCA Interrupt Block Diagram.....	155
Figure 20.3. PCA Capture Mode Diagram.....	156
Figure 20.4. PCA Software Timer Mode Diagram.....	157
Figure 20.5. PCA High Speed Output Mode Diagram.....	157
Figure 20.6. PCA PWM Mode Diagram	158
20.2. PCA Counter/Timer.....	159
Table 20.2. PCA Timebase Input Options.....	159
Figure 20.7. PCA Counter/Timer Block Diagram.....	159
20.3. Register Descriptions for PCA	160
Figure 20.8. PCA0CN: PCA Control Register	160
Figure 20.9. PCA0MD: PCA Mode Register	161
Figure 20.10. PCA0CPMn: PCA Capture/Compare Registers.....	162
Figure 20.11. PCA0L: PCA Counter/Timer Low Byte	163
Figure 20.12. PCA0H: PCA Counter/Timer High Byte	163
Figure 20.13. PCA0CPLn: PCA Capture Module Low Byte	163
Figure 20.14. PCA0CPHn: PCA Capture Module High Byte.....	163
21. JTAG (IEEE 1149.1).....	164
Figure 21.1. IR: JTAG Instruction Register	164
21.1. Boundary Scan.....	165

1. SYSTEM OVERVIEW

The C8051F000 family are fully integrated mixed-signal System on a Chip MCUs with a true 12-bit multi-channel ADC (F000/01/02/05/06/07), or a true 10-bit multi-channel ADC (F010/11/12/15/16/17). See the Product Selection Guide in Table 1.1 for a quick reference of each MCUs' feature set. Each has a programmable gain pre-amplifier, two 12-bit DACs, two voltage comparators (except for the F002/07/12/17, which have one), a voltage reference, and an 8051-compatible microcontroller core with 32kbytes of FLASH memory. There are also I2C/SMBus, UART, and SPI serial interfaces implemented in hardware (not "bit-banged" in user software) as well as a Programmable Counter/Timer Array (PCA) with five capture/compare modules. There are also 4 general-purpose 16-bit timers and 4 byte-wide general-purpose digital Port I/O. The C8051F000/01/02/10/11/12 have 256 bytes of RAM and execute up to 20MIPS, while the C8051F005/06/07/15/16/17 have 2304 bytes of RAM and execute up to 25MIPS.

With an on-board VDD monitor, WDT, and clock oscillator, the MCUs are truly stand-alone System-on-a-Chip solutions. Each MCU effectively configures and manages the analog and digital peripherals. The FLASH memory can be reprogrammed even in-circuit, providing non-volatile data storage, and also allowing field upgrades of the 8051 firmware. Each MCU can also individually shut down any or all of the peripherals to conserve power.

On-board JTAG debug support allows non-intrusive (uses no on-chip resources), full speed, in-circuit debug using the production MCU installed in the final application. This debug system supports inspection and modification of memory and registers, setting breakpoints, watchpoints, single stepping, and run and halt commands. All analog and digital peripherals are fully functional when using JTAG debug.

Each MCU is specified for 2.7V-to-3.6V operation over the industrial temperature range (-45C to +85C). The Port I/Os, /RST, and JTAG pins are tolerant for input signals up to 5V. The C8051F000/05/10/15 are available in the 64-pin TQFP (see block diagram in Figure 1.1). The C8051F001/06/11/16 are available in the 48-pin TQFP (see block diagram in Figure 1.2). The C8051F002/07/12/17 are available in the 32-pin LQFP (see block diagram in Figure 1.3).

Table 1.1. Product Selection Guide

	MIPS (Peak)	FLASH Memory	RAM	SMBus/I2C	SPI	UART	Timers (16-bit)	Programmable Counter Array	Digital Port I/O's	ADC Resolution (bits)	ADC Max Speed (ksps)	ADC Inputs	Voltage Reference	Temperature Sensor	DAC Resolution	DAC Outputs	Voltage Comparators	Package
C8051F000	20	32k	256	√	√	√	4	√	32	12	100	8	√	√	12	2	2	64TQFP
C8051F001	20	32k	256	√	√	√	4	√	16	12	100	8	√	√	12	2	2	48TQFP
C8051F002	20	32k	256	√	√	√	4	√	8	12	100	4	√	√	12	2	1	32LQFP
C8051F005	25	32k	2304	√	√	√	4	√	32	12	100	8	√	√	12	2	2	64TQFP
C8051F006	25	32k	2304	√	√	√	4	√	16	12	100	8	√	√	12	2	2	48TQFP
C8051F007	25	32k	2304	√	√	√	4	√	8	12	100	4	√	√	12	2	1	32LQFP
C8051F010	20	32k	256	√	√	√	4	√	32	10	100	8	√	√	12	2	2	64TQFP
C8051F011	20	32k	256	√	√	√	4	√	16	10	100	8	√	√	12	2	2	48TQFP
C8051F012	20	32k	256	√	√	√	4	√	8	10	100	4	√	√	12	2	1	32LQFP
C8051F015	25	32k	2304	√	√	√	4	√	32	10	100	8	√	√	12	2	2	64TQFP
C8051F016	25	32k	2304	√	√	√	4	√	16	10	100	8	√	√	12	2	2	48TQFP
C8051F017	25	32k	2304	√	√	√	4	√	8	10	100	4	√	√	12	2	1	32LQFP

Figure 5.14. 12-Bit ADC Window Interrupt Examples, Right Justified Data

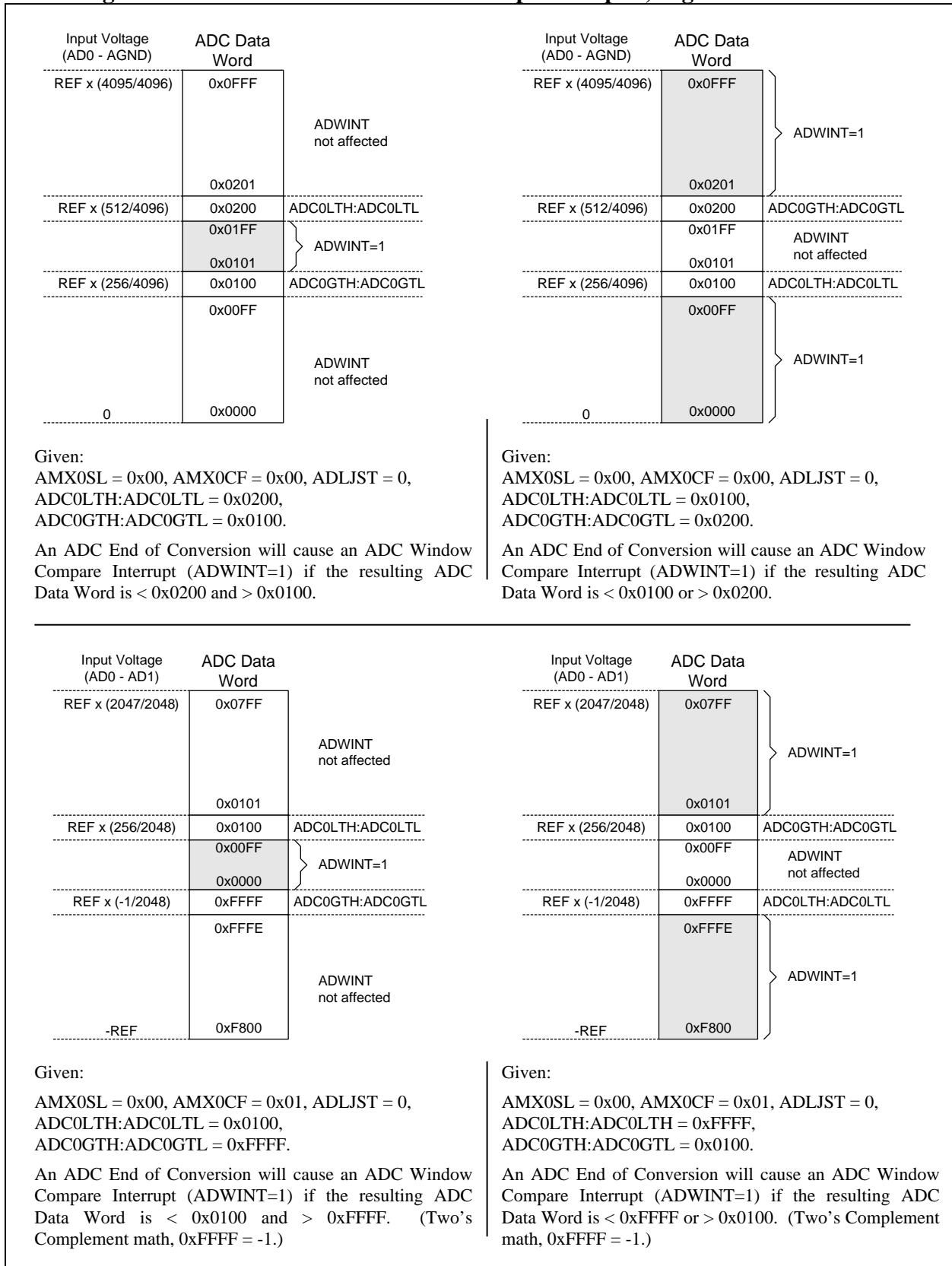


Figure 6.6. ADC0CF: ADC Configuration Register (C8051F01x)

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
ADCSC2	ADCSC1	ADCSC0	-	-	AMPGN2	AMPGN1	AMPGN0	01100000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xBC

Bits7-5: ADCSC2-0: ADC SAR Conversion Clock Period Bits
 000: SAR Conversion Clock = 1 System Clock
 001: SAR Conversion Clock = 2 System Clocks
 010: SAR Conversion Clock = 4 System Clocks
 011: SAR Conversion Clock = 8 System Clocks
 1xx: SAR Conversion Clock = 16 Systems Clocks
 (Note: Conversion clock should be ≤ 2MHz.)

Bits4-3: UNUSED. Read = 00b; Write = don't care

Bits2-0: AMPGN2-0: ADC Internal Amplifier Gain
 000: Gain = 1
 001: Gain = 2
 010: Gain = 4
 011: Gain = 8
 10x: Gain = 16
 11x: Gain = 0.5

Table 10.1. CIP-51 Instruction Set Summary

Mnemonic	Description	Bytes	Clock Cycles
ARITHMETIC OPERATIONS			
ADD A,Rn	Add register to A	1	1
ADD A,direct	Add direct byte to A	2	2
ADD A,@Ri	Add indirect RAM to A	1	2
ADD A,#data	Add immediate to A	2	2
ADDC A,Rn	Add register to A with carry	1	1
ADDC A,direct	Add direct byte to A with carry	2	2
ADDC A,@Ri	Add indirect RAM to A with carry	1	2
ADDC A,#data	Add immediate to A with carry	2	2
SUBB A,Rn	Subtract register from A with borrow	1	1
SUBB A,direct	Subtract direct byte from A with borrow	2	2
SUBB A,@Ri	Subtract indirect RAM from A with borrow	1	2
SUBB A,#data	Subtract immediate from A with borrow	2	2
INC A	Increment A	1	1
INC Rn	Increment register	1	1
INC direct	Increment direct byte	2	2
INC @Ri	Increment indirect RAM	1	2
DEC A	Decrement A	1	1
DEC Rn	Decrement register	1	1
DEC direct	Decrement direct byte	2	2
DEC @Ri	Decrement indirect RAM	1	2
INC DPTR	Increment Data Pointer	1	1
MUL AB	Multiply A and B	1	4
DIV AB	Divide A by B	1	8
DA A	Decimal Adjust A	1	1
LOGICAL OPERATIONS			
ANL A,Rn	AND Register to A	1	1
ANL A,direct	AND direct byte to A	2	2
ANL A,@Ri	AND indirect RAM to A	1	2
ANL A,#data	AND immediate to A	2	2
ANL direct,A	AND A to direct byte	2	2
ANL direct,#data	AND immediate to direct byte	3	3
ORL A,Rn	OR Register to A	1	1
ORL A,direct	OR direct byte to A	2	2
ORL A,@Ri	OR indirect RAM to A	1	2
ORL A,#data	OR immediate to A	2	2
ORL direct,A	OR A to direct byte	2	2
ORL direct,#data	OR immediate to direct byte	3	3
XRL A,Rn	Exclusive-OR Register to A	1	1
XRL A,direct	Exclusive-OR direct byte to A	2	2
XRL A,@Ri	Exclusive-OR indirect RAM to A	1	2
XRL A,#data	Exclusive-OR immediate to A	2	2
XRL direct,A	Exclusive-OR A to direct byte	2	2
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

10.2. MEMORY ORGANIZATION

The memory organization of the CIP-51 System Controller is similar to that of a standard 8051. There are two separate memory spaces: program memory and data memory. Program and data memory share the same address space but are accessed via different instruction types. There are 256 bytes of internal data memory and 64K bytes of internal program memory address space implemented within the CIP-51. The CIP-51 memory organization is shown in Figure 10.2.

10.2.1. Program Memory

The CIP-51 has a 64K-byte program memory space. The MCU implements 32896 bytes of this program memory space as in-system, reprogrammable Flash memory, organized in a contiguous block from addresses 0x0000 to 0x807F. Note: 512 bytes (0x7E00 – 0x7FFF) of this memory are reserved for factory use and are not available for user program storage.

Program memory is normally assumed to be read-only. However, the CIP-51 can write to program memory by setting the Program Store Write Enable bit (PSCCTL.0) and using the MOVX instruction. This feature provides a mechanism for the CIP-51 to update program code and use the program memory space for non-volatile data storage. Refer to Section 11 (Flash Memory) for further details.

10.2.2. Data Memory

The CIP-51 implements 256 bytes of internal RAM mapped into the data memory space from 0x00 through 0xFF. The lower 128 bytes of data memory are used for general purpose registers and scratch pad memory. Either direct or indirect addressing may be used to access the lower 128 bytes of data memory. Locations 0x00 through 0x1F are addressable as four banks of general purpose registers, each bank consisting of eight byte-wide registers. The next 16 bytes, locations 0x20 through 0x2F, may be addressed as bytes or as 128 bit locations accessible with the direct-bit addressing mode.

The upper 128 bytes of data memory are accessible only by indirect addressing. This region occupies the same address space as the Special Function Registers (SFR) but is physically separate from the SFR space. The addressing mode used by an instruction when accessing locations above 0x7F determines whether the CPU accesses the upper 128 bytes of data memory space or the SFRs. Instructions that use direct addressing will access the SFR space. Instructions using indirect addressing above 0x7F will access the upper 128 bytes of data memory. Figure 10.2 illustrates the data memory organization of the CIP-51.

The C8051F005/06/07/15/16/17 also have 2048 bytes of RAM in the external data memory space of the CIP-51, accessible using the MOVX instruction. Refer to Section 12 (External RAM) for details.

10.2.3. 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 Figure 10.6). This allows fast context switching when entering subroutines and interrupt service routines. Indirect addressing modes use registers R0 and R1 as index registers.

10.2.4. 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, 22h.3
```

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

C8051F000/1/2/5/6/7

C8051F010/1/2/5/6/7

Address	Register	Description	Page No.
0x89	TMOD	Counter/Timer Mode	143
0x91	TMR3CN	Timer 3 Control	152
0x95	TMR3H	Timer 3 High	153
0x94	TMR3L	Timer 3 Low	153
0x93	TMR3RLH	Timer 3 Reload High	153
0x92	TMR3RLL	Timer 3 Reload Low	153
0xFF	WDTCN	Watchdog Timer Control	96
0xE1	XBR0	Port I/O Crossbar Configuration 1	105
0xE2	XBR1	Port I/O Crossbar Configuration 2	107
0xE3	XBR2	Port I/O Crossbar Configuration 3	108
0x84-86, 0x96-97, 0x9C, 0xA1-A3, 0xA9-AC, 0xAE, 0xB3-B5, 0xB9, 0xBD, 0xC9, 0xCE, 0xDF, 0xE4-E5, 0xF1-F5		Reserved	

* Refers to a register in the C8051F000/1/2/5/6/7 only.

** Refers to a register in the C8051F010/1/2/5/6/7 only.

*** Refers to a register in the C8051F005/06/07/15/16/17 only.

Figure 10.10. IP: Interrupt Priority

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
-	-	PT2	PS	PT1	PX1	PT0	PX0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0 (bit addressable)	SFR Address: 0xB8

Bits7-6: UNUSED. Read = 11b, Write = don't care.

Bit5: PT2: Timer 2 Interrupt Priority Control.
This bit sets the priority of the Timer 2 interrupts.
0: Timer 2 interrupts set to low priority level.
1: Timer 2 interrupts set to high priority level.

Bit4: PS: Serial Port (UART) Interrupt Priority Control.
This bit sets the priority of the Serial Port (UART) interrupts.
0: UART interrupts set to low priority level.
1: UART interrupts set to high priority level.

Bit3: PT1: Timer 1 Interrupt Priority Control.
This bit sets the priority of the Timer 1 interrupts.
0: Timer 1 interrupts set to low priority level.
1: Timer 1 interrupts set to high priority level.

Bit2: PX1: External Interrupt 1 Priority Control.
This bit sets the priority of the External Interrupt 1 interrupts.
0: External Interrupt 1 set to low priority level.
1: External Interrupt 1 set to high priority level.

Bit1: PT0: Timer 0 Interrupt Priority Control.
This bit sets the priority of the Timer 0 interrupts.
0: Timer 0 interrupt set to low priority level.
1: Timer 0 interrupt set to high priority level.

Bit0: PX0: External Interrupt 0 Priority Control.
This bit sets the priority of the External Interrupt 0 interrupts.
0: External Interrupt 0 set to low priority level.
1: External Interrupt 0 set to high priority level.

Figure 10.14. EIP2: Extended Interrupt Priority 2

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
PXVLD	-	PX7	PX6	PX5	PX4	PADC0	PT3	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xF7

Bit7: PXVLD: External Clock Source Valid (XTLVLD) Interrupt Priority Control.
This bit sets the priority of the XTLVLD interrupt.
0: XTLVLD interrupt set to low priority level.
1: XTLVLD interrupt set to high priority level.

Bit6: Reserved: Must write 0. Reads 0.

Bit5: PX7: External Interrupt 7 Priority Control.
This bit sets the priority of the External Interrupt 7.
0: External Interrupt 7 set to low priority level.
1: External Interrupt 7 set to high priority level.

Bit4: PX6: External Interrupt 6 Priority Control.
This bit sets the priority of the External Interrupt 6.
0: External Interrupt 6 set to low priority level.
1: External Interrupt 6 set to high priority level.

Bit3: PX5: External Interrupt 5 Priority Control.
This bit sets the priority of the External Interrupt 5.
0: External Interrupt 5 set to low priority level.
1: External Interrupt 5 set to high priority level.

Bit2: PX4: External Interrupt 4 Priority Control.
This bit sets the priority of the External Interrupt 4.
0: External Interrupt 4 set to low priority level.
1: External Interrupt 4 set to high priority level.

Bit1: PADC0: ADC End of Conversion Interrupt Priority Control.
This bit sets the priority of the ADC0 End of Conversion Interrupt.
0: ADC0 End of Conversion interrupt set to low priority level.
1: ADC0 End of Conversion interrupt set to high priority level.

Bit0: PT3: Timer 3 Interrupt Priority Control.
This bit sets the priority of the Timer 3 interrupts.
0: Timer 3 interrupt set to low priority level.
1: Timer 3 interrupt set to high priority level.

13. RESET SOURCES

The reset circuitry of the MCUs allows the controller to be easily placed in a predefined default condition. On entry to this reset state, the CIP-51 halts program execution, forces the external port pins to a known state and initializes the SFRs to their defined reset values. Interrupts and timers are disabled. On exit, the program counter (PC) is reset, and program execution starts at location 0x0000.

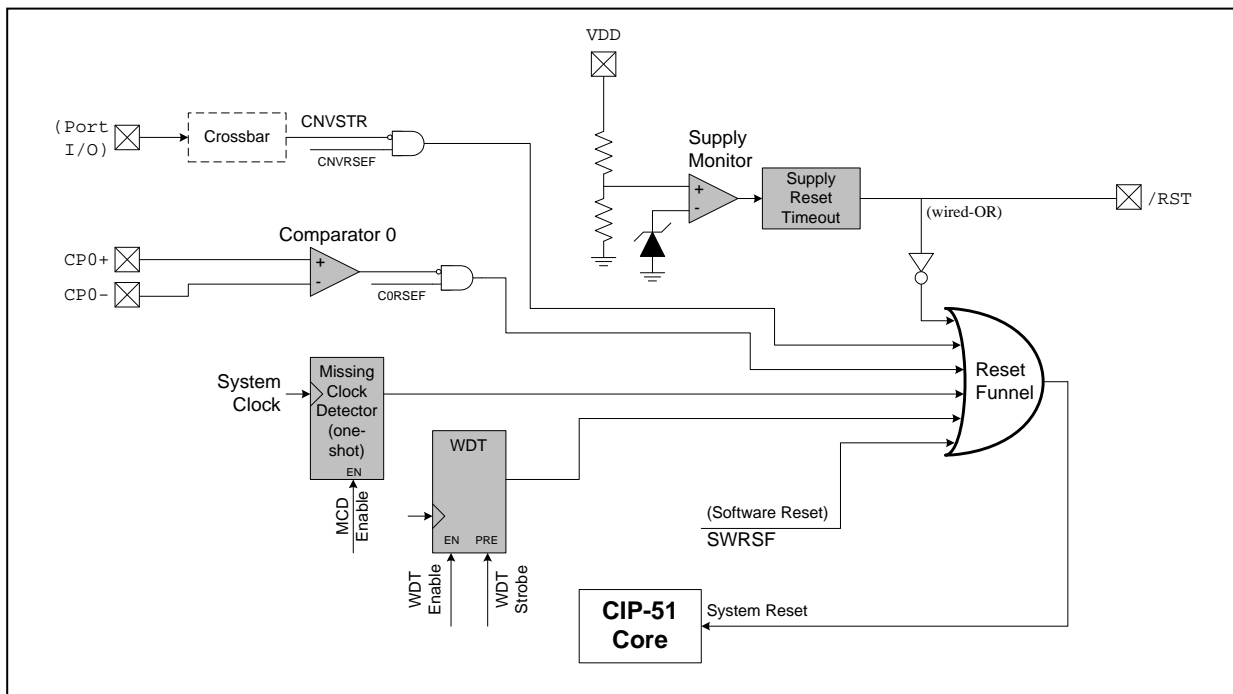
All of the SFRs are reset to predefined values. The reset values of the SFR bits are defined in the SFR detailed descriptions. The contents of internal data memory are not changed during a reset and any previously stored data is preserved. However, since the stack pointer SFR is reset, the stack is effectively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

The I/O port latches are reset to 0xFF (all logic ones), activating internal weak pull-ups which take the external I/O pins to a high state. The weak pull-ups are enabled during and after the reset. If the source of reset is from the VDD Monitor or writing a 1 to PORSEF, the /RST pin is driven low until the end of the VDD reset timeout.

On exit from the reset state, the MCU uses the internal oscillator running at 2MHz as the system clock by default. Refer to Section 14 for information on selecting and configuring the system clock source. The Watchdog Timer is enabled using its longest timeout interval. (Section 13.8 details the use of the Watchdog Timer.)

There are seven sources for putting the MCU into the reset state: power-on/power-fail, external /RST pin, external CNVSTR signal, software commanded, Comparator 0, Missing Clock Detector, and Watchdog Timer. Each reset source is described below:

Figure 13.1. Reset Sources Diagram



13.4. External Reset

The external /RST pin provides a means for external circuitry to force the MCU into a reset state. Asserting an active-low signal on the /RST pin will cause the MCU to enter the reset state. Although there is a weak internal pullup, it may be desirable to provide an external pull-up and/or decoupling of the /RST pin to avoid erroneous noise-induced resets. The MCU will remain in reset until at least 12 clock cycles after the active-low /RST signal is removed. The PINRSF flag (RSTSRC.0) is set on exit from an external reset. The /RST pin is also 5V tolerant.

13.5. Missing Clock Detector Reset

The Missing Clock Detector is essentially a one-shot circuit that is triggered by the MCU system clock. If the system clock goes away for more than 100 μ s, the one-shot will time out and generate a reset. After a Missing Clock Detector reset, the MCDRSF flag (RSTSRC.2) will be set, signifying the MSD as the reset source; otherwise, this bit reads 0. The state of the /RST pin is unaffected by this reset. Setting the MSCLKE bit in the OSCICN register (see Figure 14.2) enables the Missing Clock Detector.

13.6. Comparator 0 Reset

Comparator 0 can be configured as an active-low reset input by writing a 1 to the CORSEF flag (RSTSRC.5). Comparator 0 should be enabled using CPTOCN.7 (see Figure 8.3) at least 20 μ s prior to writing to CORSEF to prevent any turn-on chatter on the output from generating an unwanted reset. When configured as a reset, if the non-inverting input voltage (on CP0+) is less than the inverting input voltage (on CP0-), the MCU is put into the reset state. After a Comparator 0 Reset, the CORSEF flag (RSTSRC.5) will read 1 signifying Comparator 0 as the reset source; otherwise, this bit reads 0. The state of the /RST pin is unaffected by this reset. Also, Comparator 0 can generate a reset with or without the system clock.

13.7. External CNVSTR Pin Reset

The external CNVSTR signal can be configured as an active-low reset input by writing a 1 to the CNVRSEF flag (RSTSRC.6). The CNVSTR signal can appear on any of the P0, P1, or P2 I/O pins as described in Section 15.1. (Note that the Crossbar must be configured for the CNVSTR signal to be routed to the appropriate Port I/O.) The Crossbar should be configured and enabled before the CNVRSEF is set to configure CNVSTR as a reset source. When configured as a reset, CNVSTR is active-low and level sensitive. After a CNVSTR reset, the CNVRSEF flag (RSTSRC.6) will read 1 signifying CNVSTR as the reset source; otherwise, this bit reads 0. The state of the /RST pin is unaffected by this reset.

13.8. Watchdog Timer Reset

The MCU includes a programmable Watchdog Timer (WDT) running off the system clock. The WDT will force the MCU into the reset state when the watchdog timer overflows. To prevent the reset, the WDT must be restarted by application software before the overflow occurs. If the system experiences a software/hardware malfunction preventing the software from restarting the WDT, the WDT will overflow and cause a reset. This should prevent the system from running out of control.

The WDT is automatically enabled and started with the default maximum time interval on exit from all resets. 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.

13.8.1. Watchdog Usage

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 Figure 13.3.

Enable/Reset WDT

The watchdog timer is both enabled and the countdown restarted 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 restarted as a result of any system reset.

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
MOV    WDTCN,#0ADh     ; watchdog timer
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.

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 their initialization code.

Setting WDT Interval

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

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

For a 2MHz system clock, this provides an interval range of 0.032msec to 524msec. WDTCN.7 must be a 0 when setting this interval. Reading WDTCN returns the programmed interval. WDTCN.[2:0] is 111b after a system reset.

Figure 13.3. WDTCN: Watchdog Timer Control Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	xxxxx111
								SFR Address: 0xFF
<p>Bits7-0: WDT Control Writing 0xA5 both enables and reloads the WDT. Writing 0xDE followed within 4 clocks by 0xAD disables the WDT. Writing 0xFF locks out the disable feature.</p> <p>Bit4: Watchdog Status Bit (when Read) Reading the WDTCN.[4] bit indicates the Watchdog Timer Status. 0: WDT is inactive 1: WDT is active</p> <p>Bits2-0: Watchdog Timeout Interval Bits The WDTCN.[2:0] bits set the Watchdog Timeout Interval. When writing these bits, WDTCN.7 must be set to 0.</p>								

Figure 15.13. P3: Port3 Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
P3.7	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0	11111111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0 (bit addressable)	SFR Address: 0xB0

Bits7-0: P3.[7:0]
 (Write)
 0: Logic Low Output.
 1: Logic High Output (high-impedance if corresponding PRT3CF.n bit = 0)
 (Read)
 0: P3.n is logic low.
 1: P3.n is logic high.

Figure 15.14. PRT3CF: Port3 Configuration Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xA7

Bits7-0: PRT3CF.[7:0]: Output Configuration Bits for P3.7-P3.0 (respectively)
 0: Corresponding P3.n Output mode is Open-Drain.
 1: Corresponding P3.n Output mode is Push-Pull.

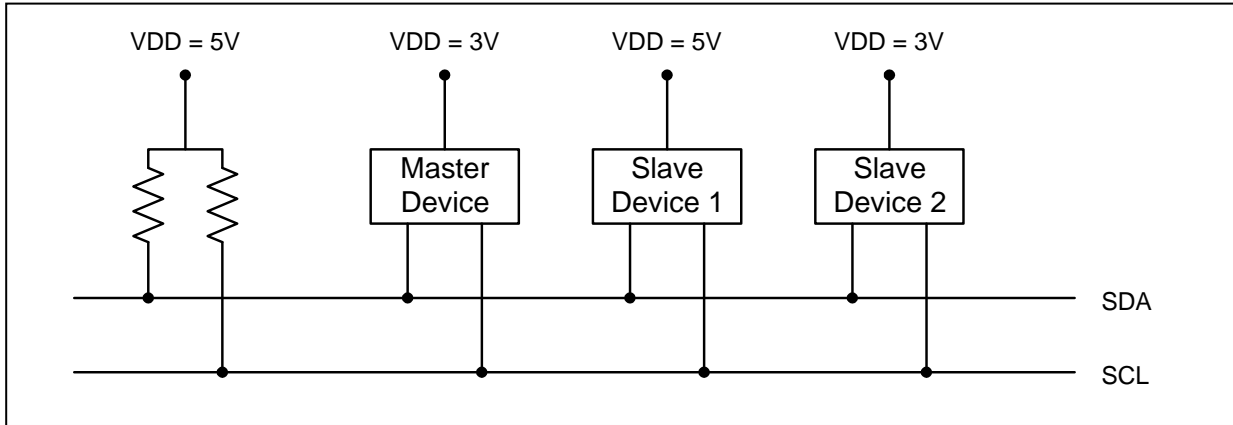
Table 15.2. Port I/O DC Electrical Characteristics

VDD = 2.7 to 3.6V, -40°C to +85°C unless otherwise specified.

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS
Output High Voltage	I _{OH} = -10uA, Port I/O push-pull	VDD - 0.1			V
	I _{OH} = -3mA, Port I/O push-pull	VDD - 0.7			
	I _{OH} = -10mA, Port I/O push-pull		VDD - 0.8		
Output Low Voltage	I _{OL} = 10uA			0.1	V
	I _{OL} = 8.5mA			0.6	
	I _{OL} = 25mA		1.0		
Input High Voltage		0.7 x VDD			V
Input Low Voltage				0.3 x VDD	V
Input Leakage Current	DGND < Port Pin < VDD, Pin Tri-state Weak Pull-up Off Weak Pull-up On		30	±1	µA
Capacitive Loading			5		pF

Figure 16.2 shows a typical SMBus configuration. The SMBus interface will work at any voltage between 3.0V and 5.0V and different devices on the bus may operate at different voltage levels. The SCL (serial clock) and SDA (serial data) lines are bi-directional. They must be connected to a positive power supply voltage through a pull-up resistor or similar circuit. When the bus is free, both lines are pulled high. Every device connected to the bus must have an open-drain or open-collector output for both the SCL and SDA lines. The maximum number of devices on the bus is limited only by the requirement that the rise and fall times on the bus will not exceed 300ns and 1000ns, respectively.

Figure 16.2. Typical SMBus Configuration



16.1. Supporting Documents

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

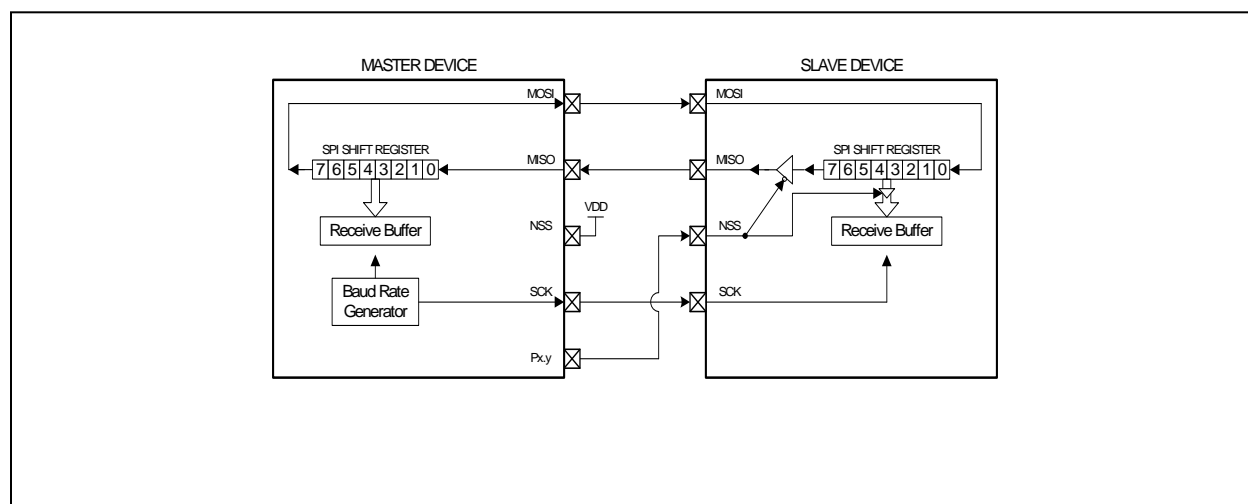
1. *The I²C-bus and how to use it (including specifications)*, Philips Semiconductor.
2. *The I²C-Bus Specification -- Version 2.0*, Philips Semiconductor.
3. *System Management Bus Specification -- Version 1.1*, SBS Implementers Forum.

17.2. Operation

Only a SPI master device can initiate a data transfer. The SPI is placed in master mode by setting the Master Enable flag (MSTEN, SPIOCN.1). Writing a byte of data to the SPI data register (SPIO DAT) when in Master Mode starts a data transfer. The SPI master immediately shifts out the data serially on the MOSI line while providing the serial clock on SCK. The SPIF (SPIOCN.7) flag is set to logic 1 at the end of the transfer. If interrupts are enabled, an interrupt request is generated when the SPIF flag is set. The SPI master can be configured to shift in/out from one to eight bits in a transfer operation in order to accommodate slave devices with different word lengths. The SPIFRS bits in the SPI Configuration Register (SPIOCFG.[2:0]) are used to select the number of bits to shift in/out in a transfer operation.

While the SPI master transfers data to a slave on the MOSI line, the addressed SPI slave device simultaneously transfers the contents of its shift register to the SPI master on the MISO line in a full-duplex operation. The data byte received from the slave replaces the data in the master's data register. Therefore, the SPIF flag serves as both a transmit-complete and receive-data-ready flag. The data transfer in both directions is synchronized with the serial clock generated by the master. Figure 17.3 illustrates the full-duplex operation of an SPI master and an addressed slave.

Figure 17.3. Full Duplex Operation



The SPI data register is double buffered on reads, but not on a write. If a write to SPIO DAT is attempted during a data transfer, the WCOL flag (SPIOCN.6) will be set to logic 1 and the write is ignored. The current data transfer will continue uninterrupted. A read of the SPI data register by the system controller actually reads the receive buffer. If the receive buffer still holds unread data from a previous transfer when the last bit of the current transfer is shifted into the SPI shift register, a receive overrun occurs and the RXOVRN flag (SPIOCN.4) is set to logic 1. The new data is not transferred to the receive buffer, allowing the previously received data byte to be read. The data byte causing the overrun is lost.

When the SPI is enabled and not configured as a master, it will operate as an SPI slave. Another SPI device acting as a master will initiate a transfer by driving the NSS signal low. The master then shifts data out of the shift register on the MOSI pin using its serial clock. The SPIF flag is set to logic 1 at the end of a data transfer (when the NSS signal goes high). The slave can load its shift register for the next data transfer by writing to the SPI data register. The slave must make the write to the data register at least one SPI serial clock cycle before the master starts the next transmission. Otherwise, the byte of data already in the slave's shift register will be transferred.

Multiple masters may reside on the same bus. A Mode Fault flag (MODF, SPIOCN.5) is set to logic 1 when the SPI is configured as a master (MSTEN = 1) and its slave select signal NSS is pulled low. When the Mode Fault flag is set, the MSTEN and SPIEN bits of the SPI control register are cleared by hardware, thereby placing the SPI module

19.1.3. 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. The TL0 holds the count and TH0 holds the reload value. When the count in TL0 overflows from all ones to 0x00, the timer overflow flag TF0 (TCON.5) is set and the counter in TL0 is reloaded from TH0. If 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.

Figure 19.2. T0 Mode 2 Block Diagram

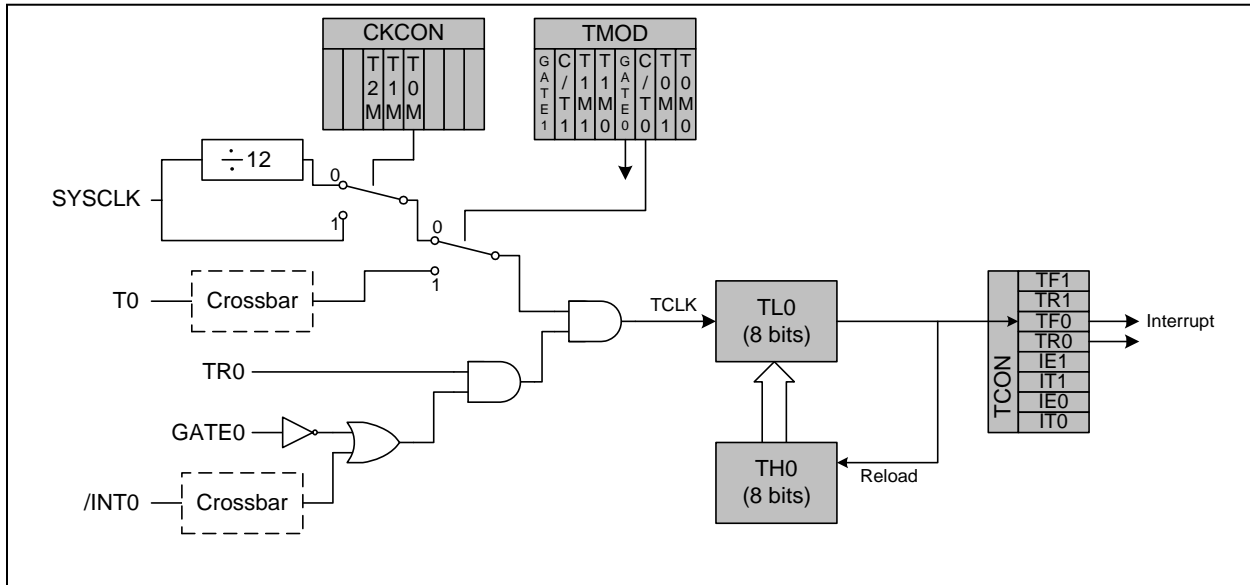


Figure 20.10. PCA0CPMn: PCA Capture/Compare Registers

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
-	ECOMn	CAPPn	CAPNn	MATn	TOGn	PWMn	ECCFn	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xDA-0xDE

PCA0CPMn Address: PCA0CPM0 = 0xDA (n = 0)
PCA0CPM1 = 0xDB (n = 1)
PCA0CPM2 = 0xDC (n = 2)
PCA0CPM3 = 0xDD (n = 3)
PCA0CPM4 = 0xDE (n = 4)

Bit7: UNUSED. Read = 0, Write = don't care.

Bit6: ECOMn: Comparator Function Enable.
This bit enables/disables the comparator function for PCA module *n*.
0: Disabled.
1: Enabled.

Bit5: CAPPn: Capture Positive Function Enable.
This bit enables/disables the positive edge capture for PCA module *n*.
0: Disabled.
1: Enabled.

Bit4: CAPNn: Capture Negative Function Enable.
This bit enables/disables the negative edge capture for PCA module *n*.
0: Disabled.
1: Enabled.

Bit3: MATn: Match Function Enable.
This bit enables/disables the match function for PCA module *n*. When enabled, matches of the PCA counter with a module's capture/compare register cause the CCFn bit in PCA0MD register to be set.
0: Disabled.
1: Enabled.

Bit2: TOGn: Toggle Function Enable.
This bit enables/disables the toggle function for PCA module *n*. When enabled, matches of the PCA counter with a module's capture/compare register cause the logic level on the CEXn pin to toggle.
0: Disabled.
1: Enabled.

Bit1: PWMn: Pulse Width Modulation Mode Enable.
This bit enables/disables the comparator function for PCA module *n*. When enabled, a pulse width modulated signal is output on the CEXn pin.
0: Disabled.
1: Enabled.

Bit0: ECCFn: Capture/Compare Flag Interrupt Enable.
This bit sets the masking of the Capture/Compare Flag (CCFn) interrupt.
0: Disable CCFn interrupts.
1: Enable a Capture/Compare Flag interrupt request when CCFn is set.

Figure 20.11. PCA0L: PCA Counter/Timer Low Byte

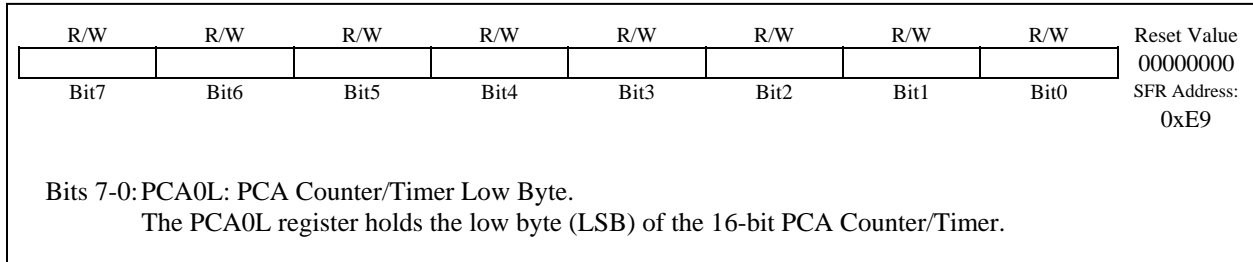


Figure 20.12. PCA0H: PCA Counter/Timer High Byte

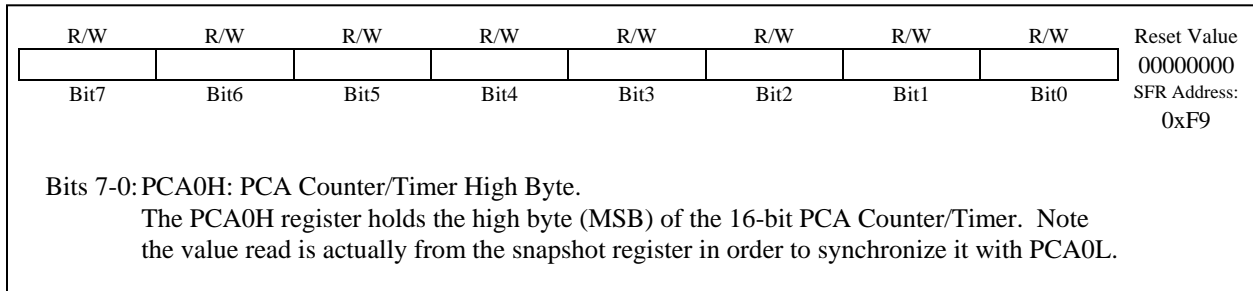


Figure 20.13. PCA0CPLn: PCA Capture Module Low Byte

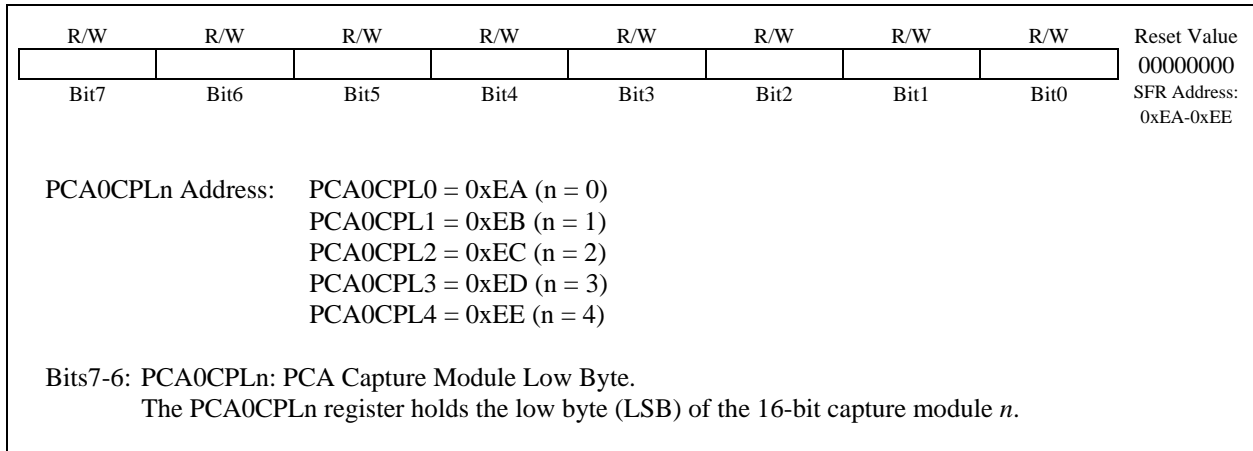
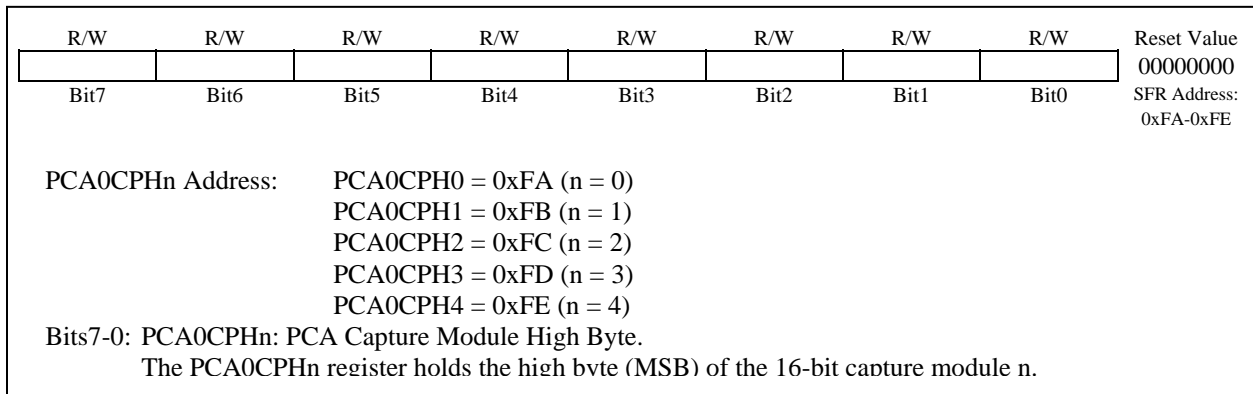


Figure 20.14. PCA0CPHn: PCA Capture Module High Byte



21.3. Debug Support

Each MCU has on-chip JTAG and debug circuitry that provide *non-intrusive, full speed, in-circuit debug using the production part installed in the end application* using the four pin JTAG I/F. Silicon Labs' debug system supports inspection and modification of memory and registers, setting breakpoints, watchpoints, single stepping, and run and halt commands. No additional target RAM, program memory, or communications channels are required. All the digital and analog peripherals are functional and work correctly (remain in sync) while debugging. The WDT is disabled when the MCU is halted during single stepping or at a breakpoint.

The C8051F000DK, C8051F005DK, C8051F010DK, and C8051F015DK are development kits with all the hardware and software necessary to develop application code and perform in-circuit debugging with each MCU in the C8051F000 family. Each kit includes an Integrated Development Environment (IDE) which has a debugger and integrated 8051 assembler. It has an RS-232 to JTAG protocol translator module referred to as the EC. There is also a target application board with a C8051F000, F005, F010, or F015 installed and with a large prototyping area. The kit also includes RS-232 and JTAG cables, and wall-mount power supply.