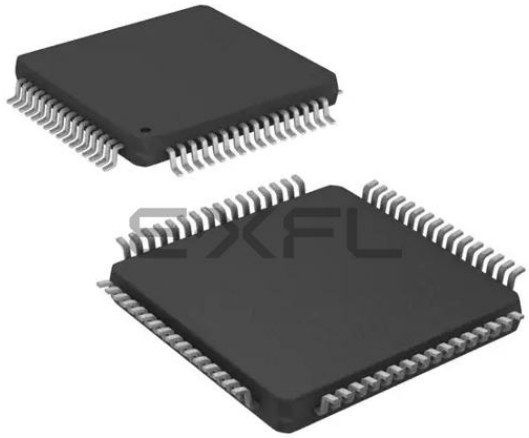


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Not For New Designs
Core Processor	8051
Core Size	8-Bit
Speed	25MHz
Connectivity	SMBus (2-Wire/I²C), SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, Temp Sensor, WDT
Number of I/O	32
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2.25K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 8x12b; D/A 2x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-TQFP
Supplier Device Package	64-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f005-gq

TABLE OF CONTENTS

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

1.1.3. Additional Features

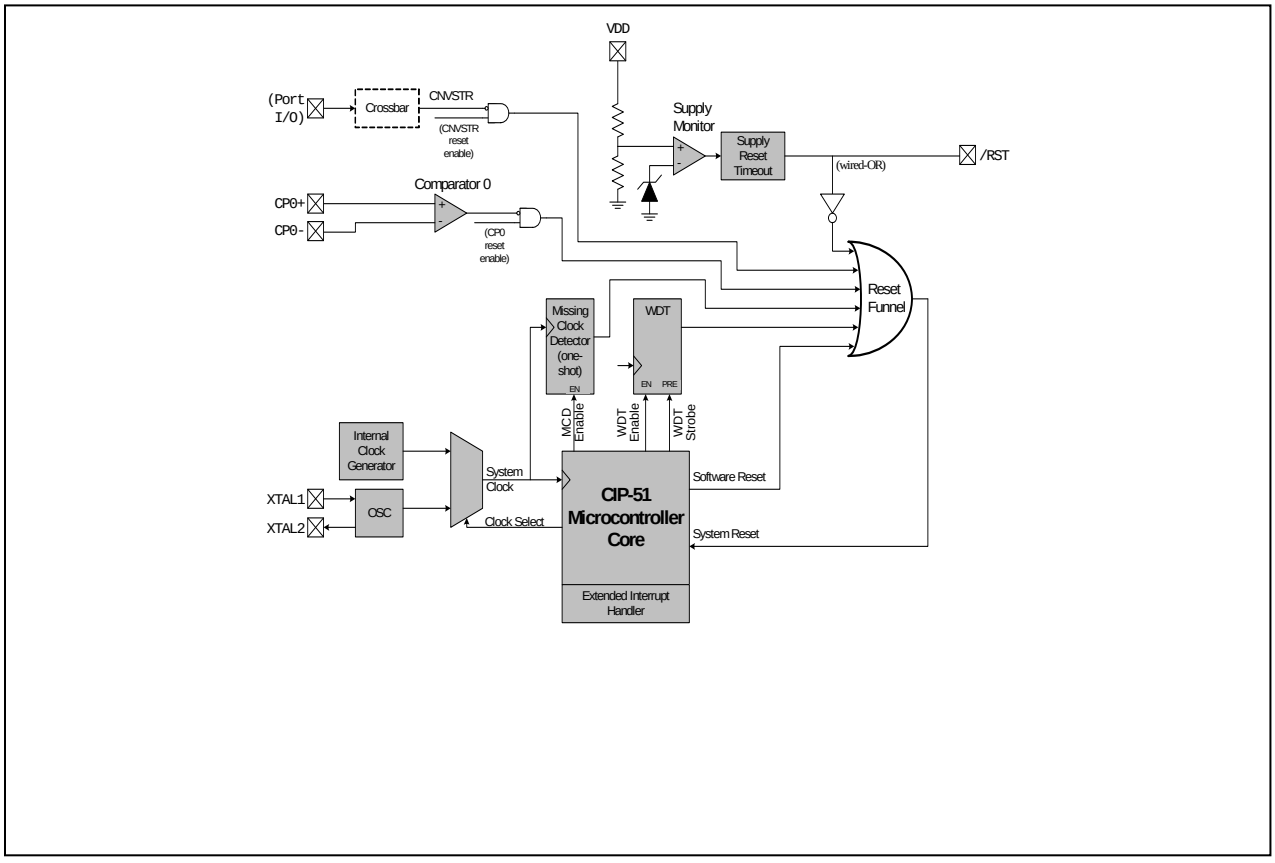
The C8051F000 MCU family has several key enhancements both inside and outside the CIP-51 core to improve its overall performance and ease of use in the end applications.

The extended interrupt handler provides 21 interrupt sources into the CIP-51 (as opposed to 7 for the standard 8051), allowing the numerous analog and digital peripherals to interrupt the controller. An interrupt driven system requires less intervention by the MCU, giving it more effective throughput. The extra interrupt sources are very useful when building multi-tasking, real-time systems.

There are up to seven reset sources for the MCU: an on-board VDD monitor, a Watchdog Timer, a missing clock detector, a voltage level detection from Comparator 0, a forced software reset, the CNVSTR pin, and the /RST pin. The /RST pin is bi-directional, accommodating an external reset, or allowing the internally generated POR to be output on the /RST pin. Each reset source except for the VDD monitor and Reset Input Pin may be disabled by the user in software. The WDT may be permanently enabled in software after a power-on reset during MCU initialization.

The MCU has an internal, stand alone clock generator which is used by default as the system clock after any reset. If desired, the clock source may be switched on the fly to the external oscillator, which can use a crystal, ceramic resonator, capacitor, RC, or external clock source to generate the system clock. This can be extremely useful in low power applications, allowing the MCU to run from a slow (power saving) external crystal source, while periodically switching to the fast (up to 16MHz) internal oscillator as needed.

Figure 1.5. On-Board Clock and Reset

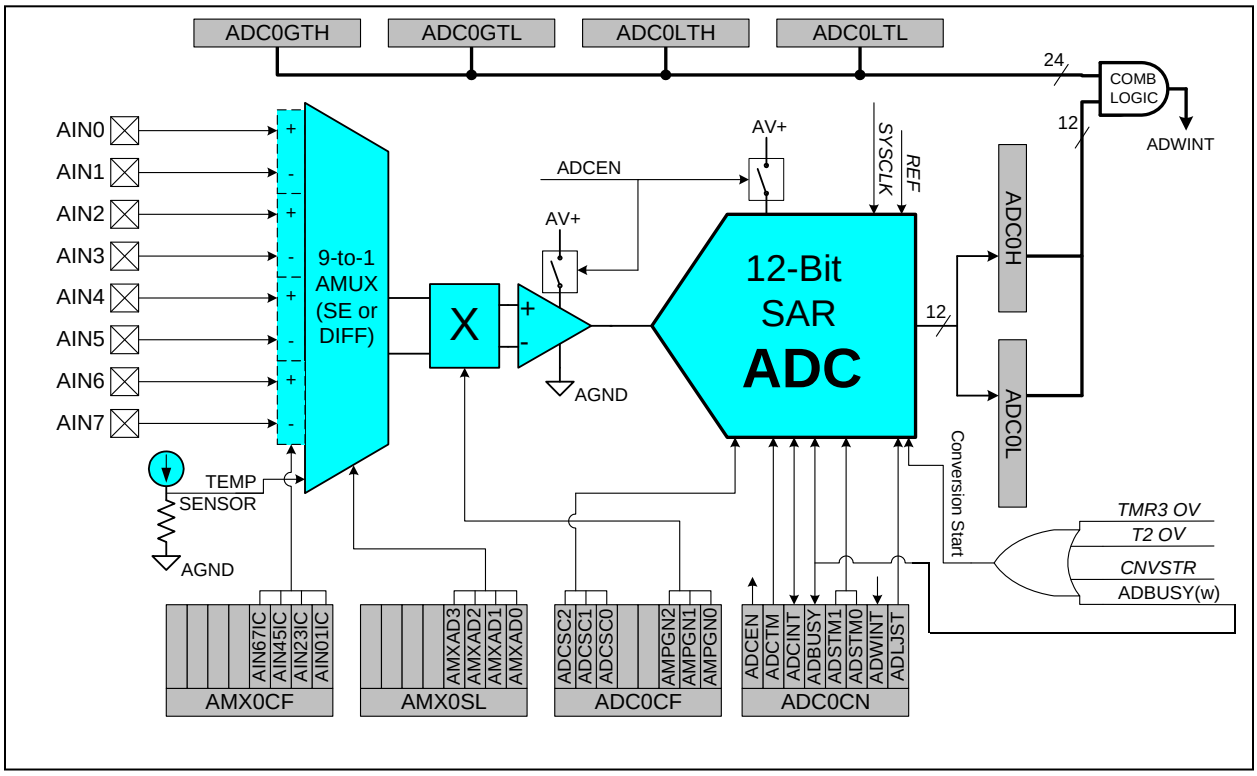


Name	Pin Numbers			Type	Description
	F000 F005 F010 F015	F001 F006 F011 F016	F002 F007 F012 F017		
AIN6	13	10		A In	Analog Mux Channel Input 6. (See ADC Specification for complete description).
AIN7	14	11		A In	Analog Mux Channel Input 7. (See ADC Specification for complete description).
P0.0	39	31	19	D I/O	Port0 Bit0. (See the Port I/O Sub-System section for complete description).
P0.1	42	34	22	D I/O	Port0 Bit1. (See the Port I/O Sub-System section for complete description).
P0.2	47	35	23	D I/O	Port0 Bit2. (See the Port I/O Sub-System section for complete description).
P0.3	48	36	24	D I/O	Port0 Bit3. (See the Port I/O Sub-System section for complete description).
P0.4	49	37	25	D I/O	Port0 Bit4. (See the Port I/O Sub-System section for complete description).
P0.5	50	38	26	D I/O	Port0 Bit5. (See the Port I/O Sub-System section for complete description).
P0.6	55	39	27	D I/O	Port0 Bit6. (See the Port I/O Sub-System section for complete description).
P0.7	56	40	28	D I/O	Port0 Bit7. (See the Port I/O Sub-System section for complete description).
P1.0	38	30		D I/O	Port1 Bit0. (See the Port I/O Sub-System section for complete description).
P1.1	37	29		D I/O	Port1 Bit1. (See the Port I/O Sub-System section for complete description).
P1.2	36	28		D I/O	Port1 Bit2. (See the Port I/O Sub-System section for complete description).
P1.3	35	26		D I/O	Port1 Bit3. (See the Port I/O Sub-System section for complete description).
P1.4	34	25		D I/O	Port1 Bit4. (See the Port I/O Sub-System section for complete description).
P1.5	32	24		D I/O	Port1 Bit5. (See the Port I/O Sub-System section for complete description).
P1.6	60	42		D I/O	Port1 Bit6. (See the Port I/O Sub-System section for complete description).
P1.7	59	41		D I/O	Port1 Bit7. (See the Port I/O Sub-System section for complete description).
P2.0	33			D I/O	Port2 Bit0. (See the Port I/O Sub-System section for complete description).
P2.1	27			D I/O	Port2 Bit1. (See the Port I/O Sub-System section for complete description).
P2.2	54			D I/O	Port2 Bit2. (See the Port I/O Sub-System section for complete description).
P2.3	53			D I/O	Port2 Bit3. (See the Port I/O Sub-System section for complete description).
P2.4	52			D I/O	Port2 Bit4. (See the Port I/O Sub-System section for complete description).
P2.5	51			D I/O	Port2 Bit5. (See the Port I/O Sub-System section for complete description).
P2.6	44			D I/O	Port2 Bit6. (See the Port I/O Sub-System section for complete description).
P2.7	43			D I/O	Port2 Bit7. (See the Port I/O Sub-System section for complete description).
P3.0	26			D I/O	Port3 Bit0. (See the Port I/O Sub-System section for complete description).
P3.1	25			D I/O	Port3 Bit1. (See the Port I/O Sub-System section for complete description).
P3.2	24			D I/O	Port3 Bit2. (See the Port I/O Sub-System section for complete description).
P3.3	23			D I/O	Port3 Bit3. (See the Port I/O Sub-System section for complete description).
P3.4	58			D I/O	Port3 Bit4. (See the Port I/O Sub-System section for complete description).
P3.5	57			D I/O	Port3 Bit5. (See the Port I/O Sub-System section for complete description).
P3.6	46			D I/O	Port3 Bit6. (See the Port I/O Sub-System section for complete description).
P3.7	45			D I/O	Port3 Bit7. (See the Port I/O Sub-System section for complete description).

5. ADC (12-Bit, C8051F000/1/2/5/6/7 Only)

The ADC subsystem for the C8051F000/1/2/5/6/7 consists of a 9-channel, configurable analog multiplexer (AMUX), a programmable gain amplifier (PGA), and a 100ksps, 12-bit successive-approximation-register ADC with integrated track-and-hold and programmable window detector (see block diagram in Figure 5.1). The AMUX, PGA, Data Conversion Modes, and Window Detector are all configurable under software control via the Special Function Register's shown in Figure 5.1. The ADC subsystem (ADC, track-and-hold and PGA) is enabled only when the ADCEN bit in the ADC Control register (ADC0CN, Figure 5.7) is set to 1. The ADC subsystem is in low power shutdown when this bit is 0. The Bias Enable bit (BIASE) in the REF0CN register (see Figure 9.2) must be set to 1 in order to supply bias to the ADC.

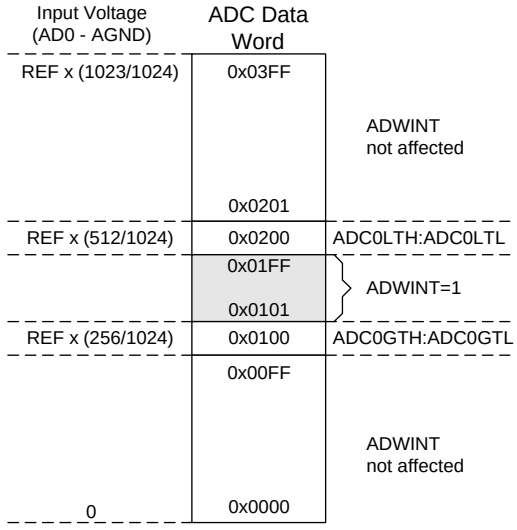
Figure 5.1. 12-Bit ADC Functional Block Diagram



5.1. Analog Multiplexer and PGA

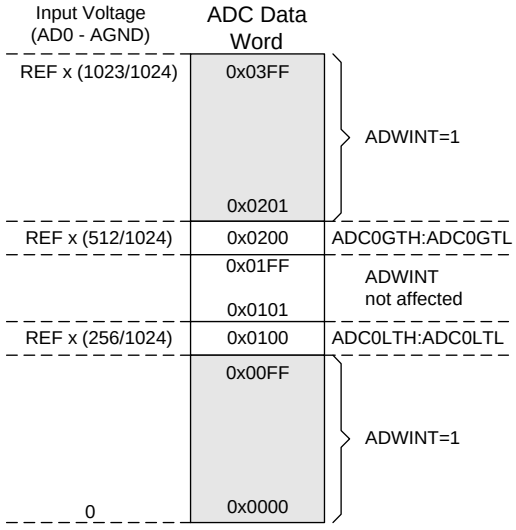
Eight of the AMUX channels are available for external measurements while the ninth channel is internally connected to an on-board temperature sensor (temperature transfer function is shown in Figure 5.3). Note that the PGA gain is applied to the temperature sensor reading. AMUX input pairs can be programmed to operate in either the differential or single-ended mode. This allows the user to select the best measurement technique for each input channel, and even accommodates mode changes "on-the-fly". The AMUX defaults to all single-ended inputs upon reset. There are two registers associated with the AMUX: the Channel Selection register AMX0SL (Figure 5.5), and the Configuration register AMX0CF (Figure 5.4). The table in Figure 5.5 shows AMUX functionality by channel for each possible configuration. The PGA amplifies the AMUX output signal by an amount determined by the AMPGN2-0 bits in the ADC Configuration register, ADC0CF (Figure 5.6). The PGA can be software-programmed for gains of 0.5, 1, 2, 4, 8 or 16. It defaults to unity gain on reset.

Figure 6.14. 10-Bit ADC Window Interrupt Examples, Right Justified Data



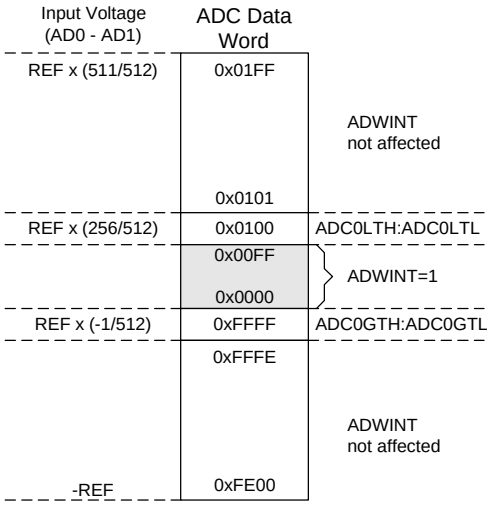
Given:
AMX0SL = 0x00, AMX0CF = 0x00, ADLJST = 0,
ADC0LTH:ADC0LTL = 0x0200,
ADC0GTH:ADC0GTL = 0x0100.

An ADC End of Conversion will cause an ADC Window Compare Interrupt (ADWINT=1) if the resulting ADC Data Word is < 0x0200 and > 0x0100.



Given:
AMX0SL = 0x00, AMX0CF = 0x00, ADLJST = 0,
ADC0LTH:ADC0LTL = 0x0100,
ADC0GTH:ADC0GTL = 0x0200.

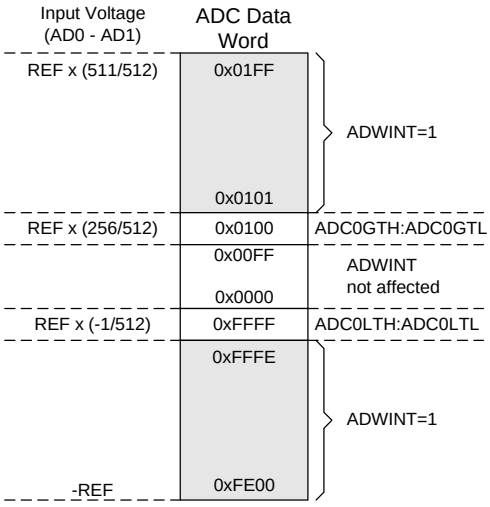
An ADC End of Conversion will cause an ADC Window Compare Interrupt (ADWINT=1) if the resulting ADC Data Word is < 0x0100 or > 0x0200.



Given:

AMX0SL = 0x00, AMX0CF = 0x01, ADLJST = 0,
ADC0LTH:ADC0LTL = 0x0100,
ADC0GTH:ADC0GTL = 0xFFFF.

An ADC End of Conversion will cause an ADC Window Compare Interrupt (ADWINT=1) if the resulting ADC Data Word is < 0x0100 and > 0xFFFF. (Two's Complement math, 0xFFFF = -1.)



Given:

AMX0SL = 0x00, AMX0CF = 0x01, ADLJST = 0,
ADC0LTH:ADC0LTH = 0xFFFF,
ADC0GTH:ADC0GTL = 0x0100.

An ADC End of Conversion will cause an ADC Window Compare Interrupt (ADWINT=1) if the resulting ADC Data Word is < 0xFFFF or > 0x0100. (Two's Complement math, 0xFFFF = -1.)

7. DACs, 12 BIT VOLTAGE MODE

The C8051F000 MCU family has two 12-bit voltage-mode Digital to Analog Converters. Each DAC has an output swing of 0V to VREF-1LSB for a corresponding input code range of 0x000 to 0xFFF. Using DAC0 as an example, the 12-bit data word is written to the low byte (DAC0L) and high byte (DAC0H) data registers. Data is latched into DAC0 after a write to the corresponding DAC0H register, **so the write sequence should be DAC0L followed by DAC0H** if the full 12-bit resolution is required. The DAC can be used in 8-bit mode by initializing DAC0L to the desired value (typically 0x00), and writing data to only DAC0H with the data shifted to the left. DAC0 Control Register (DAC0CN) provides a means to enable/disable DAC0 and to modify its input data formatting.

The DAC0 enable/disable function is controlled by the DAC0EN bit (DAC0CN.7). Writing a 1 to DAC0EN enables DAC0 while writing a 0 to DAC0EN disables DAC0. While disabled, the output of DAC0 is maintained in a high-impedance state, and the DAC0 supply current falls to 1μA or less. Also, the Bias Enable bit (BIASE) in the REF0CN register (see Figure 9.2) must be set to 1 in order to supply bias to DAC0. The voltage reference for DAC0 must also be set properly (see Section 9).

In some instances, input data should be shifted prior to a DAC0 write operation to properly justify data within the DAC input registers. This action would typically require one or more load and shift operations, adding software overhead and slowing DAC throughput. To alleviate this problem, the data-formatting feature provides a means for the user to program the orientation of the DAC0 data word within data registers DAC0H and DAC0L. The three DAC0DF bits (DAC0CN.[2:0]) allow the user to specify one of five data word orientations as shown in the DAC0CN register definition.

DAC1 is functionally the same as DAC0 described above. The electrical specifications for both DAC0 and DAC1 are given in Table 7.1.

Figure 7.1. DAC Functional Block Diagram

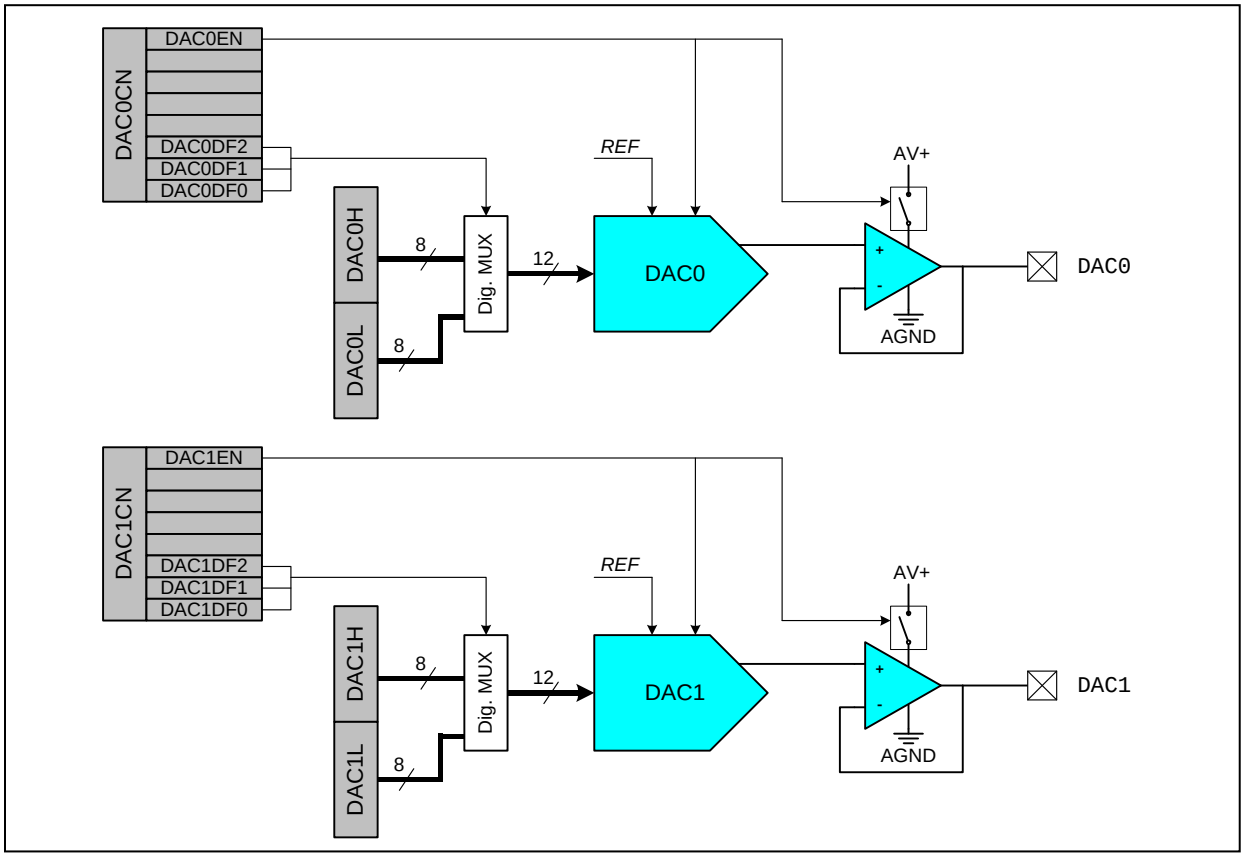


Figure 7.2. DAC0H: DAC0 High Byte Register

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: 0xD3

Bits7-0: DAC0 Data Word Most Significant Byte.

Figure 7.3. DAC0L: DAC0 Low Byte 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: 0xD2

Bits7-0: DAC0 Data Word Least Significant Byte.

Figure 7.4. DAC0CN: DAC0 Control Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
DAC0EN	-	-	-	-	DAC0DF2	DAC0DF1	DAC0DF0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xD4

Bit7: DAC0EN: DAC0 Enable Bit
0: DAC0 Disabled. DAC0 Output pin is disabled; DAC0 is in low power shutdown mode.
1: DAC0 Enabled. DAC0 Output is pin active; DAC0 is operational.
Bits6-3: UNUSED. Read = 0000b; Write = don't care
Bits2-0: DAC0DF2-0: DAC0 Data Format Bits
000: The most significant nybble of the DAC0 Data Word is in DAC0H[3:0], while the least significant byte is in DAC0L.

DAC0H								DAC0L							
				MSB											LSB

001: The most significant 5-bits of the DAC0 Data Word is in DAC0H[4:0], while the least significant 7-bits is in DAC0L[7:1].

DAC0H								DAC0L							
			MSB											LSB	

010: The most significant 6-bits of the DAC0 Data Word is in DAC0H[5:0], while the least significant 6-bits is in DAC0L[7:2].

DAC0H								DAC0L							
		MSB											LSB		

011: The most significant 7-bits of the DAC0 Data Word is in DAC0H[6:0], while the least significant 5-bits is in DAC0L[7:3].

DAC0H								DAC0L							
	MSB											LSB			

1xx: The most significant byte of the DAC0 Data Word is in DAC0H, while the least significant nybble is in DAC0L[7:4].

DAC0H								DAC0L							
MSB											LSB				

10.3. SPECIAL FUNCTION REGISTERS

The direct-access data memory locations from 0x80 to 0xFF constitute the special function registers (SFRs). The SFRs provide control and data exchange with the CIP-51's resources and peripherals. The CIP-51 duplicates the SFRs found in a typical 8051 implementation as well as implementing additional SFRs used to configure and access the sub-systems unique to the MCU. This allows the addition of new functionality while retaining compatibility with the MCS-51™ instruction set. Table 10.3 lists the SFRs implemented in the CIP-51 System Controller.

The SFR registers are accessed any time the direct addressing mode is used to access memory locations from 0x80 to 0xFF. SFRs with addresses ending in 0x0 or 0x8 (e.g. P0, TCON, P1, SCON, IE, etc.) are bit-addressable as well as byte-addressable. All other SFRs are byte-addressable only. Unoccupied addresses in the SFR space are reserved for future use. Accessing these areas will have an indeterminate effect and should be avoided. Refer to the corresponding pages of the datasheet, as indicated in Table 10.3, for a detailed description of each register.

Table 10.2. Special Function Register Memory Map

F8	SPI0CN	PCA0H	PCA0CPH0	PCA0CPH1	PCA0CPH2	PCA0CPH3	PCA0CPH4	WDTCN
F0	B						EIP1	EIP2
E8	ADC0CN	PCA0L	PCA0CPL0	PCA0CPL1	PCA0CPL2	PCA0CPL3	PCA0CPL4	RSTSRC
E0	ACC	XBR0	XBR1	XBR2			EIE1	EIE2
D8	PCA0CN	PCA0MD	PCA0CPM0	PCA0CPM1	PCA0CPM2	PCA0CPM3	PCA0CPM4	
D0	PSW	REF0CN	DAC0L	DAC0H	DAC0CN	DAC1L	DAC1H	DAC1CN
C8	T2CON		RCAP2L	RCAP2H	TL2	TH2		SMB0CR
C0	SMB0CN	SMB0STA	SMB0DAT	SMB0ADR	ADC0GTL	ADC0GTH	ADC0LTL	ADC0LTH
B8	IP		AMX0CF	AMX0SL	ADC0CF		ADC0L	ADC0H
B0	P3	OSCXCN	OSCICN				FLSCL	FLACL***
A8	IE					PRT1IF		EMI0CN***
A0	P2				PRT0CF	PRT1CF	PRT2CF	PRT3CF
98	SCON	SBUF	SPI0CFG	SPI0DAT		SPI0CKR	CPT0CN	CPT1CN
90	P1	TMR3CN	TMR3RLL	TMR3RLH	TMR3L	TMR3H		
88	TCON	TMOD	TL0	TL1	TH0	TH1	CKCON	PSCTL
80	P0	SP	DPL	DPH				PCON
	0(8)	1(9)	2(A)	3(B)	4(C)	5(D)	6(E)	7(F)

Bit Addressable

Table 10.3. Special Function Registers

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

Address	Register	Description	Page No.
0xE0	ACC	Accumulator	76
0xBC	ADC0CF	ADC Configuration	33*, 42**
0xE8	ADC0CN	ADC Control	34*, 45**
0xC5	ADC0GTH	ADC Greater-Than Data Word (High Byte)	36*, 47**
0xC4	ADC0GTL	ADC Greater-Than Data Word (Low Byte)	36*, 47**
0xBF	ADC0H	ADC Data Word (High Byte)	35*, 46**
0xBE	ADC0L	ADC Data Word (Low Byte)	35*, 46**

Figure 10.13. EIP1: Extended Interrupt Priority 1

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
PCP1R	PCP1F	PCP0R	PCP0F	PPCA0	PWADC0	PSMB0	PSPI0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xF6

Bit7: PCP1R: Comparator 1 (CP1) Rising Interrupt Priority Control.
This bit sets the priority of the CP1 interrupt.
0: CP1 rising interrupt set to low priority level.
1: CP1 rising interrupt set to high priority level.

Bit6: PCP1F: Comparator 1 (CP1) Falling Interrupt Priority Control.
This bit sets the priority of the CP1 interrupt.
0: CP1 falling interrupt set to low priority level.
1: CP1 falling interrupt set to high priority level.

Bit5: PCP0R: Comparator 0 (CP0) Rising Interrupt Priority Control.
This bit sets the priority of the CP0 interrupt.
0: CP0 rising interrupt set to low priority level.
1: CP0 rising interrupt set to high priority level.

Bit4: PCP0F: Comparator 0 (CP0) Falling Interrupt Priority Control.
This bit sets the priority of the CP0 interrupt.
0: CP0 falling interrupt set to low priority level.
1: CP0 falling interrupt set to high priority level.

Bit3: PPCA0: Programmable Counter Array (PCA0) Interrupt Priority Control.
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.

Bit2: PWADC0: ADC0 Window Comparator Interrupt Priority Control.
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.

Bit1: PSMB0: SMBus 0 Interrupt Priority Control.
This bit sets the priority of the SMBus interrupt.
0: SMBus interrupt set to low priority level.
1: SMBus interrupt set to high priority level.

Bit0: PSPI0: Serial Peripheral Interface 0 Interrupt Priority Control.
This bit sets the priority of the SPI0 interrupt.
0: SPI0 interrupt set to low priority level.
1: SPI0 interrupt set to high priority level.

10.5. Power Management Modes

The CIP-51 core has two software programmable power management modes: Idle and Stop. Idle mode halts the CPU while leaving the external peripherals and internal clocks active. In Stop mode, the CPU is halted, all interrupts and timers (except the Missing Clock Detector) are inactive, and the system clock is stopped. 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. Figure 10.15 describes the Power Control Register (PCON) used to control the CIP-51's power management modes.

Although the CIP-51 has Idle and Stop modes built in (as with any standard 8051 architecture), power management of the entire MCU is better accomplished by enabling/disabling individual peripherals as needed. Each analog peripheral can be disabled when not in use and put into low power mode. Digital peripherals, such as timers or serial buses, draw little power whenever they are not in use. Turning off the oscillator saves even more power, but requires a reset to restart the MCU.

10.5.1. Idle Mode

Setting the Idle Mode Select bit (PCON.0) causes the CIP-51 to halt the CPU and enter Idle mode as soon as the instruction that sets the bit completes. 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 or /RST is asserted. The assertion of an enabled interrupt will cause the Idle Mode Selection bit (PCON.0) to be cleared and the CPU will 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.

Note: 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. Any instructions that set the IDLE bit should be followed by an instruction that has 2 or more op-code 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 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 13.8 Watchdog Timer for more information on the use and configuration of the WDT.

10.5.2. Stop Mode

Setting the Stop Mode Select bit (PCON.1) causes the CIP-51 to enter Stop mode as soon as the instruction that sets the bit completes. In Stop mode, the CPU and oscillators are stopped, effectively shutting down all digital peripherals. Each analog peripheral must be shut down individually prior to entering Stop Mode. Stop mode can only be terminated by an internal or external reset. On reset, the CIP-51 performs the normal reset sequence and begins program execution at address 0x0000.

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 100µs, 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 C0RSEF flag (RSTSRC.5). Comparator 0 should be enabled using CPT0CN.7 (see Figure 8.3) at least 20µs prior to writing to C0RSEF 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 C0RSEF 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.

15.3. General Purpose Port I/O

Each MCU has four byte-wide, bi-directional parallel ports that can be used general purpose I/O. Each port is accessed through a corresponding special function register (SFR) that is both byte addressable and bit addressable. When writing to a port, the value written to the SFR is latched to maintain the output data value at each pin. When reading, the logic levels of the port's input pins are returned regardless of the XBRn settings (i.e. even when the pin is assigned to another signal by the Crossbar, the Port Register can always still read its corresponding Port I/O pin). The exception to this is the execution of the *read-modify-write* instructions. The *read-modify-write* instructions when operating on a port SFR are the following: ANL, ORL, XRL, JBC, CPL, INC, DEC, DJNZ and MOV, CLR or SET, when the destination is an individual bit in a port SFR. For these instructions, the value of the port register (not the pin) is read, modified, and written back to the SFR.

15.4. Configuring Ports Which are not Pinned Out

P2 and P3 are not pinned out on the F001/06/11/16. P1, P2, and P3 are not pinned out on the F002/07/12/17. These port registers (and corresponding interrupts, where applicable) are still available for software use in these reduced pin count MCUs. Whether used or not in software, it is recommended not to let these port drivers go to high impedance state. This is prevented after reset by having the weak pull-ups enabled as described in the XBR2 register. It is recommended that each output driver for ports not pinned out should be configured as push-pull using the corresponding PRTnCF register. This will inhibit a high impedance state even if the weak pull-up is disabled.

Figure 15.6. P0: Port0 Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0	11111111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0 (bit addressable)	SFR Address: 0x80

Bits7-0: P0.[7:0]
(Write – Output appears on I/O pins per XBR0, XBR1, and XBR2 Registers)
0: Logic Low Output.
1: Logic High Output (high-impedance if corresponding PRT0CF.n bit = 0)
(Read – Regardless of XBR0, XBR1, and XBR2 Register settings).
0: P0.n pin is logic low.
1: P0.n pin is logic high.

Figure 15.7. PRT0CF: Port0 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: 0xA4

Bits7-0: PRT0CF.[7:0]: Output Configuration Bits for P0.7-P0.0 (respectively)
0: Corresponding P0.n Output mode is Open-Drain.
1: Corresponding P0.n Output mode is Push-Pull.

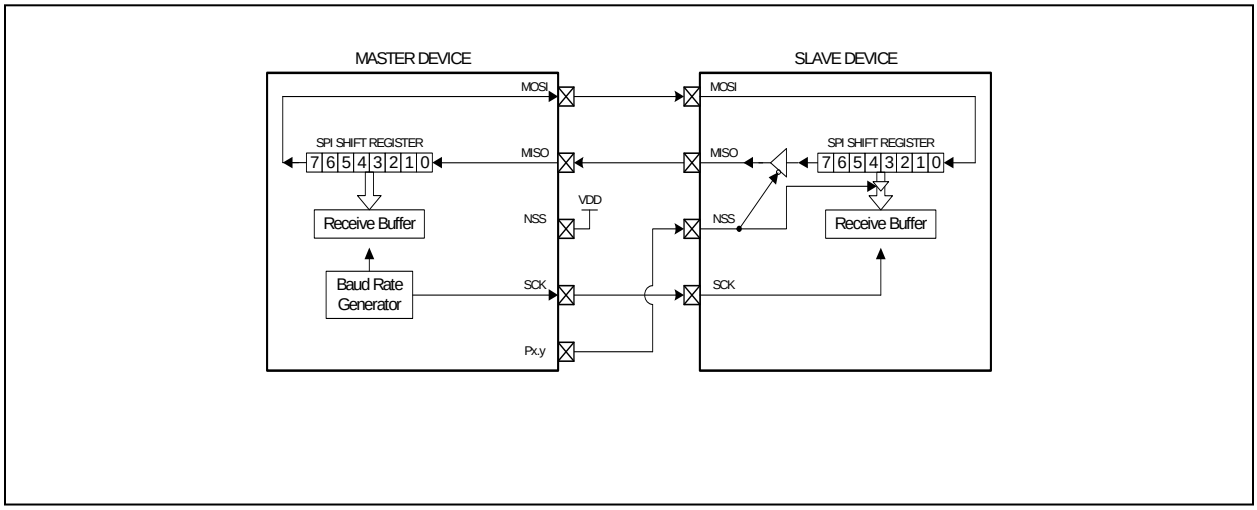
(Note: When SDA, SCL, and RX appear on any of the P0 I/O, each are open-drain regardless of the value of PRT0CF).

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, SPI0CN.1). Writing a byte of data to the SPI data register (SPI0DAT) 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 (SPI0CN.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 (SPI0CFG.[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 SPI0DAT is attempted during a data transfer, the WCOL flag (SPI0CN.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 (SPI0CN.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 the 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, SPI0CN.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

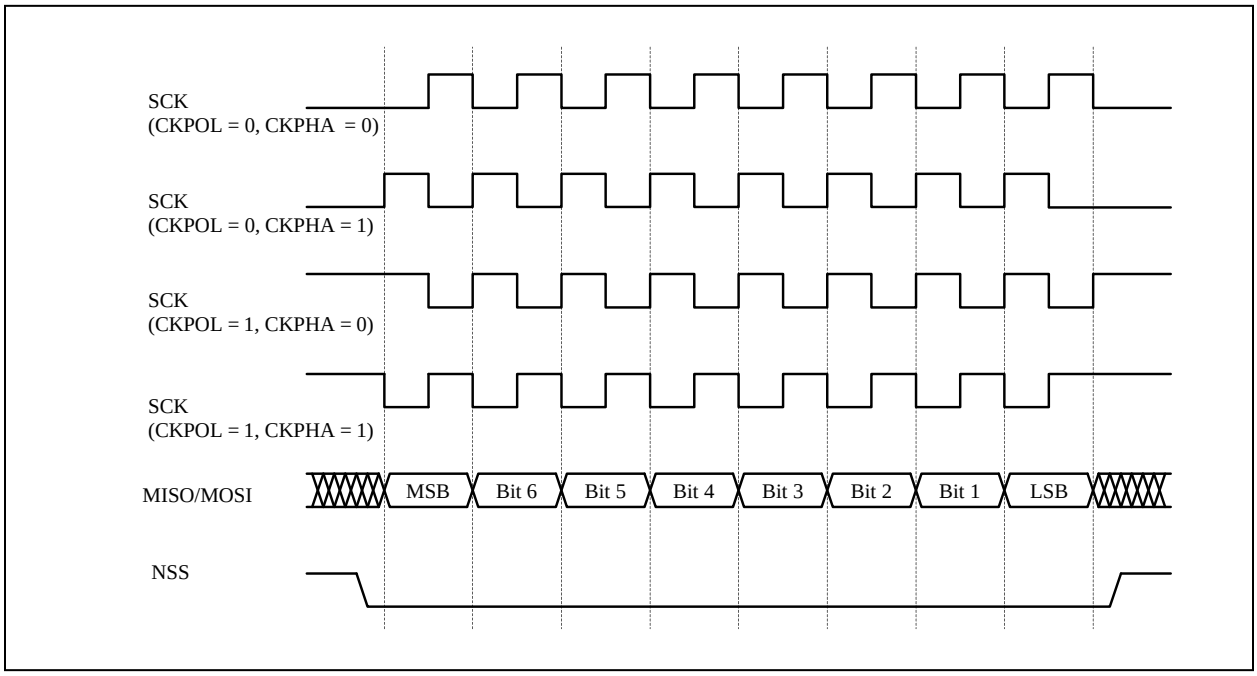
in an “off-line” state. In a multiple-master environment, the system controller should check the state of the SLVSEL flag (SPI0CN.2) to ensure the bus is free before setting the MSTEN bit and initiating a data transfer.

17.3. Serial Clock Timing

As shown in Figure 17.4, four combinations of serial clock phase and polarity can be selected using the clock control bits in the SPI Configuration Register (SPI0CFG). The CKPHA bit (SPI0CFG.7) selects one of two clock phases (edge used to latch the data). The CKPOL bit (SPI0CFG.6) selects between an active-high or active-low clock. Both master and slave devices must be configured to use the same clock phase and polarity. Note: the SPI should be disabled (by clearing the SPIEN bit, SPI0CN.0) while changing the clock phase and polarity.

The SPI Clock Rate Register (SPI0CKR) as shown in Figure 17.7 controls the master mode serial clock frequency. This register is ignored when operating in slave mode.

Figure 17.4. Data/Clock Timing Diagram



17.4. SPI Special Function Registers

The SPI is accessed and controlled through four special function registers in the system controller: SPI0CN Control Register, SPI0DAT Data Register, SPI0CFG Configuration Register, and SPI0CKR Clock Rate Register. The four special function registers related to the operation of the SPI Bus are described in the following section.

Figure 17.5. SPI0CFG: SPI Configuration Register

R/W	R/W	R	R	R	R/W	R/W	R/W	Reset Value
CKPHA	CKPOL	BC2	BC1	BC0	SPIFRS2	SPIFRS1	SPIFRS0	00000111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0x9A

Bit7: CKPHA: SPI Clock Phase.
This bit controls the SPI clock phase.
0: Data sampled on first edge of SCK period.
1: Data sampled on second edge of SCK period.

Bit6: CKPOL: SPI Clock Polarity.
This bit controls the SPI clock polarity.
0: SCK line low in idle state.
1: SCK line high in idle state.

Bits5-3: BC2-BC0: SPI Bit Count.
Indicates which of the up to 8 bits of the SPI word have been transmitted.

BC2-BC0			Bit Transmitted
0	0	0	Bit 0 (LSB)
0	0	1	Bit 1
0	1	0	Bit 2
0	1	1	Bit 3
1	0	0	Bit 4
1	0	1	Bit 5
1	1	0	Bit 6
1	1	1	Bit 7 (MSB)

Bits2-0: SPIFRS2-SPIFRS0: SPI Frame Size.
These three bits determine the number of bits to shift in/out of the SPI shift register during a data transfer in master mode. They are ignored in slave mode.

SPIFRS			Bits Shifted
0	0	0	1
0	0	1	2
0	1	0	3
0	1	1	4
1	0	0	5
1	0	1	6
1	1	0	7
1	1	1	8

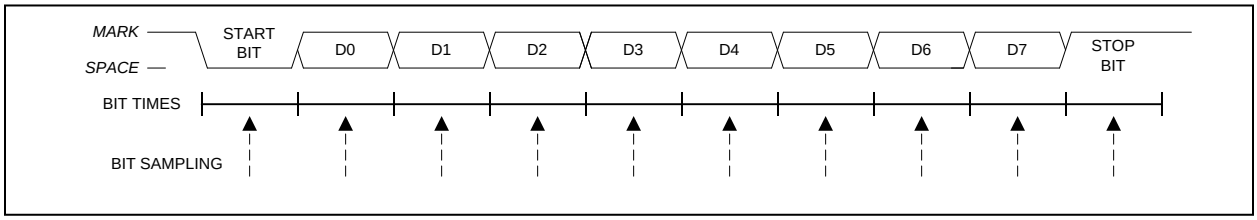
.

Mode 1 provides standard asynchronous, full duplex communication using a total of 10 bits per data byte: one start bit, eight data bits (LSB first), and one stop bit (see the timing diagram in Figure 18.4). Data are transmitted from the TX pin and received at the RX pin (see the interconnection diagram in Figure 18.5). On receive, the eight data bits are stored in SBUF and the stop bit goes into RB8 (SCON.2).

Data transmission begins when an instruction writes a data byte to the SBUF register. The TI Transmit Interrupt Flag (SCON.1) is set at the end of the transmission (the beginning of the stop-bit time). Data reception can begin any time after the REN Receive Enable bit (SCON.4) is set to logic 1. After the stop bit is received, the data byte will be loaded into the SBUF receive register if the following conditions are met: RI must be logic 0, and if SM2 is logic 1, the stop bit must be logic 1.

If these conditions are met, the eight bits of data are stored in SBUF, the stop bit is stored in RB8, and the RI flag is set. If these conditions are not met, SBUF and RB8 will not be loaded and the RI flag will not be set. An interrupt will occur if enabled when either TI or RI is set.

Figure 18.4. UART Mode 1 Timing Diagram



The baud rate generated in Mode 1 is a function of timer overflow. The UART can use Timer 1 operating in *8-bit Counter/Timer with Auto-Reload Mode*, or Timer 2 operating in *Baud Rate Generator Mode* to generate the baud rate (note that the TX and RX clock sources are selected separately). On each timer overflow event (a rollover from all ones (0xFF for Timer 1, 0xFFFF for Timer 2) to zero), a clock is sent to the baud rate logic.

When Timer 1 is selected as a baud rate source, the SMOD bit (PCON.7) selects whether or not to divide the Timer 1 overflow rate by two. On reset, the SMOD bit is logic 0, thus selecting the lower speed baud rate by default. The SMOD bit affects the baud rate generated by Timer 1 as follows:

*Mode 1 Baud Rate = (1 / 32) * T1_OVERFLOWRATE (when the SMOD bit is set to logic 0).*
*Mode 1 Baud Rate = (1 / 16) * T1_OVERFLOWRATE (when the SMOD bit is set to logic 1).*

When Timer 2 is selected as a baud rate source, the baud rate generated by Timer 2 is as follows:

*Mode 1 Baud Rate = (1 / 16) * T2_OVERFLOWRATE.*

The Timer 1 overflow rate is determined by the Timer 1 clock source (T1CLK) and reload value (TH1). The frequency of T1CLK can be selected as SYSCLK, SYSCLK/12, or an external clock source. The Timer 1 overflow rate can be calculated as follows:

T1_OVERFLOWRATE = T1CLK / (256 – TH1).

For example, assume TMOD = 0x20.
If T1M (CKCON.4) is logic 1, then the above equation becomes:

T1_OVERFLOWRATE = (SYSCLK) / (256 – TH1).

If T1M (CKCON.4) is logic 0, then the above equation becomes:

T1_OVERFLOWRATE = (SYSCLK/12) / (256 – TH1).

20.1. Capture/Compare Modules

Each module can be configured to operate independently in one of four operation modes: Edge-triggered Capture, Software Timer, High Speed Output, or Pulse Width Modulator. Each module has Special Function Registers (SFRs) associated with it in the CIP-51 system controller. These registers are used to exchange data with a module and configure the module's mode of operation.

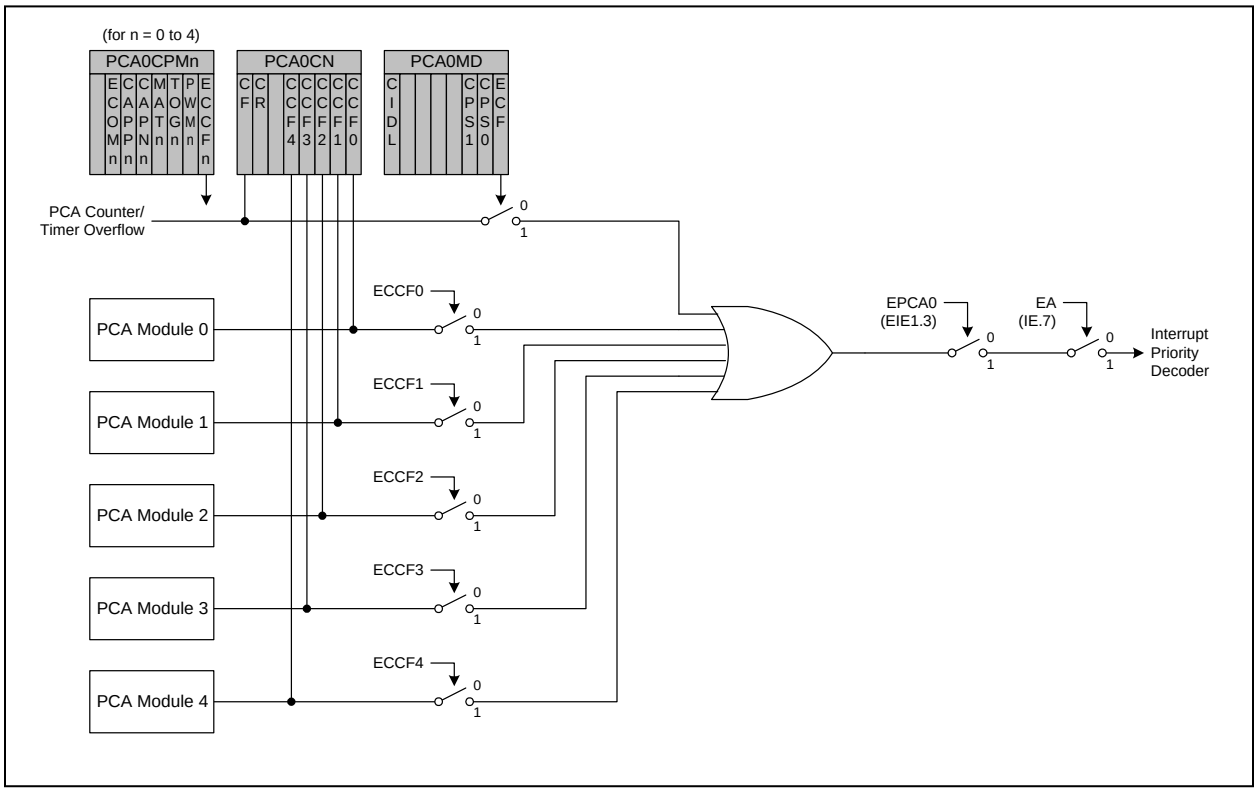
Table 20.1 summarizes the bit settings in the PCA0CPMn registers used to place the PCA capture/compare modules into different operating modes. Setting the ECCFn bit in a PCA0CPMn register enables the module's CCFn interrupt. Note: PCA0 interrupts must be globally enabled before individual CCFn interrupts are recognized. PCA0 interrupts are globally enabled by setting the EA bit (IE.7) and the EPCA0 bit (EIE1.3) to logic 1. See Figure 20.2 for details on the PCA interrupt configuration.

Table 20.1. PCA0CPM Register Settings for PCA Capture/Compare Modules

ECOM	CAPP	CAPN	MAT	TOG	PWM	ECCF	Operation Mode
X	1	0	0	0	0	X	Capture triggered by positive edge on CEXn
X	0	1	0	0	0	X	Capture triggered by negative edge on CEXn
X	1	1	0	0	0	X	Capture triggered by transition on CEXn
1	0	0	1	0	0	X	Software Timer
1	0	0	1	1	0	X	High Speed Output
1	0	0	X	0	1	X	Pulse Width Modulator

X = Don't Care

Figure 20.2. PCA Interrupt Block Diagram



20.1.4. Pulse Width Modulator Mode

All of the modules can be used independently to generate pulse width modulated (PWM) outputs on their respective CEXn pin. The frequency of the output is dependent on the timebase for the PCA counter/timer. The duty cycle of the PWM output signal is varied using the module's PCA0CPLn capture/compare register. When the value in the low byte of the PCA counter/timer (PCA0L) is equal to the value in PCA0CPLn, the output on the CEXn pin will be set. When the count value in PCA0L overflows, the CEXn output will be reset (see Figure 20.6). Also, when the counter/timer low byte (PCA0L) overflows from 0xFF to 0x00, PCA0CPLn is reloaded automatically with the value stored in the PCA0CPHn without software intervention. It is good practice to write to PCA0CPHn instead of PCA0CPLn to avoid glitches in the digital comparator. Setting the ECOMn and PWMn bits in the PCA0CPMn register enables Pulse Width Modulator mode.

Figure 20.6. PCA PWM Mode Diagram

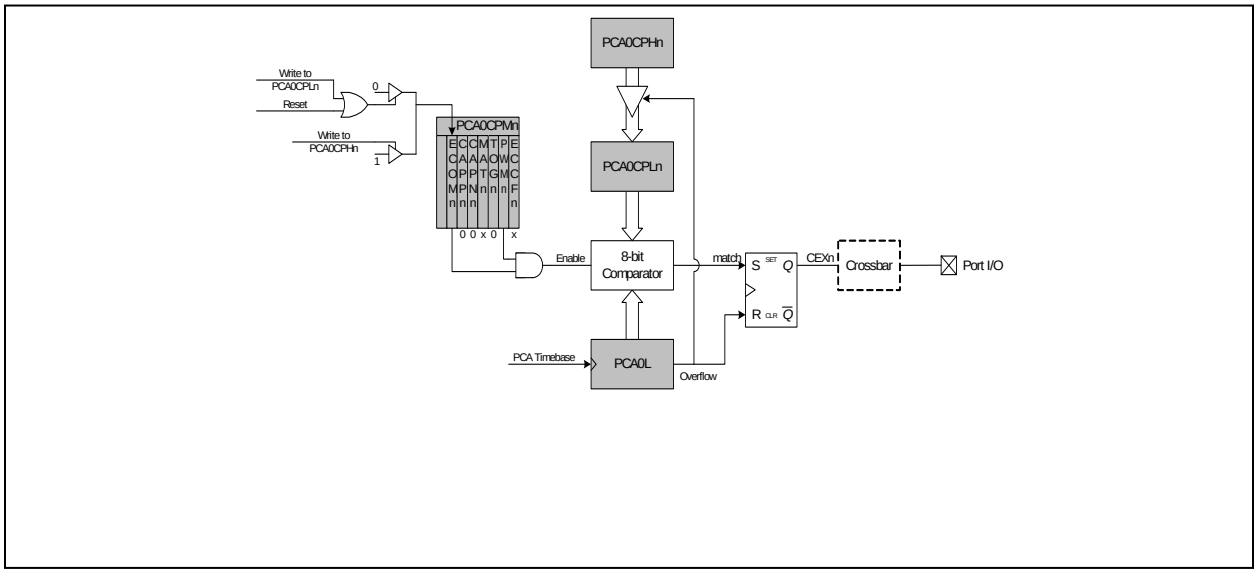


Figure 20.10. PCA0CPMn: PCA Capture/Compare Registers

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.