PRELIMINARY

C8051F206
C8051F220/1/6
C8051F230/1/6

Mixed-Signal 8KB ISP FLASH MCU Family

ANALOG PERIPHERALS

- **SAR ADC**
  - 12-bit Resolution ('F206)
  - 8-Bit Resolution ('F220/1/6)
  - ±1/4 LSB INL (8-bit) and ±2 LSB INL (12-bit)
  - Up to 100ksps
  - Up to 32 Channel Input Multiplexer; Each Port I/O Pin can be an ADC Input
- **Two Comparators**
  - 16 Programmable Hysteresis States
  - Configurable to Generate Interrupts or Reset
- **VDD Monitor** and **Brown-out Detector**

ON-CHIP JTAG DEBUG

- On-Chip Debug Circuitry Facilitates Full Speed, Non-intrusive In-system Debug (**No Emulator Required!**)
- Provides Breakpoints, Single-Stepping, Watchpoints, Stack Monitor
- Inspect/Modify Memory and Registers
- Superior Performance to Emulation Systems Using ICE-Chips, Target Pods, and Sockets
- Complete, Low Cost Development Kit: $99

HIGH SPEED 8051 µC Core

- Pipe-lined Instruction Architecture; Executes 70% of Instructions in 1 or 2 System Clocks
- Up to 25MIPS Throughput with 25MHz Clock
- Expanded Interrupt Handler

MEMORY

- 256 Bytes Internal Data RAM
- 1024 Bytes XRAM (available on 'F206/226/236)
- 8k Bytes FLASH; In-System Programmable in 512 byte Sectors

DIGITAL PERIPHERALS

- Four byte wide Port I/O; All are 5V tolerant
- Hardware UART and SPI bus
- 3 General Purpose 16-Bit Counter/Timers
- Dedicated Watch-Dog Timer
- Bi-directional Reset
- System Clock: Internal Programmable Oscillator, External Crystal, External RC, or External Clock

SUPPLY VOLTAGE ....2.7V to 3.6V

- Typical Operating Current: 10mA @ 25MHz
- Multiple Power Saving Sleep and Shutdown Modes

(48-Pin TQFP and 32-Pin LQFP Version Available)
Temperature Range: –40°C to +85°C

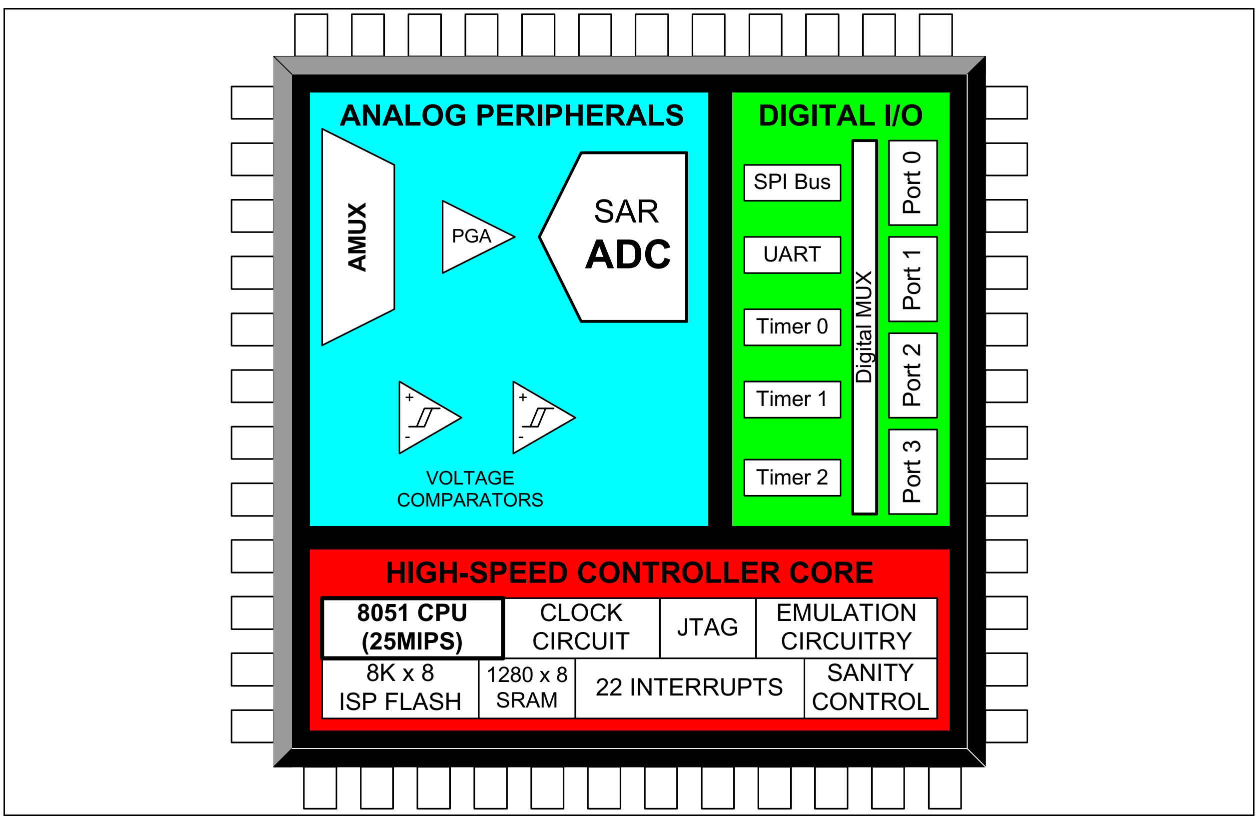


TABLE OF CONTENTS

1. SYSTEM OVERVIEW

The C8051F2xx is a family of fully integrated, mixed-signal System on a Chip MCU's available with a true 8-bit multi-channel ADC ('F220/1/6 and 'F206), or without an ADC ('F230/1/6). Each model features an 8051-compatible microcontroller core with 8kbytes of FLASH memory. There are also UART and SPI serial interfaces implemented in hardware (not "bit-banged" in user software). Products in this family feature 22 or 32 general purpose I/O pins, some of which can be used for assigned digital peripheral interface. Any pins may be configured for use as analog input to the analog-to-digital converter ('F220/1/6 and 'F206 only). (See the Product Selection Guide in Table 1.1.1 for a quick reference of each MCUs' feature set.)

Other features include an on-board VDD monitor, WDT, and clock oscillator. On-board FLASH memory can be reprogrammed in-circuit, and may also be used for non-volatile data storage. Integrated peripherals can also individually shut down any or all of the peripherals to conserve power. All parts have 256 bytes of SRAM. Also, an additional 1024 bytes of RAM is available in the 'F226/'F236.

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, run and halt commands. All analog and digital peripherals are fully functional when emulating using JTAG.

Each MCU is specified for 2.7V to 3.6V operation over the industrial temperature range (-45C to +85C) and is available in the 48-pin TFQP and 32-pin LFQP. The Port I/Os are tolerant for input signals up to 5V.

Table 1.1.1. Product Selection Guide

Part Number	MIPS (Peak)	FLASH Memory	RAM	SPI	UART	Timers (16-bit)	Digital Port I/O's	ADC Resolution (bits)	ADC Max Speed (ksps)	ADC Inputs	Voltage Comparators	Package
C8051F206	25	8k	1280	√	√	3	32	12	100	32	2	48TQFP
C8051F220	25	8k	256	√	√	3	32	8	100	32	2	48TQFP
C8051F221	25	8k	256	√	√	3	22	8	100	22	2	32LQFP
C8051F226	25	8k	1280	√	√	3	32	8	100	32	2	48TQFP
C8051F230	25	8k	256	√	√	3	32	-	-	-	2	48TQFP
C8051F231	25	8k	256	√	√	3	22	-	-	-	2	32LQFP
C8051F236	25	8k	1280	√	√	3	32	-	-	-	2	48TQFP

Figure 1.1. C8051F206, C8051F220 and C8051F226 Block Diagram (48 TQFP)

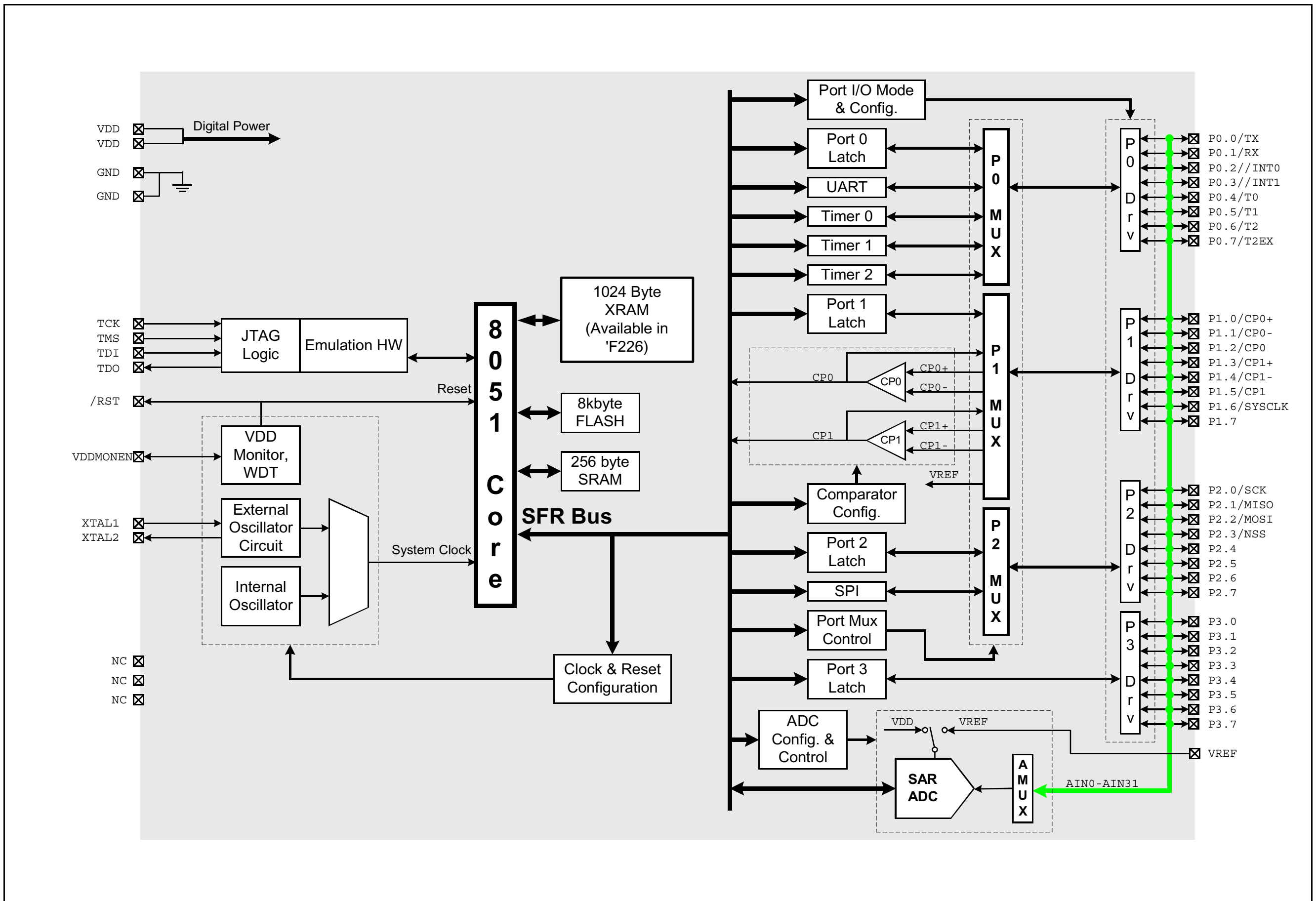


Figure 1.2 C8051F221 Block Diagram (32 LQFP)

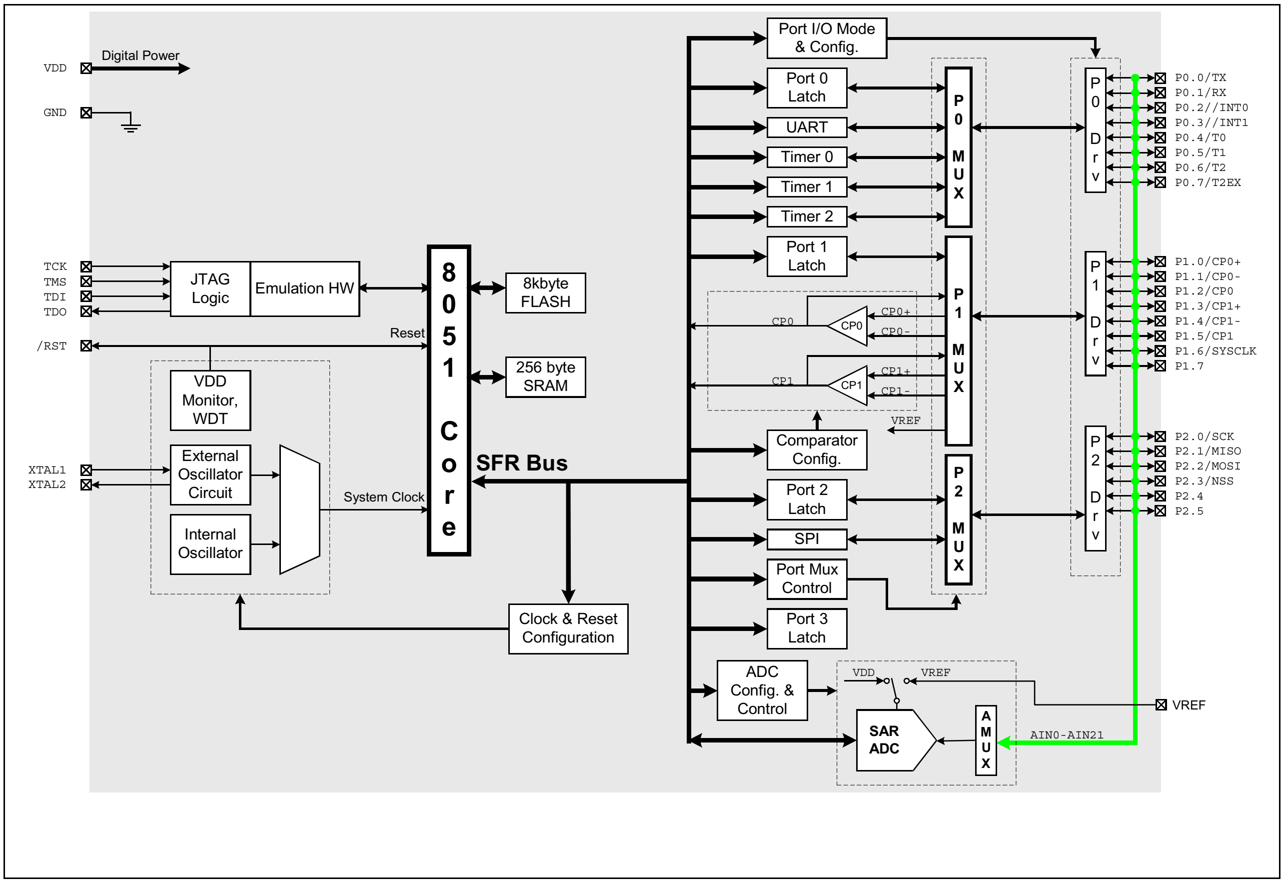


Figure 1.3 C8051F230 and C8051F236 Block Diagram (48 TQFP)

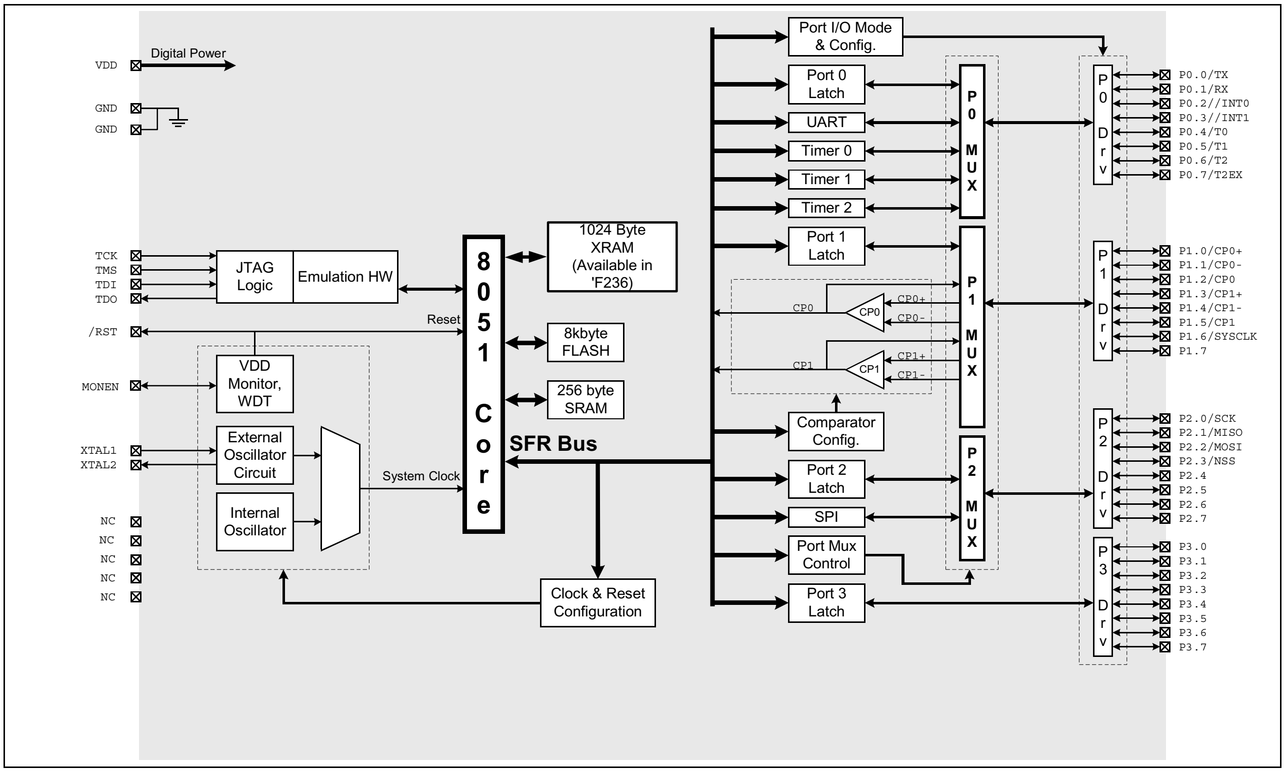
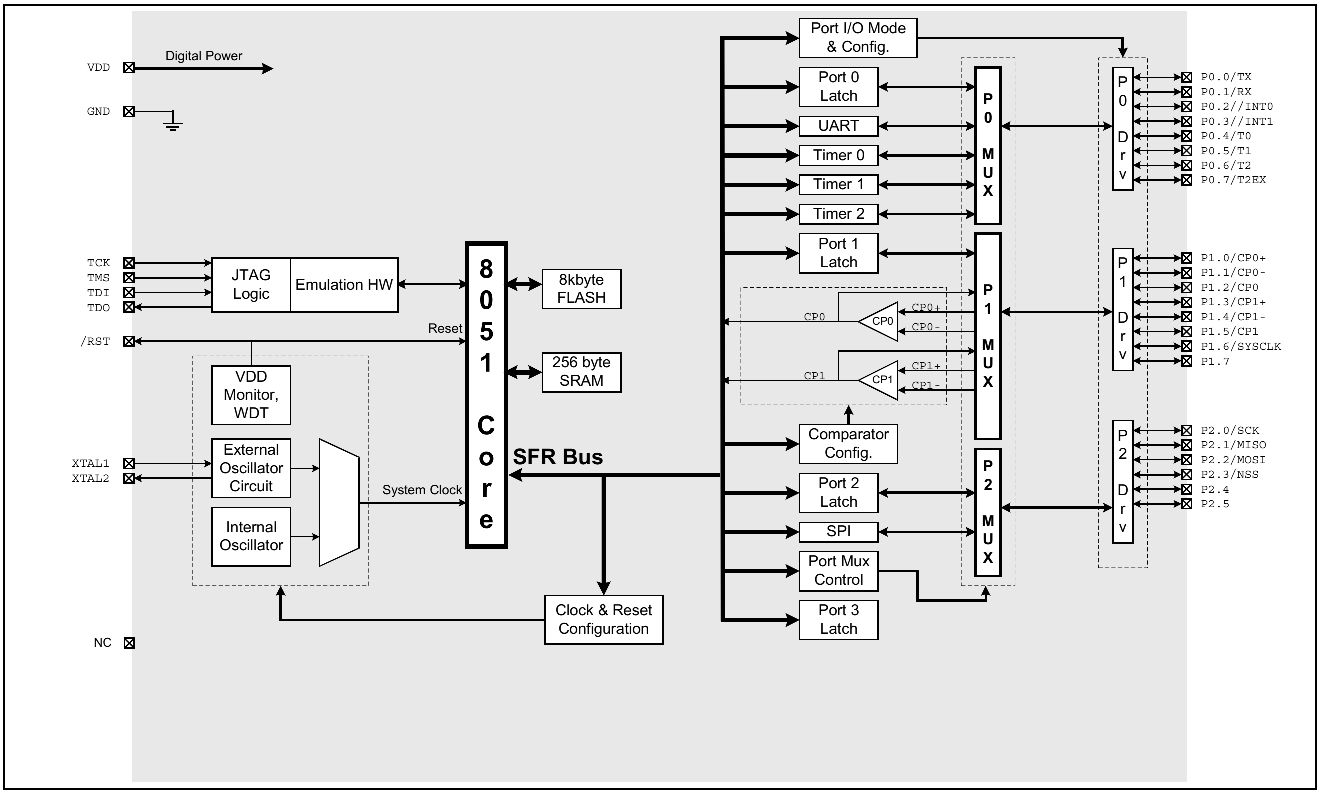


Figure 1.4 C8051F231 Block Diagram (32 LQFP)



1.1. CIP-51™ Microcontroller Core

1.1.1. Fully 8051 Compatible

The C8051F206, C8051F220/1/6 and C8051F230/1/6 utilize Cygnal's proprietary CIP-51 microcontroller core. 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 core contains the peripherals included with a standard 8052, including three 16-bit counter/timers, a full-duplex UART, 256 bytes of internal RAM, an optional 1024 bytes of XRAM, 128 byte Special Function Register (SFR) address space, and four byte-wide I/O Ports.

1.1.2. Improved Throughput

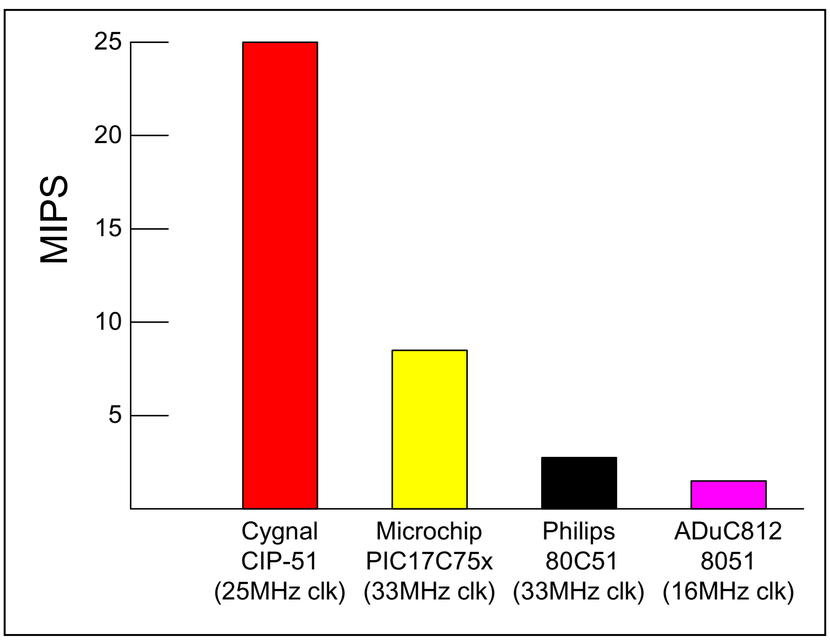
The CIP-51 employs a pipelined architecture that greatly increases its instruction throughput over the standard 8051 architecture. In a standard 8051, all instructions except for MUL and DIV take 12 or 24 system clock cycles to execute with a maximum system clock of 12MHz. By contrast, the CIP-51 core executes 70% of its instructions in one or two system clock cycles, with only four instructions taking more than four system clock cycles.

The CIP-51 has a total of 109 instructions. The number of instructions versus the system clock cycles to execute them is as follows:

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

With the CIP-51's maximum system clock at 25MHz, it has a peak throughput of 25MIPS. Figure 1.5 shows a comparison of peak throughputs of various 8-bit microcontroller cores with their maximum system clocks.

Figure 1.5. Comparison of Peak MCU Throughputs



1.1.3. Additional Features

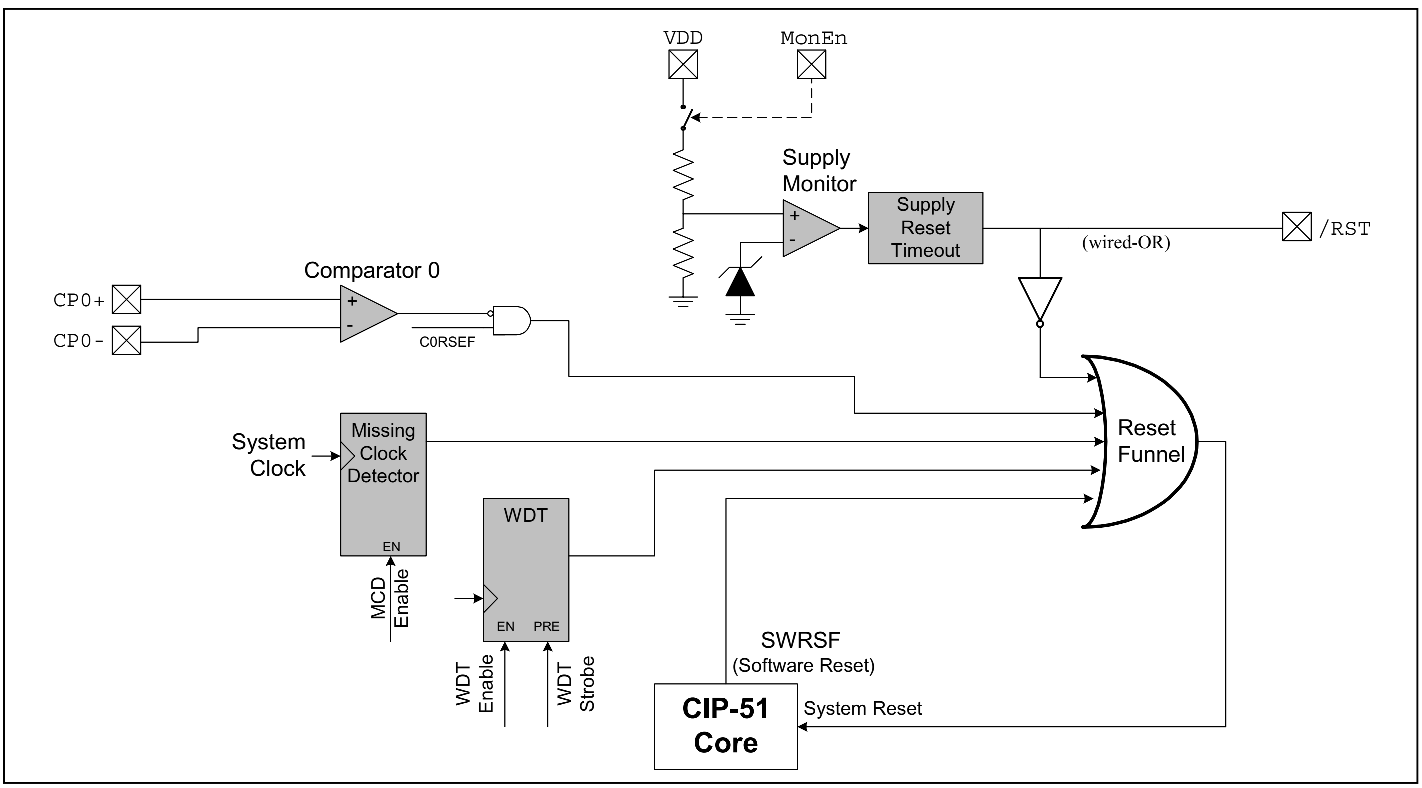
The C8051F206, C8051F220/1/6 and C8051F230/1/6 have several key enhancements both inside and outside the CIP-51 core to improve overall performance and ease of use in end applications.

The extended interrupt handler provides 22 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 six 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, and an external reset pin. The /RST pin is bi-directional, accommodating an external reset, or allowing the internally generated reset to be output on the /RST pin. The on-board VDD monitor is enabled by pulling the MONEN pin high (digital 1). The user may disable each reset source except for the VDD monitor and Reset Input Pin from software. The watchdog timer may be permanently enabled in software after a power-on reset during MCU initialization.

The MCU has an internal, stand-alone clock generator that is used by default as the system clock after 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 1.6. On-Board Clock and Reset

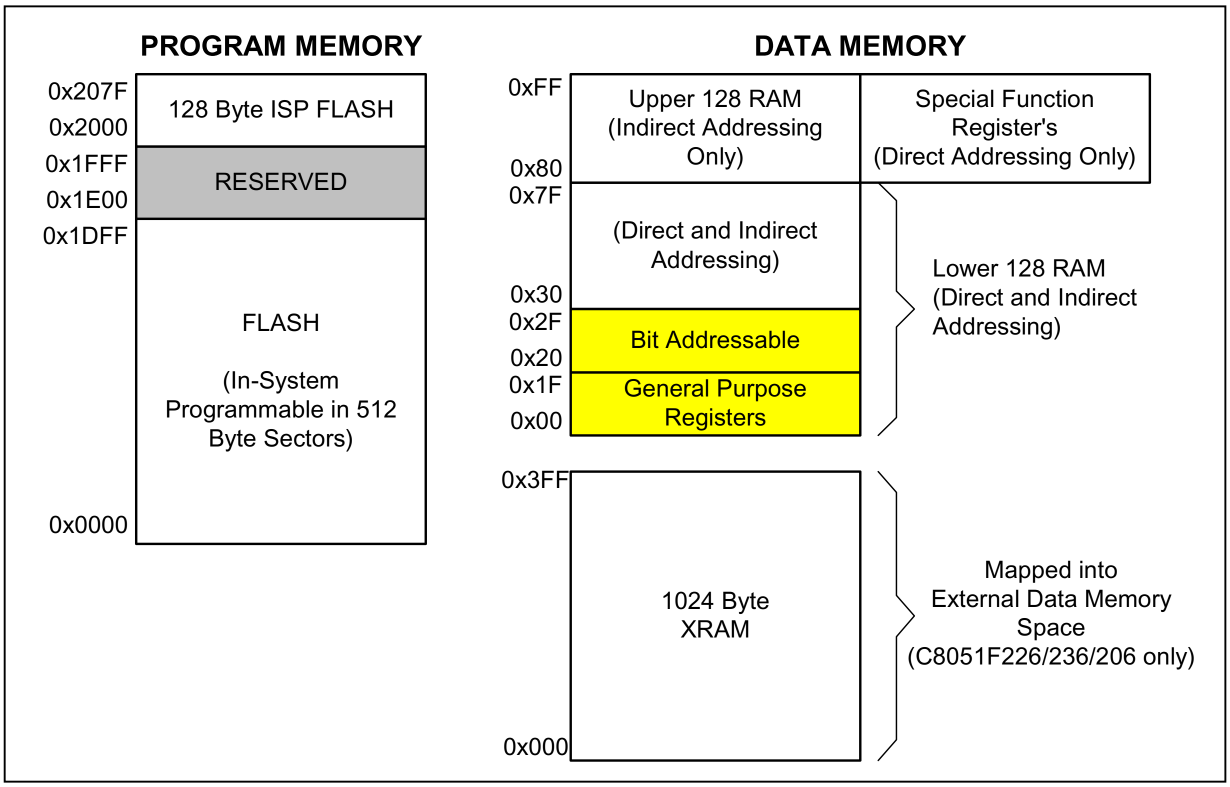


1.2. On-Board Memory

The CIP-51 has a standard 8051 program and data address configuration. It includes 256 bytes of data RAM, with the upper 128 bytes dual-mapped. An optional 1024 bytes of XRAM is available on the ‘F206, ‘F226 and ‘F236. Indirect addressing accesses the upper 128 bytes of general purpose RAM, and direct addressing accesses the 128-byte SFR address space. The lower 128 bytes of RAM are accessible via direct or indirect addressing. The first 32 bytes are addressable as four banks of general purpose registers, and the next 16 bytes can be byte addressable or bit addressable.

The MCU’s program memory consists of 8k + 128 bytes of FLASH. This memory may be reprogrammed in-system in 512 byte sectors, and requires no special off-chip programming voltage. The 512 bytes from addresses 0x1E00 to 0x1FFF are reserved for factory use. There is also a user programmable 128-byte sector at address 0x2000 to 0x207F, which may be useful as a table for storing software constants, nonvolatile configuration information, or as additional program space. See Figure 1.7 for the MCU system memory map.

Figure 1.7. On-Board Memory Map

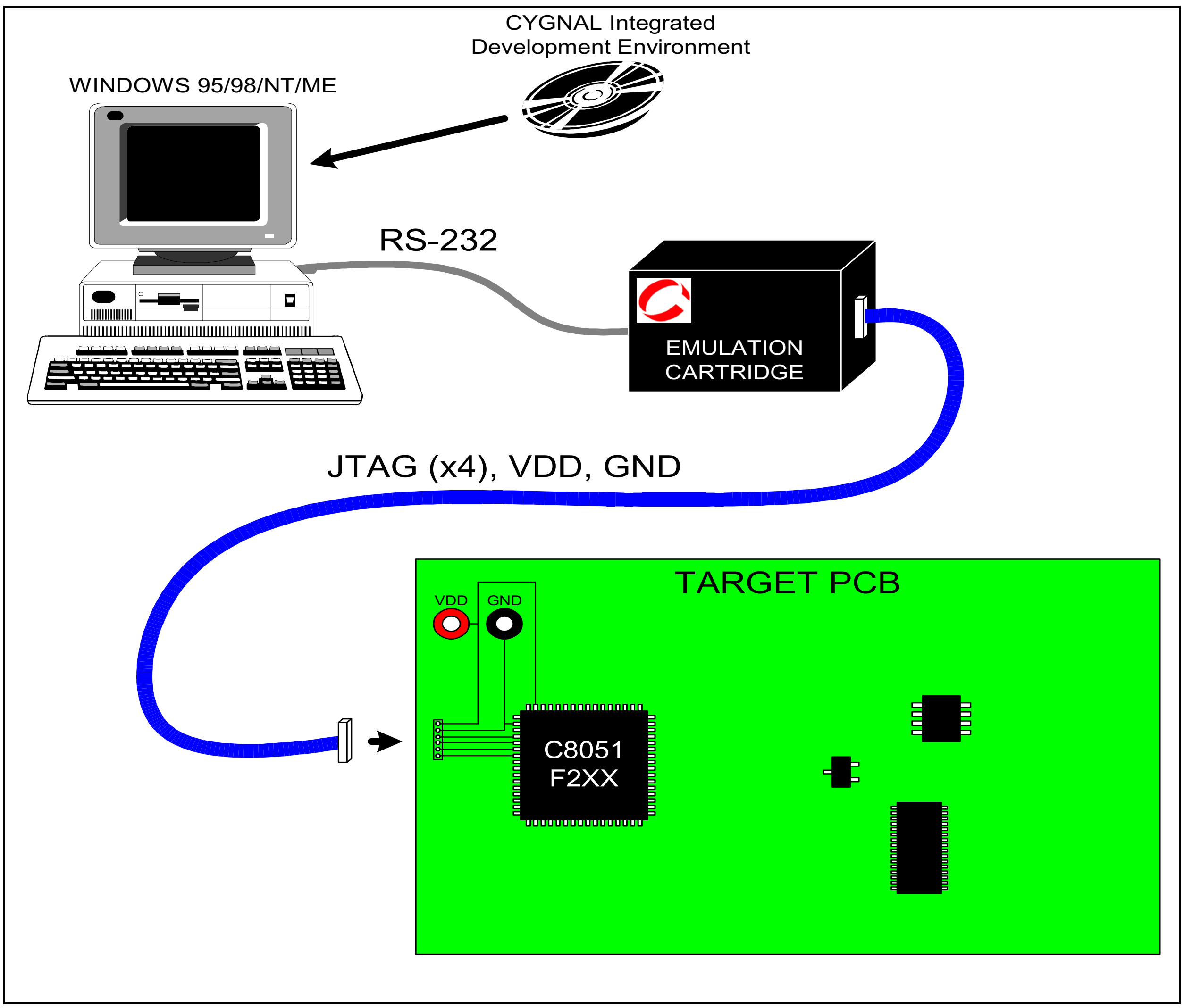


1.3. JTAG

The C8051F2xx have on-chip JTAG and debug logic 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. The C8051F2xxDK is a development kit with all the hardware and software necessary to develop application code and perform in-circuit debug with the C8051F2xx. The kit includes software with a developer's studio and debugger, an integrated 8051 assembler, and an RS-232 to JTAG interface module referred to as the EC. It also has a target application board with a C8051F2xx installed and large prototyping area, plus the RS-232 and JTAG cables, and wall-mount power supply. The Development Kit requires a Windows 9x, NT, or ME computer with one available RS-232 serial port. As shown in Figure 1.8, the PC is connected via RS-232 to the EC. A six-inch ribbon cable connects the EC to the user's application board, picking up the four JTAG pins and VDD and GND. The EC takes its power from the application board. It requires roughly 20mA at 2.7-3.6V. For applications where there is not sufficient power available from the target board, the provided power supply can be connected directly to the EC.

This is a vastly superior configuration for developing and debugging embedded applications compared to standard MCU Emulators, which use on-board "ICE Chips" and target cables and require the MCU in the application board to be socketed. Cygnal's debug environment both increases ease of use, and preserves the performance of the precision analog peripherals.

Figure 1.8. Debug Environment Diagram



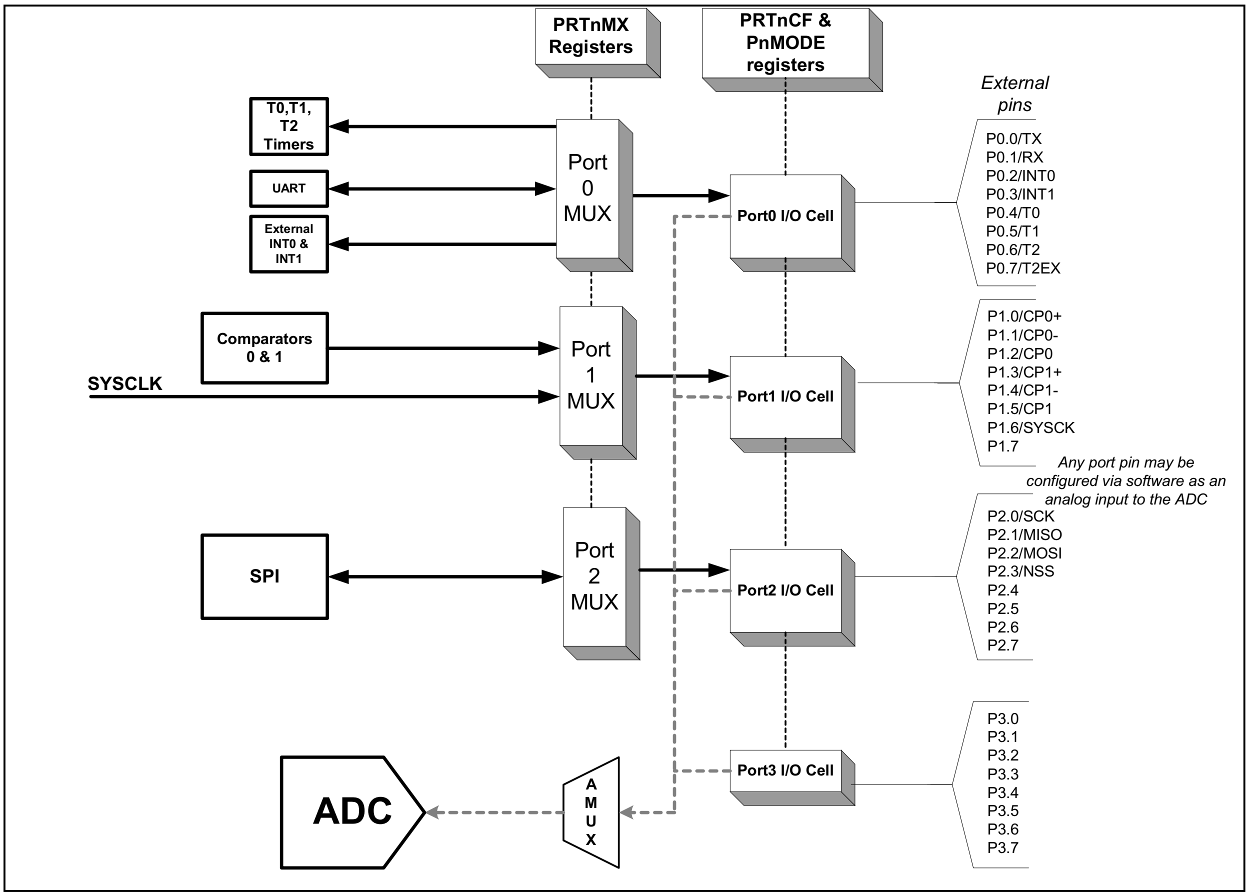
1.4. Digital/Analog Configurable I/O

The standard 8051 Ports (0, 1, 2, and 3) are available on the device. The ports behave like standard 8051 ports with a few enhancements.

Each port pin can be configured as either a push-pull or open-drain output. Any input that is configured as an analog input will have its corresponding weak pull-up turned off.

Digital resources (timers, SPI, UART, system clock, and comparators) are routed to corresponding I/O pins by configuring the port multiplexer. Port multiplexers are programmed by setting bits in SFR's (please see Section 14). Any of the 32 external port pins may be configured as either analog inputs or digital I/O (See Figure 1.9), so effectively, all port pins are dual function.

Figure 1.9. Port I/O Functional Block Diagram



1.5. Serial Ports

The C8051F206, C8051F220/1/6 and C8051F230/1/6 include a Full-Duplex UART and SPI Bus. Each of the serial buses is fully implemented in hardware and makes extensive use of the CIP-51's interrupts, thus requiring very little intervention by the CPU. The serial buses do not have to "share" resources such as timers, interrupts, or Port I/O, so both of the serial buses may be used simultaneously. (You may use Timer1, Timer 2, or SYSCLK to generate baud rates for UART).

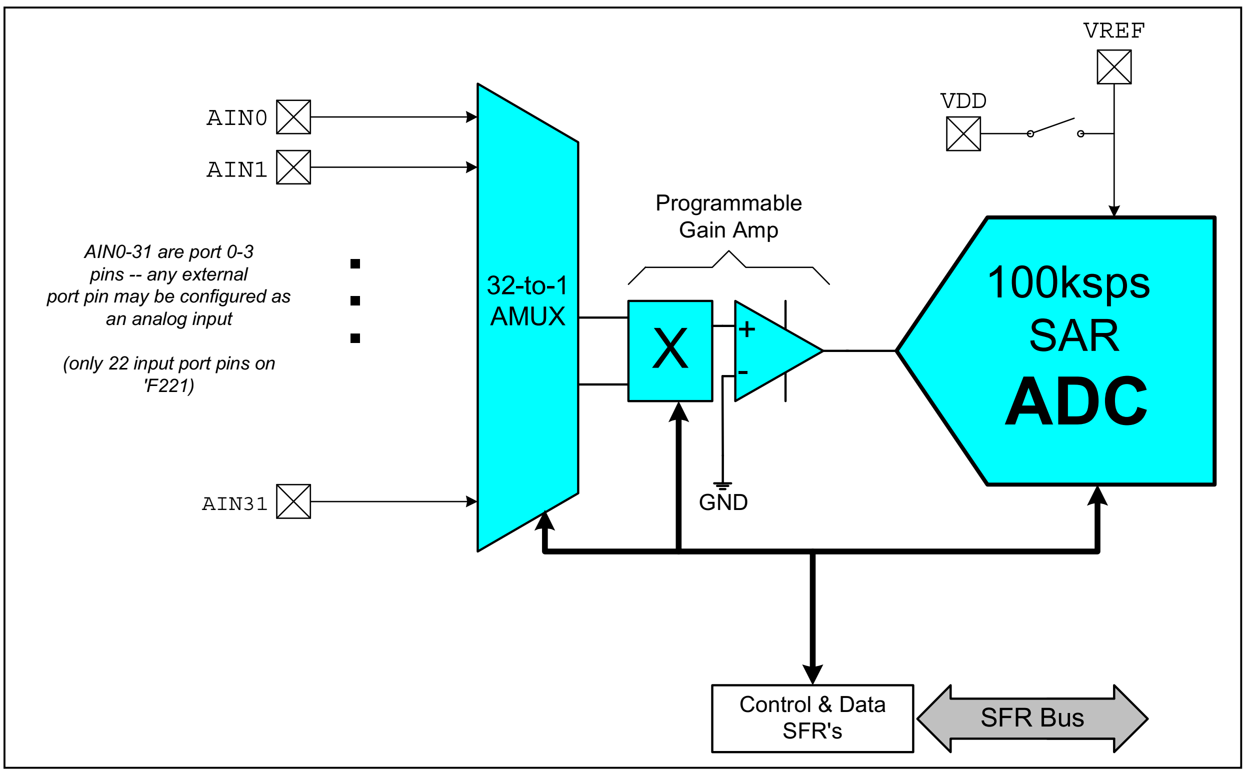
1.6. Analog to Digital Converter

The C8051F220/1/6 has an on-chip 8-bit SAR ADC and the C8051F206 has a 12-bit SAR ADC with a programmable gain amplifier. With a maximum throughput of 100ksps, the ADC offers true 8-bit with an INL of ±1/4 LSB, and or 12-bit accuracy with ±2 LSB. The voltage reference can be the power supply (VDD), or an external reference voltage (VREF). Also, the system controller can place the ADC into a power-saving shutdown mode when not in use. A programmable gain amplifier follows the analog multiplexer. The gain can be set in software from 0.5 to 16 in powers of 2.

Conversions can be initiated in two ways; a software command or an overflow on Timer 2. This flexibility allows the start of conversion to be triggered by software events, or convert continuously. A completed conversion causes an interrupt, or a status bit can be polled in software to determine the end of conversion. The resulting 8-bit data word is latched into an SFR upon completion of a conversion.

ADC data is continuously monitored by a programmable window detector, which interrupts the CPU when data is within the user-programmed window. This allows the ADC to monitor key system voltages in background mode, without the use of CPU resources.

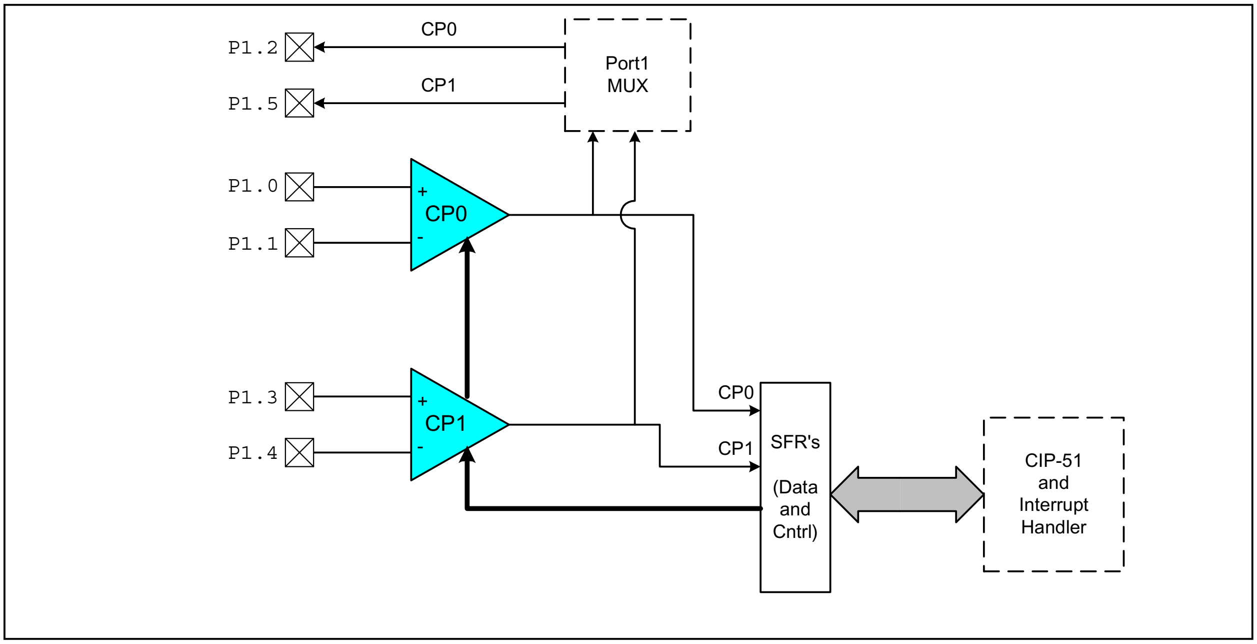
Figure 1.10. ADC Diagram



1.7. Comparators

The MCU's have two on-chip voltage comparators. The inputs of the comparators are available at package pins as illustrated in Figure 1.11. Each comparator's hysteresis is software programmable via special function registers (SFR's). Both voltage level and positive/negative going symmetry can be easily programmed by the user. Additionally, comparator interrupts can be implemented on either rising or falling-edge output transitions. Please see section 8 for details.

Figure 1.11. Comparator Diagram



2. ABSOLUTE MAXIMUM RATINGS*

Ambient temperature under bias .... -55 to 125°C
Storage Temperature .... -65 to 150°C
Voltage on any Pin (except VDD and Port I/O) with respect to DGND .... -0.3V to (VDD + 0.3V)
Voltage on any Port I/O Pin or /RST with respect to DGND .... -0.3V to 5.8V
Voltage on VDD with respect to DGND .... -0.3V to 4.2V
Total Power Dissipation 1.0W
Maximum output current sink by any Port pin 200mA
Maximum output current sink by any other I/O pin 25mA
Maximum output current sourced by any Port pin 200mA
Maximum output current sourced by any other I/O pin 25mA

*Note: Stresses above those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the devices at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

3. GLOBAL DC ELECTRICAL CHARACTERISTICS

-40°C to +85°C unless otherwise specified.

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS
Power supply voltage	(Note 1)	2.7		3.6	V
VDD supply current with ADC and comparators active, and CPU active	Clock=25MHz Clock=1MHz Clock=32kHz		9.5 3.6 125		mA µA
VDD supply current with ADC and comparators active, and CPU inactive (Idle Mode).	Clock=25MHz Clock=1MHz Clock=32kHz		5 1.8 125		mA µA
VDD supply current with ADC and comparators inactive, and CPU active	Clock=25MHz Clock=1MHz Clock=32kHz		9 1 20		mA µA
Digital Supply Current with CPU inactive (Idle Mode)	Clock=25MHz Clock=1MHz Clock=32kHz		4.5 0.1 10		mA µA
Digital Supply Current (Stop mode), VDD monitor enabled.	Oscillator not running		10		µA
Digital Supply Current (Stop Mode), VDD monitor disabled	Oscillator not running		0.1		µA
Digital Supply RAM Data Retention Voltage			1.5		V
Specified Operating Temperature Range		-40		+85	°C

Note 1: Power Supply must be greater than 1V and the MONEN pin must be pulled high for VDD monitor to operate.

4. PINOUT AND PACKAGE DEFINITIONS

Table 4.1 Pin Definitions

Name	‘F220, 226, 230, 236 48-Pin	‘F221, 231 32-Pin	Type	Description
VDD	11, 31	8		Digital Voltage Supply.
GND	5,6, 8, 13, 32	9		Ground. (**Note:** Pins 5,6, and 8 on the 48-pin package are not connected (NC), but it is recommended that they be connected to ground.)
MONEN	12		D In	Monitor Enable (on 48 pin package ONLY). Enables reset voltage monitor function when pulled high (logic “1”).
TCK	25	17	D In	JTAG Test Clock with internal pull-up.
TMS	26	18	D In	JTAG Test-Mode Select with internal pull-up.
TDI	28	20	D In	JTAG Test Data Input with internal pull-up. TDI is latched on a rising edge of TCK.
TDO	27	19	D Out	JTAG Test Data Output. Data is shifted out on TDO on the falling edge of TCK. TDO output is a tri-state driver.
XTAL1	9	6	A In	Crystal Input. This pin is the return for the internal oscillator circuit for a crystal or ceramic resonator. For a precision internal clock, connect a crystal or ceramic resonator from XTAL1 to XTAL2. If overdriven by an external CMOS clock, this becomes the system clock.
XTAL2	10	7	A Out	Crystal Output. This pin is the excitation driver for a crystal or ceramic resonator.
/RST	14	10	D I/O	Chip Reset. Open-drain output of internal Voltage Supply monitor. Is driven low when VDD is < 2.7V and MONEN=1, or when a ‘1’is written to PORSF. An external source can force a system reset by driving this pin low.
VREF	7	5	A I/O	Voltage Reference. When configured as an input, this pin is the voltage reference for the ADC. Otherwise, VDD will be the reference. NOTE: this pin is Not Connected (NC) on ‘F230/1/6.
CP0+	4	4	A In	Comparator 0 Non-Inverting Input.
CP0-	3	3	A In	Comparator 0 Inverting Input.
CP0	2	2	D Out	Comparator 0 Output
CP1+	1	1	A In	Comparator 1 Non-Inverting Input.
CP1-	48	32	A In	Comparator 1 Inverting Input.
CP1	47	31	D Out	Comparator 1 Output
P0.0/ TX	40	28	D I/O A In	Port0 Bit0. (See the Port I/O Sub-System section for complete description).
P0.1/ RX	39	27	D I/O A In	Port0 Bit1. (See the Port I/O Sub-System section for complete description).
P0.2/ INT0	38	26	D I/O A In	Port0 Bit2. (See the Port I/O Sub-System section for complete description).
P0.3/ INT1	37	25	D I/O A In	Port0 Bit3. (See the Port I/O Sub-System section for complete description).
P0.4/ T0	36	24	D I/O A In	Port0 Bit4. (See the Port I/O Sub-System section for complete description).
P0.5/ T1	35	23	D I/O A In	Port0 Bit5. (See the Port I/O Sub-System section for complete description).
P0.6/ T2	34	22	D I/O A In	Port0 Bit6. (See the Port I/O Sub-System section for complete description).
P0.7/ T2EX	33	21	D I/O A In	Port0 Bit7. (See the Port I/O Sub-System section for complete description).
P1.0/ CP0+	4	4	D I/O A In	Port1 Bit0. (See the Port I/O Sub-System section for complete description).

Name	'F220, 226, 230, 236 48-Pin	'F221, 231 32-Pin	Type	Description
P1.1/ CP0-	3	3	D I/O A In	Port1 Bit1. (See the Port I/O Sub-System section for complete description).
P1.2/ CP0	2	2	D I/O A In	Port1 Bit2. (See the Port I/O Sub-System section for complete description).
P1.3/ CP1+	1	1	D I/O A In	Port1 Bit3. (See the Port I/O Sub-System section for complete description).
P1.4/ CP1-	48	32	D I/O A In	Port1 Bit4. (See the Port I/O Sub-System section for complete description).
P1.5/ CP1	47	31	D I/O A In	Port1 Bit5. (See the Port I/O Sub-System section for complete description).
P1.6/ SYSCLK	46	30	D I/O A In	Port1 Bit6. (See the Port I/O Sub-System section for complete description).
P1.7	45	29	D I/O A In	Port1 Bit7. (See the Port I/O Sub-System section for complete description).
P2.0/ SCK	24	16	D I/O A In	Port2 Bit0. (See the Port I/O Sub-System section for complete description).
P2.1/ MISO	23	15	D I/O A In	Port2 Bit1. (See the Port I/O Sub-System section for complete description).
P2.2/ MOSI	22	14	D I/O A In	Port2 Bit2. (See the Port I/O Sub-System section for complete description).
P2.3/ NSS	21	13	D I/O A In	Port2 Bit3. (See the Port I/O Sub-System section for complete description).
P2.4	15	11	D I/O A In	Port2 Bit4. (See the Port I/O Sub-System section for complete description).
P2.5	16	12	D I/O A In	Port2 Bit5. (See the Port I/O Sub-System section for complete description).
P2.6	17		D I/O A In	Port2 Bit6. (See the Port I/O Sub-System section for complete description).
P2.7	18		D I/O A In	Port2 Bit7. (See the Port I/O Sub-System section for complete description).
P3.0	44		D I/O A In	Port3 Bit0. (See the Port I/O Sub-System section for complete description).
P3.1	43		D I/O A In	Port3 Bit1. (See the Port I/O Sub-System section for complete description).
P3.2	42		D I/O A In	Port3 Bit2. (See the Port I/O Sub-System section for complete description).
P3.3	41		D I/O A In	Port3 Bit3. (See the Port I/O Sub-System section for complete description).
P3.4	30		D I/O A In	Port3 Bit4. (See the Port I/O Sub-System section for complete description).
P3.5	29		D I/O A In	Port3 Bit5. (See the Port I/O Sub-System section for complete description).
P3.6	20		D I/O A In	Port3 Bit6. (See the Port I/O Sub-System section for complete description).
P3.7	19		D I/O A In	Port3 Bit7. (See the Port I/O Sub-System section for complete description).

Figure 4.1 TQFP-48 Pin Diagram

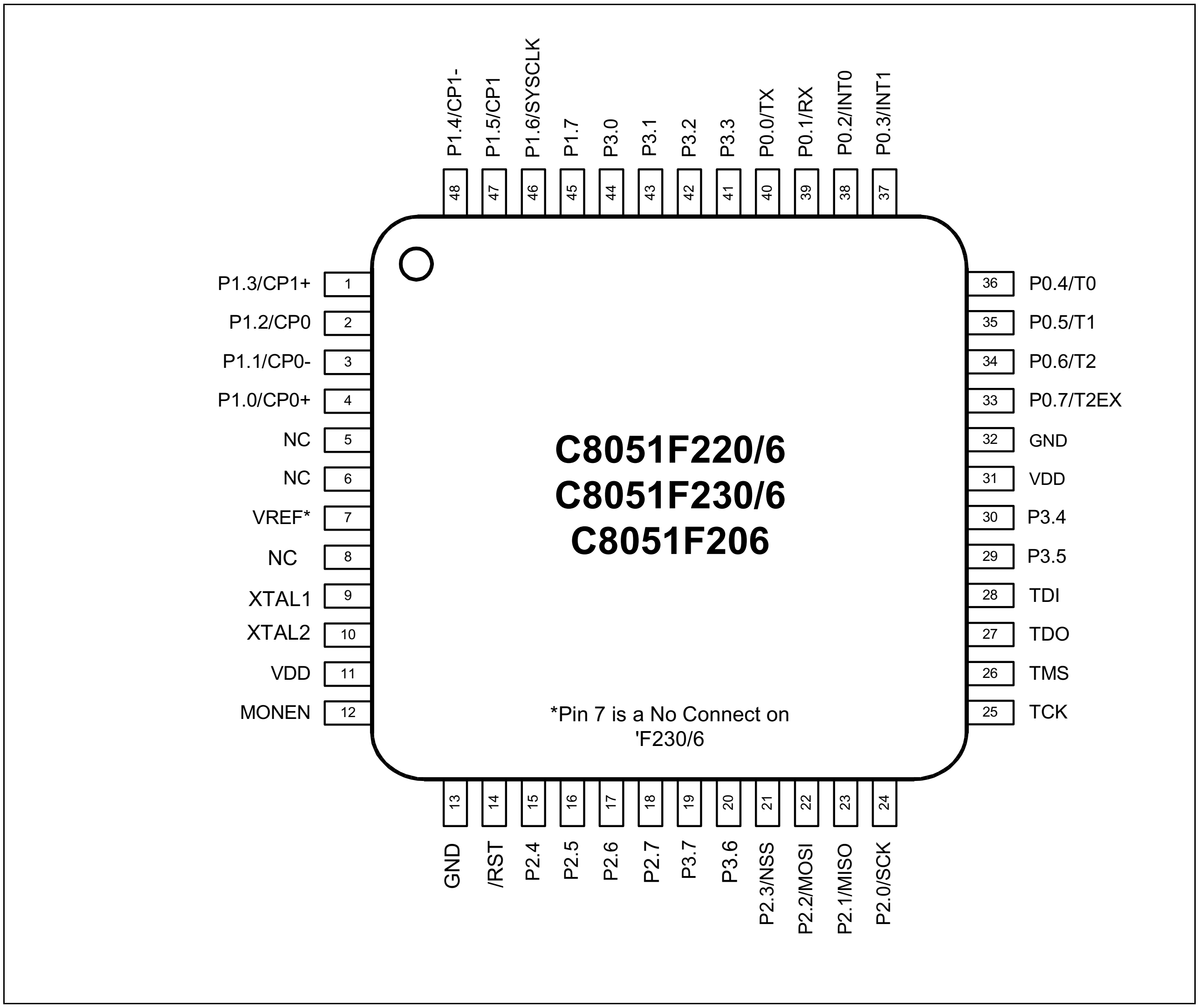


Figure 4.2 LQFP-32 Pin Diagram

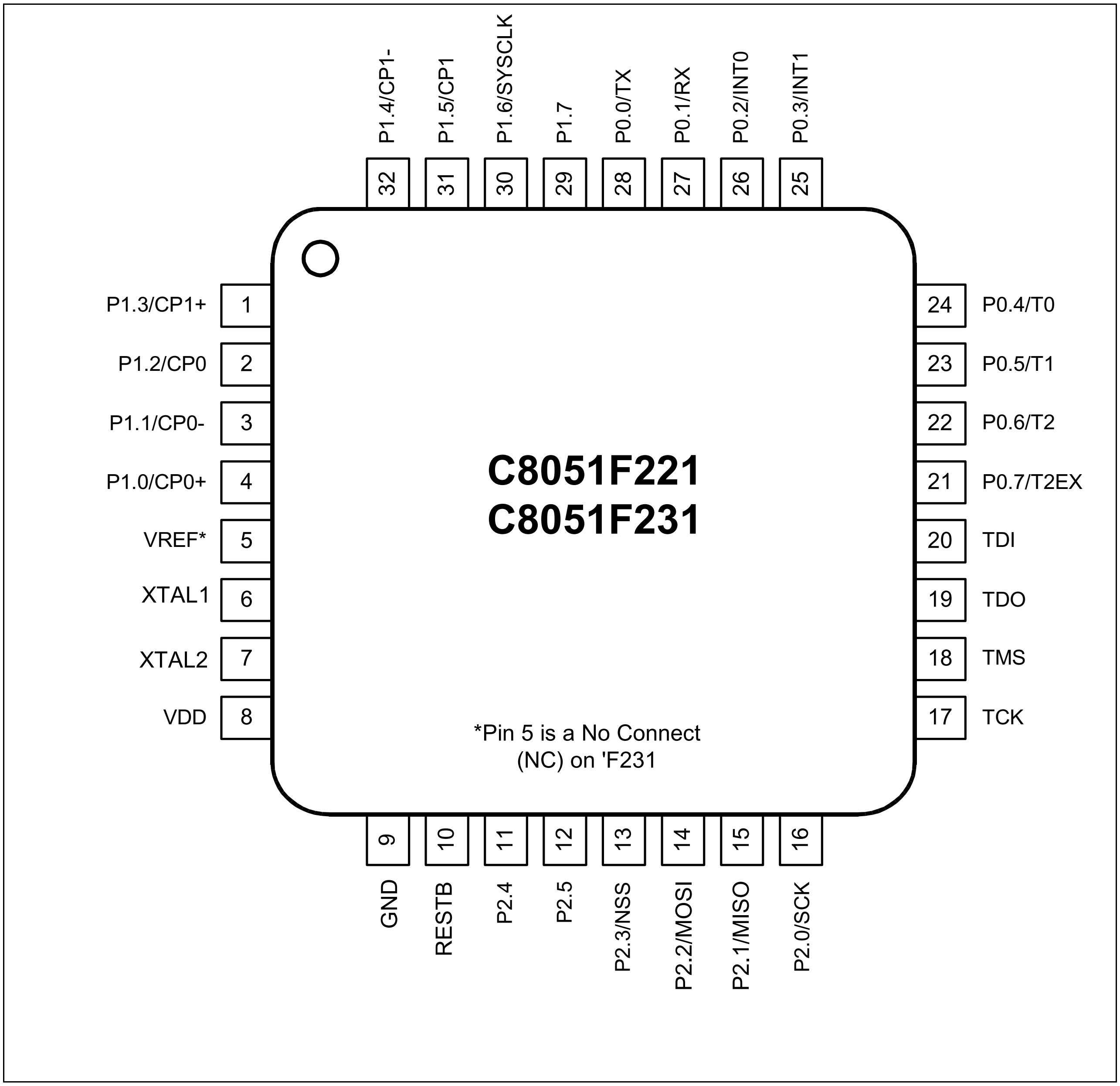
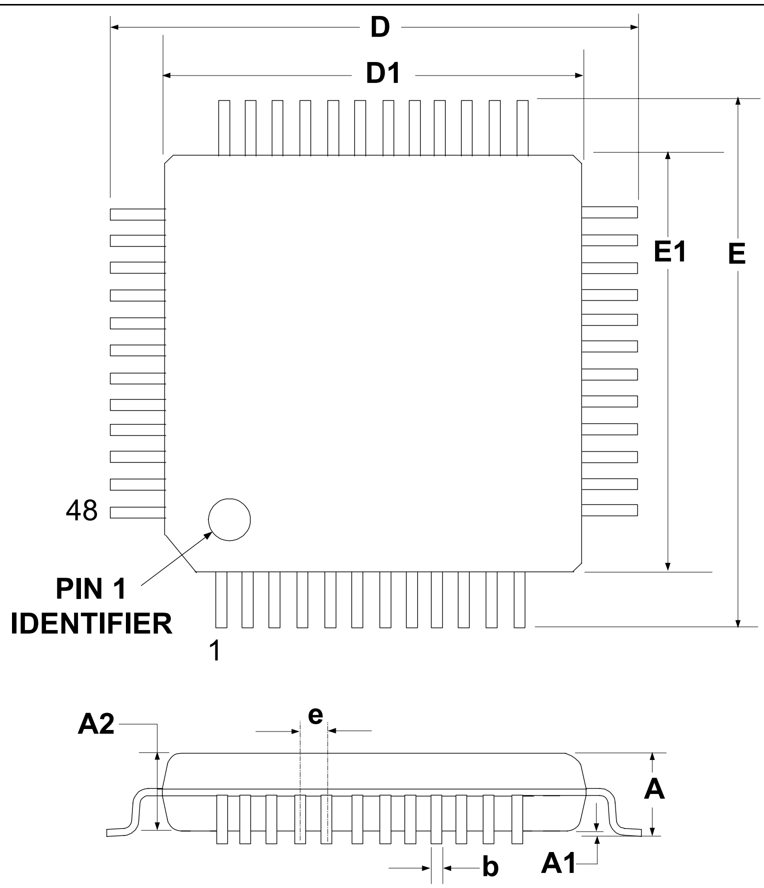
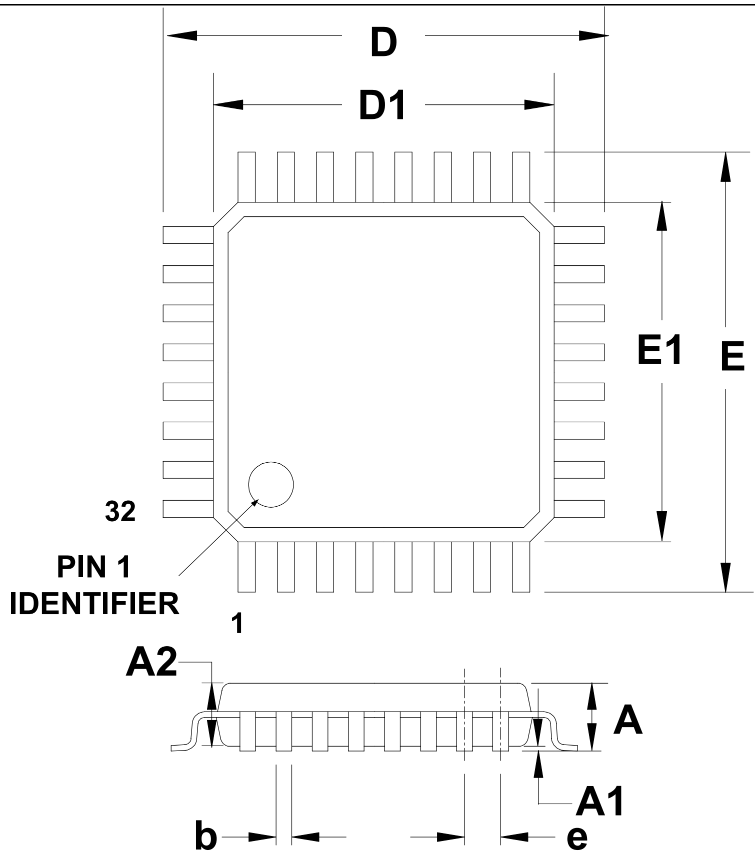


Figure 4.3 TQFP-48 Package Drawing



	MIN (mm)	NOM (mm)	MAX (mm)
A	-	-	1.20
A1	0.05	-	0.15
A2	0.95	1.00	1.05
b	0.17	0.22	0.27
D	-	9.00	-
D1	-	7.00	-
e	-	0.50	-
E	-	9.00	-
E1	-	7.00	-

Figure 4.4 LQFP-32 Package Drawing



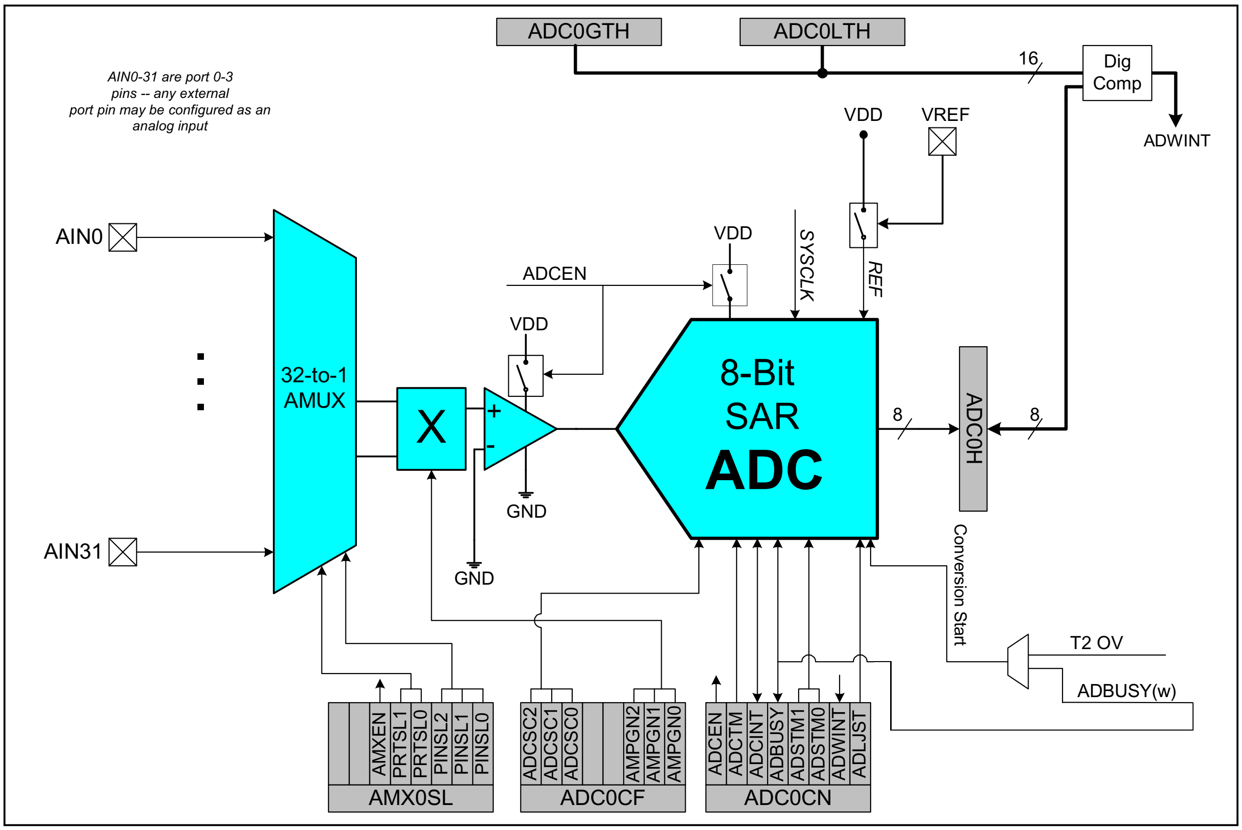
	MIN (mm)	NOM (mm)	MAX (mm)
A	-	-	1.60
A1	0.05	-	0.15
A2	1.35	1.40	1.45
b	0.30	0.37	0.45
D	-	9.00	-
D1	-	7.00	-
e	-	0.80	-
E	-	9.00	-
E1	-	7.00	-

5. ADC (8-Bit, C8051F220/1/6 Only)

Description

The ADC subsystem for the C8051F220/1/6 consists of configurable analog multiplexer (AMUX), a programmable gain amplifier (PGA), and a 100ksps, 8-bit successive-approximation-register ADC with integrated track-and-hold and programmable window detector (see 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 5.5) is set to 1. The ADC subsystem is in low power shutdown when this bit is 0.

Figure 5.1. 8-Bit ADC Functional Block Diagram



5.1. Analog Multiplexer and PGA

Any external port pin (ports 0-3) may be selected via software. The AMX0SL SFR is used to select the desired analog input pin. (See Figure 5.3). When the AMUX is enabled, the user selects which port is to be used (bits PRTSL0-1), and then the pin in the selected port (bits PINSL0-2) to be the analog input.
The table in shows AMUX functionality by channel for each possible configuration. The PGA amplifies the AMUX output signal by an amount determined by the states of the AMPGN2-0 bits in the ADC Configuration register, ADC0CF (Figure 5.4). The PGA can be software-programmed for gains of 0.5, 1, 2, 4, 8 or 16. It defaults to a gain of 1 on reset.

5.2. ADC Modes of Operation

The ADC has a maximum conversion speed of 100ksps. The ADC conversion clock is derived from the system clock. The ADC conversion clock is derived from a divided version of SYSCLK. Divide ratios of 1,2,4,8, or 16 are supported by setting the ADCSC bits in the ADC0CF Register. This is useful to adjust conversion speed to accommodate different system clock speeds.

A conversion can be initiated in one of two ways, depending on the programmed states of the ADC Start of Conversion Mode bits (ADSTM1, ADSTM0) in ADC0CN. Conversions may be initiated by:

1. Writing a 1 to the ADBUSY bit of ADC0CN;
2. A Timer 2 overflow (i.e. timed continuous conversions).

Writing a 1 to ADBUSY provides software control of the ADC whereby conversions are performed "on-demand". During conversion, the ADBUSY bit is set to 1 and restored to 0 when conversion is complete. The falling edge of ADBUSY triggers an interrupt (when enabled) and sets the ADCINT interrupt flag in the ADC0CN register. Converted data is available in the ADC data word register, ADC0H.

The ADCTM bit in register ADC0CN controls the ADC track-and-hold mode. In its default state, the ADC input is continuously tracked, except when a conversion is in progress. Setting ADCTM to 1 allows one of two different low power track-and-hold modes to be specified by states of the ADSTM1-0 bits (also in ADC0CN):

1. Tracking begins with a write of 1 to ADBUSY and lasts for 3 SAR clocks;
2. Tracking starts with an overflow of Timer 2 and lasts for 3 SAR clocks.

Tracking can be disabled (shutdown) when the entire chip is in low power standby or sleep modes.

Figure 5.2. 12-Bit ADC Track and Conversion Example Timing

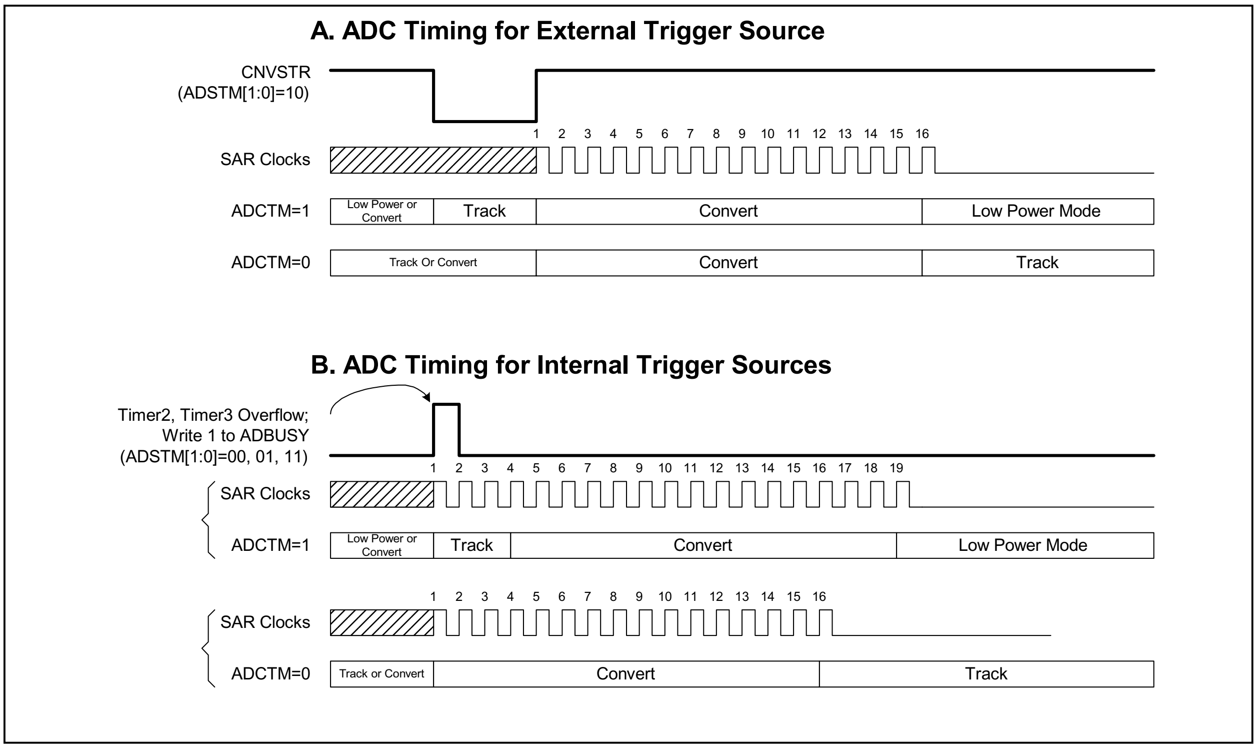


Figure 5.3. AMX0SL: AMUX Channel Select Register (C8051F220/1/6 and C8051F206)

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
-	-	AMXEN	PRTSL1	PRTSL0	PINSL2	PINSL1	PINSL0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xBB

Bits 7-6: UNUSED. Read = 00b; Write = don't care

Bit 5: AMXEN enable
- 0: AMXEN disabled and port pins are unavailable for analog use.
- 1: AMXEN enabled to use/select port pins for analog use.

Bits 4-3: PRTSL1-0: Port Select Bits*.
- 00: Port0 select to configure pin for analog input from this port.
- 01: Port1 select to configure pin for analog input from this port.
- 10: Port2 select to configure pin for analog input from this port.
- 11: Port3 select to configure pin for analog input from this port.

Bits 2-0: PINSL2-0: Pin Select Bits
- 000: Pin 0 of selected port (above) to be used for analog input.
- 001: Pin 1 of selected port (above) to be used for analog input.
- 010: Pin 2 of selected port (above) to be used for analog input.
- 011: Pin 3 of selected port (above) to be used for analog input.
- 100: Pin 4 of selected port (above) to be used for analog input.
- 101: Pin 5 of selected port (above) to be used for analog input.
- 110: Pin 6 of selected port (above) to be used for analog input.
- 111: Pin 7 of selected port (above) to be used for analog input.

*** Selecting a port for analog input does NOT default all pins of that port as analog input. After selecting a port for analog input, a pin must be selected using pin select bits (PINSL2-0). For example, after setting the AMXEN to '1', setting PRTSL1-0 to "11", and setting PINSL2-0 to "100" P3.4 is configured as analog input. All other Port 3 pins remain as GPIO pins. Also note that in order to use a port pin as analog input, its input mode should be set to *analog*. Please see section 14.2.**

Figure 5.4. ADC0CF: ADC Configuration Register (C8051F220/1/6 and C8051F206)

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: SAR conversion clock should be less than or equal to 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

Figure 5.5. ADC0CN: ADC Control Register (C8051F220/1/6 and C8051F206)

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
ADCEN	ADCTM	ADCINT	ADBUSY	ADSTM1	ADSTM0	ADWINT	ADLJST	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
							(bit addressable)	0xE8

Bit7: ADCEN: ADC Enable Bit
0: ADC Disabled. ADC is in low power shutdown.
1: ADC Enabled. ADC is active and ready for data conversions.

Bit6: ADCTM: ADC Track Mode Bit
0: When the ADC is enabled, tracking is continuous unless a conversion is in process
1: Tracking Defined by ADSTM1-0 bits
ADSTM1-0:
00: Tracking starts with the write of 1 to ADBUSY and lasts for 3 SAR clocks
01: RESERVED
10: RESERVED
11: Tracking started by the overflow of Timer 2 and last for 3 SAR clocks

Bit5: ADCINT: ADC Conversion Complete Interrupt Flag (cleared by software).
0: ADC has not completed a data conversion since the last time this flag was cleared
1: ADC has completed a data conversion

Bit4: ADBUSY: ADC Busy Bit
Read
0: ADC Conversion complete or no valid data has been converted since a reset. The falling edge of ADBUSY generates an interrupt when enabled.
1: ADC Busy converting data
Write
0: No effect
1: Starts ADC Conversion if ADSTM1-0 = 00b

Bits3-2: ADSTM1-0: ADC Start of Conversion Mode Bits
00: ADC conversion started upon a write of 1 to ADBUSY
01: RESERVED
10: RESERVED
11: ADC conversions initiated on overflows of Timer 2

Bit1: ADWINT: ADC Window Compare Interrupt Flag
0: ADC Window Comparison Data match has not occurred
1: ADC Window Comparison Data match occurred

Bit0: ADLJST: ADC Left Justify Data Bit (Used on C8051F206 only)
0: Data in ADC0H:ADC0L registers are right justified.
1: Data in ADC0H:ADC0L registers are left justified.

Figure 5.6. ADC0H: ADC Data Word Register (C8051F220/1/6 and C8051F206)

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

Bits7-0: ADC Data Word Bits

EXAMPLE: ADC Data Word Conversion Map

AIN – GND(Volts)	**ADC0H**
REF x (255/256)	0xFF
REF x ½	0x80
REF x (127/256)	0x7F
0	0x00

5.3. ADC Programmable Window Detector

The ADC programmable window detector is very useful in many applications. It continuously compares the ADC output to user-programmed limits and notifies the system when an out-of-band condition is detected. This is especially effective in an interrupt-driven system, saving code space and CPU bandwidth while delivering faster system response times. The window detector interrupt flag (ADWINT in ADC0CN) can also be used in polled mode. The high and low bytes of the reference words are loaded into the ADC Greater-Than and ADC Less-Than registers (ADC0GTH and ADC0LTH).

Figure 5.7. ADC0GTH: ADC Greater-Than Data Register (C8051F220/1/6 and C8051F206)

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								11111111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xC5

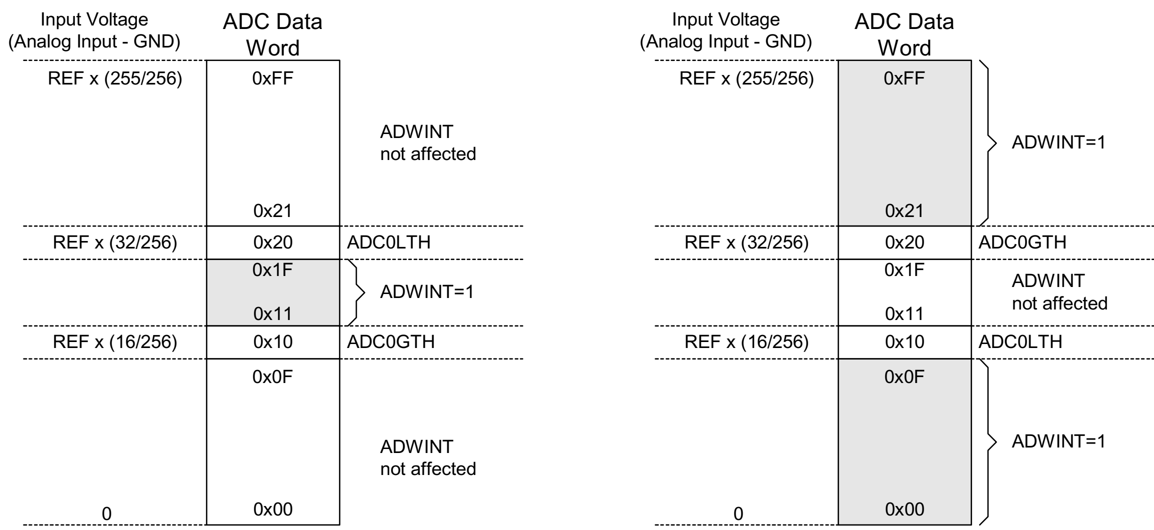
Bits7-0:
The high byte of the ADC Greater-Than Data Word.

Figure 5.8. ADC0LTH: ADC Less-Than Data Byte Register (C8051F220/1/6 and C8051F206)

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: 0xC7

Bits7-0:
The high byte of the ADC Less-Than Data Word.

Figure 5.9. 8-Bit ADC Window Interrupt Examples



Given:
AMX0SL = 0x00, AMX0CF = 0x00, ADLJST = 0,
ADC0LTH = 0x20, ADC0GTH = 0x10.

An ADC End of Conversion will cause an ADC Window Compare Interrupt (ADWINT=1) if the resulting ADC Data Word is < 0x20 and > 0x10.

Given:
AMX0SL = 0x00, AMX0CF = 0x00, ADLJST = 0,
ADC0LTH = 0x10, ADC0GTH = 0x20.

An ADC End of Conversion will cause an ADC Window Compare Interrupt (ADWINT=1) if the resulting ADC Data Word is < 0x10 or > 0x20.

Table 5.1. 8-Bit ADC Electrical Characteristics

VDD = 3.0V, VREF = 2.40V, PGA Gain = 1, -40°C to +85°C unless otherwise specified.

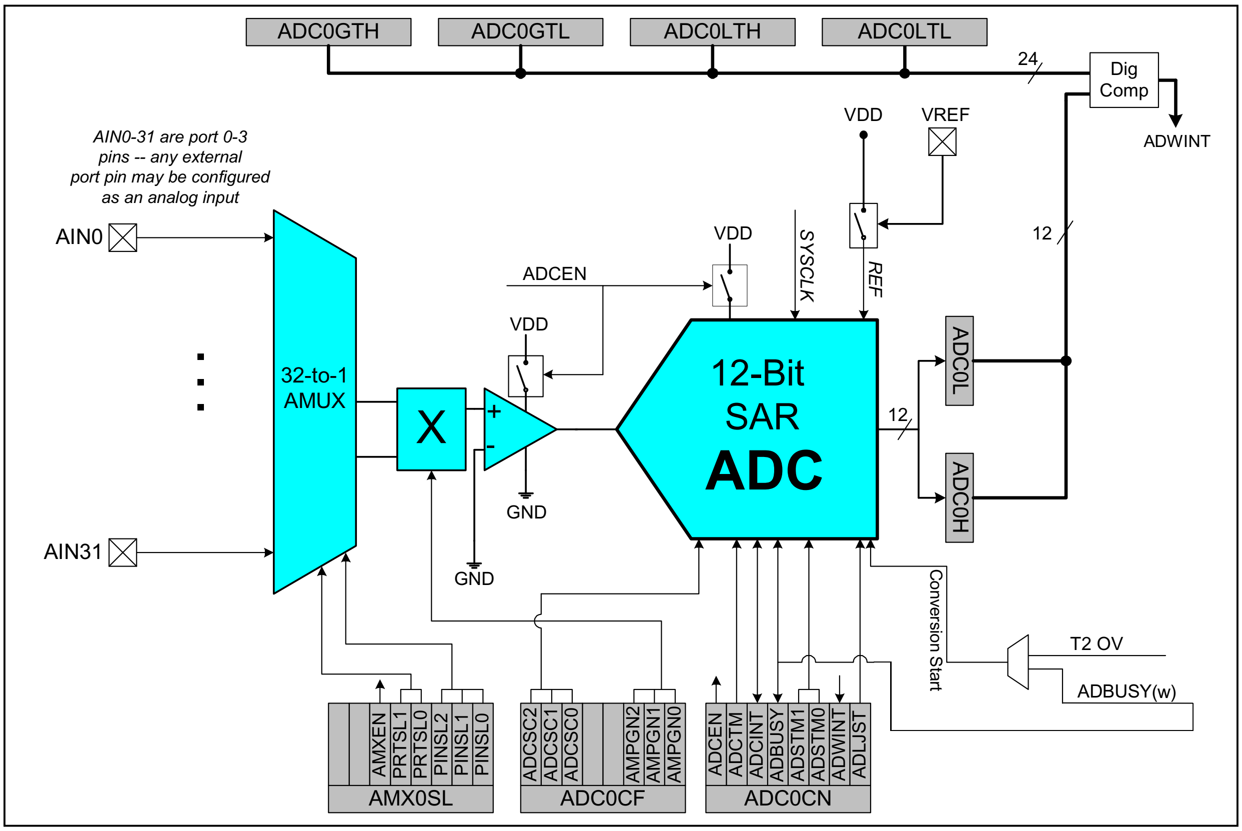
PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS
DC ACCURACY					
Resolution			8		bits
Integral Nonlinearity				±1/4	LSB
Differential Nonlinearity	Guaranteed Monotonic			±1/2	LSB
Offset Error				±1/2	LSB
Gain Error				±1/2	LSB
Total Unadjusted Error				±1/2	LSB
DYNAMIC PERFORMANCE (10kHz sine-wave input, 0 to –1dB of full scale, 100ksps)					
Signal-to-Noise Plus Distortion			49.5		dB
Total Harmonic Distortion	Up to the 5^{th} harmonic		-65		dB
Spurious-Free Dynamic Range			-65		dB
CONVERSION RATE					
Throughput Rate				100	ksps
ANALOG INPUTS					
Input Voltage Range		0		VDD	V
Input Capacitance			10		pF
POWER SPECIFICATIONS					
Power Supply Current	Operating Mode, 100ksps			1.0	mA
Power Supply Current in Shutdown			0.1	1	µA
Power Supply Rejection			±0.3		mV/V

6. ADC (12-Bit, C8051F206 Only)

Description

The ADC subsystem for the C8051F206 consists of 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 Figure 6.1). The AMUX, PGA, Data Conversion Modes, and Window Detector are all configurable under software control via the Special Function Register's shown in Figure 6.1. The ADC subsystem (ADC, track-and-hold and PGA) is enabled only when the ADCEN bit in the ADC Control register (ADC0CN, Figure 6.5) is set to 1. The ADC subsystem is in low power shutdown when this bit is 0.

Figure 6.1. 12-Bit ADC Functional Block Diagram



6.1. Analog Multiplexer and PGA

Any external port pin (ports 0-3) may be selected via software. The AMX0SL SFR is used to select the desired analog input pin. (See Figure 6.3). When the AMUX is enabled, the user selects which port is to be used (bits PRTSL0-1), and then the pin in the selected port (bits PINSL0-2) to be the analog input.

The table in shows AMUX functionality by channel for each possible configuration. The PGA amplifies the AMUX output signal by an amount determined by the states of the AMPGN2-0 bits in the ADC Configuration register, ADC0CF (Figure 6.4). The PGA can be software-programmed for gains of 0.5, 1, 2, 4, 8 or 16. It defaults to a gain of 1 on reset.

6.2. ADC Modes of Operation

The ADC has a maximum conversion speed of 100ksps. The ADC conversion clock is derived from the system clock. The ADC conversion clock is derived from a divided version of SYSCLK. Divide ratios of 1,2,4,8, or 16 are supported by setting the ADCSC bits in the ADC0CF Register. This is useful to adjust conversion speed to accommodate different system clock speeds.

A conversion can be initiated in one of two ways, depending on the programmed states of the ADC Start of Conversion Mode bits (ADSTM1, ADSTM0) in ADC0CN. Conversions may be initiated by:

1. Writing a 1 to the ADBUSY bit of ADC0CN;
2. A Timer 2 overflow (i.e. timed continuous conversions).

Writing a 1 to ADBUSY provides software control of the ADC whereby conversions are performed "on-demand". During conversion, the ADBUSY bit is set to 1 and restored to 0 when conversion is complete. The falling edge of ADBUSY triggers an interrupt (when enabled) and sets the ADCINT interrupt flag in the ADC0CN register. Converted data is available in the ADC data word register, ADC0H.

The ADCTM bit in register ADC0CN controls the ADC track-and-hold mode. In its default state, the ADC input is continuously tracked, except when a conversion is in progress. Setting ADCTM to 1 allows one of two different low power track-and-hold modes to be specified by states of the ADSTM1-0 bits (also in ADC0CN):

1. Tracking begins with a write of 1 to ADBUSY and lasts for 3 SAR clocks;
2. Tracking starts with an overflow of Timer 2 and lasts for 3 SAR clocks.

Tracking can be disabled (shutdown) when the entire chip is in low power standby or sleep modes.

Figure 6.2. 12-Bit ADC Track and Conversion Example Timing

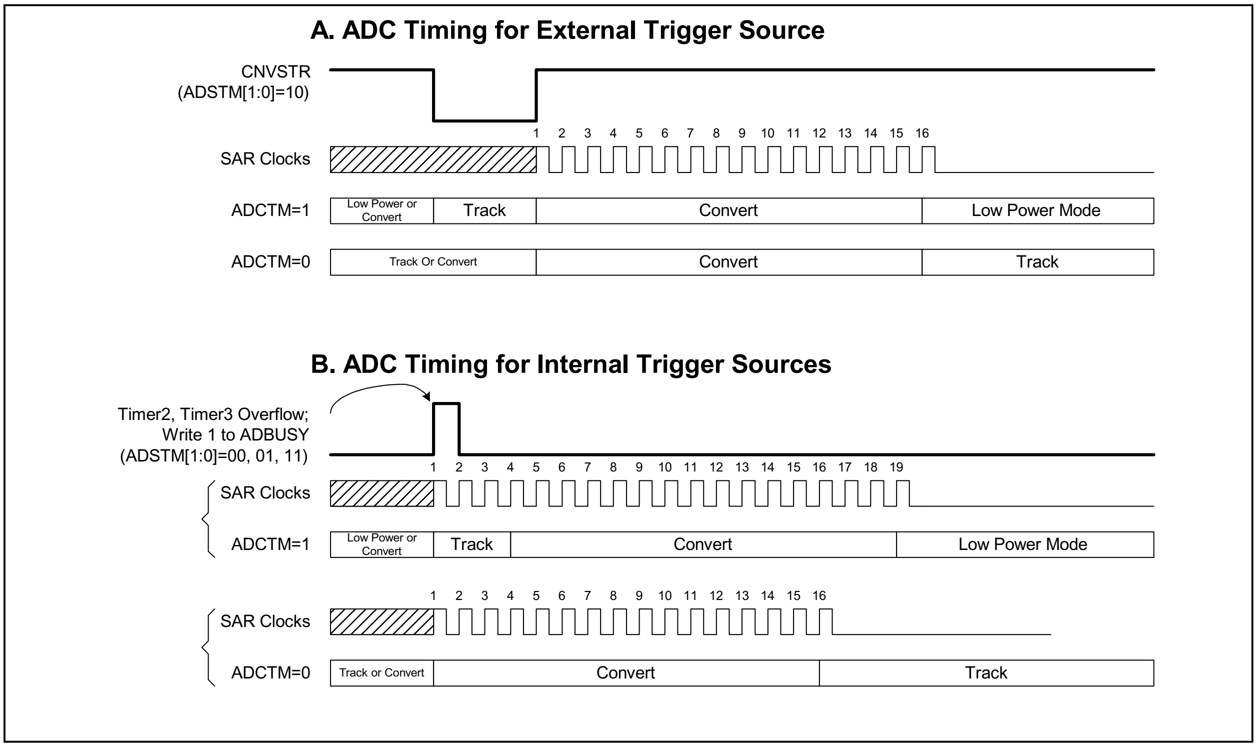


Figure 6.3. AMX0SL: AMUX Channel Select Register (C8051F220/1/6 and C8051F206)

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
-	-	AMXEN	PRTSL1	PRTSL0	PINSL2	PINSL1	PINSL0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xBB

Bits 7-6: UNUSED. Read = 00b; Write = don't care

Bit 5: AMXEN enable
- 0: AMXEN disabled and port pins are unavailable for analog use.
- 1: AMXEN enabled to use/select port pins for analog use.

Bits 4-3: PRTSL1-0: Port Select Bits*.
- 00: Port0 select to configure pin for analog input from this port.
- 01: Port1 select to configure pin for analog input from this port.
- 10: Port2 select to configure pin for analog input from this port.
- 11: Port3 select to configure pin for analog input from this port.

Bits 2-0: PINSL2-0: Pin Select Bits
- 000: Pin 0 of selected port (above) to be used for analog input.
- 001: Pin 1 of selected port (above) to be used for analog input.
- 010: Pin 2 of selected port (above) to be used for analog input.
- 011: Pin 3 of selected port (above) to be used for analog input.
- 100: Pin 4 of selected port (above) to be used for analog input.
- 101: Pin 5 of selected port (above) to be used for analog input.
- 110: Pin 6 of selected port (above) to be used for analog input.
- 111: Pin 7 of selected port (above) to be used for analog input.

*** Selecting a port for analog input does NOT default all pins of that port as analog input. After selecting a port for analog input, a pin must be selected using pin select bits (PINSL2-0). For example, after setting the AMXEN to '1', setting PRTSL1-0 to "11", and setting PINSL2-0 to "100" P3.4 is configured as analog input. All other Port 3 pins remain as GPIO pins. Also note that in order to use a port pin as analog input, its input mode should be set to *analog*. Please see section 14.2.**

Figure 6.4. ADC0CF: ADC Configuration Register (C8051F220/1/6 and C8051F206)

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: SAR conversion clock should be less than or equal to 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

Figure 6.5. ADC0CN: ADC Control Register (C8051F220/1/6 and C8051F206)

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
ADCEN	ADCTM	ADCINT	ADBUSY	ADSTM1	ADSTM0	ADWINT	ADLJST	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0 (bit addressable)	SFR Address: 0xE8

Bit7: ADCEN: ADC Enable Bit
0: ADC Disabled. ADC is in low power shutdown.
1: ADC Enabled. ADC is active and ready for data conversions.

Bit6: ADCTM: ADC Track Mode Bit
0: When the ADC is enabled, tracking is continuous unless a conversion is in process
1: Tracking Defined by ADSTM1-0 bits
ADSTM1-0:
00: Tracking starts with the write of 1 to ADBUSY and lasts for 3 SAR clocks
01: RESERVED
10: RESERVED
11: Tracking started by the overflow of Timer 2 and last for 3 SAR clocks

Bit5: ADCINT: ADC Conversion Complete Interrupt Flag (cleared by software).
0: ADC has not completed a data conversion since the last time this flag was cleared
1: ADC has completed a data conversion

Bit4: ADBUSY: ADC Busy Bit
Read
0: ADC Conversion complete or no valid data has been converted since a reset. The falling edge of ADBUSY generates an interrupt when enabled.
1: ADC Busy converting data
Write
0: No effect
1: Starts ADC Conversion if ADSTM1-0 = 00b

Bits3-2: ADSTM1-0: ADC Start of Conversion Mode Bits
00: ADC conversion started upon a write of 1 to ADBUSY
01: RESERVED
10: RESERVED
11: ADC conversions initiated on overflows of Timer 2

Bit1: ADWINT: ADC Window Compare Interrupt Flag
0: ADC Window Comparison Data match has not occurred
1: ADC Window Comparison Data match occurred

Bit0: ADLJST: ADC Left Justify Data Bit
0: Data in ADC0H:ADC0L registers are right justified.
1: Data in ADC0H:ADC0L registers are left justified.

Figure 6.6. ADC0H: ADC Data Word MSB Register (C8051F206)

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: 0xBF

Bits7-0: ADC Data Word Bits
For ADLJST = 1: Upper 8-bits of the 12-bit ADC Data Word.
For ADLJST = 0: Bits7-4 are the sign extension of Bit3. Bits 3-0 are the upper 4-bits of the 12-bit ADC Data Word.

Figure 6.7. ADC0L: ADC Data Word LSB Register (C8051F206)

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: 0xBE

Bits7-0: ADC Data Word Bits
For ADLJST = 1: Bits7-4 are the lower 4-bits of the 12-bit ADC Data Word. Bits3-0 will always read 0.
For ADLJST = 0: Bits7-0 are the lower 8-bits of the 12-bit ADC Data Word.

NOTE: Resulting 12-bit ADC Data Word appears in the ADC Data Word Registers as follows:
ADC0H[3:0]:ADC0L[7:0], if ADLJST = 0
(ADC0H[7:4] will be sign extension of ADC0H.3 if a differential reading, otherwise = 0000b)

ADC0H[7:0]:ADC0L[7:4], if ADLJST = 1
(ADC0L[3:0] = 0000b)

EXAMPLE: ADC Data Word Conversion Map, AIN0 Input in Single-Ended Mode
(AMX0CF=0x00, AMX0SL=0x00)

AIN0 – AGND (Volts)	**ADC0H:ADC0L** (ADLJST = 0)	**ADC0H:ADC0L** (ADLJST = 1)
REF x (4095/4096)	0x0FFF	0xFFF0
REF x ½	0x0800	0x8000
REF x (2047/4096)	0x07FF	0x7FF0
0	0x0000	0x0000

6.3. ADC Programmable Window Detector

The ADC programmable window detector is very useful in many applications. It continuously compares the ADC output to user-programmed limits and notifies the system when an out-of-band condition is detected. This is especially effective in an interrupt-driven system, saving code space and CPU bandwidth while delivering faster system response times. The window detector interrupt flag (ADWINT in ADC0CN) can also be used in polled mode. The high and low bytes of the reference words are loaded into the ADC Greater-Than and ADC Less-Than registers (ADC0GTH, ADC0GTL, ADC0LTH, and ADC0LTL). Figure 6.12 and Figure 6.13 show example comparisons for reference. Notice that the window detector flag can be asserted when the measured data is inside or outside the user-programmed limits, depending on the programming of the ADC0GTx and ADC0LTx registers.

Figure 6.8. ADC0GTH: ADC Greater-Than Data High Byte Register (C8051F206)

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								11111111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xC5

Bits7-0:
The high byte of the ADC Greater-Than Data Word.

Figure 6.9. ADC0GTL: ADC Greater-Than Data Low Byte Register (C8051F206)

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								11111111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xC4

Bits7-0:
The low byte of the ADC Greater-Than Data Word.

Definition:
ADC Greater-Than Data Word = ADC0GTH:ADC0GTL

Figure 6.10. ADC0LTH: ADC Less-Than Data High Byte Register (C8051F206)

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: 0xC7

Bits7-0:
The high byte of the ADC Less-Than Data Word.

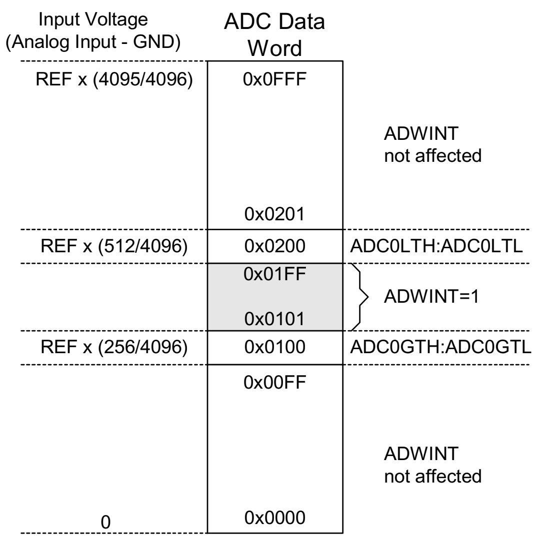
Figure 6.11. ADC0LTL: ADC Less-Than Data Low Byte Register (C8051F206)

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: 0xC6

Bits7-0:
These bits are the low byte of the ADC Less-Than Data Word.

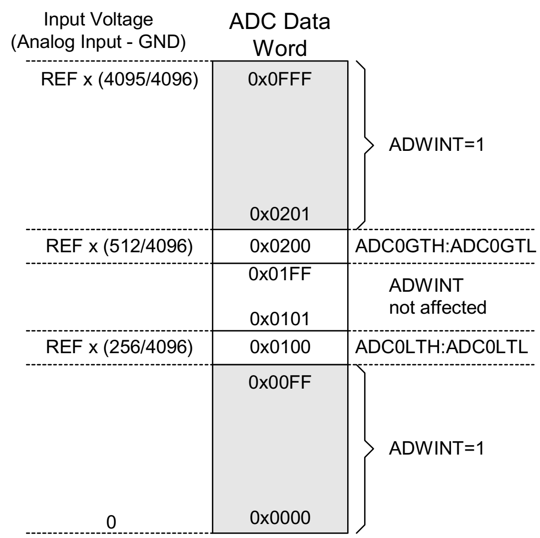
Definition:
ADC Less-Than Data Word = ADC0LTH:ADC0LTL

Figure 6.12. 12-Bit ADC Window Interrupt Examples, Right Justified Data



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

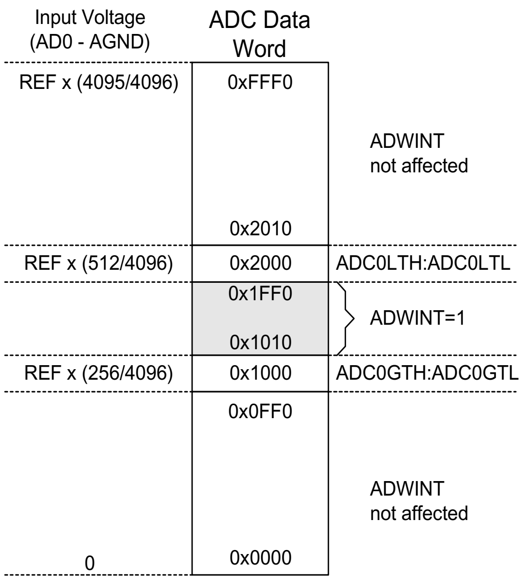
An ADC End of Conversion will cause an ADC Window Compare Interrupt (ADWINT=1) if the resulting ADC Data Word is < 0x0200 and > 0x0100.



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

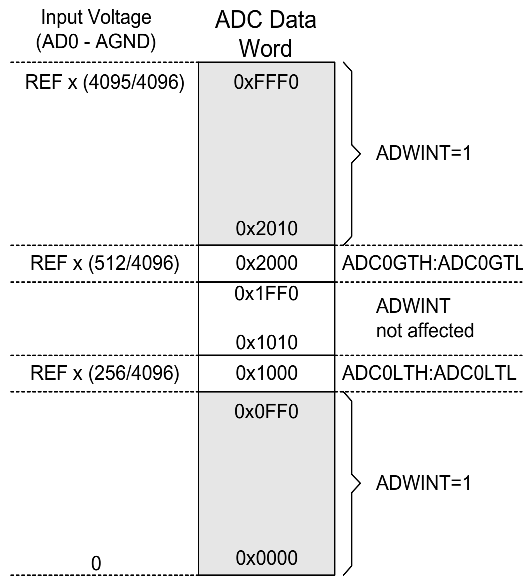
An ADC End of Conversion will cause an ADC Window Compare Interrupt (ADWINT=1) if the resulting ADC Data Word is < 0x0100 or > 0x0200.

Figure 6.13. 12-Bit ADC Window Interrupt Examples, Left Justified Data



Given:
AMX0SL = 0x00, AMX0CF = 0x00, ADLJST = 1,
ADC0LTH:ADC0LTL = 0x2000,
ADC0GTH:ADC0GTL = 0x1000.

An ADC End of Conversion will cause an ADC Window Compare Interrupt (ADWINT=1) if the resulting ADC Data Word is < 0x2000 and > 0x1000.



Given:
AMX0SL = 0x00, AMX0CF = 0x00, ADLJST = 1,
ADC0LTH:ADC0LTL = 0x1000,
ADC0GTH:ADC0GTL = 0x2000.

An ADC End of Conversion will cause an ADC Window Compare Interrupt (ADWINT=1) if the resulting ADC Data Word is < 0x1000 or > 0x2000.

Table 6.1. 12-Bit ADC Electrical Characteristics (C8015F206 only)

VDD = 3.0V, VREF = 2.40V (REFBE=0), PGA Gain = 1, -40°C to +85°C unless otherwise specified.

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS
DC ACCURACY					
Resolution			12		bits
Integral Nonlinearity			± 1	± 2	LSB
Differential Nonlinearity	Guaranteed Monotonic			± 1	LSB
Offset Error			-3 ± 2		LSB
Full Scale Error	Differential mode		-20 ± 3		LSB
Offset Temperature Coefficient			± 0.25		ppm/°C
DYNAMIC PERFORMANCE (10kHz sine-wave input, 0 to –1dB of full scale, 100ksps)					
Signal-to-Noise Plus Distortion		64			dB
Total Harmonic Distortion	Up to the 5th harmonic		-75		dB
Spurious-Free Dynamic Range			80		dB
CONVERSION RATE					
Conversion Time in System Clocks	ADC0CF = 000xxxxxb	16			clocks
Track/Hold Acquisition Time		1.5			µs
Throughput Rate				100	ksps
ANALOG INPUTS					
Voltage Conversion Range		0		VREF	V
Input Voltage	Any pin (in Analog Input Mode)	GND		VDD	V
Input Capacitance			10		pF
POWER SPECIFICATIONS					
Power Supply Current (VDD supplied to ADC)	Operating Mode, 100ksps		450	900	µA
Power Supply Rejection			± 0.3		mV/V

7. VOLTAGE REFERENCE (C8051F206/220/221/226)

The voltage reference circuit selects between an externally connected reference and the power supply voltage (VDD). (See Figure 7.1).

An external reference can be connected to the VREF pin and selected by setting the REF0CN special function register per Figure 7.1. The external reference supply must be between VDD-0.3V and 1V. VDD may also be selected using REF0CN per Figure 7.2. The electrical specifications for the Voltage Reference are given in Table 7.1

Figure 7.1. Voltage Reference Functional Block Diagram

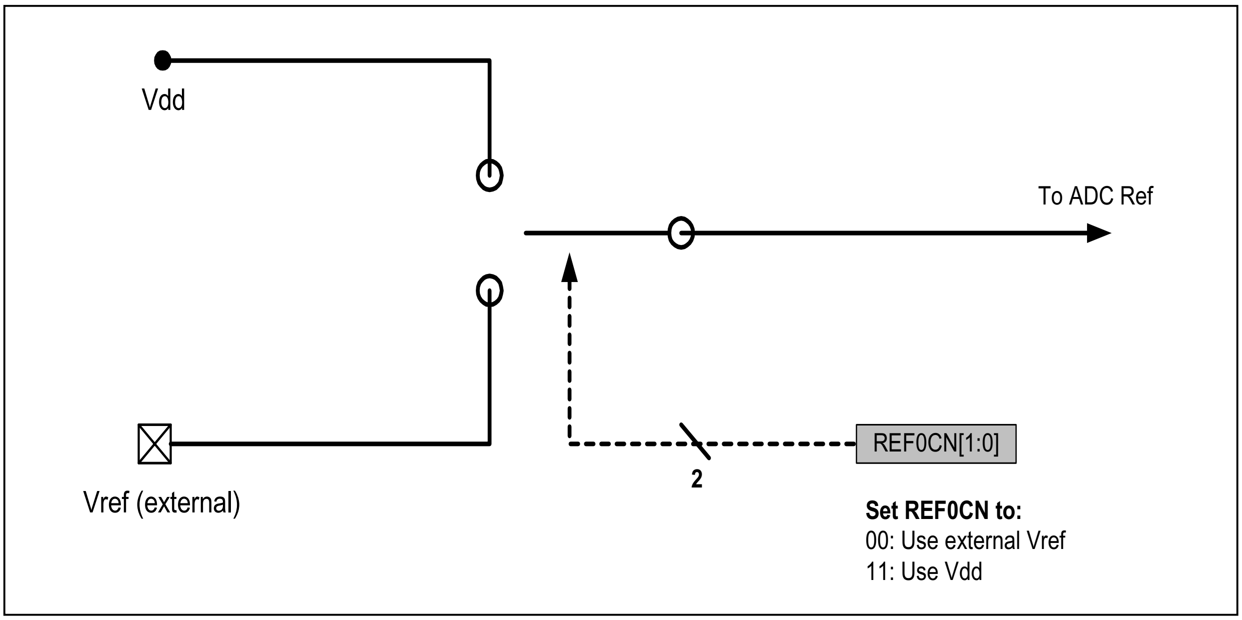


Figure 7.2. REF0CN: Reference Control Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
-	-	-	-	-	-	REFSL1	REFSL0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xD1

Bits7-2: UNUSED. Read = 00000b; Write = don't care

Bit1-0: REFSL1- REFSL0: Voltage reference selection.
Bits control which reference is selected.
00: External VREF source is selected.
01: Reserved.
10: Reserved.
11: VDD selected as VREF source.

Table 7.1. Reference Electrical Characteristics

EXTERNAL REFERENCE ([REFSL1: REFSL0] = 00), VREF = 2.4V)		MIN	TYP	MAX	UNITS
Input Voltage Range		1.00		(VDD) – 0.3V	V
Input Current			0.1	10	µA
Input Resistance		100			MΩ

8. COMPARATORS

The MCU has two on-board voltage comparators as shown in Figure 8.1. The inputs of each Comparator are available at the package pins. The output of each comparator is optionally available at port1 by configuring (see Section 14). When assigned to package pins, each comparator output can be programmed to operate in open drain or push-pull modes (see section 14.2).

The hysteresis of each comparator is software-programmable via its respective Comparator Control Register (CPT0CN, CPT1CN). The user can program both the amount of hysteresis voltage (referred to the input voltage) and the positive-going and negative-going symmetry of this hysteresis around the threshold voltage. The output of the comparator can be polled in software, or can be used as an interrupt source. Each comparator can be individually enabled or disabled (shutdown). When disabled, the comparator output (if assigned to a Port I/O pin via the Port1 MUX) defaults to the logic low state and its interrupt capability is suspended. Comparator inputs can be externally driven from -0.25V to (VDD) + 0.25V without damage or upset.

The Comparator 0 hysteresis is programmed using bits 3-0 in the Comparator 0 Control Register CPT0CN (shown in Figure 8.3). The amount of *negative* hysteresis voltage is determined by the settings of the CP0HYN bits. As shown in Figure 8.2, settings of 10, 4 or. 2mV of negative hysteresis can be programmed, or negative hysteresis can be disabled. In a similar way, the amount of *positive* hysteresis is determined by the setting the CP0HYP bits.

Comparator interrupts can be generated on both rising-edge and falling-edge output transitions. (For Interrupt enable and priority control, see Section 9.4). The CP0FIF flag is set upon a Comparator 0 falling-edge interrupt, and the CP0RIF flag is set upon the Comparator 0 rising-edge interrupt. Once set, these bits remain set until cleared by the user software. The Output State of Comparator 0 can be obtained at any time by reading the CP0OUT bit. Comparator 0 is enabled by setting the CP0EN bit, and is disabled by clearing this bit. Note there is a 20µS power on time between setting CP0EN and the output stabilizing. Comparator 0 can also be programmed as a reset source. For details, see Section 11. The operation of Comparator 1 is identical to that of Comparator 0, except the Comparator 1 is controlled by the CPT1CN Register (Figure 8.4). Also, Comparator 1 can not be programmed as a reset source. The complete electrical specifications for the Comparators are given in Table 8.1.

Figure 8.1. Comparator Functional Block Diagram

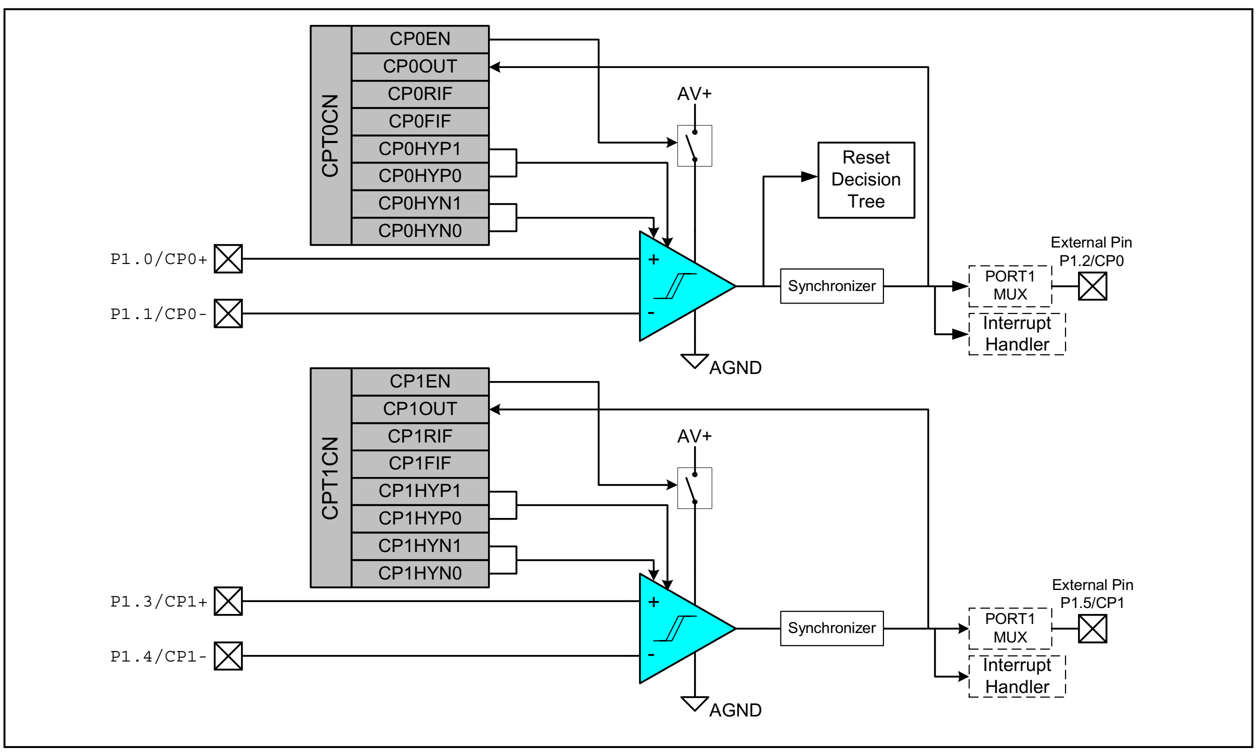


Figure 8.2. Comparator Hysteresis Plot

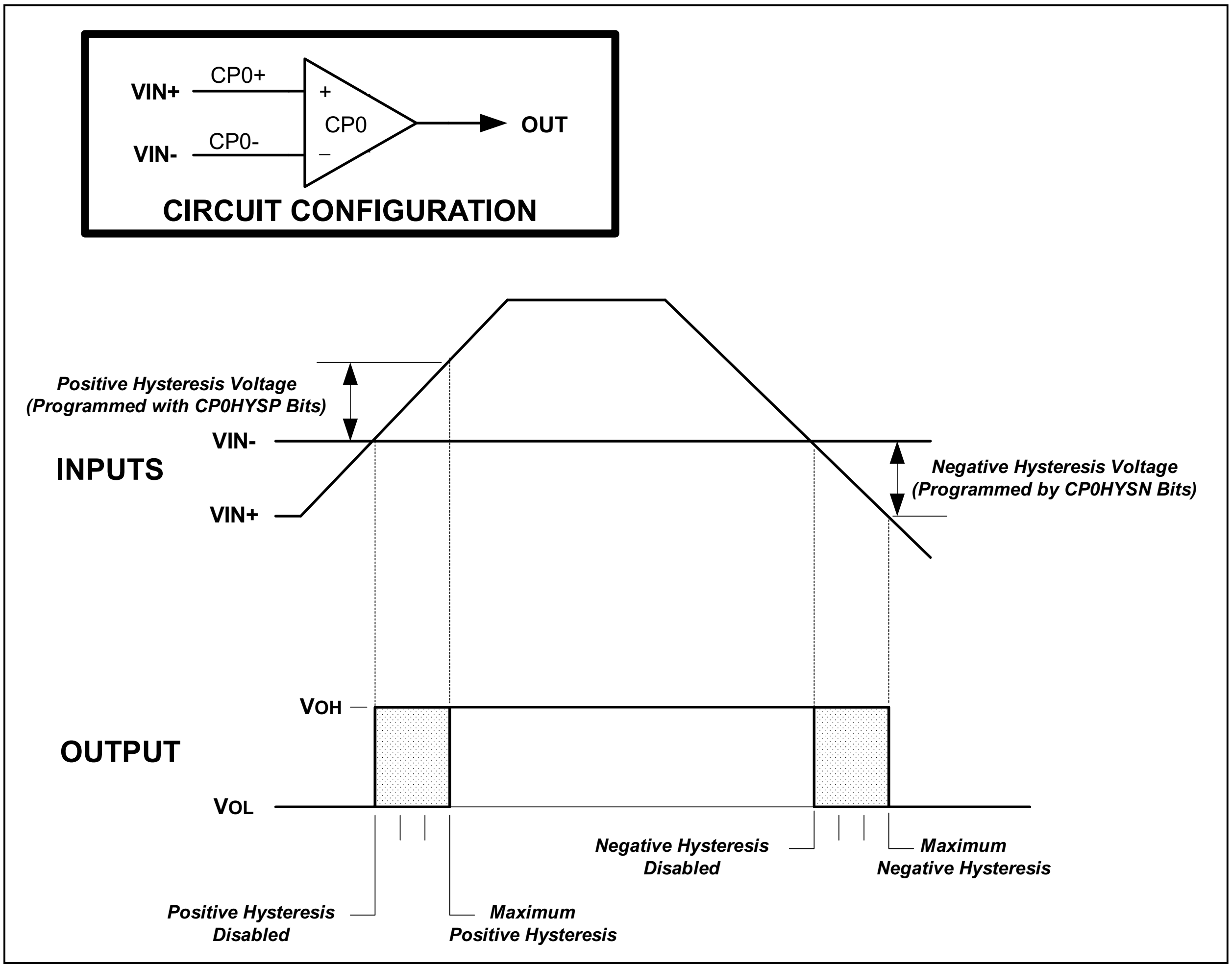


Figure 8.3. CPT0CN: Comparator 0 Control Register

R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
CP0EN	CP0OUT	CP0RIF	CP0FIF	CP0HYP1	CP0HYP0	CP0HYN1	CP0HYN0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0x9E

Bit7: CP0EN: Comparator 0 Enable Bit
0: Comparator 0 Disabled.
1: Comparator 0 Enabled.

Bit6: CP0OUT: Comparator 0 Output State Flag
0: Voltage on CP0+ < CP0-
1: Voltage on CP0+ > CP0-

Bit5: CP0RIF: Comparator 0 Rising-Edge Interrupt Flag
0: No Comparator 0 Rising-Edge Interrupt has occurred since this flag was cleared
1: Comparator 0 Rising-Edge Interrupt has occurred since this flag was cleared

Bit4: CP0FIF: Comparator 0 Falling-Edge Interrupt Flag
0: No Comparator 0 Falling-Edge Interrupt has occurred since this flag was cleared
1: Comparator 0 Falling-Edge Interrupt has occurred since this flag was cleared

Bit3-2: CP0HYP1-0: Comparator 0 Positive Hysteresis Control Bits
00: Positive Hysteresis Disabled
01: Positive Hysteresis = 2mV
10: Positive Hysteresis = 4mV
11: Positive Hysteresis = 10mV

Bit1-0: CP0HYN1-0: Comparator 0 Negative Hysteresis Control Bits
00: Negative Hysteresis Disabled
01: Negative Hysteresis = 2mV
10: Negative Hysteresis = 4mV
11: Negative Hysteresis = 10mV

Figure 8.4. CPT1CN: Comparator 1 Control Register

R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
CP1EN	CP1OUT	CP1RIF	CP1FIF	CP1HYP1	CP1HYP0	CP1HYN1	CP1HYN0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0x9F

Bit7: CP1EN: Comparator 1 Enable Bit
0: Comparator 1 Disabled.
1: Comparator 1 Enabled.

Bit6: CP1OUT: Comparator 1 Output State Flag
0: Voltage on CP1+ < CP1-
1: Voltage on CP1+ > CP1-

Bit5: CP1RIF: Comparator 1 Rising-Edge Interrupt Flag
0: No Comparator 1 Rising-Edge Interrupt has occurred since this flag was cleared
1: Comparator 1 Rising-Edge Interrupt has occurred since this flag was cleared

Bit4: CP1FIF: Comparator 1 Falling-Edge Interrupt Flag
0: No Comparator 1 Falling-Edge Interrupt has occurred since this flag was cleared
1: Comparator 1 Falling-Edge Interrupt has occurred since this flag was cleared

Bit3-2: CP1HYP1-0: Comparator 1 Positive Hysteresis Control Bits
00: Positive Hysteresis Disabled
01: Positive Hysteresis = 2mV
10: Positive Hysteresis = 4mV
11: Positive Hysteresis = 10mV

Bit1-0: CP1HYN1-0: Comparator 1 Negative Hysteresis Control Bits
00: Negative Hysteresis Disabled
01: Negative Hysteresis = 2mV
10: Negative Hysteresis = 4mV
11: Negative Hysteresis = 10mV

Table 8.1. Comparator Electrical Characteristics

VDD = 3.0V, -40°C to +85°C unless otherwise specified.

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS
Response Time1	(CP+) – (CP-) = 100mV **NOTE 1**		4		µs
Response Time2	(CP+) – (CP-) = 10mV **NOTE 1**		12		µs
Common Mode Rejection Ratio			1.5	4	mV/V
Positive Hysteresis1	CPnHYP1-0 = 00		0	1	mV
Positive Hysteresis2	CPnHYP1-0 = 01	2	4.5	7	mV
Positive Hysteresis3	CPnHYP1-0 = 10	4	9	15	mV
Positive Hysteresis4	CPnHYP1-0 = 11	10	17	25	mV
Negative Hysteresis1	CPnHYN1-0 = 00		0	1	mV
Negative Hysteresis2	CPnHYN1-0 = 01	2	4.5	7	mV
Negative Hysteresis3	CPnHYN1-0 = 10	4	9	15	mV
Negative Hysteresis4	CPnHYN1-0 = 11	10	17	25	mV
Inverting or Non-inverting Input Voltage Range		-0.25		(VDD) + 0.25	V
Input Capacitance			7		pF
Input Bias Current		-5	0.001	+5	nA
Input Offset Voltage		-10		+10	mV
POWER SUPPLY					
Power-up Time	CPnEN from 0 to 1		20		µs
Power Supply Rejection			0.1	1	mV/V
Supply Current	Operating Mode (each comparator) at DC		1.5	4	µA

NOTES: (1) CPnHYP1-0 = CPnHYN1-0 = 00.

9. CIP-51 MICROCONTROLLER

General Description

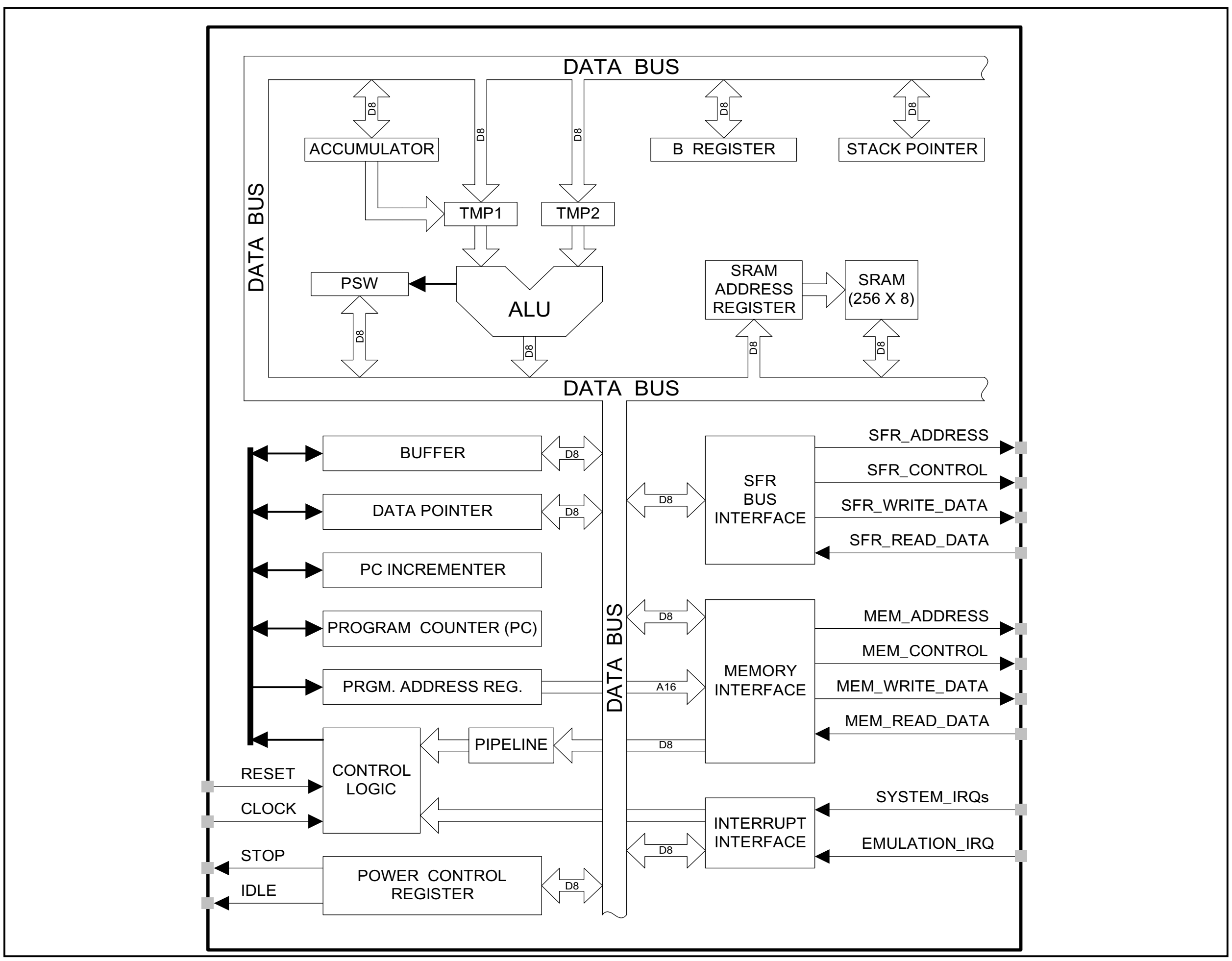
The MCU's system controller core is the CIP-51 microcontroller. The CIP-51 is fully compatible with the MCS-51™ instruction set. Standard 803x/805x assemblers and compilers can be used to develop software. The MCU has a superset of all the peripherals included with a standard 8051. Included are three 16-bit counter/timers (see description in Section 17), a full-duplex UART (see description in Section 16), 256 bytes of internal RAM, 128 byte Special Function Register (SFR) address space (see Section 9.3), and four byte-wide I/O Ports (see description in Section 14). The CIP-51 also includes on-chip debug hardware (see description in Section 18), and interfaces directly with the MCU's analog and digital subsystems providing a complete data acquisition or control-system solution in a single integrated circuit.

Features

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 9.1 for a block diagram). The CIP-51 includes the following features:

- Fully Compatible with MCS-51 Instruction Set
- 25 MIPS Peak Throughput with 25MHz Clock
- 0 to 25MHz Clock Frequency
- 256 Bytes of Internal RAM
- Optional 1024 Bytes of XRAM
- 8k Byte Flash Program Memory
- Four Byte-Wide I/O Ports
- Extended Interrupt Handler
- Reset Input
- Power Management Modes
- On-chip Debug Circuitry
- Program and Data Memory Security

Figure 9.1. CIP-51 Block Diagram



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 12MHz. 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.

With the CIP-51's maximum system clock at 25MHz, it has a peak throughput of 25MIPS. The CIP-51 has a total of 109 instructions. The number of instructions versus the system clock cycles required to execute them is as follows:

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

Programming and Debugging Support

A JTAG-based serial interface is provided for in-system programming of the Flash program memory and communication with on-chip debug support circuitry. The reprogrammable Flash can also be read and changed a single byte at a time by the application software using the MOVC and MOVX instructions. This feature allows program memory to be used for non-volatile data storage as well as updating program code under software control.

The on-chip debug support circuitry facilitates full speed in-circuit debugging, allowing the setting of hardware breakpoints and watchpoints, starting, stopping and single stepping through program execution (including interrupt service routines), examination of the program's call stack, and reading/writing the contents of registers and memory. This method of on-chip debugging is completely non-intrusive and non-evasive, requiring no RAM, Stack, timers, or other on-chip resources.

The CIP-51 is supported by development tools from Cygnal Integrated Products and third party vendors. Cygnal provides an integrated development environment (IDE) including editor, macro assembler, debugger and programmer. The IDE's debugger and programmer interface to the CIP-51 via its JTAG interface to provide fast and efficient in-system device programming and debugging. Third party macro assemblers and C compilers are also available.

9.1. INSTRUCTION SET

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

9.1.1. Instruction and CPU Timing

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

Due to the pipelined architecture of the CIP-51, most instructions execute in the same number of clock cycles as there are program bytes in the instruction. Conditional branch instructions take one less clock cycle to complete when the branch is not taken as opposed to when the branch is taken. Table 9.1 is the CIP-51 Instruction Set Summary, which includes the mnemonic, number of bytes, and number of clock cycles for each instruction.

9.1.2. MOVX Instruction and Program Memory

The MOVX instruction is typically used to access external data memory. The CIP-51 does not support external data or program memory. In the CIP-51, the MOVX instruction accesses the on-chip program memory space implemented as re-programmable Flash memory and the 1024 bytes of XRAM (optionally available on 'F226/236 and 'F206). 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 10 (Flash Memory) and Section 11 (External RAM) for further details.

Table 9.1. 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

Mnemonic	Description	Bytes	Clock Cycles
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
RR A	Rotate A right	1	1
RRC A	Rotate A right through carry	1	1
SWAP A	Swap nibbles of A	1	1
	DATA TRANSFER		
MOV A,Rn	Move register to A	1	1
MOV A,direct	Move direct byte to A	2	2
MOV A,@Ri	Move indirect RAM to A	1	2
MOV A,#data	Move immediate to A	2	2
MOV Rn,A	Move A to register	1	1
MOV Rn,direct	Move direct byte to register	2	2
MOV Rn,#data	Move immediate to register	2	2
MOV direct,A	Move A to direct byte	2	2
MOV direct,Rn	Move register to direct byte	2	2
MOV direct,direct	Move direct byte to direct	3	3
MOV direct,@Ri	Move indirect RAM to direct byte	2	2
MOV direct,#data	Move immediate to direct byte	3	3
MOV @Ri,A	Move A to indirect RAM	1	2
MOV @Ri,direct	Move direct byte to indirect RAM	2	2
MOV @Ri,#data	Move immediate to indirect RAM	2	2
MOV DPTR,#data16	Load data pointer with 16-bit constant	3	3
MOVC A,@A+DPTR	Move code byte relative DPTR to A	1	3
MOVC A,@A+PC	Move code byte relative PC to A	1	3
MOVX A,@Ri	Move external data (8-bit address) to A	1	3
MOVX @Ri,A	Move A to external data (8-bit address)	1	3
MOVX A,@DPTR	Move external data (16-bit address) to A	1	3
MOVX @DPTR,A	Move A to external data (16-bit address)	1	3

Mnemonic	Description	Bytes	Clock Cycles
PUSH direct	Push direct byte onto stack	2	2
POP direct	Pop direct byte from stack	2	2
XCH A,Rn	Exchange register with A	1	1
XCH A,direct	Exchange direct byte with A	2	2
XCH A,@Ri	Exchange indirect RAM with A	1	2
XCHD A,@Ri	Exchange low nibble of indirect RAM with A	1	2
BOOLEAN MANIPULATION			
CLR C	Clear carry	1	1
CLR bit	Clear direct bit	2	2
SETB C	Set carry	1	1
SETB bit	Set direct bit	2	2
CPL C	Complement carry	1	1
CPL bit	Complement direct bit	2	2
ANL C,bit	AND direct bit to carry	2	2
ANL C,/bit	AND complement of direct bit to carry	2	2
ORL C,bit	OR direct bit to carry	2	2
ORL C,/bit	OR complement of direct bit to carry	2	2
MOV C,bit	Move direct bit to carry	2	2
MOV bit,C	Move carry to direct bit	2	2
JC rel	Jump if carry is set	2	2/3
JNC rel	Jump if carry not set	2	2/3
JB bit,rel	Jump if direct bit is set	3	3/4
JNB bit,rel	Jump if direct bit is not set	3	3/4
JBC bit,rel	Jump if direct bit is set and clear bit	3	3/4
PROGRAM BRANCHING			
ACALL addr11	Absolute subroutine call	2	3
LCALL addr16	Long subroutine call	3	4
RET	Return from subroutine	1	5
RETI	Return from interrupt	1	5
AJMP addr11	Absolute jump	2	3
LJMP addr16	Long jump	3	4
SJMP rel	Short jump (relative address)	2	3
JMP @A+DPTR	Jump indirect relative to DPTR	1	3
JZ rel	Jump if A equals zero	2	2/3
JNZ rel	Jump if A does not equal zero	2	2/3
CJNE A,direct,rel	Compare direct byte to A and jump if not equal	3	3/4
CJNE A,#data,rel	Compare immediate to A and jump if not equal	3	3/4
CJNE Rn,#data,rel	Compare immediate to register and jump if not equal	3	3/4
CJNE @Ri,#data,rel	Compare immediate to indirect and jump if not equal	3	4/5
DJNZ Rn,rel	Decrement register and jump if not zero	2	2/3
DJNZ direct,rel	Decrement direct byte and jump if not zero	3	3/4
NOP	No operation	1	1

Notes on Registers, Operands and Addressing Modes:
Rn - Register R0-R7 of the currently selected register bank. **@Ri -** Data RAM location addressed indirectly through register R0-R1 **rel -** 8-bit, signed (two's compliment) offset relative to the first byte of the following instruction. Used by SJMP and all conditional jumps. **direct -** 8-bit internal data location's address. This could be a direct-access Data RAM location (0x00-0x7F) or an SFR (0x80-0xFF). **#data -** 8-bit constant **#data 16 -** 16-bit constant **bit -** Direct-addressed bit in Data RAM or SFR. **addr 11 -** 11-bit destination address used by ACALL and AJMP. The destination must be within the same 2K-byte page of program memory as the first byte of the following instruction. **addr 16 -** 16-bit destination address used by LCALL and LJMP. The destination may be anywhere within the 8K-byte program memory space. There is one unused opcode (0xA5) that performs the same function as NOP. All mnemonics copyrighted © Intel Corporation 1980.

9.2. 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 8K bytes of internal program memory address space implemented within the CIP-51. The CIP-51 memory organization is shown in Figure 9.2.

9.2.1. Program Memory

The CIP-51 has a 8K-byte program memory space. The MCU implements 8320 bytes of this program memory space as in-system, reprogrammable Flash memory, organized in a contiguous block from addresses 0x0000 to 0x207F. Note: 512 bytes (0x1E00 – 0x1FFF) 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 (PSCTL.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 10 Flash Memory for further details.

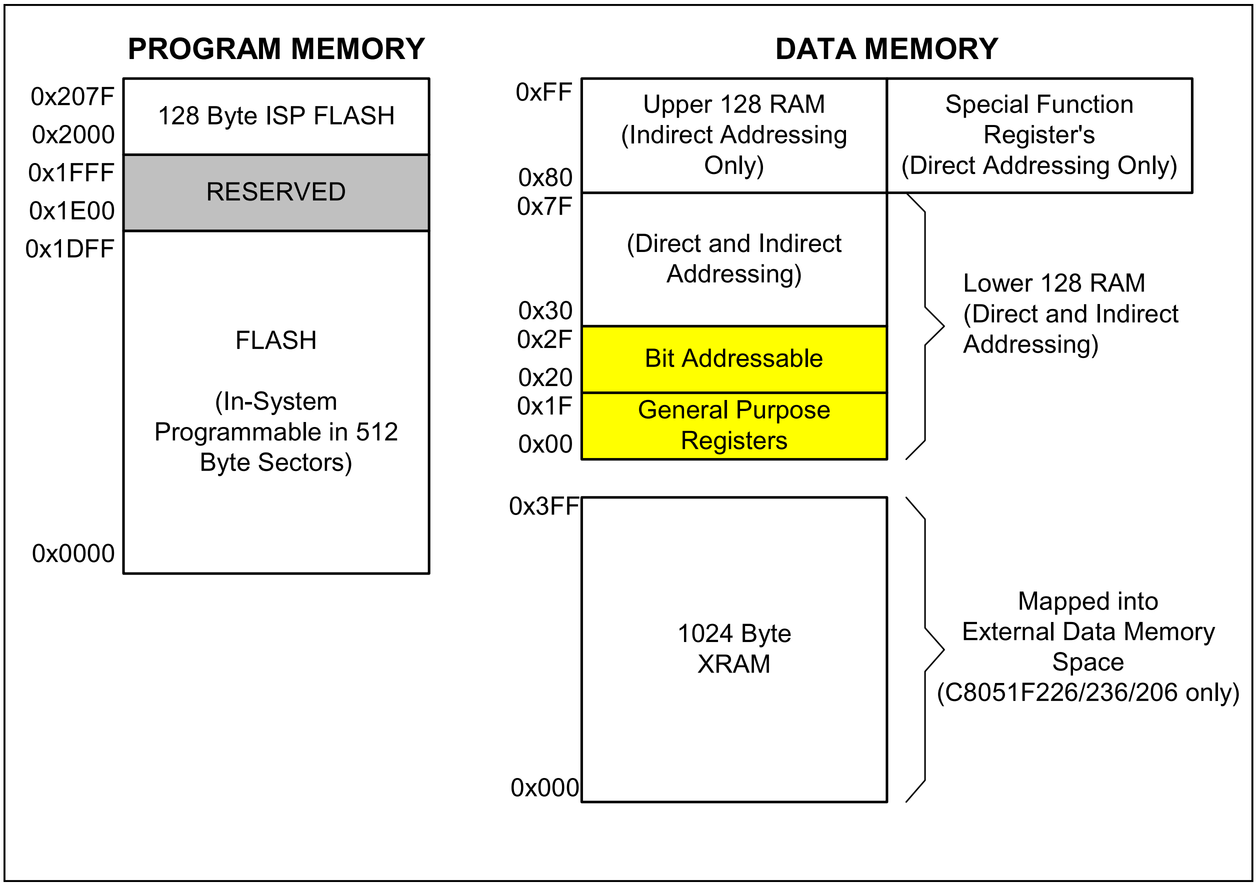
9.2.2. Data Memory

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 either 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 9.2 illustrates the data memory organization of the CIP-51.

Additionally, the C8051F206/226/236 feature 1024 Bytes of RAM mapped in the external data memory space. All address locations may be accessed using the MOVX instruction. (Please see Section 11).

Figure 9.2. Memory Map



9.2.3. 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 9.6). This allows fast context switching when entering subroutines and interrupt service routines. Indirect addressing modes use registers R0 and R1 as index registers.

9.2.4. 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.

9.2.5. Stack

A programmer's stack can be located anywhere in the 256-byte data memory. The stack area is designated using the Stack Pointer (SP, 0x81) SFR. The SP will point to the last location used. The next value pushed on the stack is placed at SP+1 and then SP is incremented. A reset initializes the stack pointer to location 0x07. Therefore, the first value pushed on the stack is placed at location 0x08, which is also the first register (R0) of register bank 1. Thus, if more than one register bank is to be used, the SP should be initialized to a location in the data memory not being used for data storage. The stack depth can extend up to 256 bytes.

The MCU also has built-in hardware for a stack record. The stack record is a 32-bit shift register, where each Push or increment SP pushes one record bit onto the register, and each Call pushes two record bits onto the register. (A Pop or decrement SP pops one record bit, and a Return pops two record bits, also.) The stack record circuitry can also detect an overflow or underflow on the 32-bit shift register, and can notify the emulator software even with the MCU running full-speed debug.

9.3. 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 9.3 lists the SFRs implemented in the CIP-51 System Controller.

The SFR registers are accessed anytime 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 9.3, for a detailed description of each register.

Table 9.2. Special Function Register Memory Map

	0(8)	1(9)	2(A)	3(B)	4(C)	5(D)	6(E)	7(F)
F8	SPI0CN							WDTCN
F0	B	P0MODE	P1MODE	P2MODE	P3MODE[2]		EIP1	EIP2
E8	ADC0CN[1]							RSTSRC
E0	ACC	PRT0MX	PRT1MX	PRT2MX			EIE1	EIE2
D8								
D0	PSW	REF0CN						
C8	T2CON		RCAP2L	RCAP2H	TL2	TH2		
C0					ADC0GTL[4]	ADC0GTH[1]	ADC0LTL[4]	ADC0LTH[1]
B8	IP			AMX0SL[1]	ADC0CF[1]		ADC0L[4]	ADC0H[1]
B0	P3	OSCXCN	OSCICN				FLSCL	FLACL
A8	IE					PRT1IF		EMI0CN[3]
A0	P2				PRT0CF	PRT1CF	PRT2CF	PRT3CF
98	SCON	SBUF	SPI0CFG	SPI0DAT		SPI0CKR	CPT0CN	CPT1CN
90	P1							
88	TCON	TMOD	TL0	TL1	TH0	TH1	CKCON	PSCTL
80	P0	SP	DPL	DPH				PCON

Bit Addressable

[1]C8051F230/1/6 Do not have these registers.
[2]C8051F221/231 Does not have this register (32 pin package).
[3]On the C8051F206 and C8051F226/236 only.
[4]On the C8051F206 only (12-bit ADC)

Table 9.3. Special Function Registers

SFR's are listed in alphabetical order.

Address	**Register**	Description	Page No.
0xE0	ACC	Accumulator	64
0xBC	ADC0CF	ADC Configuration	28
0xE8	ADC0CN	ADC Control	29
0xC5	ADC0GTH[1]	ADC Greater-Than Data Word (High Byte)	30
0xC4	ADC0GTL[4]	ADC Greater Than Data Word (Low Byte)	39

Address	Register	Description	Page No.
0xBF	ADC0H[1]	ADC Data Word (High Byte)	30
0xBE	ADC0L[4]	ADC Data Word (Low Byte)	38
0xC7	ADC0LTH[1]	ADC Less-Than Data Word (High Byte)	30
0xCE	ADC0LTL[4]	ADC Less Than Data Word (Low Byte)	39
0xBB	AMX0SL	ADC MUX Channel Selection	27
0xF0	B	B Register	64
0x8E	CKCON	Clock Control	121
0x9E	CPT0CN	Comparator 0 Control	46
0x9F	CPT1CN	Comparator 1 Control	48
0x83	DPH	Data Pointer (High Byte)	62
0x82	DPL	Data Pointer (Low Byte)	62
0xE6	EIE1	Extended Interrupt Enable 1	69
0xE7	EIE2	Extended Interrupt Enable 2	70
0xF6	EIP1	External Interrupt Priority 1	71
0xF7	EIP2	External Interrupt Priority 2	72
0xAF	EMI0CN[3]	External Memory Interface Control	80
0xB7	FLACL	Flash Memory Read Limit	79
0xB6	FLSCL	Flash Memory Timing Prescaler	79
0xA8	IE	Interrupt Enable	67
0xB8	IP	Interrupt Priority Control	68
0xB2	OSCICN	Internal Oscillator Control	88
0xB1	OSCXCN	External Oscillator Control	89
0x80	P0	Port 0 Latch	95
0x90	P1	Port 1 Latch	96
0xA0	P2	Port 2 Latch	98
0xB0	P3	Port 3 Latch	98
0xF1	P0MODE	Port0 Digital/Analog Output Mode	98
0xF2	P1MODE	Port1 Digital/Analog Output Mode	98
0xF3	P2MODE	Port2 Digital/Analog Output Mode	98
0xF4	P3MODE[2]	Port3 Digital/Analog Output Mode	88
0x87	PCON	Power Control	74
0xA4	PRT0CF	Port 0 Configuration	95
0xA5	PRT1CF	Port 1 Configuration	96
0xAD	PRT1IF	Port 1 Interrupt Flags	97
0xA6	PRT2CF	Port 2 Configuration	98
0xA7	PRT3CF	Port 3 Configuration	99
0xE1	PRT0MX	Port 0 Multiplexer I/O Configuration	79
0xE2	PRT1MX	Port 1 Multiplexer I/O Configuration	80
0xE3	PRT2MX	Port 2 Multiplexer I/O Configuration	80
0x8F	PSCTL	Program Store RW Control	78
0xD0	PSW	Program Status Word	63

Address	Register	Description	Page No.
0xCB	RCAP2H	Counter/Timer 2 Capture (High Byte)	128
0xCA	RCAP2L	Counter/Timer 2 Capture (Low Byte)	128
0xD1	REF0CN	Voltage Reference Control Register	44
0xEF	RSTSRC	Reset Source Register	85
0x99	SBUF	Serial Data Buffer (UART)	113
0x98	SCON	Serial Port Control (UART)	114
0x81	SP	Stack Pointer	62
0x9A	SPI0CFG	Serial Peripheral Interface Configuration	105
0x9D	SPI0CKR	SPI Clock Rate	107
0xF8	SPI0CN	SPI Bus Control	106
0x9B	SPI0DAT	SPI Port 1Data	107
0xC8	T2CON	Counter/Timer 2 Control	127
0x88	TCON	Counter/Timer Control	119
0x8C	TH0	Counter/Timer 0 Data Word (High Byte)	122
0x8D	TH1	Counter/Timer 1 Data Word (High Byte)	122
0xCD	TH2	Counter/Timer 2 Data Word (High Byte)	128
0x8A	TL0	Counter/Timer 0 Data Word (Low Byte)	122
0x8B	TL1	Counter/Timer 1 Data Word (Low Byte)	122
0xCC	TL2	Counter/Timer 2 Data Word (Low Byte)	128
0x89	TMOD	Counter/Timer Mode	120
0xFF	WDTCN	Watchdog Timer Control	84
0x84-86, 0x91-97, 0x9C, 0xA1-A3, 0xA9-AC, 0xAE, 0xB3-B5, 0xB9-BA, 0xBD-BE,0xC0-C4, 0xC6,0xCE-CF,0xD2-DF,0xE9-EE,0xF5,0xF9-FE		Reserved	

[1]C8051F230/1/6 Do not have these registers.
[2]C8051F221/231 Does not have this register (32 pin package).
[3]On the C8051F206 and C8051F226/236 only.
[4]On the C8051F206 only (12-bit ADC)

9.3.1. Register Descriptions

Following are descriptions of SFRs related to the operation of the CIP-51 System Controller. Reserved bits should be set to logic 0. Future product versions may use these bits to implement new features in which case the reset value of the bit will be logic 0, selecting the feature's default state. Detailed descriptions of the remaining SFRs are included in the sections of the datasheet associated with their corresponding system function.

Figure 9.3. SP: Stack Pointer

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0x81

Bits 7-0: SP: Stack Pointer.
The stack pointer holds the location of the top of the stack. The stack pointer is incremented before every PUSH operation. The SP register defaults to 0x07 after reset.

Figure 9.4. DPL: Data Pointer Low Byte

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: 0x82

Bits 7-0: DPL: Data Pointer Low.
The DPL register is the low byte of the 16-bit DPTR. DPTR is used to access indirectly addressed RAM.

Figure 9.5. DPH: Data Pointer High Byte

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: 0x83

Bits 7-0: DPH: Data Pointer High.
The DPH register is the high byte of the 16-bit DPTR. DPTR is used to access indirectly addressed RAM.

Figure 9.6. PSW: Program Status Word

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
CY	AC	F0	RS1	RS0	OV	F1	PARITY	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
							(bit addressable)	0xD0

Bit7: CY: Carry Flag.
This bit is set when the last arithmetic operation results in a carry (addition) or a borrow (subtraction). It is cleared to 0 by all other arithmetic operations.

Bit6: AC: Auxiliary Carry Flag.
This bit is set when the last arithmetic operation results in a carry into (addition) or a borrow from (subtraction) the high order nibble. It is cleared to 0 by all other arithmetic operations.

Bit5: F0: User Flag 0.
This is a bit-addressable, general-purpose flag for use under software control.

Bits4-3: RS1-RS0: Register Bank Select.
These bits select which register bank is used during register accesses.

RS1	RS0	Register Bank	Address
0	0	0	0x00-0x07
0	1	1	0x08-0x0F
1	0	2	0x10-0x17
1	1	3	0x18-0x1F

Bit2: OV: Overflow Flag.
This bit is set to 1 if the last arithmetic operation resulted in a carry (addition), borrow (subtraction), or overflow (multiply or divide). It is cleared to 0 by all other arithmetic operations.

Bit1: F1: User Flag 1.
This is a bit-addressable, general purpose flag for use under software control.

Bit0: PARITY: Parity Flag.
This bit is set to 1 if the sum of the eight bits in the accumulator is odd and cleared if the sum is even.

Figure 9.7. ACC: Accumulator

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
ACC.7	ACC.6	ACC.5	ACC.4	ACC.3	ACC.2	ACC.1	ACC.0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
							(bit addressable)	0xE0

Bits 7-0: ACC: Accumulator
This register is the accumulator for arithmetic operations.

Figure 9.8. B: B Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
B.7	B.6	B.5	B.4	B.3	B.2	B.1	B.0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
							(bit addressable)	0xF0

Bits 7-0: B: B Register
This register serves as a second accumulator for certain arithmetic operations.

9.4. INTERRUPT HANDLER

The CIP-51 includes an extended interrupt system supporting up to 22 interrupt sources with two priority levels. The allocation of interrupt sources between on-chip peripherals and external inputs pins varies according to the specific version of the device. Each interrupt source has one or more associated interrupt-pending flag(s) located in an SFR. When a peripheral or external source meets a valid interrupt condition, the associated interrupt-pending flag is set to logic 1.

If interrupts are enabled for the source, an interrupt request is generated when the interrupt-pending flag is set. As soon as execution of the current instruction is complete, the CPU generates an LCALL to a predetermined address to begin execution of an interrupt service routine (ISR). Each ISR must end with an RETI instruction, which returns program execution to the next instruction that would have been executed if the interrupt request had not occurred. If interrupts are not enabled, the interrupt-pending flag is ignored by the hardware and program execution continues as normal. (The interrupt-pending flag is set to logic 1 regardless of the interrupt's enable/disable state.)

Each interrupt source can be individually enabled or disabled through the use of an associated interrupt enable bit in an SFR (IE-EIE2). However, interrupts must first be globally enabled by setting the EA bit (IE.7) to logic 1 before the individual interrupt enables are recognized. Setting the EA bit to logic 0 disables all interrupt sources regardless of the individual interrupt-enable settings.

Some interrupt-pending flags are automatically cleared by the hardware when the CPU vectors to the ISR. However, most are not cleared by the hardware and must be cleared by software before returning from the ISR. If an interrupt-pending flag remains set after the CPU completes the return-from-interrupt (RETI) instruction, a new interrupt request will be generated immediately and the CPU will re-enter the ISR after the completion of the next instruction.

9.4.1. MCU Interrupt Sources and Vectors

The MCU allocates 9 interrupt sources to on-chip peripherals. Software can simulate an interrupt by setting any interrupt-pending flag to logic 1. If interrupts are enabled for the flag, an interrupt request will be generated and the CPU will vector to the ISR address associated with the interrupt-pending flag. The MCU interrupt sources, associated vector addresses, priority order and control bits are summarized in Table 9.4. Refer to the datasheet section associated with a particular on-chip peripheral for information regarding valid interrupt conditions for the peripheral and the behavior of its interrupt-pending flag(s).

9.4.2. External Interrupts

Two of the external interrupt sources (/INT0 and /INT1) are configurable as active-low level-sensitive or active-low edge-sensitive inputs depending on the setting of IT0 (TCON.0) and IT1 (TCON.2). IE0 (TCON.1) and IE1 (TCON.3) serve as the interrupt-pending flag for the /INT0 and /INT1 external interrupts, respectively. If an /INT0 or /INT1 external interrupt is configured as edge-sensitive, the corresponding interrupt-pending flag is automatically cleared by the hardware when the CPU vectors to the ISR. When configured as level sensitive, the interrupt-pending flag follows the state of the external interrupt's input pin. The external interrupt source must hold the input active until the interrupt request is recognized. It must then deactivate the interrupt request before execution of the ISR completes or another interrupt request will be generated.

The interrupt-pending flags for these interrupts are in the Port 1 Interrupt Flag Register shown in Fig 13.10.

Table 9.4. Interrupt Summary

Interrupt Source	Interrupt Vector	Priority Order	Interrupt-Pending Flag	Enable
Reset	0x0000	Top	None	Always enabled
External Interrupt 0 (/INT0)	0x0003	0	IE0 (TCON.1)	EX0 (IE.0)
Timer 0 Overflow	0x000B	1	TF0 (TCON.5)	ET0 (IE.1)
External Interrupt 1 (/INT1)	0x0013	2	IE1 (TCON.3)	EX1 (IE.2)
Timer 1 Overflow	0x001B	3	TF1 (TCON.7)	ET1 (IE.3)
Serial Port (UART)	0x0023	4	RI (SCON.0) TI (SCON.1)	ES (IE.4)
Timer 2 Overflow (or EXF2)	0x002B	5	TF2 (T2CON.7)	ET2 (IE.5)
Serial Peripheral Interface	0x0033	6	SPIF (SPI0STA.7)	ESPI0 (EIE1.0)
ADC0 Window Comparison	0x0043	8	ADWINT (ADC0CN.2)	EWADC0 (EIE1.2)
Comparator 0 Falling Edge	0x0053	10	CP0FIF (CPT0CN.4)	ECP0F (EIE1.4)
Comparator 0 Rising Edge	0x005B	11	CP0RIF (CPT0CN.3)	ECP0R (EIE1.5)
Comparator 1 Falling Edge	0x0063	12	CP1FIF (CPT1CN.4)	ECP1F (EIE1.6)
Comparator 1 Rising Edge	0x006B	13	CP1RIF (CPT1CN.3)	ECP1R (EIE1.7)
ADC0 End of Conversion	0x007B	15	ADCINT (ADC0CN.5)	EADC0 (EIE2.1)
External Interrupt 4	0x0083	16	IE4 (PRT1IF.4)	EX4 (EIE2.2)
External Interrupt 5	0x008B	17	IE5 (PRT1IF.5)	EX5 (EIE2.3)
External Interrupt 6	0x0093	18	IE6 (PRT1IF.6)	EX6 (EIE2.4)
External Interrupt 7	0x009B	19	IE7 (PRT1IF.7)	EX7 (EIE2.5)
Unused Interrupt Location	0x00A3	20	None	Reserved (EIE2.6)
External Crystal OSC Ready	0x00AB	21	XTLVLD (OSCXCN.7)	EXVLD (EIE2.7)

9.4.3. Interrupt Priorities

Each interrupt source can be individually programmed to one of two priority levels: low or high. A low priority interrupt service routine can be preempted by a high priority interrupt. A high priority interrupt cannot be preempted. Each interrupt has an associated interrupt priority bit in an SFR (IP-EIP2) used to configure its priority level. Low priority is the default. If two interrupts are recognized simultaneously, the interrupt with the higher priority is serviced first. If both interrupts have the same priority level, a fixed priority order is used to arbitrate.

9.4.4. Interrupt Latency

Interrupt response time depends on the state of the CPU when the interrupt occurs. Pending interrupts are sampled and priority decoded each system clock cycle. Therefore, the fastest possible response time is 5 system clock cycles: 1 clock cycle to detect the interrupt and 4 clock cycles to complete the LCALL to the ISR. If an interrupt is pending when a RETI is executed, a single instruction is executed before an LCALL is made to service the pending interrupt. Therefore, the maximum response time for an interrupt (when no other interrupt is currently being serviced or the new interrupt is of greater priority) occurs when the CPU is performing an RETI instruction followed by a DIV as the next instruction. In this case, the response time is 18 system clock cycles: 1 clock cycle to detect the interrupt, 5 clock cycles to execute the RETI, 8 clock cycles to complete the DIV instruction and 4 clock cycles to execute the LCALL to the ISR. **NOTE**: If a FLASH write or erase is performed, the MCU is stalled during the operation and interrupts will not be serviced until the operation is complete. If the CPU is executing an ISR for an interrupt with equal or higher priority, the new interrupt will not be serviced until the current ISR completes, including the RETI and following instruction.

9.4.5. Interrupt Register Descriptions

The SFRs used to enable the interrupt sources and set their priority level are described below. Refer to the datasheet section associated with a particular on-chip peripheral for information regarding valid interrupt conditions for the peripheral and the behavior of its interrupt-pending flag(s).

Figure 9.9. IE: Interrupt Enable

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
EA	-	ET2	ES	ET1	EX1	ET0	EX0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
							(bit addressable)	0xA8

Bit7: EA: Enable All Interrupts.
This bit globally enables/disables all interrupts. It overrides the individual interrupt mask settings.
0: Disable all interrupt sources.
1: Enable each interrupt according to its individual mask setting.

Bit6: UNUSED. Read = 0, Write = don't care.

Bit5: ET2: Enable Timer 2 Interrupt.
This bit sets the masking of the Timer 2 interrupt.
0: Disable all Timer 2 interrupts.
1: Enable interrupt requests generated by the TF2 flag (T2CON.7)

Bit4: ES: Enable Serial Port (UART) Interrupt.
This bit sets the masking of the Serial Port (UART) interrupt.
0: Disable all UART interrupts.
1: Enable interrupt requests generated by the R1 flag (SCON.0) or T1 flag (SCON.1).

Bit3: ET1: Enable Timer 1 Interrupt.
This bit sets the masking of the Timer 1 interrupt.
0: Disable all Timer 1 interrupts.
1: Enable interrupt requests generated by the TF1 flag (TCON.7).

Bit2: EX1: Enable External Interrupt 1.
This bit sets the masking of external interrupt 1.
0: Disable external interrupt 1.
1: Enable interrupt requests generated by the /INT1 pin.

Bit1: ET0: Enable Timer 0 Interrupt.
This bit sets the masking of the Timer 0 interrupt.
0: Disable all Timer 0 interrupts.
1: Enable interrupt requests generated by the TF0 flag (TCON.5).

Bit0: EX0: Enable External Interrupt 0.
This bit sets the masking of external interrupt 0.
0: Disable external interrupt 0.
1: Enable interrupt requests generated by the /INT0 pin.

Figure 9.10. 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 = 00b, Write = don't care.

Bit5: PT2 Timer 2 Interrupt Priority Control.
This bit sets the priority of the Timer 2 interrupts.
0: Timer 2 interrupt priority determined by default priority order.
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 interrupt priority determined by default priority order.
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 interrupt priority determined by default priority order.
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 priority determined by default priority order.
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 priority determined by default priority order.
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 priority determined by default priority order.
1: External Interrupt 0 set to high priority level.

Figure 9.11. EIE1: Extended Interrupt Enable 1

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
ECP1R	ECP1F	ECP0R	ECP0F	-	EWADC0	-	ESPI0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xE6

Bit7: ECP1R: Enable Comparator 1 (CP1) Rising Edge Interrupt.
This bit sets the masking of the CP1 interrupt.
0: Disable CP1 Rising Edge interrupt.
1: Enable interrupt requests generated by the CP1RIF flag (CPT1CN.3).

Bit6: ECP1F: Enable Comparator 1 (CP1) Falling Edge Interrupt.
This bit sets the masking of the CP1 interrupt.
0: Disable CP1 Falling Edge interrupt.
1: Enable interrupt requests generated by the CP1FIF flag (CPT1CN.4).

Bit5: ECP0R: Enable Comparator 0 (CP0) Rising Edge Interrupt.
This bit sets the masking of the CP0 interrupt.
0: Disable CP0 Rising Edge interrupt.
1: Enable interrupt requests generated by the CP0RIF flag (CPT0CN.3).

Bit4: ECP0F: Enable Comparator 0 (CP0) Falling Edge Interrupt.
This bit sets the masking of the CP0 interrupt.
0: Disable CP0 Falling Edge interrupt.
1: Enable interrupt requests generated by the CP0FIF flag (CPT0CN.4).

Bit3: Reserved. Read = 0, Write = don’t care.

Bit2: EWADC0: Enable Window Comparison ADC0 Interrupt.
This bit sets the masking of ADC0 window compare interrupt.
0: Disable ADC0 Window Comparison Interrupt.
1: Enable Interrupt requests generated by ADC0 Window Comparisons.

Bit1: Reserved. Read = 0, Write = don’t care.

Bit0: ESPI0: Enable Serial Peripheral Interface 0 Interrupt.
This bit sets the masking of SPI0 interrupt.
0: Disable all SPI0 interrupts.
1: Enable Interrupt requests generated by the SPIF flag (SPI0CN.7).

Figure 9.12. EIE2: Extended Interrupt Enable 2

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
EXVLD	-	EX7	EX6	EX5	EX4	EADC0	-	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xE7

Bit7: EXVLD: Enable External Clock Source Valid (XTLVLD) Interrupt.
This bit sets the masking of the XTLVLD interrupt.
0: Disable all XTLVLD interrupts.
1: Enable interrupt requests generated by the XTLVLD flag (OSCXCN.7)

Bit6: Reserved. Must write 0. Reads 0.

Bit5: EX7: Enable External Interrupt 7.
This bit sets the masking of External Interrupt 7.
0: Disable External Interrupt 7.
1: Enable interrupt requests generated by the External Interrupt 7 input pin.

Bit4: EX6: Enable External Interrupt 6.
This bit sets the masking of External Interrupt 6.
0: Disable External Interrupt 6.
1: Enable interrupt requests generated by the External Interrupt 6 input pin.

Bit3: EX5: Enable External Interrupt 5.
This bit sets the masking of External Interrupt 5.
0: Disable External Interrupt 5.
1: Enable interrupt requests generated by the External Interrupt 5 input pin.

Bit2: EX4: Enable External Interrupt 4.
This bit sets the masking of External Interrupt 4.
0: Disable External Interrupt 4.
1: Enable interrupt requests generated by the External Interrupt 4 input pin.

Bit1: EADC0: Enable ADC0 End of Conversion Interrupt.
This bit sets the masking of the ADC0 End of Conversion Interrupt.
0: Disable ADC0 Conversion Interrupt.
1: Enable interrupt requests generated by the ADC0 Conversion Interrupt.

Bit0: Reserved. Read = 0, Write = don't care.

Figure 9.13. 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	-	PWADC0	-	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: Reserved. Read = 0, Write = don't care.

Bit2: PWADC0: Analog-to-Digital Converter 0 window compare (ADC0) Interrupt Priority Control.
This bit sets the priority of the ADC0 window compare interrupt.
0: ADC0 window compare interrupt set to low priority level.
1: ADC0 window compare interrupt set to high priority level.

Bit1: UNUSED. Read = 0, Write = don't care.

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.

Figure 9.14. 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	-	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: Reserved. Read = 0, Write = don’t care.

9.5. 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 9.15 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. Turning off the active oscillator saves even more power, but requires a reset to restart the MCU.

9.5.1. 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.

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 12.7 Watchdog Timer for more information on the use and configuration of the WDT.

9.5.2. 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.

If enabled, the Missing Clock Detector will cause an internal reset and thereby terminate the Stop mode. The Missing Clock Detector should be disabled if the CPU is to be put to sleep for longer than the MCD timeout of 100µsec.

Figure 9.15. PCON: Power Control Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
SMOD	GF4	GF3	GF2	GF1	GF0	STOP	IDLE	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0x87

Bit7: SMOD: Serial Port Baud Rate Doubler Enable.
0: Serial Port baud rate is that defined by Serial Port Mode in SCON.
1: Serial Port baud rate is double that defined by Serial Port Mode in SCON.

Bits6-2: GF4-GF0: General Purpose Flags 4-0.
These are general purpose flags for use under software control.

Bit1: STOP: Stop Mode Select.
Setting this bit will place the CIP-51 in Stop mode. This bit will always be read as 0.
1: Goes into power down mode. (Turns off oscillator).

Bit0: IDLE: Idle Mode Select.
Setting this bit will place the CIP-51 in Idle mode. This bit will always be read as 0.
1: Goes into idle mode. (Shuts off clock to CPU, but clock to Timers, Interrupts, Serial Ports, and Analog Peripherals are still active.)

10. FLASH MEMORY

This MCU includes 8k + 128 bytes of on-chip, reprogrammable Flash memory for program code and non-volatile data storage. The Flash memory can be programmed in-system, a single byte at a time, through the JTAG interface or by software using the MOVX instruction. Once cleared to 0, a Flash bit must be erased to set it back to 1. The bytes would typically be erased (set to 0xFF) before being reprogrammed. The write and erase operations are automatically timed by hardware for proper execution. Data polling to determine the end of the write/erase operation is not required. The Flash memory is designed to withstand at least 10,000 write/erase cycles. Refer to Table 10.1 for the electrical characteristics of the Flash memory.

10.1. Programming The Flash Memory

The simplest means of programming the Flash memory is through the JTAG interface using programming tools provided by Cygnal or a third party vendor. This is the only means for programming a non-initialized device. For details on the JTAG commands to program Flash memory, see Section 18.1.

The Flash memory can be programmed by software using the MOVX instruction with the address and data byte to be programmed provided as normal operands. Before writing to Flash memory using MOVX, flash write operations must be enabled by setting the PSWE Program Store Write Enable bit (PSCTL.0) to logic 1. Writing to Flash remains enabled until the PSWE bit is cleared by software.

Writes to Flash memory can clear bits but cannot set them. Only an erase operation can set bits in Flash. Therefore, the byte location to be programmed must be erased before a new value can be written. The 8kbyte Flash memory is organized in 512-byte sectors. The erase operation applies to an entire sector (setting all bytes in the sector to 0xFF). Setting the PSEE Program Store Erase Enable bit (PSCTL.1) and PSWE Program Store Write Enable bit (PSCTL.0) to logic 1 and then using the MOVX command to write a data byte to any byte location within the sector will erase an entire 512-byte sector. The data byte written can be of any value because it is not actually written to the Flash. Flash erasure remains enabled until the PSEE bit is cleared by software. The following sequence illustrates the algorithm for programming the Flash memory by software:

1. Enable Flash Memory write/erase in FLSCL Register using FLASCL bits.
2. Set PSEE (PSCTL.1) to enable Flash sector erase.
3. Set PSWE (PSCTL.0) to enable Flash writes.
4. Use MOVX to write a data byte to any location within the 512-byte sector to be erased.
5. Clear PSEE to disable Flash sector erase.
6. Use MOVX to write a data byte to the desired byte location within the erased 512-byte sector. Repeat until finished. (Any number of bytes can be written from a single byte to and entire sector.)
7. Clear the PSWE bit to disable Flash writes.

Write/Erase timing is automatically controlled by hardware based on the prescaler value held in the Flash Memory Timing Prescaler register (FLSCL). The 4-bit prescaler value FLASCL determines the time interval for write/erase operations. The FLASCL value required for a given system clock is shown in Figure 10.3, along with the formula used to derive the FLASCL values. When FLASCL is set to 1111b, the write/erase operations are disabled. Note that code execution in the 8051 is stalled while the Flash is being programmed or erased.

Table 10.1. FLASH Memory Electrical Characteristics

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

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS
Endurance		10k	100k		Erase/Wr
Erase/Write Cycle Time			10		ms

10.2. Non-volatile Data Storage

The Flash memory can be used for non-volatile data storage as well as program code. This allows data such as calibration coefficients to be calculated and stored at run time. Data is written using the MOVX instruction and read using the MOVC instruction.

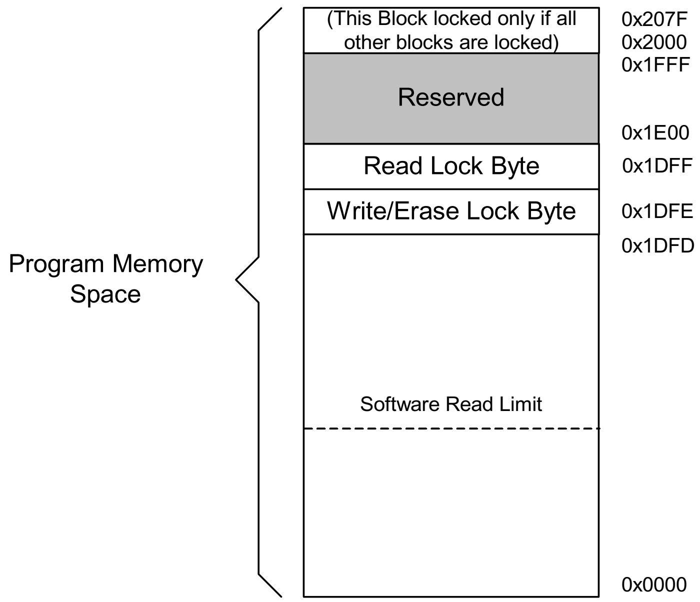
The MCU incorporates an additional 128-byte sector of Flash memory located at 0x2000 – 0x207F. This sector can be used for program code or data storage. However, its smaller sector size makes it particularly well suited as general purpose, non-volatile scratchpad memory. Even though Flash memory can be written a single byte at a time, an entire sector must be erased first. In order to change a single byte of a multi-byte data set, the data must be moved to temporary storage. Next, the sector is erased, the data set updated and the data set returned to the original sector. The 128-byte sector-size facilitates updating data without wasting program memory space by allowing the use of internal data RAM for temporary storage. (A normal 512-byte sector is too large to be stored in the 256-byte internal data memory.)

10.3. Security Options

The CIP-51 provides security options to protect the Flash memory from inadvertent modification by software as well as prevent the viewing of proprietary program code and constants. The Program Store Write Enable (PSCTL.0) and the Program Store Erase Enable (PSCTL.1) bits protect the Flash memory from accidental modification by software. These bits must be explicitly set to logic 1 before software can modify the Flash memory. Additional security features prevent proprietary program code and data constants from being read or altered across the JTAG interface or by software running on the system controller.

A set of security lock bytes stored at 0x1DFE and 0x1DFF protect the Flash program memory from being read or altered across the JTAG interface. Each bit in a security lock-byte protects one 1kbyte block of memory. Clearing a bit to logic 0 in a Read lock byte prevents the corresponding block of Flash memory from being read across the JTAG interface. Clearing a bit in the Write/Erase lock byte protects the block from JTAG erasures and/or writes. The Read lock byte is at location 0x1DFF. The Write/Erase lock byte is located at 0x1DFE. Figure 10.1 shows the location and bit definitions of the security bytes. The 512-byte sector containing the lock byte cannot be erased by software.

Figure 10.1. Flash Program Memory Security Bytes



Read and Write/Erase Security Bits.
(Bit 7 is MSB.)

Bit	Memory Block
7	0x1C00 - 0x1DFD
6	0x1800 - 0x1BFF
5	0x1400 - 0x17FF
4	0x1000 - 0x13FF
3	0x0C00 - 0x0FFF
2	0x0800 - 0x0BFF
1	0x0400 - 0x07FF
0	0x0000 - 0x03FF

FLASH Read Lock Byte

Bits7-0: Each bit locks a corresponding block of memory. (Bit 7 is MSB.)

0: Read operations are locked (disabled) for corresponding block across the JTAG interface.

1: Read operations are unlocked (enabled) for corresponding block across the JTAG interface.

FLASH Write/Erase Lock Byte

Bits7-0: Each bit locks a corresponding block of memory.

0: Write/Erase operations are locked (disabled) for corresponding block across the JTAG interface.

1: Write/Erase operations are unlocked (enabled) for corresponding block across the JTAG interface.

FLASH Access Limit Register (FLACL)

The content of this register is used as the high byte of the 16-bit software read limit address. The 16-bit read limit address value is calculated as 0xNN00 where NN is replaced by content of this register on reset. Software running at or above this address is prohibited from using the MOVX and MOVC instructions to read, write, or erase, locations below this address. Any attempts to read locations below this limit will return the value 0x00.

The lock bits can always be read and cleared to logic 0 regardless of the security setting applied to the block containing the security bytes. This allows additional blocks to be protected after the block containing the security bytes has been locked. However, the only means of removing a lock once set is to erase the entire program memory space by performing a JTAG erase operation. **NOTE: Erasing the Flash memory block containing the security bytes will automatically initiate erasure of the entire program memory space (except for the reserved area). This erasure can only be performed via the JTAG. If a non-security byte in the 0x1C00-0x1DFF page is written to in order to perform an erasure of that page, then that page *including the security bytes* will be erased.**

The Flash Access Limit security feature protects proprietary program code and data from being read by software running on the CIP-51. This feature provides support for OEMs that wish to program the MCU with proprietary value-added firmware before distribution. The value-added firmware can be protected while allowing additional code to be programmed in remaining program memory space later.

The Software Read Limit (SRL) is a 16-bit address that establishes two logical partitions in the program memory space. The first is an upper partition consisting of all the program memory locations at or above the SRL address, and the second is a lower partition consisting of all the program memory locations starting at 0x0000 up to (but excluding) the SRL address. Software in the upper partition can execute code in the lower partition, but is prohibited from reading locations in the lower partition using the MOVC instruction. (Executing a MOVC instruction from the upper partition with a source address in the lower partition will always return a data value of 0x00.) Software running in the lower partition can access locations in both the upper and lower partition without restriction.

The Value-added firmware should be placed in the lower partition. On reset, control is passed to the value-added firmware via the reset vector. Once the value-added firmware completes its initial execution, it branches to a predetermined location in the upper partition. If entry points are published, software running in the upper partition may execute program code in the lower partition, but it cannot read the contents of the lower partition. Parameters may be passed to the program code running in the lower partition either through the typical method of placing them on the stack or in registers before the call or by placing them in prescribed memory locations in the upper partition.

The SRL address is specified using the contents of the Flash Access Register. The 16-bit SRL address is calculated as 0xNN00, where NN is the contents of the SRL Security Register. Thus, the SRL can be located on 256-byte boundaries anywhere in program memory space. However, the 512-byte erase sector size essentially requires that a 512 boundary be used. The contents of a non-initialized SRL security byte is 0x00, thereby setting the SRL address to 0x0000 and allowing read access to all locations in program memory space by default.

Figure 10.2. PSCTL: Program Store RW Control

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
-	-	-	-	-	-	PSEE	PSWE	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0x8F

Bits7-2: UNUSED. Read = 000000b, Write = don't care.

Bit1: PSEE: Program Store Erase Enable.
Setting this bit allows an entire page of the Flash program memory to be erased (provided the PSWE bit is set to '1'). After setting this bit, 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.

Bit0: PSWE: Program Store Write Enable.
Setting this bit allows writing a byte of data to the Flash program memory using the MOVX instruction. The location must be erased before writing data.
0: Write to Flash program memory disabled.
1: Write to Flash program memory enabled.

Figure 10.3. FLSCL: Flash Memory Timing Prescaler

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
FOSE	FRAE	-	-	FLASCL				10001111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xB6

Bit7: FOSE: Flash One-Shot Timer Enable
0: Flash One-shot timer disabled.
1: Flash One-shot timer enabled

Bit6: FRAE: Flash Read Always Enable
0: Flash reads per one-shot timer
1: Flash always in read mode

Bits5-4: UNUSED. Read = 00b, Write = don't care.

Bits3-0: FLASCL: Flash Memory Timing Prescaler.
This register specifies the prescaler value for a given system clock required to generate the correct timing for Flash write/erase operations. If the prescaler is set to 1111b, Flash write/erase operations are disabled.
0000: System Clock < 50kHz
0001: 50kHz ≤ System Clock < 100kHz
0010: 100kHz ≤ System Clock < 200kHz
0011: 200kHz ≤ System Clock < 400kHz
0100: 400kHz ≤ System Clock < 800kHz
0101: 800kHz ≤ System Clock < 1.6MHz
0110: 1.6MHz ≤ System Clock < 3.2MHz
0111: 3.2MHz ≤ System Clock < 6.4MHz
1000: 6.4MHz ≤ System Clock < 12.8MHz
1001: 12.8MHz ≤ System Clock < 25.6MHz
1010: 25.6MHz ≤ System Clock < 51.2MHz*
1011, 1100, 1101, 1110: Reserved Values
1111: Flash Memory Write/Erase Disabled

The prescaler value is the smallest value satisfying the following equation:
FLASCL > $\log_2$(System Clock / 50kHz)

*For test purposes. The C8051F2xx is not guaranteed to operate over 25MHz.

Figure 10.4. FLACL: Flash Access Limit

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: 0xB7

Bits 7-0: FLACL: Flash Memory Access Limit.
This register holds the high byte of the 16-bit program memory read/write/erase limit address. The entire 16-bit access limit address value is calculated as 0xNN00 where NN is replaced by contents of FLACL. A write to this register sets the Flash Access Limit. Any subsequent writes are ignored until the next reset.

11. ON-BOARD XRAM (C8051F226/236/206)

The C8051F226/F236/206 features 1024 Bytes of RAM mapped into the external data memory space. All address locations may be accessed using the external move instruction (MOVX) and the data pointer (DPTR), or using indirect MOVX addressing mode. If the MOVX instruction is used with an 8-bit operand (such as @R1), then the high byte is the External Memory Interface Control Register (EMI0CN, shown in Figure 11.1). Addressing using 8 bits will map to one of four 256-byte pages, and these pages are selected by setting the PGSEL bits in the EMI0CN register.

NOTE: The MOVX instruction is also used for write to the FLASH memory. Please see section 10 for details. The MOVX instruction will access XRAM by default.

For any of the addressing modes, the upper 6 bits of the 16-bit external data memory address word are "don't cares". As a result, the 1024-byte RAM is mapped modulo style ("wrap around") over the entire 64k of possible address values. For example, the XRAM byte at address 0x0000 is also at address 0x0400, 0x0800, 0x0C00, 0x1000, etc. This feature is useful when doing a linear memory fill, as the address pointer does not have to be reset when reaching the RAM block boundary.

Figure 11.1. EMI0CN: External Memory Interface Control

R	R	R	R	R	R	R/W	R/W	Reset Value
-	-	-	-	-	-	PGSEL1	PGSEL0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xAF

Bits7-2: Not Used –read only 000000b

Bits1-0: XRAM Page Select Bits PGSEL[1:0]

The XRAM Page Select bits provide the high byte of the 16-bit external memory address when using an 8-bit MOVX command, effectively selecting a 256-byte page of RAM. The upper 6 bits are "don't cares", so the 1k address blocks are repeated modulo over the entire data memory address space.

00:0x000 – 0x0FF
01:0x100 – 0x1FF
10:0x200 – 0x2FF
11:0x300 – 0x3FF

12. RESET SOURCES

The reset circuitry of the MCU 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.

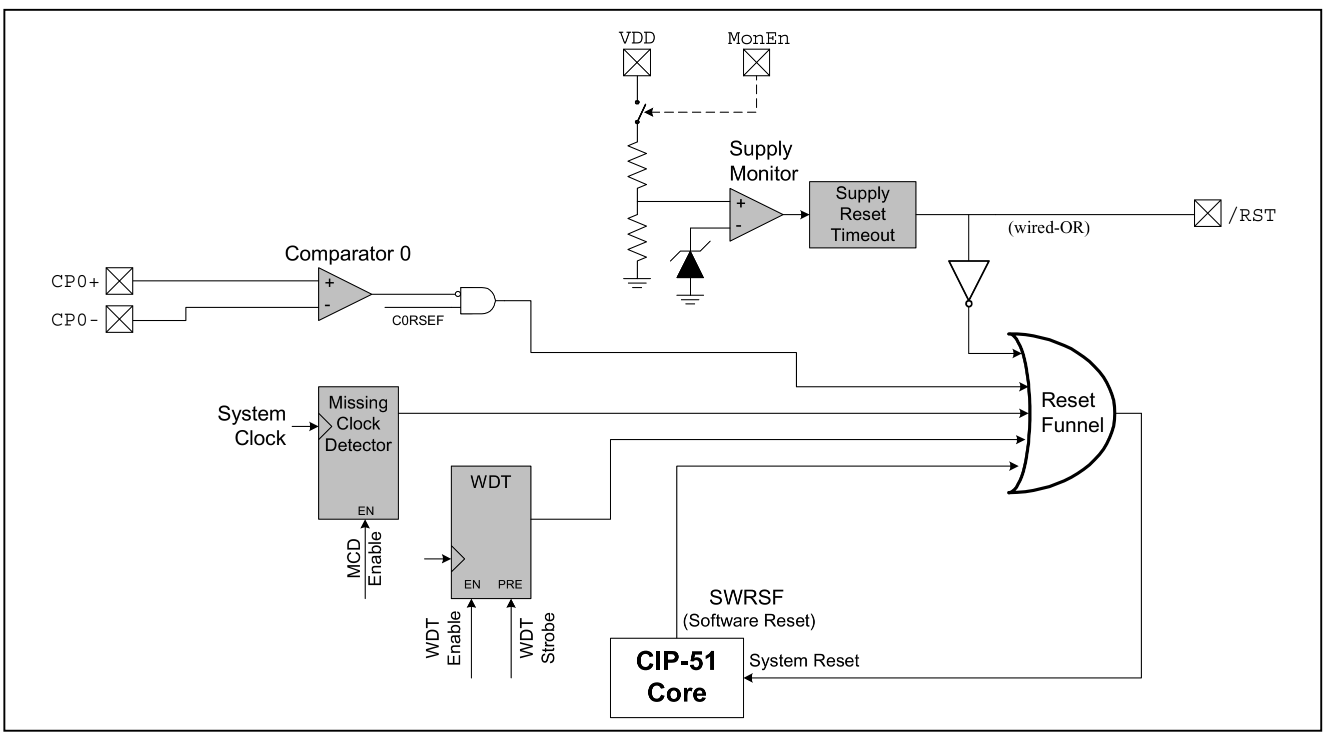
All of the SFRs are reset to predefined values. The reset values of the SFR bits are defined in the SFR detailed descriptions. The contents of internal data memory are not changed during a reset and any previously stored data is preserved. However, since the stack pointer SFR is reset, the stack is effectively 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 external I/O pins do not go high immediately, but will go high within 4 system clock cycles after entering the reset state. If the source of reset is from the VDD Monitor or writing a '1' to the PORSF bit, 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 13 for information on selecting and configuring the system clock source. The Watchdog Timer is enabled using its longest timeout interval. (Section 12.7 details the use of the Watchdog Timer.) Once the system clock source is stable, program execution begins at location 0x0000.

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

Figure 12.1. Reset Sources Diagram



12.1. Power-on Reset

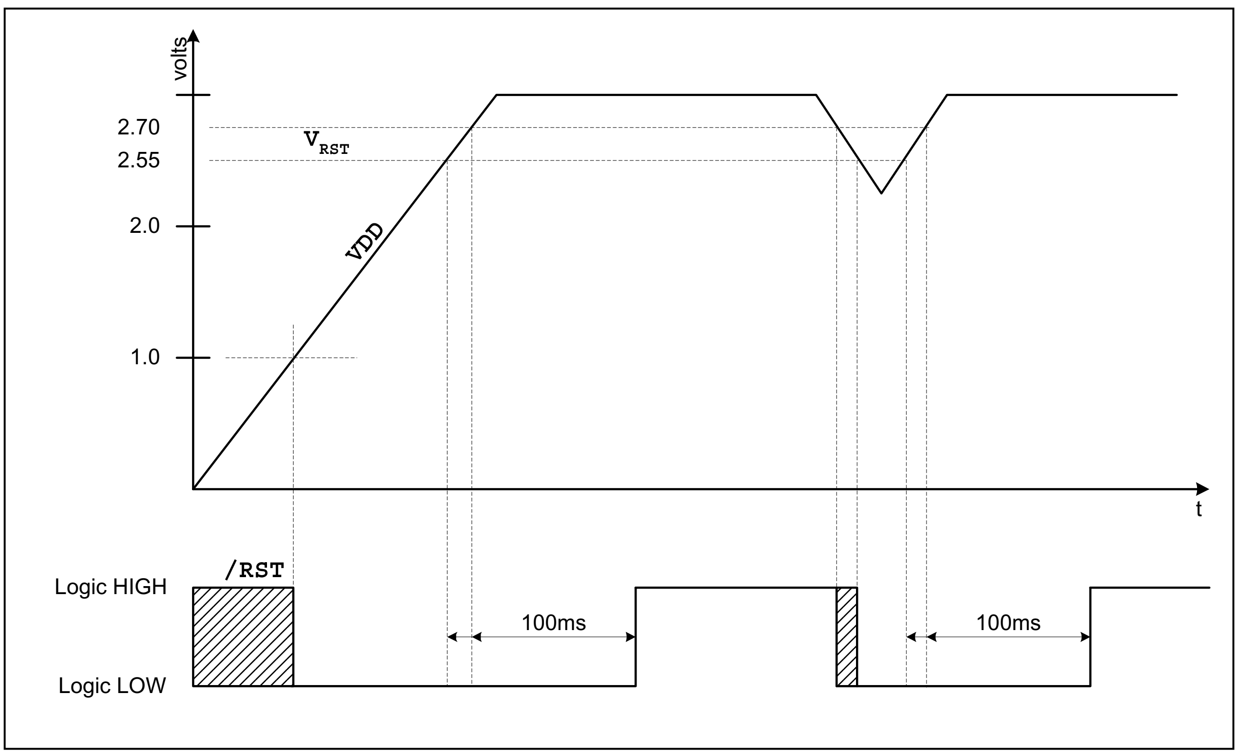
The CIP-51 incorporates a power supply monitor that holds the MCU in the reset state until VDD rises above the V_{RST} level during power-up. (See Figure 12.2 for timing diagram, and refer to Table 12.1 for the Electrical Characteristics of the power supply monitor circuit.) The /RST pin is asserted (low) until the end of the 100msec VDD Monitor timeout in order to allow the VDD supply to become stable. The VDD monitor is enabled by pulling the MONEN pin high (available only on 48-pin packages). The VDD monitor may be disabled by pulling the MONEN pin low.

On exit from a power-on reset, the PORSF flag (RSTSRC.1) is set by hardware to logic 1. All of the other reset flags in the RSTSRC Register are indeterminate. PORSF is cleared by all other resets. Since all resets cause program execution to begin at the same location (0x0000), software can read the PORSF flag to determine if a power-up was the cause of reset. The content of internal data memory should be assumed to be undefined after a power-on reset.

12.2. Software Forced Reset

Writing a 1 to the PORSF bit forces a Power-On Reset as described in Section 12.1.

Figure 12.2. VDD Monitor Timing Diagram



12.3. Power-fail Reset

When the VDD monitor is enabled the MONEN pin (not on C8051F221/F231 32 pin parts) is "pulled high",and power-down transition or power irregularity causes VDD to drop below V_{RST}, the power supply monitor will drive the /RST pin low and return the CIP-51 to the reset state (see Figure 12.2). When VDD returns to a level above V_{RST}, the CIP-51 will leave the reset state in the same manner as that for the power-on reset. Note that even though internal data memory contents are not altered by the power-fail reset, it is impossible to determine if VDD dropped below the level required for data retention. If the PORSF flag is set, the data may no longer be valid.

12.4. External Reset

The external /RST pin provides a means for external circuitry to force the CIP-51 into a reset state. Asserting an active-low signal on the /RST pin will cause the CIP-51 to enter the reset state. Although there is a weak pull-up, it may be desirable to provide an external pull-up and/or decoupling of the /RST pin to avoid erroneous noise-induced resets. The CIP-51 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 5V tolerant.

12.5. 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 100µsec, 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 13.2) enables the Missing Clock Detector.

12.6. Comparator 0 Reset

Comparator 0 can be configured as a reset input by writing a 1 to the C0RSEF flag (RSTSRC.5). Comparator 0 should be enabled using CPT0CN.7 (see Figure 8.3) 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.

12.7. 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.

12.7.1. 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 12.3.

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.

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 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.

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:

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

For a 2.0 MHz system clock, this provides an interval range of 32msec to 524msec. WDTCN.7 must be written as 0 when setting this interval. Reading WDTCN returns the programmed interval. WDTCN.[2:0] is 111b after a system reset.

Figure 12.3. WDTCN: Watchdog Timer Control Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								xxxxx111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xFF

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.

Bit4: Watchdog Status Bit (when Read)
Reading the WDTCN.[4] bit indicates the Watchdog Timer Status.
0: WDT is inactive
1: WDT is active

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.

Figure 12.4. RSTSRC: Reset Source Register

	R	R/W	R/W	R	R	R/W	R	Reset Value
-		C0RSEF	SWRSEF	WDTRSF	MCDRSF	PORSF	PINRSF	xxxxxxxx
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xEF

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

Bit7: RESERVED.
Bit6: Not Used. Read only 0b.
Bit5: C0RSEF: Comparator 0 Reset Enable and Flag
Write
0: Comparator 0 is not a reset source
1: Comparator 0 is a reset source
Read
0: Source of prior reset was not from Comparator 0
1: Source of prior reset was from Comparator 0
Bit4: SWRSF: Software Reset Force and Flag
Write
0: No Effect
1: Forces an internal reset. /RST pin is not affected.
Read
0: Prior reset source was not from write to the SWRSF bit.
1: Prior reset source was from write to the SWRSF bit.
Bit3: WDTRSF: Watchdog Timer Reset Flag (Read only)
0: Source of prior reset was not from WDT timeout.
1: Source of prior reset was from WDT timeout.
Bit2: MCDRSF: Missing Clock Detector Flag (Read only)
0: Source of prior reset was not from Missing Clock Detector timeout.
1: Source of prior reset was from Missing Clock Detector timeout.
Bit1: PORSF: Power-On Reset Force and Flag
Write
0: No effect
1: Forces a Power-On Reset. /RST is driven low.
Read
0: Source of prior reset was not from POR.
1: Source of prior reset was from POR.
Bit0: PINRSF: HW Pin Reset Flag
0: Source of prior reset was not from /RST pin.
1: Source of prior reset was from /RST pin.

Table 12.1. VDD Monitor Electrical Characteristics

-40°C to +85°C unless otherwise specified.

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS
/RST Output Low Voltage	I_{OL} = 8.5mA, VDD = 2.7 to 3.6V			0.6	V
/RST Input High Voltage		0.8 x VDD			V
/RST Input Low Voltage				0.2 x VDD	V
/RST Input Leakage Current	/RST = 0.0V			50	µA
VDD for /RST Output Valid		1.0			V
Reset Threshold (Vrst)		2.40	2.55	2.70	V
Reset Time Delay	/RST rising edge after crossing reset threshold	80	100	120	ms
Missing Clock Detector Timeout	Time from last system clock to reset generation	100	220	500	µs

13. OSCILLATOR

The MCU includes an internal oscillator and an external oscillator drive circuit, either of which can generate the system clock. The MCU boots from the internal oscillator after any reset. This internal oscillator can be enabled/disabled and its frequency can be set using the Internal Oscillator Control Register (OSCICN) as shown in Figure 13.2. The internal oscillator's electrical specifications are given in Table 13.1.

Both oscillators are disabled when the /RST pin is held low. The MCU can run from the internal oscillator permanently, or it can switch to the external oscillator if desired using CLKSL bit in the OSCICN Register. The external oscillator requires an external resonator, crystal, capacitor, or RC network connected to the XTAL1/XTAL2 pins (see Figure 13.1). The oscillator circuit must be configured for one of these sources in the OSCXCN register. An external CMOS clock can also provide the system clock by driving the XTAL1 pin. The XTAL1 and XTAL2 pins are NOT 5V tolerant.

Figure 13.1. Oscillator Diagram

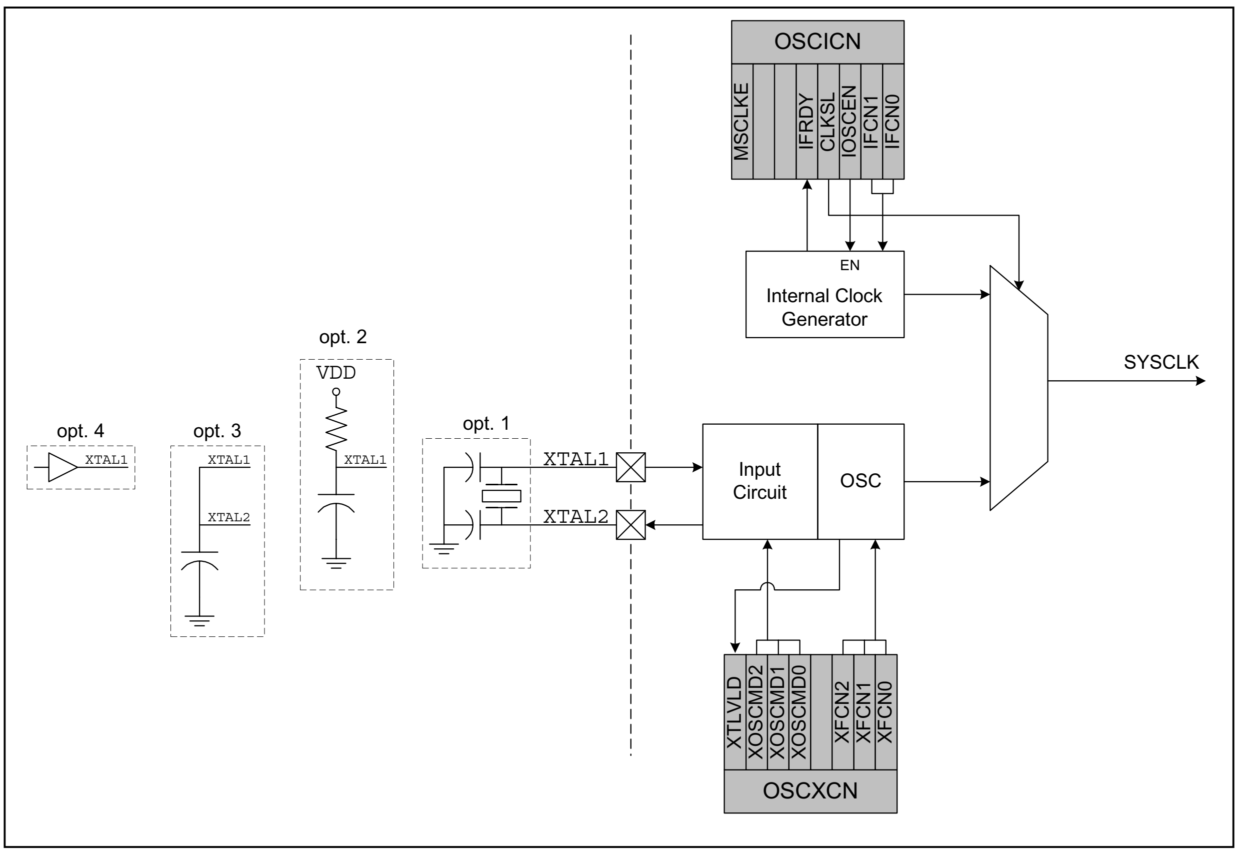


Figure 13.2. OSCICN: Internal Oscillator Control Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
MSCLKE	-	-	IFRDY	CLKSL	IOSCEN	IFCN1	IFCN0	00000100
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xB2

Bit7: MSCLKE: Missing Clock Enable Bit
0: Missing Clock Detector Disabled
1: Missing Clock Detector Enabled; triggers a reset if a missing clock is detected

Bits6-5: UNUSED. Read = 00b, Write = don't care

Bit4: IFRDY: Internal Oscillator Frequency Ready Flag
0: Internal Oscillator Frequency not running at speed specified by the IFCN bits.
1: Internal Oscillator Frequency running at speed specified by the IFCN bits.

Bit3: CLKSL: System Clock Source Select Bit
0: Uses Internal Oscillator as System Clock.
1: Uses External Oscillator as System Clock.

Bit2: IOSCEN: Internal Oscillator Enable Bit
0: Internal Oscillator Disabled
1: Internal Oscillator Enabled

Bits1-0: IFCN1-0: Internal Oscillator Frequency Control Bits
00: Internal Oscillator typical frequency is 2MHz.
01: Internal Oscillator typical frequency is 4MHz.
10: Internal Oscillator typical frequency is 8MHz.
11: Internal Oscillator typical frequency is 16MHz.

Table 13.1. Internal Oscillator Electrical Characteristics

-40°C to +85°C unless otherwise specified.

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS
Internal Oscillator Frequency	OSCICN.[1:0] = 00 OSCICN.[1:0] = 01 OSCICN.[1:0] = 10 OSCICN.[1:0] = 11	1.6 3.2 6.4 12.8	2 4 8 16	2.4 4.8 9.6 19.2	MHz
Internal Oscillator Current Consumption	OSCICN.2 = 1		200		µA
Internal Oscillator Temperature Stability			4		ppm/°C
Internal Oscillator Power Supply (VDD) Stability			6.4		%/V

Figure 13.3. OSCXCN: External Oscillator Control Register

R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
XTLVLD	XOSCMD2	XOSCMD1	XOSCMD0	-	XFCN2	XFCN1	XFCN0	00110000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xB1

Bit7: XTLVLD: Crystal Oscillator Valid Flag
(Valid only when XOSCMD = 1xx.)
0: Crystal Oscillator is unused or not yet stable
1: Crystal Oscillator is running and stable

Bits6-4: XOSCMD2-0: External Oscillator Mode Bits
00x: Off. XTAL1 pin is grounded internally.
010: System Clock from External CMOS Clock on XTAL1 pin.
011: System Clock from External CMOS Clock on XTAL1 pin divided by 2.
10x: RC/C Oscillator Mode with divide by 2 stage.
110: Crystal Oscillator Mode
111: Crystal Oscillator Mode with divide by 2 stage.

Bit3: RESERVED. Read = undefined, Write = don't care

Bits2-0: XFCN2-0: External Oscillator Frequency Control Bits
000-111: see table below

XFCN	Crystal (XOSCMD = 11x)	RC (XOSCMD = 10x)	C (XOSCMD = 10x)
000	Power Factor = 30 (10^3)	$f \leq 25$kHz	K Factor = 0.44
001	Power Factor = 90 (10^3)	25kHz $< f \leq$ 50kHz	K Factor = 1.4
010	Power Factor = 260 (10^3)	50kHz $< f \leq$ 100kHz	K Factor = 4.4
011	Power Factor = 740 (10^3)	100kHz $< f \leq$ 200kHz	K Factor = 13
100	Power Factor = 2.10 (10^6)	200kHz $< f \leq$ 400kHz	K Factor = 38
101	Power Factor = 5.80 (10^6)	400kHz $< f \leq$ 800kHz	K Factor = 100
110	Power Factor = 22.0 (10^6)	800kHz $< f \leq$ 1.6MHz	K Factor = 420
111	Power Factor = 65.0 (10^6)	1.6MHz $< f \leq$ 3.2MHz	K Factor = 1400

CRYSTAL MODE (Circuit from Figure 13.1, Option 1; XOSCMD = 11x)
Choose XFCN value to match the crystal frequency.

RC MODE (Circuit from Figure 13.1, Option 2; XOSCMD = 10x)
Choose oscillation frequency range where:
$\mathbf{f = 1.23(10^3) / (R * C)}$, where
f = frequency of oscillation in MHz
C = capacitor value in pF
R = Pull-up resistor value in kΩ

C MODE (Circuit from Figure 13.1, Option 3; XOSCMD = 10x)
Choose K Factor (KF) for the oscillation frequency desired:
$\mathbf{f = KF / (C * AV+)}$, where
f = frequency of oscillation in MHz
C = capacitor value on XTAL1, XTAL2 pins in pF
AV+ = Analog Power Supply on MCU in volts

13.1. External Crystal Example

If a crystal were used to generate the system clock for the MCU, the circuit would be as shown in Figure 13.1, Option 1. For an ECS-110.5-20-4 crystal, the resonate frequency is 11.0592MHz, the intrinsic capacitance is 7pF, and the ESR is 60Ω. The compensation capacitors should be 33pF each, and the PWB parasitic capacitance is estimated to be 2pF. The appropriate External Oscillator Frequency Control value (XFCN) from the Crystal column in the table in Figure 13.3 (OSCXCN Register) should be 111b.

When the crystal oscillator is enabled, a transient pulse may appear on XTAL2, the crystal driver output, that is sufficient to cause the XTLVLD bit in OSCXCN to go to '1' before the crystal has actually started. Introducing a blanking interval of 1ms between enabling the crystal oscillator and checking the XTLVLD bit will prevent a premature switch to the external oscillator. The recommend procedure is:
1. Enable the external oscillator
2. Wait 1 ms
3. Poll for XTLVLD '0' ==> '1'
4. Switch to the external oscillator

Switching to the external oscillator before the crystal oscillator has stabilized could result in unpredictable behavior.

NOTE: Crystal oscillator circuits are quite sensitive to PCB layout. The crystal should be placed as close as possible to the XTAL pins on the device, keeping the traces as short as possible and shielded with ground plane from any other traces which could introduce noise or interference.

13.2. External RC Example

If an external RC network were used to generate the system clock for the MCU, the circuit would be as shown in Figure 13.1, Option 2. The capacitor must be no greater than 100pF, but using a very small capacitor will increase the frequency drift due to the PWB parasitic capacitance. To determine the required External Oscillator Frequency Control value (XFCN) in the OSCXCN Register, first select the RC network value to produce the desired frequency of oscillation. If the frequency desired is 100kHz, let R = 246kΩ and C = 50pF:

$$f = 1.23(10^3)/RC = 1.23(10^3) / [246 * 50] = 0.1MHz = 100kHz$$

$$XFCN \geq \log_2(f/25kHz)$$
$$XFCN \geq \log_2(100kHz/25kHz) = \log_2(4)$$
$$XFCN \geq 2, \text{ or code } 010$$

13.3. External Capacitor Example

If an external capacitor were used to generate the system clock for the MCU, the circuit would be as shown in Figure 13.1, Option 3. The capacitor must be no greater than 100pF, but using a very small capacitor will increase the frequency drift due to the PWB parasitic capacitance. To determine the required External Oscillator Frequency Control value (XFCN) in the OSCXCN Register, select the capacitor to be used and find the frequency of oscillation from the equations below. Assume VDD = 3.0V and C = 50pF:

$$f = KF / (C * VDD) = KF / (50 * 3)$$
$$f = KF / 150$$

If a frequency of roughly 90kHz is desired, select the K Factor from the table in Figure 13.3 as KF = 13:

$$f = 13 / 150 = 0.087MHz, \text{ or } 87kHz$$

Therefore, the XFCN value to use in this example is 011.

14. PORT INPUT/OUTPUT

Description

The C8051F221/231 have three I/O Ports: Port0, Port1, and Port2. The C8051F206, C8051F220/6 and C8051F230/6 have four I/O Ports: Port0, Port1, Port2, and Port3. A wide array of digital resources can be assigned to these ports by the simple configuration of the port's corresponding multiplexer (MUX). Please see Figure 8.1. Additionally, all external port pins are available as analog input.

14.1. Port I/O Initialization

Port I/O initialization is straightforward. Registers PRT0MX, PRT1MX and PRT2MX must be loaded with the appropriate values to select the digital I/O functions required by the design. The output driver characteristics of the I/O pins are defined using the Port Configuration Registers PRT0CF, PRT1CF, PRT2CF and PRT3CF. Each Port Output driver can be configured as either Open Drain or Push-Pull. This is required even for the digital resources selected in the PRTnMX registers, and is not automatic.

Any or all pins may be configured as digital I/O or as analog input. The default mode is digital I/O. The P0MODE, P1MODE, P2MODE, and P3MODE special function registers are used to configure the port pins as digital or analog as defined in this section.

The final step is initializing the individual resources selected using the appropriate setup registers. Initialization procedures for the various digital resources may be found in the detailed explanation of each available function. The reset state of each register is shown in the figures that describe each individual register.

NOTE: The input mode of pins configured for use with Timer 0, 1, or 2 must be manually configured.

1. The output mode of all ports pins must be configured regardless of whether the port pin is either standard general-purpose I/O or controlled by a digital peripheral.
2. For all pins used as Timer inputs (P0.4/T0, P0.5/T1, P0.6/T2, and P0.7/T2EX), the output mode must be "open-drain" (which is the reset state), and "1" must be written to the associated port pin to prevent possible contention for the port pin that could result in an overcurrent condition. For example, to configure a Timer0, set PRT0MX's T0E Timer0 enable bit to '1' to route Timer0 to Port Pin P0.4. Then place P0.4/T0 in open-drain configuration (which is set in PRT0CF by default), and write a '1' to P0.4 to set its output state to high impedance for use as a digital peripheral input (port pins also default to logic high state upon reset). Lastly, ensure P0MODE.4 is '1' for digital input mode. (All pins default to digital input mode upon reset.)

Figure 14.1. Port I/O Functional Block Diagram

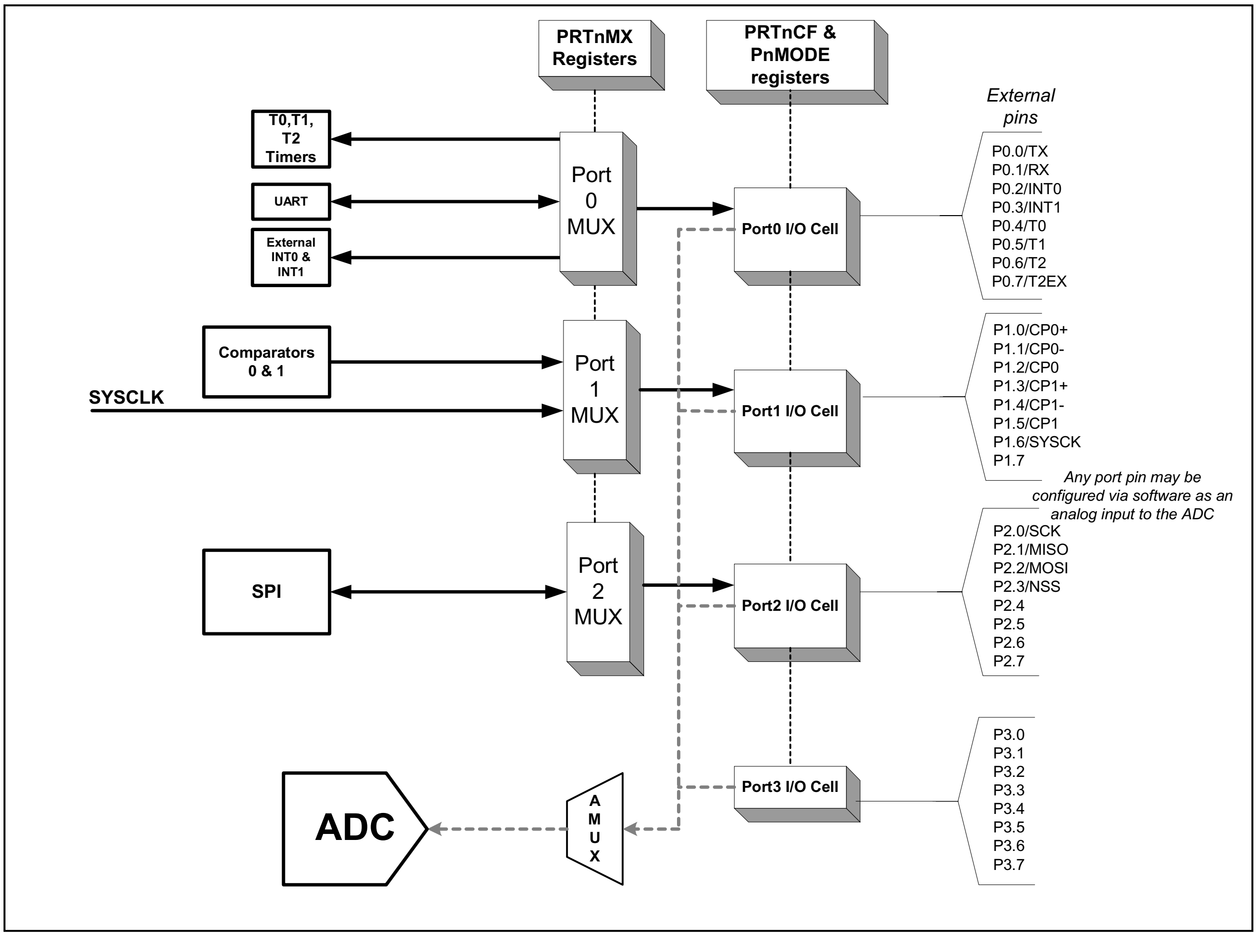


Figure 14.2. Port I/O Cell Block Diagram

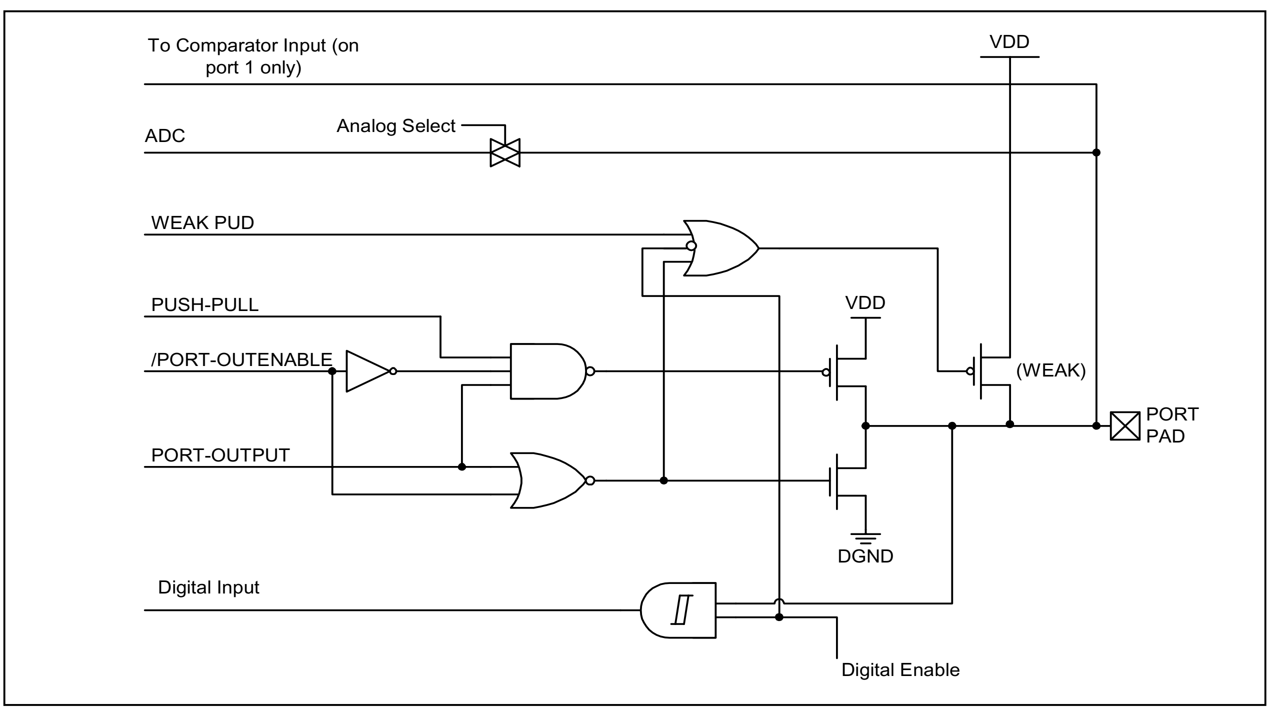


Figure 14.3. PRT0MX: Port I/O MUX Register 0

R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	Reset Value
T2EXE	T2E	T1E	T0E	INT1E	INT0E	-	UARTEN	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xE1

Bit7: T2EXE: T2EX Enable Bit
0: T2EX unavailable at Port pin.
1: T2EX routed to Port Pin.

Bit6: T2E: T2 Enable Bit
0: T2 unavailable at Port pin.
1: T2 routed to Port Pin.

Bit5: T1E: T1 Enable Bit
0: T1 unavailable at Port pin.
1: T1 routed to Port Pin.

Bit4: T0E: T0 Enable Bit
0: T0 unavailable at Port pin.
1: T0 routed to Port Pin.

Bit3: INT1E: /INT1 Enable Bit
0: /INT1 unavailable at Port pin.
1: /INT1 routed to port pin.

Bit2: INT0E: /INT0 Enable Bit
0: /INT0 unavailable at Port pin.
1: /INT0 routed to Port Pin.

Bit1: UNUSED. Read = 0, Write = don’t care.

Bit0: UARTEN: UART I/O Enable
0: UART I/O unavailable at port pins.
1: TX, RX routed to pins P0.0 and P0.1, respectively.

Figure 14.4. PRT1MX: Port I/O MUX Register 1

R	R/W	R	R	R	R	R/W	R/W	Reset Value
-	SYSCKE	-	-	-	-	CP1OEN	CP0OEN	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xE2

Bit7: UNUSED. Read = 0.
Bit6: SYSCKE: SYSCLK Output Enable Bit
0: SYSCLK unavailable at the port pin.
1: SYSCLK output routed to pin P1.6
Bits 5-2: UNUSED. Read = 0000b, Write = don't care.
Bit1: CP1OEN: Comparator 1 Output Enable bit.
0: CP1 unavailable at Port pin.
1: CP1 routed to Port Pin P1.5.
Bit0: CP0OEN: Comparator 0 Output Enable Bit
0: CP0 unavailable at port pin.
1: CP0 routed to port pin P1.2.

Figure 14.5. PRT2MX: Port I/O MUX Register 2

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
GWPUD	P3WPUD	P2WPUD	P1WPUD	P0WPUD	-	-	SPI0OEN	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xE3

Bit 7: GWPUD: Global Port I/O Weak Pull-up Disable Bit
0: Weak Pull-ups Enabled for all ports.
1: Weak Pull-ups Disabled (Bits 6-3 Don't cares)
Bit 6: P3WPUD: Port 3 Weak Pull-up Disable Bit
0: Weak Pull-ups Enabled for port 3
1: Weak Pull-ups Disabled for port 3
Bit 5: P2WPUD: Port 2 Weak Pull-up Disable Bit
0: Weak Pull-ups Enabled for port 2.
1: Weak Pull-ups Disabled for port 2
Bit 4: P1WPUD: Port 1 Weak Pull-up Disable Bit
0: Weak Pull-ups Enabled for port 1
1: Weak Pull-ups Disabled for port 1
Bit 3: P0WPUD: Port 0 Weak Pull-up Disable Bit
0: Weak Pull-ups Enabled for port 0
1: Weak Pull-ups Enabled for port 0
Bits 2-1: UNUSED. Read = 00b, Write = don't care.
Bit 0: SPI0OEN: SPI Bus I/O Enable Bit.
0: SPI I/O unavailable at port pins.
1: SCK, MISO, MOSI, NSS routed to pins P2.0, P2.1, P2.2, and P2.3 respectively.

14.2. General Purpose Port I/O

Each I/O 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 PRTnMX settings (i.e., even when the pin is assigned to another signal by the MUX, the Port Register can always still read its corresponding Port I/O pin), provided its pin is configured for digital input mode. 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 SETB, when the destination is an individual bit in a port SFR. For these instructions, the value of the register (not the pin) is read, modified, and written back to the SFR.

Figure 14.6. 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	SFR Address:
							(bit addressable)	0x80

Bits7-0: P0.[7:0]
(Write – Output appears on I/O pins per PRT0MX, PRT1MX, and PRT2MX Registers)
0: Logic Low Output.
1: Logic High Output (high impedance if corresponding PRT0CF.n bit = 0)
(Read – Regardless of PRT0MX, PRT1MX, and PRT2MX Register settings).
0: P0.n pin is logic low.
1: P0.n pin is logic high.

Figure 14.7. 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.

Figure 14.8. P0MODE: Port0 Digital/Analog Input Mode

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								11111111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xF1

Bits7-0: Port0 Digital/Analog Input Mode
0: Corresponding Port0 pin Digital Input disabled. (For analog use, i.e., ADC or comparators).
1: Corresponding Port0 pin Digital Input is enabled.

Figure 14.9. P1: Port1 Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0	11111111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0 (bit addressable)	SFR Address: 0x90

Bits7-0: P1.[7:0]
(Write – Output appears on I/O pins per PRT0MX, PRT1MX, and PRT2MX registers)
0: Logic Low Output.
1: Logic High Output (high impedance if corresponding PRT1CF.n bit = 0)
(Read – Regardless of PRT0MX, PRT1MX, and PRT2MX Register settings).
0: P1.n pin is logic low.
1: P1.n pin is logic high.

Figure 14.10. PRT1CF: Port1 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: 0xA5

Bits7-0: PRT1CF.[7:0]: Output Configuration Bits for P1.7-P1.0 (respectively)
0: Corresponding P1.n Output Mode is Open-Drain.
1: Corresponding P1.n Output Mode is Push-Pull.

Figure 14.11. P1MODE: Port1 Digital/Analog Input Mode

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								11111111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xF2

Bits7-0: Port0 Digital/Analog Output Mode
0: Corresponding Port1 pin Digital Input disabled. (For analog use, i.e., ADC or comparators).
1: Corresponding Port1 pin Digital Input is enabled.

Figure 14.12. PRT1IF: Port1 Interrupt Flag Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
IE7	IE6	IE5	IE4	-	-	-	-	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xAD

Bit7: IE7: External Interrupt 7 Pending Flag.
0: No falling edge detected on P1.7.
1: This flag is set by hardware when a falling edge on P1.7 is detected.

Bit6: IE6: External Interrupt 6 Pending Flag.
0: No falling edge detected on P1.6.
1: This flag is set by hardware when a falling edge on P1.6 is detected.

Bit5: IE5: External Interrupt 5 Pending Flag.
0: No falling edge detected on P1.5.
1: This flag is set by hardware when a falling edge on P1.5 is detected.

Bit4: IE4: External Interrupt 4 Pending Flag.
0: No falling edge detected on P1.4.
1: This flag is set by hardware when a falling edge on P1.4 is detected.

Bits3-0: UNUSED. Read = 0000b, Write = don't care.
Note: The Input Mode must be configured to Digital Mode in order for the falling edges to be detected.

Figure 14.13. P2: Port2 Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
P2.7	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0	11111111
Bit7	Bit6	Bit	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
							(bit addressable)	0xA0

Bits7-0: P2.[7:0]
(Write – Output appears on I/O pins per PRT0MX, PRT1MX, and PRT2MX registers)
0: Logic Low Output.
1: Logic High Output (high impedance if corresponding PRT2CF.n bit = 0)
(Read – Regardless of PRT0MX, PRT1MX, and PRT2MX Register settings).
0: P2.n is logic low.
1: P2.n is logic high.

Figure 14.14. PRT2CF: Port2 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:
								0xA6

Bits7-0: PRT2CF.[7:0]: Output Configuration Bits for P2.7-P2.0 (respectively)
0: Corresponding P2.n Output Mode is Open-Drain.
1: Corresponding P2.n Output Mode is Push-Pull.

Figure 14.15. P2MODE: Port2 Digital/Analog Input Mode

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								11111111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
								0xF3

Bits7-0: Port0 Digital/Analog Output Mode
0: Corresponding Port2 pin Digital Input disabled. (For analog use, i.e., ADC or comparators).
1: Corresponding Port2 pin Digital Input is enabled.

Figure 14.16. 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	SFR Address:
							(bit addressable)	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 14.17. 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.

Figure 14.18. P3MODE: Port3 Digital/Analog Input Mode*

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								11111111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
								0xF4

Bits7-0: Port0 Digital/Analog Output Mode
0: Corresponding Port3 pin Digital Input disabled. (For analog use, i.e., ADC or comparators).
1: Corresponding Port3 pin Digital Input is enabled.

*** (Available on C8051F206, C8051F220/6 and C8051F230/6)**

Table 14.1. 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 I_{OH} = -3mA, Port I/O push-pull I_{OH} = -10mA, Port I/O push-pull	VDD – 0.1 VDD – 0.7	 VDD – 0.8		V
Output Low Voltage	I_{OL} = 10uA I_{OL} = 8.5mA I_{OL} = 25mA		 1.0	0.1 0.6	V
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			3		pF

15. SERIAL PERIPHERAL INTERFACE BUS

The Serial Peripheral Interface (SPI) provides access to a four-wire, full-duplex, serial bus. SPI supports the connection of multiple slave devices to a master device on the same bus. A separate slave-select signal (NSS) is used to select a slave device and enable a data transfer between the master and the selected slave. Multiple masters on the same bus are also supported. Collision detection is provided when two or more masters attempt a data transfer at the same time. The SPI can operate as either a master or a slave. When the SPI is configured as a master, the maximum data transfer rate (bits/sec) is one-half the system clock frequency.

When the SPI is configured as a slave, the maximum data transfer rate (bits/sec) for full-duplex operation is 1/10 the system clock frequency, provided that the master issues SCK, NSS, and the serial input data synchronously with the system clock. If the master issues SCK, NSS, and the serial input data asynchronously, the maximum data transfer rate (bits/sec) must be less that 1/10 the system clock frequency. In the special case where the master only wants to transmit data to the slave and does not need to receive data from the slave (i.e. half-duplex operation), the SPI slave can receive data at a maximum data transfer rate (bits/sec) of ¼ the system clock frequency. This is provided that the master issues SCK, NSS, and the serial input data synchronously with the system clock.

Figure 15.1. SPI Block Diagram

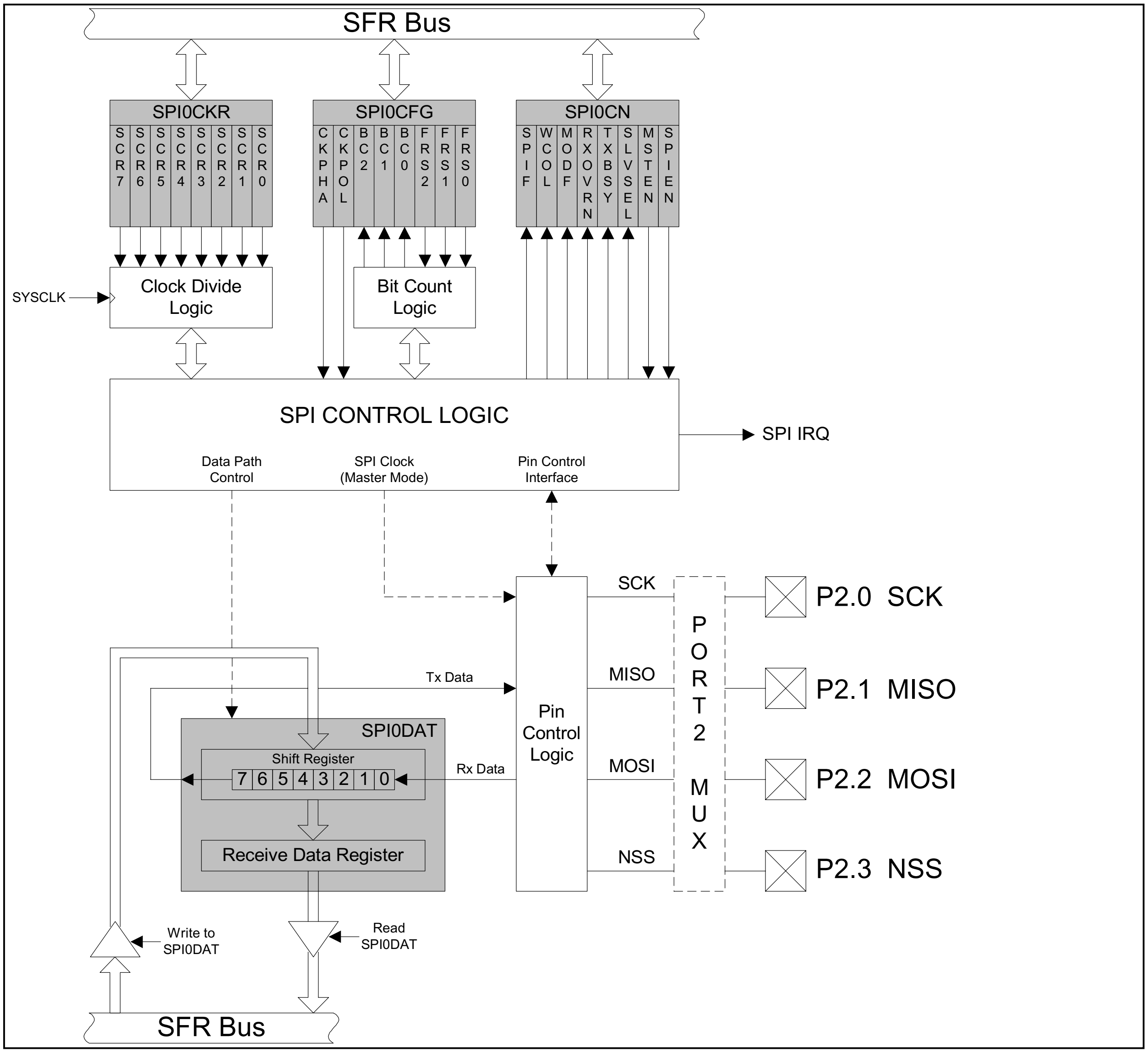
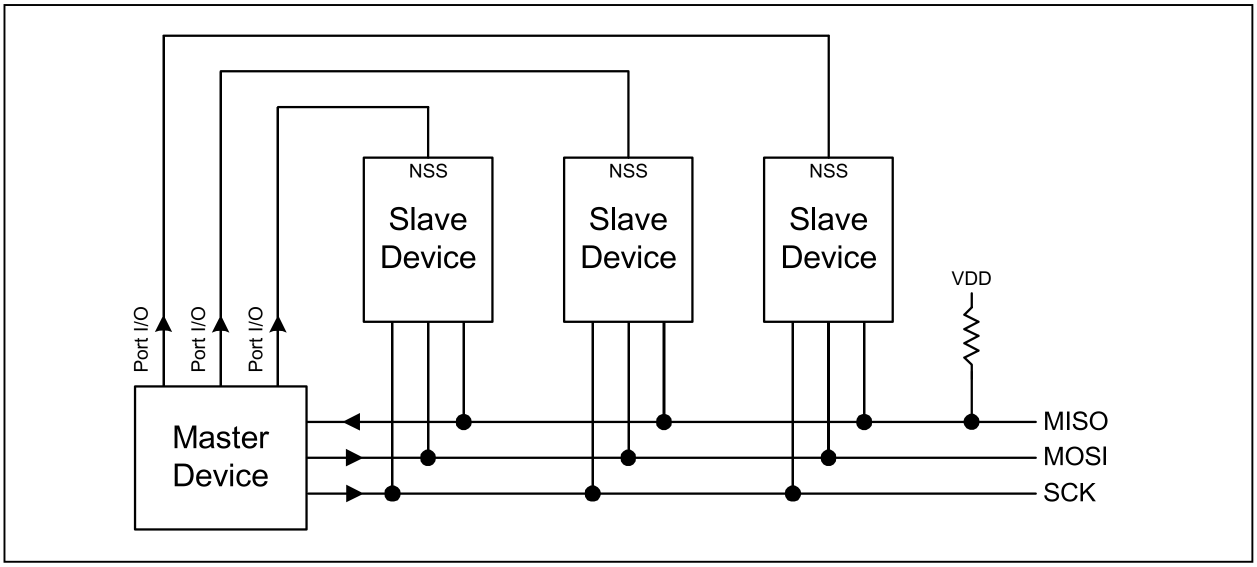


Figure 15.2. Typical SPI Interconnection



15.1. Signal Descriptions

The four signals used by the SPI (MOSI, MISO, SCK, NSS) are described below.

15.1.1. Master Out, Slave In

The master-out, slave-in (MOSI) signal is an output from a master device and an input to slave devices. It is used to serially transfer data from the master to the slave. Data is transferred most-significant bit first.

15.1.2. Master In, Slave Out

The master-in, slave-out (MISO) signal is an output from a slave device and an input to the master device. It is used to serially transfer data from the slave to the master. Data is transferred most-significant bit first. A SPI slave places the MISO pin in a high-impedance state when the slave is not selected.

15.1.3. Serial Clock

The serial clock (SCK) signal is an output from the master device and an input to slave devices. It is used to synchronize the transfer of data between the master and slave on the MOSI and MISO lines.

15.1.4. Slave Select

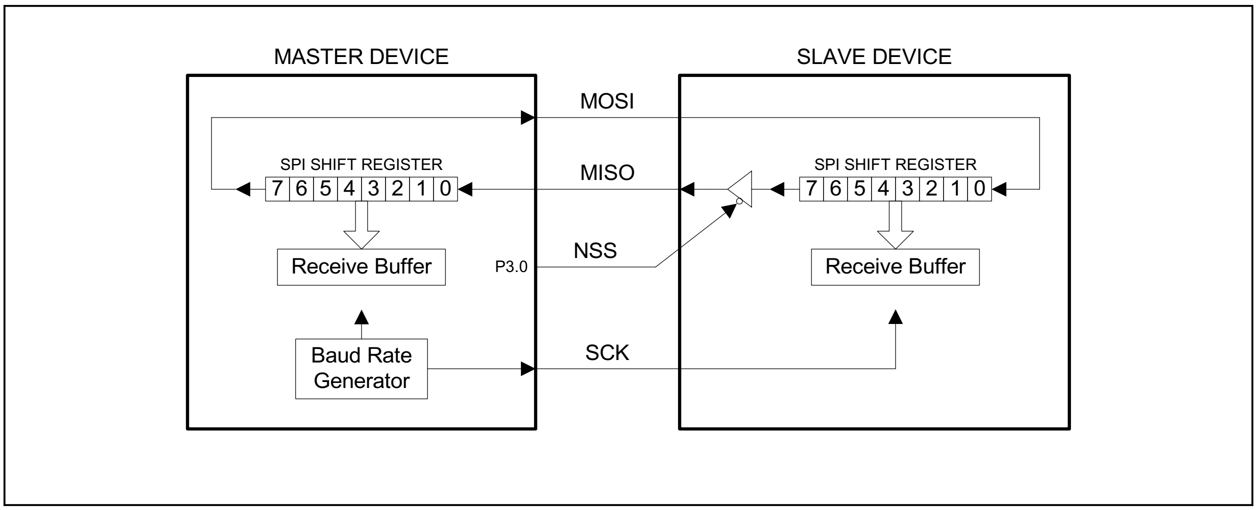
The slave select (NSS) signal is an input used to select the SPI module when in slave mode by a master, or to disable the SPI module when in master mode. When in slave mode, it is pulled low to initiate a data transfer and remains low for the duration of the transfer.

15.2. 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 15.3 illustrates the full-duplex operation of an SPI master and an addressed slave.

Figure 15.3. 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.

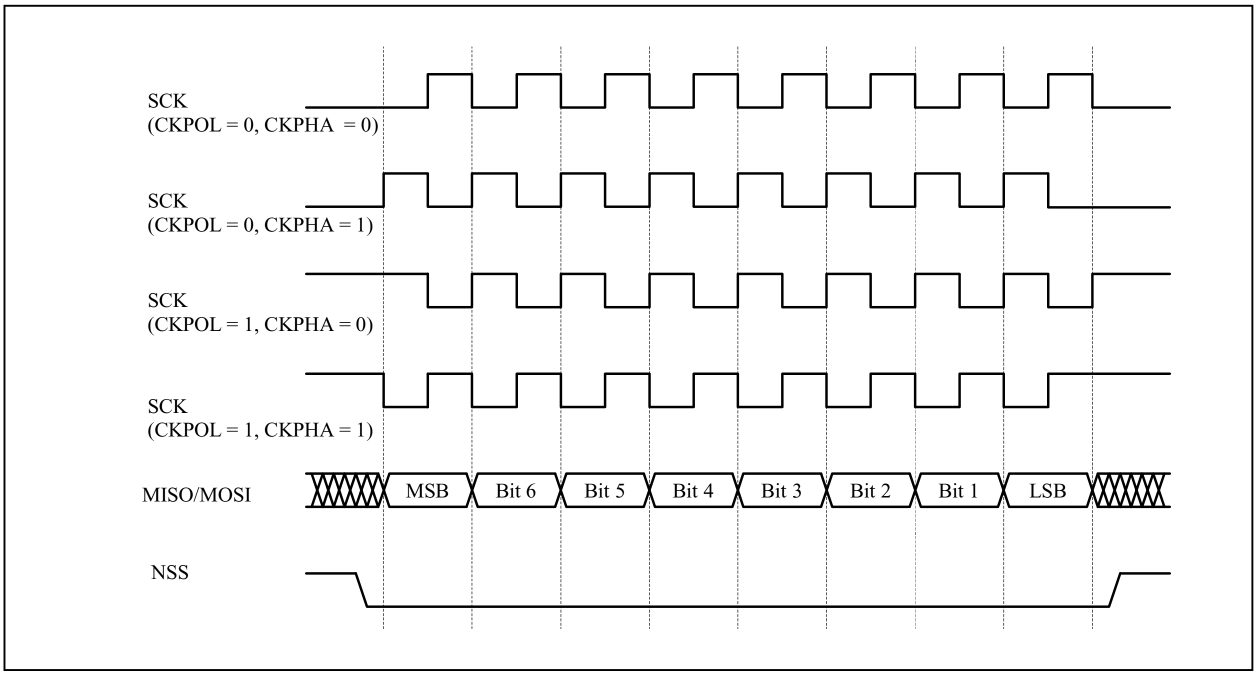
Multiple masters may reside on the same bus. A Mode Fault flag (MODF, SPI0CN.5) is set to logic 1 when the SPI is configured as a master (MSTEN = 1) and its slave select signal NSS is pulled low. When the Mode Fault flag is set, the MSTEN and SPIEN bits of the SPI control register are cleared by hardware, 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.

15.3. Serial Clock Timing

As shown in Figure 15.4, 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 15.7 controls the master mode serial clock frequency. This register is ignored when operating in slave mode.

Figure 15.4. Data/Clock Timing Diagram



15.4. 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 15.5. 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

.

Figure 15.6. SPI0CN: SPI Control Register

R/W	R/W	R/W	R/W	R	R	R/W	R/W	Reset Value
SPIF	WCOL	MODF	RXOVRN	TXBSY	SLVSEL	MSTEN	SPIEN	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xF8

Bit7: SPIF: SPI Interrupt Flag.
This bit is set to logic 1 by hardware at the end of a data transfer. If interrupts are enabled, setting this bit causes the CPU to vector to the SPI0 interrupt service routine. This bit is not automatically cleared by hardware. It must be cleared by software.

Bit6: WCOL: Write Collision Flag.
This bit is set to logic 1 by hardware (and generates a SPI interrupt) to indicate a write to the SPI data register was attempted while a data transfer was in progress. It is cleared by software.

Bit5: MODF: Mode Fault Flag.
This bit is set to logic 1 by hardware (and generates a SPI interrupt) when a master mode collision is detected (NSS is low and MSTEN = 1). This bit is not automatically cleared by hardware. It must be cleared by software.

Bit4: RXOVRN: Receive Overrun Flag.
This bit is set to logic 1 by hardware (and generates a SPI interrupt) when the receive buffer still holds unread data from a previous transfer and the last bit of the current transfer is shifted into the SPI shift register. This bit is not automatically cleared by hardware. It must be cleared by software.

Bit3: TXBSY: Transmit Busy Flag.
This bit is set to logic 1 by hardware while a master mode transfer is in progress. It is cleared by hardware at the end of the transfer.

Bit2: SLVSEL: Slave Selected Flag.
This bit is set to logic 1 whenever the NSS pin is low indicating it is enabled as a slave. It is cleared to logic 0 when NSS is high (slave disabled).

Bit1: MSTEN: Master Mode Enable.
0: Disable master mode. Operate in slave mode.
1: Enable master mode. Operate as a master.

Bit0: SPIEN: SPI Enable.
This bit enables/disables the SPI.
0: SPI disabled.
1: SPI enabled.

Figure 15.7. SPI0CKR: SPI Clock Rate Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
SCR7	SCR6	SCR5	SCR4	SCR3	SCR2	SCR1	SCR0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0x9D

Bits7-0: SCR7-SCR0: SPI Clock Rate
These bits determine the frequency of the SCK output when the SPI module is configured for master mode operation. The SCK clock frequency is a divided down version of the system clock, and is given in the following equations:

$f_{SCK} = 0.5 * f_{SYSCLK} / (SPI0CKR + 1)$, for $0 <= SPI0CKR <= 255$,

Figure 15.8. SPI0DAT: SPI Data 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: 0x9B

Bits7-0: SPI0DAT: SPI0 Transmit and Receive Data.
The SPI0DAT register is used to transmit and receive SPI data. Writing data to SPI0DAT places the data immediately into the shift register and initiates a transfer when in Master Mode. A read of SPI0DAT returns the contents of the receive buffer.

16. UART

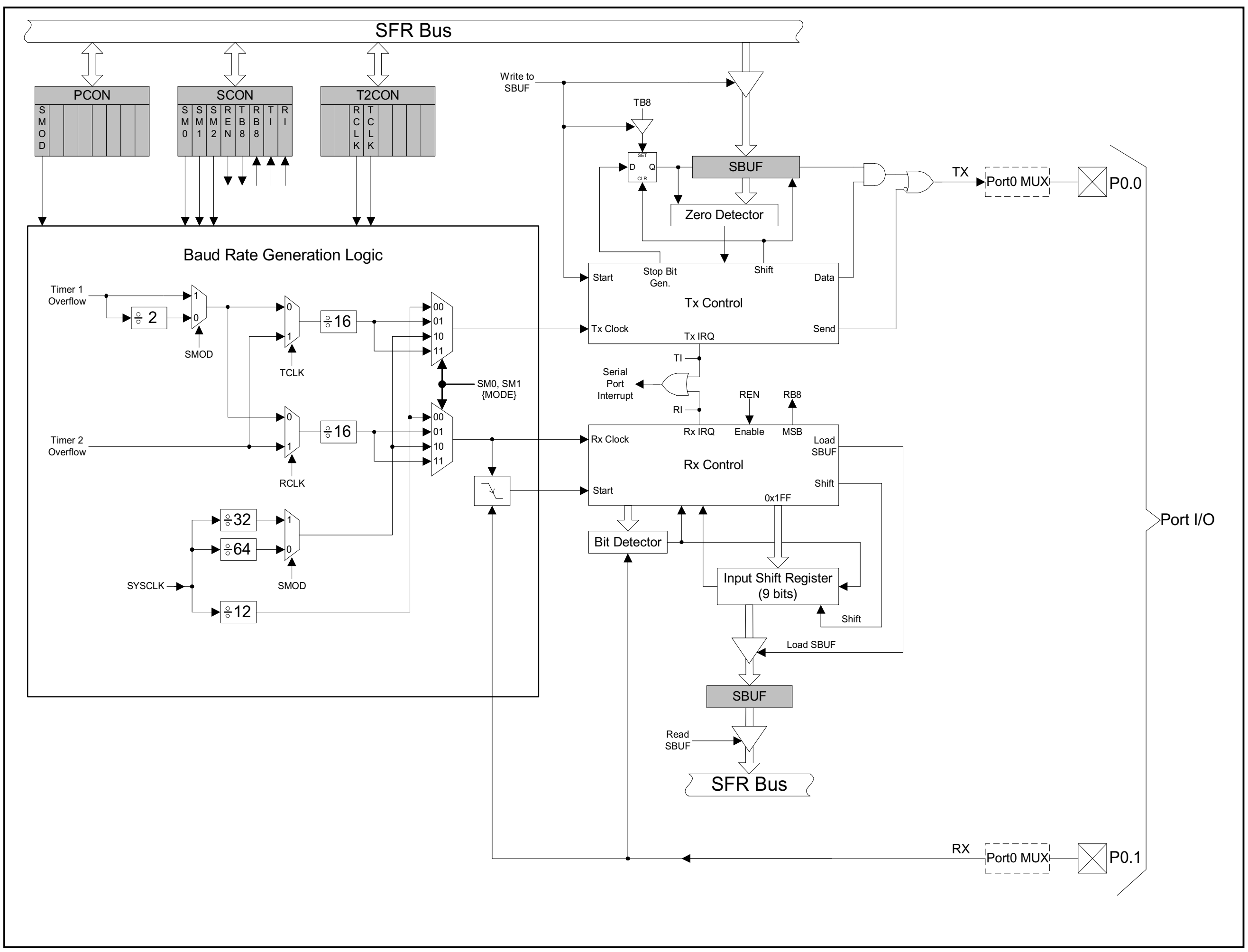
Description

The CIP-51 includes a serial port (UART) capable of asynchronous transmission. The UART can function in full duplex mode. In all modes, receive data is buffered in a holding register. This allows the UART to start reception of a second incoming data byte before software has finished reading the previous data byte.

The UART has an associated Serial Control Register (SCON) and a Serial Data Buffer (SBUF) in the SFRs. The single SBUF location provides access to both transmit and receive registers. Reads access the Receive register and writes access the Transmit register automatically.

The UART is capable of generating interrupts if enabled. The UART has two sources of interrupts: a Transmit Interrupt flag, TI (SCON.1) set when transmission of a data byte is complete, and a Receive Interrupt flag, RI (SCON.0) set when reception of a data byte is complete. The UART interrupt flags are not cleared by hardware when the CPU vectors to the interrupt service routine. They must be cleared manually by software. This allows software to determine the cause of the UART interrupt (transmit complete or receive complete).

Figure 16.1. UART Block Diagram



16.1. UART Operational Modes

The UART provides four operating modes (one synchronous and three asynchronous) selected by setting configuration bits in the SCON register. These four modes offer different baud rates and communication protocols. The four modes are summarized in Table 16.1 below. Detailed descriptions follow.

Table 16.1. UART Modes

Mode	Synchronization	Baud Clock	Data Bits	Start/Stop Bits
0	Synchronous	SYSCLK/12	8	None
1	Asynchronous	Timer 1 or Timer 2 Overflow	8	1 Start, 1 Stop
2	Asynchronous	SYSCLK/32 or SYSCLK/64	9	1 Start, 1 Stop
3	Asynchronous	Timer 1 or Timer 2 Overflow	9	1 Start, 1 Stop

16.1.1. Mode 0: Synchronous Mode

Mode 0 provides synchronous, half-duplex communication. Serial data is transmitted and received on the RX pin. The TX pin provides the shift clock for both transmit and receive. The MCU must be the master since it generates the shift clock for transmission in both directions (see the interconnect diagram in Figure 16.2).

Eight data bits are transmitted/received, LSB first (see the timing diagram in Figure 16.3). 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 eighth bit time. Data reception begins when the REN Receive Enable bit (SCON.4) is set to logic 1 and the RI Receive Interrupt Flag (SCON.0) is cleared. One cycle after the eighth bit is shifted in, the RI flag is set and reception stops until software clears the RI bit. An interrupt will occur if enabled when either TI or RI are set.

The Mode 0 baud rate is system clock frequency divided by twelve.

Figure 16.2. UART Mode 0 Interconnect

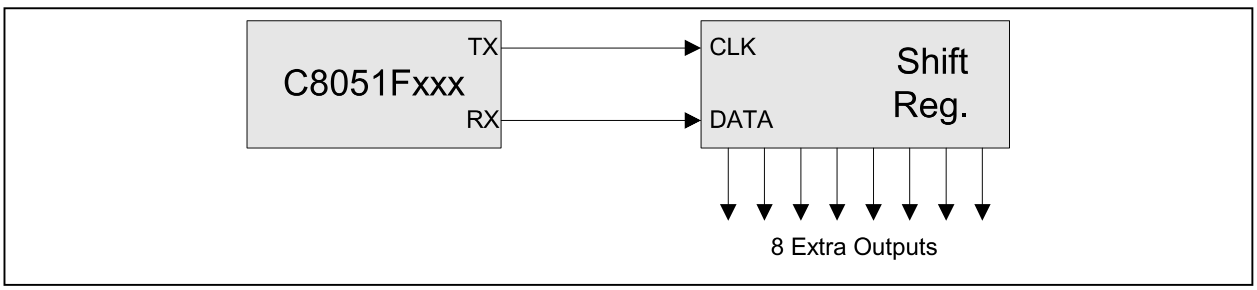
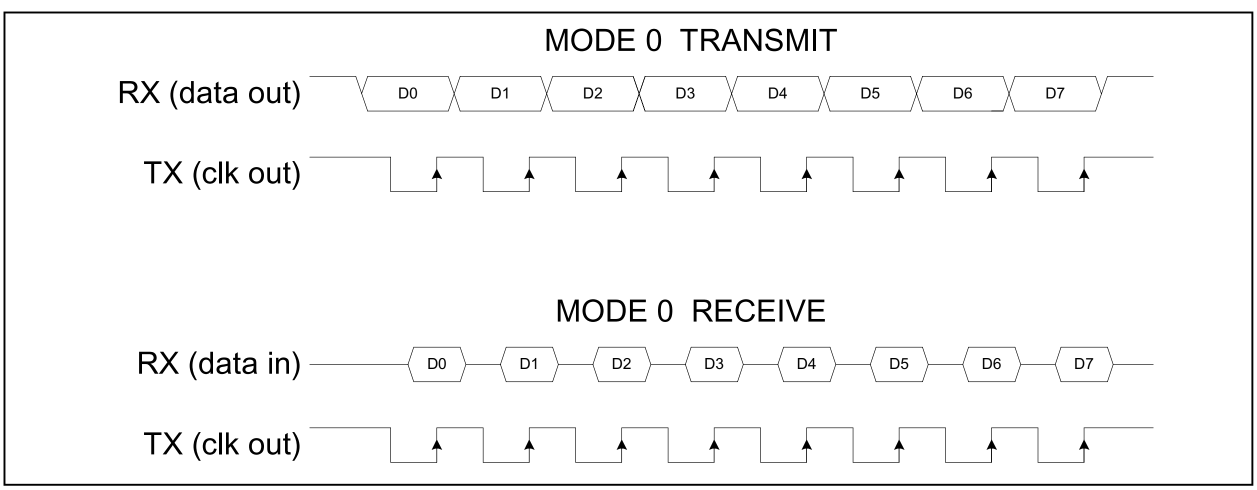


Figure 16.3. UART Mode 0 Timing Diagram



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. Data are transmitted from the TX pin and received at the RX pin. 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 is 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 are set.

The baud rate generated in Mode 1 is a function of timer overflow. The UART can use either Timer 1 or Timer 2 operating in auto-reload mode to generate the baud rate. 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 circuit. This clock is divided by 16 to generate the baud rate.

Timer 1 should be configured for *8-bit Counter/Timer with Auto-Reload* mode and its interrupt disabled when used as a baud rate generator. The combination of system clock frequency and the reload value stored in TH1 determine the baud rate as follows:

$$Mode\ 1\ Baud\ Rate = (2^{SMOD} / 32) * (SYSCLK / (12^{(T1M-1)}\ / (256 - TH1)).$$

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. Selecting the timebase used by Timer 1 allows further control of baud rate generation. Using the system clock divided by one (setting T1M in CKCON) changes the twelve in the denominator of the equation above to a one.

To use Timer 2 for baud rate generation, configure the timer *Baud Rate Generator* mode and set RCLK and/or TCLK to logic 1. Setting RCLK and/or TCLK automatically disables Timer 2 interrupts and configures Timer 2 to use the system clock divided by two as its timebase. If a different timebase is required, setting the C/T2 bit to logic 1 will allow the timebase to be derived from a clock supplied to the external input pin T2. The combination of clock frequency and the reload value stored in capture registers determine the baud rate as follows:

$$Mode\ 1\ Baud\ Rate = SYSCLK / [32 * (65536 - [RCAP2H:RCAP2L])\ ],$$

where [RCAP2H:RCAP2L] is the 16-bit value held in the capture registers.

Figure 16.4. UART Mode 1 Timing Diagram

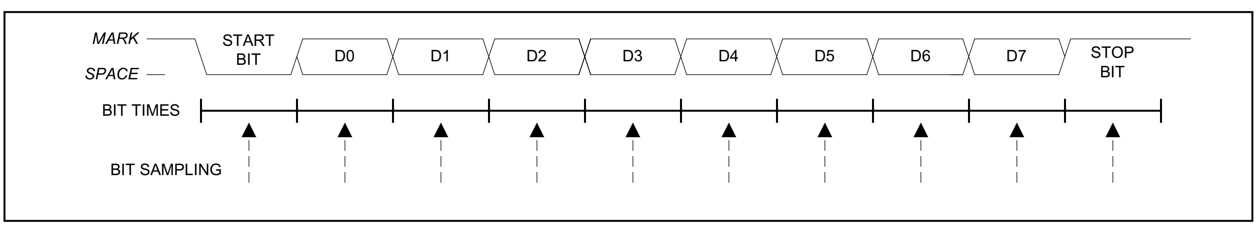
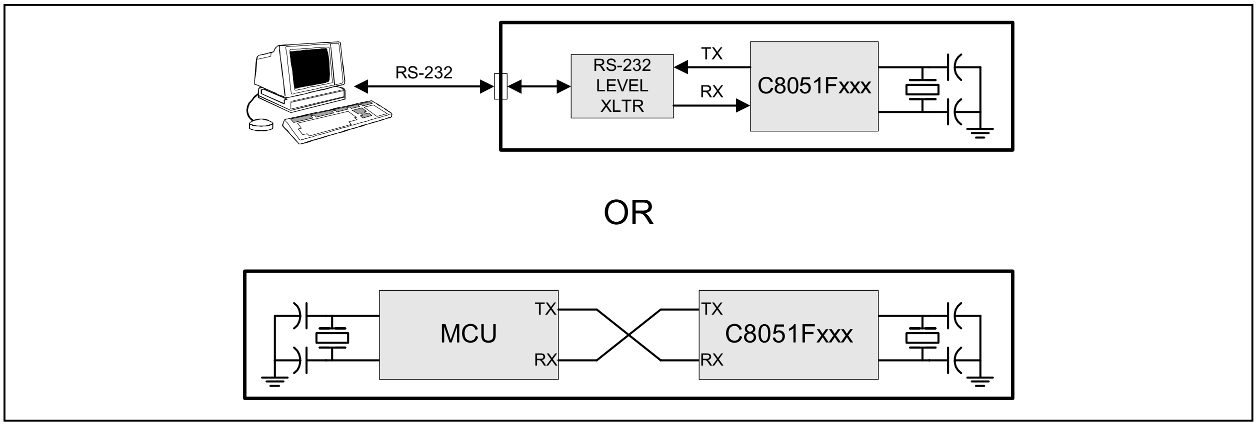


Figure 16.5. UART Modes 1, 2, and 3 Interconnect Diagram



16.1.3. Mode 2: 9-Bit UART, Fixed Baud Rate

Mode 2 provides asynchronous, full-duplex communication using a total of eleven bits per data byte: a start bit, 8 data bits (LSB first), a programmable ninth data bit, and a stop bit. On transmit, the ninth data bit is determined by the value in TB8 (SCON.3). It can be assigned the value of the parity flag P in the PSW or used in multiprocessor communications. On receive, the ninth data bit goes into RB8 (SCON.2) and the stop bit is ignored.

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 9^{th} bit must be logic 1.

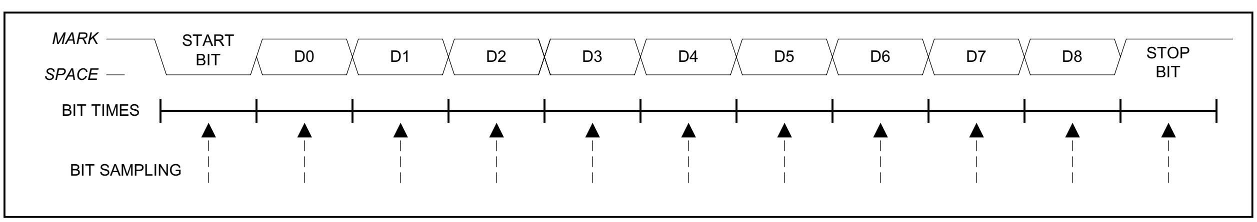
If these conditions are met, the eight bits of data is stored in SBUF, the ninth bit is stored in RB8 and the RI flag is set. If these conditions are not met, SBUF and RB8 will not be loaded and the RI flag will not be set. An interrupt will occur if enabled when either TI or RI are set.

The baud rate in Mode 2 is a direct function of the system clock frequency as follows:

$$\textit{Mode 2 Baud Rate} = 2^{SMOD} * (SYSCLK / 64).$$

The SMOD bit (PCON.7) selects whether to divide SYSCLK by 32 or 64. In the formula, 2 is raised to the power SMOD, resulting in a baud rate of either 1/32 or 1/64 of the system clock frequency. On reset, the SMOD bit is logic 0, thus selecting the lower speed baud rate by default.

Figure 16.6. UART Modes 2 and 3 Timing Diagram



16.1.4. Mode 3: 9-Bit UART, Variable Baud Rate

Mode 3 is the same as Mode 2 in all respects except the baud rate is variable. The baud rate is determined in the same manner as for Mode 1. Mode 3 operation transmits 11 bits: a start bit, 8 data bits (LSB first), a programmable ninth data bit, and a stop bit. Timer 1 or Timer 2 overflows generate the baud rate just as with Mode 1. In summary, Mode 3 transmits using the same protocol as Mode 2 but with Mode 1 baud rate generation.

16.2. Multiprocessor Communications

Modes 2 and 3 support multiprocessor communication between a master processor and one or more slave processors by special use of the ninth data bit. When a master processor wants to transmit to one or more slaves, it first sends an address byte to select the target(s). An address byte differs from a data byte in that its ninth bit is logic 1; in a data byte, the ninth bit is always set to logic 0.

Setting the SM2 bit (SCON.5) of a slave processor configures its UART such that when a stop bit is received, the UART will generate an interrupt only if the ninth bit is logic one (RB8 = 1) signifying an address byte has been received. In the UART's interrupt handler, software will compare the received address with the slave's own assigned 8-bit address. If the addresses match, the slave will clear its SM2 bit to enable interrupts on the reception of the following data byte(s). Slaves that weren't addressed leave their SM2 bits set and do not generate interrupts on the reception of the following data bytes, thereby ignoring the data. Once the entire message is received, the addressed slave resets its SM2 bit to ignore all transmissions until it receives the next address byte.

Multiple addresses can be assigned to a single slave and/or a single address can be assigned to multiple slaves, thereby enabling "broadcast" transmissions to more than one slave simultaneously. The master processor can be configured to receive all transmissions or a protocol can be implemented such that the master/slave role is temporarily reversed to enable half-duplex transmission between the original master and slave(s).

Figure 16.7. UART Multi-Processor Mode Interconnect Diagram

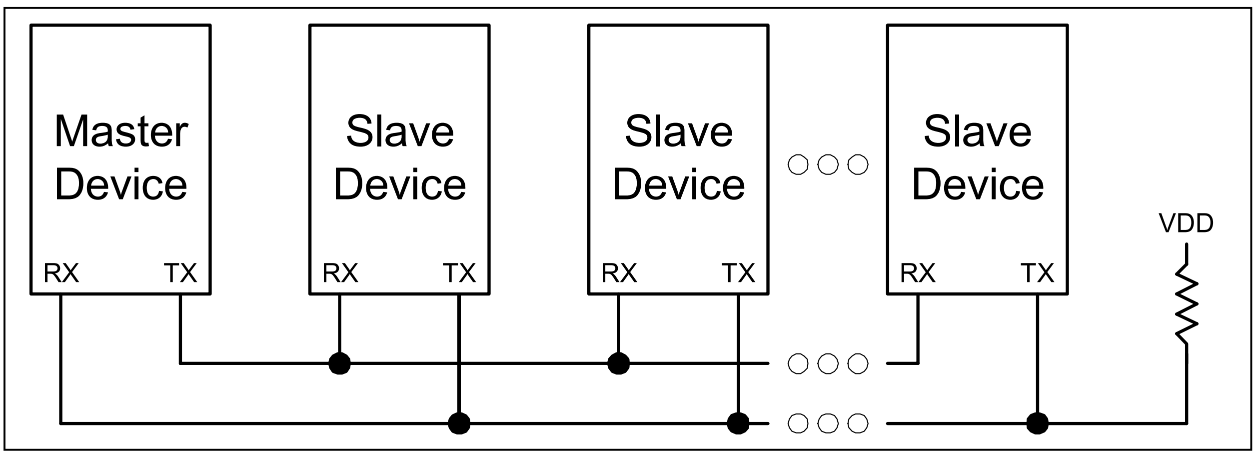


Table 16.2. Oscillator Frequencies for Standard Baud Rates

Oscillator Frequency (MHz)	Divide Factor	Timer 1 Load Value*	Resulting Baud Rate**
24.0	208	0xF3	115200 (115384)
23.592	205	0xF3	115200 (113423)
22.1184	192	0xF4	115200
18.432	160	0xF6	115200
16.5888	144	0xF7	115200
14.7456	128	0xF8	115200
12.9024	112	0xF9	115200
11.0592	96	0xFA	115200
9.216	80	0xFB	115200
7.3728	64	0xFC	115200
5.5296	48	0xFD	115200
3.6864	32	0xFE	115200
1.8432	16	0xFF	115200
24.576	320	0xEC	76800
25.0	434	0xE5	57600 (57870)
25.0	868	0xCA	28800
24.576	848	0xCB	28800 (28921)
24.0	833	0xCC	28800 (28846)
23.592	819	0xCD	28800 (28911)
22.1184	768	0xD0	28800
18.432	640	0xD8	28800
16.5888	576	0xDC	28800
14.7456	512	0xE0	28800
12.9024	448	0xE4	28800
11.0592	348	0xE8	28800
9.216	320	0xEC	28800
7.3728	256	0xF0	28800
5.5296	192	0xF4	28800
3.6864	128	0xF8	28800
1.8432	64	0xFC	28800

* Assumes SMOD=1, T1M=1.
** Numbers in parenthesis show the actual baud rate.

Figure 16.8. SBUF: Serial (UART) Data Buffer 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: 0x99

Bits7-0: SBUF.[7:0]: Serial Data Buffer Bits 7-0 (MSB-LSB)
This is actually two registers; a transmit and a receive buffer register. When data is moved to SBUF, it goes to the transmit buffer and is held for serial transmission. Moving a byte to SBUF is what initiates the transmission. When data is moved from SBUF, it comes from the receive buffer.

Figure 16.9. SCON: Serial Port Control Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
SM0	SM1	SM2	REN	TB8	RB8	TI	RI	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
							(bit addressable)	0x98

Bits7-6: SM0-SM1: Serial Port Operation Mode.
These bits select the Serial Port Operation Mode.

SM0	SM1	Mode
0	0	Mode 0: Synchronous Mode
0	1	Mode 1: 8-Bit UART, Variable Baud Rate
1	0	Mode 2: 9-Bit UART, Fixed Baud Rate
1	1	Mode 3: 9-Bit UART, Variable Baud Rate

Bit5: SM2: Multiprocessor Communication Enable.
The function of this bit is dependent on the Serial Port Operation Mode.
Mode 0: No effect
Mode 1: Checks for valid stop bit.
0: Logic level of stop bit is ignored.
1: RI will only be activated if stop bit is logic level 1.
Mode 2 and 3: Multiprocessor Communications Enable.
0: Logic level of ninth bit is ignored.
1: RI is set and an interrupt is generated only when the ninth bit is logic 1.

Bit4: REN: Receive Enable.
This bit enables/disables the UART receiver.
0: UART reception disabled.
1: UART reception enabled.

Bit3: TB8: Ninth Transmission Bit.
The logic level of this bit will be assigned to the ninth transmission bit in Modes 2 and 3. It is not used in Modes 0 and 1. Set or cleared by software as required.

Bit2: RB8: Ninth Receive Bit.
The bit is assigned the logic level of the ninth bit received in Modes 2 and 3. In Mode 1, if SM2 is logic 0, RB8 is assigned the logic level of the received stop bit. RB8 is not used in Mode 0.

Bit1: TI: Transmit Interrupt Flag.
Set by hardware when a byte of data has been transmitted by the UART (after the 8th bit in Mode 0, or at the beginning of the stop bit in other modes). When the UART interrupt is enabled, setting this bit causes the CPU to vector to the UART interrupt service routine. This bit must be cleared manually by software

Bit0: RI: Receive Interrupt Flag.
Set by hardware when a byte of data has been received by the UART (after the 8th bit in Mode 0, or after the stop bit in other modes – see SM2 bit for exception). When the UART interrupt is enabled, setting this bit causes the CPU to vector to the UART interrupt service routine. This bit must be cleared manually by software.

17. TIMERS

The CIP-51 implements three, 16-bit counter/timers comparable with those found in the standard 8051 MCU's. These can be used to measure time intervals, count external events and generate periodic interrupt requests. Timer 0 and Timer 1 are nearly identical and have four primary modes of operation. Timer 2 offers additional capabilities not available in Timers 0 and 1, such as capture and baud rate generation.

Timer 0 and Timer 1:	Timer 2:
13-bit counter/timer	16-bit counter/timer with auto-reload
16-bit counter/timer	16-bit counter/timer with capture
8-bit counter/timer with auto-reload	Baud rate generator
Two 8-bit counter/timers (Timer 0 only)	

When functioning as a timer, the counter/timer registers are incremented on each clock tick. Clock ticks are derived from the system clock divided by either one or twelve as specified by the Timer Clock Select bits (T2M-T0M) in CKCON. The twelve-clocks-per-tick option provides compatibility with the older generation of the 8051 family. Applications that require a faster timer can use the one-clock-per-tick option.

When functioning as a counter, a counter/timer register is incremented on each high-to-low transition at the selected input pin (P0.4/T0, P0.5/T1, or P0.6/T2. Events with a frequency of up to one-fourth the system clock's frequency can be counted. The input signal need not be periodic, but it should be held at a given level for at least two full system clock cycles to ensure the level is sampled.

17.1. Timer 0 and Timer 1

Timer 0 and Timer 1 are accessed and controlled through SFR's. Each counter/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 (TCON) register is used to enable Timer 0 and Timer 1 as well as indicate their status. Both counter/timers operate in one of four primary modes selected by setting the Mode Select bits M1-M0 in the Counter/Timer Mode (TMOD) register. Each timer can be configured independently. Following is a detailed description of each operating mode.

17.1.1. Mode 0: 13-bit Counter/Timer

Timer 0 and Timer 1 operate as a 13-bit counter/timer 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 MSB's 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 (TCON.5) is set and an interrupt will occur if enabled.

The C/T0 bit (TMOD.2) selects the counter/timer's clock source. Clearing C/T selects the system clock as the input for the timer. When C/T0 is set to logic 1, high-to-low transitions at the selected input pin increment the timer register. (Refer to section 14 for information on selecting and configuring external I/O pins.)

Setting the TR0 bit (TCON.4) enables the timer when either GATE0 (TMOD.3) is 0 or the input signal /INT0 is logic-level one. Setting GATE0 to logic 1 allows the timer to be controlled by the external input signal /INT0, facilitating pulse width measurements.

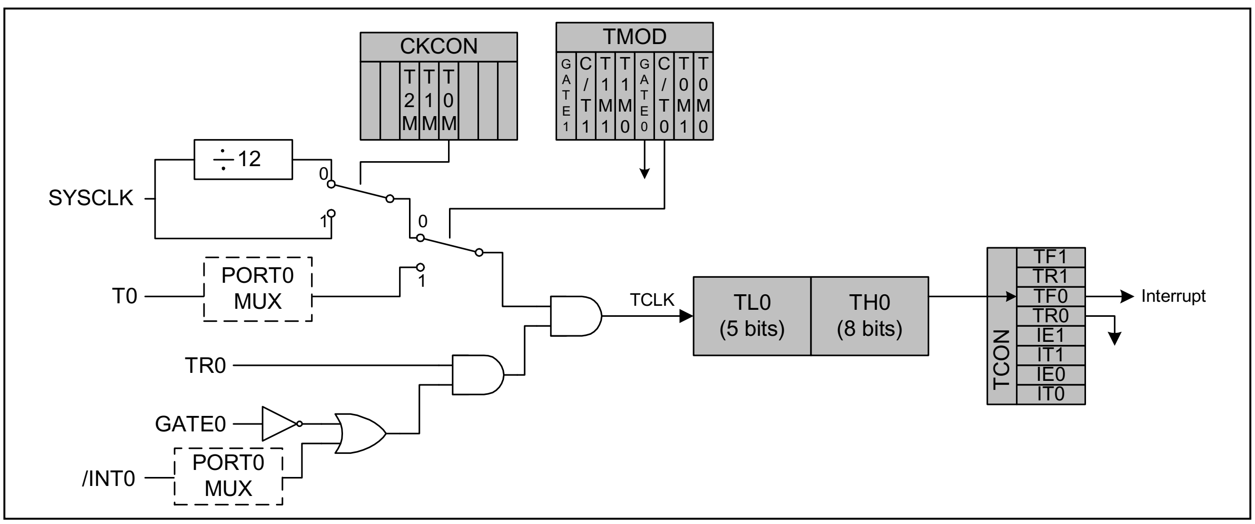
TR0	GATE0	/INT0	Counter/Timer
0	X	X	Disabled
1	0	X	Enabled
1	1	0	Disabled
1	1	1	Enabled

X = Don't Care

Setting TR0 does not reset the timer register. The timer register should be initialized to the desired value before enabling the timer.

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.

Figure 17.1. T0 Mode 0 Block Diagram



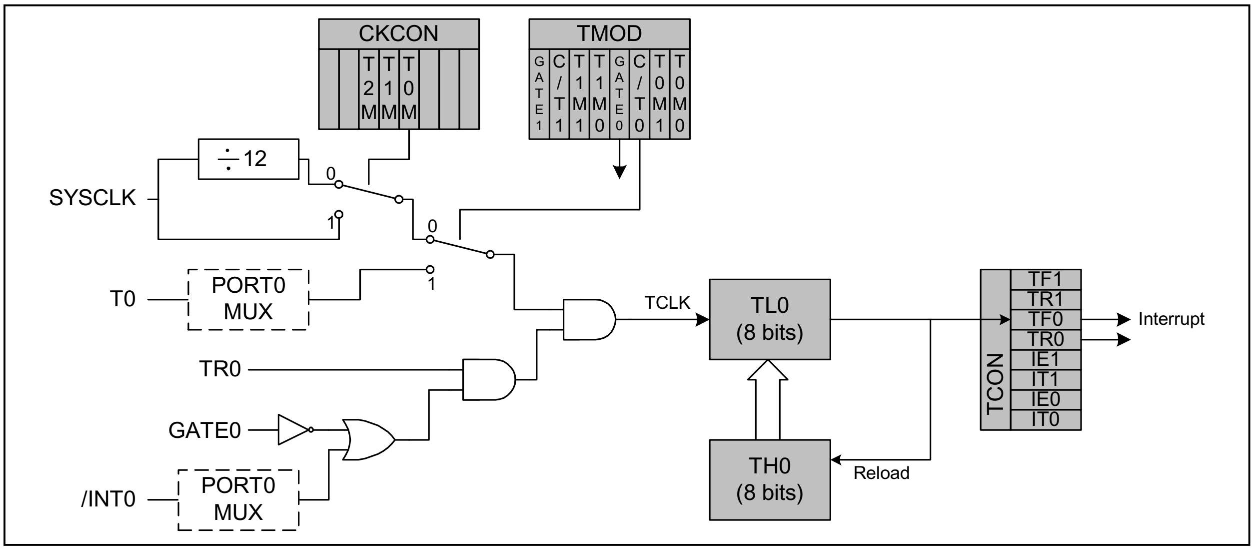
17.1.2. Mode 1: 16-bit Counter/Timer

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

17.1.3. 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 counter 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 17.2. T0 Mode 2 Block Diagram



17.1.4. Mode 3: Two 8-bit Counter/Timers (Timer 0 Only)

Timer 0 and Timer 1 behave differently in Mode 3. Timer 0 is configured as two separate 8-bit counter/timers held in TL0 and TH0. The counter/timer in TL0 is controlled using the Timer 0 control/status bits in TCON and TMOD: TR0, C/T0, GATE0 and TF0. It can use either the system clock or an external input signal as its time base. The TH0 register is restricted to a timer function sourced by the system clock. TH0 is enabled using the Timer 1 run control bit TR1. TH0 sets the Timer 1 overflow flag TF1 on overflow and thus controls the Timer 1 interrupt.

Timer 1 is inactive in Mode 3, so with Timer 0 in Mode 3, Timer 1 can be turned off and on by switching it into and out of its Mode 3. When Timer 0 is in Mode 3, Timer 1 can be operated in Modes 0, 1 or 2, but cannot be clocked by external signals nor set the TF1 flag and generate an interrupt. However, the Timer 1 overflow can be used for baud rate generation. Refer to Section 16 (UART) for information on configuring Timer 1 for baud rate generation.

Figure 17.3. T0 Mode 3 Block Diagram

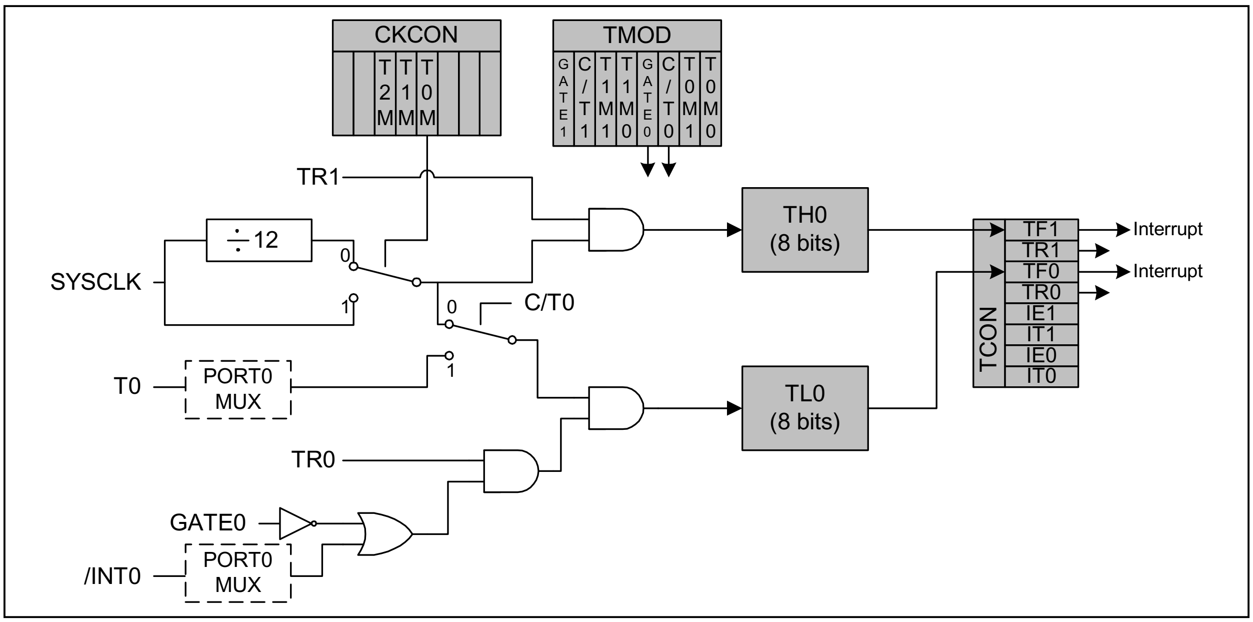


Figure 17.4. TCON: Timer Control Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0x88

Bit7: TF1: Timer 1 Overflow Flag.
Set by hardware when Timer 1 overflows. This flag can be cleared by software but is automatically cleared when the CPU vectors to the Timer 1 interrupt service routine.
0: No Timer 1 overflow detected.
1: Timer 1 has overflowed.

Bit6: TR1: Timer 1 Run Control.
0: Timer 1 disabled.
1: Timer 1 enabled.

Bit5: TF0: Timer 0 Overflow Flag.
Set by hardware when Timer 0 overflows. This flag can be cleared by software but is automatically cleared when the CPU vectors to the Timer 0 interrupt service routine.
0: No Timer 0 overflow detected.
1: Timer 0 has overflowed.

Bit4: TR0: Timer 0 Run Control.
0: Timer 0 disabled.
1: Timer 0 enabled.

Bit3: IE1: External Interrupt 1.
This flag is set by hardware when an edge/level of type defined by IT1 is detected. It can be cleared by software but is automatically cleared when the CPU vectors to the External Interrupt 1 service routine if IT1 = 1. This flag is the inverse of the /INT1 input signal's logic level when IT1 = 0.

Bit2: IT1: Interrupt 1 Type Select.
This bit selects whether the configured /INT1 signal will detect falling edge or active-low level-sensitive interrupts.
0: /INT1 is level triggered.
1: /INT1 is edge triggered.

Bit1: IE0: External Interrupt 0.
This flag is set by hardware when an edge/level of type defined by IT0 is detected. It can be cleared by software but is automatically cleared when the CPU vectors to the External Interrupt 0 service routine if IT0 = 1. This flag is the inverse of the /INT0 input signal's logic level when IT0 = 0.

Bit0: IT0: Interrupt 0 Type Select.
This bit selects whether the configured /INT0 signal will detect falling edge or active-low level-sensitive interrupts.
0: /INT0 is level triggered.
1: /INT0 is edge triggered.

Figure 17.5. TMOD: Timer Mode Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
GATE1	C/T1	T1M1	T1M0	GATE0	C/T0	T0M1	T0M0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0x89

Bit7: GATE1: Timer 1 Gate Control.
0: Timer 1 enabled when TR1 = 1 irrespective of /INT1 logic level.
1: Timer 1 enabled only when TR1 = 1 AND /INT1 = logic level one.

Bit6: C/T1: Counter/Timer 1 Select.
0: Timer Function: Timer 1 incremented by clock defined by T1M bit (CKCON.4).
1: Counter Function: Timer 1 incremented by high-to-low transitions on external input pin P0.5/T1.

Bits5-4: T1M1-T1M0: Timer 1 Mode Select.
These bits select the Timer 1 operation mode.

T1M1	T1M0	Mode
0	0	Mode 0: 13-bit counter/timer
0	1	Mode 1: 16-bit counter/timer
1	0	Mode 2: 8-bit counter/timer with auto-reload
1	1	Mode 3: Timer 1 Inactive/stopped

Bit3: GATE0: Timer 0 Gate Control.
0: Timer 0 enabled when TR0 = 1 irrespective of /INT0 logic level.
1: Timer 0 enabled only when TR0 = 1 AND /INT0 = logic level one.

Bit2: C/T0: Counter/Timer Select.
0: Timer Function: Timer 0 incremented by clock defined by T0M bit (CKCON.3).
1: Counter Function: Timer 0 incremented by high-to-low transitions on external input pin P0.4/T0.

Bits1-0: T0M1-T0M0: Timer 0 Mode Select.
These bits select the Timer 0 operation mode.

T0M1	T0M0	Mode
0	0	Mode 0: 13-bit counter/timer
0	1	Mode 1: 16-bit counter/timer
1	0	Mode 2: 8-bit counter/timer with auto-reload
1	1	Mode 3: Two 8-bit counter/timers

Figure 17.6. CKCON: Clock Control Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
-	-	T2M	T1M	T0M	-	-	-	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0x8E

Bits7-6: UNUSED. Read = 00b, Write = don't care.

Bit5: T2M: Timer 2 Clock Select.
This bit controls the division of the system clock supplied to Timer 2. This bit is ignored when the timer is in baud rate generator mode or counter mode (i.e. C/T2 = 1).
0: Timer 2 uses the system clock divided by 12.
1: Timer 2 uses the system clock.

Bit4: T1M: Timer 1 Clock Select.
This bit controls the division of the system clock supplied to Timer 1.
0: Timer 1 uses the system clock divided by 12.
1: Timer 1 uses the system clock.

Bit3: T0M: Timer 0 Clock Select.
This bit controls the division of the system clock supplied to Counter/Timer 0.
0: Counter/Timer uses the system clock divided by 12.
1: Counter/Timer uses the system clock.

Bits2-0: UNUSED. Read = 000b, Write = don't care.

Figure 17.7. TL0: Timer 0 Low Byte

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: 0x8A

Bits 7-0: TL0: Timer 0 Low Byte.
The TL0 register is the low byte of the 16-bit Timer 0.

Figure 17.8. TL1: Timer 1 Low Byte

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: 0x8B

Bits 7-0: TL1: Timer 1 Low Byte.
The TL1 register is the low byte of the 16-bit Timer 1.

Figure 17.9. TH0: Timer 0 High Byte

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: 0x8C

Bits 7-0: TH0: Timer 0 High Byte.
The TH0 register is the high byte of the 16-bit Timer 0.

Figure 17.10. TH1: Timer 1 High Byte

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: 0x8D

Bits 7-0: TH1: Timer 1 High Byte.
The TH1 register is the high byte of the 16-bit Timer 1.

17.2. Timer 2

Timer 2 is a 16-bit counter/timer formed by the two 8-bit SFR's: TL2 (low byte) and TH2 (high byte). As with Timers 0 and 1, Timer 2 can use either the system clock or transitions on an external input pin as its clock source. The Counter/Timer Select bit C/T2 bit (T2CON.1) selects the clock source for Timer 2. Clearing C/T2 selects the system clock as the input for the timer (divided by either one or twelve as specified by the Timer Clock Select bit T2M in CKCON). When C/T2 is set to 1, high-to-low transitions at the T2 input pin increment the counter/timer register. (Refer to Section 14 for information on selecting and configuring external I/O pins.) Timer 2 can also be used to start an ADC Data Conversion (see section 5).

Timer 2 offers capabilities not found in Timer 0 and Timer 1. It operates in one of three modes: 16-bit Counter/Timer with Capture, 16-bit Counter/Timer with Auto-Reload or Baud Rate Generator Mode. Timer 2's operating mode is selected by setting configuration bits in the Timer 2 Control (T2CON) register. Below is a summary of the Timer 2 operating modes and the T2CON bits used to configure the counter/timer. Detailed descriptions of each mode follow.

RCLK	TCLK	CP/RL2	TR2	Mode
0	0	1	1	16-bit Counter/Timer with Capture
0	0	0	1	16-bit Counter/Timer with Auto-Reload
0	1	X	1	Baud Rate Generator for TX
1	0	X	1	Baud Rate Generator for RX
1	1	X	1	Baud Rate Generator for TX and RX
X	X	X	0	Off

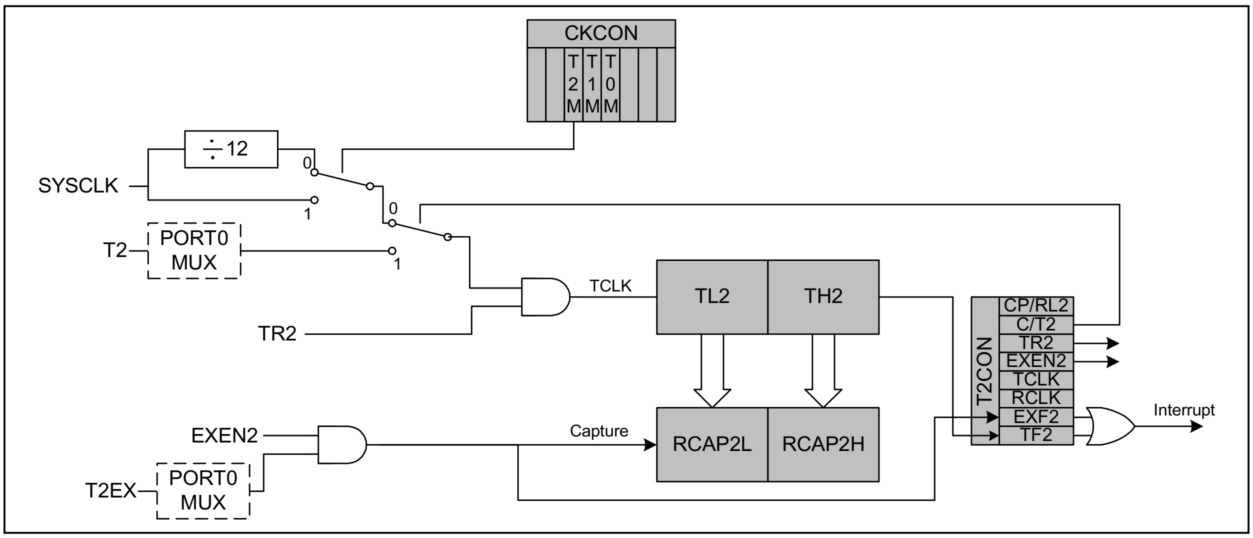
17.2.1. Mode 0: 16-bit Counter/Timer with Capture

In this mode, Timer 2 operates as a 16-bit counter/timer with capture facility. A high-to-low transition on the T2EX input pin causes the 16-bit value in Timer 2 (TH2, TL2) to be loaded into the capture registers (RCAP2H, RCAP2L).

Timer 2 can use either SYSCLK, SYSCLK divided by 12, or high-to-low transitions on the external T2 input pin as its clock source when operating in Counter/Timer with Capture mode. Clearing the C/T2 bit (T2CON.1) selects the system clock as the input for the timer (divided by one or twelve as specified by the Timer Clock Select bit T2M in CKCON). When C/T2 is set to logic 1, a high-to-low transition at the T2 input pin increments the counter/timer register. As the 16-bit counter/timer register increments and overflows from 0xFFFF to 0x0000, the TF2 timer overflow flag (T2CON.7) is set and an interrupt will occur if the interrupt is enabled.

Counter/Timer with Capture mode is selected by setting the Capture/Reload Select bit CP/RL2 (T2CON.0) and the Timer 2 Run Control bit TR2 (T2CON.2) to logic 1. The Timer 2 External Enable EXEN2 (T2CON.3) must also be set to logic 1 to enable a capture. If EXEN2 is cleared, transitions on T2EX will be ignored.

Figure 17.11. T2 Mode 0 Block Diagram

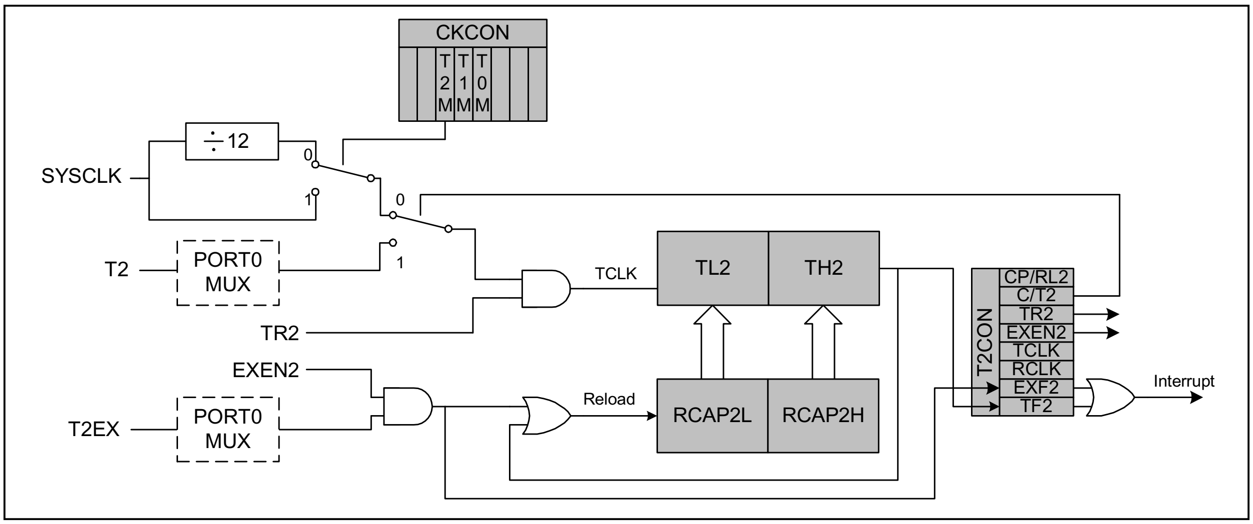


17.2.2. Mode 1: 16-bit Counter/Timer with Auto-Reload

The Counter/Timer with Auto-Reload mode sets the TF2 timer overflow flag when the counter/timer register overflows from 0xFFFF to 0x0000. An interrupt is generated if enabled. On overflow, the 16-bit value held in the two capture registers (RCAP2H, RCAP2L) is automatically loaded into the counter/timer register and the timer is restarted.

Counter/Timer with Auto-Reload mode is selected by clearing the CP/RL2 bit. Setting TR2 to logic 1 enables and starts the timer. Timer 2 can use either the system clock or transitions on an external input pin as its clock source, as specified by the C/T2 bit. If EXEN2 is set to logic 1, a high-to-low transition on T2EX will also cause Timer 2 to be reloaded. If EXEN2 is cleared, transitions on T2EX will be ignored.

Figure 17.12. T2 Mode 1 Block Diagram



17.2.3. Mode 2: Baud Rate Generator

Timer 2 can be used as a baud rate generator for the serial port (UART) when the UART is operated in modes 1 or 3 (refer to Section 16.1 for more information on UART operational modes). In Baud Rate Generator mode, Timer 2 works similarly to the auto-reload mode. On overflow, the 16-bit value held in the two capture registers (RCAP2H, RCAP2L) is automatically loaded into the counter/timer register. However, the TF2 overflow flag is not set and no interrupt is generated. Instead, the overflow event is used as the input to the UART's shift clock. Timer 2 overflows can be used to generate baud rates for transmit and/or receive independently.

The Baud Rate Generator mode is selected by setting RCLK (T2CON.5) and/or TCLK (T2CON.4) to logic one. When RCLK or TCLK is set to logic 1, Timer 2 operates in the auto-reload mode regardless of the state of the CP/RL2 bit. The baud rate for the UART, when operating in mode 1 or 3, is determined by the Timer 2 overflow rate:

Baud Rate = Timer 2 Overflow Rate / 16.

Note, in all other modes, the time base for the timer is the system clock divided by one or twelve as selected by the T2M bit in CKCON. However, in Baud Rate Generator mode, the time base is the system clock divided by two. No other divisor selection is possible. If a different time base is required, setting the C/T2 bit to logic 1 will allow the time base to be derived from the external input pin T2. In this case, the baud rate for the UART is calculated as:

*Baud Rate = FCLK / [32 * (65536 – [RCAP2H:RCAP2L])]*

Where FCLK is the frequency of the signal supplied to T2 and [RCAP2H:RCAP2L] is the 16-bit value held in the capture registers.

As explained above, in Baud Rate Generator mode, Timer 2 does not set the TF2 overflow flag and therefore cannot generate an interrupt. However, if EXEN2 is set to logic 1, a high-to-low transition on the T2EX input pin will set the EXF2 flag and a Timer 2 interrupt will occur if enabled. Therefore, the T2EX input may be used as an additional external interrupt source.

Figure 17.13. T2 Mode 2 Block Diagram

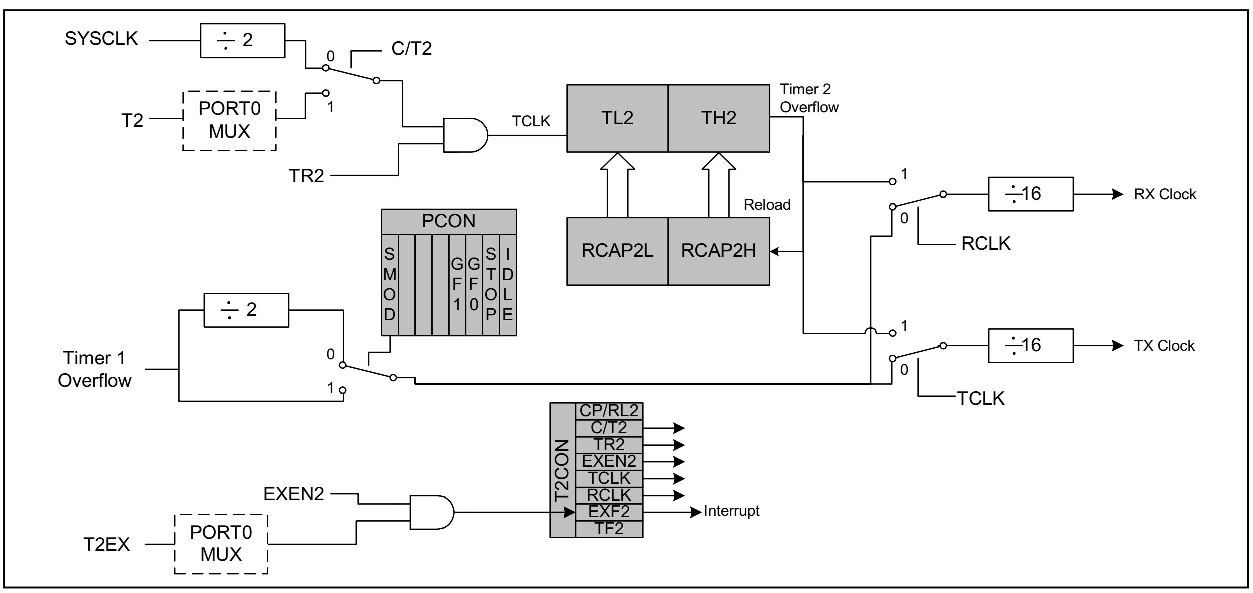


Figure 17.14. T2CON: Timer 2 Control Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2	CP/RL2	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0 (bit addressable)	SFR Address: 0xC8

Bit7: TF2: Timer 2 Overflow Flag.
Set by hardware when Timer 2 overflows from 0xFFFF to 0x0000. When the Timer 2 interrupt is enabled, setting this bit causes the CPU to vector to the Timer 2 interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software. TF2 will not be set when RCLK and/or TCLK are logic 1.

Bit6: EXF2: Timer 2 External Flag.
Set by hardware when either a capture or reload is caused by a high-to-low transition on the T2EX input pin and EXEN2 is logic 1. When the Timer 2 interrupt is enabled, setting this bit causes the CPU to vector to the Timer 2 Interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software.

Bit5: RCLK: Receive Clock Flag.
Selects which timer is used for the UART's receive clock in modes 1 or 3.
0: Timer 1 overflows used for receive clock.
1: Timer 2 overflows used for receive clock.

Bit4: TCLK: Transmit Clock Flag.
Selects which timer is used for the UART's transmit clock in modes 1 or 3.
0: Timer 1 overflows used for transmit clock.
1: Timer 2 overflows used for transmit clock.

Bit3: EXEN2: Timer 2 External Enable.
Enables high-to-low transitions on T2EX to trigger captures or reloads when Timer 2 is not operating in Baud Rate Generator mode.
0: High-to-low transitions on T2EX ignored.
1: High-to-low transitions on T2EX cause a capture or reload.

Bit2: TR2: Timer 2 Run Control.
This bit enables/disables Timer 2.
0: Timer 2 disabled.
1: Timer 2 enabled.

Bit1: C/T2: Counter/Timer Select.
0: Timer Function: Timer 2 incremented by clock defined by T2M (CKCON.5).
1: Counter Function: Timer 2 incremented by high-to-low transitions on external input pin P0.6/T2.

Bit0: CP/RL2: Capture/Reload Select.
This bit selects whether Timer 2 functions in capture or auto-reload mode. EXEN2 must be logic 1 for high-to-low transitions on T2EX to be recognized and used to trigger captures or reloads. If RCLK or TCLK is set, this bit is ignored and Timer 2 will function in auto-reload mode.
0: Auto-reload on Timer 2 overflow or high-to-low transition at T2EX (EXEN2 = 1).
1: Capture on high-to-low transition at T2EX (EXEN2 = 1).

Figure 17.15. RCAP2L: Timer 2 Capture Register Low Byte

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: 0xCA

Bits 7-0: RCAP2L: Timer 2 Capture Register Low Byte.
The RCAP2L register captures the low byte of Timer 2 when Timer 2 is configured in capture mode. When Timer 2 is configured in auto-reload mode, it holds the low byte of the reload value.

Figure 17.16. RCAP2H: Timer 2 Capture Register High Byte

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: 0xCB

Bits 7-0: RCAP2H: Timer 2 Capture Register High Byte.
The RCAP2H register captures the high byte of Timer 2 when Timer 2 is configured in capture mode. When Timer 2 is configured in auto-reload mode, it holds the high byte of the reload value.

Figure 17.17. TL2: Timer 2 Low Byte

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: 0xCC

Bits 7-0: TL2: Timer 2 Low Byte.
The TL2 register contains the low byte of the 16-bit Timer 2.

Figure 17.18. TH2: Timer 2 High Byte

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: 0xCD

Bits 7-0: TH2: Timer 2 High Byte.
The TH2 register contains the high byte of the 16-bit Timer 2.

18. JTAG

Description

The MCU has an on-chip JTAG interface and logic to support FLASH read and write operations and non-intrusive in-circuit debug. The C8051F2xx may be placed in a JTAG test chain in order to maintain only one JTAG interface in a system for boundary scan of other parts, and still utilize the C8051F2xx debug and FLASH programming. However, the C8051F2xx does NOT support boundary scan and will act as BYPASS as specified in IEEE 1149.1. The JTAG interface is implemented via four dedicated pins on the MCU, which are TCK, TMS, TDI, and TDO. These pins are all 5 volt tolerant.

Through the 16-bit JTAG Instruction Register (IR), five instructions shown in Figure 18.1 can be commanded. These commands can either select the device ID code, or select registers for FLASH programming operations. BYPASS is shown to illustrate its default setting. There are four Data Registers associated with the Flash read and write operations on the MCU.

Figure 18.1. IR: JTAG Instruction Register

Bit15															Bit0	Reset Value
																0x0000

IR value	Instruction	Description
0x0004	IDCODE	Selects device ID Register
0xFFFF	BYPASS	Selects bypass Data Register and is DEFAULT for the device. Note: The device does NOT support boundary scan. However, it may be placed in a scan chain and bypassed in a system of other devices utilizing boundary scan.
0x0082	Flash Control	Selects FLASHCON Register to control how the interface logic responds to reads and writes to the FLASHDAT Register
0x0083	Flash Data	Selects FLASHDAT Register for reads and writes to the Flash memory
0x0084	Flash Address	Selects FLASHADR Register which holds the address of all Flash read, write, and erase operations
0x0085	Flash Scale	Selects FLASHSCL Register which controls the prescaler used to generate timing signals for Flash operations

18.1. Flash Programming Commands

The Flash memory can be programmed directly over the JTAG interface using the Flash Control, Flash Data, Flash Address, and Flash Scale registers. These Indirect Data Registers are accessed via the JTAG Instruction Register. Read and write operations on indirect data registers are performed by first setting the appropriate DR address in the IR register. Each read or write is then initiated by writing the appropriate Indirect Operation Code (IndOpCode) to the selected data register. Incoming commands to this register have the following format:

19:18	17:0
IndOpCode	WriteData

IndOpCode: These bit set the operation to perform according to the following table:

IndOpCode	Operation
0x	Poll
10	Read
11	Write

The Poll operation is used to check the Busy bit as described below. Although a Capture-DR is performed, no Update-DR is allowed for the Poll operation. Since updates are disabled, polling can be accomplished by shifting in/out a single bit.

The Read operation initiates a read from the register addressed by the DRAddress. Reads can be initiated by shifting only 2 bits into the indirect register. After the read operation is initiated, polling of the Busy bit must be performed to determine when the operation is complete.

The write operation initiates a write of WriteData to the register addressed by DRAddress. Registers of any width up to 18 bits can be written. If the register to be written contains fewer than 18 bits, the data in WriteData should be left-justified, i.e. its MSB should occupy bit 17 above. This allows shorter registers to be written in fewer JTAG clock cycles. For example, an 8-bit register could be written by shifting only 10 bits. After a Write is initiated, the Busy bit should be polled to determine when the next operation can be initiated. The contents of the Instruction Register should not be altered while either a read or write operation is in progress.

Outgoing data from the indirect Data Register has the following format:

19	18:1	0
0	ReadData	Busy

The Busy bit indicates that the current operation is not complete. It goes high when an operation is initiated and returns low when complete. Read and Write commands are ignored while Busy is high. In fact, if polling for Busy to be low will be followed by another read or write operation, JTAG writes of the next operation can be made while checking for Busy to be low. They will be ignored until Busy is read low, at which time the new operation will initiate. This bit is placed at bit 0 to allow polling by single-bit shifts. When waiting for a Read to complete and Busy is 0, the following 18 bits can be shifted out to obtain the resulting data. ReadData is always right-justified. This allows registers shorter than 18 bits to be read using a reduced number of shifts. For example, the result from a byte-read requires 9 bit shifts (Busy + 8 bits).

Figure 18.2 FLASHCON: JTAG Flash Control Register

WRMD3	WRMD2	WRMD1	WRMD0	RDMD3	RDMD2	RDMD1	RDMD0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000

This register determines how the Flash interface logic will respond to reads and writes to the FLASHDAT Register.

Bits7-4: WRMD3-0: Write Mode Select Bits.
The Write Mode Select Bits control how the interface logic responds to writes to the FLASHDAT Register per the following values:
0000: A FLASHDAT write replaces the data in the FLASHDAT register, but is otherwise ignored.
0001: A FLASHDAT write initiates a write of FLASHDAT into the memory address selected by the FLASHADR register. FLASHADR is incremented by one when complete.
0010: A FLASHDAT write initiates an erasure (sets all bytes to 0xFF) of the Flash page containing the address in FLASHADR. FLASHDAT must be 0xA5 for the erase to occur. FLASHADR is not affected. If FLASHADR = 0x1DFE – 0x1DFF, the entire user space will be erased (i.e. entire Flash memory except for Reserved area 0x1E00 – 0x1FFF).
(All other values for WRMD3-0 are reserved.)

Bits3-0: RDMD3-0: Read Mode Select Bits.
The Read Mode Select Bits control how the interface logic responds to reads to the FLASHDAT Register per the following values:
0000: A FLASHDAT read provides the data in the FLASHDAT register, but is otherwise ignored.
0001: A FLASHDAT read initiates a read of the byte addressed by the FLASHADR register if no operation is currently active. This mode is used for block reads.
0010: A FLASHDAT read initiates a read of the byte addressed by FLASHADR only if no operation is active and any data from a previous read has already been read from FLASHDAT. This mode allows single bytes to be read (or the last byte of a block) without initiating an extra read.
(All other values for RDMD3-0 are reserved.)

Figure 18.3. FLASHADR: JTAG Flash Address Register

																Reset Value
Bit15															Bit0	0x0000

This register holds the address for all JTAG Flash read, write, and erase operations. This register autoincrements after each read or write, regardless of whether the operation succeeded or failed.

Bits15-0: Flash Operation 16-bit Address.

Figure 18.4. FLASHDAT: JTAG Flash Data Register

DATA7	DATA6	DATA5	DATA4	DATA3	DATA2	DATA1	DATA0	FAIL	BUSY	Reset Value
Bit9	Bit8	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	0000000000

This register is used to read or write data to the Flash memory across the JTAG interface.

Bits9-2: DATA7-0: Flash Data Byte.

Bit1: FAIL: Flash Fail Bit.
- 0: Previous Flash memory operation was successful.
- 1: Previous Flash memory operation failed. Usually indicates the associated memory location was locked.

Bit0: BUSY: Flash Busy Bit.
- 0: Flash interface logic is not busy.
- 1: Flash interface logic is processing a request. Reads or writes while BUSY = 1 will not initiate another operation

Figure 18.5. FLASHSCL: JTAG Flash Scale Register

FOSE	FRAE	-	-	FLSCL3	FLSCL2	FLSCL1	FLSCL0	Reset Value
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	00000000

This register controls the Flash read timing circuit and the prescaler required to generate the correct timing for Flash operations.

Bit7: FOSE: Flash One-Shot Enable Bit.
- 0: Flash read strobe is a full clock-cycle wide.
- 1: Flash read strobe is 50nsec.

Bit6: FRAE: Flash Read Always Bit.
- 0: The Flash output enable and sense amplifier enable are on only when needed to read the Flash memory.
- 1: The Flash output enable and sense amplifier enable are always on. This can be used to limit the variations in digital supply current due to switching the sense amplifiers, thereby reducing digitally induced noise.

Bits5-4: UNUSED. Read = 00b, Write = don't care.

Bits3-0: FLSCL3-0: Flash Prescaler Control Bits.
The FLSCL3-0 bits control the prescaler used to generate timing signals for Flash operations. Its value should be written before any Flash write or erase operations are initiated. The value written should be the smallest integer for which:

$$FLSCL[3:0] > \log_2(f_{SYSCLK} / 50kHz)$$

Where f_{SYSCLK} is the system clock frequency. All Flash read/write/erase operations are disallowed when FLSCL[3:0] = 1111b.

18.2. Boundary Scan Bypass and ID Code

The MCU does not support boundary scan (IEEE 1149.1), however, it does support the bypass and ID code functions. Because the MCU utilizes JTAG for FLASH memory programming and debug support, and other devices in a system may use JTAG boundary scan, the MCU supports being placed in BYPASS so the user may maintain a single JTAG port for a system. Additionally, the MCU supports an ID code.

18.2.1. BYPASS Instruction

The BYPASS instruction is accessed via the IR. It provides access to the standard 1-bit JTAG Bypass data register.

18.2.2. IDCODE Instruction

The IDCODE instruction is accessed via the IR. It provides access to the 32-bit Device ID register.

Figure 18.6. DEVICEID: JTAG Device ID Register

Version	Part Number	Manufacturer ID	1	Reset Value
Bit31 … Bit28	Bit27 … Bit12	Bit11 … Bit1	Bit0	0xn0000243

Version = 0000b (Revision A)
= 0001b (Revision B)

Part Number = 0000 0000 0000 0001b (C8051F220/1/6, C8051F230/1/6)

Manufacturer ID = 0010 0100 001b (Cygnal Integrated Products)

18.3. Debug Support

The 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. Cygnal's debug system supports inspection and modification of memory and registers, breakpoints, stack tracing, and single stepping. 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 emulating. The WDT is disabled when the MCU is halted during single stepping or at a breakpoint.

The C8051F2xxDK is a development kit with all the hardware and software necessary to develop application code and perform in-circuit debugging with the C8061F206, C8051F220/1/6 and C8051F230/1/6. The kit includes an Integrated Development Environment (IDE) which has a debugger and integrated 8051 assembler. It has an RS-232 to JTAG interface module referred to as the EC. The kit also includes RS-232 and JTAG cables, and wall-mount power supply.

CYGNAL Integrated Products, Inc.
4301 Westbank Drive
Suite B-100
Austin, TX 78746
512-327-7088
(fax) 512-327-7087
www.cygnal.com