Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Not For New Designs
Core Processor	8051
Core Size	8-Bit
Speed	25MHz
Connectivity	SMBus (2-Wire/I²C), SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, Temp Sensor, WDT
Number of I/O	16
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2.25K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 8x10b; D/A 2x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	48-TQFP
Supplier Device Package	48-TQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f016-gq

Figure 1.1. C8051F000/05/10/15 Block Diagram

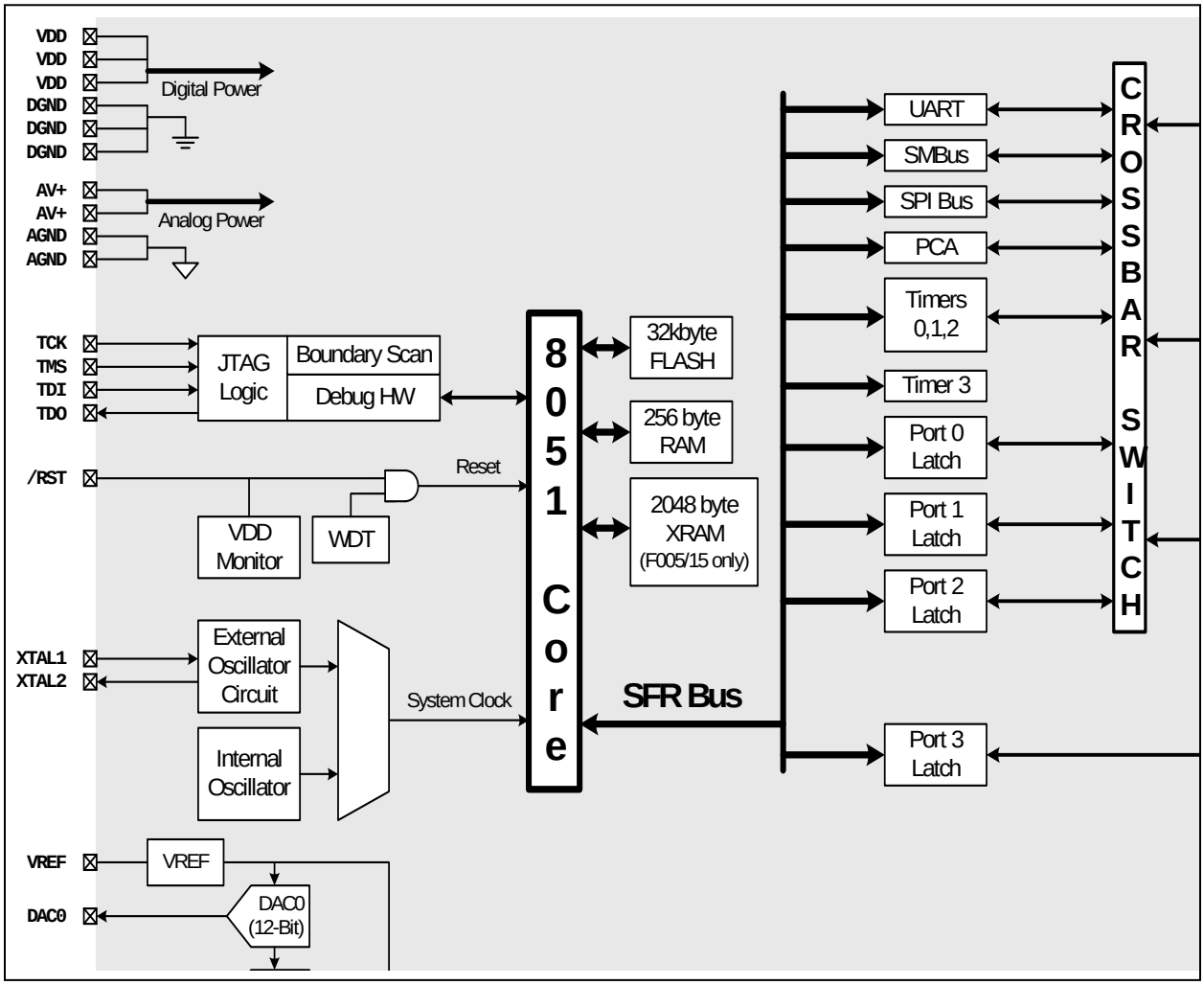


Figure 4.3. TQFP-48 Pinout Diagram

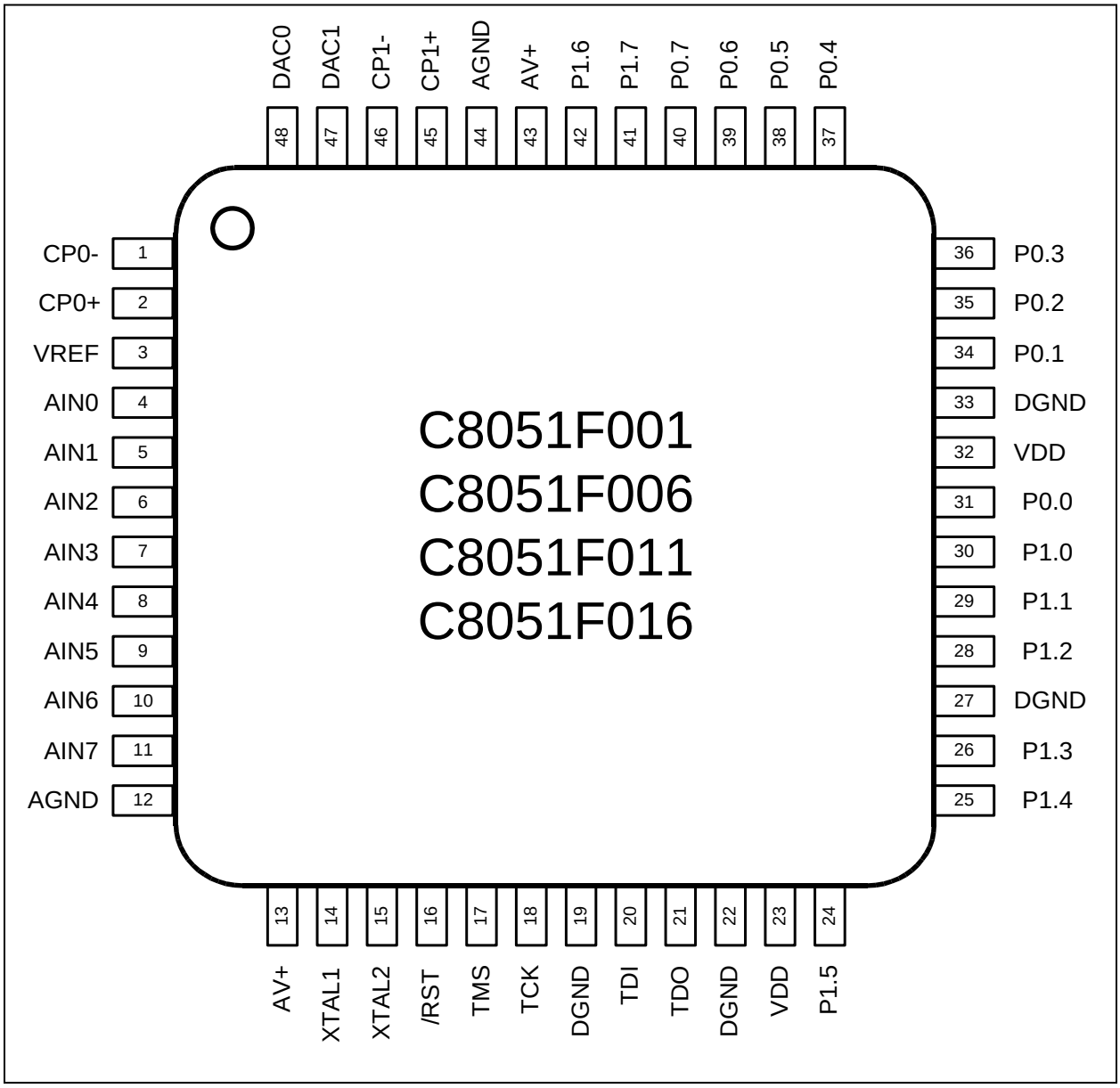
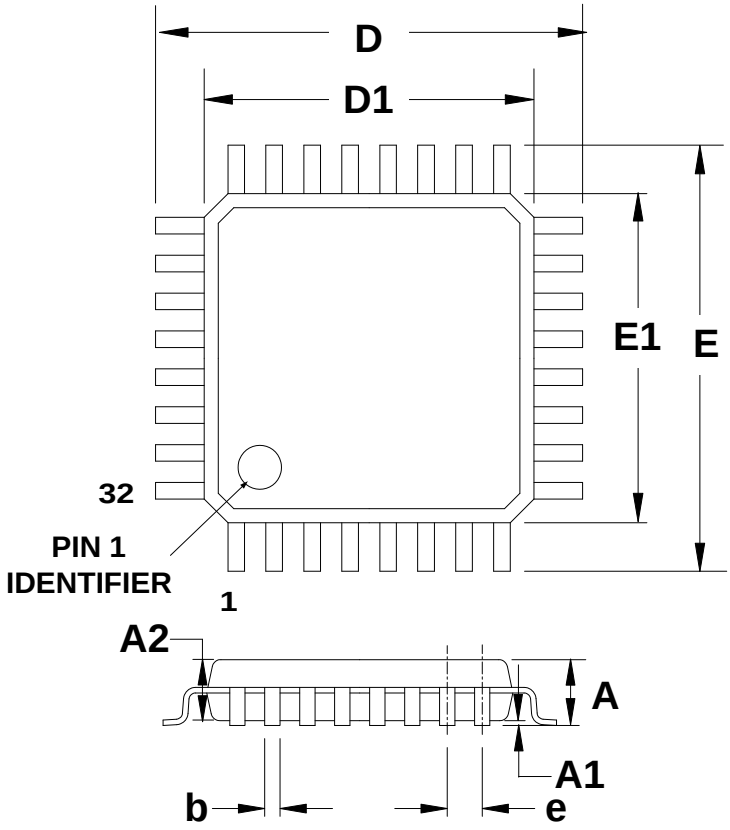


Figure 4.6. LQFP-32 Package Drawing



	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	-

5.2. ADC Modes of Operation

The ADC uses VREF to determine its full-scale voltage, thus the reference must be properly configured before performing a conversion (see Section 9). The ADC has a maximum conversion speed of 100ksps. The ADC conversion clock is derived from the system clock. Conversion clock speed can be reduced by a factor of 2, 4, 8 or 16 via 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 four 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 3 overflow (i.e. timed continuous conversions);
3. A rising edge detected on the external ADC convert start signal, CNVSTR;
4. 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. **Note: When conversions are performed "on-demand", the ADCINT flag, not ADBUSY, should be polled to determine when the conversion has completed.** Converted data is available in the ADC data word MSB and LSB registers, ADC0H, ADC0L. Converted data can be either left or right justified in the ADC0H:ADC0L register pair (see example in Figure 5.9) depending on the programmed state of the ADLJST bit in the ADC0CN register.

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 four 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 3 and lasts for 3 SAR clocks;
3. Tracking is active only when the CNVSTR input is low;
4. Tracking starts with an overflow of Timer 2 and lasts for 3 SAR clocks.

Modes 1, 2 and 4 (above) are useful when the start of conversion is triggered with a software command or when the ADC is operated continuously. Mode 3 is used when the start of conversion is triggered by external hardware. In this case, the track-and-hold is in its low power mode at times when the CNVSTR input is high. Tracking can also 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

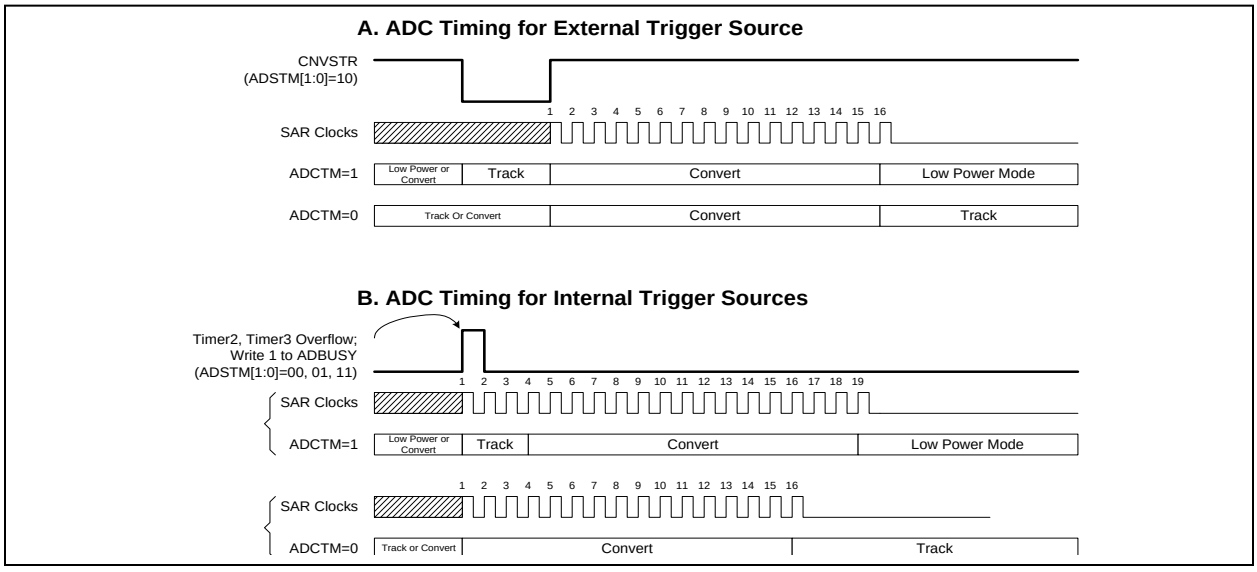


Figure 5.5. AMX0SL: AMUX Channel Select Register (C8051F00x)

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
-	-	-	-	AMXAD3	AMXAD2	AMXAD1	AMXAD0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xBB

Bits7-4: UNUSED. Read = 0000b; Write = don't care
Bits3-0: AMXAD3-0: AMUX Address Bits
0000-1111: ADC Inputs selected per chart below

		AMXAD3-0								
		0000	**0001**	**0010**	**0011**	**0100**	**0101**	**0110**	**0111**	**1xxx**
AMX0CF BITS 3-0	**0000**	AIN0	AIN1	AIN2	AIN3	AIN4	AIN5	AIN6	AIN7	TEMP SENSOR
	0001	+(AIN0) -(AIN1)		AIN2	AIN3	AIN4	AIN5	AIN6	AIN7	TEMP SENSOR
	0010	AIN0	AIN1	+(AIN2) -(AIN3)		AIN4	AIN5	AIN6	AIN7	TEMP SENSOR
	0011	+(AIN0) -(AIN1)		+(AIN2) -(AIN3)		AIN4	AIN5	AIN6	AIN7	TEMP SENSOR
	0100	AIN0	AIN1	AIN2	AIN3	+(AIN4) -(AIN5)		AIN6	AIN7	TEMP SENSOR
	0101	+(AIN0) -(AIN1)		AIN2	AIN3	+(AIN4) -(AIN5)		AIN6	AIN7	TEMP SENSOR
	0110	AIN0	AIN1	+(AIN2) -(AIN3)		+(AIN4) -(AIN5)		AIN6	AIN7	TEMP SENSOR
	0111	+(AIN0) -(AIN1)		+(AIN2) -(AIN3)		+(AIN4) -(AIN5)		AIN6	AIN7	TEMP SENSOR
	1000	AIN0	AIN1	AIN2	AIN3	AIN4	AIN5	+(AIN6) -(AIN7)		TEMP SENSOR
	1001	+(AIN0) -(AIN1)		AIN2	AIN3	AIN4	AIN5	+(AIN6) -(AIN7)		TEMP SENSOR
	1010	AIN0	AIN1	+(AIN2) -(AIN3)		AIN4	AIN5	+(AIN6) -(AIN7)		TEMP SENSOR
	1011	+(AIN0) -(AIN1)		+(AIN2) -(AIN3)		AIN4	AIN5	+(AIN6) -(AIN7)		TEMP SENSOR
	1100	AIN0	AIN1	AIN2	AIN3	+(AIN4) -(AIN5)		+(AIN6) -(AIN7)		TEMP SENSOR
	1101	+(AIN0) -(AIN1)		AIN2	AIN3	+(AIN4) -(AIN5)		+(AIN6) -(AIN7)		TEMP SENSOR
	1110	AIN0	AIN1	+(AIN2) -(AIN3)		+(AIN4) -(AIN5)		+(AIN6) -(AIN7)		TEMP SENSOR
	1111	+(AIN0) -(AIN1)		+(AIN2) -(AIN3)		+(AIN4) -(AIN5)		+(AIN6) -(AIN7)		TEMP SENSOR

Figure 5.8. ADC0H: ADC Data Word MSB Register (C8051F00x)

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 5.9. ADC0L: ADC Data Word LSB Register (C8051F00x)

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

EXAMPLE: ADC Data Word Conversion Map, AIN0-AIN1 Differential Input Pair
(AMX0CF=0x01, AMX0SL=0x00)

AIN0 – AIN1 (Volts)	**ADC0H:ADC0L** (ADLJST = 0)	**ADC0H:ADC0L** (ADLJST = 1)
REF x (2047/2048)	0x07FF	0x7FF0
0	0x0000	0x0000
-REF x (1/2048)	0xFFFF	0xFFF0
-REF	0xF800	0x8000

Figure 6.3. Temperature Sensor Transfer Function

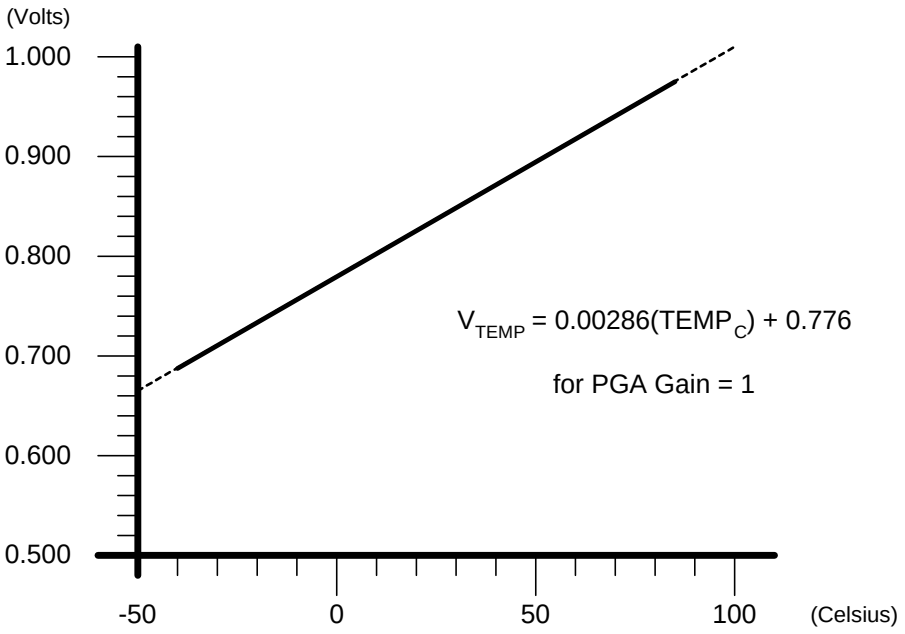


Figure 6.4. AMX0CF: AMUX Configuration Register (C8051F01x)

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
-	-	-	-	AIN67IC	AIN45IC	AIN23IC	AIN01IC	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xBA

Bits7-4: UNUSED. Read = 0000b; Write = don't care
Bit3: AIN67IC: AIN6, AIN7 Input Pair Configuration Bit
0: AIN6 and AIN7 are independent singled-ended inputs
1: AIN6, AIN7 are (respectively) +, - differential input pair
Bit2: AIN45IC: AIN4, AIN5 Input Pair Configuration Bit
0: AIN4 and AIN5 are independent singled-ended inputs
1: AIN4, AIN5 are (respectively) +, - differential input pair
Bit1: AIN23IC: AIN2, AIN3 Input Pair Configuration Bit
0: AIN2 and AIN3 are independent singled-ended inputs
1: AIN2, AIN3 are (respectively) +, - differential input pair
Bit0: AIN01IC: AIN0, AIN1 Input Pair Configuration Bit
0: AIN0 and AIN1 are independent singled-ended inputs
1: AIN0, AIN1 are (respectively) +, - differential input pair

NOTE: The ADC Data Word is in 2's complement format for channels configured as differential.

8. COMPARATORS

The MCU family has two on-chip analog 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 the package pins via the I/O crossbar (see Section 15.1). When assigned to package pins, each comparator output can be programmed to operate in open drain or push-pull modes (see section 15.3).

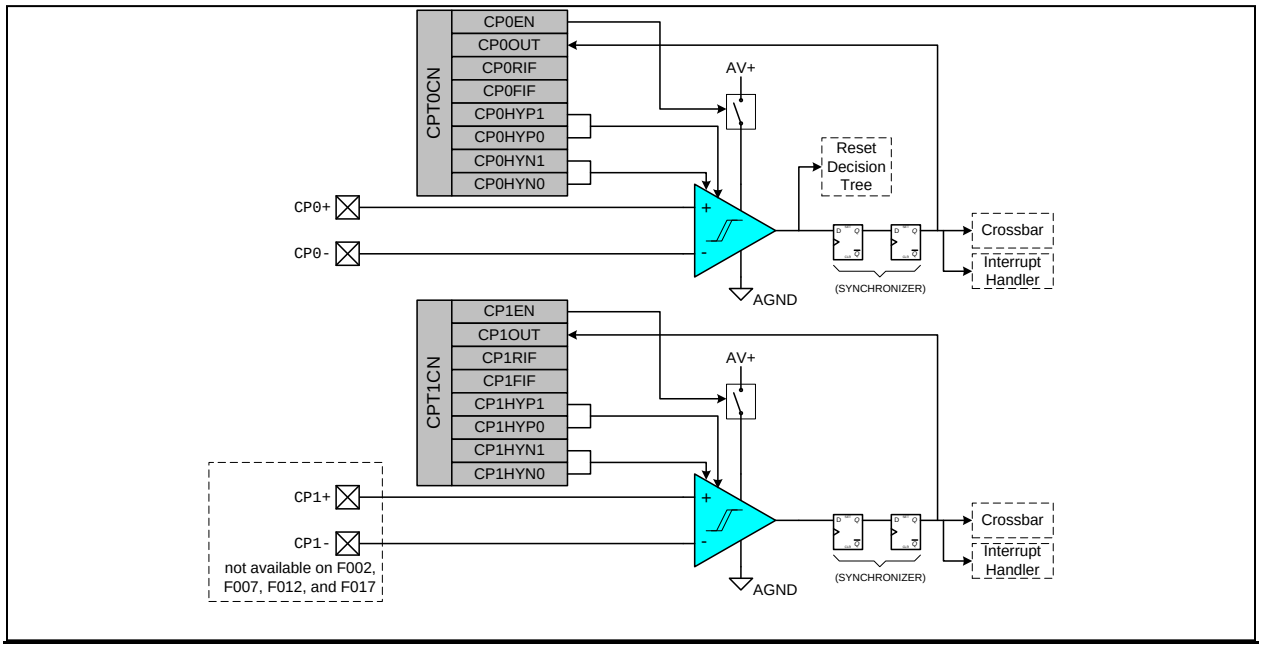
The hysteresis of each comparator is software-programmable via its respective Comparator control register (CPT0CN, CPT1CN). The user can program both the amount of hysteresis voltage (referred to the input voltage) and the positive and negative-going symmetry of this hysteresis around the threshold voltage. 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 Crossbar) defaults to the logic low state, its interrupt capability is suspended and its supply current falls to less than 1μA. Comparator 0 inputs can be externally driven from -0.25V to (AV+) + 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 10.4). 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 CPU. The Output State of Comparator 0 can be obtained at any time by reading the CP0OUT bit. Note the comparator output and interrupt should be ignored until the comparator settles after power-up. Comparator 0 is enabled by setting the CP0EN bit, and is disabled by clearing this bit. Note there is a 20usec settling time for the comparator output to stabilize after setting the CP0EN bit or a power-up. Comparator 0 can also be programmed as a reset source. For details, see Section 13.

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). Comparator 1 can not be programmed as a reset source. Also, the input pins for Comparator 1 are not pinned out on the F002, F007, F012, or F017 devices. The complete electrical specifications for the Comparators are given in Table 8.1.

Figure 8.1. Comparator Functional Block Diagram



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

* Refers to a register in the C8051F000/1/2/5/6/7 only.
** Refers to a register in the C8051F010/1/2/5/6/7 only.
*** Refers to a register in the C8051F005/06/07/15/16/17 only.

Figure 10.7. 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 (bit addressable)	SFR Address: 0xE0

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

Figure 10.8. 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 (bit addressable)	SFR Address: 0xF0

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

Figure 10.14. EIP2: Extended Interrupt Priority 2

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

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

Bit6: Reserved: Must write 0. Reads 0.

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

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

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

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

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

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

11.2. 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 0x8000 – 0x807F. 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.)

11.3. 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 0x7DFE and 0x7DFF protect the Flash program memory from being read or altered across the JTAG interface. Each bit in a security lock-byte protects one 4kbyte 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 0x7DFF. The Write/Erase lock byte is located at 0x7DFE. Figure 11.2 shows the location and bit definitions of the security bytes. The 512-byte sector containing the lock bytes can be written to, but not erased by software. Writing to the reserved area should not be performed.

Figure 11.1. 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 also set. 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.

16.6.5. Status Register

The SMB0STA Status register holds an 8-bit status code indicating the current state of the SMBus. There are 28 possible SMBus states, each with a corresponding unique status code. The five most significant bits of the status code vary while the three least-significant bits of a valid status code are fixed at zero when SI = 1. Therefore, all possible status codes are multiples of eight. This facilitates the use of status codes in software as an index used to branch to appropriate service routines (allowing 8 bytes of code to service the state or jump to a more extensive service routine).

For the purposes of user software, the contents of the SMB0STA register is only defined when the SI flag is logic 1. Software should never write to the SMB0STA register. Doing so will yield indeterminate results. The 28 SMBus states, along with their corresponding status codes, are given in Table 16.1.

Figure 16.8. SMB0STA: SMBus Status Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
STA7	STA6	STA5	STA4	STA3	STA2	STA1	STA0	11111000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xC1

Bits7-3: STA7-STA3: SMBus Status Code.
These bits contain the SMBus Status Code. There are 28 possible status codes. Each status code corresponds to a single SMBus state. A valid status code is present in SMB0STA when the SI flag (SMB0CN.3) is set. The content of SMB0STA is not defined when the SI flag is logic 0. Writing to the SMB0STA register at any time will yield indeterminate results.

Bits2-0: STA2-STA0: The three least significant bits of SMB0STA are always read as logic 0 when the SI flag is logic 1.

Figure 18.9. 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 (bit addressable)	SFR Address: 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.

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.

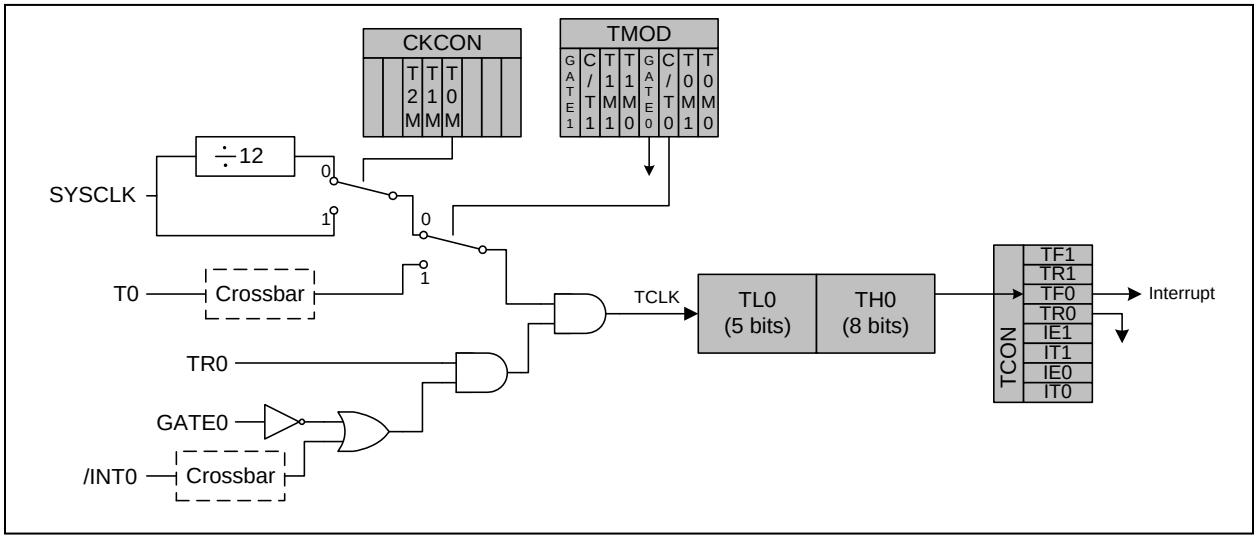
TR0	GATE0	/INT0	Counter/Timer
0	X	X	Disabled
1	0	X	Enabled
1	1	0	Disabled
1	1	1	Enabled

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 19.1. T0 Mode 0 Block Diagram



19.1.2. 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.

20.2. PCA Counter/Timer

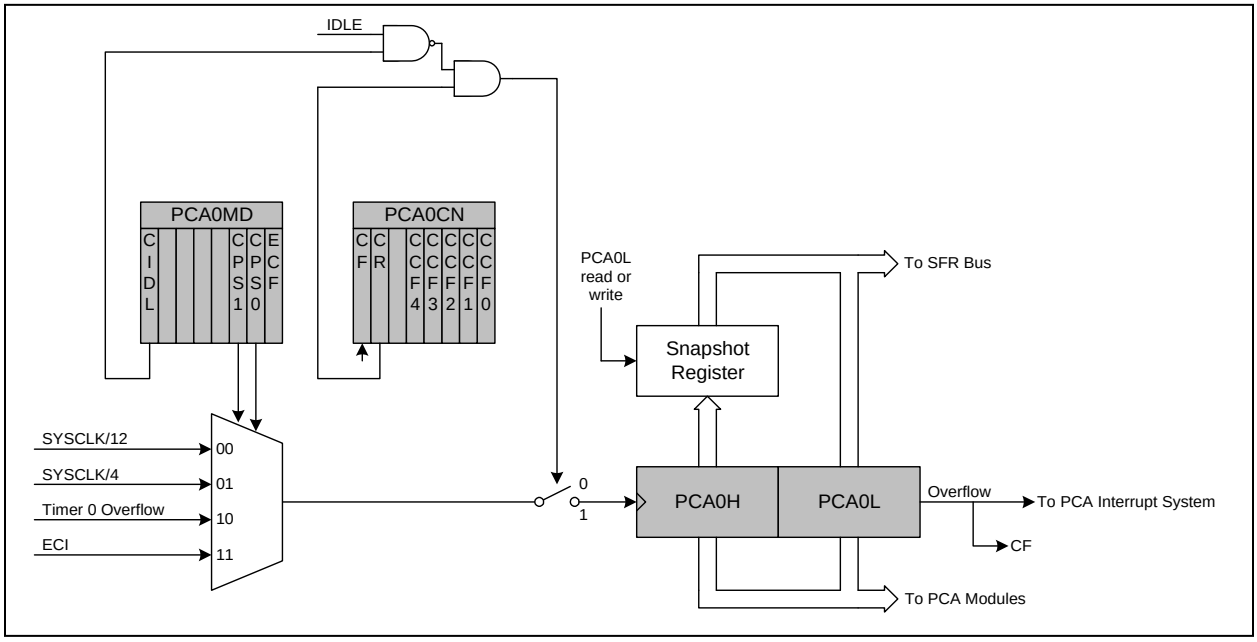
The 16-bit PCA counter/timer consists of two 8-bit SFRs: PCA0L and PCA0H. PCA0H is the high byte (MSB) of the 16-bit counter/timer and PCA0L is the low byte (LSB). Reading PCA0L automatically latches the value of PCA0H at the same time. By reading the PCA0L Register first, this allows the PCA0H value to be held (at the time PCA0L was read) until the user reads the PCA0H Register. Reading PCA0H or PCA0L does not disturb the counter operation. The CPS1 and CPS0 bits in the PCA0MD register select the timebase for the counter/timer as shown in Table 20.2.

When the counter/timer overflows from 0xFFFF to 0x0000, the Counter Overflow Flag (CF) in PCA0MD is set to logic 1 and an interrupt request is generated if CF interrupts are enabled. Setting the ECF bit in PCA0MD to logic 1 enables the CF flag to generate an interrupt request. The CF bit is not automatically cleared by hardware when the CPU vectors to the interrupt service routine, and must be cleared by software. (Note: PCA0 interrupts must be globally enabled before CF interrupts are recognized. PCA0 interrupts are globally enabled by setting the EA bit (IE.7) and the EPCA0 bit in EIE1 to logic 1.) Clearing the CIDL bit in the PCA0MD register allows the PCA to continue normal operation while the microcontroller core is in Idle mode.

Table 20.2. PCA Timebase Input Options

CPS1	CPS0	Timebase
0	0	System clock divided by 12
0	1	System clock divided by 4
1	0	Timer 0 overflow
1	1	High-to-low transitions on ECI (max rate = system clock divided by 4)

Figure 20.7. PCA Counter/Timer Block Diagram



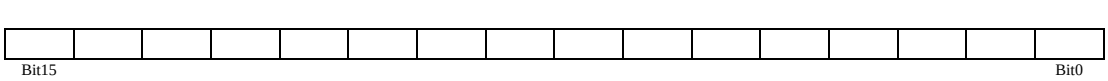
21. JTAG (IEEE 1149.1)

Each MCU has an on-chip JTAG interface and logic to support boundary scan for production and in-system testing, Flash read and write operations, and non-intrusive in-circuit debug. The JTAG interface is fully compliant with the IEEE 1149.1 specification. Refer to this specification for detailed descriptions of the Test Interface and Boundary-Scan Architecture. Access of the JTAG Instruction Register (IR) and Data Registers (DR) are as described in the Test Access Port and Operation of the IEEE 1149.1 specification.

The JTAG interface is via four dedicated pins on the MCU, which are TCK, TMS, TDI, and TDO. These pins are all 5V tolerant.

Through the 16-bit JTAG Instruction Register (IR), any of the eight instructions shown in Figure 21.1 can be commanded. There are three Data Registers (DR's) associated with JTAG Boundary-Scan, and four associated with Flash read/write operations on the MCU.

Figure 21.1. IR: JTAG Instruction Register



IR value	Instruction	Description
0x0000	EXTEST	Selects the Boundary Data Register for control and observability of all device pins
0x0002	SAMPLE/ PRELOAD	Selects the Boundary Data Register for observability and presetting the scan-path latches
0x0004	IDCODE	Selects device ID Register
0xFFFF	BYPASS	Selects Bypass Data Register
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