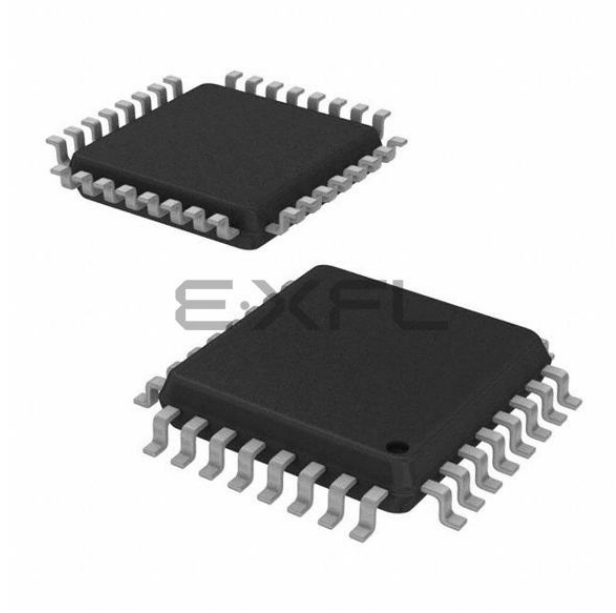


Welcome to E-XFL.COM

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	20MHz
Connectivity	SMBus (2-Wire/I²C), SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, Temp Sensor, WDT
Number of I/O	8
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 4x10b; D/A 2x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	32-LQFP
Supplier Device Package	32-LQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f012-gqr

4. PINOUT AND PACKAGE DEFINITIONS

Table 4.1. Pin Definitions

Name	Pin Numbers			Type	Description
	F000 F005 F010 F015	F001 F006 F011 F016	F002 F007 F012 F017		
VDD	31, 40, 62	23, 32	18, 20		Digital Voltage Supply.
DGND	30, 41, 61	22, 33, 27, 19	17, 21		Digital Ground.
AV+	16, 17	13, 43	9, 29		Positive Analog Voltage Supply.
AGND	5, 15	44, 12	8, 30		Analog Ground.
TCK	22	18	14	D In	JTAG Test Clock with internal pull-up.
TMS	21	17	13	D In	JTAG Test-Mode Select with internal pull-up.
TDI	28	20	15	D In	JTAG Test Data Input with internal pull-up. TDI is latched on a rising edge of TCK.
TDO	29	21	16	D Out	JTAG Test Data Output with internal pull-up. Data is shifted out on TDO on the falling edge of TCK. TDO output is a tri-state driver.
XTAL1	18	14	10	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	19	15	11	A Out	Crystal Output. This pin is the excitation driver for a crystal or ceramic resonator.
/RST	20	16	12	D I/O	Chip Reset. Open-drain output of internal Voltage Supply monitor. Is driven low when VDD is < 2.7V. An external source can force a system reset by driving this pin low.
VREF	6	3	3	A I/O	Voltage Reference. When configured as an input, this pin is the voltage reference for the MCU. Otherwise, the internal reference drives this pin.
CP0+	4	2	2	A In	Comparator 0 Non-Inverting Input.
CP0-	3	1	1	A In	Comparator 0 Inverting Input.
CP1+	2	45		A In	Comparator 1 Non-Inverting Input.
CP1-	1	46		A In	Comparator 1 Inverting Input.
DAC0	64	48	32	A Out	Digital to Analog Converter Output 0. The DAC0 voltage output. (See Section 7 DAC Specification for complete description).
DAC1	63	47	31	A Out	Digital to Analog Converter Output 1. The DAC1 voltage output. (See Section 7 DAC Specification for complete description).
AIN0	7	4	4	A In	Analog Mux Channel Input 0. (See ADC Specification for complete description).
AIN1	8	5	5	A In	Analog Mux Channel Input 1. (See ADC Specification for complete description).
AIN2	9	6	6	A In	Analog Mux Channel Input 2. (See ADC Specification for complete description).
AIN3	10	7	7	A In	Analog Mux Channel Input 3. (See ADC Specification for complete description).
AIN4	11	8		A In	Analog Mux Channel Input 4. (See ADC Specification for complete description).
AIN5	12	9		A In	Analog Mux Channel Input 5. (See ADC Specification for complete description).

Figure 4.3. TQFP-48 Pinout Diagram

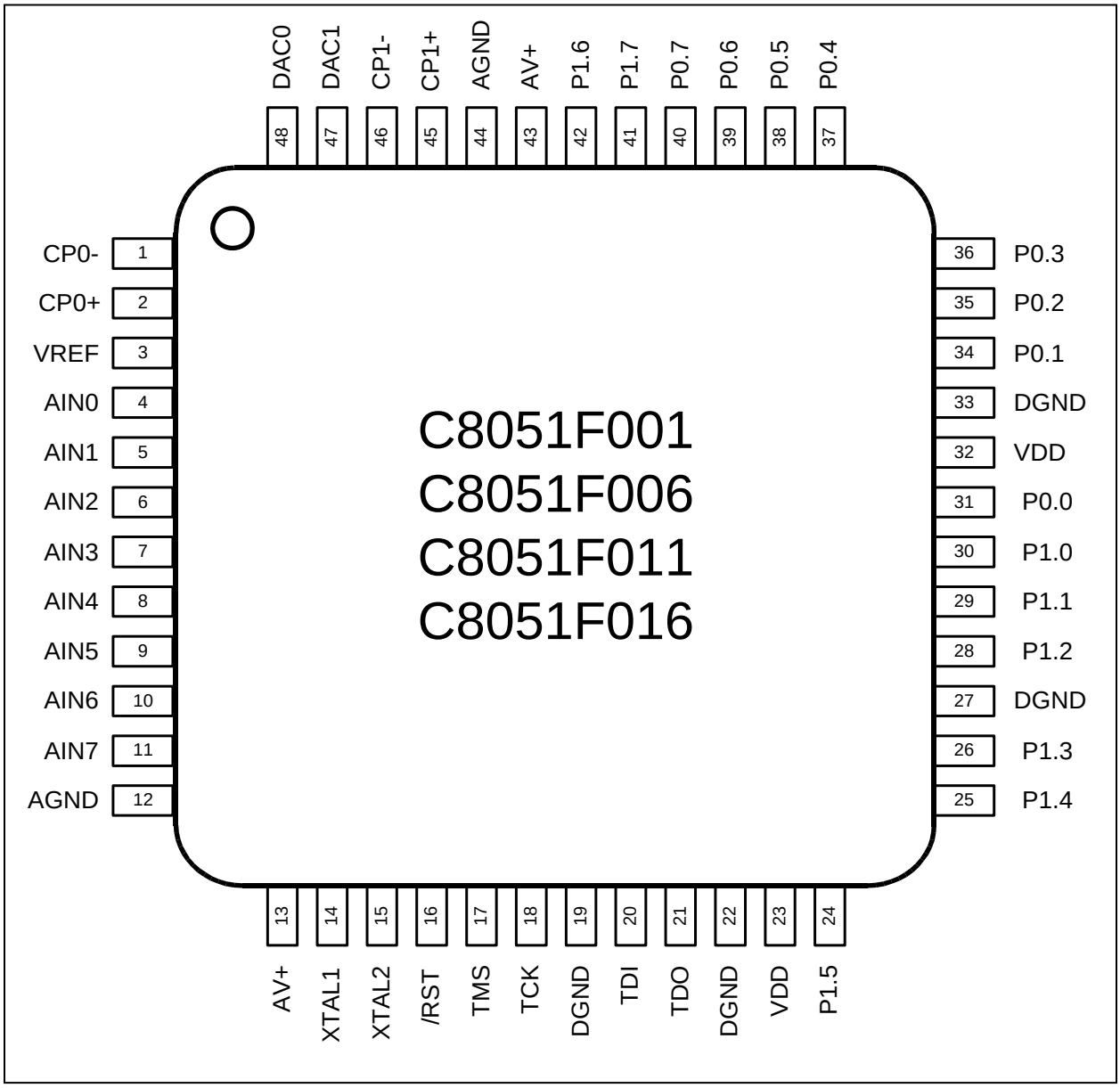


Figure 5.6. ADC0CF: ADC Configuration Register (C8051F00x)

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: the SAR Conversion Clock should be ≤ 2MHz)
Bits4-3: UNUSED. Read = 00b; Write = don't care
Bits2-0: AMPGN2-0: ADC Internal Amplifier Gain
000: Gain = 1
001: Gain = 2
010: Gain = 4
011: Gain = 8
10x: Gain = 16
11x: Gain = 0.5

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, ADC0GTL, ADC0LTH, and ADC0LTL). Figure 5.14 and Figure 5.15 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 5.10. ADC0GTH: ADC Greater-Than Data High Byte Register (C8051F00x)

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 5.11. ADC0GTL: ADC Greater-Than Data Low Byte Register (C8051F00x)

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 5.12. ADC0LTH: ADC Less-Than Data High Byte 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: 0xC7

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

Figure 5.13. ADC0LTL: ADC Less-Than Data Low Byte 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: 0xC6

Bits7-0:
These bits are the low byte of the ADC Less-Than Data Word.
Definition:
ADC Less-Than Data Word = ADC0LTH:ADC0LTL

7. DACs, 12 BIT VOLTAGE MODE

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

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

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

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

Figure 7.1. DAC Functional Block Diagram

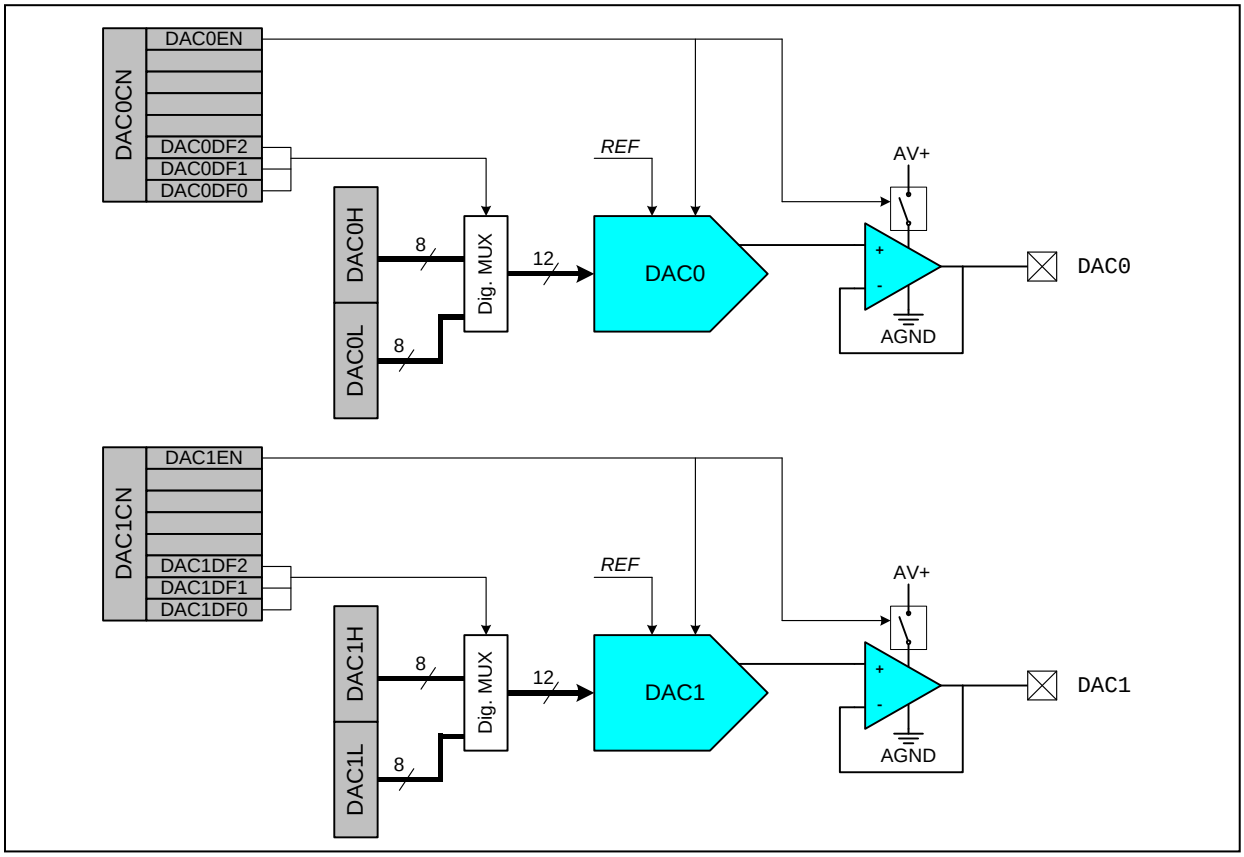


Figure 7.5. DAC1H: DAC1 High Byte Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xD6

Bits7-0: DAC1 Data Word Most Significant Byte.

Figure 7.6. DAC1L: DAC1 Low Byte Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xD5

Bits7-0: DAC1 Data Word Least Significant Byte.

Figure 7.7. DAC1CN: DAC1 Control Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
DAC1EN	-	-	-	-	DAC1DF2	DAC1DF1	DAC1DF0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xD7

Bit7: DAC1EN: DAC1 Enable Bit
0: DAC1 Disabled. DAC1 Output pin is disabled; DAC1 is in low power shutdown mode.
1: DAC1 Enabled. DAC1 Output is pin active; DAC1 is operational.
Bits6-3: UNUSED. Read = 0000b; Write = don't care
Bits2-0: DAC1DF2-0: DAC1 Data Format Bits
000: The most significant nybble of the DAC1 Data Word is in DAC1H[3:0], while the least significant byte is in DAC1L.

DAC1H								DAC1L							
				MSB											LSB

001: The most significant 5-bits of the DAC1 Data Word is in DAC1H[4:0], while the least significant 7-bits is in DAC1L[7:1].

DAC1H								DAC1L							
			MSB											LSB	

010: The most significant 6-bits of the DAC1 Data Word is in DAC1H[5:0], while the least significant 6-bits is in DAC1L[7:2].

DAC1H								DAC1L							
		MSB											LSB		

011: The most significant 7-bits of the DAC1 Data Word is in DAC1H[6:0], while the least significant 5-bits is in DAC1L[7:3].

DAC1H								DAC1L							
	MSB											LSB			

1xx: The most significant byte of the DAC1 Data Word is in DAC1H, while the least significant nybble is in DAC1L[7:4].

DAC1H								DAC1L							
MSB											LSB				

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

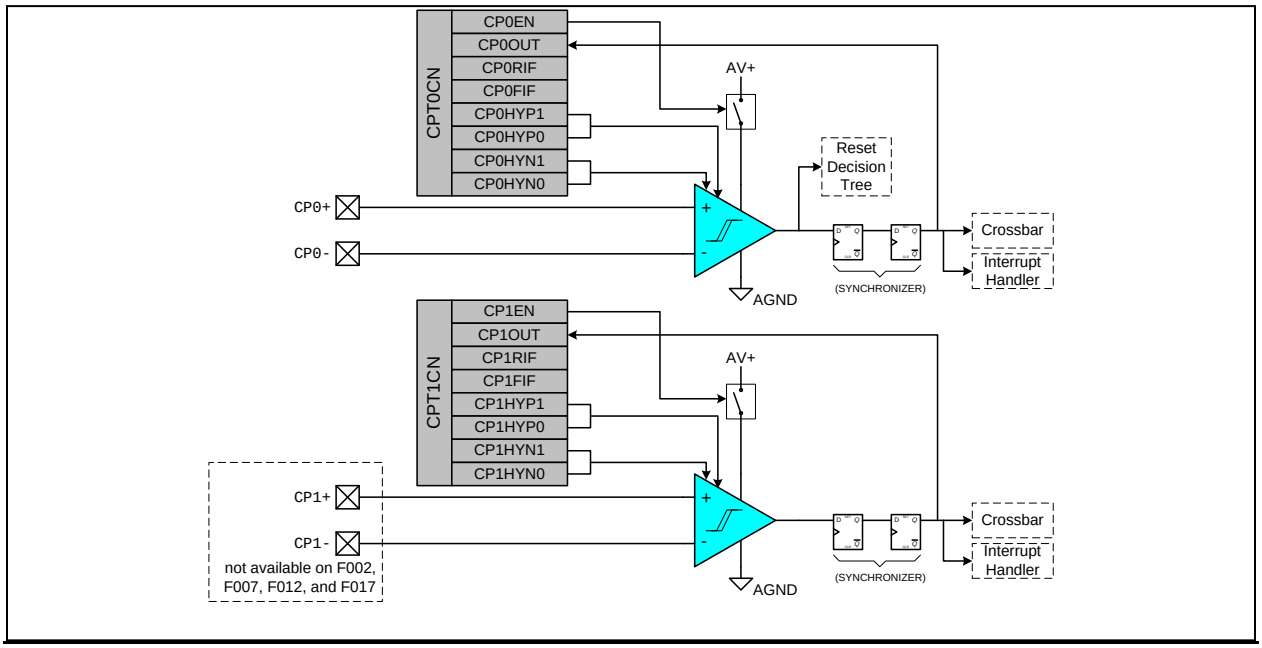


Figure 8.3. CPT0CN: Comparator 0 Control Register

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

Table 10.1. CIP-51 Instruction Set Summary

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

Figure 10.6. 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 (bit addressable)	SFR Address: 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

Note: Any instruction which changes the RS1-RS0 bits must not be immediately followed by the “MOV Rn, A” instruction.

Bit2: OV: Overflow Flag.
This bit is set to 1 under the following circumstances:

- An ADD, ADDC, or SUBB instruction causes a sign-change overflow.
- A MUL instruction results in an overflow (result is greater than 255) .
- A DIV instruction causes a divide-by-zero condition.

The OV bit is cleared to 0 by the ADD, ADDC, SUBB, MUL, and DIV instructions in all other cases.

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

Bit0: PARITY: Parity Flag.
(Read only)
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 10.13. EIP1: Extended Interrupt Priority 1

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

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

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

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

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

Bit3: PPCA0: Programmable Counter Array (PCA0) Interrupt Priority Control.
This bit sets the priority of the PCA0 interrupt.
0: PCA0 interrupt set to low priority level.
1: PCA0 interrupt set to high priority level.

Bit2: PWADC0: ADC0 Window Comparator Interrupt Priority Control.
This bit sets the priority of the ADC0 Window interrupt.
0: ADC0 Window interrupt set to low priority level.
1: ADC0 Window interrupt set to high priority level.

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

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

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.

13.8.1. Watchdog Usage

The WDT consists of a 21-bit timer running from the programmed system clock. The timer measures the period between specific writes to its control register. If this period exceeds the programmed limit, a WDT reset is generated. The WDT can be enabled and disabled as needed in software, or can be permanently enabled if desired. Watchdog features are controlled via the Watchdog Timer Control Register (WDTCN) shown in Figure 13.3.

Enable/Reset WDT

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

Disable WDT

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

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

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

Disable WDT Lockout

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

Setting WDT Interval

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

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

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

Figure 13.3. WDTCN: Watchdog Timer Control Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								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.

not affect the push-pull Port I/O. Furthermore, the weak pullup is turned off on an open-drain output that is driving a 0 to avoid unnecessary power dissipation.

The third and final step is to initialize 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.

Figure 15.1. Port I/O Functional Block Diagram

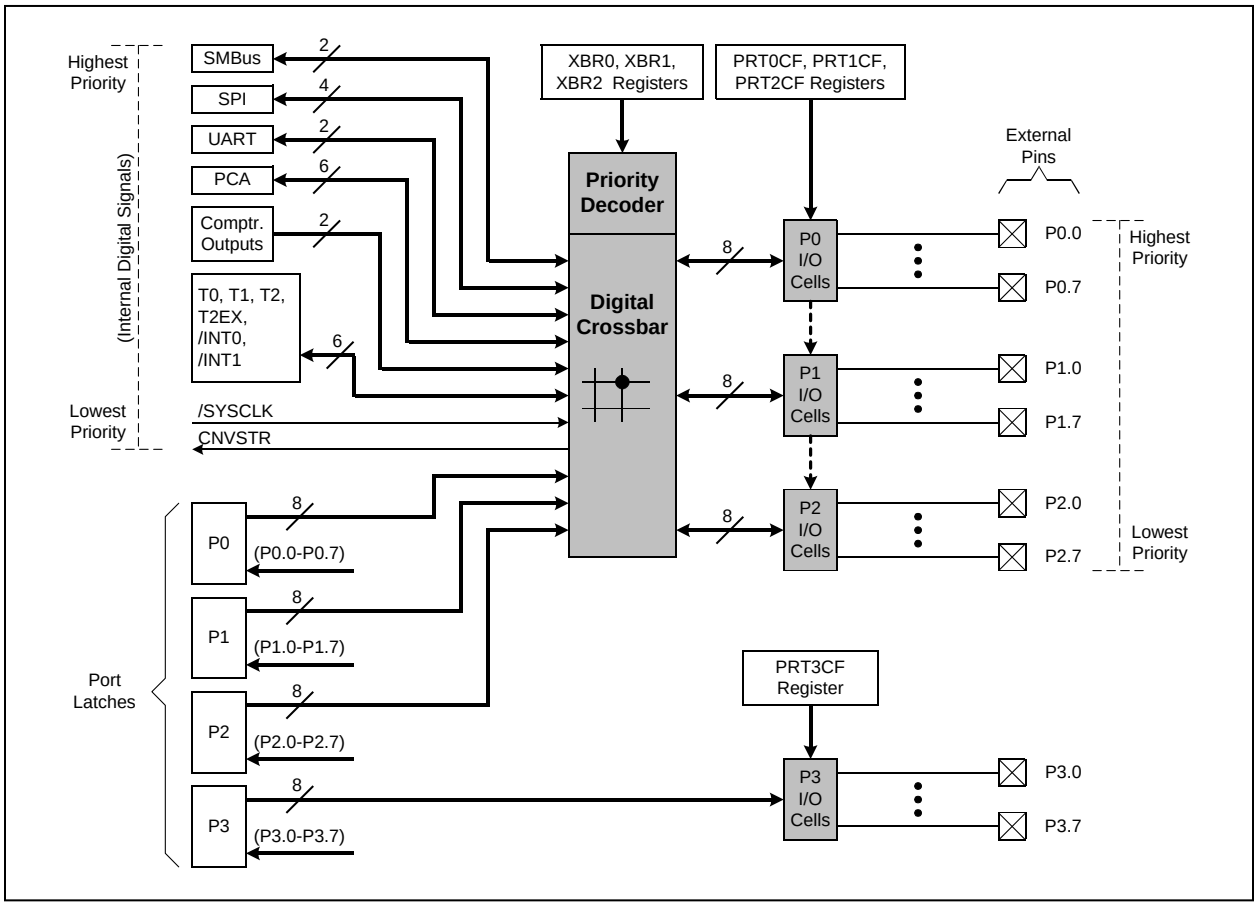


Figure 15.2. Port I/O Cell Block Diagram

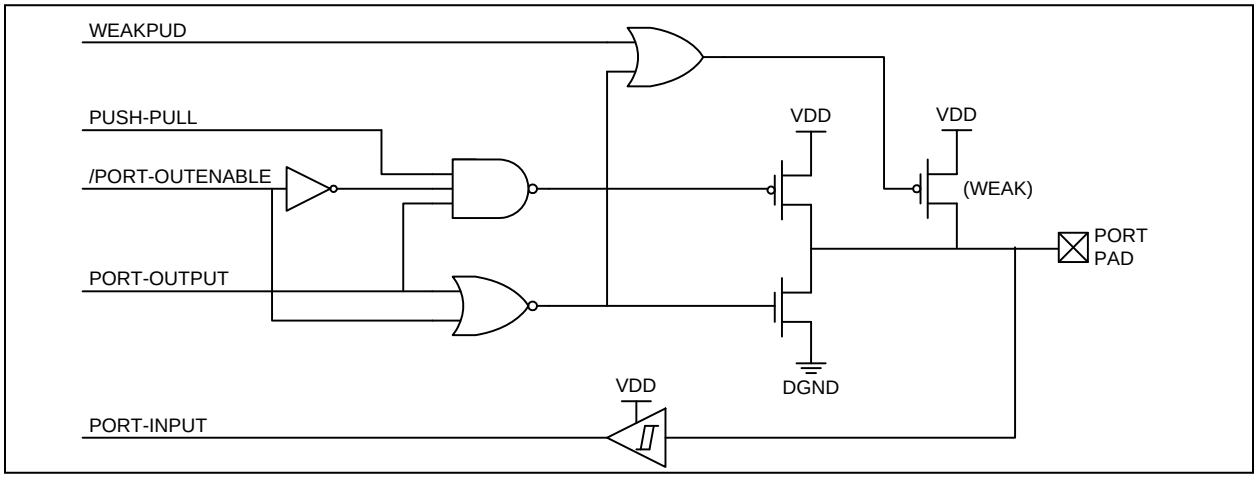
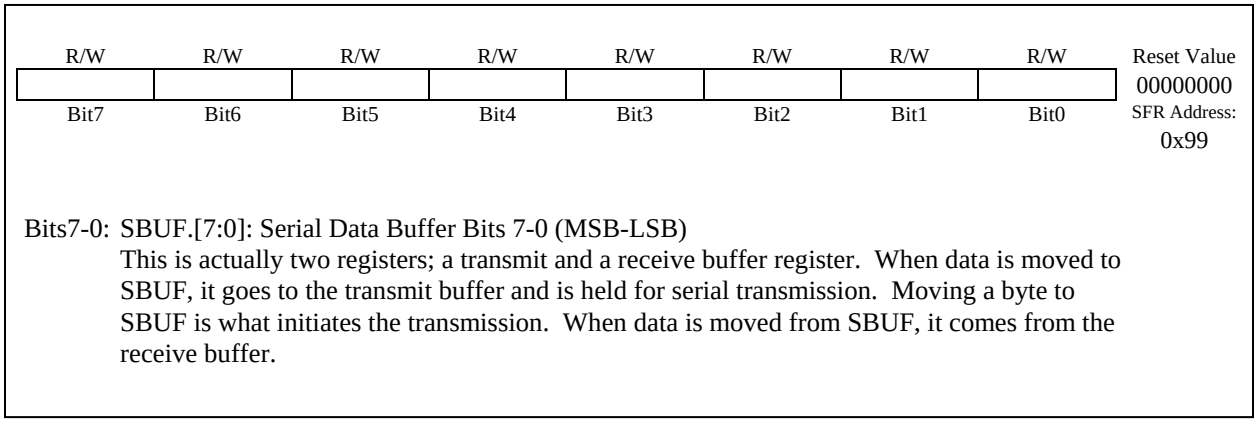


Table 18.2. Oscillator Frequencies for Standard Baud Rates

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	384	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 and T1M=1.
** Numbers in parenthesis show the actual baud rate.

Figure 18.8. SBUF: Serial (UART) Data Buffer Register

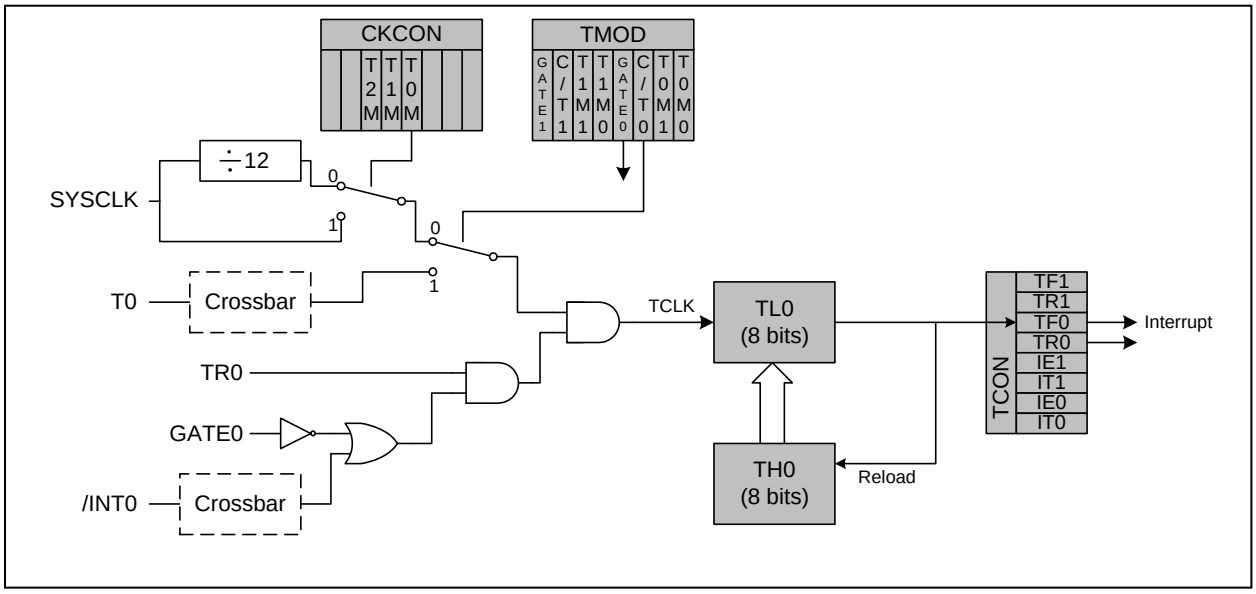


Bits7-0: SBUF.[7:0]: Serial Data Buffer Bits 7-0 (MSB-LSB)
This is actually two registers; a transmit and a receive buffer register. When data is moved to SBUF, it goes to the transmit buffer and is held for serial transmission. Moving a byte to SBUF is what initiates the transmission. When data is moved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

19.1.3. Mode 2: 8-bit Counter/Timer with Auto-Reload

Mode 2 configures Timer 0 and Timer 1 to operate as 8-bit counter/timers with automatic reload of the start value. The TL0 holds the count and TH0 holds the reload value. When the count in TL0 overflows from all ones to 0x00, the timer overflow flag TF0 (TCON.5) is set and the counter in TL0 is reloaded from TH0. If enabled, an interrupt will occur when the TF0 flag is set. The reload value in TH0 is not changed. TL0 must be initialized to the desired value before enabling the timer for the first count to be correct. When in Mode 2, Timer 1 operates identically to Timer 0. Both counter/timers are enabled and configured in Mode 2 in the same manner as Mode 0.

Figure 19.2. T0 Mode 2 Block Diagram



19.3. Timer 3

Timer 3 is a 16-bit timer formed by the two 8-bit SFRs, TMR3L (low byte) and TMR3H (high byte). The input for Timer 3 is the system clock (divided by either one or twelve as specified by the Timer 3 Clock Select bit T3M in the Timer 3 Control Register TMR3CN). Timer 3 is always configured as an auto-reload timer, with the reload value held in the TMR3RLL (low byte) and TMR3RLH (high byte) registers. Timer 3 can be used to start an ADC Data Conversion, for SMBus timing (see Section 16.5), or as a general-purpose timer. Timer 3 does not have a counter mode.

Figure 19.19. Timer 3 Block Diagram

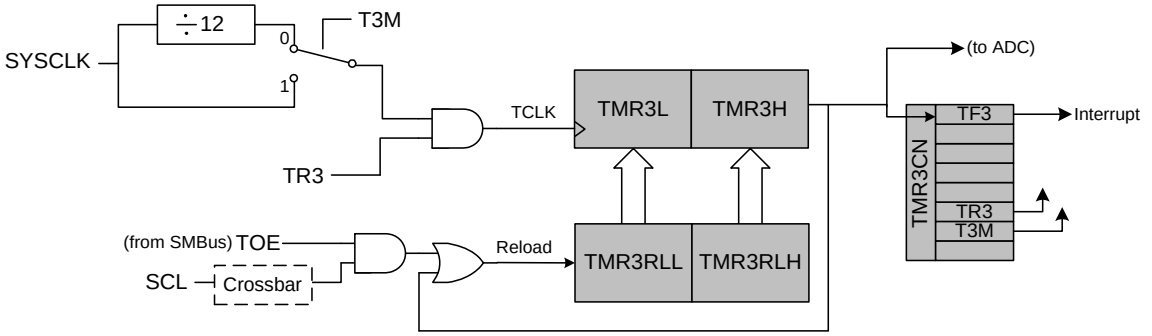


Figure 19.20. TMR3CN: Timer 3 Control Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
TF3	-	-	-	-	TR3	T3M	-	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0x91

Bit7: TF3: Timer3 Overflow Flag.
Set by hardware when Timer 3 overflows from 0xFFFF to 0x0000. When the Timer 3 interrupt is enabled, setting this bit causes the CPU to vector to the Timer 3 Interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software.

Bits6-3: UNUSED. Read = 0000b, Write = don't care.

Bit2: TR3: Timer 3 Run Control.
This bit enables/disables Timer 3.
0: Timer 3 disabled.
1: Timer 3 enabled.

Bit1: T3M: Timer 3 Clock Select.
This bit controls the division of the system clock supplied to Counter/Timer 3.
0: Counter/Timer 3 uses the system clock divided by 12.
1: Counter/Timer 3 uses the system clock.

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

Figure 21.5. FLASHDAT: JTAG Flash Data Register

DATA7	DATA6	DATA5	DATA4	DATA3	DATA2	DATA1	DATA0	FAIL	FBUSY	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: FBUSY: Flash Busy Bit.
- 0: Flash interface logic is not busy.
- 1: Flash interface logic is processing a request. Reads or writes while FBUSY = 1 will not initiate another operation

Figure 21.6. 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 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.

21.3. Debug Support

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

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