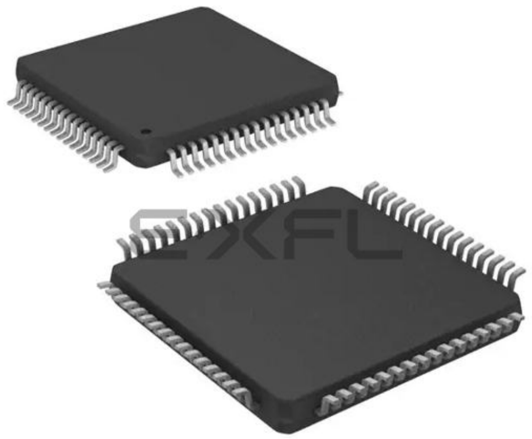


Welcome to E-XFL.COM

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	Obsolete
Core Processor	8051
Core Size	8-Bit
Speed	25MHz
Connectivity	EBI/EMI, SMBus (2-Wire/I²C), SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, Temp Sensor, WDT
Number of I/O	32
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4.25K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 8x8b, 8x12b; D/A 2x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-TQFP
Supplier Device Package	64-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f021r

1.3. JTAG Debug and Boundary Scan

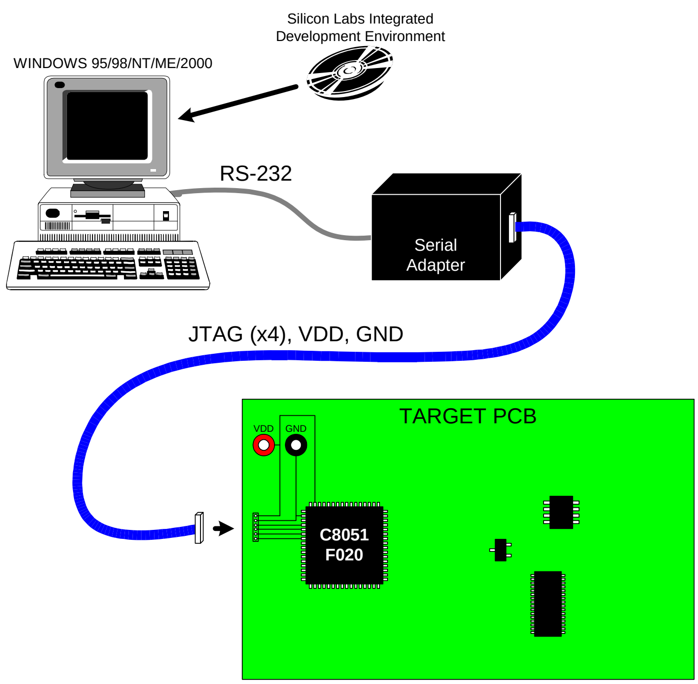
The C8051F020 family has on-chip JTAG boundary scan and debug circuitry that provides *non-intrusive, full speed, in-circuit debugging using the production part installed in the end application*, via the four-pin JTAG interface. The JTAG port is fully compliant to IEEE 1149.1, providing full boundary scan for test and manufacturing purposes.

Silicon Labs' debugging system supports inspection and modification of memory and registers, breakpoints, watchpoints, a stack monitor, and single stepping. No additional target RAM, program memory, timers, or communications channels are required. All the digital and analog peripherals are functional and work correctly while debugging. All the peripherals (except for the ADC and SMBus) are stalled when the MCU is halted, during single stepping, or at a breakpoint in order to keep them synchronized.

The C8051F020DK development kit provides all the hardware and software necessary to develop application code and perform in-circuit debugging with the C8051F020/1/2/3 MCUs. The kit includes software with a developer's studio and debugger, an integrated 8051 assembler, and an RS-232 to JTAG serial adapter. It also has a target application board with the associated MCU installed, plus the RS-232 and JTAG cables, and wall-mount power supply. The Development Kit requires a Windows 95/98/NT/ME/2000 computer with one available RS-232 serial port. As shown in Figure 1.8, the PC is connected via RS-232 to the Serial Adapter. A six-inch ribbon cable connects the Serial Adapter to the user's application board, picking up the four JTAG pins and VDD and GND. The Serial Adapter takes its power from the application board; it requires roughly 20 mA at 2.7-3.6 V. For applications where there is not sufficient power available from the target system, the provided power supply can be connected directly to the Serial Adapter.

Silicon Labs' debug environment 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. Silicon Labs' debug environment both increases ease of use and preserves the performance of the precision analog peripherals.

Figure 1.8. Development/In-System Debug Diagram



1.8. 8-Bit Analog to Digital Converter

The C8051F020/1/2/3 has an on-board 8-bit SAR ADC (ADC1) with an 8-channel input multiplexer and programmable gain amplifier. This ADC features a 500 ksps maximum throughput and true 8-bit accuracy with an INL of ±1LSB. Eight input pins are available for measurement. The ADC is under full control of the CIP-51 microcontroller via the Special Function Registers. The ADC1 voltage reference is selected between the analog power supply (AV+) and an external VREF pin. On C8051F020/2 devices, ADC1 has its own dedicated VREF1 input pin; on C8051F021/3 devices, ADC1 shares the VREFA input pin with the 12/10-bit ADC0. User software may put ADC1 into shutdown mode to save power.

A programmable gain amplifier follows the analog multiplexer. The gain stage can be especially useful when different ADC input channels have widely varied input voltage signals, or when it is necessary to "zoom in" on a signal with a large DC offset (in differential mode, a DAC could be used to provide the DC offset). The PGA gain can be set in software to 0.5, 1, 2, or 4.

A flexible conversion scheduling system allows ADC1 conversions to be initiated by software commands, timer overflows, or an external input signal. ADC1 conversions may also be synchronized with ADC0 software-commanded conversions. Conversion completions are indicated by a status bit and an interrupt (if enabled), and the resulting 8-bit data word is latched into an SFR upon completion.

Figure 1.12. 8-Bit ADC Diagram

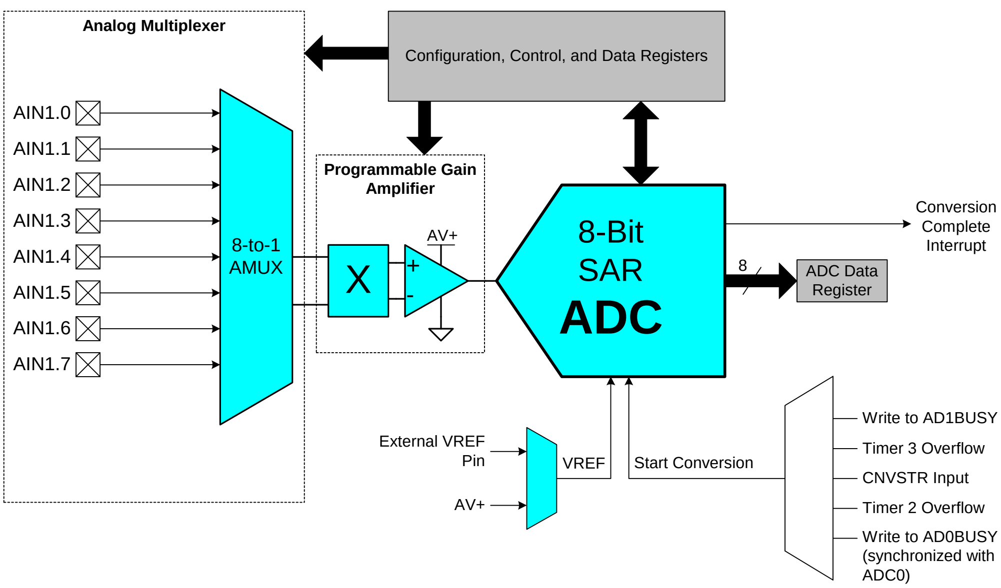


Figure 5.8. ADC0CN: ADC0 Control Register (C8051F020/1)

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
AD0EN	AD0TM	AD0INT	AD0BUSY	AD0CM1	AD0CM0	AD0WINT	AD0LJST	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
							(bit addressable)	0xE8

Bit7: AD0EN: ADC0 Enable Bit.
0: ADC0 Disabled. ADC0 is in low-power shutdown.
1: ADC0 Enabled. ADC0 is active and ready for data conversions.

Bit6: AD0TM: 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

Bit5: AD0INT: ADC0 Conversion Complete Interrupt Flag.
This flag must be cleared by software.
0: ADC0 has not completed a data conversion since the last time this flag was cleared.
1: ADC0 has completed a data conversion.

Bit4: AD0BUSY: ADC0 Busy Bit.
Read:
0: ADC0 Conversion is complete or a conversion is not currently in progress. AD0INT is set to logic 1 on the falling edge of AD0BUSY.
1: ADC0 Conversion is in progress.
Write:
0: No Effect.
1: Initiates ADC0 Conversion if AD0STM1-0 = 00b

Bit3-2: AD0CM1-0: ADC0 Start of Conversion Mode Select.
If AD0TM = 0:
00: ADC0 conversion initiated on every write of ‘1’ to AD0BUSY.
01: ADC0 conversion initiated on overflow of Timer 3.
10: ADC0 conversion initiated on rising edge of external CNVSTR.
11: ADC0 conversion initiated on overflow of Timer 2.
If AD0TM = 1:
00: Tracking starts with the write of ‘1’ to AD0BUSY and lasts for 3 SAR clocks, followed by conversion.
01: Tracking started by the overflow of Timer 3 and last for 3 SAR clocks, followed by conversion.
10: ADC0 tracks only when CNVSTR input is logic low; conversion starts on rising CNVSTR edge.
11: Tracking started by the overflow of Timer 2 and last for 3 SAR clocks, followed by conversion.

Bit1: AD0WINT: ADC0 Window Compare Interrupt Flag.
This bit must be cleared by software.
0: ADC0 Window Comparison Data match has not occurred since this flag was last cleared.
1: ADC0 Window Comparison Data match has occurred.

Bit0: AD0LJST: ADC0 Left Justify Select.
0: Data in ADC0H:ADC0L registers are right-justified.
1: Data in ADC0H:ADC0L registers are left-justified.

6.3. ADC0 Programmable Window Detector

The ADC0 Programmable Window Detector continuously compares the ADC0 output to user-programmed limits, and notifies the system when an out-of-bound 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 (AD0WINT in ADC0CN) can also be used in polled mode. The high and low bytes of the reference words are loaded into the ADC0 Greater-Than and ADC0 Less-Than registers (ADC0GTH, ADC0GTL, ADC0LTH, and ADC0LTL). Reference comparisons are shown starting on page 70. 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 6.12. ADC0GTH: ADC0 Greater-Than Data High Byte Register (C8051F022/3)

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: High byte of ADC0 Greater-Than Data Word.

Figure 6.13. ADC0GTL: ADC0 Greater-Than Data Low Byte Register (C8051F022/3)

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: Low byte of ADC0 Greater-Than Data Word.

Figure 6.14. ADC0LTH: ADC0 Less-Than Data High Byte Register (C8051F022/3)

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: High byte of ADC0 Less-Than Data Word.

Figure 6.15. ADC0LTL: ADC0 Less-Than Data Low Byte Register (C8051F022/3)

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: Low byte of ADC0 Less-Than Data Word.

Figure 7.6. ADC1CN: ADC1 Control Register (C8051F020/1/2/3)

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
AD1EN	AD1TM	AD1INT	AD1BUSY	AD1CM2	AD1CM1	AD1CM0	-	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xAA

Bit7: AD1EN: ADC1 Enable Bit.
0: ADC1 Disabled. ADC1 is in low-power shutdown.
1: ADC1 Enabled. ADC1 is active and ready for data conversions.

Bit6: AD1TM: ADC1 Track Mode Bit.
0: Normal Track Mode: When ADC1 is enabled, tracking is continuous unless a conversion is in process.
1: Low-power Track Mode: Tracking Defined by AD1STM2-0 bits (see below).

Bit5: AD1INT: ADC1 Conversion Complete Interrupt Flag.
This flag must be cleared by software.
0: ADC1 has not completed a data conversion since the last time this flag was cleared.
1: ADC1 has completed a data conversion.

Bit4: AD1BUSY: ADC1 Busy Bit.
Read:
0: ADC1 Conversion is complete or a conversion is not currently in progress. AD1INT is set to logic 1 on the falling edge of AD1BUSY.
1: ADC1 Conversion is in progress.
Write:
0: No Effect.
1: Initiates ADC1 Conversion if AD1STM2-0 = 000b

Bit3-1: AD1CM2-0: ADC1 Start of Conversion Mode Select.
AD1TM = 0:
000: ADC1 conversion initiated on every write of ‘1’ to AD1BUSY.
001: ADC1 conversion initiated on overflow of Timer 3.
010: ADC1 conversion initiated on rising edge of external CNVSTR.
011: ADC1 conversion initiated on overflow of Timer 2.
1xx: ADC1 conversion initiated on write of ‘1’ to AD0BUSY (synchronized with ADC0 software-commanded conversions).
AD1TM = 1:
000: Tracking initiated on write of ‘1’ to AD1BUSY and lasts 3 SAR1 clocks, followed by conversion.
001: Tracking initiated on overflow of Timer 3 and lasts 3 SAR1 clocks, followed by conversion.
010: ADC1 tracks only when CNVSTR input is logic low; conversion starts on rising CNVSTR edge.
011: Tracking initiated on overflow of Timer 2 and lasts 3 SAR1 clocks, followed by conversion.
1xx: Tracking initiated on write of ‘1’ to AD0BUSY and lasts 3 SAR1 clocks, followed by conversion.

Bit0: UNUSED. Read = 0b. Write = don’t care.

Notes

Figure 12.7. 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

Bits7-0: ACC: Accumulator.
This register is the accumulator for arithmetic operations.

Figure 12.8. 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

Bits7-0: B: B Register.
This register serves as a second accumulator for certain arithmetic operations.

12.3.5. 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 12.9. IE: Interrupt Enable

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
EA	IEGF0	ET2	ES0	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. When set to '0', individual interrupt mask settings are overridden.
0: Disable all interrupt sources.
1: Enable each interrupt according to its individual mask setting.

Bit6: IEGF0: General Purpose Flag 0.
This is a general purpose flag for use under software control.

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

Bit4: ES0: Enable UART0 Interrupt.
This bit sets the masking of the UART0 interrupt.
0: Disable UART0 interrupt.
1: Enable UART0 interrupt.

Bit3: ET1: Enable Timer 1 Interrupt.
This bit sets the masking of the Timer 1 interrupt.
0: Disable all Timer 1 interrupt.
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 interrupt.
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.

14. OSCILLATORS

Each MCU includes an internal oscillator and an external oscillator drive circuit, either of which can generate the system clock. The MCUs operate 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 14.1. The internal oscillator's electrical specifications are given in Table 14.1.

Both oscillators are disabled when the /RST pin is held low. The MCUs can run from the internal oscillator permanently, or 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 Table 14.1). 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; in this configuration, the XTAL1 pin is used as the CMOS clock input. The XTAL1 and XTAL2 pins are NOT 5V tolerant.

Figure 14.1. Oscillator Diagram

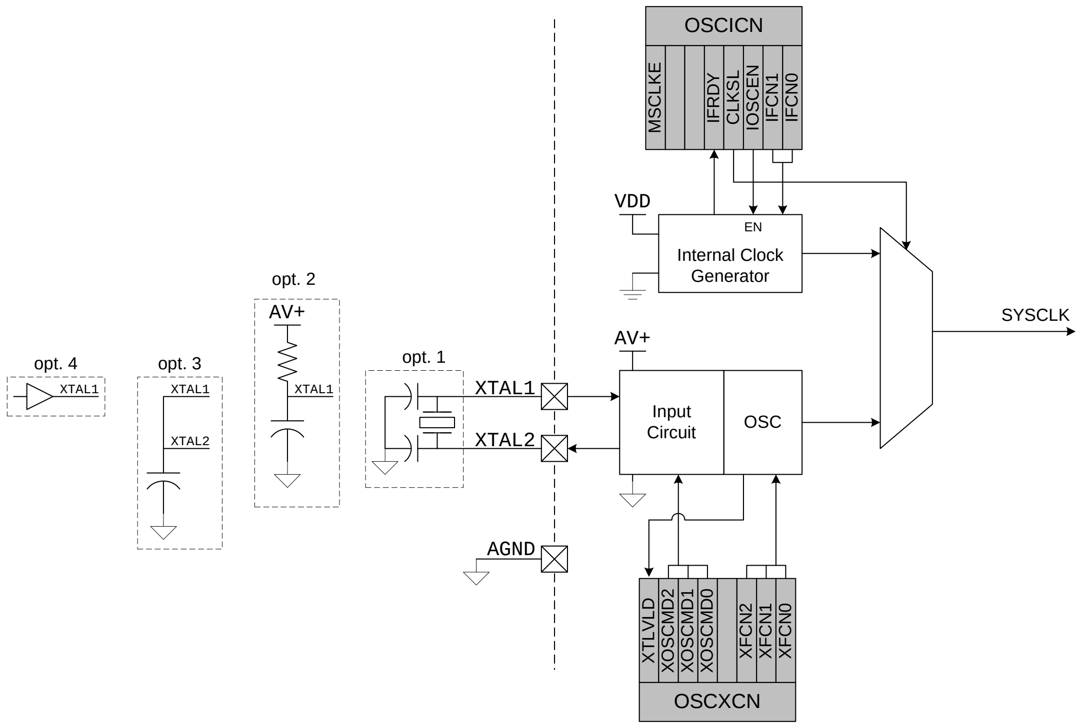


Figure 17.7. XBR0: Port I/O Crossbar Register 0

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
CP0E	ECI0E	PCA0ME			UART0EN	SPI0EN	SMB0EN	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xE1

Bit7: CP0E: Comparator 0 Output Enable Bit.
0: CP0 unavailable at Port pin.
1: CP0 routed to Port pin.

Bit6: ECI0E: PCA0 External Counter Input Enable Bit.
0: PCA0 External Counter Input unavailable at Port pin.
1: PCA0 External Counter Input (ECI0) routed to Port pin.

Bits5-3: PCA0ME: PCA0 Module I/O Enable Bits.
000: All PCA0 I/O unavailable at Port pins.
001: CEX0 routed to Port pin.
010: CEX0, CEX1 routed to 2 Port pins.
011: CEX0, CEX1, and CEX2 routed to 3 Port pins.
100: CEX0, CEX1, CEX2, and CEX3 routed to 4 Port pins.
101: CEX0, CEX1, CEX2, CEX3, and CEX4 routed to 5 Port pins.
110: RESERVED
111: RESERVED

Bit2: UART0EN: UART0 I/O Enable Bit.
0: UART0 I/O unavailable at Port pins.
1: UART0 TX routed to P0.0, and RX routed to P0.1.

Bit1: SPI0EN: SPI0 Bus I/O Enable Bit.
0: SPI0 I/O unavailable at Port pins.
1: SPI0 SCK, MISO, MOSI, and NSS routed to 4 Port pins.

Bit0: SMB0EN: SMBus0 Bus I/O Enable Bit.
0: SMBus0 I/O unavailable at Port pins.
1: SMBus0 SDA and SCL routed to 2 Port pins.

Figure 17.10. P0: Port0 Data 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]: Port0 Output Latch Bits.
(Write - Output appears on I/O pins per XBR0, XBR1, XBR2, and XBR3 Registers)
0: Logic Low Output.
1: Logic High Output (open if corresponding P0MDOUT.n bit = 0).
(Read - Regardless of XBR0, XBR1, XBR2, and XBR3 Register settings).
0: P0.n pin is logic low.
1: P0.n pin is logic high.

Note: P0.7 (/WR), P0.6 (/RD), and P0.5 (ALE) can be driven by the External Data Memory Interface. See **Section "16. EXTERNAL DATA MEMORY INTERFACE AND ON-CHIP XRAM" on page 145** for more information. See also Figure 17.9 for information about configuring the Crossbar for External Memory accesses.

Figure 17.11. P0MDOUT: Port0 Output Mode 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: P0MDOUT.[7:0]: Port0 Output Mode Bits.
0: Port Pin output mode is configured as Open-Drain.
1: Port Pin output mode is configured as Push-Pull.

Note: SDA, SCL, and RX0 (when UART0 is in Mode 0) and RX1 (when UART1 is in Mode 0) are always configured as Open-Drain when they appear on Port pins.

Figure 17.14. P1MDOUT: Port1 Output Mode 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: P1MDOUT.[7:0]: Port1 Output Mode Bits.
0: Port Pin output mode is configured as Open-Drain.
1: Port Pin output mode is configured as Push-Pull.

Note: SDA, SCL, and RX0 (when UART0 is in Mode 0) and RX1 (when UART1 is in Mode 0) are always configured as Open-Drain when they appear on Port pins.

Figure 17.15. P2: Port2 Data 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	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0 (bit addressable)	SFR Address: 0xA0

Bits7-0: P2.[7:0]: Port2 Output Latch Bits.
(Write - Output appears on I/O pins per XBR0, XBR1, XBR2, and XBR3 Registers)
0: Logic Low Output.
1: Logic High Output (open if corresponding P2MDOUT.n bit = 0).
(Read - Regardless of XBR0, XBR1, XBR2, and XBR3 Register settings).
0: P2.n pin is logic low.
1: P2.n pin is logic high.

Note: P2.[7:0] can be driven by the External Data Memory Interface (as Address[15:8] in Multiplexed mode, or as Address[7:0] in Non-multiplexed mode). See **Section "16. EXTERNAL DATA MEMORY INTERFACE AND ON-CHIP XRAM" on page 145** for more information about the External Memory Interface.

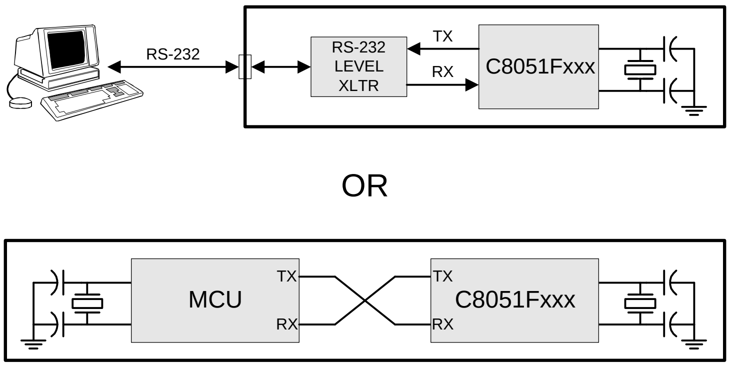
Figure 17.16. P2MDOUT: Port2 Output Mode Register

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: P2MDOUT.[7:0]: Port2 Output Mode Bits.
0: Port Pin output mode is configured as Open-Drain.
1: Port Pin output mode is configured as Push-Pull.

Note: SDA, SCL, and RX0 (when UART0 is in Mode 0) and RX1 (when UART1 is in Mode 0) are always configured as Open-Drain when they appear on Port pins.

Figure 20.6. UART Modes 1, 2, and 3 Interconnect Diagram



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

Mode 3 uses the Mode 2 transmission protocol with the Mode 1 baud rate generation. Mode 3 operation transmits 11 bits: a start bit, 8 data bits (LSB first), a programmable ninth data bit, and a stop bit. The baud rate is derived from Timer 1 or Timer 2 overflows, as defined by Equation 20.1 and Equation 20.2. Multiprocessor communications and hardware address recognition are supported, as described in **Section 20.2**.

20.2. 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 and the built-in UART0 address recognition hardware. A master processor begins a transfer with an address byte to select one or more target slave devices. 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.

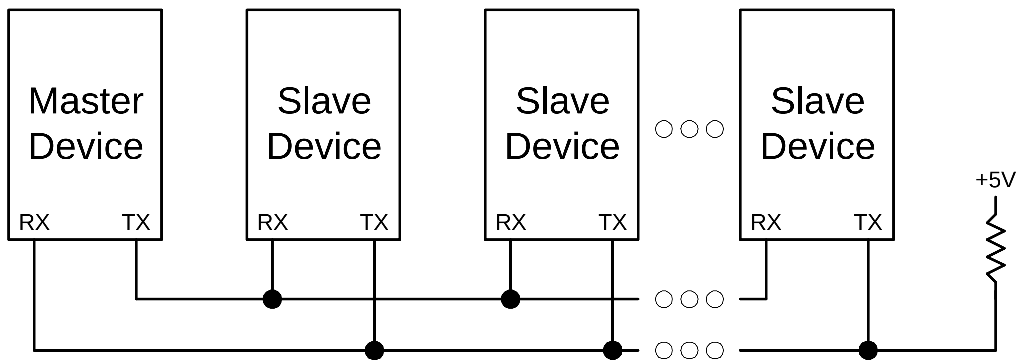
The UART0 address is configured via two SFRs: SADDR0 (Serial Address) and SADEN0 (Serial Address Enable). SADEN0 sets the bit mask for the address held in SADDR0: bits set to logic 1 in SADEN0 correspond to bits in SADDR0 that are checked against the received address byte; bits set to logic 0 in SADEN0 correspond to “don’t care” bits in SADDR0.

	Example 1		Example 2		Example 3
SADDR0	= 00110101	SADDR0	= 00110101	SADDR0	= 00110101
SADEN0	= 00001111	SADEN0	= 11110011	SADEN0	= 11000000
UART0 Address	= xxxx0101	UART0 Address	= 0011xx01	UART0 Address	= 00xxxxxx

Setting the SM20 bit (SCON0.5) configures UART0 such that when a stop bit is received, UART0 will generate an interrupt only if the ninth bit is logic 1 (RB80 = 1) and the received data byte matches the UART0 slave address. Following the received address interrupt, the slave should clear its SM20 bit to enable interrupts on the reception of the following data byte(s). Once the entire message is received, the addressed slave should reset its SM20 bit to ignore all transmissions until it receives the next address byte. While SM20 is logic 1, UART0 ignores all bytes that do not match the UART0 address and include a ninth bit that is logic 1.

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 broadcast address is the logical OR of registers SADDR0 and SADEN0, and ‘0’s of the result are treated as “don’t cares”. Typically a broadcast address of 0xFF (hexadecimal) is acknowledged by all slaves, assuming “don’t care” bits as ‘1’s. 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 20.7. UART Multi-Processor Mode Interconnect Diagram



20.3. Frame and Transmission Error Detection

Frame error detection is available in the following modes when the SSTAT0 bit in register PCON is set to logic 1. Note: The SSTAT0 bit must be logic 1 to access any of the status bits (FE0, RXOVR0, and TXCOL0). To access the UART0 Mode Select bits (SM00, SM10, and SM20), the SSTAT0 bit must be logic 0.

All Modes:
The Transmit Collision bit (TXCOL0 bit in register SCON0) reads '1' if user software writes data to the SBUF0 register while a transmit is in progress. Note that the TXCOL0 bit also functions as the SM20 bit when the SSTAT0 bit in register PCON is logic 0.

Modes 1, 2, and 3:
The Receive Overrun bit (RXOVR0 in register SCON0) reads '1' if a new data byte is latched into the receive buffer before software has read the previous byte. Note that the RXOVR0 bit also functions as the SM10 bit when the SSTAT0 bit in register PCON is logic 0.

The Frame Error bit (FE0 in register SCON0) reads '1' if an invalid (low) STOP bit is detected. Note that the FE0 bit also functions as the SM00 bit when the SSTAT0 bit in register PCON is logic 0.

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

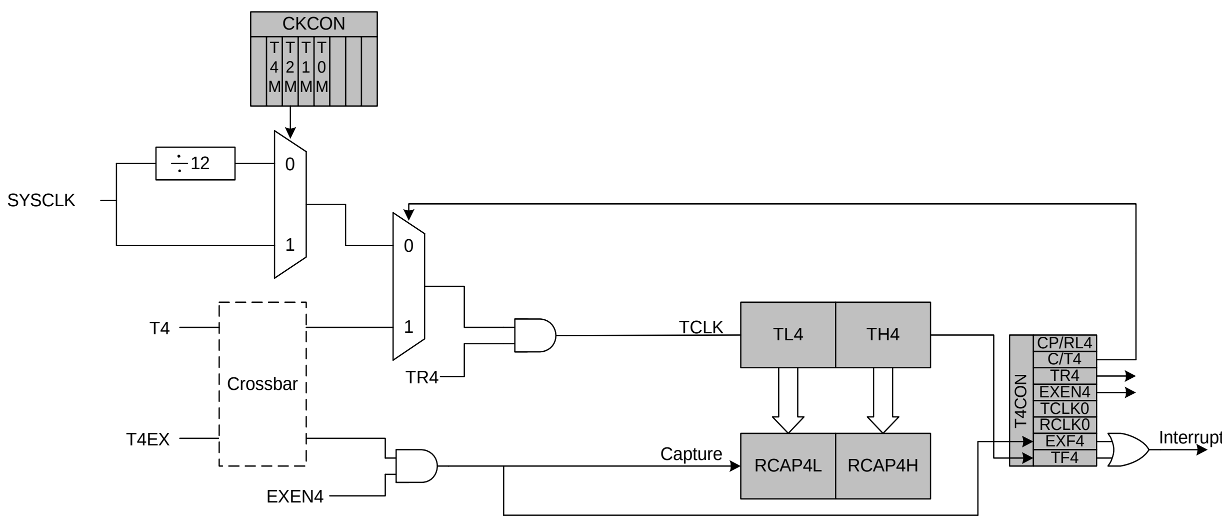
In this mode, Timer 4 operates as a 16-bit counter/timer with capture facility. A high-to-low transition on the T4EX input pin causes the following to occur:

1. The 16-bit value in Timer 4 (TH4, TL4) is loaded into the capture registers (RCAP4H, RCAP4L).
2. The Timer 4 External Flag (EXF2) is set to '1'.
3. A Timer 4 interrupt is generated if enabled.

Timer 4 can use either SYSCLK, SYSCLK divided by 12, or high-to-low transitions on the T4 input pin as its clock source when operating in Capture mode. Clearing the C/T4 bit (T4CON.1) selects the system clock as the input for the timer (divided by one or twelve as specified by the Timer Clock Select bit T4M in CKCON). When C/T4 is set to logic 1, a high-to-low transition at the T4 input pin increments the counter/timer register. As the 16-bit counter/timer register increments and overflows from 0xFFFF to 0x0000, the TF4 timer overflow flag (T4CON.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/RL4 (T4CON.0) and the Timer 4 Run Control bit TR4 (T4CON.2) to logic 1. The Timer 4 External Enable EXEN4 (T4CON.3) must also be set to logic 1 to enable a capture. If EXEN4 is cleared, transitions on T4EX will be ignored.

Figure 22.25. T4 Mode 0 Block Diagram



22.3.3. Mode 2: Baud Rate Generator

Timer 4 can be used as a baud rate generator for UART1 when UART1 is operated in modes 1 or 3 (refer to Section "21.1. UART1 Operational Modes" on page 216 for more information on the UART1 operational modes). In Baud Rate Generator mode, Timer 4 works similarly to the auto-reload mode. On overflow, the 16-bit value held in the two capture registers (RCAP4H, RCAP4L) is automatically loaded into the counter/timer register. However, the TF4 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 4 overflows can be selected to generate baud rates for transmit and/or receive independently.

The Baud Rate Generator mode is selected by setting RCLK1 (T4CON.5) and/or TCLK1 (T4CON.4) to '1'. When RCLK1 or TCLK1 is set to logic 1, Timer 4 operates in the auto-reload mode regardless of the state of the CP/RL4 bit. Note that in Baud Rate Generator mode, the Timer 4 timebase is the system clock divided by two. When selected as the UART1 baud clock source, Timer 4 defines the UART1 baud rate as follows:

*Baud Rate = SYSCLK / ((65536 - [RCAP4H, RCAP4L]) * 32)*

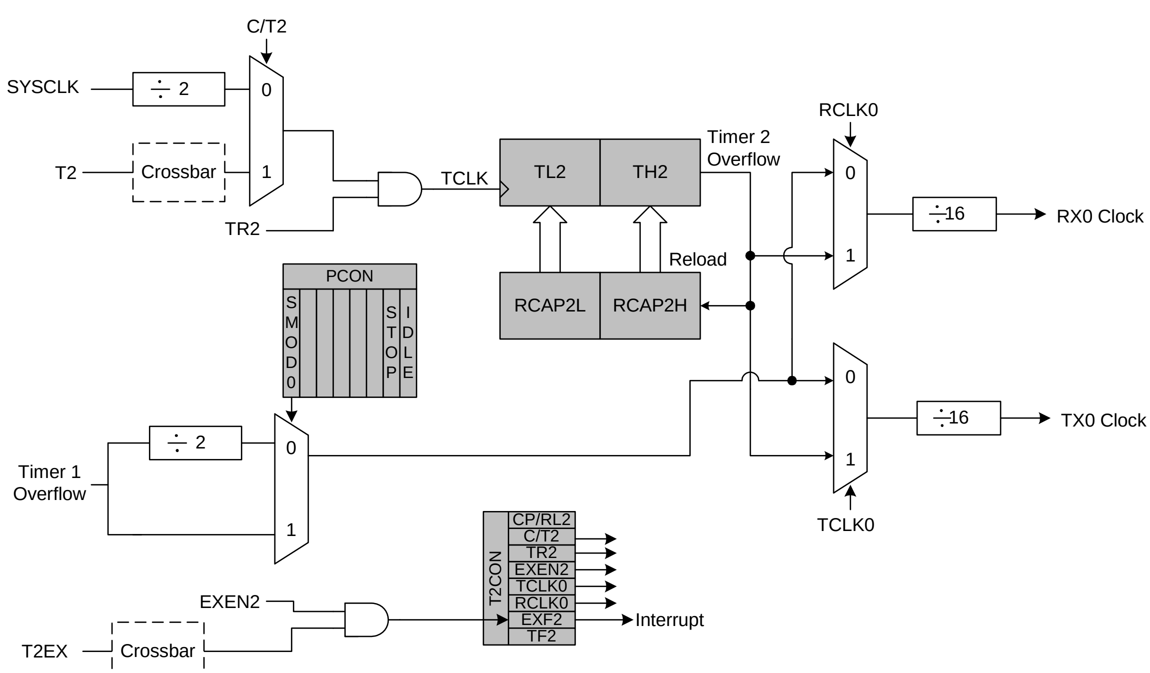
If a different time base is required, setting the C/T4 bit to logic 1 will allow the timebase to be derived from the external input pin T4. In this case, the baud rate for the UART is calculated as:

Baud Rate = F_{CLK} */ ((65536 - [RCAP4H, RCAP4L]) * 16)*

Where F_{CLK} is the frequency of the signal (TCLK) supplied to Timer 4 and [RCAP4H, RCAP4L] is the 16-bit value held in the capture registers.

As explained above, in Baud Rate Generator mode, Timer 4 does not set the TF4 overflow flag and therefore cannot generate an interrupt. However, if EXEN4 is set to logic 1, a high-to-low transition on the T4EX input pin will set the EXF4 flag and a Timer 4 interrupt will occur if enabled. Therefore, the T4EX input may be used as an additional external interrupt source.

Figure 22.27. T4 Mode 2 Block Diagram



SILICON LABS

23.1. 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 into a “snapshot” register; the following PCA0H read accesses this “snapshot” register. Reading the PCA0L Register first guarantees an accurate reading of the entire 16-bit PCA0 counter. Reading PCA0H or PCA0L does not disturb the counter operation. The CPS2-CPS0 bits in the PCA0MD register select the timebase for the counter/timer as shown in Table 23.1. **Note that in ‘External oscillator source divided by 8’ mode, the external oscillator source is synchronized with the system clock, and must have a frequency less than or equal to the system clock.**

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 CPU is in Idle mode.

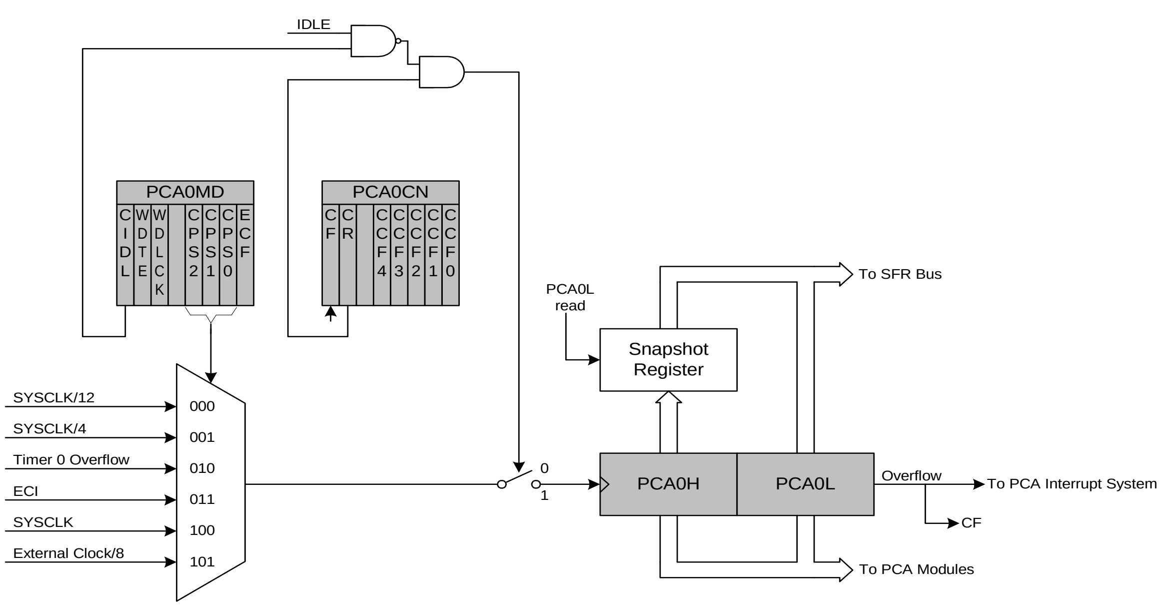
Table 23.1. PCA Timebase Input Options

CPS2	CPS1	CPS0	Timebase
0	0	0	System clock divided by 12
0	0	1	System clock divided by 4
0	1	0	Timer 0 overflow
0	1	1	High-to-low transitions on ECI‡ (max rate = system clock divided by 4)
1	0	0	System clock
1	0	1	External oscillator source divided by 8†

†External oscillator source divided by 8 is synchronized with the system clock.

‡The minimum high or low time for the ECI input signal is at least 2 system clock cycles.

Figure 23.2. PCA Counter/Timer Block Diagram



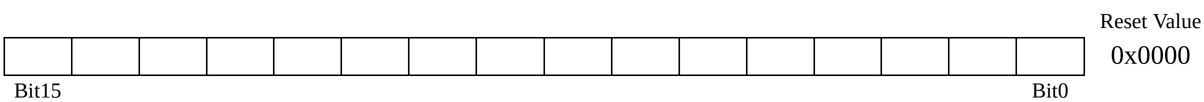
24. JTAG (IEEE 1149.1)

Each MCU has an on-chip JTAG interface and logic to support boundary scan for production and in-system testing, Flash read/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 accessed via four dedicated pins on the MCU: TCK, TMS, TDI, and TDO.

Through the 16-bit JTAG Instruction Register (IR), any of the seven instructions shown in Figure 24.1 can be commanded. There are three DR's associated with JTAG Boundary-Scan, and four associated with Flash read/write operations on the MCU.

Figure 24.1. 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

24.3. Debug Support

Each MCU has on-chip JTAG and debug logic that provides non-intrusive, full speed, in-circuit debug support using the production part installed in the end application, via the four pin JTAG I/F. Silicon Labs' debug system supports inspection and modification of memory and registers, breakpoints, 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 synchronized) while debugging. The Watchdog Timer (WDT) is disabled when the MCU is halted during single stepping or at a breakpoint.

The C8051F020DK is a development kit with all the hardware and software necessary to develop application code and perform in-circuit debug with each MCU in the C8051F020 family. Each kit includes an Integrated Development Environment (IDE) which has a debugger and integrated 8051 assembler. The kit also includes an RS-232 to JTAG interface module referred to as the Serial Adapter, a target application board with a C8051F020 installed, RS-232 and JTAG cables, and wall-mount power supply.