



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	Active
Core Processor	8051
Core Size	8-Bit
Speed	25MHz
Connectivity	SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, Temp Sensor, WDT
Number of I/O	16
Program Memory Size	4KB (4K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 5.25V
Data Converters	A/D 16x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	20-TSSOP (0.173", 4.40mm Width)
Supplier Device Package	20-TSSOP
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f534a-it

List of Registers

SFR Definition 4.4. ADC0MX: ADC0 Channel Select	64
SFR Definition 4.5. ADC0CF: ADC0 Configuration	65
SFR Definition 4.6. ADC0H: ADC0 Data Word MSB	66
SFR Definition 4.7. ADC0L: ADC0 Data Word LSB	66
SFR Definition 4.8. ADC0CN: ADC0 Control	67
SFR Definition 4.9. ADC0TK: ADC0 Tracking Mode Select	68
SFR Definition 4.10. ADC0GTH: ADC0 Greater-Than Data High Byte	69
SFR Definition 4.11. ADC0GTL: ADC0 Greater-Than Data Low Byte	69
SFR Definition 4.12. ADC0LTH: ADC0 Less-Than Data High Byte	70
SFR Definition 4.13. ADC0LTL: ADC0 Less-Than Data Low Byte	70
SFR Definition 5.1. REF0CN: Reference Control	73
SFR Definition 6.1. REG0CN: Regulator Control	75
SFR Definition 7.1. CPT0CN: Comparator0 Control	78
SFR Definition 7.2. CPT0MX: Comparator0 MUX Selection	79
SFR Definition 7.3. CPT0MD: Comparator0 Mode Selection	80
SFR Definition 8.1. SP: Stack Pointer	87
SFR Definition 8.2. DPL: Data Pointer Low Byte	87
SFR Definition 8.3. DPH: Data Pointer High Byte	87
SFR Definition 8.4. PSW: Program Status Word	88
SFR Definition 8.5. ACC: Accumulator	89
SFR Definition 8.6. B: B Register	89
SFR Definition 8.7. PCON: Power Control	91
SFR Definition 10.1. IE: Interrupt Enable	100
SFR Definition 10.2. IP: Interrupt Priority	101
SFR Definition 10.3. EIE1: Extended Interrupt Enable 1	102
SFR Definition 10.4. EIP1: Extended Interrupt Priority 1	103
SFR Definition 10.5. IT01CF: INT0/INT1 Configuration	105
SFR Definition 11.1. VDDMON: VDD Monitor Control	109
SFR Definition 11.2. RSTSRC: Reset Source	112
SFR Definition 12.1. PSCTL: Program Store R/W Control	119
SFR Definition 12.2. FLKEY: Flash Lock and Key	119
SFR Definition 13.1. XBR0: Port I/O Crossbar Register 0	127
SFR Definition 13.2. XBR1: Port I/O Crossbar Register 1	128
SFR Definition 13.3. P0: Port0	129
SFR Definition 13.4. P0MDIN: Port0 Input Mode	129
SFR Definition 13.5. P0MDOUT: Port0 Output Mode	130
SFR Definition 13.6. P0SKIP: Port0 Skip	130
SFR Definition 13.7. P0MAT: Port0 Match	131
SFR Definition 13.8. P0MASK: Port0 Mask	131
SFR Definition 13.9. P1: Port1	132
SFR Definition 13.10. P1MDIN: Port1 Input Mode	132
SFR Definition 13.11. P1MDOUT: Port1 Output Mode	133
SFR Definition 13.12. P1SKIP: Port1 Skip	133

C8051F52x/F52xA/F53x/F53xA

Table 2.2. Global DC Electrical Characteristics

–40 to +125 °C, 25 MHz System Clock unless otherwise specified. Typical values are given at 25 °C

Parameter	Conditions	Min	Typ	Max	Units
Digital Supply Current—CPU Inactive (Idle Mode, not fetching instructions from Flash)					
Idle $I_{DD}^{3,4}$	$V_{DD} = 2.1$ V:				
	Clock = 32 kHz	—	8	—	μ A
	Clock = 200 kHz	—	22	—	μ A
	Clock = 1 MHz	—	0.09	—	mA
	Clock = 25 MHz	—	2.2	5	mA
	$V_{DD} = 2.6$ V:				
	Clock = 32 kHz	—	9	—	μ A
	Clock = 200 kHz	—	30	—	μ A
Idle I_{DD} Frequency Sensitivity ^{3,6}	T = 25 °C:				
	$V_{DD} = 2.1$ V, F \leq 1 MHz	—	90	—	μ A/MHz
	$V_{DD} = 2.1$ V, F > 1 MHz	—	90	—	μ A/MHz
	$V_{DD} = 2.6$ V, F \leq 1 MHz	—	118	—	μ A/MHz
Digital Supply Current ³ (Stop or Suspend Mode)	Oscillator not running, V_{DD} Monitor Disabled.				
	T = 25 °C	—	2	—	μ A
	T = 60 °C	—	3	—	μ A
	T = 125 °C	—	50	—	μ A
Notes:					
<ol style="list-style-type: none"> 1. For more information on V_{REGIN} characteristics, see Table 2.6 on page 30. 2. SYSCLK must be at least 32 kHz to enable debugging. 3. Based on device characterization data; Not production tested. 4. Does not include internal oscillator or internal regulator supply current. 5. I_{DD} can be estimated for frequencies \leq 12 MHz by multiplying the frequency of interest by the frequency sensitivity number for that range. When using these numbers to estimate $I_{DD} > 12$ MHz, the estimate should be the current at 25 MHz minus the difference in current indicated by the frequency sensitivity number. For example: $V_{DD} = 2.6$ V; F= 20 MHz, $I_{DD} = 7.3$ mA – (25 MHz – 20 MHz) x 0.184 mA/MHz = 6.38 mA. 6. Idle I_{DD} can be estimated for frequencies \leq 1 MHz by multiplying the frequency of interest by the frequency sensitivity number for that range. When using these numbers to estimate $I_{DD} > 1$ MHz, the estimate should be the current at 25 MHz minus the difference in current indicated by the frequency sensitivity number. For example: $V_{DD} = 2.6$ V; F= 5 MHz, Idle $I_{DD} = 3$ mA – (25 MHz– 5 MHz) x 118 μA/MHz = 0.64 mA. 					

C8051F52x/F52xA/F53x/F53xA

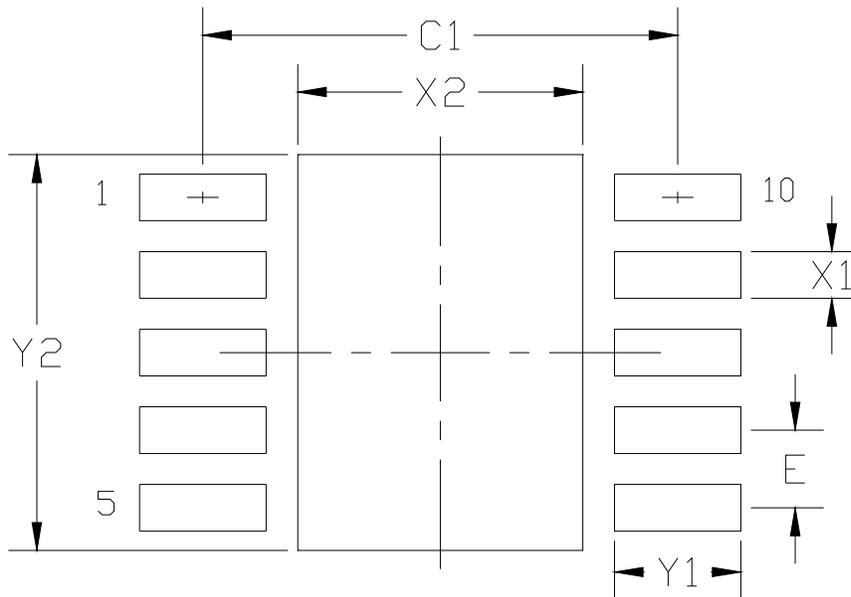


Figure 3.3. DFN-10 Landing Diagram

Table 3.3. DFN-10 Landing Diagram Dimensions

Dimension	Min	Max
C1	2.90	3.00
E	0.50 BSC.	
X1	0.20	0.30
X2	1.70	1.80
Y1	0.70	0.80
Y2	2.45	2.55

Notes:

General

- All dimensions shown are in millimeters (mm) unless otherwise noted.
- This land pattern design is based on the IPC-7351 guidelines.

Solder Mask Design

- All metal pads are to be non-solder mask defined (NSMD). Clearance between the solder mask and the metal pad is to be 60 μm minimum, all the way around the pad.

Stencil Design

- A stainless steel, laser-cut and electro-polished stencil with trapezoidal walls should be used to assure good solder paste release.
- The stencil thickness should be 0.125 mm (5 mils).
- The ratio of stencil aperture to land pad size should be 1:1 for all perimeter pads.
- A 4x1 array of 1.60 x 0.45 mm openings on 0.65 mm pitch should be used for the center ground pad.

Card Assembly

- A No-Clean, Type-3 solder paste is recommended.
- The recommended card reflow profile is per the JEDEC/IPC J-STD-020 specification for Small Body Components.

C8051F52x/F52xA/F53x/F53xA

Table 3.9. QFN-20 Landing Diagram Dimensions

Symbol	Min	Max
C1	3.90	4.00
C2	3.90	4.00
E	0.50 BSC.	
X1	0.20	0.30
X2	2.75	2.85
Y1	0.65	0.75
Y2	2.75	2.85

Notes:

General

1. All dimensions shown are in millimeters (mm) unless otherwise noted.
2. This land pattern design is based on the IPC-7351 guidelines.

Solder Mask Design

3. All metal pads are to be non-solder mask defined (NSMD). Clearance between the solder mask and the metal pad is to be 60 μm minimum, all the way around the pad.

Stencil Design

4. A stainless steel, laser-cut and electro-polished stencil with trapezoidal walls should be used to assure good solder paste release.
5. The stencil thickness should be 0.125 mm (5 mils).
6. The ratio of stencil aperture to land pad size should be 1:1 for all perimeter pads.
7. A 2x2 array of 1.10 x 1.10 mm openings on 1.30 mm pitch should be used for the center ground pad.

Card Assembly

8. A No-Clean, Type-3 solder paste is recommended.
9. The recommended card reflow profile is per the JEDEC/IPC J-STD-020 specification for Small Body Components.

4. 12-Bit ADC (ADC0)

The ADC0 on the C8051F52x/F52xA/F53x/F53xA Family consists of an analog multiplexer (AMUX0) with 16/6 total input selections, and a 200 kpsps, 12-bit successive-approximation-register (SAR) ADC with integrated track-and-hold, programmable window detector, programmable gain, and hardware accumulator. The ADC0 subsystem has a special Burst Mode which can automatically enable ADC0, capture and accumulate samples, then place ADC0 in a low power shutdown mode without CPU intervention. The AMUX0, data conversion modes, and window detector are all configurable under software control via the Special Function Registers shown in Figure 4.1. ADC0 inputs are single-ended and may be configured to measure P0.0-P1.7, the Temperature Sensor output, V_{DD} , or GND with respect to GND. The voltage reference for the ADC is selected as described in Section “5. Voltage Reference” on page 72. ADC0 is enabled when the AD0EN bit in the ADC0 Control register (ADC0CN) is set to logic 1, or when performing conversions in Burst Mode. ADC0 is in low power shutdown when AD0EN is logic 0 and no Burst Mode conversions are taking place.

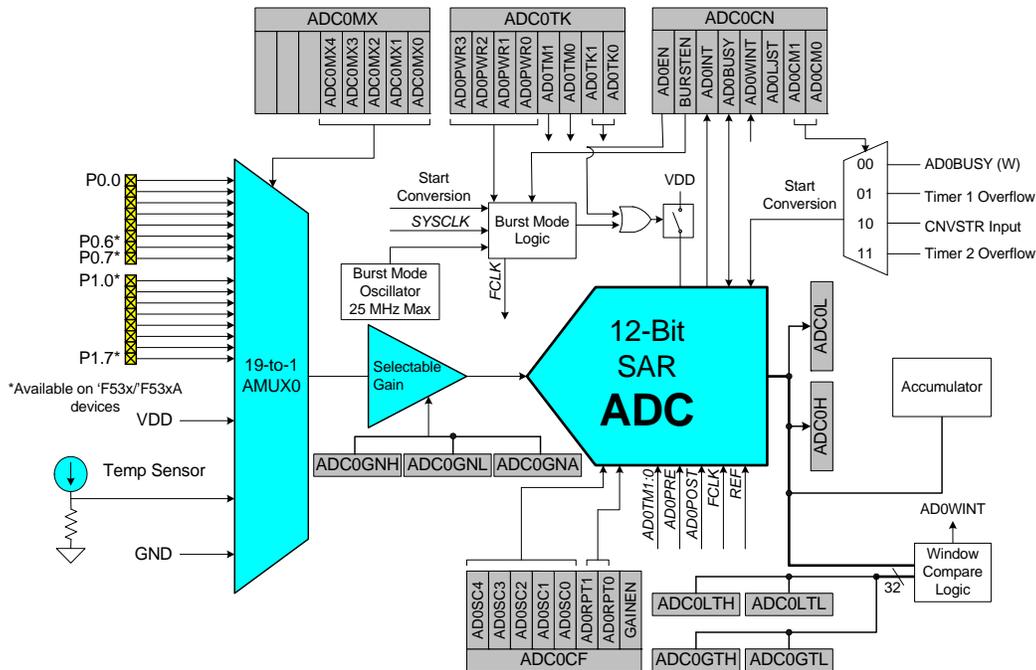


Figure 4.1. ADC0 Functional Block Diagram

4.1. Analog Multiplexer

AMUX0 selects the input channel to the ADC. Any of the following may be selected as an input: P0.0–P1.7, the on-chip temperature sensor, the core power supply (V_{DD}), or ground (GND). **ADC0 is single-ended and all signals measured are with respect to GND.** The ADC0 input channels are selected using the ADC0MX register as described in SFR Definition 4.4.

Important Note About ADC0 Input Configuration: Port pins selected as ADC0 inputs should be configured as analog inputs, and should be skipped by the Digital Crossbar. To configure a Port pin for analog input, set to 0 the corresponding bit in register PnMDIN (for $n = 0, 1$). To force the Crossbar to skip a Port pin, set to 1 the corresponding bit in register PnSKIP (for $n = 0, 1$). See Section “13. Port Input/Output” on page 120 for more Port I/O configuration details.

C8051F52x/F52xA/F53x/F53xA

Post-Tracking Mode is selected when AD0TM is set to 01b. A programmable tracking time based on AD0TK is started immediately following the convert start signal. Conversions are started after the programmed tracking time ends. After a conversion is complete, ADC0 does not track the input. Rather, the sampling capacitor remains disconnected from the input making the input pin high-impedance until the next convert start signal.

Dual-Tracking Mode is selected when AD0TM is set to 11b. A programmable tracking time based on AD0TK is started immediately following the convert start signal. Conversions are started after the programmed tracking time ends. After a conversion is complete, ADC0 tracks continuously until the next conversion is started.

Depending on the output connected to the ADC input, additional tracking time, more than is specified in Table 2.3 on page 28, may be required after changing MUX settings. See the settling time requirements described in Section “4.3.6. Settling Time Requirements” on page 60.

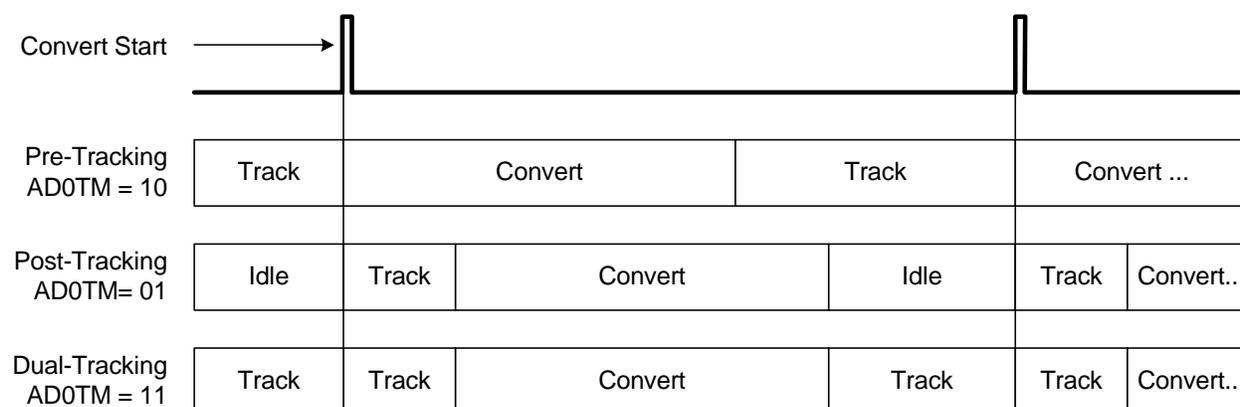


Figure 4.3. ADC0 Tracking Modes

4.3.3. Timing

ADC0 has a maximum conversion speed specified in Table 2.3 on page 28. ADC0 is clocked from the ADC0 Subsystem Clock (FCLK). The source of FCLK is selected based on the BURSTEN bit. When BURSTEN is logic 0, FCLK is derived from the current system clock. When BURSTEN is logic 1, FCLK is derived from the Burst Mode Oscillator, which is an independent clock source whose maximum frequency is specified in Table 2.3 on page 28.

When ADC0 is performing a conversion, it requires a clock source that is typically slower than FCLK. The ADC0 SAR conversion clock (SAR clock) is a divided version of FCLK. The divide ratio can be configured using the AD0SC bits in the ADC0CF register. The maximum SAR clock frequency is listed in Table 2.3 on page 28.

ADC0 can be in one of three states at any given time: tracking, converting, or idle. Tracking time depends on the tracking mode selected. For Pre-Tracking Mode, tracking is managed by software and ADC0 starts conversions immediately following the convert start signal. For Post-Tracking and Dual-Tracking Modes, the tracking time after the convert start signal is equal to the value determined by the AD0TK bits plus 2 FCLK cycles. Tracking is immediately followed by a conversion. The ADC0 conversion time is always 13 SAR clock cycles plus an additional 2 FCLK cycles to start and complete a conversion. Figure 4.4 shows timing diagrams for a conversion in Pre-Tracking Mode and tracking plus conversion in Post-Tracking or Dual-Tracking Mode. In this example, repeat count is set to one.

C8051F52x/F52xA/F53x/F53xA

4.3.5. Output Conversion Code

The registers ADC0H and ADC0L contain the high and low bytes of the output conversion code. When the repeat count is set to 1, conversion codes are represented in 12-bit unsigned integer format and the output conversion code is updated after each conversion. Inputs are measured from 0 to $V_{REF} \times 4095/4096$. Data can be right-justified or left-justified, depending on the setting of the AD0LJST bit (ADC0CN.2). Unused bits in the ADC0H and ADC0L registers are set to 0. Example codes are shown below for both right-justified and left-justified data.

Input Voltage	Right-Justified ADC0H:ADC0L (AD0LJST = 0)	Left-Justified ADC0H:ADC0L (AD0LJST = 1)
$V_{REF} \times 4095/4096$	0x0FFF	0xFF0
$V_{REF} \times 2048/4096$	0x0800	0x8000
$V_{REF} \times 2047/4096$	0x07FF	0x7FF0
0	0x0000	0x0000

When the ADC0 Repeat Count is greater than 1, the output conversion code represents the accumulated result of the conversions performed and is updated after the last conversion in the series is finished. Sets of 4, 8, or 16 consecutive samples can be accumulated and represented in unsigned integer format. The repeat count can be selected using the AD0RPT bits in the ADC0CF register. The value must be right-justified (AD0LJST = "0"), and unused bits in the ADC0H and ADC0L registers are set to '0'. The following example shows right-justified codes for repeat counts greater than 1. Notice that accumulating 2^n samples is equivalent to left-shifting by n bit positions when all samples returned from the ADC have the same value.

Input Voltage	Repeat Count = 4	Repeat Count = 8	Repeat Count = 16
$V_{REF} \times 4095/4096$	0x3FFC	0x7FF8	0xFF0
$V_{REF} \times 2048/4096$	0x2000	0x4000	0x8000
$V_{REF} \times 2047/4096$	0x1FFC	0x3FF8	0x7FF0
0	0x0000	0x0000	0x0000

4.3.6. Settling Time Requirements

A minimum tracking time is required before an accurate conversion can be performed. This tracking time is determined by the AMUX0 resistance, the ADC0 sampling capacitance, any external source resistance, and the accuracy required for the conversion.

Figure 4.6 shows the equivalent ADC0 input circuit. The required ADC0 settling time for a given settling accuracy (SA) may be approximated by Equation 4.1. When measuring the Temperature Sensor output, use the settling time specified in Table 2.3 on page 28. See Table 2.3 on page 28 for ADC0 minimum settling time requirements.

$$t = \ln\left(\frac{2^n}{SA}\right) \times R_{TOTAL} C_{SAMPLE}$$

Equation 4.1. ADC0 Settling Time Requirements

Where:

SA is the settling accuracy, given as a fraction of an LSB (for example, 0.25 to settle within 1/4 LSB)

t is the required settling time in seconds

R_{TOTAL} is the sum of the AMUX0 resistance and any external source resistance.

n is the ADC resolution in bits (12).

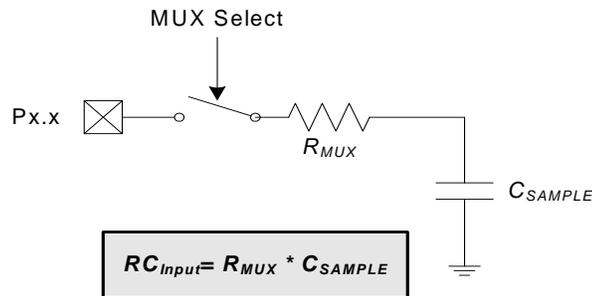


Figure 4.6. ADC0 Equivalent Input Circuits

4.4. Selectable Gain

ADC0 on the C8051F52x/52xA/53x/53xA family of devices implements a selectable gain adjustment option. By writing a value to the gain adjust address range, the user can select gain values between 0 and 1.016.

For example, three analog sources to be measured have full-scale outputs of 5.0 V, 4.0 V, and 3.0 V, respectively. Each ADC measurement would ideally use the full dynamic range of the ADC with an internal voltage reference of 1.5 V or 2.2 V (set to 2.2 V for this example). When selecting signal one (5.0 V full-scale), a gain value of 0.44 (5 V full scale * 0.44 = 2.2 V full scale) provides a full-scale signal of 2.2 V when the input signal is 5.0 V. Likewise, a gain value of 0.55 (4 V full scale * 0.55 = 2.2 V full scale) for the second source and 0.73 (3 V full scale * 0.73 = 2.2 V full scale) for the third source provide full-scale ADC0 measurements when the input signal is full-scale.

Additionally, some sensors or other input sources have small part-to-part variations that must be accounted for to achieve accurate results. In this case, the programmable gain value could be used as a calibration value to eliminate these part-to-part variations.

C8051F52x/F52xA/F53x/F53xA

4.5.1. Window Detector In Single-Ended Mode

Figure 4.7 shows two example window comparisons for right-justified data with $ADC0LTH:ADC0LTL = 0x0200$ (512d) and $ADC0GTH:ADC0GTL = 0x0100$ (256d). The input voltage can range from 0 to $V_{REF} \times (4095/4096)$ with respect to GND, and is represented by a 12-bit unsigned integer value. The repeat count is set to one. In the left example, an $AD0WINT$ interrupt will be generated if the $ADC0$ conversion word ($ADC0H:ADC0L$) is within the range defined by $ADC0GTH:ADC0GTL$ and $ADC0LTH:ADC0LTL$ (if $0x0100 < ADC0H:ADC0L < 0x0200$). In the right example, and $AD0WINT$ interrupt will be generated if the $ADC0$ conversion word is outside of the range defined by the $ADC0GT$ and $ADC0LT$ registers (if $ADC0H:ADC0L < 0x0100$ or $ADC0H:ADC0L > 0x0200$). Figure 4.8 shows an example using left-justified data with the same comparison values.

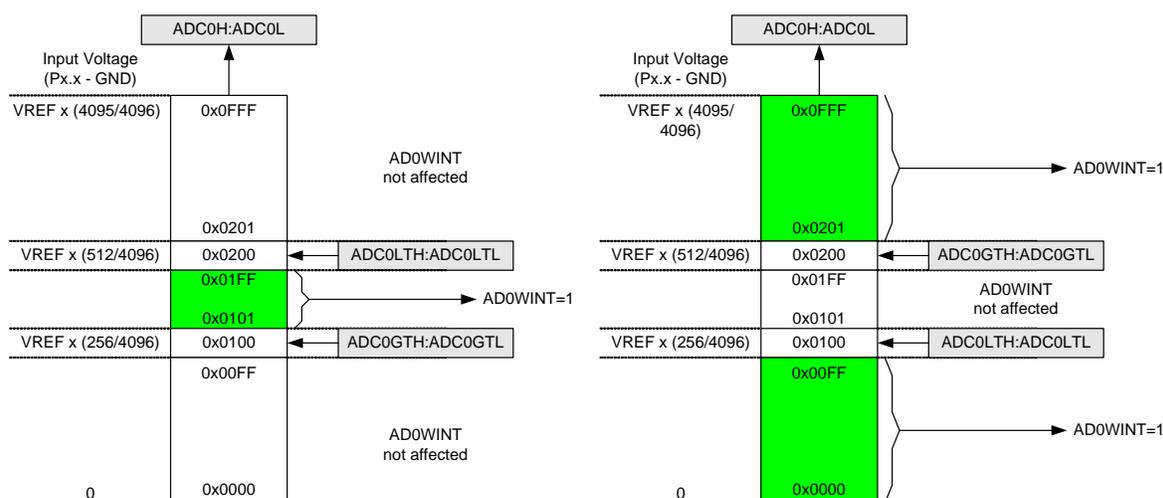


Figure 4.7. ADC Window Compare Example: Right-Justified Single-Ended Data

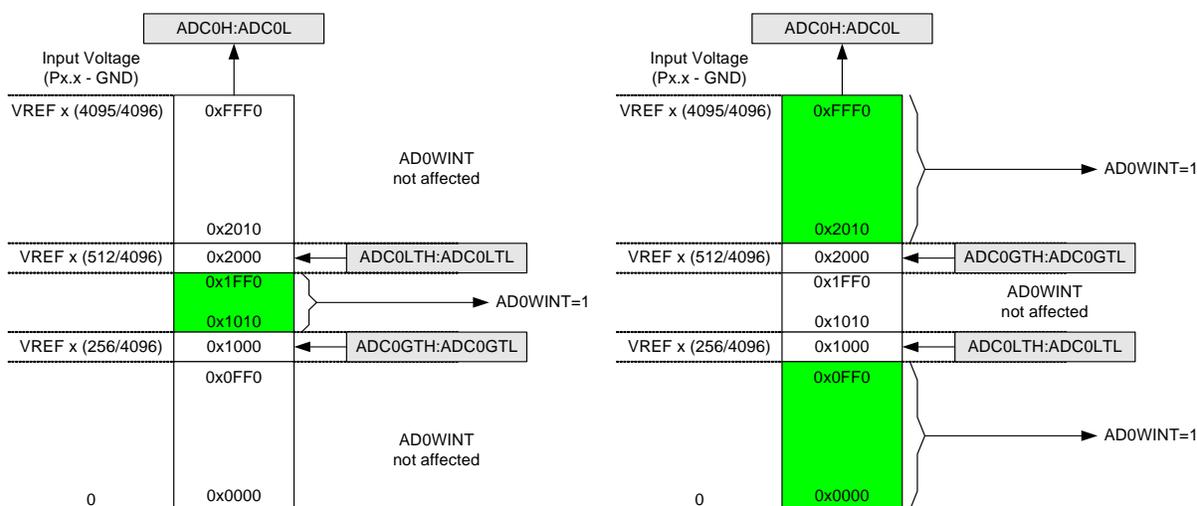


Figure 4.8. ADC Window Compare Example: Left-Justified Single-Ended Data

C8051F52x/F52xA/F53x/F53xA

SFR Definition 8.7. PCON: Power Control

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	STOP	IDLE	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0x87

Bits7–2: RESERVED.

Bit1: STOP: STOP Mode Select.
Writing a 1 to this bit will place the CIP-51 into STOP mode. This bit will always read 0.
1: CIP-51 forced into power-down mode. (Turns off internal oscillator).

Bit0: IDLE: IDLE Mode Select.
Writing a 1 to this bit will place the CIP-51 into IDLE mode. This bit will always read 0.
1: CIP-51 forced into IDLE mode. (Shuts off clock to CPU, but clock to Timers, Interrupts, and all peripherals remain active.)

C8051F52x/F52xA/F53x/F53xA

9.2. Data Memory

The C8051F52x/F52xA/F53x/F53xA includes 256 bytes of internal RAM mapped into the data memory space from 0x00 through 0xFF. The lower 128 bytes of data memory are used for general purpose registers and scratch pad memory. Either direct or indirect addressing may be used to access the lower 128 bytes of data memory. Locations 0x00 through 0x1F are addressable as four banks of general purpose registers, each bank consisting of eight byte-wide registers. The next 16 bytes, locations 0x20 through 0x2F, may either be addressed as bytes or as 128 bit locations accessible with the direct addressing mode.

The upper 128 bytes of data memory are accessible only by indirect addressing. This region occupies the same address space as the Special Function Registers (SFRs) but is physically separate from the SFR space. The addressing mode used by an instruction when accessing locations above 0x7F determines whether the CPU accesses the upper 128 bytes of data memory space or the SFRs. Instructions that use direct addressing will access the SFR space. Instructions using indirect addressing above 0x7F access the upper 128 bytes of data memory. Figure 9.1 illustrates the data memory organization of the C8051F52x/F52xA/F53x/F53xA.

9.3. General Purpose Registers

The lower 32 bytes of data memory (locations 0x00 through 0x1F) may be addressed as four banks of general-purpose registers. Each bank consists of eight byte-wide registers designated R0 through R7. Only one of these banks may be enabled at a time. Two bits in the program status word, RS0 (PSW.3) and RS1 (PSW.4), select the active register bank (see description of the PSW in SFR Definition 8.4. PSW: Program Status Word). This allows fast context switching when entering subroutines and interrupt service routines. Indirect addressing modes use registers R0 and R1 as index registers.

9.4. Bit Addressable Locations

In addition to direct access to data memory organized as bytes, the sixteen data memory locations at 0x20 through 0x2F are also accessible as 128 individually addressable bits. Each bit has a bit address from 0x00 to 0x7F. Bit 0 of the byte at 0x20 has bit address 0x00 while bit 7 of the byte at 0x20 has bit address 0x07. Bit 7 of the byte at 0x2F has bit address 0x7F. A bit access is distinguished from a full byte access by the type of instruction used (bit source or destination operands as opposed to a byte source or destination).

The MCS-51™ assembly language allows an alternate notation for bit addressing of the form XX.B where XX is the byte address and B is the bit position within the byte. For example, the instruction:

```
MOV    C, 22.3h
```

moves the Boolean value at 0x13 (bit 3 of the byte at location 0x22) into the Carry flag.

9.5. Stack

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

9.6. Special Function Registers

The direct-access data memory locations from 0x80 to 0xFF constitute the special function registers (SFRs). The SFRs provide control and data exchange with the CIP-51's resources and peripherals. The CIP-51 duplicates the SFRs found in a typical 8051 implementation as well as implementing additional

C8051F52x/F52xA/F53x/F53xA

18 system clock cycles: 1 clock cycle to detect the interrupt, 5 clock cycles to execute the RETI, 8 clock cycles to complete the DIV instruction, and 4 clock cycles to execute the LCALL to the ISR. If the CPU is executing an ISR for an interrupt with equal or higher priority, the new interrupt will not be serviced until the current ISR completes, including the RETI and following instruction.

Table 10.1. Interrupt Summary

Interrupt Source	Interrupt Vector	Priority Order	Pending Flag	Bit addressable?	Cleared by HW?	Enable Flag	Priority Control
Reset	0x0000	Top	None	N/A	N/A	Always Enabled	Always Highest
External Interrupt 0 ($\overline{\text{INT0}}$)	0x0003	0	IE0 (TCON.1)	Y	Y	EX0 (IE.0)	PX0 (IP.0)
Timer 0 Overflow	0x000B	1	TF0 (TCON.5)	Y	Y	ET0 (IE.1)	PT0 (IP.1)
External Interrupt 1 ($\overline{\text{INT0}}$)	0x0013	2	IE1 (TCON.3)	Y	Y	EX1 (IE.2)	PX1 (IP.2)
Timer 1 Overflow	0x001B	3	TF1 (TCON.7)	Y	Y	ET1 (IE.3)	PT1 (IP.3)
UART	0x0023	4	RI0 (SCON0.0) TI0 (SCON0.1)	Y	N	ES0 (IE.4)	PS0 (IP.4)
Timer 2 Overflow	0x002B	5	TF2H (TMR2CN.7) TF2L (TMR2CN.6)	Y	N	ET2 (IE.5)	PT2 (IP.5)
SPI0	0x0033	6	SPIF (SPI0CN.7) WCOL (SPI0CN.6) MODF (SPI0CN.5) RXOVRN (SPI0CN.4)	Y	N	ESPI0 (IE.6)	PSPI0 (IP.6)
ADC0 Window Comparator	0x003B	7	AD0WINT (ADC0CN.3)	Y	N	EWADC0 (EIE1.0)	PWADC0 (EIP1.0)
ADC0 End of Conversion	0x0043	8	AD0INT (ADC0CN.5)	Y	N	EADC0 (EIE1.1)	PADC0 (EIP1.1)
Programmable Counter Array	0x004B	9	CF (PCA0CN.7) CCFn (PCA0CN.n)	Y	N	EPCA0 (EIE1.2)	PPCA0 (EIP1.2)
Comparator Falling Edge	0x0053	10	CP0FIF (CPT0CN.4)	N	N	ECPF (EIE1.3)	PCPF (EIP1.3)
Comparator Rising Edge	0x005B	11	CP0RIF (CPT0CN.5)	N	N	ECPR (EIE1.4)	PCPR (EIP1.4)
LIN Interrupt	0x0063	12	LININT (LINST.3)	N	N*	ELIN (EIE1.5)	PLIN (EIP1.5)
Voltage Regulator Dropout	0x006B	13	N/A	N/A	N/A	EREG0 (EIE1.6)	PREG0 (EIP1.6)
Port Match	0x0073	14	N/A	N/A	N/A	EMAT (EIE1.7)	PMAT (EIP1.7)

Note: Software must set the RSTINT bit (LINCTRL.3) to clear the LININT flag.

12.2. Flash Write and Erase Guidelines

Any system which contains routines which write or erase Flash memory from software involves some risk that the write or erase routines will execute unintentionally if the CPU is operating outside its specified operating range of V_{DD} , system clock frequency, or temperature. This accidental execution of Flash modifying code can result in alteration of Flash memory contents causing a system failure that is only recoverable by re-Flashing the code in the device.

The following guidelines are recommended for any system which contains routines which write or erase Flash from code.

12.2.1. V_{DD} Maintenance and the V_{DD} monitor

1. If the system power supply is subject to voltage or current "spikes," add sufficient transient protection devices to the power supply to ensure that the supply voltages listed in the Absolute Maximum Ratings table are not exceeded.
2. Make certain that the maximum V_{DD} ramp time specification (if applicable) is met. See Section 20.4 on page 211 for more details on V_{DD} ramp time. If the system cannot meet this ramp time specification, then add an external V_{DD} brownout circuit to the \overline{RST} pin of the device that holds the device in reset until V_{DD} reaches the minimum specified V_{DD} and re-asserts \overline{RST} if V_{DD} drops below that level. $V_{DD}(\text{min})$ is specified in Table 2.2 on page 26.
3. Enable the on-chip V_{DD} monitor (VDDMON0) and enable it as a reset source as early in code as possible. This should be the first set of instructions executed after the Reset Vector. For C-based systems, this will involve modifying the startup code added by the C compiler. See your compiler documentation for more details. Make certain that there are no delays in software between enabling the V_{DD} monitor (VDDMON0) and enabling it as a reset source. Code examples showing this can be found in "AN201: Writing to Flash from Firmware", available from the Silicon Laboratories web site.
4. As an added precaution, explicitly enable the V_{DD} monitor (VDDMON0) and enable the V_{DD} monitor as a reset source inside the functions that write and erase Flash memory. The V_{DD} monitor enable instructions should be placed just after the instruction to set PSWE to a 1, but before the Flash write or erase operation instruction.
5. Make certain that all writes to the RSTSRC (Reset Sources) register use direct assignment operators and explicitly DO NOT use the bit-wise operators (such as AND or OR). For example, "RSTSRC = 0x02" is correct. "RSTSRC |= 0x02" is incorrect.
6. Make certain that all writes to the RSTSRC register explicitly set the PORSF bit to a 1. Areas to check are initialization code which enables other reset sources, such as the Missing Clock Detector or Comparator, for example, and instructions which force a Software Reset. A global search on "RSTSRC" can quickly verify this.

12.2.2. PSWE Maintenance

1. Reduce the number of places in code where the PSWE bit (PSCTL.0) is set to a 1. There should be exactly one routine in code that sets PSWE to a 1 to write Flash bytes and one routine in code that sets PSWE and PSEE both to a 1 to erase Flash pages.
2. Minimize the number of variable accesses while PSWE is set to a 1. Handle pointer address updates and loop variable maintenance outside the "PSWE = 1;... PSWE = 0;" area. Code examples showing this can be found in "AN201: Writing to Flash from Firmware," available from the Silicon Laboratories web site.
3. Disable interrupts prior to setting PSWE to a 1 and leave them disabled until after PSWE has been reset to '0'. Any interrupts posted during the Flash write or erase operation will be serviced in priority order after the Flash operation has been completed and interrupts have been re-enabled by software.
4. Make certain that the Flash write and erase pointer variables are not located in XRAM. See your compiler documentation for instructions regarding how to explicitly locate variables in different memory areas.

12.3. 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 write instruction and read using the MOVC instruction. Note: MOVX read instructions always target XRAM.

Note: See Section “12.1. Programming The Flash Memory” on page 113 for minimum V_{DD} and temperature requirements for flash erase and write operations.

12.4. Security Options

The CIP-51 provides security options to protect the Flash memory from inadvertent modification by software as well as to prevent the viewing of proprietary program code and constants. The Program Store Write Enable (bit PSWE in register PSCTL) and the Program Store Erase Enable (bit PSEE in register PSCTL) bits protect the Flash memory from accidental modification by software. PSWE must be explicitly set to 1 before software can modify the Flash memory; both PSWE and PSEE must be set to 1 before software can erase Flash memory. Additional security features prevent proprietary program code and data constants from being read or altered across the C2 interface.

A Security Lock Byte located at the last byte of Flash user space offers protection of the Flash program memory from access (reads, writes, or erases) by unprotected code or the C2 interface. The Flash security mechanism allows the user to lock n 512-byte Flash pages, starting at page 0 (addresses 0x0000 to 0x01FF), where n is the 1's complement number represented by the Security Lock Byte. **Note that the page containing the Flash Security Lock Byte is unlocked when no other Flash pages are locked (all bits of the Lock Byte are 1) and locked when any other Flash pages are locked (any bit of the Lock Byte is 0).** See example below.

Security Lock Byte:	11111101b
1's Complement:	0000010b
Flash pages locked:	3 (First two Flash pages + Lock Byte Page)
Addresses locked:	0x0000 to 0x03FF (first two Flash pages) 0x1C00 to 0x1DFF in 'F520/0A/1/1A and 'F530/0A/1/1A 0x0C00 to 0x0FFF in 'F523/3A/4/4A and 'F533/3A/4/4A and 0x0600 to 0x07FF in 'F526/6A/7/7A and 'F536/6A/7/7A

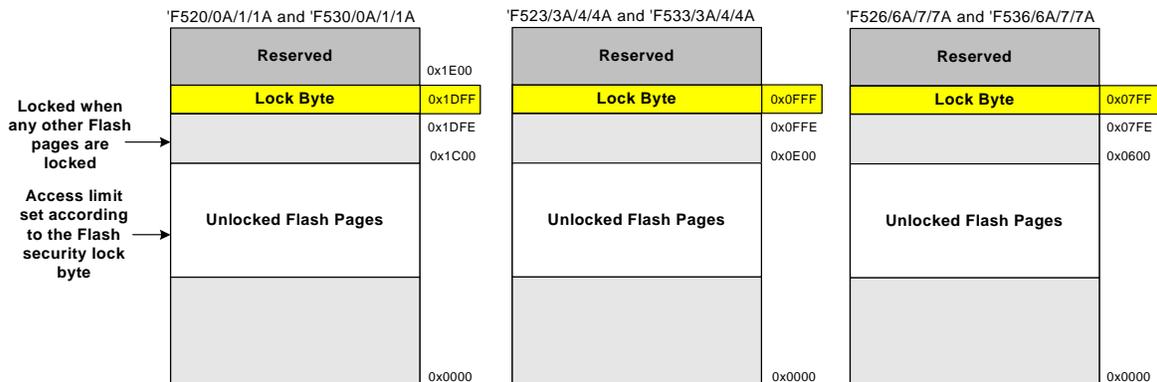


Figure 12.1. Flash Program Memory Map

14.2. External Oscillator Drive Circuit

The external oscillator circuit may drive an external crystal, ceramic resonator, capacitor, or RC network. A CMOS clock may also provide a clock input. For a crystal or ceramic resonator configuration, the crystal/resonator must be wired across the XTAL1 and XTAL2 pins as shown in Option 1 of Figure 14.1. A 10 M Ω resistor also must be wired across the XTAL1 and XTAL2 pins for the crystal/resonator configuration. In RC, capacitor, or CMOS clock configuration, the clock source should be wired to the XTAL2 pin as shown in Option 2, 3, or 4 of Figure 14.1. The type of external oscillator must be selected in the OSCXCN register, and the frequency control bits (XFCN) must be selected appropriately (see SFR Definition 14.4. OSCXCN: External Oscillator Control).

Important Note on External Oscillator Usage: Port pins must be configured when using the external oscillator circuit. When the external oscillator drive circuit is enabled in crystal/resonator mode, Port pins P0.7 and P1.0 ('F53x/'F53xA) or P0.2 and P0.3 ('F52x/'F52xA) are used as XTAL1 and XTAL2 respectively. When the external oscillator drive circuit is enabled in capacitor, RC, or CMOS clock mode, Port pin P1.0 ('F53x/'F53xA) or P0.3 ('F52x/'F52xA) is used as XTAL2. The Port I/O Crossbar should be configured to skip the Port pins used by the oscillator circuit; see Section "13.1. Priority Crossbar Decoder" on page 122 for Crossbar configuration. Additionally, when using the external oscillator circuit in crystal/resonator, capacitor, or RC mode, the associated Port pins should be configured as **analog inputs**. In CMOS clock mode, the associated pin should be configured as a **digital input**. See Section "13.2. Port I/O Initialization" on page 126 for details on Port input mode selection.

14.2.1. Clocking Timers Directly Through the External Oscillator

The external oscillator source divided by eight is a clock option for the timers (Section "18. Timers" on page 182) and the Programmable Counter Array (PCA) (Section "19. Programmable Counter Array (PCA0)" on page 195). When the external oscillator is used to clock these peripherals, but is not used as the system clock, the external oscillator frequency must be less than or equal to the system clock frequency. In this configuration, the clock supplied to the peripheral (external oscillator / 8) is synchronized with the system clock; the jitter associated with this synchronization is limited to ± 0.5 system clock cycles.

14.2.2. External Crystal Example

If a crystal or ceramic resonator is used as an external oscillator source for the MCU, the circuit should be configured as shown in Figure 14.1, Option 1. The External Oscillator Frequency Control value (XFCN) should be chosen from the Crystal column of the table in SFR Definition 14.4. For example, a 12 MHz crystal requires an XFCN setting of 111b.

When the crystal oscillator is first enabled, the oscillator amplitude detection circuit requires a settling time to achieve proper bias. Introducing a delay of 1 ms between enabling the oscillator and checking the XTLVLD bit will prevent a premature switch to the external oscillator as the system clock. Switching to the external oscillator before the crystal oscillator has stabilized can result in unpredictable behavior. The recommended procedure is:

1. Configure XTAL1 and XTAL2 pins by writing 1 to the port latch.
2. Configure XTAL1 and XTAL2 as analog inputs.
3. Enable the external oscillator.
4. Wait at least 1 ms.
5. Poll for XTLVLD => 1.
6. Switch the system clock to the external oscillator.

Note: Tuning-fork crystals may require additional settling time before XTLVLD returns a valid result.

The capacitors shown in the external crystal configuration provide the load capacitance required by the crystal for correct oscillation. These capacitors are "in series" as seen by the crystal and "in parallel" with the stray capacitance of the XTAL1 and XTAL2 pins.

15. UART0

UART0 is an asynchronous, full duplex serial port offering modes 1 and 3 of the standard 8051 UART. Enhanced baud rate support allows a wide range of clock sources to generate standard baud rates (details in Section “15.1. Enhanced Baud Rate Generation” on page 145). Received data buffering allows UART0 to start reception of a second incoming data byte before software has finished reading the previous data byte. (Please refer to Section “20. Device Specific Behavior” on page 210 for more information on the pins associated with the UART interface.)

UART0 has two associated SFRs: Serial Control Register 0 (SCON0) and Serial Data Buffer 0 (SBUF0). The single SBUF0 location provides access to both transmit and receive registers. **Writes to SBUF0 always access the Transmit register. Reads of SBUF0 always access the buffered Receive register; it is not possible to read data from the Transmit register.**

With UART0 interrupts enabled, an interrupt is generated each time a transmit is completed (TI0 is set in SCON0), or a data byte has been received (RI0 is set in SCON0). The UART0 interrupt flags are not cleared by hardware when the CPU vectors to the interrupt service routine. They must be cleared manually by software, allowing software to determine the cause of the UART0 interrupt (transmit complete or receive complete).

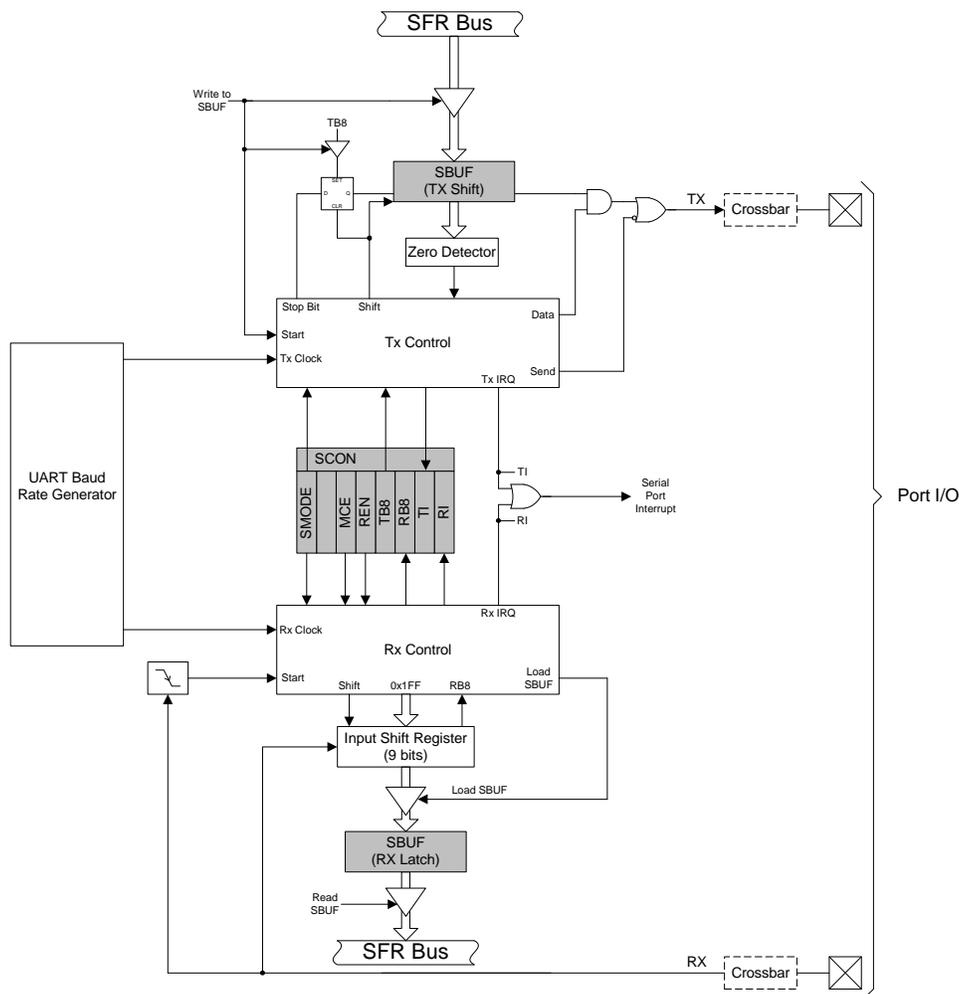


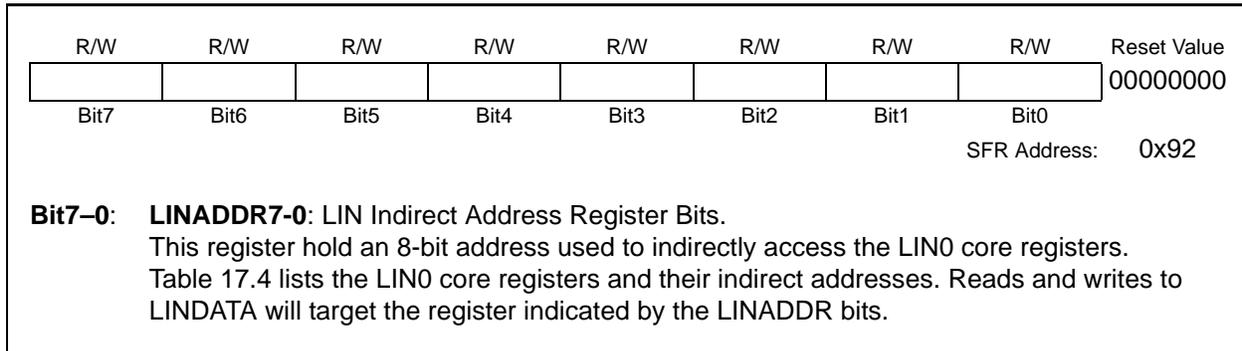
Figure 15.1. UART0 Block Diagram

17.7. LIN Registers

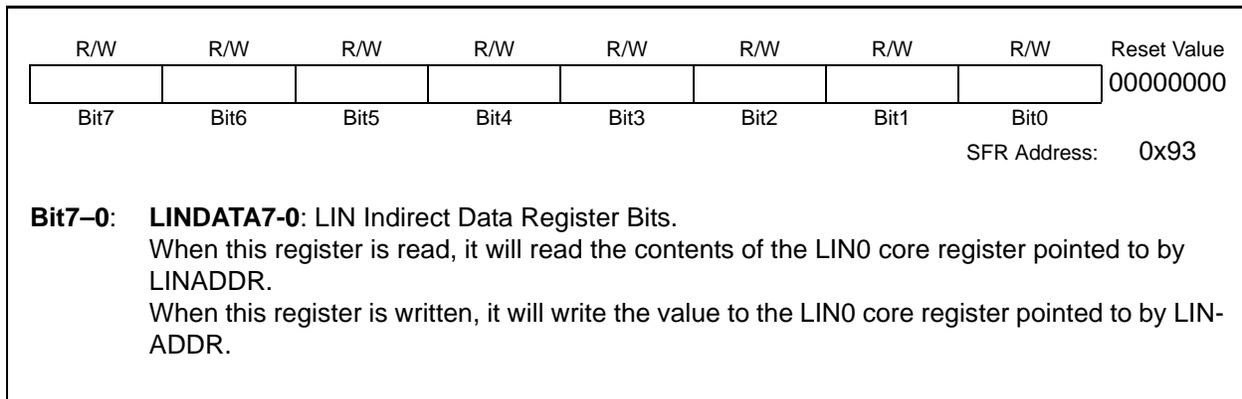
The following Special Function Registers (SFRs) are available:

17.7.1. LIN Direct Access SFR Registers Definition

SFR Definition 17.1. LINADDR: Indirect Address Register



SFR Definition 17.2. LINDATA: LIN Data Register



C8051F52x/F52xA/F53x/F53xA

18.2.2. 8-bit Timers with Auto-Reload

When T2SPLIT is set, Timer 2 operates as two 8-bit timers (TMR2H and TMR2L). Both 8-bit timers operate in auto-reload mode as shown in Figure 18.5. TMR2RLL holds the reload value for TMR2L; TMR2RLH holds the reload value for TMR2H. The TR2 bit in TMR2CN handles the run control for TMR2H. TMR2L is always running when configured for 8-bit Mode.

Each 8-bit timer may be configured to use SYSCLK, SYSCLK divided by 12, or the external oscillator clock source divided by 8. The Timer 2 Clock Select bits (T2MH and T2ML in CKCON) select either SYSCLK or the clock defined by the Timer 2 External Clock Select bit (T2XCLK in TMR2CN), as follows:

T2MH	T2XCLK	TMR2H Clock Source
0	0	SYSCLK / 12
0	1	External Clock / 8
1	X	SYSCLK

T2ML	T2XCLK	TMR2L Clock Source
0	0	SYSCLK / 12
0	1	External Clock / 8
1	X	SYSCLK

The TF2H bit is set when TMR2H overflows from 0xFF to 0x00; the TF2L bit is set when TMR2L overflows from 0xFF to 0x00. When Timer 2 interrupts are enabled (IE.5), an interrupt is generated each time TMR2H overflows. If Timer 2 interrupts are enabled and TF2LEN (TMR2CN.5) is set, an interrupt is generated each time either TMR2L or TMR2H overflows. When TF2LEN is enabled, software must check the TF2H and TF2L flags to determine the source of the Timer 2 interrupt. The TF2H and TF2L interrupt flags are not cleared by hardware and must be manually cleared by software.

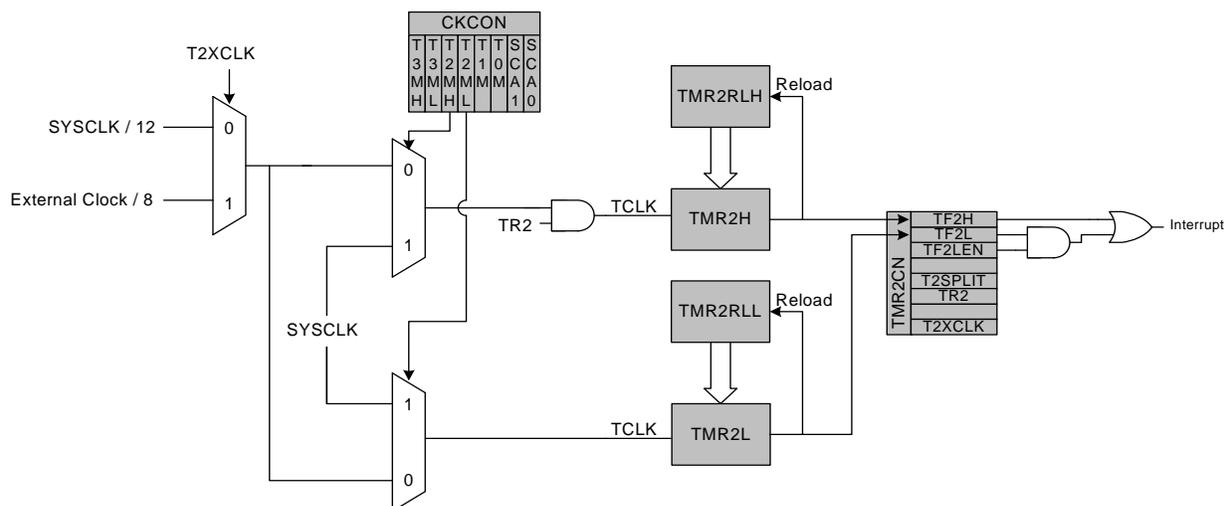
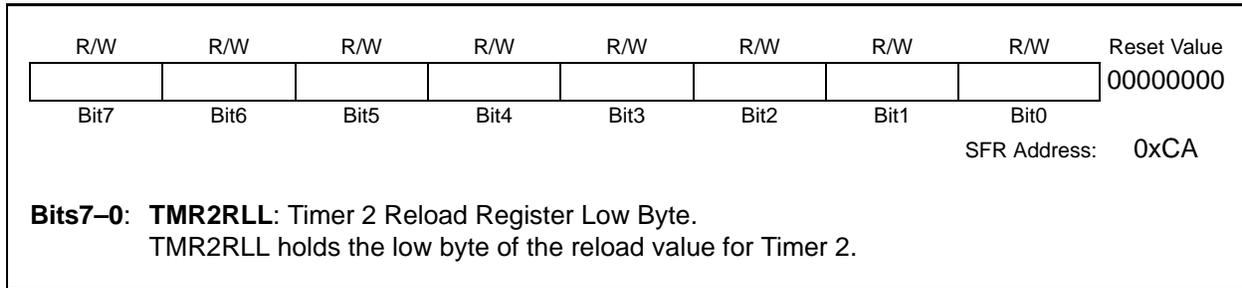


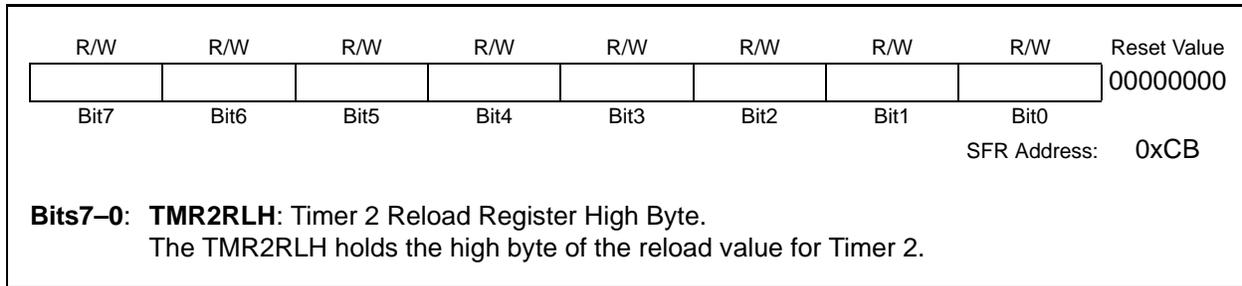
Figure 18.5. Timer 2 8-Bit Mode Block Diagram

C8051F52x/F52xA/F53x/F53xA

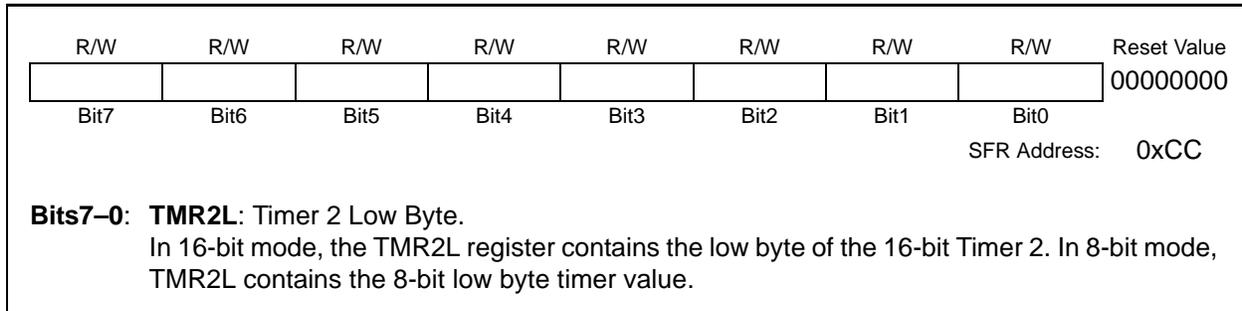
SFR Definition 18.9. TMR2RLL: Timer 2 Reload Register Low Byte



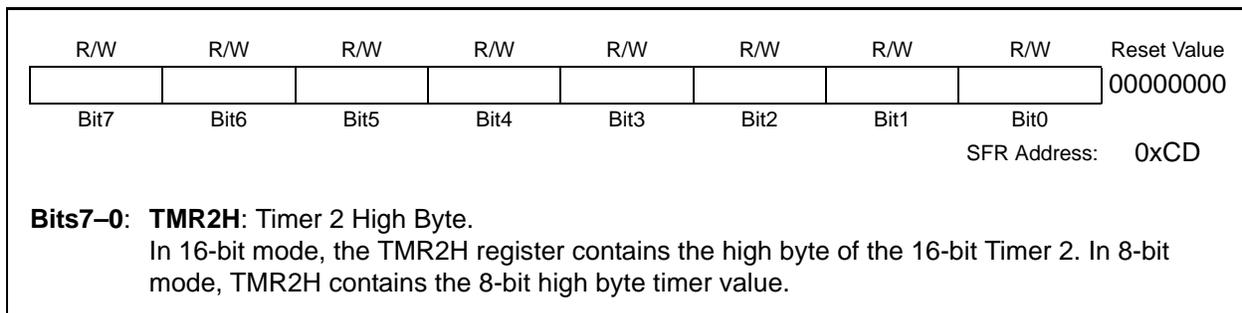
SFR Definition 18.10. TMR2RLH: Timer 2 Reload Register High Byte



SFR Definition 18.11. TMR2L: Timer 2 Low Byte



SFR Definition 18.12. TMR2H Timer 2 High Byte



C8051F52x/F52xA/F53x/F53xA

19.3.1. Watchdog Timer Operation

While the WDT is enabled:

- PCA counter is forced on.
- Writes to PCA0L and PCA0H are not allowed.
- PCA clock source bits (CPS2-CPS0) are frozen.
- PCA Idle control bit (CIDL) is frozen.
- Module 2 is forced into software timer mode.
- Writes to the Module 2 mode register (PCA0CPM2) are disabled.

While the WDT is enabled, writes to the CR bit will not change the PCA counter state; the counter will run until the WDT is disabled. The PCA counter run control (CR) will read zero if the WDT is enabled but user software has not enabled the PCA counter. If a match occurs between PCA0CPH2 and PCA0H while the WDT is enabled, a reset will be generated. To prevent a WDT reset, the WDT may be updated with a write of any value to PCA0CPH2. Upon a PCA0CPH2 write, PCA0H plus the offset held in PCA0CPL2 is loaded into PCA0CPH2 (See Figure 19.10).

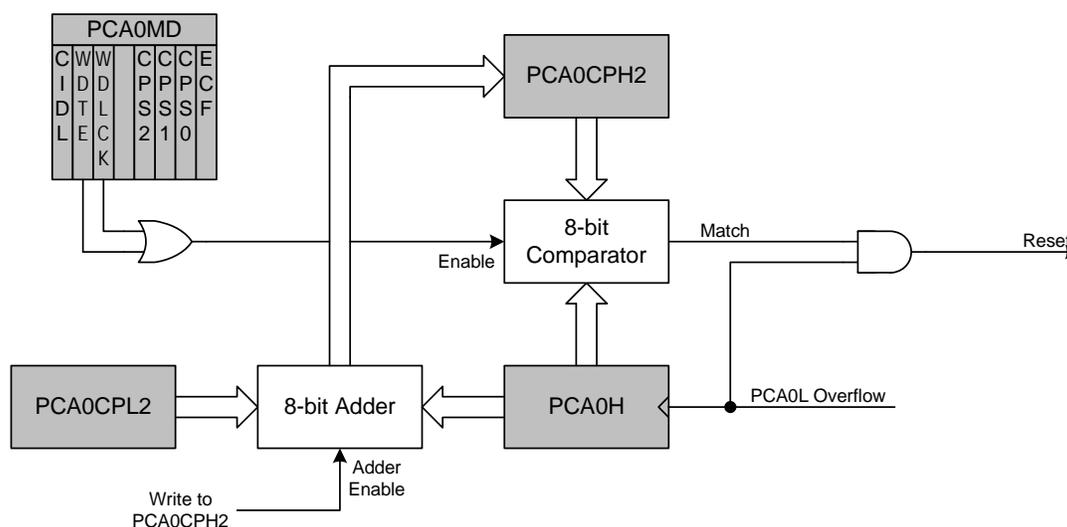


Figure 19.10. PCA Module 2 with Watchdog Timer Enabled

Note that the 8-bit offset held in PCA0CPH2 is compared to the upper byte of the 16-bit PCA counter. This offset value is the number of PCA0L overflows before a reset. Up to 256 PCA clocks may pass before the first PCA0L overflow occurs, depending on the value of the PCA0L when the update is performed. The total offset is then given (in PCA clocks) by Equation 19.4, where PCA0L is the value of the PCA0L register at the time of the update.

$$Offset = (256 \times PCA0CPL2) + (256 - PCA0L)$$

Equation 19.4. Watchdog Timer Offset in PCA Clocks

The WDT reset is generated when PCA0L overflows while there is a match between PCA0CPH2 and PCA0H. Software may force a WDT reset by writing a 1 to the CCF2 flag (PCA0CN.2) while the WDT is enabled.