



Welcome to [E-XFL.COM](https://www.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	LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, Temp Sensor, WDT
Number of I/O	16
Program Memory Size	2KB (2K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 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/c8051f536-c-itr

C8051F52x/F52xA/F53x/F53xA

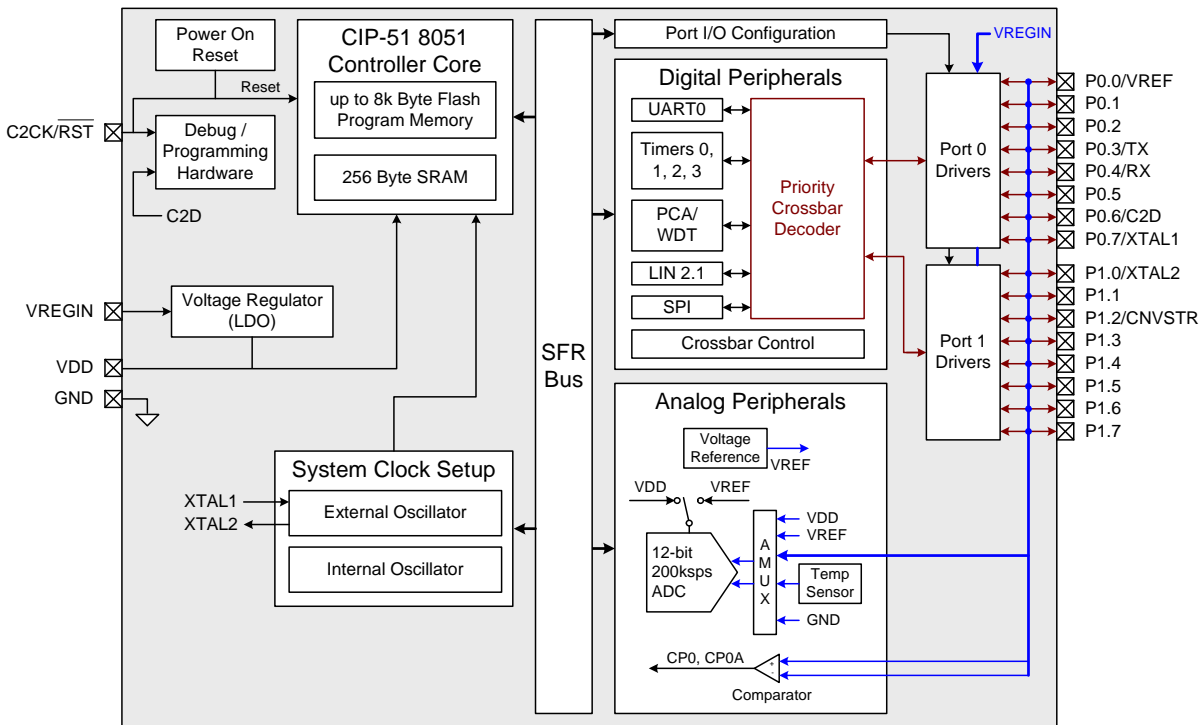


Figure 1.3. C8051F53x Block Diagram (Silicon Revision A)

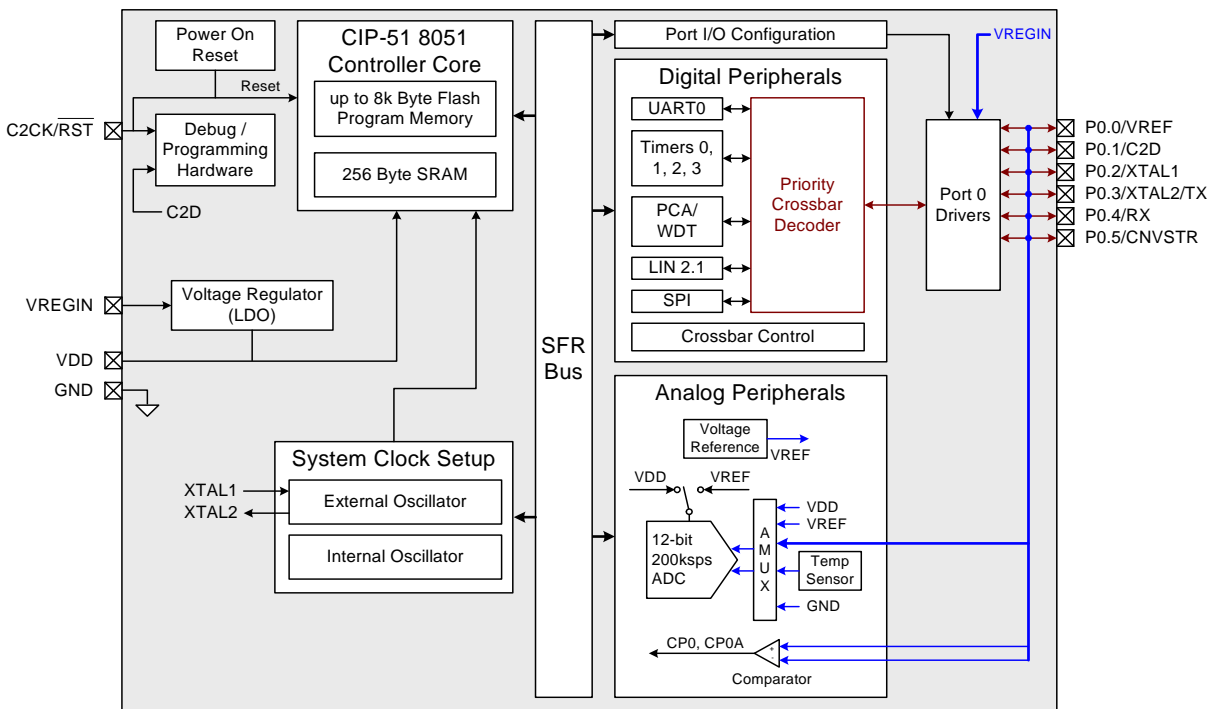


Figure 1.4. C8051F52x Block Diagram (Silicon Revision A)

7. Comparator

C8051F52x/F52xA/F53x/F53xA devices include one on-chip programmable voltage comparator. The Comparator is shown in Figure 7.1.

The Comparator offers programmable response time and hysteresis, an analog input multiplexer, and two outputs that are optionally available at the Port pins: a synchronous “latched” output (CP0), or an asynchronous “raw” output (CP0A). The asynchronous CP0A signal is available even when the system clock is not active. This allows the Comparator to operate and generate an output with the device in STOP or SUSPEND mode. When assigned to a Port pin, the Comparator output may be configured as open drain or push-pull (see Section “13.2. Port I/O Initialization” on page 126). The Comparator may also be used as a reset source (see Section “11.5. Comparator Reset” on page 110).

The Comparator inputs are selected in the CPT0MX register (SFR Definition 7.2). The CMX0P3–CMX0P0 bits select the Comparator0 positive input; the CMX0N3–CMX0N0 bits select the Comparator0 negative input.

Important Note About Comparator Inputs: The Port pins selected as Comparator inputs should be configured as analog inputs in their associated Port configuration register and configured to be skipped by the Crossbar (for details on Port configuration, see Section “13.3. General Purpose Port I/O” on page 128).

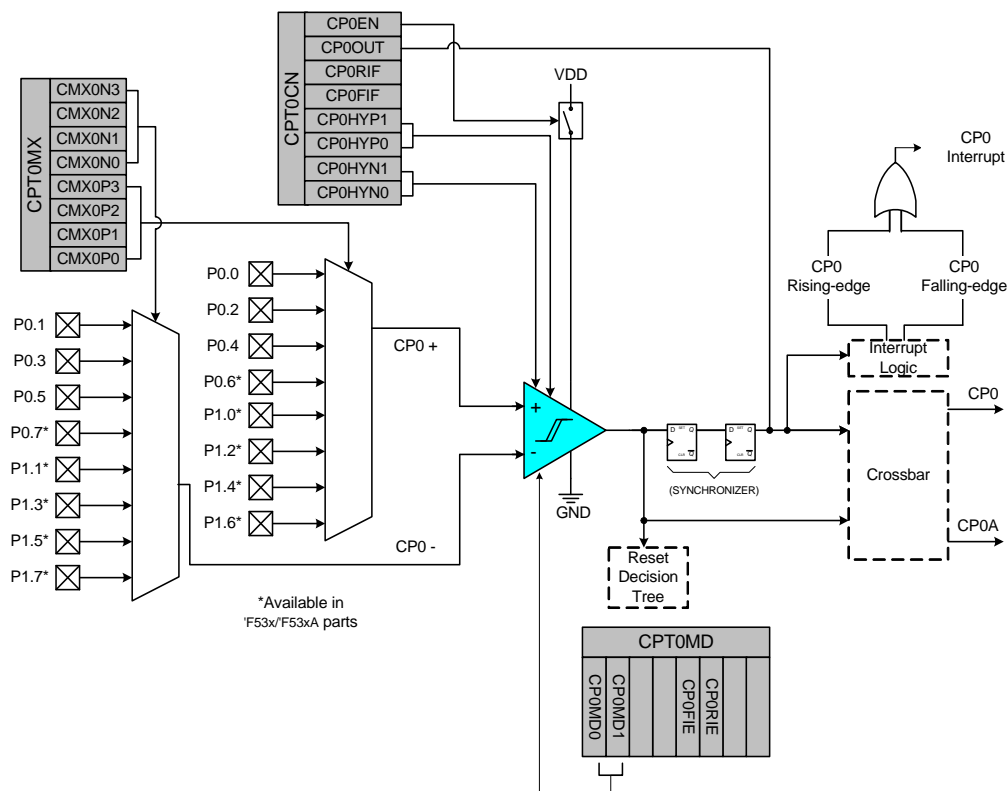


Figure 7.1. Comparator Functional Block Diagram

The Comparator output can be polled in software, used as an interrupt source, internal oscillator suspend awakening source and/or routed to a Port pin. When routed to a Port pin, the Comparator output is available asynchronous or synchronous to the system clock; the asynchronous output is available even in STOP or SUSPEND mode (with no system clock active). When disabled, the Comparator output (if assigned to a Port I/O pin via the Crossbar) defaults to the logic low state, and its supply current falls to

SFR Definition 8.1. SP: Stack Pointer

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0x81

Bits7–0: **SP:** Stack Pointer.
The Stack Pointer holds the location of the top of the stack. The stack pointer is incremented before every PUSH operation. The SP register defaults to 0x07 after reset.

SFR Definition 8.2. DPL: Data Pointer Low Byte

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

SFR Address: 0x82

Bits7–0: **DPL:** Data Pointer Low.
The DPL register is the low byte of the 16-bit DPTR. DPTR is used to access indirectly addressed XRAM and Flash memory.

SFR Definition 8.3. DPH: Data Pointer High Byte

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

SFR Address: 0x83

Bits7–0: **DPH:** Data Pointer High.
The DPH register is the high byte of the 16-bit DPTR. DPTR is used to access indirectly addressed XRAM and Flash memory.

9. Memory Organization and SFRs

The memory organization of the C8051F52x/F52xA/F53x/F53xA is similar to that of a standard 8051. There are two separate memory spaces: program memory and data memory. Program and data memory share the same address space but are accessed via different instruction types. The memory map is shown in Figure 9.1.

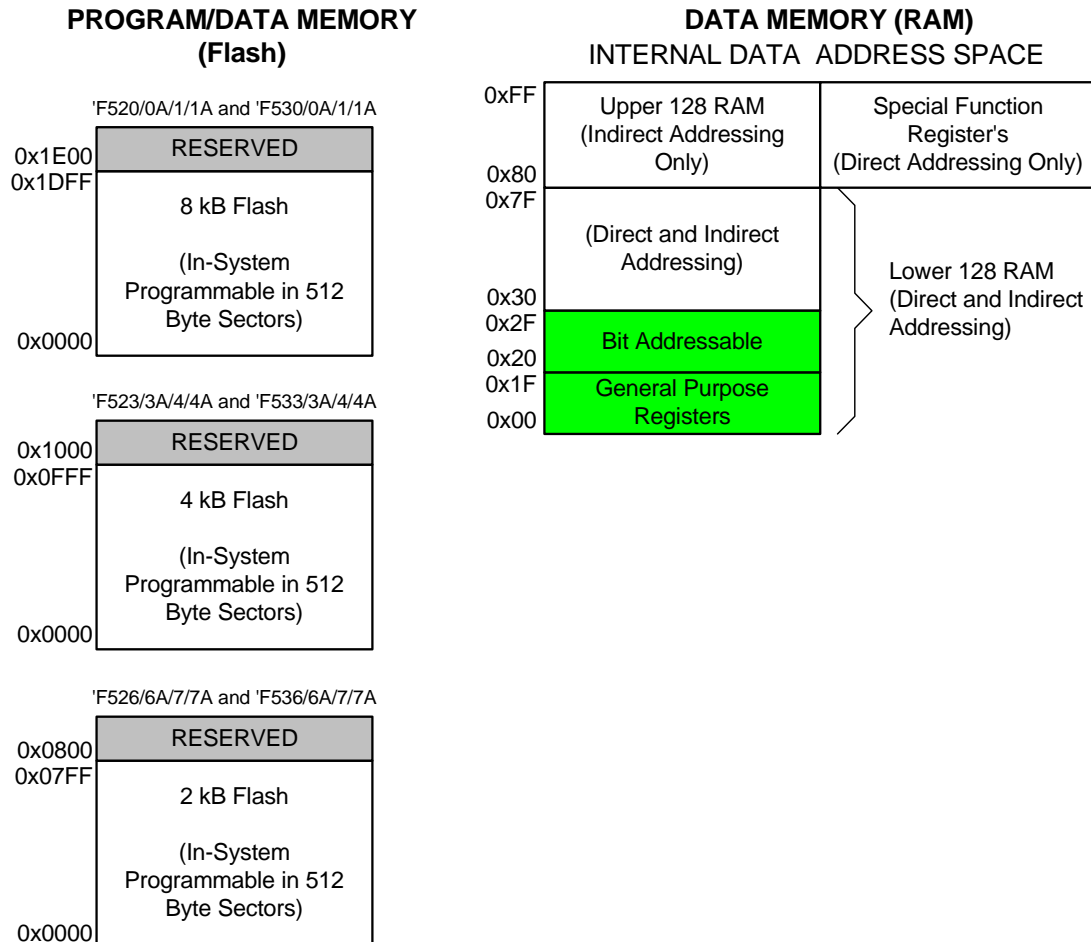


Figure 9.1. Memory Map

9.1. Program Memory

The CIP-51 core has a 64 kB program memory space. The C8051F520/0A/1/1A and C8051F530/0A/1/1A implement 8 kB of this program memory space as in-system, re-programmable Flash memory, organized in a contiguous block from addresses 0x0000 to 0x1FFF. Addresses above 0x1DFF are reserved on the 8 kB devices. The C8051F523/3A/4/4A and C8051F533/3A/4/4A implement 4 kB of Flash from addresses 0x0000 to 0x0FFF. The C8051F526/6A/7/7A and C8051F536/6A/7/7A implement 2 kB of Flash from addresses 0x0000 to 0x07FF.

Program memory is normally assumed to be read-only. However, the C8051F52x/F52xA/F53x/F53xA can write to program memory by setting the Program Store Write Enable bit (PSCTL.0) and using the MOVX write instruction. This feature provides a mechanism for updates to program code and use of the program memory space for non-volatile data storage. Refer to Section “12. Flash Memory” on page 113 for further details.

C8051F52x/F52xA/F53x/F53xA

SFR Definition 10.5. IT01CF: INT0/INT1 Configuration

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
IN1PL	IN1SL2	IN1SL1	IN1SL0	IN0PL	IN0SL2	IN0SL1	IN0SL0	00000001
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xE4

Note: Refer to SFR Definition 18.1. "TCON: Timer Control" on page 186 for INT0/1 edge- or level-sensitive interrupt selection.

Bit 7: **IN1PL:** INT0 Polarity

0: INT0 input is active low.

1: INT0 input is active high.

Bits 6–4: **IN1SL2–0:** INT0 Port Pin Selection Bits

These bits select which Port pin is assigned to INT0. Note that this pin assignment is independent of the Crossbar; INT0 will monitor the assigned Port pin without disturbing the peripheral that has been assigned the Port pin via the Crossbar. The Crossbar will not assign the Port pin to a peripheral if it is configured to skip the selected pin (accomplished by setting to 1 the corresponding bit in register P0SKIP).

IN1SL2-0	INT1 Port Pin
000	P0.0
001	P0.1
010	P0.2
011	P0.3
100	P0.4
101	P0.5
110	P0.6*
111	P0.7*

Note: Available in the C80151F53x/C8051F53xA parts.

Bit 3: **IN0PL:** INT0 Polarity

0: INT0 interrupt is active low.

1: INT0 interrupt is active high.

Bits 2–0: **IN0SL2–0:** INT0 Port Pin Selection Bits

These bits select which Port pin is assigned to INT0. Note that this pin assignment is independent of the Crossbar. INT0 will monitor the assigned Port pin without disturbing the peripheral that has been assigned the Port pin via the Crossbar. The Crossbar will not assign the Port pin to a peripheral if it is configured to skip the selected pin (accomplished by setting to 1 the corresponding bit in register P0SKIP).

IN0SL2-0	INT0 Port Pin
000	P0.0
001	P0.1
010	P0.2
011	P0.3
100	P0.4
101	P0.5
110	P0.6*
111	P0.7*

Note: Available in the C80151F53x/C8051F53xA parts.

12. Flash Memory

On-chip, re-programmable Flash memory is included for program code and non-volatile data storage. The Flash memory can be programmed in-system through the C2 interface or by software using the MOVX write instruction. Once cleared to logic 0, a Flash bit must be erased to set it back to logic 1. Flash bytes would typically be erased (set to 0xFF) before being reprogrammed. The write and erase operations are automatically timed by hardware for proper execution; data polling to determine the end of the write/erase operations is not required. Code execution is stalled during Flash write/erase operations. Refer to Table 2.9 on page 33 for complete Flash memory electrical characteristics.

12.1. Programming The Flash Memory

The simplest means of programming the Flash memory is through the C2 interface using programming tools provided by Silicon Laboratories or a third party vendor. This is the only means for programming a non-initialized device. For details on the C2 commands to program Flash memory, see Section “21. C2 Interface” on page 214.

To protect the integrity of Flash contents, the V_{DD} monitor must be enabled to the higher setting (VDMLVL = '1') and selected as a reset source if software contains routines which erase or write Flash memory. If the V_{DD} monitor is not enabled, any erase or write performed on Flash memory will cause a Flash Error device reset. See Section “11.2. Power-Fail Reset / VDD Monitors (VDDMON0 and VDDMON1)” on page 108 for more information regarding the VDD monitor and the high threshold setting.

The V_{DD} monitor must be enabled before it is selected as a reset source. Selecting the V_{DD} monitor as a reset source before it is enabled and stabilized may cause a system reset. The procedure for re-enabling the V_{DD} monitor and configuring the V_{DD} monitor as a reset source is shown below:

1. Enable the V_{DD} monitor (VDMEN bit in VDDMON = 1).
2. Wait for the V_{DD} monitor to stabilize (see Table 2.8 on page 32 for the V_{DD} Monitor turn-on time). **Note: This delay should be omitted if software contains routines which write or erase Flash memory.**
3. Select the V_{DD} monitor as a reset source (PORSF bit in RSTSRC = 1).

Note: 8-bit MOVX instructions cannot be used to erase or write to Flash memory at addresses higher than 0x00FF.

Important Note: For –I (industrial Grade) parts, flash should be programmed (erase/write) at a minimum temperature of 0 °C for reliable flash operation across the entire temperature range of –40 to +125 °C. This minimum programming temperature does not apply to –A (Automotive Grade) parts.

12.1.1. Flash Lock and Key Functions

Flash writes and erases by user software are protected with a lock and key function. The Flash Lock and Key Register (FLKEY) must be written with the correct key codes, in sequence, before Flash operations may be performed. The key codes are: 0xA5, 0xF1. The timing does not matter, but the codes must be written in order. If the key codes are written out of order, or the wrong codes are written, Flash writes and erases will be disabled until the next system reset. Flash writes and erases will also be disabled if a Flash write or erase is attempted before the key codes have been written properly. The Flash lock resets after each write or erase; the key codes must be written again before a following Flash operation can be performed. The FLKEY register is detailed in SFR Definition 12.2.

SFR Definition 13.5. P0MDOUT: Port0 Output Mode

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: Output Configuration Bits for P0.7–P0.0 (respectively): ignored if corresponding bit in register P0MDIN is logic 0. 0: Corresponding P0.n Output is open-drain. 1: Corresponding P0.n Output is push-pull.								
Note: When SDA and SCL appear on any of the Port I/O, each are open-drain regardless of the value of P0MDOUT.								

SFR Definition 13.6. P0SKIP: Port0 Skip

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: 0xD4								
Bits7–0: P0SKIP[7:0]: Port0 Crossbar Skip Enable Bits. These bits select Port pins to be skipped by the Crossbar Decoder. Port pins used as analog inputs (for ADC or Comparator) or used as special functions (V_{REF} input, external oscillator circuit, CNVSTR input) should be skipped by the Crossbar. 0: Corresponding P0.n pin is not skipped by the Crossbar. 1: Corresponding P0.n pin is skipped by the Crossbar.								

14.2.3. External RC Example

If an RC network is used as an external oscillator source for the MCU, the circuit should be configured as shown in Figure 14.1, Option 2. The capacitor should be no greater than 100 pF; however for very small capacitors, the total capacitance may be dominated by parasitic capacitance in the PCB layout. To determine the required External Oscillator Frequency Control value (XFCN) in the OSCXCN Register, first select the RC network value to produce the desired frequency of oscillation. If the frequency desired is 100 kHz, let $R = 246 \text{ k}\Omega$ and $C = 50 \text{ pF}$:

$$f = 1.23 (10^3) / RC = 1.23 (10^3) / [246 \times 50] = 0.1 \text{ MHz} = 100 \text{ kHz}$$

Referring to the table in SFR Definition 14.4, the required XFCN setting is 010b. Programming XFCN to a higher setting in RC mode will improve frequency accuracy at a slightly increased external oscillator supply current.

14.2.4. External Capacitor Example

If a capacitor is used as an external oscillator for the MCU, the circuit should be configured as shown in Figure 14.1, Option 3. The capacitor should be no greater than 100 pF; however for very small capacitors, the total capacitance may be dominated by parasitic capacitance in the PCB layout. To determine the required External Oscillator Frequency Control value (XFCN) in the OSCXCN Register, select the frequency of oscillation and calculate the capacitance to be used from the equations below. Assume $V_{DD} = 2.1 \text{ V}$ and $f = 75 \text{ kHz}$:

$$f = KF / (C \times V_{DD})$$

$$0.075 \text{ MHz} = KF / (C \times 2.1)$$

Since the frequency of roughly 75 kHz is desired, select the K Factor from the table in SFR Definition 14.4 as $KF = 7.7$:

$$0.075 \text{ MHz} = 7.7 / (C \times 2.1)$$

$$C \times 2.1 = 7.7 / 0.075 \text{ MHz}$$

$$C = 102.6 / 2.0 \text{ pF} = 51.3 \text{ pF}$$

Therefore, the XFCN value to use in this example is 010b.

14.3. System Clock Selection

The internal oscillator requires little start-up time and may be selected as the system clock immediately following the OSCICN write that enables the internal oscillator. External crystals and ceramic resonators typically require a start-up time before they are settled and ready for use. The Crystal Valid Flag (XTLVLD in register OSCXCN) is set to 1 by hardware when the external oscillator is settled. **To avoid reading a false XTLVLD in crystal mode, the software should delay at least 1 ms between enabling the external oscillator and checking XTLVLD.** RC and C modes typically require no startup time.

The CLKSL bit in register CLKSEL selects which oscillator source is used as the system clock. CLKSL must be set to 1 for the system clock to run from the external oscillator; however the external oscillator may still clock certain peripherals (timers, PCA) when another oscillator is selected as the system clock. The system clock may be switched on-the-fly between the internal oscillator and external oscillator, as long as the selected clock source is enabled and has settled.

SFR Definition 14.5. CLKSEL: Clock Select

R	R	R/W	R/W	R	R/W	R/W	R/W	Reset Value
-	-	Reserved	Reserved	-	Reserved	Reserved	CLKSL	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
								SFR Address: 0xA9
Bits7–6: Unused. Read = 00b; Write = don't care. Bits5–4: Reserved. Read = 00b; Must write 00b. Bit3: Unused. Read = 0b; Write = don't care. Bits2–1: Reserved. Read = 00b; Must write 00b. Bit0: CLKSL: System Clock Select 0: Internal Oscillator (as determined by the IFCN bits in register OSCICN). 1: External Oscillator.								

C8051F52x/F52xA/F53x/F53xA

16.1. Signal Descriptions

The four signals used by SPI0 (MOSI, MISO, SCK, NSS) are described below.

16.1.1. Master Out, Slave In (MOSI)

The master-out, slave-in (MOSI) signal is an output from a master device and an input to slave devices. It is used to serially transfer data from the master to the slave. This signal is an output when SPI0 is operating as a master and an input when SPI0 is operating as a slave. Data is transferred most-significant bit first. When configured as a master, MOSI is driven by the MSB of the shift register in both 3- and 4-wire mode.

16.1.2. Master In, Slave Out (MISO)

The master-in, slave-out (MISO) signal is an output from a slave device and an input to the master device. It is used to serially transfer data from the slave to the master. This signal is an input when SPI0 is operating as a master and an output when SPI0 is operating as a slave. Data is transferred most-significant bit first. The MISO pin is placed in a high-impedance state when the SPI module is disabled and when the SPI operates in 4-wire mode as a slave that is not selected. When acting as a slave in 3-wire mode, MISO is always driven by the MSB of the shift register.

16.1.3. Serial Clock (SCK)

The serial clock (SCK) signal is an output from the master device and an input to slave devices. It is used to synchronize the transfer of data between the master and slave on the MOSI and MISO lines. SPI0 generates this signal when operating as a master. The SCK signal is ignored by a SPI slave when the slave is not selected (NSS = 1) in 4-wire slave mode.

16.1.4. Slave Select (NSS)

The function of the slave-select (NSS) signal is dependent on the setting of the NSSMD1 and NSSMD0 bits in the SPI0CN register. There are three possible modes that can be selected with these bits:

1. NSSMD[1:0] = 00: 3-Wire Master or 3-Wire Slave Mode: SPI0 operates in 3-wire mode, and NSS is disabled. When operating as a slave device, SPI0 is always selected in 3-wire mode. Since no select signal is present, SPI0 must be the only slave on the bus in 3-wire mode. This is intended for point-to-point communication between a master and one slave.
2. NSSMD[1:0] = 01: 4-Wire Slave or Multi-Master Mode: SPI0 operates in 4-wire mode, and NSS is enabled as an input. When operating as a slave, NSS selects the SPI0 device. When operating as a master, a 1-to-0 transition of the NSS signal disables the master function of SPI0 so that multiple master devices can be used on the same SPI bus.
3. NSSMD[1:0] = 1x: 4-Wire Master Mode: SPI0 operates in 4-wire mode, and NSS is enabled as an output. The setting of NSSMD0 determines what logic level the NSS pin will output. This configuration should only be used when operating SPI0 as a master device.

See Figure 16.2, Figure 16.3, and Figure 16.4 for typical connection diagrams of the various operational modes. **Note that the setting of NSSMD bits affects the pinout of the device.** When in 3-wire master or 3-wire slave mode, the NSS pin will not be mapped by the crossbar. In all other modes, the NSS signal will be mapped to a pin on the device. See Section “13. Port Input/Output” on page 120 for general purpose port I/O and crossbar information.

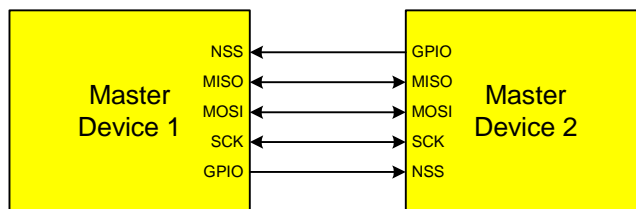


Figure 16.2. Multiple-Master Mode Connection Diagram

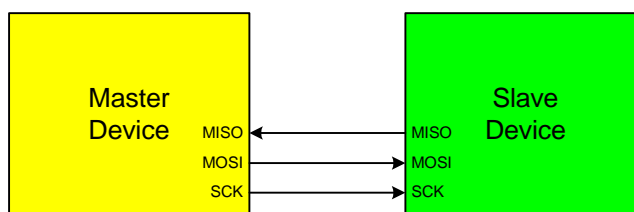


Figure 16.3. 3-Wire Single Master and Slave Mode Connection Diagram

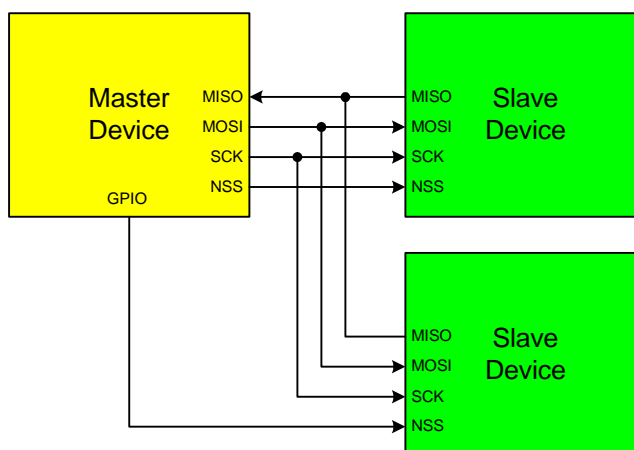


Figure 16.4. 4-Wire Single Master and Slave Mode Connection Diagram

16.3. SPI0 Slave Mode Operation

When SPI0 is enabled and not configured as a master, it will operate as a SPI slave. As a slave, bytes are shifted in through the MOSI pin and out through the MISO pin by a master device controlling the SCK signal. A bit counter in the SPI0 logic counts SCK edges. When 8 bits have been shifted into the shift register, the SPIF flag is set to logic 1, and the byte is copied into the receive buffer. Data is read from the receive buffer by reading SPI0DAT. A slave device cannot initiate transfers. Data to be transferred to the master device is pre-loaded into the shift register by writing to SPI0DAT. Writes to SPI0DAT are double-buffered, and are placed in the transmit buffer first. If the shift register is empty, the contents of the transmit buffer will immediately be transferred into the shift register. When the shift register already contains data, the SPI will load the shift register with the transmit buffer's contents after the last SCK edge of the next (or current) SPI transfer.

C8051F52x/F52xA/F53x/F53xA

SFR Definition 17.5. LIN0DT2: LIN0 Data Byte 2

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	Address: 0x01 (indirect)
Bit7–0: LIN0DT2: LIN Data Byte 2. Serial Data Byte 2 that is received or transmitted across the LIN interface.								

SFR Definition 17.6. LIN0DT3: LIN0 Data Byte 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	Address: 0x02 (indirect)
Bit7–0: LIN0DT3: LIN Data Byte 3. Serial Data Byte 3 that is received or transmitted across the LIN interface.								

SFR Definition 17.7. LIN0DT4: LIN0 Data Byte 4

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	Address: 0x03 (indirect)
Bit7–0: LIN0DT4: LIN Data Byte 4. Serial Data Byte 4 that is received or transmitted across the LIN interface.								

18.1.2. Mode 1: 16-bit Counter/Timer

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

18.1.3. 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. TL0 holds the count and TH0 holds the reload value. When the counter in TL0 overflows from all ones to 0x00, the timer overflow flag TF0 (TCON.5) is set and the counter in TL0 is reloaded from TH0. If Timer 0 interrupts are 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. Setting the TR0 bit (TCON.4) enables the timer when either GATE0 (TMOD.3) is logic 0 or when the input signal INT0 is active as defined by bit IN0PL in register IT01CF (see Section “10.5. External Interrupts” on page 104 for details on the external input signals INT0 and INT0).

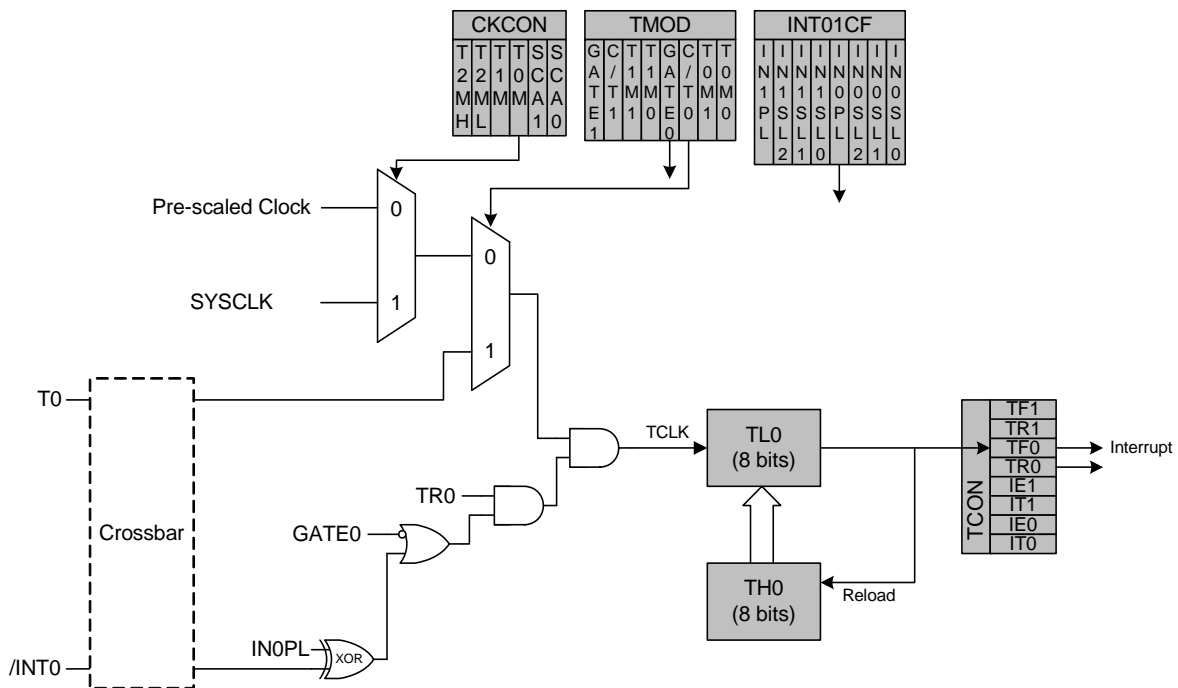


Figure 18.2. T0 Mode 2 Block Diagram

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.

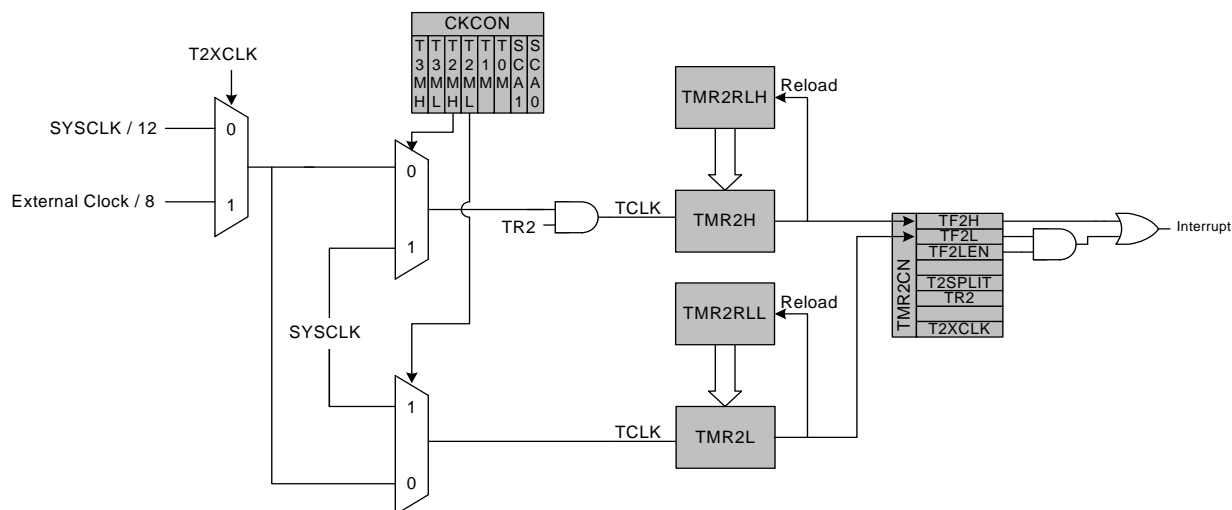


Figure 18.5. Timer 2 8-Bit Mode Block Diagram

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

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
SFR Address: 0xCA								
Bits7–0: TMR2RLL: Timer 2 Reload Register Low Byte. TMR2RLL holds the low byte of the reload value for Timer 2.								

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

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
SFR Address: 0xCB								
Bits7–0: TMR2RLH: Timer 2 Reload Register High Byte. The TMR2RLH holds the high byte of the reload value for Timer 2.								

SFR Definition 18.11. TMR2L: Timer 2 Low Byte

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
SFR Address: 0xCC								
Bits7–0: TMR2L: Timer 2 Low Byte. In 16-bit mode, the TMR2L register contains the low byte of the 16-bit Timer 2. In 8-bit mode, TMR2L contains the 8-bit low byte timer value.								

SFR Definition 18.12. TMR2H Timer 2 High Byte

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
SFR Address: 0xCD								
Bits7–0: TMR2H: Timer 2 High Byte. In 16-bit mode, the TMR2H register contains the high byte of the 16-bit Timer 2. In 8-bit mode, TMR2H contains the 8-bit high byte timer value.								

19.2.2. Software Timer (Compare) Mode

In Software Timer mode, the PCA counter/timer value is compared to the module's 16-bit capture/compare register (PCA0CPHn and PCA0CPLn). When a match occurs, the Capture/Compare Flag (CCFn) in PCA0CN is set to logic 1 and an interrupt request is generated if CCF interrupts are enabled. The CCFn bit is not automatically cleared by hardware when the CPU vectors to the interrupt service routine, and must be cleared by software. Setting the ECOMn and MATn bits in the PCA0CPMn register enables Software Timer mode.

Important Note About Capture/Compare Registers: When writing a 16-bit value to the PCA0 Capture/Compare registers, the low byte should always be written first. Writing to PCA0CPLn clears the ECOMn bit to 0; writing to PCA0CPHn sets ECOMn to 1.

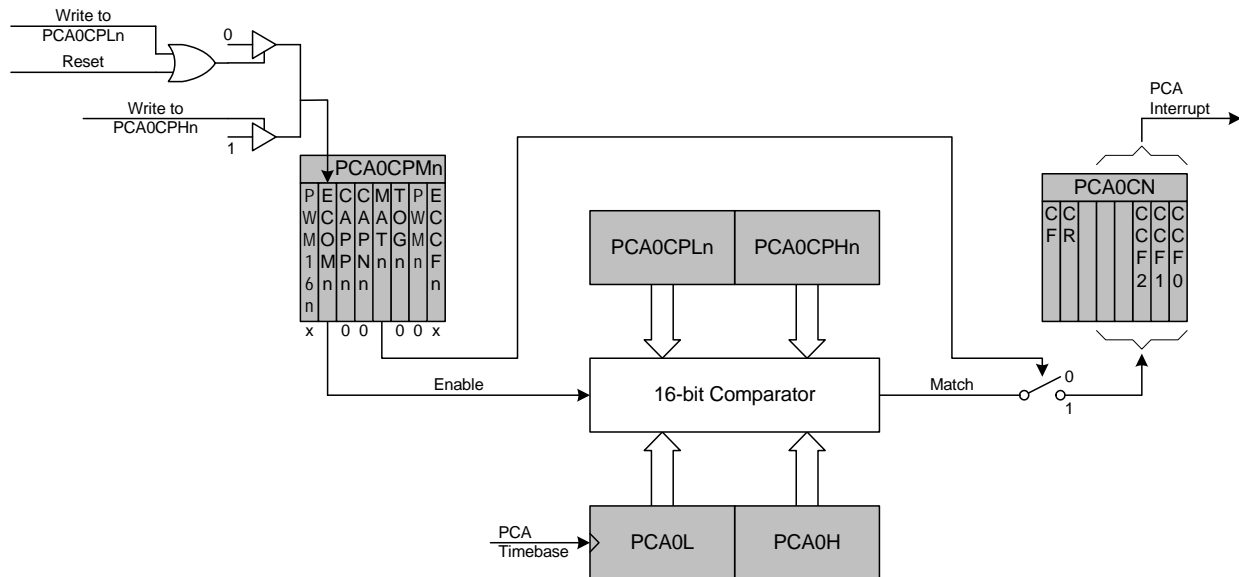


Figure 19.5. PCA Software Timer Mode Diagram

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).

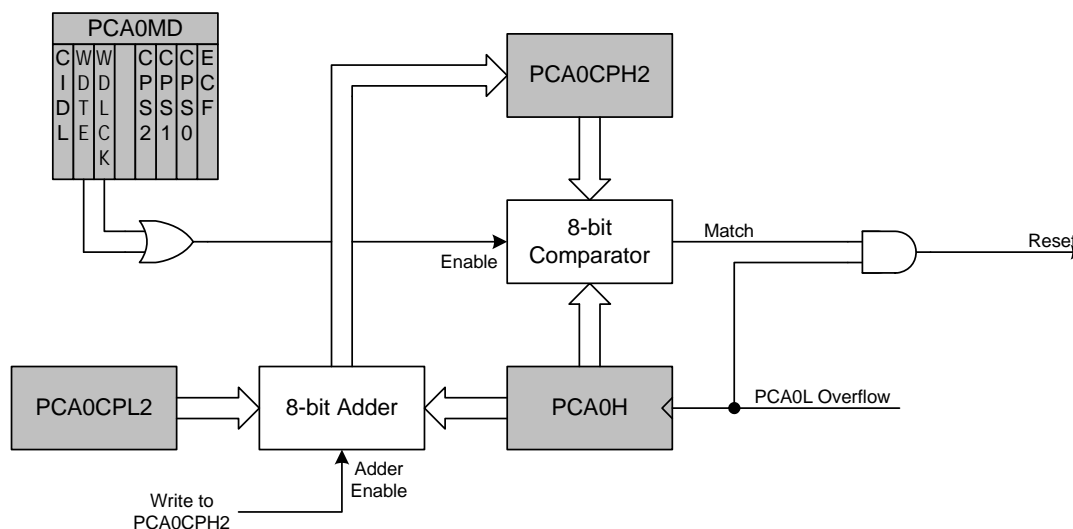


Figure 19.10. PCA Module 2 with Watchdog Timer Enabled

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.

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

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.

SFR Definition 19.2. PCA0MD: PCA Mode

R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	Reset Value
CIDL	WDTE	WDLCK	-	CPS2	CPS1	CPS0	ECF	01000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xD9

- Bit7:** **CIDL:** PCA Counter/Timer Idle Control.
Specifies PCA behavior when CPU is in Idle Mode.
0: PCA continues to function normally while the system controller is in Idle Mode.
1: PCA operation is suspended while the system controller is in Idle Mode.
- Bit6:** **WDTE:** Watchdog Timer Enable
If this bit is set, PCA Module 2 is used as the watchdog timer.
0: Watchdog Timer disabled.
1: PCA Module 2 enabled as Watchdog Timer.
- Bit5:** **WDLCK:** Watchdog Timer Lock
This bit locks/unlocks the Watchdog Timer Enable. When WDLCK is set, the Watchdog Timer may not be disabled until the next system reset.
0: Watchdog Timer Enable unlocked.
1: Watchdog Timer Enable locked.
- Bit4:** **UNUSED.** Read = 0b, Write = don't care.
- Bits3–1:** **CPS2–CPS0:** PCA Counter/Timer Pulse Select.
These bits select the timebase source for the PCA counter.

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 clock divided by 8*
1	1	0	Reserved
1	1	1	Reserved

Note: External clock divided by 8 is synchronized with the system clock.

- Bit0:** **ECF:** PCA Counter/Timer Overflow Interrupt Enable.
This bit sets the masking of the PCA Counter/Timer Overflow (CF) interrupt.
0: Disable the CF interrupt.
1: Enable a PCA Counter/Timer Overflow interrupt request when CF (PCA0CN.7) is set.

Note: When the WDTE bit is set to 1, the PCA0MD register cannot be modified. To change the contents of the PCA0MD register, the Watchdog Timer must first be disabled.

20.5. V_{DD} Monitor (VDDMON0) High Threshold Setting

The calibration behavior of the internal voltage regulator (REG0) and its impact on V_{DD} monitor (VDDMON0) high threshold setting differs between the silicon revisions of C8051F52x/52xA/F53x/F53xA devices.

The following note applies to **Revision A and Revision B devices**: The output of the internal voltage regulator (REG0) is calibrated by the MCU immediately after any reset event. The output of the un-calibrated internal regulator could be below the high threshold setting ($V_{RST-HIGH}$) of the V_{DD} Monitor (VDDMON0). If this is the case and the V_{DD} Monitor is set to the high threshold setting and if the MCU receives a non-power on reset, the MCU will remain in reset until a power-on reset (POR) occurs (i.e. V_{DD} Monitor will keep the device in reset). A POR will force the V_{DD} Monitor to the low threshold setting which is guaranteed to be below the un-calibrated output of the internal regulator. The device will then exit reset and resume normal operation. It is for this reason Silicon Labs strongly recommends that the V_{DD} Monitor is always left in the low threshold setting (i.e., default value upon POR).

When programming the Flash in-system, the V_{DD} Monitor (VDDMON0) must be set to the high threshold setting. For the highest system reliability, the time the V_{DD} Monitor is set to the high threshold setting should be minimized (e.g., setting the V_{DD} Monitor to the high threshold setting just before the Flash write operation and then changing it back to the low threshold setting immediately after the Flash write operation).

The following note applies to **Revision C devices**: The output of the internal voltage regulator (REG0) is calibrated by the MCU immediately after a power-on reset (POR). This calibrated output setting will stay calibrated through any type of reset other than POR. Because of this change in behavior of REG0, the “low threshold” recommendation noted above for Revision A and Revision B devices does not apply to Revision C devices; the V_{DD} Monitor (VDDMON0) can be set to the high threshold as needed depending on the application.

20.6. Reset Low Time

The maximum reset low time differs between the silicon revisions of C8051F52x/52xA/F53x/F53xA devices.

Reset low time is the duration for which the \overline{RST} pin is driven low by an external circuit while power is applied to the device. On Revision A and Revision B devices with assembly build date code earlier than 1124 (year 2011, work week 24), the reset low time should be a maximum of 1 second. For longer reset low times, a percentage of devices within a narrow range of temperatures (a 5 to 10 C window) may “lock up” and fail to execute code. The condition is cleared only by cycling power.

Revision B devices with assembly date code 1124 or later and Revision C devices do not have any restrictions on reset low time.

20.7. Internal Oscillator Suspend Mode

The required bias setting for the internal oscillator before entering suspend mode differs between the silicon revisions of C8051F52x/52xA/F53x/F53xA devices.

On Revision A and Revision B devices, firmware must set the ZTCEN bit in REF0CN (SFR Definition 5.1) before entering suspend mode. If ZTCEN is not set to 1, there is a low probability of the device remaining in suspend even when a wake-up condition is triggered. On Revision C devices, this bit need not be set to 1 before entering suspend mode.



Simplicity Studio

One-click access to MCU and wireless tools, documentation, software, source code libraries & more. Available for Windows, Mac and Linux!



IoT Portfolio
www.silabs.com/IoT



SW/HW
www.silabs.com/simplicity



Quality
www.silabs.com/quality



Support and Community
community.silabs.com

Disclaimer

Silicon Laboratories intends to provide customers with the latest, accurate, and in-depth documentation of all peripherals and modules available for system and software implementers using or intending to use the Silicon Laboratories products. Characterization data, available modules and peripherals, memory sizes and memory addresses refer to each specific device, and "Typical" parameters provided can and do vary in different applications. Application examples described herein are for illustrative purposes only. Silicon Laboratories reserves the right to make changes without further notice and limitation to product information, specifications, and descriptions herein, and does not give warranties as to the accuracy or completeness of the included information. Silicon Laboratories shall have no liability for the consequences of use of the information supplied herein. This document does not imply or express copyright licenses granted hereunder to design or fabricate any integrated circuits. The products must not be used within any Life Support System without the specific written consent of Silicon Laboratories. A "Life Support System" is any product or system intended to support or sustain life and/or health, which, if it fails, can be reasonably expected to result in significant personal injury or death. Silicon Laboratories products are generally not intended for military applications. Silicon Laboratories products shall under no circumstances be used in weapons of mass destruction including (but not limited to) nuclear, biological or chemical weapons, or missiles capable of delivering such weapons.

Trademark Information

Silicon Laboratories Inc., Silicon Laboratories, Silicon Labs, SiLabs and the Silicon Labs logo, CMEMS®, EFM, EFM32, EFR, Energy Micro, Energy Micro logo and combinations thereof, "the world's most energy friendly microcontrollers", Ember®, EZLink®, EZMac®, EZRadio®, EZRadioPRO®, DSPLL®, ISOmodem®, Precision32®, ProSLIC®, SiPHY®, USBXpress® and others are trademarks or registered trademarks of Silicon Laboratories Inc. ARM, CORTEX, Cortex-M3 and THUMB are trademarks or registered trademarks of ARM Holdings. Keil is a registered trademark of ARM Limited. All other products or brand names mentioned herein are trademarks of their respective holders.



Silicon Laboratories Inc.
 400 West Cesar Chavez
 Austin, TX 78701
 USA

<http://www.silabs.com>