

Welcome to E-XFL.COM

What is "[Embedded - Microcontrollers](#)"?

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

Applications of "[Embedded - Microcontrollers](#)"

Details	
Product Status	Not For New Designs
Core Processor	8051
Core Size	8-Bit
Speed	25MHz
Connectivity	SMBus (2-Wire/I ² C), SPI, UART/USART
Peripherals	POR, PWM, Temp Sensor, WDT
Number of I/O	17
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	768 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 16x10b; D/A 1x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	20-VFQFN Exposed Pad
Supplier Device Package	20-MLP (4x4)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f330-gmr

C8051F330/1/2/3/4/5

9.2.5. Stack	78
9.2.6. Special Function Registers.....	79
9.2.7. Register Descriptions	83
9.3. Interrupt Handler	85
9.3.1. MCU Interrupt Sources and Vectors	86
9.3.2. External Interrupts	87
9.3.3. Interrupt Priorities	87
9.3.4. Interrupt Latency	87
9.3.5. Interrupt Register Descriptions.....	89
9.4. Power Management Modes	94
9.4.1. Idle Mode.....	94
9.4.2. Stop Mode	95
10. Reset Sources.....	97
10.1. Power-On Reset	98
10.2. Power-Fail Reset/VDD Monitor	98
10.3. External Reset	99
10.4. Missing Clock Detector Reset	99
10.5. Comparator0 Reset	100
10.6. PCA Watchdog Timer Reset	100
10.7. Flash Error Reset	100
10.8. Software Reset.....	100
11. Flash Memory	103
11.1. Programming The Flash Memory	103
11.1.1. Flash Lock and Key Functions	103
11.1.2. Flash Erase Procedure	103
11.1.3. Flash Write Procedure	104
11.2. Non-volatile Data Storage	104
11.3. Security Options	105
11.4. Flash Write and Erase Guidelines	107
11.4.1. V_{DD} Maintenance and the V_{DD} monitor	107
11.4.2. PSWE Maintenance	107
11.4.3. System Clock	108
12. External RAM	111
13. Oscillators	113
13.1. Programmable Internal High-Frequency (H-F) Oscillator	113
13.2. Programmable Internal Low-Frequency (L-F) Oscillator	115
13.2.1. Calibrating the Internal L-F Oscillator.....	115
13.3. External Oscillator Drive Circuit.....	116
13.3.1. External Crystal Example.....	118
13.3.2. External RC Example.....	120
13.3.3. External Capacitor Example.....	120
13.4. System Clock Selection.....	121
14. Port Input/Output.....	123
14.1. Priority Crossbar Decoder	125
14.2. Port I/O Initialization	127

C8051F330/1/2/3/4/5

C8051F330/1/2/3/4/5

measured from '0' to $V_{REF} \times 1023/1024$. Example codes are shown below for both right-justified and left-justified data. Unused bits in the ADC0H and ADC0L registers are set to '0'.

Input Voltage	Right-Justified ADC0H:ADC0L (AD0LJST = 0)	Left-Justified ADC0H:ADC0L (AD0LJST = 1)
$V_{REF} \times 1023/1024$	0x03FF	0xFFC0
$V_{REF} \times 512/1024$	0x0200	0x8000
$V_{REF} \times 256/1024$	0x0100	0x4000
0	0x0000	0x0000

When in Differential Mode, conversion codes are represented as 10-bit signed 2's complement numbers. Inputs are measured from $-V_{REF}$ to $V_{REF} \times 511/512$. Example codes are shown below for both right-justified and left-justified data. For right-justified data, the unused MSBs of ADC0H are a sign-extension of the data word. For left-justified data, the unused LSBs in the ADC0L register are set to '0'.

Input Voltage	Right-Justified ADC0H:ADC0L (AD0LJST = 0)	Left-Justified ADC0H:ADC0L (AD0LJST = 1)
$V_{REF} \times 511/512$	0x01FF	0x7FC0
$V_{REF} \times 256/512$	0x0100	0x4000
0	0x0000	0x0000
$-V_{REF} \times 256/512$	0xFF00	0xC000
$-V_{REF}$	0xFE00	0x8000

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 "14. Port Input/Output"** on page 123 for more Port I/O configuration details.

5.2. Temperature Sensor

The typical temperature sensor transfer function is shown in Figure 5.2. The output voltage (V_{TEMP}) is the positive ADC input when the temperature sensor is selected by bits AMX0P4–0 in register AMX0P.

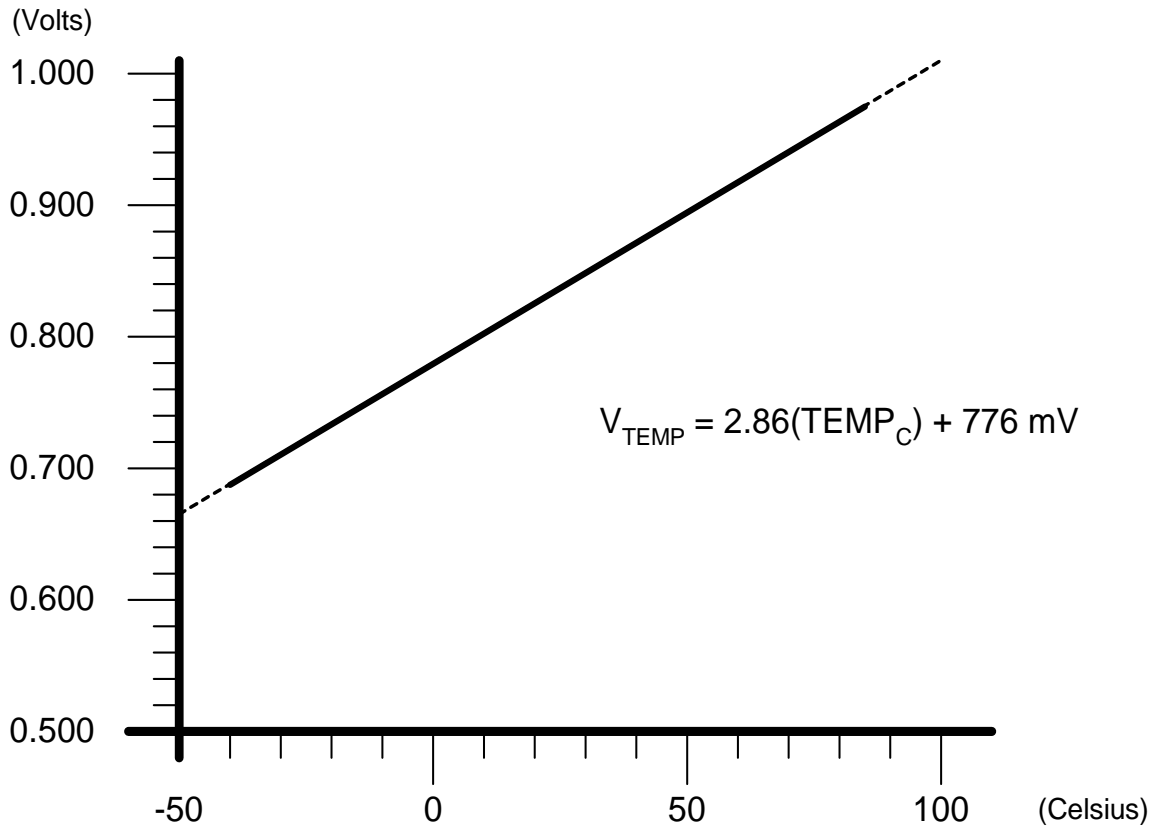


Figure 5.2. Typical Temperature Sensor Transfer Function

5.3. Modes of Operation

ADC0 has a maximum conversion speed of 200 ksps. The ADC0 conversion clock is a divided version of the system clock, determined by the AD0SC bits in the ADC0CF register (system clock divided by (AD0SC + 1) for $0 \leq AD0SC \leq 31$).

5.3.2. Tracking Modes

Each ADC0 conversion must be preceded by a minimum tracking time in order for the converted result to be accurate. The minimum tracking time is given in Table 5.1. The AD0TM bit in register ADC0CN controls the ADC0 track-and-hold mode. In its default state, the ADC0 input is continuously tracked, except when a conversion is in progress. When the AD0TM bit is logic 1, ADC0 operates in low-power track-and-hold mode. In this mode, each conversion is preceded by a tracking period of 3 SAR clocks (after the start-of-conversion signal). When the CNVSTR signal is used to initiate conversions in low-power tracking mode, ADC0 tracks only when CNVSTR is low; conversion begins on the rising edge of CNVSTR (see Figure 5.3). Tracking can also be disabled (shutdown) when the device is in low power standby or sleep modes. Low-power track-and-hold mode is also useful when AMUX settings are frequently changed, due to the settling time requirements described in **Section “5.3.3. Settling Time Requirements” on page 46.**

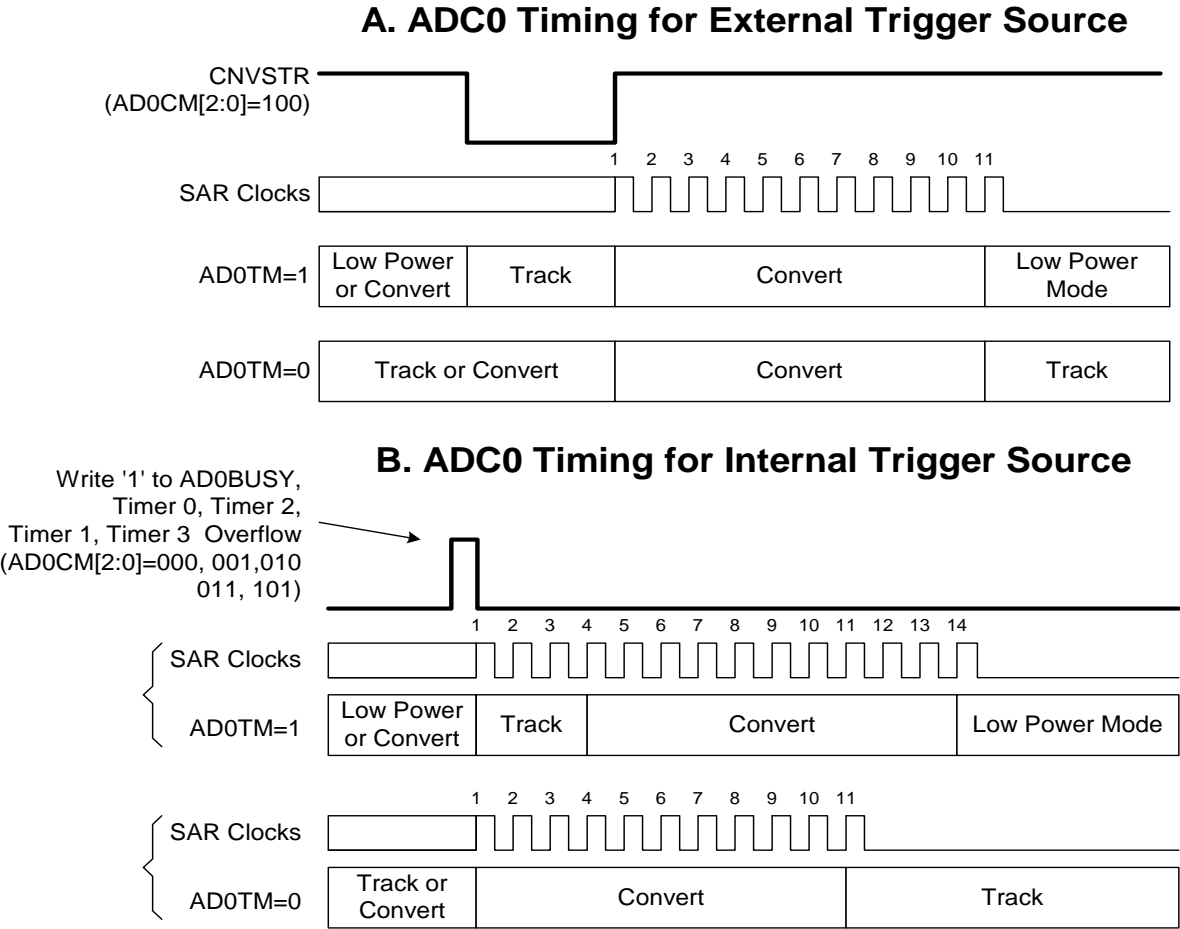
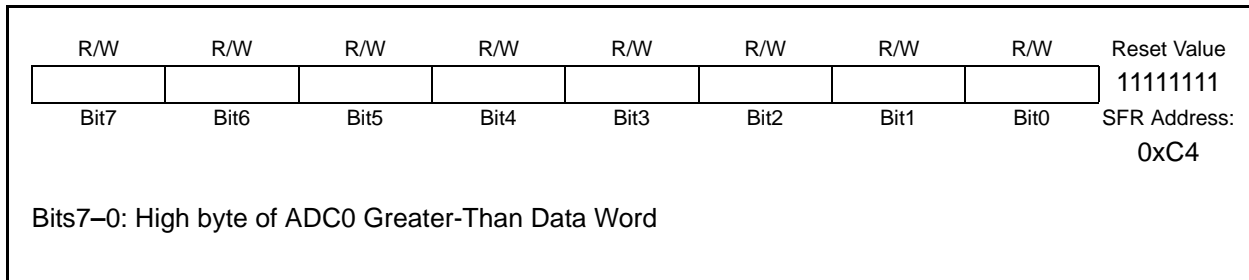


Figure 5.3. 10-Bit ADC Track and Conversion Example Timing

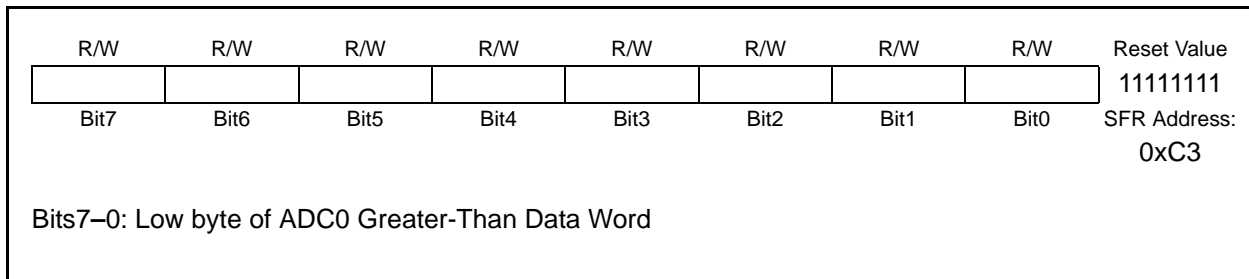
5.4. Programmable Window Detector

The ADC Programmable Window Detector continuously compares the ADC0 output registers to user-programmed limits, and notifies the system when a desired 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 register ADC0CN) can also be used in polled mode. The ADC0 Greater-Than (ADC0GTH, ADC0GTL) and Less-Than (ADC0LTH, ADC0LTL) registers hold the comparison values. The window detector flag can be programmed to indicate when measured data is inside or outside of the user-programmed limits, depending on the contents of the ADC0 Less-Than and ADC0 Greater-Than registers.

SFR Definition 5.7. ADC0GTH: ADC0 Greater-Than Data High Byte



SFR Definition 5.8. ADC0GTL: ADC0 Greater-Than Data Low Byte



C8051F330/1/2/3/4/5

With the CIP-51's maximum system clock at 25 MHz, it has a peak throughput of 25 MIPS. The CIP-51 has a total of 109 instructions. The table below shows the total number of instructions that require each execution time.

Clocks to Execute	1	2	2/3	3	3/4	4	4/5	5	8
Number of Instructions	26	50	5	14	7	3	1	2	1

Programming and Debugging Support

In-system programming of the Flash program memory and communication with on-chip debug support logic is accomplished via the Silicon Labs 2-Wire Development Interface (C2). Note that the re-programmable Flash can also be read and changed a single byte at a time by the application software using the MOVC and MOVX instructions. This feature allows program memory to be used for non-volatile data storage as well as updating program code under software control.

The on-chip debug support logic facilitates full speed in-circuit debugging, allowing the setting of hardware breakpoints, starting, stopping and single stepping through program execution (including interrupt service routines), examination of the program's call stack, and reading/writing the contents of registers and memory. This method of on-chip debugging is completely non-intrusive, requiring no RAM, Stack, timers, or other on-chip resources. C2 details can be found in **Section "20. C2 Interface" on page 209**.

The CIP-51 is supported by development tools from Silicon Labs and third party vendors. Silicon Labs provides an integrated development environment (IDE) including editor, macro assembler, debugger and programmer. The IDE's debugger and programmer interface to the CIP-51 via the C2 interface to provide fast and efficient in-system device programming and debugging. Third party macro assemblers and C compilers are also available.

9.1. Instruction Set

The instruction set of the CIP-51 System Controller is fully compatible with the standard MCS-51™ instruction set. Standard 8051 development tools can be used to develop software for the CIP-51. All CIP-51 instructions are the binary and functional equivalent of their MCS-51™ counterparts, including opcodes, addressing modes and effect on PSW flags. However, instruction timing is different than that of the standard 8051.

9.1.1. Instruction and CPU Timing

In many 8051 implementations, a distinction is made between machine cycles and clock cycles, with machine cycles varying from 2 to 12 clock cycles in length. However, the CIP-51 implementation is based solely on clock cycle timing. All instruction timings are specified in terms of clock cycles.

Due to the pipelined architecture of the CIP-51, most instructions execute in the same number of clock cycles as there are program bytes in the instruction. Conditional branch instructions take one less clock cycle to complete when the branch is not taken as opposed to when the branch is taken. Table 9.1 is the CIP-51 Instruction Set Summary, which includes the mnemonic, number of bytes, and number of clock cycles for each instruction.

9.1.2. MOVX Instruction and Program Memory

The MOVX instruction is typically used to access external data memory (Note: the C8051F330/1/2/3/4/5 does not support off-chip data or program memory). In the CIP-51, the MOVX instruction can be used to access on-chip XRAM or on-chip program memory space implemented as re-programmable Flash memory. The Flash access feature provides a mechanism for the CIP-51 to update program code and use the

Table 9.3. Special Function Registers (Continued)

SFRs are listed in alphabetical order. All undefined SFR locations are reserved

Register	Address	Description	Page
OSCXCN	0xB1	External Oscillator Control	117
P0	0x80	Port 0 Latch	130
P0MDIN	0xF1	Port 0 Input Mode Configuration	130
P0MDOUT	0xA4	Port 0 Output Mode Configuration	131
P0SKIP	0xD4	Port 0 Skip	131
P1	0x90	Port 1 Latch	131
P1MDIN	0xF2	Port 1 Input Mode Configuration	132
P1MDOUT	0xA5	Port 1 Output Mode Configuration	132
P1SKIP	0xD5	Port 1 Skip	132
P2	0xA0	Port 2 Latch	133
P2MDOUT	0xA6	Port 2 Output Mode Configuration	133
PCA0CN	0xD8	PCA Control	205
PCA0CPH0	0xFC	PCA Capture 0 High	208
PCA0CPH1	0xEA	PCA Capture 1 High	208
PCA0CPH2	0xEC	PCA Capture 2 High	208
PCA0CPL0	0xFB	PCA Capture 0 Low	208
PCA0CPL1	0xE9	PCA Capture 1 Low	208
PCA0CPL2	0xEB	PCA Capture 2 Low	208
PCA0CPM0	0xDA	PCA Module 0 Mode Register	207
PCA0CPM1	0xDB	PCA Module 1 Mode Register	207
PCA0CPM2	0xDC	PCA Module 2 Mode Register	207
PCA0H	0xFA	PCA Counter High	208
PCA0L	0xF9	PCA Counter Low	208
PCA0MD	0xD9	PCA Mode	206
PCON	0x87	Power Control	95
PSCTL	0x8F	Program Store R/W Control	108
PSW	0xD0	Program Status Word	84
REF0CN	0xD1	Voltage Reference Control	62
RSTSRC	0xEF	Reset Source Configuration/Status	101
SBUF0	0x99	UART0 Data Buffer	159
SCON0	0x98	UART0 Control	158
SMB0CF	0xC1	SMBus Configuration	142
SMB0CN	0xC0	SMBus Control	144
SMB0DAT	0xC2	SMBus Data	146

C8051F330/1/2/3/4/5

```
EA = 0; // this is a dummy instruction with two-byte opcode.
```

```
; in assembly:
```

```
CLR EA ; clear EA bit.
```

```
CLR EA ; this is a dummy instruction with two-byte opcode.
```

If an interrupt is posted during the execution phase of a "CLR EA" opcode (or any instruction which clears the EA bit), and the instruction is followed by a single-cycle instruction, the interrupt may be taken. However, a read of the EA bit will return a '0' inside the interrupt service routine. When the "CLR EA" opcode is followed by a multi-cycle instruction, the interrupt will not be taken.

Some interrupt-pending flags are automatically cleared by the hardware when the CPU vectors to the ISR. However, most are not cleared by the hardware and must be cleared by software before returning from the ISR. If an interrupt-pending flag remains set after the CPU completes the return-from-interrupt (RETI) instruction, a new interrupt request will be generated immediately and the CPU will re-enter the ISR after the completion of the next instruction.

9.3.1. MCU Interrupt Sources and Vectors

The MCUs support 13 interrupt sources. Software can simulate an interrupt by setting any interrupt-pending flag to logic 1. If interrupts are enabled for the flag, an interrupt request will be generated and the CPU will vector to the ISR address associated with the interrupt-pending flag. MCU interrupt sources, associated vector addresses, priority order and control bits are summarized in Table 9.4 on page 88. 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).

C8051F330/1/2/3/4/5

10.1. Power-On Reset

During power-up, the device is held in a reset state and the $\overline{\text{RST}}$ pin is driven low until V_{DD} settles above V_{RST} . A delay occurs before the device is released from reset; the delay decreases as the V_{DD} ramp time increases (V_{DD} ramp time is defined as how fast V_{DD} ramps from 0 V to V_{RST}). Figure 10.2. plots the power-on and V_{DD} monitor reset timing. The maximum V_{DD} ramp time is 1 ms; slower ramp times may cause the device to be released from reset before V_{DD} reaches the V_{RST} level. For ramp times less than 1 ms, the power-on reset delay (T_{PORDelay}) is typically less than 0.3 ms.

On exit from a power-on reset, the PORSF flag (RSTSRC.1) is set by hardware to logic 1. When PORSF is set, all of the other reset flags in the RSTSRC Register are indeterminate (PORSF is cleared by all other resets). Since all resets cause program execution to begin at the same location (0x0000) software can read the PORSF flag to determine if a power-up was the cause of reset. The content of internal data memory should be assumed to be undefined after a power-on reset. The V_{DD} monitor is disabled following a power-on reset.

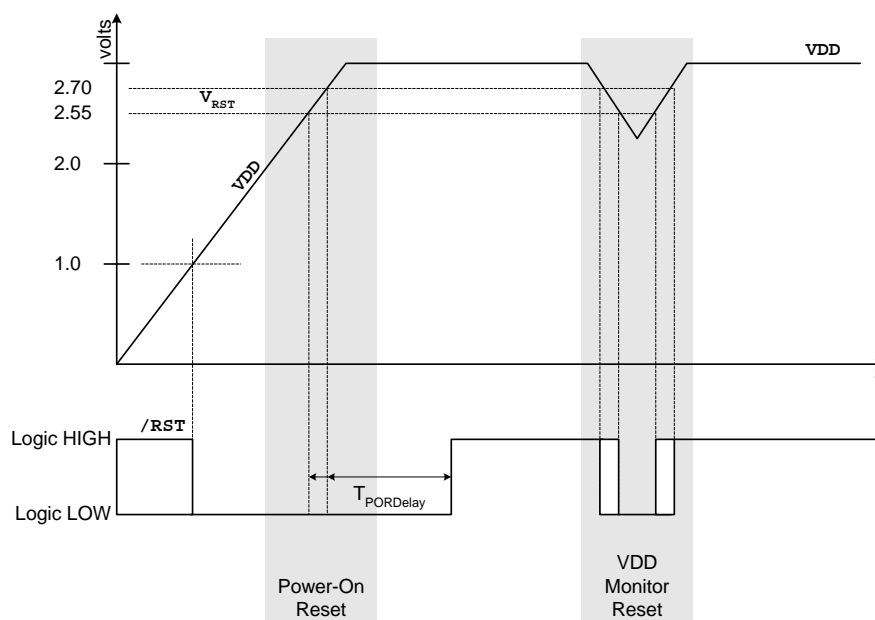


Figure 10.2. Power-On and V_{DD} Monitor Reset Timing

10.2. Power-Fail Reset/ V_{DD} Monitor

When a power-down transition or power irregularity causes V_{DD} to drop below V_{RST} , the power supply monitor will drive the $\overline{\text{RST}}$ pin low and hold the CIP-51 in a reset state (see Figure 10.2). When V_{DD} returns to a level above V_{RST} , the CIP-51 will be released from the reset state. Note that even though internal data memory contents are not altered by the power-fail reset, it is impossible to determine if V_{DD} dropped below the level required for data retention. If the PORSF flag reads '1', the data may no longer be valid. The V_{DD} monitor is disabled after power-on resets; however its defined state (enabled/disabled) is not altered by any other reset source. For example, if the V_{DD} monitor is enabled and a software reset is performed, the V_{DD} monitor will still be enabled after the reset.

13.4. 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. **In crystal mode, to avoid reading a false XTLVLD, 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[1:0] bits in register CLKSEL select which oscillator source is used as the system clock. CLKSL[1:0] must be set to 01b for the system clock to run from the external oscillator; however the external oscillator may still clock certain peripherals (timers, PCA) when the internal oscillator is selected as the system clock. The system clock may be switched on-the-fly between the internal oscillator, external oscillator, and Clock Multiplier so long as the selected clock source is enabled and has settled.

SFR Definition 13.5. CLKSEL: Clock Select

R	R	R	R	R	R	R/W	R/W	Reset Value
-	-	-	-	-	-	CLKSL1	CLKSL0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xA9

Bits7–2: UNUSED. Read = 000000b, Write = don't care.
 Bits1–0: CLKSL[1:0]: System Clock Source Select Bits.
 00: SYSCLK derived from the Internal High-Frequency Oscillator and scaled per the IFCN bits in register OSCICN.
 01: SYSCLK derived from the External Oscillator circuit.
 10: SYSCLK derived from the Internal Low-Frequency Oscillator and scaled per the OSCLD bits in register OSCLCN.
 11: reserved.

Figure 15.4 shows the typical SCL generation described by Equation 15.2. Notice that T_{HIGH} is typically twice as large as T_{LOW} . The actual SCL output may vary due to other devices on the bus (SCL may be extended low by slower slave devices, or driven low by contending master devices). The bit rate when operating as a master will never exceed the limits defined by equation Equation 15.1.

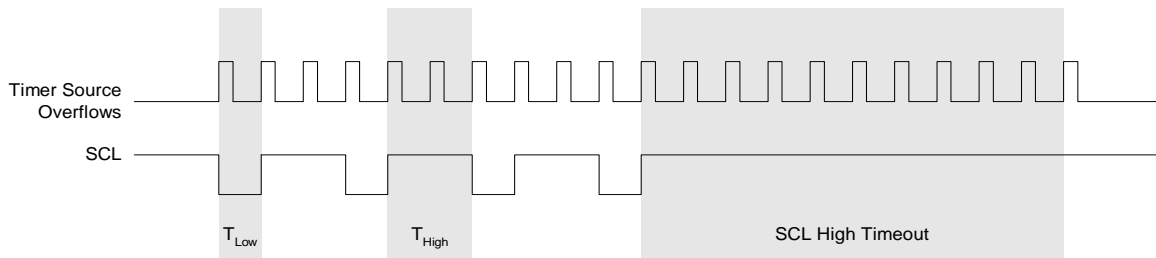


Figure 15.4. Typical SMBus SCL Generation

Setting the EXTHOLD bit extends the minimum setup and hold times for the SDA line. The minimum SDA setup time defines the absolute minimum time that SDA is stable before SCL transitions from low-to-high. The minimum SDA hold time defines the absolute minimum time that the current SDA value remains stable after SCL transitions from high-to-low. EXTHOLD should be set so that the minimum setup and hold times meet the SMBus Specification requirements of 250 ns and 300 ns, respectively. Table 15.2 shows the minimum setup and hold times for the two EXTHOLD settings. Setup and hold time extensions are typically necessary when SYSCLK is above 10 MHz.

Table 15.2. Minimum SDA Setup and Hold Times

EXTHOLD	Minimum SDA Setup Time	Minimum SDA Hold Time
0	$T_{low} - 4$ system clocks or 1 system clock + s/w delay*	3 system clocks
1	11 system clocks	12 system clocks

***Note:** Setup Time for ACK bit transmissions and the MSB of all data transfers. The s/w delay occurs between the time SMB0DAT or ACK is written and when SI is cleared. Note that if SI is cleared in the same write that defines the outgoing ACK value, s/w delay is zero.

With the SMBTOE bit set, Timer 3 should be configured to overflow after 25 ms in order to detect SCL low timeouts (see **Section “15.3.3. SCL Low Timeout” on page 138**). The SMBus interface will force Timer 3 to reload while SCL is high, and allow Timer 3 to count when SCL is low. The Timer 3 interrupt service routine should be used to reset SMBus communication by disabling and re-enabling the SMBus.

SMBus Free Timeout detection can be enabled by setting the SMBFTE bit. When this bit is set, the bus will be considered free if SDA and SCL remain high for more than 10 SMBus clock source periods (see Figure 15.4). When a Free Timeout is detected, the interface will respond as if a STOP was detected (an interrupt will be generated, and STO will be set).

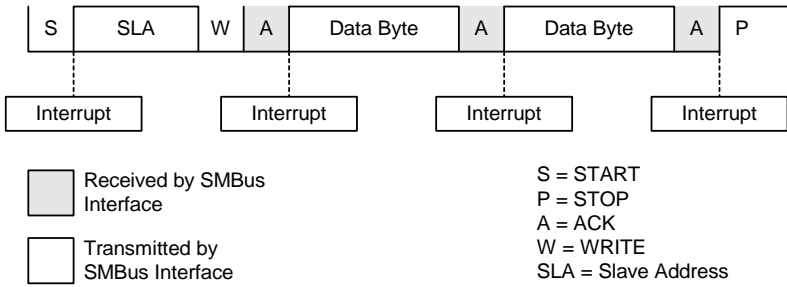


Figure 15.5. Typical Master Transmitter Sequence

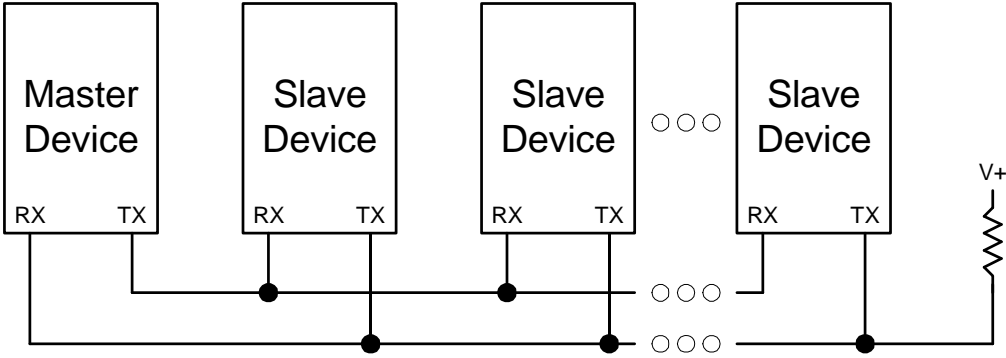


Figure 16.6. UART Multi-Processor Mode Interconnect Diagram

Table 16.3. Timer Settings for Standard Baud Rates Using an External 22.1184 MHz Oscillator

Frequency: 22.1184 MHz							
	Target Baud Rate (bps)	Baud Rate % Error	Oscillator Divide Factor	Timer Clock Source	SCA1–SCA0 (pre-scale select) ¹	T1M ¹	Timer 1 Reload Value (hex)
SYSCLK from External Osc.	230400	0.00%	96	SYSCLK	XX ²	1	0xD0
	115200	0.00%	192	SYSCLK	XX	1	0xA0
	57600	0.00%	384	SYSCLK	XX	1	0x40
	28800	0.00%	768	SYSCLK / 12	00	0	0xE0
	14400	0.00%	1536	SYSCLK / 12	00	0	0xC0
	9600	0.00%	2304	SYSCLK / 12	00	0	0xA0
	2400	0.00%	9216	SYSCLK / 48	10	0	0xA0
1200	0.00%	18432	SYSCLK / 48	10	0	0x40	
SYSCLK from Internal Osc.	230400	0.00%	96	EXTCLK / 8	11	0	0xFA
	115200	0.00%	192	EXTCLK / 8	11	0	0xF4
	57600	0.00%	384	EXTCLK / 8	11	0	0xE8
	28800	0.00%	768	EXTCLK / 8	11	0	0xD0
	14400	0.00%	1536	EXTCLK / 8	11	0	0xA0
	9600	0.00%	2304	EXTCLK / 8	11	0	0x70

Notes:

1. SCA1–SCA0 and T1M bit definitions can be found in **Section 18.1**.
2. X = Don't care.

Table 16.4. Timer Settings for Standard Baud Rates Using an External 18.432 MHz Oscillator

Frequency: 18.432 MHz							
	Target Baud Rate (bps)	Baud Rate % Error	Oscillator Divide Factor	Timer Clock Source	SCA1–SCA0 (pre-scale select) ¹	T1M ¹	Timer 1 Reload Value (hex)
SYSCLK from External Osc.	230400	0.00%	80	SYSCLK	XX ²	1	0xD8
	115200	0.00%	160	SYSCLK	XX	1	0xB0
	57600	0.00%	320	SYSCLK	XX	1	0x60
	28800	0.00%	640	SYSCLK / 4	01	0	0xB0
	14400	0.00%	1280	SYSCLK / 4	01	0	0x60
	9600	0.00%	1920	SYSCLK / 12	00	0	0xB0
	2400	0.00%	7680	SYSCLK / 48	10	0	0xB0
	1200	0.00%	15360	SYSCLK / 48	10	0	0x60
SYSCLK from Internal Osc.	230400	0.00%	80	EXTCLK / 8	11	0	0xFB
	115200	0.00%	160	EXTCLK / 8	11	0	0xF6
	57600	0.00%	320	EXTCLK / 8	11	0	0xEC
	28800	0.00%	640	EXTCLK / 8	11	0	0xD8
	14400	0.00%	1280	EXTCLK / 8	11	0	0xB0
	9600	0.00%	1920	EXTCLK / 8	11	0	0x88

Notes:

1. SCA1–SCA0 and T1M bit definitions can be found in **Section 18.1**.
2. X = Don't care.

17. Enhanced Serial Peripheral Interface (SPI0)

The Enhanced Serial Peripheral Interface (SPI0) provides access to a flexible, full-duplex synchronous serial bus. SPI0 can operate as a master or slave device in both 3-wire or 4-wire modes, and supports multiple masters and slaves on a single SPI bus. The slave-select (NSS) signal can be configured as an input to select SPI0 in slave mode, or to disable Master Mode operation in a multi-master environment, avoiding contention on the SPI bus when more than one master attempts simultaneous data transfers. NSS can also be configured as a chip-select output in master mode, or disabled for 3-wire operation. Additional general purpose port I/O pins can be used to select multiple slave devices in master mode.

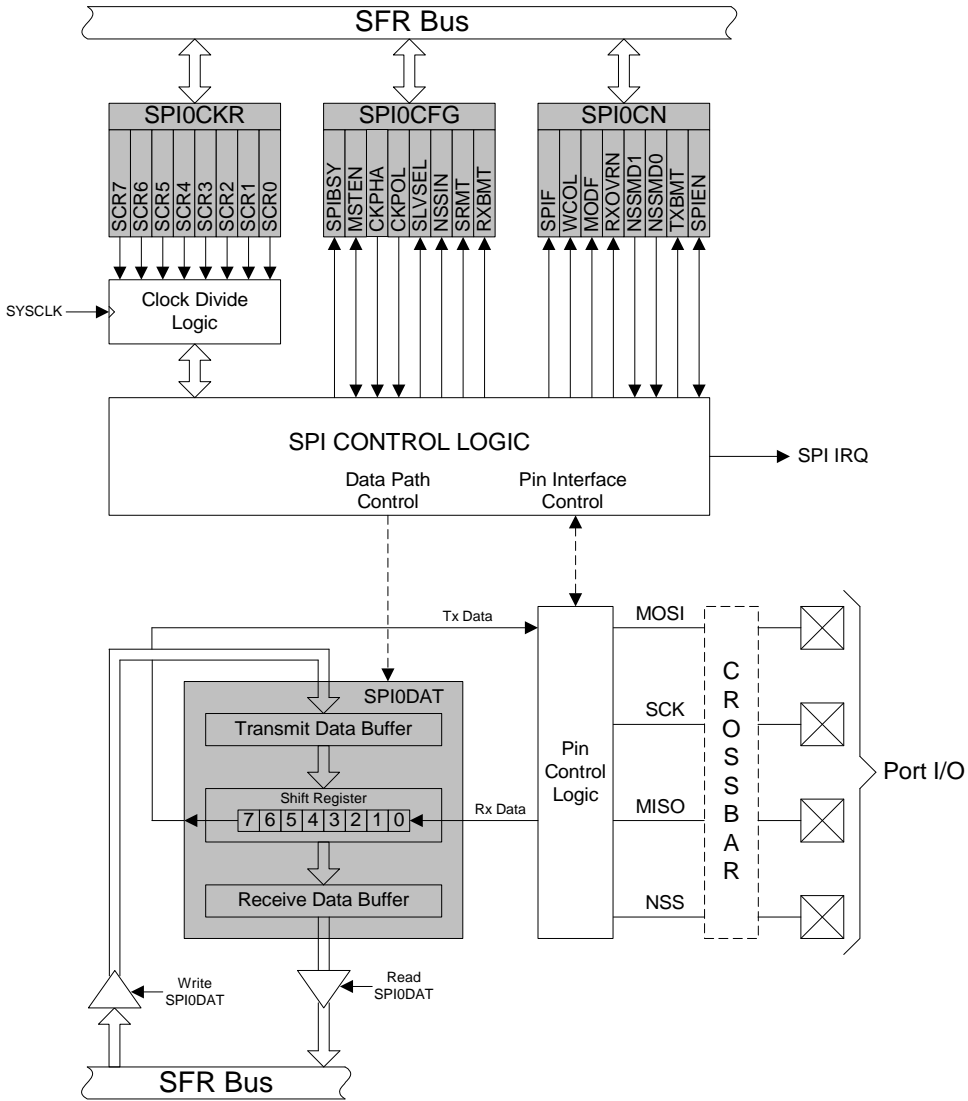


Figure 17.1. SPI Block Diagram

C8051F330/1/2/3/4/5

17.5. Serial Clock Timing

Four combinations of serial clock phase and polarity can be selected using the clock control bits in the SPI0 Configuration Register (SPI0CFG). The CKPHA bit (SPI0CFG.5) selects one of two clock phases (edge used to latch the data). The CKPOL bit (SPI0CFG.4) selects between an active-high or active-low clock. Both master and slave devices must be configured to use the same clock phase and polarity. SPI0 should be disabled (by clearing the SPIEN bit, SPI0CN.0) when changing the clock phase or polarity. The clock and data line relationships for master mode are shown in Figure 17.5. For slave mode, the clock and data relationships are shown in Figure 17.6 and Figure 17.7. Note that CKPHA must be set to '0' on both the master and slave SPI when communicating between two of the following devices: C8051F04x, C8051F06x, C8051F12x, C8051F31x, C8051F32x, and C8051F33x

The SPI0 Clock Rate Register (SPI0CKR) as shown in SFR Definition 17.3 controls the master mode serial clock frequency. This register is ignored when operating in slave mode. When the SPI is configured as a master, the maximum data transfer rate (bits/sec) is one-half the system clock frequency or 12.5 MHz, whichever is slower. When the SPI is configured as a slave, the maximum data transfer rate (bits/sec) for full-duplex operation is 1/10 the system clock frequency, provided that the master issues SCK, NSS (in 4-wire slave mode), and the serial input data synchronously with the slave's system clock. If the master issues SCK, NSS, and the serial input data asynchronously, the maximum data transfer rate (bits/sec) must be less than 1/10 the system clock frequency. In the special case where the master only wants to transmit data to the slave and does not need to receive data from the slave (i.e. half-duplex operation), the SPI slave can receive data at a maximum data transfer rate (bits/sec) of 1/4 the system clock frequency. This is provided that the master issues SCK, NSS, and the serial input data synchronously with the slave's system clock.

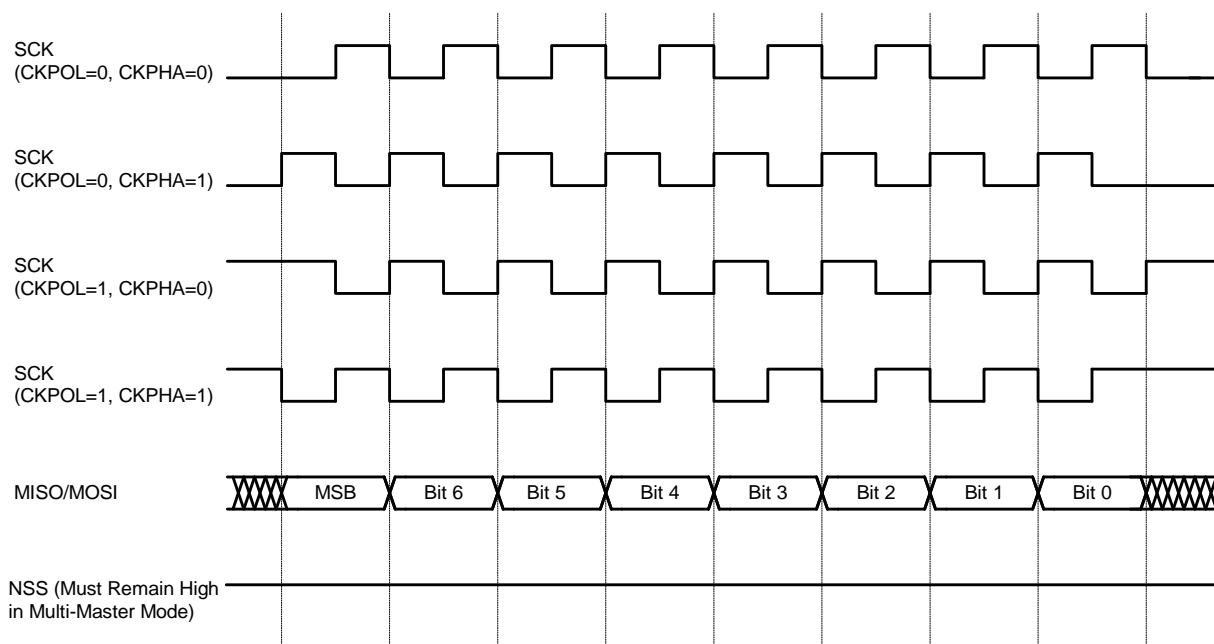


Figure 17.5. Master Mode Data/Clock Timing

C8051F330/1/2/3/4/5

SFR Definition 18.4. TL0: Timer 0 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: 0x8A

Bits 7–0: TL0: Timer 0 Low Byte.
The TL0 register is the low byte of the 16-bit Timer 0.

SFR Definition 18.5. TL1: Timer 1 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: 0x8B

Bits 7–0: TL1: Timer 1 Low Byte.
The TL1 register is the low byte of the 16-bit Timer 1.

SFR Definition 18.6. TH0: Timer 0 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: 0x8C

Bits 7–0: TH0: Timer 0 High Byte.
The TH0 register is the high byte of the 16-bit Timer 0.

SFR Definition 18.7. TH1: Timer 1 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: 0x8D

Bits 7–0: TH1: Timer 1 High Byte.
The TH1 register is the high byte of the 16-bit Timer 1.

C8051F330/1/2/3/4/5

DOCUMENT CHANGE LIST

Revision 1.3 to Revision 1.4

- Removed references to C8051F330D throughout the data sheet because the 'F330D device is functionally identical to the C8051F330 device (these two part numbers differ by package type only).
- Updated titles of Chapters 5, 6, and 7 to show supported devices.
- Updated Table 1.1, "Product Selection Guide," on page 18.
 - Added ordering part number information for lead-free parts.
- Added Table 3.2, "Index to Electrical Characteristics Tables," on page 34
- Added Table 11.2, "Flash Security Summary," on page 106 for clarity, replacing the Flash security summaries text.

Revision 1.4 to Revision 1.5

- Updated Table 3.1 - Added supply current data from characterization.
- Updated Table 5.1 - Added MIN/MAX numbers for ADC Offset and Full Scale Error.
- Fixed SFR Definition 8.2 - Typo in bit descriptions - "2-0" changed to "3-0".
- Fixed SFR Definition 9.4 - Text at bottom of figure was cut off.
- Added **Section "11.4. Flash Write and Erase Guidelines" on page 107.**
- Fixed **Section "12. External RAM" on page 111**, paragraph 2 - Typo in description - "upper 6-bits" changed to "upper 7 bits".
- Fixed text in **Section "19.3.2. Watchdog Timer Usage" on page 203** to read "256 PCA clock cycles, or 3072 system clock cycles".
- Changed Table 19.4, Note 2 to refer to SYSCLK reset frequency = Internal Oscillator / 8.
- Fixed Equation 19.6, "Watchdog Timer Offset in PCA Clocks," - Typo in equation - "PCA0CPL4" changed to "PCA0CPL2".

Revision 1.5 to Revision 1.6

- Updated package drawings.

Revision 1.6 to Revision 1.7

- Removed PDIP package information.