



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 "<u>Embedded -</u> <u>Microcontrollers</u>"

Details

Product Status	Obsolete
Core Processor	8051
Core Size	8-Bit
Speed	25MHz
Connectivity	SMBus (2-Wire/I²C), SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, Temp Sensor, WDT
Number of I/O	16
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4.25K x 8
Voltage - Supply (Vcc/Vdd)	0.9V ~ 3.6V
Data Converters	A/D 15x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	24-WFQFN Exposed Pad
Supplier Device Package	24-QFN (4x4)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f931-f-gm

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong

13.1.3.Flash Write Procedure	146
13.2.Non-volatile Data Storage	147
13.3.Security Options	147
13.4.Determining the Device Part Number at Run Time	149
13.5.Flash Write and Erase Guidelines	150
13.5.1.VDD Maintenance and the VDD Monitor	150
13.5.2.PSWE Maintenance	151
13.5.3.System Clock	151
13.6.Minimizing Flash Read Current	152
14. Power Management	156
14.1.Normal Mode	157
14.2.Idle Mode	158
14.3.Stop Mode	158
14.4.Suspend Mode	159
14.5.Sleep Mode	159
14.6.Configuring Wakeup Sources	160
14.7. Determining the Event that Caused the Last Wakeup	161
14.8.Power Management Specifications	163
15. Cyclic Redundancy Check Unit (CRC0)	164
15.1.CRC Algorithm	164
15.2.Preparing for a CRC Calculation	166
15.3.Performing a CRC Calculation	166
15.4.Accessing the CRC0 Result	166
15.5.CRC0 Bit Reverse Feature	170
16. On-Chip DC-DC Converter (DC0)	171
16.1.Startup Behavior	172
16.2. High Power Applications	173
16.3.Pulse Skipping Mode	173
16.4.Enabling the DC-DC Converter	174
16.5.Minimizing Power Supply Noise	175
16.6. Selecting the Optimum Switch Size	175
16.7.DC-DC Converter Clocking Options	175
16.8.DC-DC Converter Behavior in Sleep Mode	175
16.9.DC-DC Converter Register Descriptions	177
16.10.DC-DC Converter Specifications	
17. Voltage Regulator (VREG0)	179
17.1.Voltage Regulator Electrical Specifications	179
18 Reset Sources	180
18.1 Power-On (VBAT Supply Monitor) Reset	181
18.2 Power-Fail (VDD/DC+ Supply Monitor) Reset	
18.3 External Reset	184
18 4 Missing Clock Detector Reset	184
18.5 Comparator() Reset	184
18.6 PCA Watchdog Timer Reset	18/
18 7 Flash Frror Reset	185
	100









20



Figure 4.10. Typical VOL Curves, 0.9–1.8 V



5.7. Voltage and Ground Reference Options

The voltage reference MUX is configurable to use an externally connected voltage reference, one of two internal voltage references, or one of two power supply voltages (see Figure 5.10). The ground reference MUX allows the ground reference for ADC0 to be selected between the ground pin (GND) or a port pin dedicated to analog ground (P0.1/AGND).

The voltage and ground reference options are configured using the REF0CN SFR described on page 88. Electrical specifications are can be found in the Electrical Specifications Chapter.

Important Note About the V_{REF} and AGND Inputs: Port pins are used as the external V_{REF} and AGND inputs. When using an external voltage reference or the internal precision reference, P0.0/VREF should be configured as an analog input and skipped by the Digital Crossbar. When using AGND as the ground reference to ADC0, P0.1/AGND should be configured as an analog input and skipped by the Digital Crossbar. Refer to Section "21. Port Input/Output" on page 212 for complete Port I/O configuration details. The external reference voltage must be within the range $0 \le V_{REF} \le VDD/DC+$ and the external ground reference must be at the same DC voltage potential as GND.



Figure 5.10. Voltage Reference Functional Block Diagram



Mnemonic	Mnemonic Description				
ANL C, /bit	AND complement of direct bit to Carry	2	2		
ORL C, bit	OR direct bit to carry	2	2		
ORL C, /bit	OR complement of direct bit to Carry	2	2		
MOV C, bit	Move direct bit to Carry	2	2		
MOV bit, C	Move Carry to direct bit	2	2		
JC rel	Jump if Carry is set	2	2/3		
JNC rel	Jump if Carry is not set	2	2/3		
JB bit, rel	Jump if direct bit is set	3	3/4		
JNB bit, rel	JNB bit, rel Jump if direct bit is not set				
JBC bit, rel	Jump if direct bit is set and clear bit	3	3/4		
	Program Branching	•	•		
ACALL addr11	Absolute subroutine call	2	3		
LCALL addr16	Long subroutine call	3	4		
RET	Return from subroutine	1	5		
RETI	Return from interrupt	1	5		
AJMP addr11	Absolute jump	2	3		
LJMP addr16	Long jump	3	4		
SJMP rel	Short jump (relative address)	2	3		
JMP @A+DPTR	Jump indirect relative to DPTR	1	3		
JZ rel	Jump if A equals zero	2	2/3		
JNZ rel	Jump if A does not equal zero	2	2/3		
CJNE A, direct, rel	Compare direct byte to A and jump if not equal	3	3/4		
CJNE A, #data, rel	Compare immediate to A and jump if not equal	3	3/4		
CJNE Rn, #data, rel	Compare immediate to Register and jump if not equal	3	3/4		
CJNE @Ri, #data, rel	Compare immediate to indirect and jump if not equal	3	4/5		
DJNZ Rn, rel	Decrement Register and jump if not zero	2	2/3		
DJNZ direct, rel	Decrement direct byte and jump if not zero	3	3/4		
NOP	No operation	1	1		

Table 8.1. CIP-51 Instruction Set Summary (Continued)



12. Interrupt Handler

The C8051F93x-C8051F92x microcontroller family includes an extended interrupt system supporting multiple interrupt sources and two priority levels. The allocation of interrupt sources between on-chip peripherals and external input pins varies according to the specific version of the device. Refer to Table 12.1, "Interrupt Summary," on page 135 for a detailed listing of all interrupt sources supported by the device. Refer to the data sheet 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).

Each interrupt source has one or more associated interrupt-pending flag(s) located in an SFR or an indirect register. When a peripheral or external source meets a valid interrupt condition, the associated interrupt-pending flag is set to logic 1. If both global interrupts and the specific interrupt source is enabled, a CPU interrupt request is generated when the interrupt-pending flag is set.

As soon as execution of the current instruction is complete, the CPU generates an LCALL to a predetermined address to begin execution of an interrupt service routine (ISR). Each ISR must end with an RETI instruction, which returns program execution to the next instruction that would have been executed if the interrupt request had not occurred. If interrupts are not enabled, the interrupt-pending flag is ignored by the hardware and program execution continues as normal. (The interrupt-pending flag is set to logic 1 regardless of the interrupt's enable/disable state.)

Some interrupt-pending flags are automatically cleared by 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.

12.1. Enabling Interrupt Sources

Each interrupt source can be individually enabled or disabled through the use of an associated interrupt enable bit in the Interrupt Enable and Extended Interrupt Enable SFRs. However, interrupts must first be globally enabled by setting the EA bit (IE.7) to logic 1 before the individual interrupt enables are recognized. Setting the EA bit to logic 0 disables all interrupt sources regardless of the individual interrupt-enable settings. Note that interrupts which occur when the EA bit is set to logic 0 will be held in a pending state, and will not be serviced until the EA bit is set back to logic 1.

12.2. MCU Interrupt Sources and Vectors

The CPU services interrupts by generating an LCALL to a predetermined address (the interrupt vector address) to begin execution of an interrupt service routine (ISR). The interrupt vector addresses associated with each interrupt source are listed in Table 12.1 on page 135. Software should ensure that the interrupt vector for each enabled interrupt source contains a valid interrupt service routine.

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.



16.4. Enabling the DC-DC Converter

On power-on reset, the state of the DCEN pin is sampled to determine if the device will power up in onecell or two-cell mode. In two-cell mode, the dc-dc converter always remains disabled. In one-cell mode, the dc-dc converter remains disabled in Sleep Mode, and enabled in all other power modes. See Section "14. Power Management" on page 146 for complete details on available power modes.

The dc-dc converter is enabled (one-cell mode) in hardware by placing a 0.68 μ H inductor between DCEN and VBAT. The dc-dc converter is disabled (two-cell mode) by shorting DCEN directly to GND. The DCEN pin should never be left floating. Note that the device can only switch between one-cell and two-cell mode during a power-on reset. See Section "18. Reset Sources" on page 173 for more information regarding reset behavior.

Figure 16.2 shows the two dc-dc converter configuration options.



Figure 16.2. DC-DC Converter Configuration Options

When the dc-dc converter "Enabled" configuration (one-cell mode) is chosen, the following guidelines apply:

- In most cases, the GND/DC- pin should not be externally connected to GND.
- The 0.68 µH inductor should be placed as close as possible to the DCEN pin for maximum efficiency.
- The 4.7 µF capacitor should be placed as close as possible to the inductor.
- The current loop including GND, the 4.7 μ F capacitor, the 0.68 μ H inductor and the DCEN pin should be made as short as possible.
- The PCB traces connecting VDD/DC+ to the output capacitor and the output capacitor to GND/DC- should be as short and as thick as possible in order to minimize parasitic inductance.



SFR Definition 16.2. DC0CF: DC-DC Converter Configuration

Bit	7	6	5	4	3	2	1	0
Name	Reserved	CLKD	IV[1:0]	AD0CKINV	CLKINV	ILIMIT	VDDSLP	CLKSEL
Туре	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Page = 0x0; SFR Address = 0x96

Bit	Name	Function
7	Reserved	Reserved.
		Read = 0b; Must write 0b.
6:5	CLKDIV[1:0]	DC-DC Clock Divider.
		Divides the dc-dc converter clock when the system clock is selected as the clock source for dc-dc converter. These bits are ignored when the dc-dc converter is clocked from its local oscillator. 00: The dc-dc converter clock is system clock divided by 1. 01: The dc-dc converter clock is system clock divided by 2. 10: The dc-dc converter clock is system clock divided by 4. 11: The dc-dc converter clock is system clock divided by 8.
4	AD0CKINV	ADC0 Clock Inversion (Clock Invert During Sync).
		Inverts the ADC0 SAR clock derived from the dc-dc converter clock when the SYNC bit (DC0CN.3) is enabled. This bit is ignored when the SYNC bit is set to zero. 0: ADC0 SAR clock is inverted. 1: ADC0 SAR clock is not inverted.
3	CLKINV	DC-DC Converter Clock Invert.
		Inverts the system clock used as the input to the dc-dc clock divider. 0: The dc-dc converter clock is not inverted. 1: The dc-dc converter clock is inverted.
2	ILIMIT	Peak Current Limit Threshold.
		Sets the threshold for the maximum allowed peak inductor current. See Table 16.1 for peak inductor current levels. 0: Peak inductor current is set at a lower level. 1: Peak inductor current is set at a higher level.
1	VDDSLP	VDD-DC+ Sleep Mode Connection.
		Specifies the power source for VDD/DC+ in Sleep Mode when the dc-dc converter is enabled. 0: VDD-DC+ connected to VBAT in Sleep Mode. 1: VDD-DC+ is floating in Sleep Mode.
0	CLKSEL	DC-DC Converter Clock Source Select.
		Specifies the dc-dc converter clock source. 0: The dc-dc converter is clocked from its local oscillator. 1: The dc-dc converter is clocked from the system clock.

16.10. DC-DC Converter Specifications

See Table 4.14 on page 64 for a detailed listing of dc-dc converter specifications.



19. Clocking Sources

C8051F93x-C8051F92x devices include a programmable precision internal oscillator, an external oscillator drive circuit, a low power internal oscillator, and a SmaRTClock real time clock oscillator. The precision internal oscillator can be enabled/disabled and calibrated using the OSCICN and OSCICL registers, as shown in Figure 19.1. The external oscillator can be configured using the OSCXCN register. The low power internal oscillator is automatically enabled and disabled when selected and deselected as a clock source. SmaRTClock operation is described in the SmaRTClock oscillator chapter.

The system clock (SYSCLK) can be derived from the precision internal oscillator, external oscillator, low power internal oscillator, or SmaRTClock oscillator. The global clock divider can generate a system clock that is 1, 2, 4, 8, 16, 32, 64, or 128 times slower that the selected input clock source. Oscillator electrical specifications can be found in the Electrical Specifications Chapter.





The proper way of changing the system clock when both the clock source and the clock divide value are being changed is as follows:

If switching from a fast "undivided" clock to a slower "undivided" clock:

- a. Change the clock divide value.
- b. Poll for CLKRDY > 1.
- c. Change the clock source.

If switching from a slow "undivided" clock to a faster "undivided" clock:

- a. Change the clock source.
- b. Change the clock divide value.
- c. Poll for CLKRDY > 1.





Figure 19.2. 25 MHz External Crystal Example

Important Note on External Crystals: Crystal oscillator circuits are quite sensitive to PCB layout. The crystal should be placed as close as possible to the XTAL pins on the device. The traces should be as short as possible and shielded with ground plane from any other traces which could introduce noise or interference.

When using an external crystal, the external oscillator drive circuit must be configured by software for *Crystal Oscillator Mode* or *Crystal Oscillator Mode with divide by 2 stage*. The divide by 2 stage ensures that the clock derived from the external oscillator has a duty cycle of 50%. The External Oscillator Frequency Control value (XFCN) must also be specified based on the crystal frequency. The selection should be based on Table 19.1. For example, a 25 MHz crystal requires an XFCN setting of 111b.

XFCN	Crystal Frequency	Bias Current	Typical Supply Current (VDD = 2.4 V)
000	f ≤ 20 kHz	0.5 µA	3.0 µA, f = 32.768 kHz
001	20 kHz < f ≤ 58 kHz	1.5 µA	4.8 µA, f = 32.768 kHz
010	58 kHz < f ≤ 155 kHz	4.8 µA	9.6 µA, f = 32.768 kHz
011	155 kHz < f ≤ 415 kHz	14 µA	28 µA, f = 400 kHz
100	415 kHz $<$ f \leq 1.1 MHz	40 µA	71 µA, f = 400 kHz
101	$1.1 \text{ MHz} < f \le 3.1 \text{ MHz}$	120 µA	193 µA, f = 400 kHz
110	$3.1 \text{ MHz} < f \le 8.2 \text{ MHz}$	550 µA	940 µA, f = 8 MHz
111	$8.2 \text{ MHz} < f \le 25 \text{ MHz}$	2.6 mA	3.9 mA, f = 25 MHz

Table 19.1. Recommended XFCN Settings for Crystal Mode

When the crystal oscillator is first enabled, the external oscillator valid detector allows software to determine when the external system clock has stabilized. Switching to the external oscillator before the crystal oscillator has stabilized can result in unpredictable behavior. The recommended procedure for starting the crystal is:

- 1. Configure XTAL1 and XTAL2 for analog I/O and disable the digital output drivers.
- 2. Configure and enable the external oscillator.
- 3. Poll for XTLVLD \geq 1.
- 4. Switch the system clock to the external oscillator.



20.2.3. Programmable Load Capacitance

The programmable load capacitance has 16 values to support crystal oscillators with a wide range of recommended load capacitance. If Automatic Load Capacitance Stepping is enabled, the crystal load capacitors start at the smallest setting to allow a fast startup time, then slowly increase the capacitance until the final programmed value is reached. The final programmed loading capacitor value is specified using the LOADCAP bits in the RTC0XCF register. The LOADCAP setting specifies the amount of on-chip load capacitance and does not include any stray PCB capacitance. Once the final programmed loading capacitor value is reached, the LOADRDY flag will be set by hardware to logic 1.

When using the SmaRTClock oscillator in Self-Oscillate mode, the programmable load capacitance can be used to fine tune the oscillation frequency. In most cases, increasing the load capacitor value will result in a decrease in oscillation frequency. Table 20.2 shows the crystal load capacitance for various settings of LOADCAP.

LOADCAP	Crystal Load Capacitance	Equivalent Capacitance seen on XTAL3 and XTAL4
0000	4.0 pF	8.0 pF
0001	4.5 pF	9.0 pF
0010	5.0 pF	10.0 pF
0011 5.5 pF		11.0 pF
0100	6.0 pF	12.0 pF
0101	6.5 pF	13.0 pF
0110	7.0 pF	14.0 pF
0111	7.5 pF	15.0 pF
1000	8.0 pF	16.0 pF
1001	8.5 pF	17.0 pF
1010	9.0 pF	18.0 pF
1011	9.5 pF	19.0 pF
1100	10.5 pF	21.0 pF
1101	11.5 pF	23.0 pF
1110	12.5 pF	25.0 pF
1111	13.5 pF	27.0 pF

Table 20.2. SmaRTClock Load Capacitance Settings



Internal Register Definition 20.8. CAPTUREn: SmaRTClock Timer Capt
--

Bit	7	6	5	4	3	2	1	0	
Nam	9		CAPTURE[31:0]						
Туре	R/W	R/W R/W R/W R/W R/W						R/W	
Rese	t 0	0 0 0 0 0 0							
SmaR	TClock Addres	sses: CAPTI	JRE0 = 0x00	; CAPTURE	1 = 0x01; C/	APTURE2 =(0x02; CAPTI	JRE3: 0x03.	
Bit	Name				Function	า			
7:0	CAPTURE[31	I:0] SmaRT	Clock Time	er Capture.					
	These 4 registers (CAPTURE3–CAPTURE0) are used to read or set the 32-bit SmaRTClock timer. Data is transferred to or from the SmaRTClock timer when the RTC0SET or RTC0CAP bits are set.								
Note:	The least signi	ficant bit of th	e timer captur	e value is in C	APTURE0.0.				

Internal Register Definition 20.9. ALARMn: SmaRTClock Alarm Programmed Value

Bit	7	6	5	4	3	2	1	0
Name	ALARM[31:0]							
Туре	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SmaRTClock Addresses: ALARM0 = 0x08; ALARM1 = 0x09; ALARM2 = 0x0A; ALARM3 = 0x0B

Bit	Name	Function					
7:0	ALARM[31:0]	SmaRTClock Alarm Programmed Value.					
		These 4 registers (ALARM3–ALARM0) are used to set an alarm event for the SmaRTClock timer. The SmaRTClock alarm should be disabled (RTC0AEN=0) when updating these registers.					
Note:	ote: The least significant bit of the alarm programmed value is in ALARM0.0.						



21.5. Special Function Registers for Accessing and Configuring Port I/O

All Port I/O are accessed through corresponding special function registers (SFRs) that are both byte addressable and bit addressable. When writing to a Port, the value written to the SFR is latched to maintain the output data value at each pin. When reading, the logic levels of the Port's input pins are returned regardless of the XBRn settings (i.e., even when the pin is assigned to another signal by the Crossbar, the Port register can always read its corresponding Port I/O pin). The exception to this is the execution of the read-modify-write instructions that target a Port Latch register as the destination. The read-modify-write instructions when operating on a Port SFR are the following: ANL, ORL, XRL, JBC, CPL, INC, DEC, DJNZ and MOV, CLR or SETB, when the destination is an individual bit in a Port SFR. For these instructions, the value of the latch register (not the pin) is read, modified, and written back to the SFR.

Each Port has a corresponding PnSKIP register which allows its individual Port pins to be assigned to digital functions or skipped by the Crossbar. All Port pins used for analog functions, GPIO, or dedicated digital functions such as the EMIF should have their PnSKIP bit set to 1.

The Port input mode of the I/O pins is defined using the Port Input Mode registers (PnMDIN). Each Port cell can be configured for analog or digital I/O. This selection is required even for the digital resources selected in the XBRn registers, and is not automatic. The only exception to this is P2.7, which can only be used for digital I/O.

The output driver characteristics of the I/O pins are defined using the Port Output Mode registers (PnMD-OUT). Each Port Output driver can be configured as either open drain or push-pull. This selection is required even for the digital resources selected in the XBRn registers, and is not automatic. The only exception to this is the SMBus (SDA, SCL) pins, which are configured as open-drain regardless of the PnMDOUT settings.

The drive strength of the output drivers are controlled by the Port Drive Strength (PnDRV) registers. The default is low drive strength. See Section "4. Electrical Characteristics" on page 43 for the difference in output drive strength between the two modes.



22.4.2. SMB0CN Control Register

SMB0CN is used to control the interface and to provide status information (see SFR Definition 22.2). The higher four bits of SMB0CN (MASTER, TXMODE, STA, and STO) form a status vector that can be used to jump to service routines. MASTER indicates whether a device is the master or slave during the current transfer. TXMODE indicates whether the device is transmitting or receiving data for the current byte.

STA and STO indicate that a START and/or STOP has been detected or generated since the last SMBus interrupt. STA and STO are also used to generate START and STOP conditions when operating as a master. Writing a 1 to STA will cause the SMBus interface to enter Master Mode and generate a START when the bus becomes free (STA is not cleared by hardware after the START is generated). Writing a 1 to STO while in Master Mode will cause the interface to generate a STOP and end the current transfer after the next ACK cycle. If STO and STA are both set (while in Master Mode), a STOP followed by a START will be generated.

The ARBLOST bit indicates that the interface has lost an arbitration. This may occur anytime the interface is transmitting (master or slave). A lost arbitration while operating as a slave indicates a bus error condition. ARBLOST is cleared by hardware each time SI is cleared.

The SI bit (SMBus Interrupt Flag) is set at the beginning and end of each transfer, after each byte frame, or when an arbitration is lost; see Table 22.3 for more details.

Important Note About the SI Bit: The SMBus interface is stalled while SI is set; thus SCL is held low, and the bus is stalled until software clears SI.

22.4.2.1.Software ACK Generation

When the EHACK bit in register SMB0ADM is cleared to 0, the firmware on the device must detect incoming slave addresses and ACK or NACK the slave address and incoming data bytes. As a receiver, writing the ACK bit defines the outgoing ACK value; as a transmitter, reading the ACK bit indicates the value received during the last ACK cycle. ACKRQ is set each time a byte is received, indicating that an outgoing ACK value is needed. When ACKRQ is set, software should write the desired outgoing value to the ACK bit before clearing SI. A NACK will be generated if software does not write the ACK bit before clearing SI. SDA will reflect the defined ACK value immediately following a write to the ACK bit; however SCL will remain low until SI is cleared. If a received slave address is not acknowledged, further slave events will be ignored until the next START is detected.

22.4.2.2.Hardware ACK Generation

When the EHACK bit in register SMB0ADM is set to 1, automatic slave address recognition and ACK generation is enabled. More detail about automatic slave address recognition can be found in Section 22.4.3. As a receiver, the value currently specified by the ACK bit will be automatically sent on the bus during the ACK cycle of an incoming data byte. As a transmitter, reading the ACK bit indicates the value received on the last ACK cycle. The ACKRQ bit is not used when hardware ACK generation is enabled. If a received slave address is NACKed by hardware, further slave events will be ignored until the next START is detected, and no interrupt will be generated.

Table 22.3 lists all sources for hardware changes to the SMB0CN bits. Refer to Table 22.5 for SMBus status decoding using the SMB0CN register.

Refer to the C8051F930 errata when using hardware ACK generation on C8051F930/31/20/21 devices.



24.5. Serial Clock Phase and Polarity

Four combinations of serial clock phase and polarity can be selected using the clock control bits in the SPI Configuration Register (SPInCFG). The CKPHA bit (SPInCFG.5) selects one of two clock phases (edge used to latch the data). The CKPOL bit (SPInCFG.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 SPIENn bit, SPInCN.0) when changing the clock phase or polarity. The clock and data line relationships for master mode are shown in Figure 24.5. For slave mode, the clock and data relationships are shown in Figure 24.6 and Figure 24.7. 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 SPIn Clock Rate Register (SPInCKR) as shown in SFR Definition 24.3 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 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 frequency.



Figure 24.5. Master Mode Data/Clock Timing















* SCK is shown for CKPOL = 0. SCK is the opposite polarity for CKPOL = 1.





* SCK is shown for CKPOL = 0. SCK is the opposite polarity for CKPOL = 1.





25.3. Timer 3

Timer 3 is a 16-bit timer formed by two 8-bit SFRs: TMR3L (low byte) and TMR3H (high byte). Timer 3 may operate in 16-bit auto-reload mode or (split) 8-bit auto-reload mode. The T3SPLIT bit (TMR2CN.3) defines the Timer 3 operation mode. Timer 3 can also be used in Capture Mode to measure the external oscillator source or the Comparator 1 period with respect to another oscillator. The ability to measure the Comparator 1 period with respect to the system clock is makes using Touch Sense Switches very easy.

Timer 3 may be clocked by the system clock, the system clock divided by 12, external oscillator source divided by 8, or Comparator 1 output. The external oscillator source divided by 8 and Comparator 1 output is synchronized with the system clock.

25.3.1. 16-bit Timer with Auto-Reload

When T3SPLIT (TMR3CN.3) is zero, Timer 3 operates as a 16-bit timer with auto-reload. Timer 3 can be clocked by SYSCLK, SYSCLK divided by 12, external oscillator clock source divided by 8, or Comparator 1 output. As the 16-bit timer register increments and overflows from 0xFFFF to 0x0000, the 16-bit value in the Timer 3 reload registers (TMR3RLH and TMR3RLL) is loaded into the Timer 3 register as shown in Figure 25.7, and the Timer 3 High Byte Overflow Flag (TMR3CN.7) is set. If Timer 3 interrupts are enabled (if EIE1.7 is set), an interrupt will be generated on each Timer 3 overflow. Additionally, if Timer 3 interrupts are enabled and the TF3LEN bit is set (TMR3CN.5), an interrupt will be generated each time the lower 8 bits (TMR3L) overflow from 0xFF to 0x00.



Figure 25.7. Timer 3 16-Bit Mode Block Diagram



26.5. Register Descriptions for PCA0

Following are detailed descriptions of the special function registers related to the operation of the PCA.

SFR Definition 26.1. PCA0CN: PCA Control

Bit	7	6	5	4	3	2	1	0
Name	CF	CR	CCF5	CCF4	CCF3	CCF2	CCF1	CCF0
Туре	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Page = 0x0; SFR Address = 0xD8; Bit-Addressable

Bit	Name	Function				
7	CF	PCA Counter/Timer Overflow Flag.				
		Set by hardware when the PCA Counter/Timer overflows from 0xFFFF to 0x0000. When the Counter/Timer Overflow (CF) interrupt is enabled, setting this bit causes the CPU to vector to the PCA interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software.				
6	CR	PCA Counter/Timer Run Control.				
		This bit enables/disables the PCA Counter/Timer. 0: PCA Counter/Timer disabled. 1: PCA Counter/Timer enabled.				
5:0	CCF[5:0]	PCA Module n Capture/Compare Flag.				
		These bits are set by hardware when a match or capture occurs in the associated PCA Module n. When the CCFn interrupt is enabled, setting this bit causes the CPU to vector to the PCA interrupt service routine. This bit is not automatically cleared by hardware and must be cleared by software.				



SFR Definition 26.5. PCA0L: PCA Counter/Timer Low Byte

Bit	7	6	5	4	3	2	1	0
Name	PCA0[7:0]							
Туре	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Page = 0x0; SFR Address = 0xF9

Bit	Name	Function			
7:0	PCA0[7:0]	PCA Counter/Timer Low Byte.			
		The PCA0L register holds the low byte (LSB) of the 16-bit PCA Counter/Timer.			
Note:	: When the WDTE bit is set to 1, the PCA0L register cannot be modified by software. To change the contents of the PCA0L register, the Watchdog Timer must first be disabled.				

SFR Definition 26.6. PCA0H: PCA Counter/Timer High Byte

Bit	7	6	5	4	3	2	1	0
Name	PCA0[15:8]							
Туре	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Page = 0x0; SFR Address = 0xFA

Bit	Name	Function			
7:0	PCA0[15:8]	PCA Counter/Timer High Byte.			
		The PCA0H register holds the high byte (MSB) of the 16-bit PCA Counter/Timer. Reads of this register will read the contents of a "snapshot" register, whose contents are updated only when the contents of PCA0L are read (see Section 26.1).			
Note:	When the WDTE bit is set to 1, the PCA0H register cannot be modified by software. To change the contents of the PCA0H register, the Watchdog Timer must first be disabled.				

