E·XFL



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	Last Time Buy
Core Processor	8051
Core Size	8-Bit
Speed	25MHz
Connectivity	SMBus (2-Wire/I ² C), SPI, UART/USART
Peripherals	Cap Sense, 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	512 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 16x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	24-SSOP (0.154", 3.90mm Width)
Supplier Device Package	24-QSOP
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f812-gu

Email: info@E-XFL.COM

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





Figure 3.3. SOIC-16 Pinout Diagram (Top View)



4. QFN-20 Package Specifications



Figure 4.1. QFN-20 Package Drawing

Dimension	Min	Тур	Max	Dimension	Min	Тур	Max
A	0.80	0.90	1.00	L	0.45	0.55	0.65
A1	0.00	0.02	0.05	L1	0.00	—	0.15
b	0.18	0.25	0.30	aaa	_	—	0.15
D	4.00 BSC.			bbb		—	0.10
D2	2.00	2.15	2.25	ddd	_	—	0.05
е	0.50 BSC.			eee	_	—	0.08
E	4.00 BSC.			Z	_	0.43	—
E2	2.00	2.15	2.25	Y	—	0.18	—

Table 4.1. QFN-20 Package Dimensions

Notes:

- 1. All dimensions shown are in millimeters (mm) unless otherwise noted.
- 2. Dimensioning and Tolerancing per ANSI Y14.5M-1994.
- **3.** This drawing conforms to the JEDEC Solid State Outline MO-220, variation VGGD except for custom features D2, E2, Z, Y, and L which are toleranced per supplier designation.
- **4.** Recommended card reflow profile is per the JEDEC/IPC J-STD-020 specification for Small Body Components.



11. Voltage Regulator (REG0)

C8051F80x-83x devices include an internal voltage regulator (REG0) to regulate the internal core supply to 1.8 V from a V_{DD} supply of 1.8 to 3.6 V. A power-saving mode is built into the regulator to help reduce current consumption in low-power applications. This mode is accessed through the REG0CN register (SFR Definition 11.1). Electrical characteristics for the on-chip regulator are specified in Table 7.5 on page 41

Under default conditions, when the device enters STOP mode the internal regulator will remain on. This allows any enabled reset source to generate a reset for the device and bring the device out of STOP mode. For additional power savings, the STOPCF bit can be used to shut down the regulator and the internal power network of the device when the part enters STOP mode. When STOPCF is set to 1, the RST pin or a full power cycle of the device are the only methods of generating a reset.



The Comparator response time may be configured in software via the CPT0MD register (see SFR Definition 12.2). Selecting a longer response time reduces the Comparator supply current.



Figure 12.2. Comparator Hysteresis Plot

The Comparator hysteresis is software-programmable via its Comparator Control register CPT0CN. The user can program both the amount of hysteresis voltage (referred to the input voltage) and the positive and negative-going symmetry of this hysteresis around the threshold voltage.

The Comparator hysteresis is programmed using bits 3:0 in the Comparator Control Register CPT0CN (shown in SFR Definition 12.1). The amount of negative hysteresis voltage is determined by the settings of the CP0HYN bits. As shown in Figure 12.2, settings of 20, 10 or 5 mV of negative hysteresis can be programmed, or negative hysteresis can be disabled. In a similar way, the amount of positive hysteresis is determined by the setting the CP0HYP bits.

Comparator interrupts can be generated on both rising-edge and falling-edge output transitions. (For Interrupt enable and priority control, see Section "18.1. MCU Interrupt Sources and Vectors" on page 103). The CP0FIF flag is set to logic 1 upon a Comparator falling-edge occurrence, and the CP0RIF flag is set to logic 1 upon the Comparator rising-edge occurrence. Once set, these bits remain set until cleared by software. The Comparator rising-edge interrupt mask is enabled by setting CP0RIE to a logic 1. The Comparator0 falling-edge interrupt mask is enabled by setting CP0FIE to a logic 1.

The output state of the Comparator can be obtained at any time by reading the CP0OUT bit. The Comparator is enabled by setting the CP0EN bit to logic 1, and is disabled by clearing this bit to logic 0.

Note that false rising edges and falling edges can be detected when the comparator is first powered on or if changes are made to the hysteresis or response time control bits. Therefore, it is recommended that the rising-edge and falling-edge flags be explicitly cleared to logic 0 a short time after the comparator is enabled or its mode bits have been changed.



SFR Definition 12.1. CPT0CN: Comparator0 Control

Bit	7	6	5	4	3	2	1	0
Name	CP0EN	CP0OUT	CP0RIF	CP0FIF	CP0H)	YP[1:0]	CP0H	/N[1:0]
Туре	R/W	R	R/W	R/W	R/W		R/	W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x9B

Bit	Name	Function			
7	CP0EN	Comparator0 Enable Bit.			
		0: Comparator0 Disabled.			
		1: Comparator0 Enabled.			
6	CP0OUT	Comparator0 Output State Flag.			
		0: Voltage on CP0+ < CP0–.			
		1: Voltage on CP0+ > CP0–.			
5	CP0RIF	Comparator0 Rising-Edge Flag. Must be cleared by software.			
		0: No Comparator0 Rising Edge has occurred since this flag was last cleared.			
		1: Comparator0 Rising Edge has occurred.			
4	CP0FIF	Comparator0 Falling-Edge Flag. Must be cleared by software.			
		0: No Comparator0 Falling-Edge has occurred since this flag was last cleared.			
		1: Comparator0 Falling-Edge has occurred.			
3:2	CP0HYP[1:0]	Comparator0 Positive Hysteresis Control Bits.			
		00: Positive Hysteresis Disabled.			
		01: Positive Hysteresis = 5 mV.			
		10: Positive Hysteresis = 10 mV.			
		11: Positive Hysteresis = 20 mV.			
1:0	CP0HYN[1:0]	Comparator0 Negative Hysteresis Control Bits.			
		00: Negative Hysteresis Disabled.			
		01: Negative Hysteresis = 5 mV.			
		10: Negative Hysteresis = 10 mV.			
		11: Negative Hysteresis = 20 mV.			



Mnemonic Description		Bytes	Clock Cycles
Arithmetic Operations			
ADD A. Rn	Add register to A	1	1
ADD A, direct	Add direct byte to A	2	2
ADD A. @Ri	Add indirect RAM to A	1	2
ADD A #data	Add immediate to A	2	2
ADDC A. Rn	Add register to A with carry	1	1
ADDC A. direct	Add direct byte to A with carry	2	2
ADDC A. @Ri	Add indirect RAM to A with carry	1	2
ADDC A, #data	Add immediate to A with carry	2	2
SUBB A, Rn	Subtract register from A with borrow	1	1
SUBB A, direct	Subtract direct byte from A with borrow	2	2
SUBB A, @Ri	Subtract indirect RAM from A with borrow	1	2
SUBB A, #data	Subtract immediate from A with borrow	2	2
INC A	Increment A	1	1
INC Rn	Increment register	1	1
INC direct	Increment direct byte	2	2
INC @Ri	Increment indirect RAM	1	2
DEC A	Decrement A	1	1
DEC Rn	Decrement register	1	1
DEC direct	Decrement direct byte	2	2
DEC @Ri	Decrement indirect RAM	1	2
INC DPTR	Increment Data Pointer	1	1
MUL AB	Multiply A and B	1	4
DIV AB	Divide A by B	1	8
DA A	Decimal adjust A	1	1
Logical Operations		•	
ANL A, Rn	AND Register to A	1	1
ANL A, direct	AND direct byte to A	2	2
ANL A, @Ri	AND indirect RAM to A	1	2
ANL A, #data	AND immediate to A	2	2
ANL direct, A	AND A to direct byte	2	2
ANL direct, #data	AND immediate to direct byte	3	3
ORL A, Rn	OR Register to A	1	1
ORL A, direct	OR direct byte to A	2	2
ORL A, @Ri	OR indirect RAM to A	1	2
ORL A, #data	OR immediate to A	2	2
ORL direct, A	OR A to direct byte	2	2
ORL direct, #data	OR immediate to direct byte	3	3
XRL A, Rn	Exclusive-OR Register to A	1	1
XRL A, direct	Exclusive-OR direct byte to A	2	2
XRL A, @Ri	Exclusive-OR indirect RAM to A	1	2
XRL A, #data	Exclusive-OR immediate to A	2	2
XRL direct, A	Exclusive-OR A to direct byte	2	2

Table 14.1. CIP-51 Instruction Set Summary



Notes on Registers, Operands and Addressing Modes:

Rn—Register R0–R7 of the currently selected register bank.

@Ri—Data RAM location addressed indirectly through R0 or R1.

rel—8-bit, signed (twos complement) offset relative to the first byte of the following instruction. Used by SJMP and all conditional jumps.

direct—8-bit internal data location's address. This could be a direct-access Data RAM location (0x00–0x7F) or an SFR (0x80–0xFF).

#data—8-bit constant

#data16—16-bit constant

bit—Direct-accessed bit in Data RAM or SFR

addr11—11-bit destination address used by ACALL and AJMP. The destination must be within the same 2 kB page of program memory as the first byte of the following instruction.

addr16—16-bit destination address used by LCALL and LJMP. The destination may be anywhere within the 8 kB program memory space.

There is one unused opcode (0xA5) that performs the same function as NOP. All mnemonics copyrighted © Intel Corporation 1980.



SFR Definition 14.6. PSW: Program Status Word

Bit	7	6	5	4	3	2	1	0
Name	CY	AC	F0	RS[1:0]	OV	F1	PARITY
Туре	R/W	R/W	R/W	R/W		R/W	R/W	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD0; Bit-Addressable

Bit	Name	Function
7	CY	Carry Flag.
		This bit is set when the last arithmetic operation resulted in a carry (addition) or a borrow (subtraction). It is cleared to logic 0 by all other arithmetic operations.
6	AC	Auxiliary Carry Flag.
		This bit is set when the last arithmetic operation resulted in a carry into (addition) or a borrow from (subtraction) the high order nibble. It is cleared to logic 0 by all other arithmetic operations.
5	F0	User Flag 0.
		This is a bit-addressable, general purpose flag for use under software control.
4:3	RS[1:0]	Register Bank Select.
		These bits select which register bank is used during register accesses.
		00: Bank 0, Addresses 0x00-0x07
		01: Bank 1, Addresses 0x08-0x0F
		11: Bank 3, Addresses 0x10-0x17
2	OV	Overflow Flag.
		This bit is set to 1 under the following circumstances:
		 An ADD, ADDC, or SUBB instruction causes a sign-change overflow.
		 A MUL instruction results in an overflow (result is greater than 255). A DW instruction equates a divide by more condition
		• A Div Instruction causes a divide-by-zero condition. The OV bit is cleared to 0 by the ADD, ADDC, SUBB, MUL, and DIV instructions in all
		other cases.
1	F1	User Flag 1.
		This is a bit-addressable, general purpose flag for use under software control.
0	PARITY	Parity Flag.
		This bit is set to logic 1 if the sum of the eight bits in the accumulator is odd and cleared if the sum is even.



18. Interrupts

The C8051F80x-83x includes an extended interrupt system supporting a total of 15 interrupt sources with 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. Each interrupt source has one or more associated interrupt-pending flag(s) located in an SFR. When a peripheral or external source meets a valid interrupt condition, the associated interrupt-pending flag is set to logic 1.

If interrupts are enabled for the source, an 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 regard-less of the interrupt's enable/disable state.)

Each interrupt source can be individually enabled or disabled through the use of an associated interrupt enable bit in an SFR (IE–EIE1). 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.

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.



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

19.4.1. VDD Maintenance and the VDD Monitor

- 1. If the system power supply is subject to voltage or current "spikes," add sufficient transient protection devices to the power supply to ensure that the supply voltages listed in the Absolute Maximum Ratings table are not exceeded.
- 2. Make certain that the minimum VDD rise time specification of 1 ms is met. If the system cannot meet this rise time specification, then add an external VDD brownout circuit to the RST pin of the device that holds the device in reset until VDD reaches the minimum device operating voltage and re-asserts RST if VDD drops below the minimum device operating voltage.
- 3. Keep the on-chip VDD Monitor enabled and enable the VDD Monitor as a reset source as early in code as possible. This should be the first set of instructions executed after the Reset Vector. For C-based systems, this will involve modifying the startup code added by the C compiler. See your compiler documentation for more details. Make certain that there are no delays in software between enabling the VDD Monitor and enabling the VDD Monitor as a reset source. Code examples showing this can be found in "AN201: Writing to Flash from Firmware," available from the Silicon Laboratories website.
- **Note:** On C8051F80x-83x devices, both the VDD Monitor and the VDD Monitor reset source must be enabled to write or erase Flash without generating a Flash Error Device Reset.

On C8051F80x-83x devices, both the VDD Monitor and the VDD Monitor reset source are enabled by hardware after a power-on reset.

- 4. As an added precaution, explicitly enable the VDD Monitor and enable the VDD Monitor as a reset source inside the functions that write and erase Flash memory. The VDD Monitor enable instructions should be placed just after the instruction to set PSWE to a 1, but before the Flash write or erase operation instruction.
- 5. Make certain that all writes to the RSTSRC (Reset Sources) register use direct assignment operators and explicitly DO NOT use the bit-wise operators (such as AND or OR). For example, "RSTSRC = 0x02" is correct, but "RSTSRC |= 0x02" is incorrect.
- 6. Make certain that all writes to the RSTSRC register explicitly set the PORSF bit to a 1. Areas to check are initialization code which enables other reset sources, such as the Missing Clock Detector or Comparator, for example, and instructions which force a Software Reset. A global search on "RSTSRC" can quickly verify this.

19.4.2. PSWE Maintenance

- 1. Reduce the number of places in code where the PSWE bit (b0 in PSCTL) is set to a 1. There should be exactly one routine in code that sets PSWE to a 1 to write Flash bytes and one routine in code that sets both PSWE and PSEE both to a 1 to erase Flash pages.
- 2. Minimize the number of variable accesses while PSWE is set to a 1. Handle pointer address updates and loop maintenance outside the "PSWE = 1;... PSWE = 0;" area. Code examples showing this can be found in "AN201: Writing to Flash from Firmware," available from the Silicon Laboratories website.
- 3. Disable interrupts prior to setting PSWE to a 1 and leave them disabled until after PSWE has been reset to 0. Any interrupts posted during the Flash write or erase operation will be serviced in priority order after the Flash operation has been completed and interrupts have been re-enabled by software.
- Make certain that the Flash write and erase pointer variables are not located in XRAM. See your compiler documentation for instructions regarding how to explicitly locate variables in different memory areas.
- 5. Add address bounds checking to the routines that write or erase Flash memory to ensure that a routine called with an illegal address does not result in modification of the Flash.



21. Reset Sources

Reset circuitry allows the controller to be easily placed in a predefined default condition. On entry to this reset state, the following occur:

- CIP-51 halts program execution
- Special Function Registers (SFRs) are initialized to their defined reset values
- External Port pins are forced to a known state
- Interrupts and timers are disabled.

All SFRs are reset to the predefined values noted in the SFR detailed descriptions. The contents of internal data memory are unaffected during a reset; any previously stored data is preserved. However, since the stack pointer SFR is reset, the stack is effectively lost, even though the data on the stack is not altered.

The Port I/O latches are reset to 0xFF (all logic ones) in open-drain mode. Weak pullups are enabled during and after the reset. For V_{DD} Monitor and power-on resets, the \overrightarrow{RST} pin is driven low until the device exits the reset state.

On exit from the reset state, the program counter (PC) is reset, and the system clock defaults to the internal oscillator. The Watchdog Timer is enabled with the system clock divided by 12 as its clock source. Program execution begins at location 0x0000.



Figure 21.1. Reset Sources



SFR Definition 21.1. VDM0CN: V_{DD} Monitor Control

Bit	7	6	5	4	3	2	1	0
Name	VDMEN	VDDSTAT						
Туре	R/W	R	R	R	R	R	R	R
Reset	Varies	Varies	Varies	Varies	Varies	Varies	Varies	Varies

SFR Address = 0xFF

Bit	Name	Function
7	VDMEN	V _{DD} Monitor Enable.
		This bit turns the V _{DD} monitor circuit on/off. The V _{DD} Monitor cannot generate system resets until it is also selected as a reset source in register RSTSRC (SFR Definition 21.2). Selecting the V _{DD} monitor as a reset source before it has stabilized may generate a system reset. In systems where this reset would be undesirable, a delay should be introduced between enabling the V _{DD} Monitor and selecting it as a reset source. After a power-on reset, the VDD monitor is enabled, and this bit will read 1. The state of this bit is sticky through any other reset source. 0: V _{DD} Monitor Disabled. 1: V _{DD} Monitor Enabled.
6	VDDSTAT	V _{DD} Status.
		This bit indicates the current power supply status (V_{DD} Monitor output). 0: V_{DD} is at or below the V_{DD} monitor threshold. 1: V_{DD} is above the V_{DD} monitor threshold.
5:0	Unused	Read = Varies; Write = Don't care.

21.3. External Reset

The external RST pin provides a means for external circuitry to force the device into a reset state. Asserting an active-low signal on the RST pin generates a reset; an external pullup and/or decoupling of the RST pin may be necessary to avoid erroneous noise-induced resets. See Section "7. Electrical Characteristics" on page 39 for complete RST pin specifications. The PINRSF flag (RSTSRC.0) is set on exit from an external reset.

21.4. Missing Clock Detector Reset

The Missing Clock Detector (MCD) is a one-shot circuit that is triggered by the system clock. If the system clock remains high or low for more than the MCD timeout, the one-shot will time out and generate a reset. After a MCD reset, the MCDRSF flag (RSTSRC.2) will read 1, signifying the MCD as the reset source; otherwise, this bit reads 0. Writing a 1 to the MCDRSF bit enables the Missing Clock Detector; writing a 0 disables it. The state of the RST pin is unaffected by this reset.



SFR Definition 21.2. RSTSRC: Reset Source

Bit	7	6	5	4	3	2	1	0
Name		FERROR	CORSEF	SWRSF	WDTRSF	MCDRSF	PORSF	PINRSF
Туре	R	R	R/W	R/W	R	R/W	R/W	R
Reset	0	Varies						

SFR Address = 0xEF

Bit	Name	Description	Write	Read						
7	Unused	Unused.	Don't care.	0						
6	FERROR	Flash Error Reset Flag.	N/A	Set to 1 if Flash read/write/erase error caused the last reset.						
5	CORSEF	Comparator0 Reset Enable and Flag.	Writing a 1 enables Comparator0 as a reset source (active-low).	Set to 1 if Comparator0 caused the last reset.						
4	SWRSF	Software Reset Force and Flag.	Writing a 1 forces a system reset.	Set to 1 if last reset was caused by a write to SWRSF.						
3	WDTRSF	Watchdog Timer Reset Flag.	N/A	Set to 1 if Watchdog Timer overflow caused the last reset.						
2	MCDRSF	Missing Clock Detector Enable and Flag.	Writing a 1 enables the Missing Clock Detector. The MCD triggers a reset if a missing clock condition is detected.	Set to 1 if Missing Clock Detector timeout caused the last reset.						
1	PORSF	Power-On / V _{DD} Monitor Reset Flag, and V _{DD} monitor Reset Enable.	Writing a 1 enables the V_{DD} monitor as a reset source. Writing 1 to this bit before the V_{DD} monitor is enabled and stabilized may cause a system reset.	Set to 1 anytime a power- on or V _{DD} monitor reset occurs. When set to 1 all other RSTSRC flags are inde- terminate.						
0	PINRSF	HW Pin Reset Flag.	N/A	Set to 1 if RST pin caused the last reset.						
Note:	Do not use	read-modify-write operations on this	lote: Do not use read-modify-write operations on this register							



24.2. 32-bit CRC Algorithm

The C8051F80x-83x CRC unit calculates the 32-bit CRC using a poly of 0x04C11DB7. The CRC-32 algorithm is "reflected", meaning that all of the input bytes and the final 32-bit output are bit-reversed in the processing engine. The following is a description of a simplified CRC algorithm that produces results identical to the hardware:

- 1. XOR the least-significant byte of the current CRC result with the input byte. If this is the first iteration of the CRC unit, the current CRC result will be the set initial value (0x00000000 or 0xFFFFFFF).
- 2. Right-shift the CRC result.
- 3. If the LSB of the CRC result is set, XOR the CRC result with the reflected polynomial (0xEDB88320).
- 4. Repeat at Step 2 for the number of input bits (8).

For example, the 32-bit C8051F80x-83x CRC algorithm can be described by the following code:

```
unsigned long UpdateCRC (unsigned long CRC_acc, unsigned char CRC_input) {
   unsigned char i; // loop counter
   #define POLY 0xEDB88320 // bit-reversed version of the poly 0x04C11DB7
   // Create the CRC "dividend" for polynomial arithmetic (binary arithmetic
   // with no carries)
   CRC_acc = CRC_acc ^ CRC_input;
   // "Divide" the poly into the dividend using CRC XOR subtraction
   // CRC_acc holds the "remainder" of each divide
   // Only complete this division for 8 bits since input is 1 byte
   for (i = 0; i < 8; i++)
   {
      // Check if the MSB is set (if MSB is 1, then the POLY can "divide" \,
      // into the "dividend")
      if ((CRC_acc & 0x0000001) == 0x0000001)
      {
          // if so, shift the CRC value, and XOR "subtract" the poly
          CRC_acc = CRC_acc >> 1;
          CRC_acc ^= POLY;
      }
      else
      {
          // if not, just shift the CRC value
          CRC_acc = CRC_acc >> 1;
      }
   }
   return CRC_acc; // Return the final remainder (CRC value)
```

Table 24.2 lists example input values and the associated outputs using the 32-bit C8051F80x-83x CRC algorithm (an initial value of 0xFFFFFFF is used):

Table 24.2. Example 32-bit CRC Outputs

Input	Output
0x63	0xF9462090
0xAA, 0xBB, 0xCC	0x41B207B3
0x00, 0x00, 0xAA, 0xBB, 0xCC	0x78D129BC



25.1. Signal Descriptions

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

25.1.1. 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.

25.1.2. 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.

25.1.3. 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.

25.1.4. 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.
- 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 25.2, Figure 25.3, and Figure 25.4 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 "23. Port Input/Output" on page 138 for general purpose port I/O and crossbar information.

25.2. SPI0 Master Mode Operation

A SPI master device initiates all data transfers on a SPI bus. SPI0 is placed in master mode by setting the Master Enable flag (MSTEN, SPI0CN.6). Writing a byte of data to the SPI0 data register (SPI0DAT) when in master mode writes to the transmit buffer. If the SPI shift register is empty, the byte in the transmit buffer is moved to the shift register, and a data transfer begins. The SPI0 master immediately shifts out the data serially on the MOSI line while providing the serial clock on SCK. The SPIF (SPI0CN.7) flag is set to logic 1 at the end of the transfer. If interrupts are enabled, an interrupt request is generated when the SPIF flag



26.5.4. Read Sequence (Slave)

During a read sequence, an SMBus master reads data from a slave device. The slave in this transfer will be a receiver during the address byte, and a transmitter during all data bytes. When slave events are enabled (INH = 0), the interface enters Slave Receiver Mode (to receive the slave address) when a START followed by a slave address and direction bit (READ in this case) is received. If hardware ACK generation is disabled, upon entering Slave Receiver Mode, an interrupt is generated and the ACKRQ bit is set. The software must respond to the received slave address with an ACK, or ignore the received slave address with a NACK. If hardware ACK generation is enabled, the hardware will apply the ACK for a slave address which matches the criteria set up by SMB0ADR and SMB0ADM. The interrupt will occur after the ACK cycle.

If the received slave address is ignored (by software or hardware), slave interrupts will be inhibited until the next START is detected. If the received slave address is acknowledged, zero or more data bytes are transmitted. If the received slave address is acknowledged, data should be written to SMB0DAT to be transmitted. The interface enters slave transmitter mode, and transmits one or more bytes of data. After each byte is transmitted, the master sends an acknowledge bit; if the acknowledge bit is an ACK, SMB0DAT should be written with the next data byte. If the acknowledge bit is a NACK, SMB0DAT should not be written to before SI is cleared (an error condition may be generated if SMB0DAT is written following a received NACK while in slave transmitter mode). The interface exits slave transmitter mode after receiving a STOP. Note that the interface will switch to slave receiver mode if SMB0DAT is not written following a Slave Transmitter interrupt. Figure 26.8 shows a typical slave read sequence. Two transmitted data bytes are shown, though any number of bytes may be transmitted. Notice that all of the "data byte transferred" interrupts occur **after** the ACK cycle in this mode, regardless of whether hardware ACK generation is enabled.





26.6. SMBus Status Decoding

The current SMBus status can be easily decoded using the SMB0CN register. The appropriate actions to take in response to an SMBus event depend on whether hardware slave address recognition and ACK generation is enabled or disabled. Table 26.5 describes the typical actions when hardware slave address recognition and ACK generation is disabled. Table 26.6 describes the typical actions when hardware slave address recognition and ACK generation is enabled. In the tables, STATUS VECTOR refers to the four upper bits of SMB0CN: MASTER, TXMODE, STA, and STO. The shown response options are only the typical responses; application-specific procedures are allowed as long as they conform to the SMBus specification. Highlighted responses are allowed by hardware but do not conform to the SMBus specification.



SFR Definition 29.4. PCA0CPMn: PCA0 Capture/Compare Mode

Bit	7	6	5	4	3	2	1	0	
Nam	e PWM16	in ECOMn	CAPPn	CAPNn	MATn	TOGn	PWMn	ECCFn	
Туре	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Rese	t 0	0	0	0	0	0	0	0	
SFR Addresses: PCA0CPM0 = 0xDA. PCA0CPM1 = 0xDB. PCA0CPM2 = 0xDC									
Bit	Name	Function							
7	PWM16n	16-bit Pulse Width Modulation Enable.							
		This bit enables 16-bit mode when Pulse Width Modulation mode is enabled.							
		0: 8 to 15-bit F	PWM selecte	ed.					
		1: 16-bit PWN	selected.						
6	ECOMn	Comparator I	Function En	able.					
		This bit enable	es the compa	arator function	on for PCA n	nodule n whe	en set to 1.		
5	CAPPn	Capture Posi	tive Functio	on Enable.					
		This bit enable	This bit enables the positive edge capture for PCA module n when set to 1.						
4	CAPNn	Capture Nega	ative Functi	on Enable.					
		This bit enables the negative edge capture for PCA module n when set to 1.							
3	MATn	Match Function Enable.							
		This bit enables the match function for PCA module n when set to 1. When enabled, matches of the PCA counter with a module's capture/compare register cause the CCFn bit in PCA0MD register to be set to logic 1.							
2	TOGn	Toggle Function Enable.							
		This bit enables the toggle function for PCA module n when set to 1. When enabled, matches of the PCA counter with a module's capture/compare register cause the logic level on the CEXn pin to toggle. If the PWMn bit is also set to logic 1, the module operates in Frequency Output Mode.							
1	PWMn	Pulse Width Modulation Mode Enable.							
		This bit enables the PWM function for PCA module n when set to 1. When enabled, a pulse width modulated signal is output on the CEXn pin. 8 to 15-bit PWM is used if PWM16n is cleared; 16-bit mode is used if PWM16n is set to logic 1. If the TOGn bit is also set, the module operates in Frequency Output Mode.							
0	ECCFn	Capture/Compare Flag Interrupt Enable.							
		This bit sets the masking of the Capture/Compare Flag (CCFn) interrupt.							
		1: Enable a Capture/Compare Flag interrupt request when CCFn is set.							
Note:	Inter When the WDTE bit is set to 1, the PCA0CPM2 register cannot be modified, and module 2 acts as the watchdog timer. To change the contents of the PCA0CPM2 register or the function of module 2, the Watchdog Timer must be disabled.								



C2 Register Definition 30.2. DEVICEID: C2 Device ID

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

C2 Address: 0x00

Bit	Name	Function		
7:0	DEVICEID[7:0]	Device ID.		
		This read-only register returns the 8-bit device ID: 0x23 (C8051F80x-83x).		

C2 Register Definition 30.3. REVID: C2 Revision ID

Bit	7	6	5	4	3	2	1	0
Nam	e REVID[7:0]							
Type R/W								
Rese	et Varies	Varies	Varies	Varies	Varies	Varies	Varies	Varies
C2 Address: 0x01								
Bit	Name	Function						
7:0	REVID[7:0]	0] Revision ID.						
		This read-only register returns the 8-bit revision ID. For example: 0x00 = Revision A.						



DOCUMENT CHANGE LIST

Revision 0.2 to Revision 1.0

- Updated Electrical Specification Tables to reflect production characterization data.
- Added Minimum SYSCLK specification for writing or erasing Flash.
- Added caution for going into suspend with wake source active (Section 20.3)
- Corrected VDM0CN reset values to "Varies".
- Removed mention of IDAC in Pinout table.

