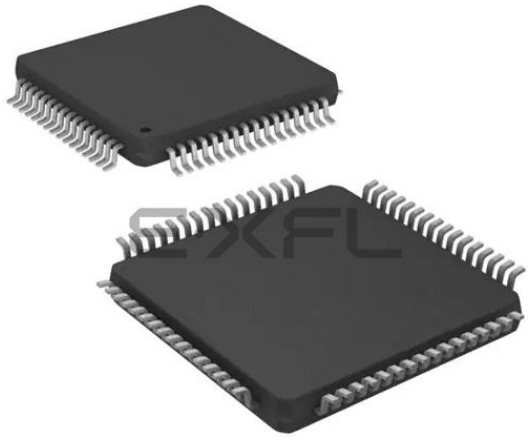


Welcome to E-XFL.COM

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	CANbus, SMBus (2-Wire/I²C), SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, Temp Sensor, WDT
Number of I/O	24
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4.25K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 2x16b, 8x10b; D/A 2x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-TQFP
Supplier Device Package	64-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f061-gq

List of Tables

1. System Overview 19
Table 1.1. Product Selection Guide 20
2. Absolute Maximum Ratings 37
Table 2.1. Absolute Maximum Ratings* 37
3. Global DC Electrical Characteristics 38
Table 3.1. Global DC Electrical Characteristics 38
4. Pinout and Package Definitions 39
Table 4.1. Pin Definitions 39
5. 16-Bit ADCs (ADC0 and ADC1) 51
Table 5.1. Conversion Timing (tConv) 55
Table 5.2. 16-Bit ADC0 and ADC1 Electrical Characteristics 73
Table 5.3. Voltage Reference 0 and 1 Electrical Characteristics 74
6. Direct Memory Access Interface (DMA0) 75
Table 6.1. DMA0 Instruction Set 76
7. 10-Bit ADC (ADC2, C8051F060/1/2/3) 87
Table 7.1. ADC2 Electrical Characteristics 101
8. DACs, 12-Bit Voltage Mode (DAC0 and DAC1, C8051F060/1/2/3) 103
Table 8.1. DAC Electrical Characteristics 109
9. Voltage Reference 2 (C8051F060/2) 111
Table 9.1. Voltage Reference Electrical Characteristics 112
10. Voltage Reference 2 (C8051F061/3) 113
Table 10.1. Voltage Reference Electrical Characteristics 114
11. Voltage Reference 2 (C8051F064/5/6/7) 115
Table 11.1. Voltage Reference Electrical Characteristics 116
12. Comparators 117
Table 12.1. Comparator Electrical Characteristics 122
13. CIP-51 Microcontroller 123
Table 13.1. CIP-51 Instruction Set Summary 126
Table 13.2. Special Function Register (SFR) Memory Map 141
Table 13.3. Special Function Registers 143
Table 13.4. Interrupt Summary 152
14. Reset Sources 163
Table 14.1. Reset Electrical Characteristics 169
15. Oscillators 171
Table 15.1. Internal Oscillator Electrical Characteristics 173
16. Flash Memory 177
Table 16.1. Flash Electrical Characteristics 178
17. External Data Memory Interface and On-Chip XRAM 187
Table 17.1. AC Parameters for External Memory Interface 202
18. Port Input/Output 203
Table 18.1. Port I/O DC Electrical Characteristics 203
19. Controller Area Network (CAN0, C8051F060/1/2/3) 225

Table 5.2. 16-Bit ADC0 and ADC1 Electrical Characteristics

VDD = 3.0 V, AV+ = 3.0 V, AVDD = 3.0 V, VREF = 2.50 V (REFBE=0), -40 to +85 °C unless otherwise specified

Parameter	Conditions	Min	Typ	Max	Units
DC Accuracy					
Resolution			16		bits
Integral Nonlinearity (C8051F060/1/4/5/6/7)	Single-Ended Differential		±0.75 ±0.5	±2 ±1	LSB
Integral Nonlinearity (C8051F062/3)	Single-Ended Differential		±1.5 ±1	±4 ±2	LSB
Differential Nonlinearity	Guaranteed Monotonic		±0.5		LSB
Offset Error			0.1		mV
Full Scale Error			0.008		%F.S.
Gain Temperature Coefficient			0.5		ppm/°C
Dynamic Performance (Sampling Rate = 1 Msps, AVDD, AV+ = 3.3V)					
Signal-to-Noise Plus Distortion	Fin = 10 kHz, Single-Ended Fin = 100 kHz, Single-Ended Fin = 10 kHz, Differential Fin = 100 kHz, Differential		86 84 89 88		dB dB dB dB
Total Harmonic Distortion	Fin = 10 kHz, Single-Ended Fin = 100 kHz, Single-Ended Fin = 10 kHz, Differential Fin = 100 kHz, Differential		96 84 103 93		dB dB dB dB
Spurious-Free Dynamic Range	Fin = 10 kHz, Single-Ended Fin = 100 kHz, Single-Ended Fin = 10 kHz, Differential Fin = 100 kHz, Differential		97 88 104 99		dB dB dB dB
CMRR	Fin = 10 kHz		86		dB
Channel Isolation			100		dB
Timing					
SAR Clock Frequency				25	MHz
Conversion Time in SAR Clocks		18			clocks
Track/Hold Acquisition Time		280			ns
Throughput Rate				1	Msps
Aperture Delay	External CNVST Signal		1.5		ns
RMS Aperture Jitter	External CNVST Signal		5		ps
Analog Inputs					
Input Voltage Range	Single-Ended (AINn - AINnG) Differential (AIN0 - AIN1)	0 -VREF		VREF VREF	V V
Input Capacitance			80		pF

Table 5.2. 16-Bit ADC0 and ADC1 Electrical Characteristics (Continued)

VDD = 3.0 V, AV+ = 3.0 V, AVDD = 3.0 V, VREF = 2.50 V (REFBE=0), -40 to +85 °C unless otherwise specified

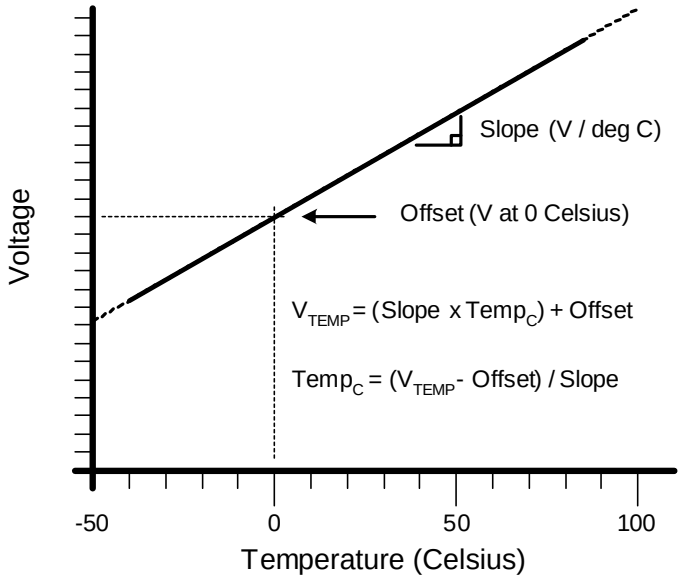
Parameter	Conditions	Min	Typ	Max	Units
Operating Input Range	AIN0 or AIN1 AIN0G or AIN1G (DC Only)	-0.2 -0.2		AV+ 0.6	V V
Power Specifications					
Power Supply Current (each ADC)	Operating Mode, 1 Msps AV+ AVDD Shutdown Mode		 4.0 2.0 <1		 mA mA µA
Power Supply Rejection	VDD ± 5%		±0.5		LSB

Table 5.3. Voltage Reference 0 and 1 Electrical Characteristics

VDD = 3.0 V, AV+ = 3.0 V, AVDD = 3.0 V, -40 to +85 °C unless otherwise specified

Parameter	Conditions	Min	Typ	Max	Units
Internal Reference					
Output Voltage	25 °C ambient	2.36	2.43	2.48	V
VREF Temperature Coefficient			15		ppm/°C
Power Supply Current (each Voltage Reference)	AV+		1.5		mA
External Reference					
Input Voltage Range		2.0		AV+	V
Input Current	ADC throughput = 1 Msps		450		µA

Figure 7.2. Temperature Sensor Transfer Function



7.2. Modes of Operation

ADC2 has a maximum conversion speed of 200 ksps. The ADC2 conversion clock is a divided version of the system clock, determined by the AD2SC bits in the ADC2CF register (system clock divided by (AD2SC + 1) for 0 ≤ AD2SC ≤ 31). The ADC2 conversion clock should be no more than 3 MHz.

7.2.1. Starting a Conversion

A conversion can be initiated in one of four ways, depending on the programmed states of the ADC2 Start of Conversion Mode bits (AD2CM1-0) in register ADC2CN. Conversions may be initiated by one of the following:

1. Writing a ‘1’ to the AD2BUSY bit of register ADC2CN
2. A Timer 3 overflow (i.e. timed continuous conversions)
3. A rising edge on the CNVSTR2 input signal (Assigned by the crossbar)
4. A Timer 2 overflow

When CNVSTR2 is used as a conversion start source, it must be enabled in the crossbar, and the corresponding pin must be set to open-drain, high-impedance mode (see Section “18. Port Input/Output” on page 203 for more details on Port I/O configuration).

Writing a ‘1’ to AD2BUSY provides software control of ADC2 whereby conversions are performed "on-demand". During conversion, the AD2BUSY bit is set to logic 1 and reset to logic 0 when the conversion is complete. The falling edge of AD2BUSY triggers an interrupt (when enabled) and sets the ADC2 interrupt flag (AD2INT). Note: When polling for ADC conversion completions, the ADC2 interrupt flag (AD2INT) should be used. Converted data is available in the ADC2 data registers, ADC2H and ADC2L, when bit AD2INT is logic 1. Note that when Timer 2 or Timer 3 overflows are used as the conversion source, low byte overflows are used if the timer is in 8-bit mode; and high byte overflows are used if the timer is in 16-bit mode. See Section “24. Timers” on page 287 for timer configuration.

7.2.3. Settling Time Requirements

A minimum tracking time is required before an accurate conversion can be performed. This tracking time is determined by the AMUX2 resistance, the ADC2 sampling capacitance, any external source resistance, and the accuracy required for the conversion. Note that in low-power tracking mode, three SAR clocks are used for tracking at the start of every conversion. For most applications, these three SAR clocks will meet the minimum tracking time requirements.

Figure 7.4 shows the equivalent ADC2 input circuits for both Differential and Single-ended modes. Notice that the equivalent time constant for both input circuits is the same. The required ADC2 settling time for a given settling accuracy (SA) may be approximated by Equation 7.1. When measuring the Temperature Sensor output, R_{TOTAL} reduces to R_{MUX}. See Table 7.1 for ADC2 minimum settling time requirements.

Equation 7.1. ADC2 Settling Time Requirements

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

Where:
SA is the settling accuracy, given as a fraction of an LSB (for example, 0.25 to settle within 1/4 LSB)
t is the required settling time in seconds
R_{TOTAL} is the sum of the AMUX2 resistance and any external source resistance.
n is the ADC resolution in bits (10).

Figure 7.4. ADC2 Equivalent Input Circuits

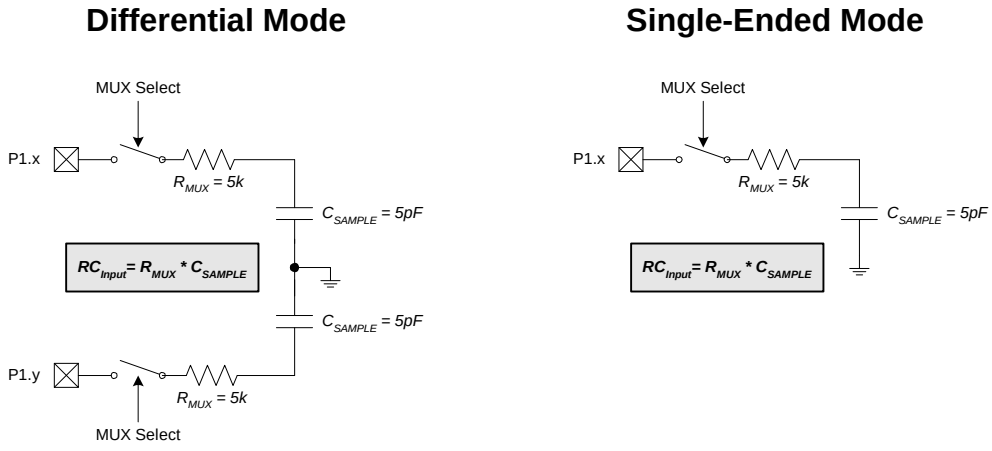


Figure 8.2. DAC0H: DAC0 High Byte Register

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: 0xD3
SFR Page: 0

Bits7-0: DAC0 Data Word Most Significant Byte.

Figure 8.3. DAC0L: DAC0 Low Byte Register

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: 0xD2
SFR Page: 0

Bits7-0: DAC0 Data Word Least Significant Byte.

13.2.6.3.SFR Page Stack Example

The following is an example that shows the operation of the SFR Page Stack during interrupts.

In this example, the SFR Page Control is left in the default enabled state (i.e., SFRPGEN = 1), and the CIP-51 is executing in-line code that is writing values to Port 5 (SFR “P5”, located at address 0xD8 on SFR Page 0x0F). The device is also using the Programmable Counter Array (PCA) and the 10-bit ADC (ADC2) window comparator to monitor a voltage. The PCA is timing a critical control function in its interrupt service routine (ISR), so its interrupt is enabled and is set to *high* priority. The ADC2 is monitoring a voltage that is less important, but to minimize the software overhead its window comparator is being used with an associated ISR that is set to *low* priority. At this point, the SFR page is set to access the Port 5 SFR (SFRPAGE = 0x0F). See Figure 13.4 below.

Figure 13.4. SFR Page Stack While Using SFR Page 0x0F To Access Port 5

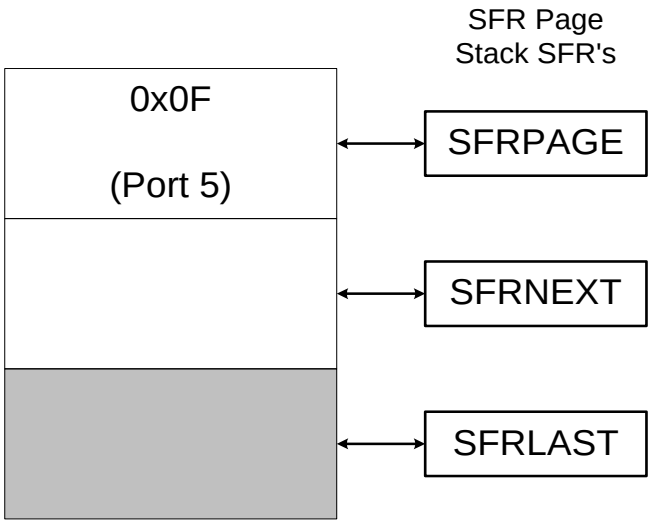


Figure 14.4. RSTSRC: Reset Source Register

R	R/W	R/W	R/W	R	R/W	R/W	R/W	Reset Value
-	CNVRSEF	C0RSEF	SWRSEF	WDTRSF	MCDRSF	PORSF	PINRSF	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xEF
SFR Page: 0

Bit7: Reserved.

Bit6: CNVRSEF: Convert Start Reset Source Enable and Flag
Write: 0: CNVSTR2 is not a reset source.
1: CNVSTR2 is a reset source (active low).
Read: 0: Source of prior reset was not CNVSTR2.
1: Source of prior reset was CNVSTR2.

Bit5: C0RSEF: Comparator0 Reset Enable and Flag.
Write: 0: Comparator0 is not a reset source.
1: Comparator0 is a reset source (active low).
Read: 0: Source of last reset was not Comparator0.
1: Source of last reset was Comparator0.

Bit4: SWRSF: Software Reset Force and Flag.
Write: 0: No effect.
1: Forces an internal reset. /RST pin is not effected.
Read: 0: Source of last reset was not a write to the SWRSF bit.
1: Source of last reset was a write to the SWRSF bit.

Bit3: WDTRSF: Watchdog Timer Reset Flag.
0: Source of last reset was not WDT timeout.
1: Source of last reset was WDT timeout.

Bit2: MCDRSF: Missing Clock Detector Flag.
Write: 0: Missing Clock Detector disabled.
1: Missing Clock Detector enabled; triggers a reset if a missing clock condition is detected.
Read: 0: Source of last reset was not a Missing Clock Detector timeout.
1: Source of last reset was a Missing Clock Detector timeout.

Bit1: PORSF: Power-On Reset Flag.
Write: If the VDD monitor circuitry is enabled (by tying the MONEN pin to a logic high state), this bit can be written to select or de-select the VDD monitor as a reset source.
0: De-select the VDD monitor as a reset source.
1: Select the VDD monitor as a reset source.
Important: At power-on, the VDD monitor is enabled/disabled using the external VDD monitor enable pin (MONEN). The PORSF bit does not disable or enable the VDD monitor circuit. It simply selects the VDD monitor as a reset source.
Read: This bit is set whenever a power-on reset occurs. This may be due to a true power-on reset or a VDD monitor reset. In either case, data memory should be considered indeterminate following the reset.
0: Source of last reset was not a power-on or VDD monitor reset.
1: Source of last reset was a power-on or VDD monitor reset.
Note: When this flag is read as '1', all other reset flags are indeterminate.

Bit0: PINRSF: HW Pin Reset Flag.
Write: 0: No effect.
1: Forces a Power-On Reset. /RST is driven low.
Read: 0: Source of prior reset was not /RST pin.
1: Source of prior reset was /RST pin.

16.3. Security Options

The CIP-51 provides security options to protect the Flash memory from inadvertent modification by software as well as prevent the viewing of proprietary program code and constants. The Program Store Write Enable (PSCTL.0) and the Program Store Erase Enable (PSCTL.1) bits protect the Flash memory from accidental modification by software. These bits must be explicitly set to logic 1 before software can write or erase the Flash memory. Additional security features prevent proprietary program code and data constants from being read or altered across the JTAG interface or by software running on the system controller.

A set of security lock bytes protect the Flash program memory from being read or altered across the JTAG interface. Each bit in a security lock-byte protects one 8k-byte block of memory. Clearing a bit to logic 0 in a Read Lock Byte prevents the corresponding block of Flash memory from being read across the JTAG interface. Clearing a bit in the Write/Erase Lock Byte protects the block from JTAG erasures and/or writes. The Scratchpad area is read or write/erase locked when all bits in the corresponding security byte are cleared to logic 0.

On the C8051F060/1/2/3/4/5, the security lock bytes are located at 0xFBFE (Write/Erase Lock) and 0xFBFF (Read Lock), as shown in Figure 16.1. On the C8051F066/7, the security lock bytes are located at 0x7FFE (Write/Erase Lock) and 0x7FFF (Read Lock), as shown in Figure 16.2. The 512-byte sector containing the lock bytes can be written to, but not erased, by software. An attempted read of a read-locked byte returns undefined data. Debugging code in a read-locked sector is not possible through the JTAG interface. The lock bits can always be read from and written to logic 0 regardless of the security setting applied to the block containing the security bytes. This allows additional blocks to be protected after the block containing the security bytes has been locked.

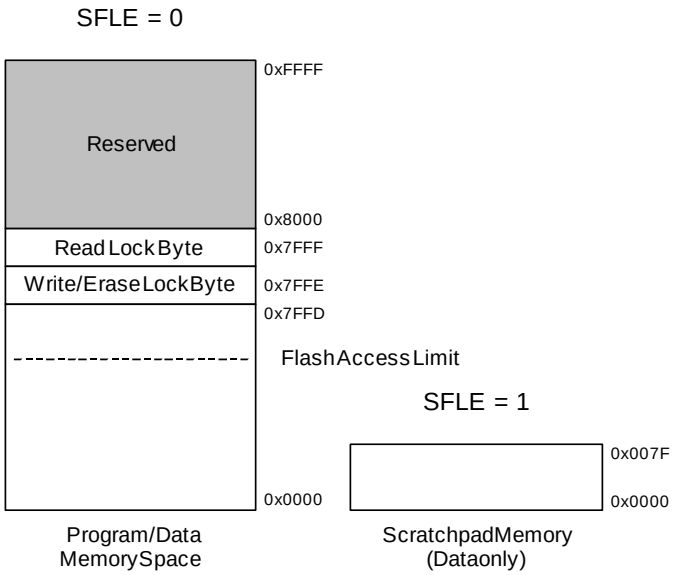
Important Note: To ensure protection from external access, the block containing the lock bytes must be Write/Erase locked. On the 64 k byte devices (C8051F060/1/2/3/4/5), the page containing the security bytes is 0xFA00-0xFBFF, and is locked by clearing bit 7 of the Write/Erase Lock Byte. On the 32 k byte devices (C8051F066/7), the page containing the security bytes is 0x7E00-0x7FFF, and is locked by clearing bit 3 of the Write/Erase Lock Byte. If the page containing the security bytes is not Write/Erase locked, it is still possible to erase this page of Flash memory through the JTAG port and reset the security bytes.

When the page containing the security bytes has been Write/Erase locked, a JTAG full device erase must be performed to unlock any areas of Flash protected by the security bytes. A JTAG full device erase is initiated by performing a normal JTAG erase operation on either of the security byte locations. This operation must be initiated through the JTAG port, and cannot be performed from firmware running on the device.

Figure 16.2. C8051F066/7 Flash Program Memory Map and Security Bytes

Read and Write/Erase Security Bits
(Bit 7 is MSB)

Bit	Memory Block
7	N/A
6	N/A
5	N/A
4	N/A
3	0x6000-0x7FFD
2	0x4000-0x5FFF
1	0x2000-0x3FFF
0	0x0000-0x1FFF



Flash Read Lock Byte

Bits7-0: Each bit locks a corresponding block of memory.

0: Read operations are locked (disabled) for corresponding block across the JTAG interface.

1: Read operations are unlocked (enabled) for corresponding block across the JTAG interface.

Flash Write/Erase Lock Byte

Bits7-0: Each bit locks a corresponding block of memory.

0: Write/Erase operations are locked (disabled) for corresponding block across the JTAG interface.

1: Write/Erase operations are unlocked (enabled) for corresponding block across the JTAG interface.

NOTE: When the block containing the security bytes is locked, the security bytes may be written but not erased.

Flash Access Limit Register (FLACL)

The Flash Access Limit is defined by the setting of the FLACL register, as described in Figure 16.3. Firmware running at or above this address is prohibited from using the MOVX and MOVC instructions to read, write, or erase Flash locations below this address.

The Flash Access Limit security feature (see Figure 16.3) protects proprietary program code and data from being read by software running on the C8051F060/1/2/3/4/5/6/7. This feature provides support for OEMs that wish to program the MCU with proprietary value-added firmware before distribution. The value-added firmware can be protected while allowing additional code to be programmed in remaining program memory space later.

The Flash Access Limit (FAL) is a 16-bit address that establishes two logical partitions in the program memory space. The first is an upper partition consisting of all the program memory locations at or above the FAL address, and the second is a lower partition consisting of all the program memory locations start-

16.3.1. Summary of Flash Security Options

There are three Flash access methods supported on the C8051F060/1/2/3/4/5/6/7; 1) Accessing Flash through the JTAG debug interface, 2) Accessing Flash from firmware residing below the Flash Access Limit, and 3) Accessing Flash from firmware residing at or above the Flash Access Limit.

Accessing Flash through the JTAG debug interface:

1. The Read and Write/Erase Lock bytes (security bytes) provide security for Flash access through the JTAG interface.
2. Any unlocked page may be read from, written to, or erased.
3. Locked pages cannot be read from, written to, or erased.
4. Reading the security bytes is always permitted.
5. Locking additional pages by writing to the security bytes is always permitted.
6. If the page containing the security bytes is **unlocked**, it can be directly erased. **Doing so will reset the security bytes and unlock all pages of Flash.**
7. If the page containing the security bytes is **locked**, it cannot be directly erased. **To unlock the page containing the security bytes, a full JTAG device erase is required.** A full JTAG device erase will erase all Flash pages, including the page containing the security bytes and the security bytes themselves.
8. The Reserved Area cannot be read from, written to, or erased at any time.

Accessing Flash from firmware residing below the Flash Access Limit:

1. The Read and Write/Erase Lock bytes (security bytes) do not restrict Flash access from user firmware.
2. Any page of Flash except the page containing the security bytes may be read from, written to, or erased.
3. **The page containing the security bytes cannot be erased.** Unlocking pages of Flash can only be performed via the JTAG interface.
4. The page containing the security bytes may be read from or written to. Pages of Flash can be locked from JTAG access by writing to the security bytes.
5. The Reserved Area cannot be read from, written to, or erased at any time.

Accessing Flash from firmware residing at or above the Flash Access Limit:

1. The Read and Write/Erase Lock bytes (security bytes) do not restrict Flash access from user firmware.
2. Any page of Flash at or above the Flash Access Limit except the page containing the security bytes may be read from, written to, or erased.
3. Any page of Flash below the Flash Access Limit cannot be read from, written to, or erased.
4. Code branches to locations below the Flash Access Limit are permitted.
5. **The page containing the security bytes cannot be erased.** Unlocking pages of Flash can only be performed via the JTAG interface.
6. The page containing the security bytes may be read from or written to. Pages of Flash can be locked from JTAG access by writing to the security bytes.
7. The Reserved Area cannot be read from, written to, or erased at any time.

Figure 16.5. PSCTL: Program Store Read/Write Control

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
-	-	-	-	-	SFLE	PSEE	PSWE	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:

SFR Address: 0x8F
SFR Page: 0

Bits 7-3: UNUSED. Read = 00000b, Write = don't care.

Bit 2: SFLE: Scratchpad Flash Memory Access Enable
When this bit is set, Flash MOVC reads and writes from user software are directed to the 128-byte Scratchpad Flash sector. When SFLE is set to logic 1, Flash accesses out of the address range 0x00-0x7F should not be attempted. Reads/Writes out of this range will yield undefined results.
0: Flash access from user software directed to the Program/Data Flash sector.
1: Flash access from user software directed to the Scratchpad sector.

Bit 1: PSEE: Program Store Erase Enable.
Setting this bit allows an entire page of the Flash program memory to be erased provided the PSWE bit is also set. After setting this bit, a write to Flash memory using the MOVX instruction will erase the entire page that contains the location addressed by the MOVX instruction. The value of the data byte written does not matter. **Note: The Flash page containing the Read Lock Byte and Write/Erase Lock Byte cannot be erased by software.**
0: Flash program memory erasure disabled.
1: Flash program memory erasure enabled.

Bit 0: PSWE: Program Store Write Enable.
Setting this bit allows writing a byte of data to the Flash program memory using the MOVX write instruction. The location must be erased prior to writing data.
0: Write to Flash program memory disabled. MOVX write operations target External RAM.
1: Write to Flash program memory enabled. MOVX write operations target Flash memory.

18. Port Input/Output

The C8051F06x family of devices are fully integrated mixed-signal System on a Chip MCUs with 59 digital I/O pins (C8051F060/2/4/6) or 24 digital I/O pins (C8051F061/3/5/7), organized as 8-bit Ports. All ports are both bit- and byte-addressable through their corresponding Port Data registers. All Port pins support configurable Open-Drain or Push-Pull output modes and weak pull-ups. Additionally, Port 0 pins are 5 V-tolerant. A block diagram of the Port I/O cell is shown in Figure 18.1. Complete Electrical Specifications for the Port I/O pins are given in Table 18.1.

Figure 18.1. Port I/O Cell Block Diagram

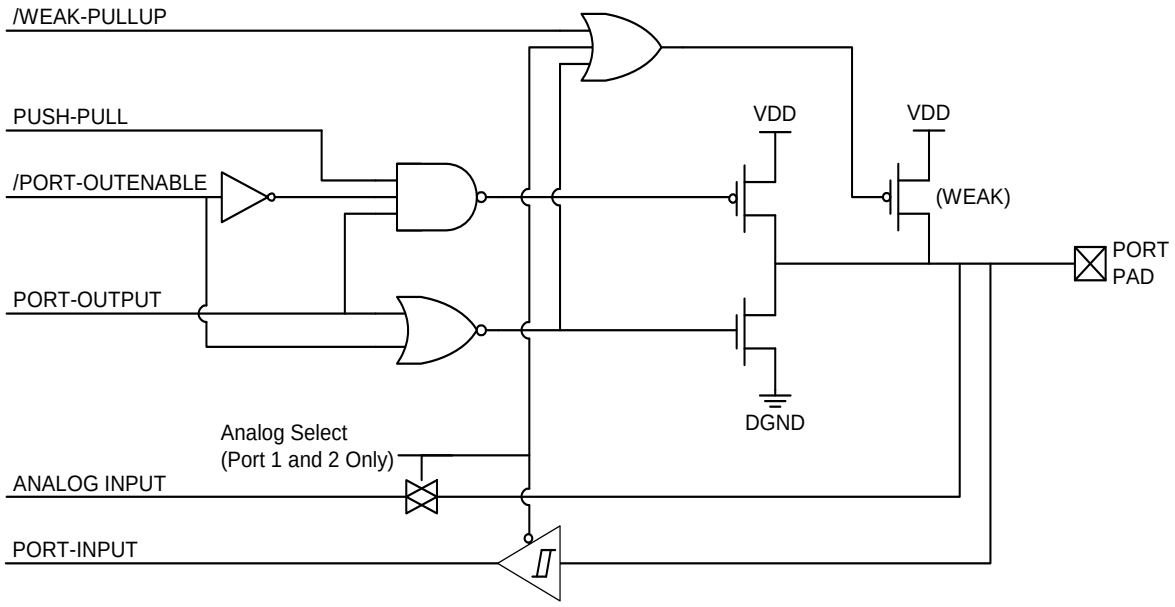


Table 18.1. Port I/O DC Electrical Characteristics

VDD = 2.7 to 3.6 V, -40 to +85 °C unless otherwise specified.

Parameter	Conditions	Min	Typ	Max	Units
Output High Voltage (V_{OH})	I_{OH} = -3 mA, Port I/O Push-Pull I_{OH} = -10 µA, Port I/O Push-Pull	VDD - 0.7 VDD - 0.1			V
Output Low Voltage (V_{OL})	I_{OL} = 8.5 mA I_{OL} = 10 µA			0.6 0.1	V
Input High Voltage (VIH)		0.7 x VDD			
Input Low Voltage (VIL)				0.3 x VDD	
Input Leakage Current	DGND < Port Pin < VDD, Pin Tri-state Weak Pull-up Off Weak Pull-up On		 10	± 1	 µA µA
Input Capacitance			5		pF

20.4.2. Clock Rate Register

Figure 20.9. SMB0CR: SMBus0 Clock Rate Register

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: 0xCF
SFR Page: 0

Bits7-0: SMB0CR.[7:0]: SMBus0 Clock Rate Preset.
The SMB0CR Clock Rate register controls the frequency of the serial clock SCL in master mode. The 8-bit word stored in the SMB0CR Register preloads a dedicated 8-bit timer. The timer counts up, and when it rolls over to 0x00, the SCL logic state toggles.

The SMB0CR setting should be bounded by the following equation , where *SMB0CR* is the unsigned 8-bit value in register SMB0CR, and *SYSCLK* is the system clock frequency in Hz:

$$SMB0CR < ((288 - 0.85 \cdot SYSCLK) / (1.125 \cdot 10^6))$$

The resulting SCL signal high and low times are given by the following equations:

$$T_{LOW} = (256 - SMB0CR) / SYSCLK$$

$$T_{HIGH} \cong (258 - SMB0CR) / SYSCLK + 625ns$$

Using the same value of SMB0CR from above, the Bus Free Timeout period is given in the following equation:

$$T_{BFT} \cong 10 \times \frac{(256 - SMB0CR) + 1}{SYSCLK}$$

20.4.3. Data Register

The SMBus0 Data register SMB0DAT holds a byte of serial data to be transmitted or one that has just been received. Software can read or write to this register while the SI flag is set to logic 1; software should not attempt to access the SMB0DAT register when the SMBus is enabled and the SI flag reads logic 0 since the hardware may be in the process of shifting a byte of data in or out of the register.

Data in SMB0DAT is always shifted out MSB first. After a byte has been received, the first bit of received data is located at the MSB of SMB0DAT. While data is being shifted out, data on the bus is simultaneously being shifted in. Therefore, SMB0DAT always contains the last data byte present on the bus. In the event of lost arbitration, the transition from master transmitter to slave receiver is made with the correct data in SMB0DAT.

Figure 20.10. SMB0DAT: SMBus0 Data Register

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: 0xC2
SFR Page: 0

Bits7-0: SMB0DAT: SMBus0 Data.
The SMB0DAT register contains a byte of data to be transmitted on the SMBus0 serial interface or a byte that has just been received on the SMBus0 serial interface. The CPU can read from or write to this register whenever the SI serial interrupt flag (SMB0CN.3) is set to logic 1. When the SI flag is not set, the system may be in the process of shifting data and the CPU should not attempt to access this register.

20.4.4. Address Register

The SMB0ADR Address register holds the slave address for the SMBus0 interface. In slave mode, the seven most-significant bits hold the 7-bit slave address. The least significant bit (Bit0) is used to enable the recognition of the general call address (0x00). If Bit0 is set to logic 1, the general call address will be recognized. Otherwise, the general call address is ignored. The contents of this register are ignored when

21.2. 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 is set. While the SPI0 master transfers data to a slave on the MOSI line, the addressed SPI slave device simultaneously transfers the contents of its shift register to the SPI master on the MISO line in a full-duplex operation. Therefore, the SPIF flag serves as both a transmit-complete and receive-data-ready flag. The data byte received from the slave is transferred MSB-first into the master's shift register. When a byte is fully shifted into the register, it is moved to the receive buffer where it can be read by the processor by reading SPI0DAT.

When configured as a master, SPI0 can operate in one of three different modes: multi-master mode, 3-wire single-master mode, and 4-wire single-master mode. The default, multi-master mode is active when NSSMD1 (SPI0CN.3) = 0 and NSSMD0 (SPI0CN.2) = 1. In this mode, NSS is an input to the device, and is used to disable the master SPI0 when another master is accessing the bus. When NSS is pulled low in this mode, MSTEN (SPI0CN.6) and SPIEN (SPI0CN.0) are set to 0 to disable the SPI master device, and a Mode Fault is generated (MODF, SPI0CN.5 = 1). Mode Fault will generate an interrupt if enabled. SPI0 must be manually re-enabled in software under these circumstances. In multi-master systems, devices will typically default to being slave devices while they are not acting as the system master device. In multi-master mode, slave devices can be addressed individually (if needed) using general-purpose I/O pins. Figure 21.2 shows a connection diagram between two master devices in multiple-master mode.

3-wire single-master mode is active when NSSMD1 (SPI0CN.3) = 0 and NSSMD0 (SPI0CN.2) = 0. In this mode, NSS is not used, and is not mapped to an external port pin through the crossbar. Any slave devices that must be addressed in this mode should be selected using general-purpose I/O pins. Figure 21.3 shows a connection diagram between a master device in 3-wire master mode and a slave device.

4-wire single-master mode is active when NSSMD1 (SPI0CN.3) = 1. In this mode, NSS is configured as an output pin, and can be used as a slave-select signal for a single SPI device. In this mode, the output value of NSS is controlled (in software) with the bit NSSMD0 (SPI0CN.2). Additional slave devices can be addressed using general-purpose I/O pins. Figure 21.4 shows a connection diagram for a master device in 4-wire master mode and two slave devices.

Figure 22.9. SSTA0: UART0 Status and Clock Selection Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
FE0	RXOV0	TXCOL0	SMOD0	S0TCLK1	S0TCLK0	S0RCLK1	S0RCLK0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0x91
SFR Page: 0

Bit7: FE0: Frame Error Flag.†
This flag indicates if an invalid (low) STOP bit is detected.
0: Frame Error has not been detected.
1: Frame Error has been detected.

Bit6: RXOV0: Receive Overrun Flag.†
This flag indicates new data has been latched into the receive buffer before software has read the previous byte.
0: Receive overrun has not been detected.
1: Receive Overrun has been detected.

Bit5: TXCOL0: Transmit Collision Flag.†
This flag indicates user software has written to the SBUF0 register while a transmission is in progress.
0: Transmission Collision has not been detected.
1: Transmission Collision has been detected.

Bit4: SMOD0: UART0 Baud Rate Doubler Enable.
This bit enables/disables the divide-by-two function of the UART0 baud rate logic for configurations described in the UART0 section.
0: UART0 baud rate divide-by-two enabled.
1: UART0 baud rate divide-by-two disabled.

Bits3-2: UART0 Transmit Baud Rate Clock Selection Bits.

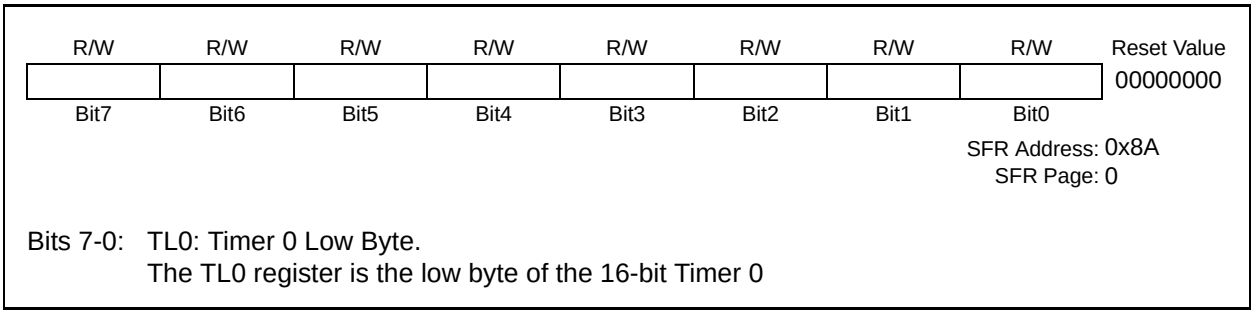
S0TCLK1	S0TCLK0	**Serial Transmit Baud Rate Clock Source**
0	0	Timer 1 generates UART0 TX Baud Rate
0	1	Timer 2 Overflow generates UART0 TX baud rate
1	0	Timer 3 Overflow generates UART0 TX baud rate
1	1	Timer 4 Overflow generates UART0 TX baud rate

Bits1-0: UART0 Receive Baud Rate Clock Selection Bits.

S0RCLK1	S0RCLK0	**Serial Receive Baud Rate Clock Source**
0	0	Timer 1 generates UART0 RX Baud Rate
0	1	Timer 2 Overflow generates UART0 RX baud rate
1	0	Timer 3 Overflow generates UART0 RX baud rate
1	1	Timer 4 Overflow generates UART0 RX baud rate

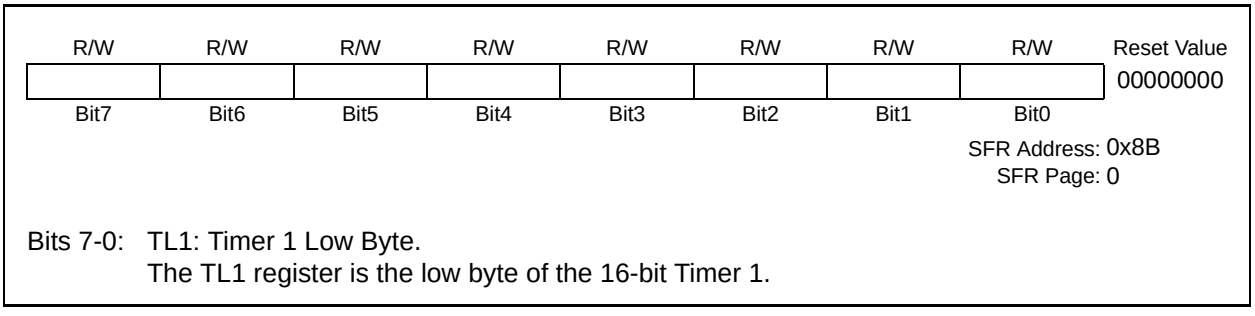
† Note: FE0, RXOV0, and TXCOL0 are flags only, and no interrupt is generated by these conditions.

Figure 24.7. TL0: Timer 0 Low Byte



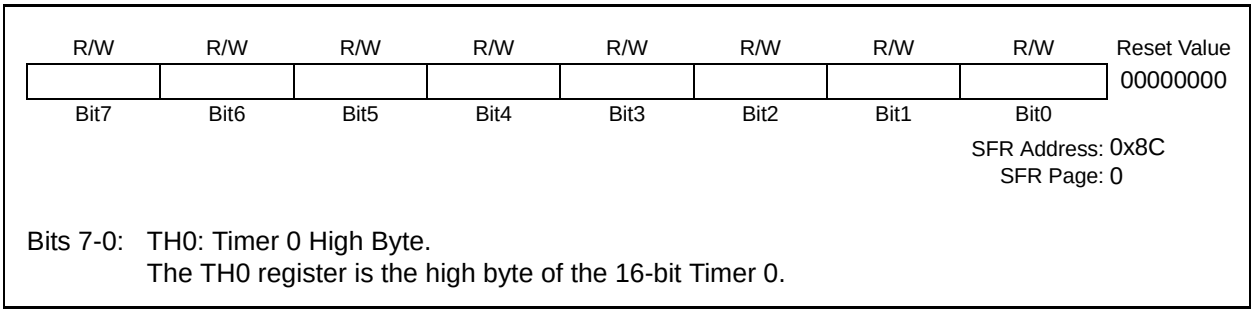
Bits 7-0: TL0: Timer 0 Low Byte.
The TL0 register is the low byte of the 16-bit Timer 0

Figure 24.8. TL1: Timer 1 Low Byte



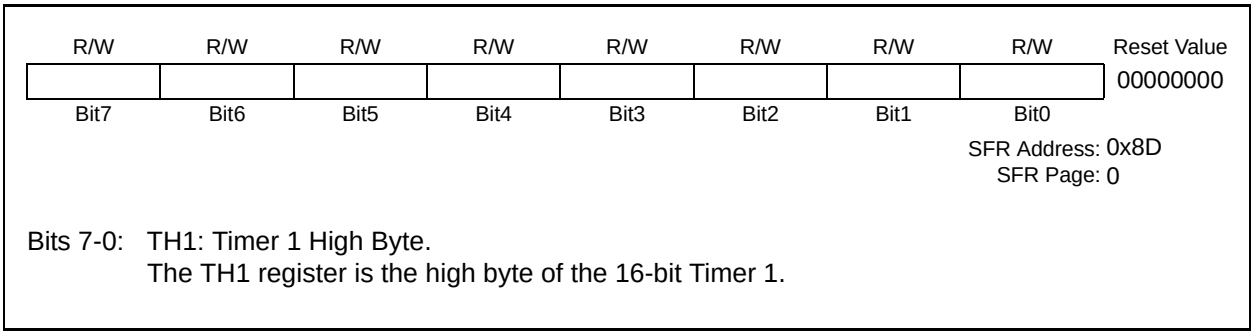
Bits 7-0: TL1: Timer 1 Low Byte.
The TL1 register is the low byte of the 16-bit Timer 1.

Figure 24.9. TH0: Timer 0 High Byte



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

Figure 24.10. TH1: Timer 1 High Byte



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

Figure 24.18. TMRnH: Timer 2, 3, and 4 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: TMR2H: 0xCD; TMR3H: 0xCD; TMR4H: 0xCD

SFR Page: TMR2H: page 0; TMR3H: page 1; TMR4H: page 2

Bits 7-0: TH2, 3, and 4: Timer 2, 3, and 4 High Byte.
The TH2, 3, and 4 register contains the high byte of the 16-bit Timer 2, 3, and 4

25.1. PCA Counter/Timer

The 16-bit PCA counter/timer consists of two 8-bit SFRs: PCA0L and PCA0H. PCA0H is the high byte (MSB) of the 16-bit counter/timer and PCA0L is the low byte (LSB). Reading PCA0L automatically latches the value of PCA0H into a “snapshot” register; the following PCA0H read accesses this “snapshot” register. Reading the PCA0L Register first guarantees an accurate reading of the entire 16-bit PCA0 counter. Reading PCA0H or PCA0L does not disturb the counter operation. The CPS2-CPS0 bits in the PCA0MD register select the timebase for the counter/timer as shown in Table 25.1.

When the counter/timer overflows from 0xFFFF to 0x0000, the Counter Overflow Flag (CF) in PCA0MD is set to logic 1 and an interrupt request is generated if CF interrupts are enabled. Setting the ECF bit in PCA0MD to logic 1 enables the CF flag to generate an interrupt request. The CF bit is not automatically cleared by hardware when the CPU vectors to the interrupt service routine, and must be cleared by software (Note: PCA0 interrupts must be globally enabled before CF interrupts are recognized. PCA0 interrupts are globally enabled by setting the EA bit (IE.7) and the EPCA0 bit in EIE1 to logic 1). Clearing the CIDL bit in the PCA0MD register allows the PCA to continue normal operation while the CPU is in Idle mode.

Table 25.1. PCA Timebase Input Options

CPS2	CPS1	CPS0	Timebase
0	0	0	System clock divided by 12
0	0	1	System clock divided by 4
0	1	0	Timer 0 overflow
0	1	1	High-to-low transitions on ECI (max rate = system clock divided by 4)
1	0	0	System clock
1	0	1	External oscillator source divided by 8 (synchronized with system clock)

Figure 25.2. PCA Counter/Timer Block Diagram