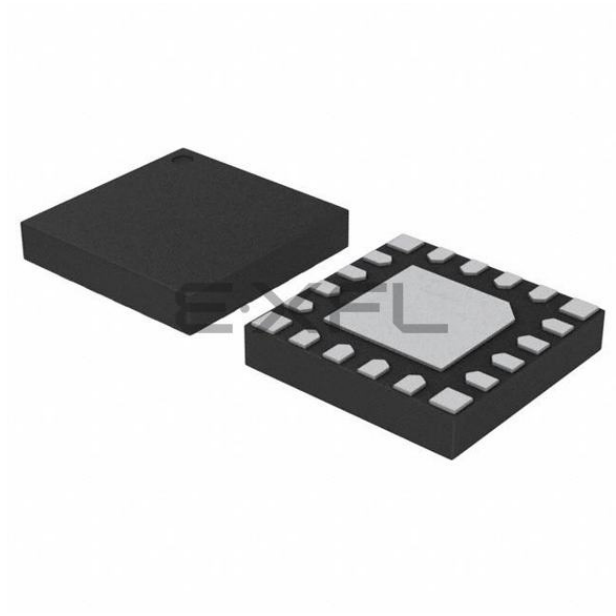

Welcome to E-XFL.COM

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	Obsolete
Core Processor	CIP-51 8051
Core Size	8-Bit
Speed	25MHz
Connectivity	SMBus (2-Wire/I²C), SPI, UART/USART
Peripherals	Brown-out Detect/Reset, Cap Sense, POR, PWM, Temp Sensor, WDT
Number of I/O	16
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 9x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	20-UFQFN Exposed Pad
Supplier Device Package	20-QFN (3x3)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f990-gmr

1.3. Serial Ports

The C8051F99x-C8051F98x Family includes an SMBus/I^2C interface, a full-duplex UART with enhanced baud rate configuration, and an Enhanced SPI interface. Each of the serial buses is fully implemented in hardware and makes extensive use of the CIP-51's interrupts, thus requiring very little CPU intervention.

1.4. Programmable Counter Array

An on-chip programmable counter/timer array (PCA) is included in addition to the four 16-bit general purpose counter/timers. The PCA consists of a dedicated 16-bit counter/timer time base with three programmable capture/compare modules. The PCA clock is derived from one of seven sources: the system clock divided by 12, the system clock divided by 4, Timer 0 overflows, an External Clock Input (ECI), the system clock, the external oscillator clock source divided by 8, or the SmaRTClock divided by 8.

Each capture/compare module can be configured to operate in a variety of modes: edge-triggered capture, software timer, high-speed output, pulse width modulator (8, 9, 10, 11, or 16-bit), or frequency output. Additionally, Capture/Compare Module 2 offers watchdog timer (WDT) capabilities. Following a system reset, Module 2 is configured and enabled in WDT mode. The PCA Capture/Compare Module I/O and External Clock Input may be routed to Port I/O via the Digital Crossbar.

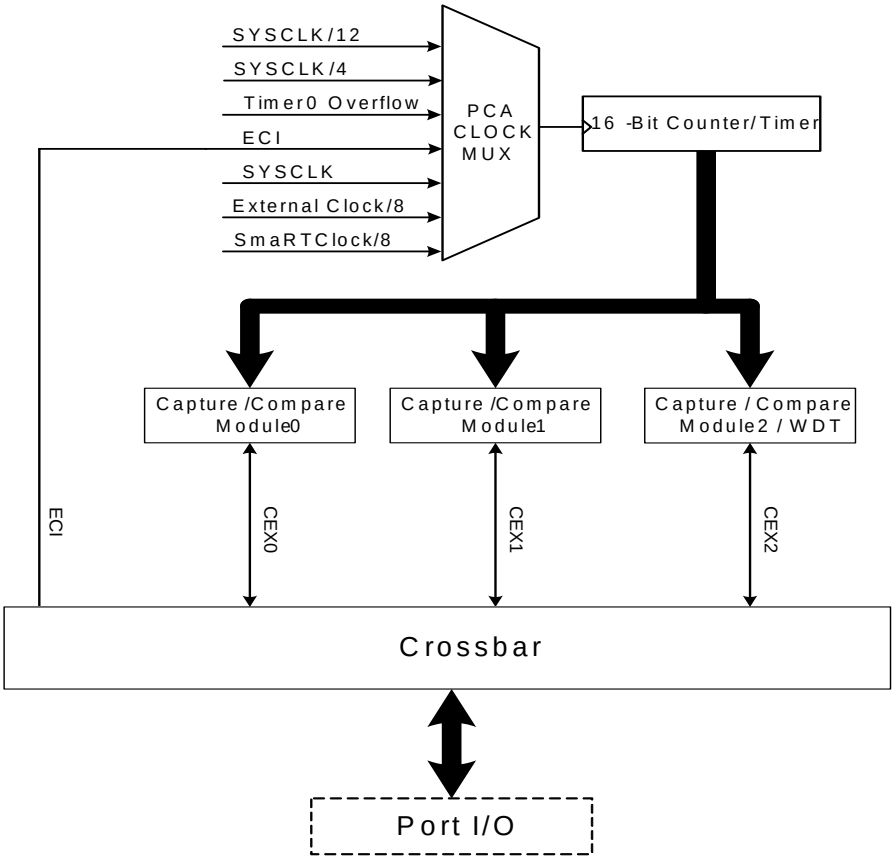

Figure 1.15. PCA Block Diagram

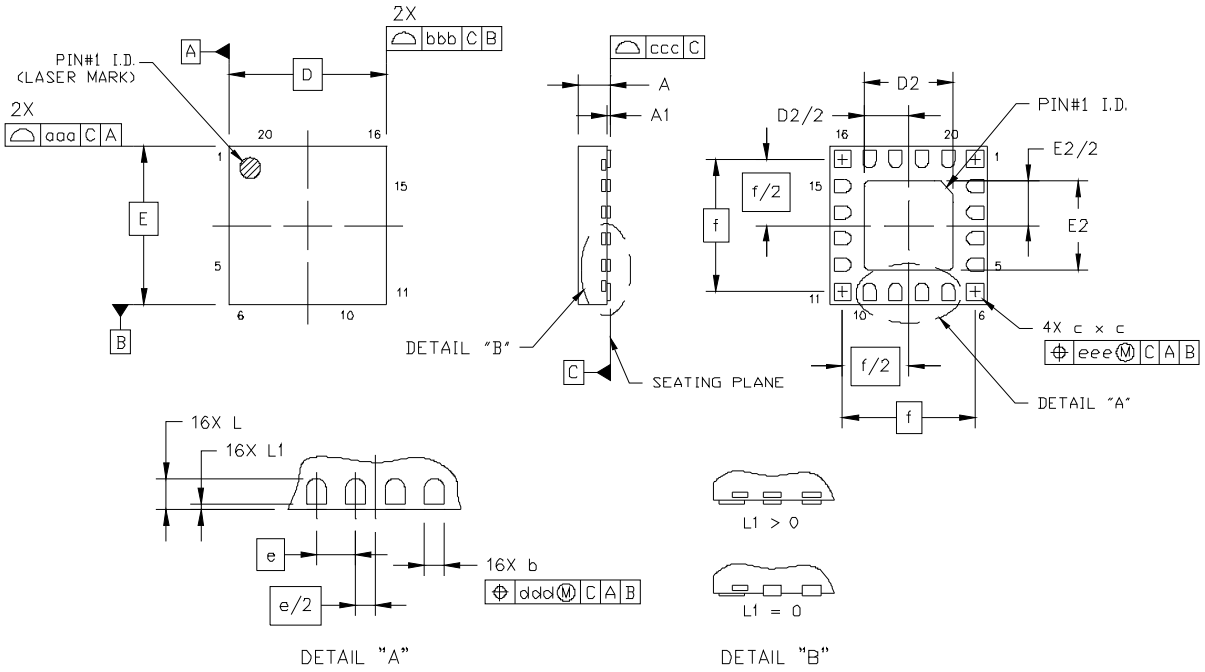

Figure 3.7. QFN-20 Package Drawing

Table 3.2. QFN-20 Package Dimensions

Dimension	Min	Typ	Max		Dimension	Min	Typ	Max
A	0.50	0.55	0.60		f	2.53 BSC		
A1	0.00	0.02	0.05		L	0.35	0.40	0.45
b	0.20	0.25	0.30		L1	0.00	—	0.10
c	0.27	0.32	0.375		aaa	—	—	0.05
D	3.00 BSC				bbb	—	—	0.05
D2	1.6	1.70	1.8		ccc		—	0.08
e	0.50 BSC				ddd	—	—	0.10
E	3.00 BSC				eee	—	—	0.10
E2	1.6	1.70	1.8					

Notes:

1. All dimensions shown are in millimeters (mm) unless otherwise noted.
2. Dimensioning and Tolerancing per ANSI Y14.5M-1994.

5.9. Voltage and Ground Reference Options

The voltage reference MUX is configurable to use an externally connected voltage reference, the internal voltage reference, 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 90. 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 215 for complete Port I/O configuration details. The external reference voltage must be within the range $0 \le V_{REF} \le VDD$ and the external ground reference must be at the same DC voltage potential as GND.

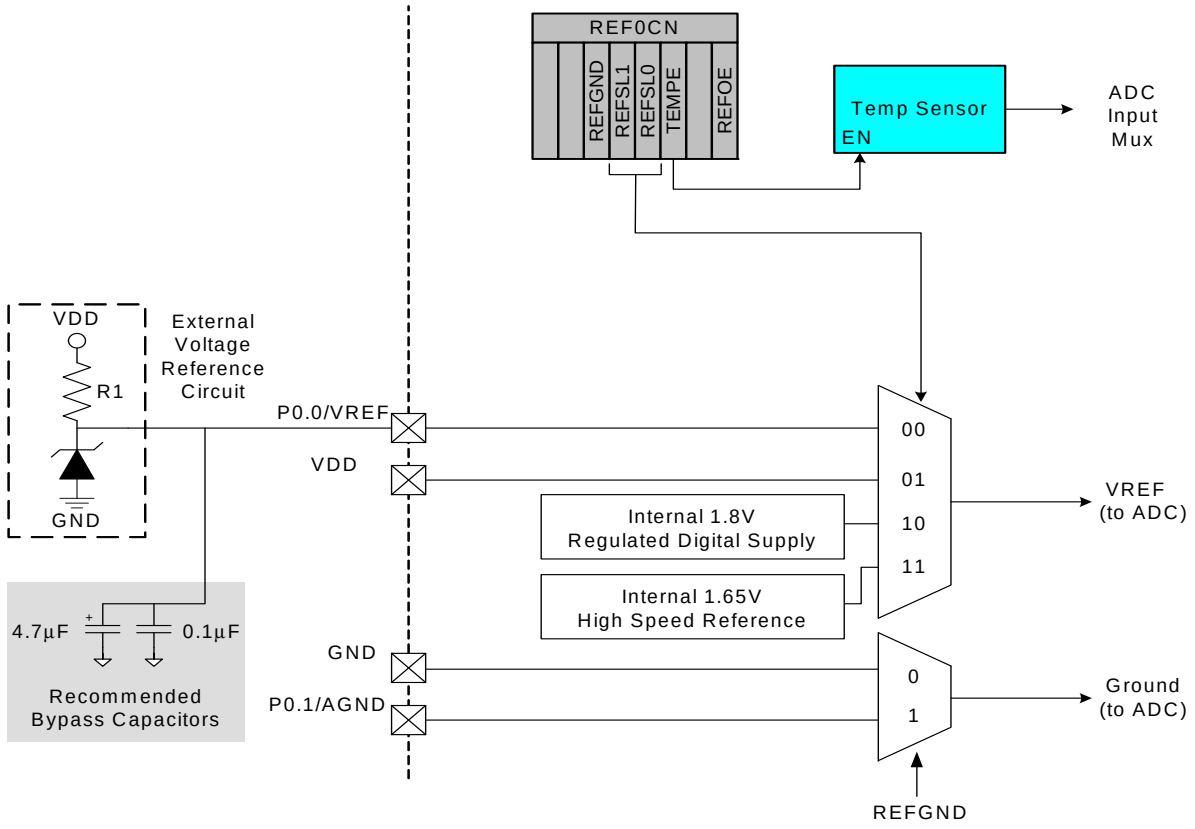

Figure 5.10. Voltage Reference Functional Block Diagram

8.12. CS0 Pin Monitor

The CS0 module provides accurate conversions in all operating modes of the CPU, peripherals and I/O ports. Pin monitoring circuits are provided to improve interference immunity from high-current output pin switching. The CS0 Pin Monitor register (CS0PM, SFR Definition 8.14) controls the operation of these pin monitors.

Conversions in the CS0 module are immune to any change on digital inputs and immune to most output switching. Even high-speed serial data transmission will not affect CS0 operation as long as the output load is limited. Output changes that switch large loads such as LEDs and heavily-loaded communications lines can affect conversion accuracy. For this reason, the CS0 module includes pin monitoring circuits that will, if enabled, automatically adjust conversion timing if necessary to eliminate any effect from high-current output pin switching.

The pin monitor enable bit should be set for any output signal that is expected to drive a large load.

Example: The SMBus in a system is heavily loaded with multiple slaves and a long PCB route. Set the SMBus pin monitor enable, SMBPM = 1.

Example: Timer2 controls an LED on Port 1, pin 3 to provide variable dimming. Set the Port SFR write monitor enable, PIOPM = 1.

Example: The SPI bus is used to communicate to a nearby host. The pin monitor is not needed because the output is not heavily loaded, SPIPM remains = 0, the default reset state.

Pin monitors should not be enabled unless they are required. The pin monitor works by repeating any portion of a conversion that may have been corrupted by a change on an output pin. Setting pin monitor enables bits will slow CS0 conversions.

The frequency of CS0 retry operations can be limited by setting the CSPMMD bits. In the default (reset) state, all converter retry requests will be performed. This is the recommended setting for all applications. The number of retries per conversion can be limited to either two or four retries by changing CSPMMD. Limiting the number of retries per conversion ensures that even in circumstances where extremely frequent high-power output switching occurs, conversions will be completed, though there may be some loss of accuracy due to switching noise.

Activity of the pin monitor circuit can be detected by reading the Pin Monitor Event bit, CS0PME, in register CS0CN. This bit will be set if any CS0 converter retries have occurred. It remains set until cleared by software or a device reset.

8.13. Adjusting CS0 For Special Situations

There are several configuration options in the CS0 module designed to modify the operation of the circuit and address special situations. In particular, any circuit with more than 500 ohms of series impedance between the sensor and the device pin may require adjustments for optimal performance. Typical applications which may require adjustments include:

- Touch panel sensors fabricated using a resistive conductor such as indium-tin-oxide (ITO).
- Circuits using a high-value series resistor to isolate the sensor element for high ESD protection.

Most systems will require no fine tuning, and the default settings for CS0DT, CS0DR, CS0IA, CS0RP and CS0LP should be used.

SFR Definition 8.11. CS0MD1: Capacitive Sense Mode 1

Bit	7	6	5	4	3	2	1	0
Name	Reserved	CS0POL	CS0DR		CS0WOI	CS0CG[2:0]		
Type	R/W					R/W	R/W	R/W
Reset	0	0	0	0	0	1	1	1

SFR Page = 0x0; SFR Address = 0xAF

Bit	Name	Description
7	Reserved	Must write 0.
6	CS0POL	**CS0 Digital Comparator Polarity Select.** 0: The digital comparator generates an interrupt if the conversion is greater than the threshold. 1: The digital comparator generates an interrupt if the conversion is less than or equal to the threshold.
5:4	CS0DR[1:0]	**CS0 Double Reset Select.** These bits adjust the secondary CS0 reset time. For most touch-sensitive switches, the default (fastest) value is sufficient. See the discussion in Section 8.13 for more information. 00: No additional time is used for secondary reset. 01: An additional 0.75 µs is used for secondary reset. 10: An additional 1.5 µs is used for secondary reset. 11: An additional 2.25 µs is used for secondary reset.
3	CS0WOI	**CS0 Wake on Interrupt Configuration.** 0: Wake-up event generated on digital comparator interrupt only. 1: Wake-up event generated on end of scan or digital comparator interrupt.
2:0	CS0CG[2:0]	**CS0 Capacitance Gain Select.** These bits select the gain applied to the capacitance measurement. Lower gain values increase the size of the capacitance that can be measured with the CS0 module. The capacitance gain is equivalent to CS0CG[2:0] + 1. 000: Gain = 1x 001: Gain = 2x 010: Gain = 3x 011: Gain = 4x 100: Gain = 5x 101: Gain = 6x 110: Gain = 7x 111: Gain = 8x (default)

9. CIP-51 Microcontroller

The MCU system controller core is the CIP-51 microcontroller. The CIP-51 is fully compatible with the MCS-51™ instruction set; standard 803x/805x assemblers and compilers can be used to develop software. The MCU family has a superset of all the peripherals included with a standard 8051. The CIP-51 also includes on-chip debug hardware (see description in Section 27) and interfaces directly with the analog and digital subsystems providing a complete data acquisition or control-system solution in a single integrated circuit.

The CIP-51 Microcontroller core implements the standard 8051 organization and peripherals as well as additional custom peripherals and functions to extend its capability (see Figure 9.1 for a block diagram). The CIP-51 includes the following features:

- Fully Compatible with MCS-51 Instruction Set
- 25 MIPS Peak Throughput with 25 MHz Clock
- 0 to 25 MHz Clock Frequency
- Extended Interrupt Handler
- Reset Input
- Power Management Modes
- On-chip Debug Logic
- Program and Data Memory Security

9.1. Performance

The CIP-51 employs a pipelined architecture that greatly increases its instruction throughput over the standard 8051 architecture. In a standard 8051, all instructions except for MUL and DIV take 12 or 24 system clock cycles to execute and usually have a maximum system clock of 12 MHz. By contrast, the CIP-51 core executes 70% of its instructions in one or two system clock cycles, with no instructions taking more than eight system clock cycles.

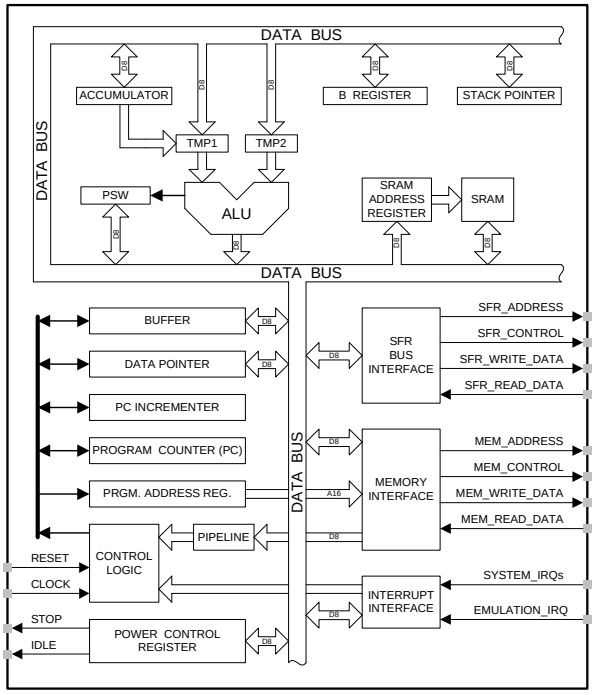

Figure 9.1. CIP-51 Block Diagram

Table 9.1. CIP-51 Instruction Set Summary (Continued)

Mnemonic	Description	Bytes	Clock Cycles
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	Jump if direct bit is not set	3	3/4
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

13.3. Interrupt Priorities

Each interrupt source can be individually programmed to one of two priority levels: low or high. A low priority interrupt service routine can be preempted by a high priority interrupt. A high priority interrupt cannot be preempted. If a high priority interrupt preempts a low priority interrupt, the low priority interrupt will finish execution after the high priority interrupt completes. Each interrupt has an associated interrupt priority bit in in the Interrupt Priority and Extended Interrupt Priority registers used to configure its priority level. Low priority is the default.

If two interrupts are recognized simultaneously, the interrupt with the higher priority is serviced first. If both interrupts have the same priority level, a fixed priority order is used to arbitrate. See Table 13.1 on page 140 to determine the fixed priority order used to arbitrate between simultaneously recognized interrupts.

13.4. Interrupt Latency

Interrupt response time depends on the state of the CPU when the interrupt occurs. Pending interrupts are sampled and priority decoded each system clock cycle. Therefore, the fastest possible response time is 7 system clock cycles: 1 clock cycle to detect the interrupt, 1 clock cycle to execute a single instruction, and 5 clock cycles to complete the LCALL to the ISR. If an interrupt is pending when a RETI is executed, a single instruction is executed before an LCALL is made to service the pending interrupt. Therefore, the maximum response time for an interrupt (when no other interrupt is currently being serviced or the new interrupt is of greater priority) occurs when the CPU is performing an RETI instruction followed by a DIV as the next instruction. In this case, the response time is 19 system clock cycles: 1 clock cycle to detect the interrupt, 5 clock cycles to execute the RETI, 8 clock cycles to complete the DIV instruction and 5 clock cycles to execute the LCALL to the ISR. If the CPU is executing an ISR for an interrupt with equal or higher priority, the new interrupt will not be serviced until the current ISR completes, including the RETI and following instruction.

14.3. Security Options

The CIP-51 provides security options to protect the Flash memory from inadvertent modification by software as well as to prevent the viewing of proprietary program code and constants. The Program Store Write Enable (bit PSWE in register PSCTL) and the Program Store Erase Enable (bit PSEE in register PSCTL) bits protect the Flash memory from accidental modification by software. PSWE must be explicitly set to 1 before software can modify the Flash memory; both PSWE and PSEE must be set to 1 before software can erase Flash memory. Additional security features prevent proprietary program code and data constants from being read or altered across the C2 interface.

A Security Lock Byte located at the last byte of Flash user space offers protection of the Flash program memory from access (reads, writes, or erases) by unprotected code or the C2 interface. See **Section "10. Memory Organization" on page 128** for the location of the security byte. The Flash security mechanism allows the user to lock *n* 512-byte Flash pages, starting at page 0 (addresses 0x0000 to 0x01FF), where *n* is the 1s complement number represented by the Security Lock Byte. **The page containing the Flash Security Lock Byte is unlocked when no other Flash pages are locked (all bits of the Lock Byte are 1) and locked when any other Flash pages are locked (any bit of the Lock Byte is 0).**

Security Lock Byte:	1111 1011b
ones Complement:	0000 0100b
Flash pages locked:	5 (First four Flash pages + Lock Byte Page)

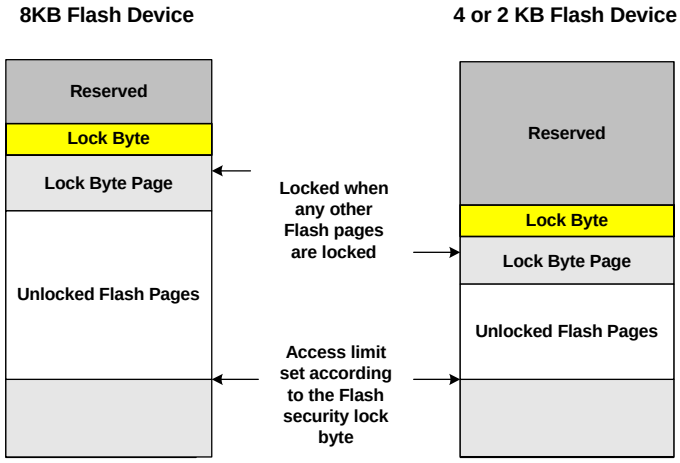

Figure 14.1. Flash Program Memory Map (8 kB and smaller devices)

The level of Flash security depends on the Flash access method. The three Flash access methods that can be restricted are reads, writes, and erases from the C2 debug interface, user firmware executing on unlocked pages, and user firmware executing on locked pages. Table 14.1 summarizes the Flash security features of the C8051F99x-C8051F98x devices.

20. SmaRTClock (Real Time Clock)

C8051F99x-C8051F98x devices include an ultra low power 32-bit SmaRTClock Peripheral (Real Time Clock) with alarm. The SmaRTClock has a dedicated 32 kHz oscillator that can be configured for use with or without a crystal. No external resistor or loading capacitors are required. The on-chip loading capacitors are programmable to 16 discrete levels allowing compatibility with a wide range of crystals. The SmaRTClock can operate directly from a 1.8–3.6 V battery voltage and remains operational even when the device goes into its lowest power down mode. The SmaRTClock output can be buffered and routed to a GPIO pin to provide an accurate, low frequency clock to other devices while the MCU is in its lowest power down mode (see "PMU0MD: Power Management Unit Mode" on page 170 for more details). C8051F99x-C8051F98x devices also support an ultra low power internal LFO that reduces sleep mode current.

The SmaRTClock allows a maximum of 36 hour 32-bit independent time-keeping when used with a 32.768 kHz Watch Crystal. The SmaRTClock provides an Alarm and Missing SmaRTClock events, which could be used as reset or wakeup sources. See Section "18. Reset Sources" on page 181 and Section "15. Power Management" on page 162 for details on reset sources and low power mode wake-up sources, respectively.

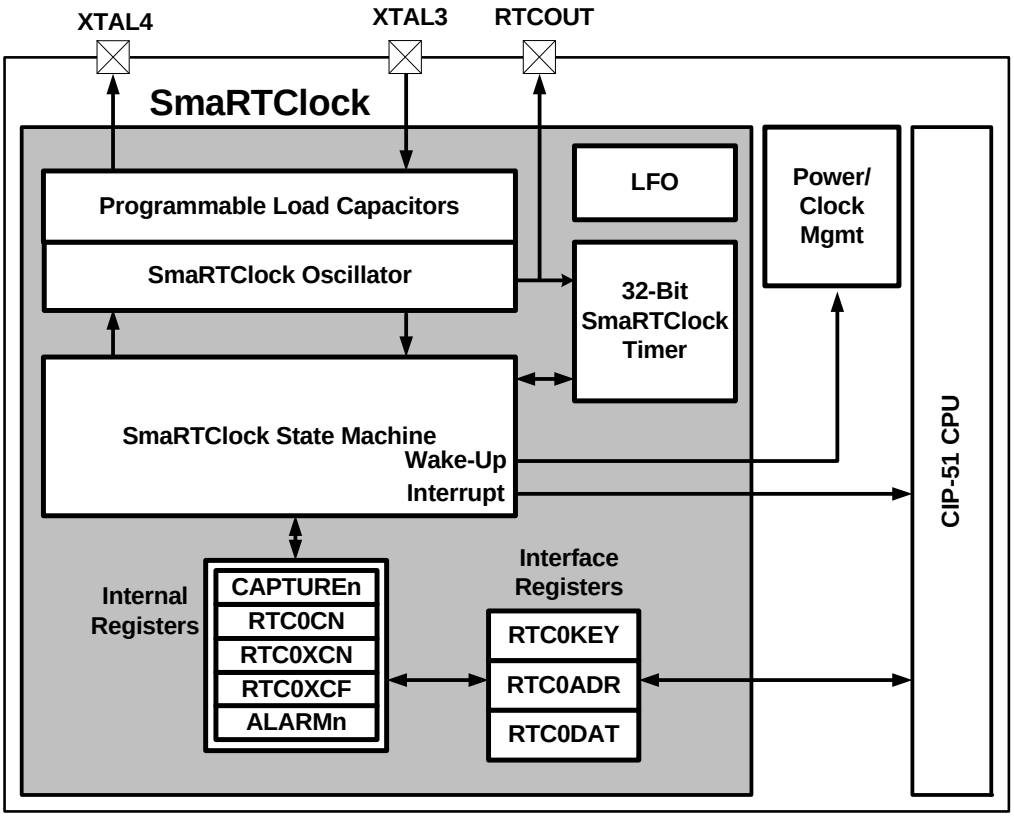

Figure 20.1. SmaRTClock Block Diagram

SFR Definition 20.2. RTC0ADR: SmaRTClock Address

Bit	7	6	5	4	3	2	1	0
Name	BUSY	AUTORD		SHORT	ADDR[3:0]			
Type	R/W	R/W	R	R/W	R/W			
Reset	0	0	0	0	0	0	0	0

SFR Page = 0x0; SFR Address = 0xAC

Bit	Name	Function
7	BUSY	**SmaRTClock Interface Busy Indicator.** Indicates SmaRTClock interface status. Writing 1 to this bit initiates an indirect read.
6	AUTORD	**SmaRTClock Interface Autoread Enable.** Enables/disables Autoread. 0: Autoread Disabled. 1: Autoread Enabled.
5	Unused	Read = 0b; Write = Don't Care.
4	SHORT	**Short Strobe Enable.** Enables/disables the Short Strobe Feature. 0: Short Strobe disabled. 1: Short Strobe enabled.
3:0	ADDR[3:0]	**SmaRTClock Indirect Register Address.** Sets the currently selected SmaRTClock register. See Table 20.1 for a listing of all SmaRTClock indirect registers.
Note: The ADDR bits increment after each indirect read/write operation that targets a CAPTUREn or ALARMn internal SmaRTClock register.		

SFR Definition 20.3. RTC0DAT: SmaRTClock Data

Bit	7	6	5	4	3	2	1	0
Name	RTC0DAT[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Page= 0x0; SFR Address = 0xAD

Bit	Name	Function
7:0	RTC0DAT	**SmaRTClock Data Bits.** Holds data transferred to/from the internal SmaRTClock register selected by RTC0ADR.
Note: Read-modify-write instructions (orl, anl, etc.) should not be used on this register.		

20.2.2. Using the SmaRTClock Oscillator in Self-Oscillate Mode

When using Self-Oscillate Mode, the XTAL3 and XTAL4 pins are internally shorted together. The following steps show how to configure SmaRTClock for use in Self-Oscillate Mode:

1. Set SmaRTClock to Self-Oscillate Mode (XMODE = 0).
2. Set the desired oscillation frequency:
   For oscillation at about 20 kHz, set BIASX2 = 0.
   For oscillation at about 40 kHz, set BIASX2 = 1.
3. The oscillator starts oscillating instantaneously.
4. Fine tune the oscillation frequency by adjusting the load capacitance (RTC0XCF).

20.2.3. Using the Low Frequency Oscillator (LFO)

The low frequency oscillator provides an ultra low power, on-chip clock source to the SmaRTClock. The typical frequency of oscillation is 16.4 kHz ±20%. No external components are required to use the LFO and the XTAL3 and XTAL4 pins do not need to be shorted together.

The following steps show how to configure SmaRTClock for use with the LFO:

1. Enable and select the Low Frequency Oscillator (LFOEN = 1).
2. The LFO starts oscillating instantaneously.

When the LFO is enabled, the SmaRTClock oscillator increments bit 1 of the 32-bit timer (instead of bit 0). This effectively multiplies the LFO frequency by 2, making the RTC timebase behave as if a 32.768 kHz crystal is connected at the output.

22.1. Supporting Documents

It is assumed the reader is familiar with or has access to the following supporting documents:

- The I^2C-Bus and How to Use It (including specifications), Philips Semiconductor.
- The I^2C-Bus Specification—Version 2.0, Philips Semiconductor.
- System Management Bus Specification—Version 1.1, SBS Implementers Forum.

22.2. SMBus Configuration

Figure 22.2 shows a typical SMBus configuration. The SMBus specification allows any recessive voltage between 3.0 V and 5.0 V; different devices on the bus may operate at different voltage levels.

Note: The port pins on C8051F99x-C8051F98x devices are not 5 V tolerant, therefore, the device may only be used in SMBus networks where the supply voltage does not exceed V_{DD}.

The bi-directional SCL (serial clock) and SDA (serial data) lines must be connected to a positive power supply voltage through a pullup resistor or similar circuit. Every device connected to the bus must have an open-drain or open-collector output for both the SCL and SDA lines, so that both are pulled high (recessive state) when the bus is free. The maximum number of devices on the bus is limited only by the requirement that the rise and fall times on the bus not exceed 300 ns and 1000 ns, respectively.

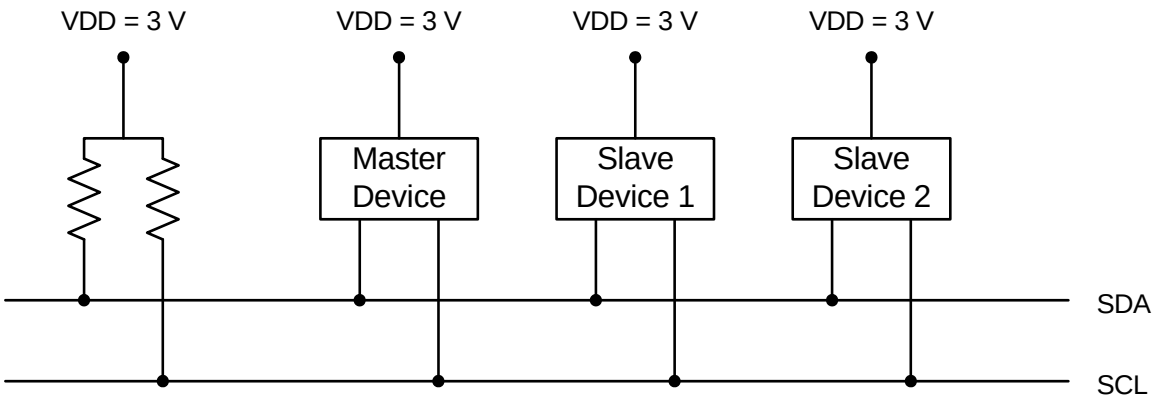

Figure 22.2. Typical SMBus Configuration

Table 22.3. Sources for Hardware Changes to SMB0CN

Bit	Set by Hardware When:	Cleared by Hardware When:
MASTER	■ A START is generated.	■ A STOP is generated. ■ Arbitration is lost.
TXMODE	■ START is generated. ■ SMB0DAT is written before the start of an SMBus frame.	■ A START is detected. ■ Arbitration is lost. ■ SMB0DAT is not written before the start of an SMBus frame.
STA	■ A START followed by an address byte is received.	■ Must be cleared by software.
STO	■ A STOP is detected while addressed as a slave. ■ Arbitration is lost due to a detected STOP.	■ A pending STOP is generated.
ACKRQ	■ A byte has been received and an ACK response value is needed (only when hardware ACK is not enabled).	■ After each ACK cycle.
ARBLOST	■ A repeated START is detected as a MASTER when STA is low (unwanted repeated START). ■ SCL is sensed low while attempting to generate a STOP or repeated START condition. ■ SDA is sensed low while transmitting a 1 (excluding ACK bits).	■ Each time SI is cleared.
ACK	■ The incoming ACK value is low (ACKNOWLEDGE).	■ The incoming ACK value is high (NOT ACKNOWLEDGE).
SI	■ A START has been generated. ■ Lost arbitration. ■ A byte has been transmitted and an ACK/NACK received. ■ A byte has been received. ■ A START or repeated START followed by a slave address + R/W has been received. ■ A STOP has been received.	■ Must be cleared by software.

SFR Definition 25.8. TMR2CN: Timer 2 Control

Bit	7	6	5	4	3	2	1	0
Name	TF2H	TF2L	TF2LEN	TF2CEN	T2SPLIT	TR2	T2XCLK[1:0]	
Type	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 = 0xC8; Bit-Addressable

Bit	Name	Function
7	TF2H	**Timer 2 High Byte Overflow Flag.** Set by hardware when the Timer 2 high byte overflows from 0xFF to 0x00. In 16 bit mode, this will occur when Timer 2 overflows from 0xFFFF to 0x0000. When the Timer 2 interrupt is enabled, setting this bit causes the CPU to vector to the Timer 2 interrupt service routine. This bit is not automatically cleared by hardware.
6	TF2L	**Timer 2 Low Byte Overflow Flag.** Set by hardware when the Timer 2 low byte overflows from 0xFF to 0x00. TF2L will be set when the low byte overflows regardless of the Timer 2 mode. This bit is not automatically cleared by hardware.
5	TF2LEN	**Timer 2 Low Byte Interrupt Enable.** When set to 1, this bit enables Timer 2 Low Byte interrupts. If Timer 2 interrupts are also enabled, an interrupt will be generated when the low byte of Timer 2 overflows.
4	TF2CEN	**Timer 2 Capture Enable.** When set to 1, this bit enables Timer 2 Capture Mode.
3	T2SPLIT	**Timer 2 Split Mode Enable.** When set to 1, Timer 2 operates as two 8-bit timers with auto-reload. Otherwise, Timer 2 operates in 16-bit auto-reload mode.
2	TR2	**Timer 2 Run Control.** Timer 2 is enabled by setting this bit to 1. In 8-bit mode, this bit enables/disables TMR2H only; TMR2L is always enabled in split mode.
1:0	T2XCLK[1:0]	**Timer 2 External Clock Select.** This bit selects the “external” and “capture trigger” clock sources for Timer 2. If Timer 2 is in 8-bit mode, this bit selects the “external” clock source for both timer bytes. Timer 2 Clock Select bits (T2MH and T2ML in register CKCON) may still be used to select between the “external” clock and the system clock for either timer. Note: External clock sources are synchronized with the system clock. 00: External Clock is SYSCLK/12. Capture trigger is SmaRTClock/8. 01: External Clock is Comparator 0. Capture trigger is SmaRTClock/8. 10: External Clock is SYSCLK/12. Capture trigger is Comparator 0. 11: External Clock is SmaRTClock/8. Capture trigger is Comparator 0.

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 SmaRTClock oscillator period with respect to another oscillator.

Timer 3 may be clocked by the system clock, the system clock divided by 12, external oscillator source divided by 8, or the SmaRTClock oscillator. The external oscillator source divided by 8 and SmaRTClock oscillator 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 SmaRTClock oscillator. 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.

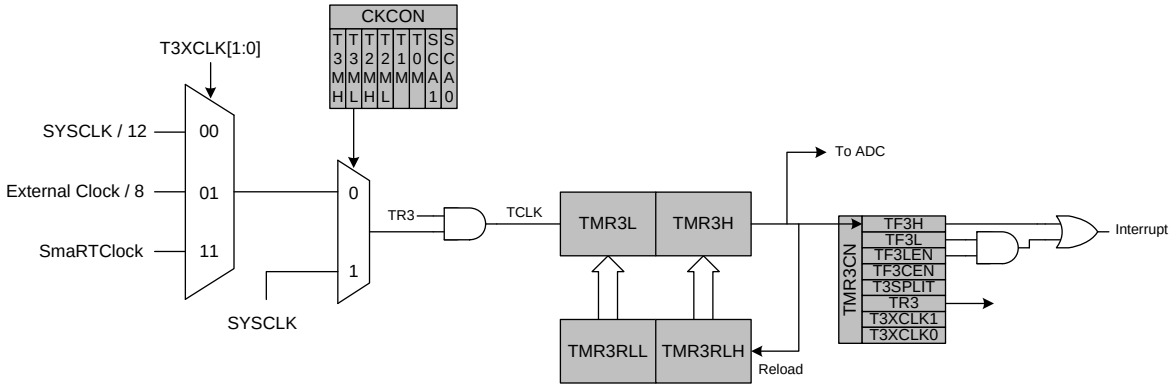

Figure 25.7. Timer 3 16-Bit Mode Block Diagram

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				CCF2	CCF1	CCF0
Type	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.
5:3	Unused	Read = 000b, Write = don't care.
2:0	CCF[2:0]	**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 26.3. PCA0PWM: PCA PWM Configuration

Bit	7	6	5	4	3	2	1	0
Name	ARSEL	ECOV	COVF				CLSEL[1:0]	
Type	R/W	R/W	R/W	R	R	R	R/W	
Reset	0	0	0	0	0	0	0	0

SFR Page = 0x0; SFR Address = 0xDF

Bit	Name	Function
7	ARSEL	**Auto-Reload Register Select.** This bit selects whether to read and write the normal PCA capture/compare registers (PCA0CPn), or the Auto-Reload registers at the same SFR addresses. This function is used to define the reload value for 9, 10, and 11-bit PWM modes. In all other modes, the Auto-Reload registers have no function. 0: Read/Write Capture/Compare Registers at PCA0CPHn and PCA0CPLn. 1: Read/Write Auto-Reload Registers at PCA0CPHn and PCA0CPLn.
6	ECOV	**Cycle Overflow Interrupt Enable.** This bit sets the masking of the Cycle Overflow Flag (COVF) interrupt. 0: COVF will not generate PCA interrupts. 1: A PCA interrupt will be generated when COVF is set.
5	COVF	**Cycle Overflow Flag.** This bit indicates an overflow of the 8th, 9th, 10th, or 11th bit of the main PCA counter (PCA0). The specific bit used for this flag depends on the setting of the Cycle Length Select bits. The bit can be set by hardware or software, but must be cleared by software. 0: No overflow has occurred since the last time this bit was cleared. 1: An overflow has occurred since the last time this bit was cleared.
4:2	Unused	Read = 000b; Write = don't care.
1:0	CLSEL[1:0]	**Cycle Length Select.** When 16-bit PWM mode is not selected, these bits select the length of the PWM cycle, between 8, 9, 10, or 11 bits. This affects all channels configured for PWM which are not using 16-bit PWM mode. These bits are ignored for individual channels configured to16-bit PWM mode. 00: 8 bits. 01: 9 bits. 10: 10 bits. 11: 11 bits.