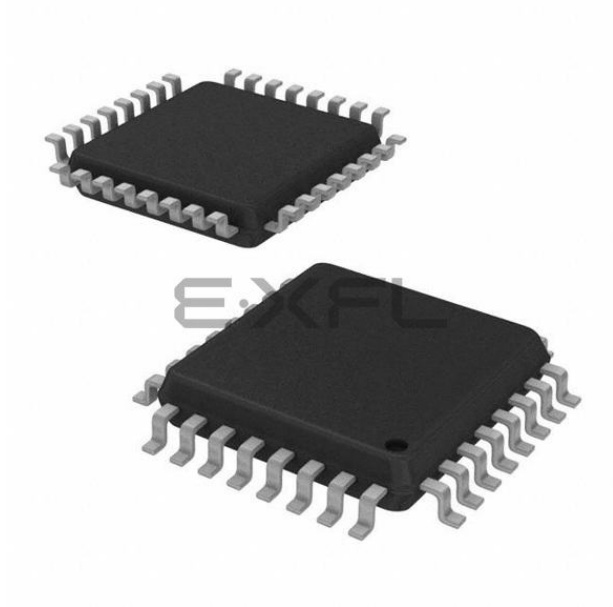

Welcome to E-XFL.COM

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	Active
Core Processor	8051
Core Size	8-Bit
Speed	50MHz
Connectivity	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	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2.25K x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.25V
Data Converters	A/D 24x12b; D/A 2x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	32-LQFP
Supplier Device Package	32-LQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f410-gqr

NOTES:

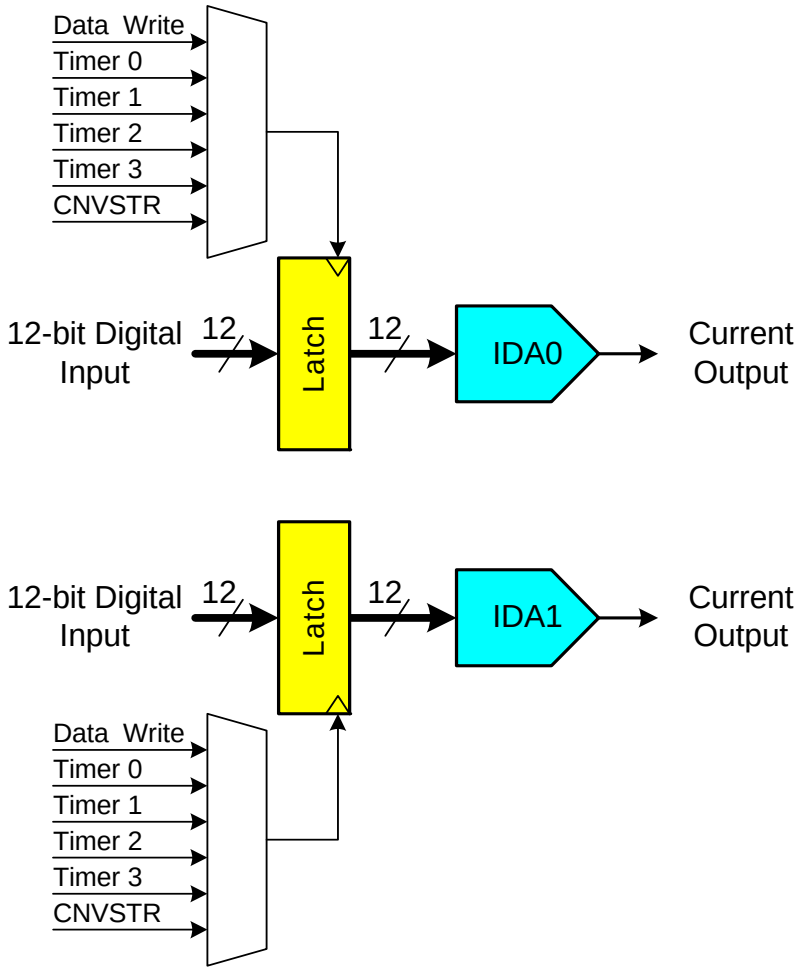

Figure 1.8. IDAC Block Diagram

1.7. Programmable Comparators

C8051F41x devices include two software-configurable voltage comparators with an input multiplexer. Each comparator offers programmable response time and hysteresis and two outputs that are optionally available at the Port pins: a synchronous “latched” output (CP0 and CP1), or an asynchronous “raw” output (CP0A and CP1A). Comparator interrupts may be generated on rising, falling, or both edges. When in IDLE or SUSPEND mode, these interrupts may be used as a “wake-up” source for the processor. Comparator0 may also be configured as a reset source. A block diagram of the comparator is shown in Figure 1.9.

3. Global DC Electrical Characteristics

Table 3.1. Global DC Electrical Characteristics

–40 to +85 °C, 50 MHz System Clock unless otherwise specified. Typical values are given at 25 °C

Parameter	Conditions	Min	Typ	Max	Units
Supply Input Voltage (V_{REGIN})[1]	Output Current = 1 mA	2.15	—	5.25	V
Core Supply Voltage (V_{DD})		2.0	—	2.75	V
I/O Supply Voltage (V_{IO})[2]		2.0	—	5.25	V
Backup Supply Voltage ($V_{RTC-BACKUP}$)[3]		1.0	—	5.25	V
Backup Supply Current ($I_{RTC-BACKUP}$) (V_{DD} = 0 V, smaRTClock clock = 32 kHz)	$V_{RTC-BACKUP}$ = 1.0 V: at –40 ºC at 25 ºC at 85 ºC $V_{RTC-BACKUP}$ = 1.8 V: at –40 ºC at 25 ºC at 85 ºC $V_{RTC-BACKUP}$ = 2.5 V: at –40 ºC at 25 ºC at 85 ºC	 — — — — — — — — —	 0.65 0.9 1.4 0.7 0.92 1.45 0.72 0.95 1.5	 1.5 1.8 2.5 — — — 1.6 1.85 2.6	 µA µA µA µA µA µA µA µA µA
Core Supply RAM Data Retention Voltage		—	1.5	—	V
SYSCLK (System Clock)[4,5]		0	—	50	MHz
Specified Operating Temperature Range		–40	—	+85	°C

Notes:

1. For more information on V_{REGIN} characteristics, see Table 8.1 on page 82.
2. VIO must be equal to or greater than VDD.
3. The Backup Supply Voltage ($V_{RTC-BACKUP}$) is used to power the smaRTClock peripheral only.
4. SYSCLK is the internal device clock. For operational speeds in excess of 25 MHz, SYSCLK must be derived from the internal clock multiplier.
5. SYSCLK must be at least 32 kHz to enable debugging.
6. Based on device characterization data, not production tested.
7. Active and Inactive IDD at voltages and frequencies other than those specified can be calculated using the IDD Supply Sensitivity. For example, if the V_{DD} is 2.2 V instead of 2.0 V at 25 MHz: I_{DD} = 5.5 mA typical at 2.0 V and f = 25 MHz. From this, I_{DD} = 5.5 mA + 1.14 x (2.2 V – 2.0 V) = 5.73 mA at 2.2 V and f = 25 MHz.
8. I_{DD} can be estimated for frequencies < 15 MHz by simply multiplying the frequency of interest by the frequency sensitivity number for that range. When using these numbers to estimate I_{DD} for > 15 MHz, the estimate should be the current at 25 MHz minus the difference in current indicated by the frequency sensitivity number. For example: V_{DD} = 2.0 V; F = 20 MHz, I_{DD} = 5.5 mA – (25 MHz – 20 MHz) x 0.16 mA/MHz = 4.7 mA.
9. Idle IDD can be estimated for frequencies < 1 MHz by simply multiplying the frequency of interest by the frequency sensitivity number for that range. When using these numbers to estimate Idle for > 1 MHz, the estimate should be the current at 25 MHz minus the difference in current indicated by the frequency sensitivity number. For example: V_{DD} = 2.0 V; F = 5 MHz, Idle I_{DD} = 2.8 mA – (25 MHz – 5 MHz) x 0.1 mA/MHz = 0.8 mA.

Table 3.2. Index to Electrical Characteristics Tables

Table Title	Page #
ADC0 Electrical Characteristics (V_{DD} = 2.5 V, V_{REF} = 2.2 V)	67
ADC0 Electrical Characteristics (V_{DD} = 2.1 V, V_{REF} = 1.5 V)	68
IDAC Electrical Characteristics	75
Voltage Reference Electrical Characteristics	79
Voltage Regulator Electrical Specifications	82
Comparator Electrical Characteristics	92
Reset Electrical Characteristics	134
Flash Electrical Characteristics	143
Port I/O DC Electrical Characteristics	163
Oscillator Electrical Characteristics	175

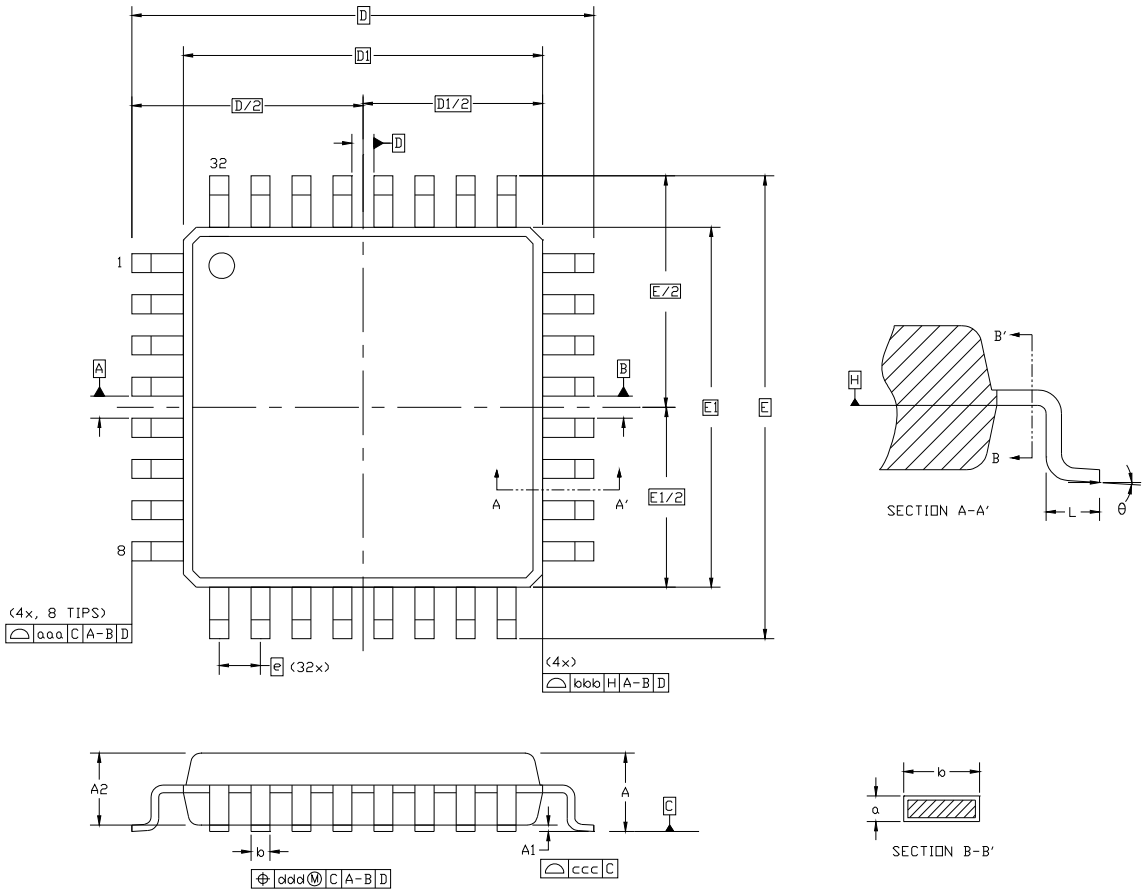

Figure 4.3. LQFP-32 Package Diagram

Table 4.2. LQFP-32 Package Dimensions

	MM		
	MIN	**TYP**	**MAX**
A	—	—	1.60
A1	0.05	—	0.15
A2	1.35	1.40	1.45
b	0.30	0.37	0.45
c	0.09	—	0.20
D	—	9.00	—
D1	—	7.00	—
e	—	0.80	—
E	—	9.00	—
E1	—	7.00	—
L	0.45	0.60	0.75

5.2. Temperature Sensor

The typical temperature sensor transfer function is shown in Figure 5.2. The output voltage (V_{TEMP}) is the positive ADC input when the temperature sensor is selected by bits AD0MX4-0 in register ADC0MX.

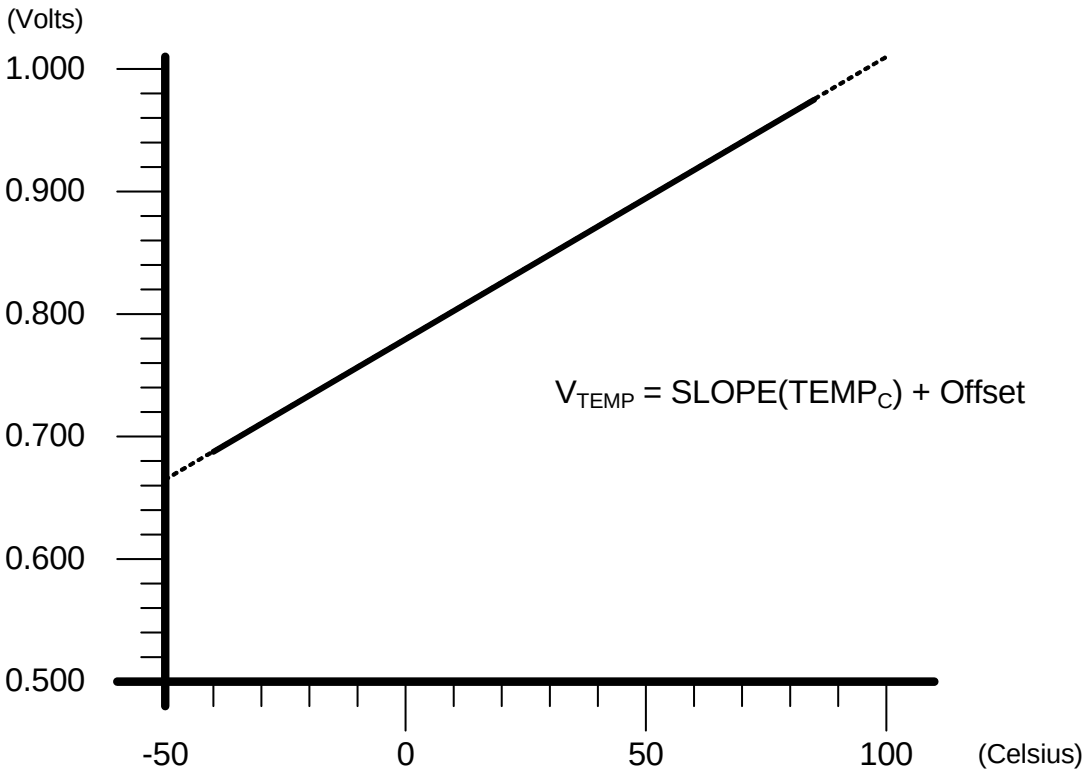

Figure 5.2. Typical Temperature Sensor Transfer Function

5.3. ADC0 Operation

In a typical system, ADC0 is configured using the following steps:

Step 1. Choose the start of conversion source.
Step 2. Choose Normal Mode or Burst Mode operation.
Step 3. If Burst Mode, choose the ADC0 Idle Power State and set the Power-Up Time.
Step 4. Choose the tracking mode. Note that Pre-Tracking Mode can only be used with Normal Mode.
Step 5. Calculate required settling time and set the post convert-start tracking time using the AD0TK bits.
Step 6. Choose the repeat count.
Step 7. Choose the output word justification (Right-Justified or Left-Justified).
Step 8. Enable or disable the End of Conversion and Window Comparator Interrupts.

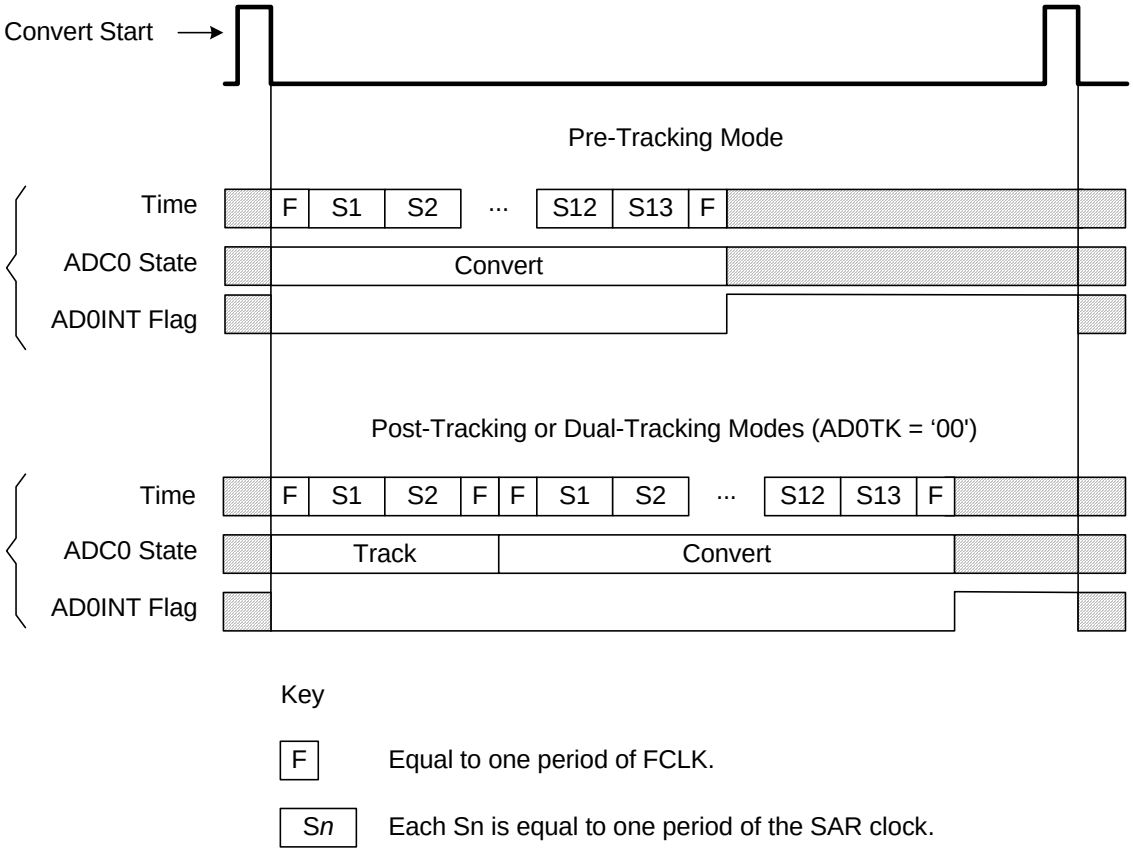

Figure 5.4. 12-Bit ADC Tracking Mode Example

SFR Definition 5.2. ADC0CF: ADC0 Configuration

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
AD0SC					AD0RPT		Reserved	11111000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xBC

Bits7–3: AD0SC4–0: ADC0 SAR Conversion Clock Period Bits.
SAR Conversion clock is derived from FCLK by the following equation, where *AD0SC* refers to the 5-bit value held in bits AD0SC4-0. SAR Conversion clock requirements are given in Table 5.3.
BURSTEN = 0: FCLK is the current system clock.
BURSTEN = 1: FCLK is a maximum of 25 MHz, independent of the current system clock.

$$AD0SC = \frac{FCLK}{CLK_{SAR}} - 1^{*} \quad \text{or} \quad CLK_{SAR} = \frac{FCLK}{AD0SC + 1}$$

***Note:** Round the result up.

Bits2–1: AD0RPT1–0: ADC0 Repeat Count.
Controls the number of conversions taken and accumulated between ADC0 End of Conversion (ADCINT) and ADC0 Window Comparator (ADCWINT) interrupts. A convert start is required for each conversion unless Burst Mode is enabled. In Burst Mode, a single convert start can initiate multiple self-timed conversions. Results in both modes are accumulated in the ADC0H:ADC0L register. **When AD0RPT1-0 are set to a value other than '00', the AD0LJST bit in the ADC0CN register must be set to '0' (right justified).**
00: 1 conversion is performed.
01: 4 conversions are performed and accumulated.
10: 8 conversions are performed and accumulated.
11: 16 conversions are performed and accumulated.

Note: The ADC0 output register is automatically reset to 0x0000 upon reaching the last conversion specified by the repeat counter. If the ADC is disabled during a conversion and re-enabled later, the ADC0H and ADC0L registers should be manually cleared to 0x00.

Bit0: RESERVED. Read = 0b; Must write 0b.

Important Note About the V_{REF} Pin: Port pin P1.2 is used as the external V_{REF} input and as an output for the internal V_{REF}. When using either an external voltage reference or the internal reference circuitry, P1.2 should be configured as an analog pin, and skipped by the Digital Crossbar. To configure P1.2 as an analog pin, clear Bit 2 in register P1MDIN to '0' and set Bit 2 in register P1 to '1'. To configure the Crossbar to skip P1.2, set Bit 2 in register P1SKIP to '1'. Refer to **Section "18. Port Input/Output" on page 147** for complete Port I/O configuration details. The TEMPE bit in register REF0CN enables/disables the temperature sensor. While disabled, the temperature sensor defaults to a high impedance state and any ADC0 measurements performed on the sensor result in meaningless data.

SFR Definition 7.1. REF0CN: Reference Control

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
IDAMRG	GF	ZTCEN	REFLV	REFSL	TEMPE	BIASE	REFBE	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xD1

Bit7: IDAMRG: IDAC Output Merge Select.
0: IDA1 Output is P0.1.
1: IDA1 Output is P0.0 (Merged with IDA0 Output).

Bit6: GF. General Purpose Flag.
This bit is a general purpose flag for use under software control.

Bit5: ZTCEN: Zero-TempCo Bias Enable Bit.
0: ZeroTC Bias Generator automatically enabled when needed.
1: ZeroTC Bias Generator forced on.

Bit4: REFLV: Voltage Reference Output Level Select.
This bit selects the output voltage level for the internal voltage reference.
0: Internal voltage reference set to 1.5 V.
1: Internal voltage reference set to 2.2 V.

Bit3: REFSL: Voltage Reference Select.
This bit selects the source for the internal voltage reference.
0: V_{REF} pin used as voltage reference.
1: V_{DD} used as voltage reference.

Bit2: TEMPE: Temperature Sensor Enable Bit.
0: Internal Temperature Sensor off.
1: Internal Temperature Sensor on.

Bit1: BIASE: Internal Analog Bias Generator Enable Bit.
0: Internal Analog Bias Generator automatically enabled when needed.
1: Internal Analog Bias Generator on.

Bit0: REFBE: Internal Reference Buffer Enable Bit.
0: Internal Reference Buffer disabled.
1: Internal Reference Buffer enabled. Internal voltage reference driven on the V_{REF} pin.

10.1.2. MOVX Instruction and Program Memory

The MOVX instruction is typically used to access data stored in XDATA memory space. In the CIP-51, the MOVX instruction can also be used to write or erase on-chip program memory space implemented as re-programmable Flash memory. The Flash access feature provides a mechanism for the CIP-51 to update program code and use the program memory space for non-volatile data storage. Refer to **Section "16. Flash Memory" on page 135** for further details.

Table 10.1. CIP-51 Instruction Set Summary[1]

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

Notes:
1. Assumes PFEN = 1 for all instruction timing.
2. MOVC instructions take 4 to 7 clock cycles depending on instruction alignment and the FLRT setting (SFR Definition 16.3. FLSCL: Flash Scale).

11.6. Special Function Registers

The direct-access data memory locations from 0x80 to 0xFF constitute the special function registers (SFRs). The SFRs provide control and data exchange with the CIP-51's resources and peripherals. The CIP-51 duplicates the SFRs found in a typical 8051 implementation as well as implementing additional SFRs used to configure and access the sub-systems unique to the MCU. This allows the addition of new functionality while retaining compatibility with the MCS-51™ instruction set. Table 11.1 lists the SFRs implemented in the CIP-51 System Controller.

The SFR registers are accessed anytime the direct addressing mode is used to access memory locations from 0x80 to 0xFF. SFRs with addresses ending in 0x0 or 0x8 (e.g. P0, TCON, IE, etc.) are bit-addressable as well as byte-addressable. All other SFRs are byte-addressable only. Unoccupied addresses in the SFR space are reserved for future use. Accessing these areas will have an indeterminate effect and should be avoided. Refer to the corresponding pages of the data sheet, as indicated in Table 11.2, for a detailed description of each register.

Table 11.1. Special Function Register (SFR) Memory Map

F8	SPI0CN	PCA0L	PCA0H	PCA0CPL0	PCA0CPH0	PCA0CPL4	PCA0CPH4	VDM0CN
F0	B	P0MDIN	P1MDIN	P2MDIN	IDA1L	IDA1H	EIP1	EIP2
E8	ADC0CN	PCA0CPL1	PCA0CPH1	PCA0CPL2	PCA0CPH2	PCA0CPL3	PCA0CPH3	RSTSRC
E0	ACC	XBR0	XBR1	PFE0CN	IT01CF		EIE1	EIE2
D8	PCA0CN	PCA0MD	PCA0CPM0	PCA0CPM1	PCA0CPM2	PCA0CPM3	PCA0CPM4	CRC0FLIP
D0	PSW	REF0CN	PCA0CPL5	PCA0CPH5	P0SKIP	P1SKIP	P2SKIP	P0MAT
C8	TMR2CN	REG0CN	TMR2RLL	TMR2RLH	TMR2L	TMR2H	PCA0CPM5	P1MAT
C0	SMB0CN	SMB0CF	SMB0DAT	ADC0GTL	ADC0GTH	ADC0LTL	ADC0LTH	P0MASK
B8	IP	IDA0CN	ADC0TK	ADC0MX	ADC0CF	ADC0L	ADC0H	P1MASK
B0	P0ODEN	OSCXCN	OSCICN	OSCICL		IDA1CN	FLSCL	FLKEY
A8	IE	CLKSEL	EMI0CN	CLKMUL	RTC0ADR	RTC0DAT	RTC0KEY	ONESHOT
A0	P2	SPI0CFG	SPI0CKR	SPI0DAT	P0MDOUT	P1MDOUT	P2MDOUT	
98	SCON0	SBUF0	CPT1CN	CPT0CN	CPT1MD	CPT0MD	CPT1MX	CPT0MX
90	P1	TMR3CN	TMR3RLL	TMR3RLH	TMR3L	TMR3H	IDA0L	IDA0H
88	TCON	TMOD	TL0	TL1	TH0	TH1	CKCON	PSCTL
80	P0	SP	DPL	DPH	CRC0CN	CRC0IN	CRC0DAT	PCON
	0(8)	1(9)	2(A)	3(B)	4(C)	5(D)	6(E)	7(F)

(bit addressable)

15. 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 $\overline{RST}$ 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. Refer to **Section “19. Oscillators” on page 165** for information on selecting and configuring the system clock source. The Watchdog Timer is enabled with the system clock divided by 12 as its clock source (**Section “25.3. Watchdog Timer Mode” on page 257** details the use of the Watchdog Timer). Program execution begins at location 0x0000.

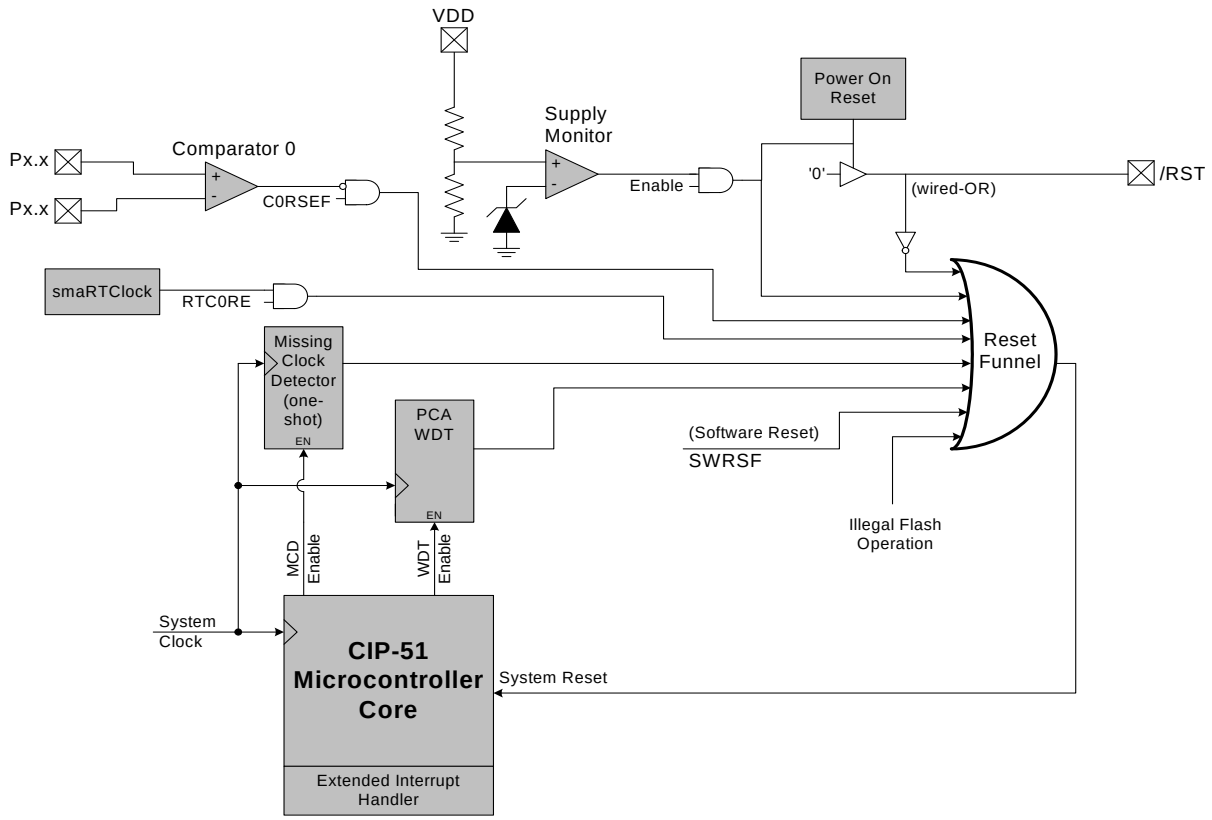

Figure 15.1. Reset Sources

SFR Definition 18.18. P2MDOUT: Port2 Output Mode

R	R	R	R	R	R	R	R/W	Reset Value
								00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xA6

Bits7–0: Output Configuration Bits for P2.7–P2.0 (respectively): ignored if corresponding bit in register P2MDIN is logic 0.
0: Corresponding P2.n Output is open-drain.
1: Corresponding P2.n Output is push-pull.

SFR Definition 18.19. P2SKIP: Port2 Skip

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: 0xD6

Bits7–0: P2SKIP[7:0]: Port2 Crossbar Skip Enable Bits.
These bits select Port pins to be skipped by the Crossbar Decoder. Port pins used as analog inputs (for ADC or Comparator) or used as special functions (V_{REF} input, external oscillator circuit, CNVSTR input) should be skipped by the Crossbar.
0: Corresponding P2.n pin is not skipped by the Crossbar.
1: Corresponding P2.n pin is skipped by the Crossbar.

SFR Definition 20.2. RTC0ADR: smaRTClock Address

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
BUSY	AUTORD	VREGEN	SHORT	RTC0ADDR				Variable
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xAC

Bit 7: BUSY: smaRTClock Interface Busy bit.
Writing a '1' to this bit initiates a smaRTClock indirect read operation. This bit is automatically cleared by hardware when the operation is complete.
0: smaRTClock Interface is not busy.
1: smaRTClock Interface is busy performing a read or write operation.

Bit 6: AUTORD: smaRTClock Interface Auto Read Enable.
0: BUSY must be written manually for each smaRTClock indirect read operation.
1: The next smaRTClock indirect read operation is initiated when RTC0DAT is read by software.

Bit 5: VREGEN: Backup Supply Voltage Regulator Enable.
This bit is automatically set to 1b when $V_{RTC\text{-}BACKUP} > V_{DD}$.
0: Backup Supply Voltage Regulator Disabled (smaRTClock powered from V_{DD}).
1: Force Backup Supply Voltage Regulator Enabled (smaRTClock powered from $V_{RTC\text{-}BACKUP}$).

Bit 4: SHORT: Short Read/Write Timing Enable.
0: smaRTClock reads and writes are 4 system clocks wide.
1: smaRTClock reads and writes are 1 system clock wide.
Note: Increasing the speed of the smaRTClock reads and writes may also slightly increase power consumption.

Bits 3–0: RTC0ADDR: smaRTClock Address Bits
These bits select the smaRTClock internal register that is targeted by reads/writes to RTC0DAT.

RTC0ADDR	smaRTClock Internal Register
0000	CAPTURE0
0001	CAPTURE1
0010	CAPTURE2
0011	CAPTURE3
0100	CAPTURE4
0101	CAPTURE5
0110	RTC0CN
0111	RTC0XCN
1000	ALARM0
1001	ALARM1
1010	ALARM2
1011	ALARM3
1100	ALARM4
1101	ALARM5
1110	RAMADDR
1111	RAMDATA

Note: The RTC0ADDR bits increment after each indirect read/write operation that targets a CAPTUREn or ALARMn internal register.

A typical SMBus transaction consists of a START condition followed by an address byte (Bits7-1: 7-bit slave address; Bit0: R/W direction bit), one or more bytes of data, and a STOP condition. Each byte that is received (by a master or slave) must be acknowledged (ACK) with a low SDA during a high SCL (see Figure 21.3). If the receiving device does not ACK, the transmitting device will read a NACK (not acknowledge), which is a high SDA during a high SCL.

The direction bit (R/W) occupies the least-significant bit position of the address byte. The direction bit is set to logic 1 to indicate a "READ" operation and cleared to logic 0 to indicate a "WRITE" operation.

All transactions are initiated by a master, with one or more addressed slave devices as the target. The master generates the START condition and then transmits the slave address and direction bit. If the transaction is a WRITE operation from the master to the slave, the master transmits the data a byte at a time waiting for an ACK from the slave at the end of each byte. For READ operations, the slave transmits the data waiting for an ACK from the master at the end of each byte. At the end of the data transfer, the master generates a STOP condition to terminate the transaction and free the bus. Figure 21.3 illustrates a typical SMBus transaction.

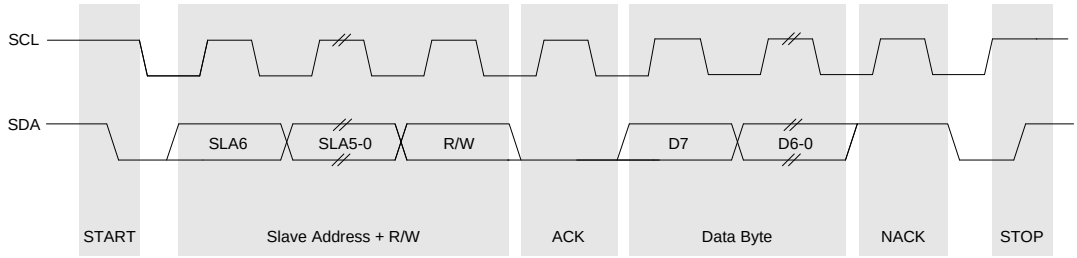

Figure 21.3. SMBus Transaction

21.3.1. Arbitration

A master may start a transfer only if the bus is free. The bus is free after a STOP condition or after the SCL and SDA lines remain high for a specified time (see **Section “21.3.4. SCL High (SMBus Free) Timeout” on page 194**). In the event that two or more devices attempt to begin a transfer at the same time, an arbitration scheme is employed to force one master to give up the bus. The master devices continue transmitting until one attempts a HIGH while the other transmits a LOW. Since the bus is open-drain, the bus will be pulled LOW. The master attempting the HIGH will detect a LOW SDA and lose the arbitration. The winning master continues its transmission without interruption; the losing master becomes a slave and receives the rest of the transfer if addressed. This arbitration scheme is non-destructive: one device always wins, and no data is lost.

21.3.2. Clock Low Extension

SMBus provides a clock synchronization mechanism, similar to I2C, which allows devices with different speed capabilities to coexist on the bus. A clock-low extension is used during a transfer in order to allow slower slave devices to communicate with faster masters. The slave may temporarily hold the SCL line LOW to extend the clock low period, effectively decreasing the serial clock frequency.

23.1. Signal Descriptions

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

23.1.1. 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.

23.1.2. 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.

23.1.3. 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.

23.1.4. 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.
2. 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 23.2, Figure 23.3, and Figure 23.4 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 "**18. Port Input/Output**" on page **147** for general purpose port I/O and crossbar information.

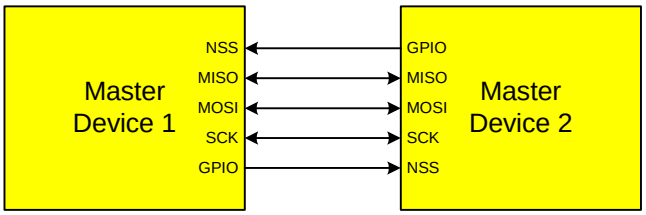

Figure 23.2. Multiple-Master Mode Connection Diagram

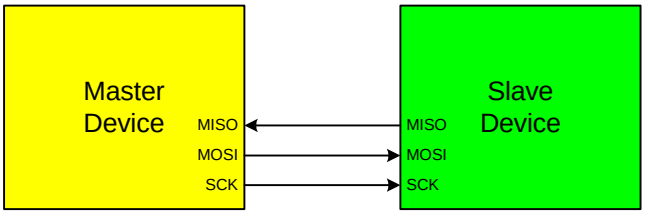

Figure 23.3. 3-Wire Single Master and Slave Mode Connection Diagram

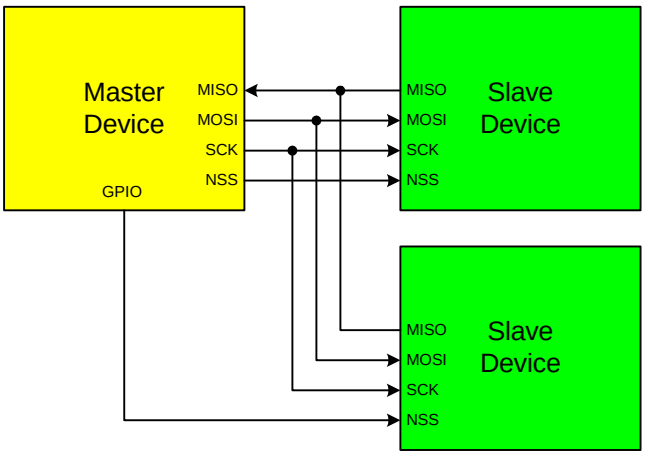

Figure 23.4. 4-Wire Single Master and Slave Mode Connection Diagram

23.3. SPI0 Slave Mode Operation

When SPI0 is enabled and not configured as a master, it will operate as a SPI slave. As a slave, bytes are shifted in through the MOSI pin and out through the MISO pin by a master device controlling the SCK signal. A bit counter in the SPI0 logic counts SCK edges. When 8 bits have been shifted into the shift register, the SPIF flag is set to logic 1, and the byte is copied into the receive buffer. Data is read from the receive buffer by reading SPI0DAT. A slave device cannot initiate transfers. Data to be transferred to the master device is pre-loaded into the shift register by writing to SPI0DAT. Writes to SPI0DAT are double-buffered, and are placed in the transmit buffer first. If the shift register is empty, the contents of the transmit buffer will immediately be transferred into the shift register. When the shift register already contains data, the SPI will load the shift register with the transmit buffer's contents after the last SCK edge of the next (or current) SPI transfer.

NOTES:

24.2.3. External/smaRTClock Capture Mode

Capture Mode allows either the external oscillator or the smaRTClock clock to be measured against the system clock. The external oscillator and smaRTClock clock can also be compared against each other. Timer 2 can be clocked from the system clock, the system clock divided by 12, the external oscillator divided by 8, or the smaRTClock clock divided by 8, depending on the T2ML (CKCON.4), T2XCLK, and T2RCLK settings. The timer will capture either every 8 external clock cycles or every 8 smaRTClock clock cycles, depending on the T2RCLK setting. When a capture event is generated, the contents of Timer 2 (TMR2H:TMR2L) are loaded into the Timer 2 reload registers (TMR2RLH:TMR2RLL) and the TF2H flag is set. By recording the difference between two successive timer capture values, the external oscillator or smaRTClock clock can be determined with respect to the Timer 2 clock. The Timer 2 clock should be much faster than the capture clock to achieve an accurate reading. Timer 2 should be in 16-bit auto-reload mode when using Capture Mode.

For example, if T2ML = 1b, T2RCLK = 0b, and TF2CEN = 1b, Timer 2 will clock every SYSCLK and capture every smaRTClock clock divided by 8. If the SYSCLK is 24.5 MHz and the difference between two successive captures is 5984, then the smaRTClock clock is:

24.5 MHz / (5984 / 8) = 0.032754 MHz or 32.754 kHz.

This mode allows software to determine the exact smaRTClock frequency in self-oscillate mode and the external oscillator frequency when an RC network or capacitor is used to generate the signal.

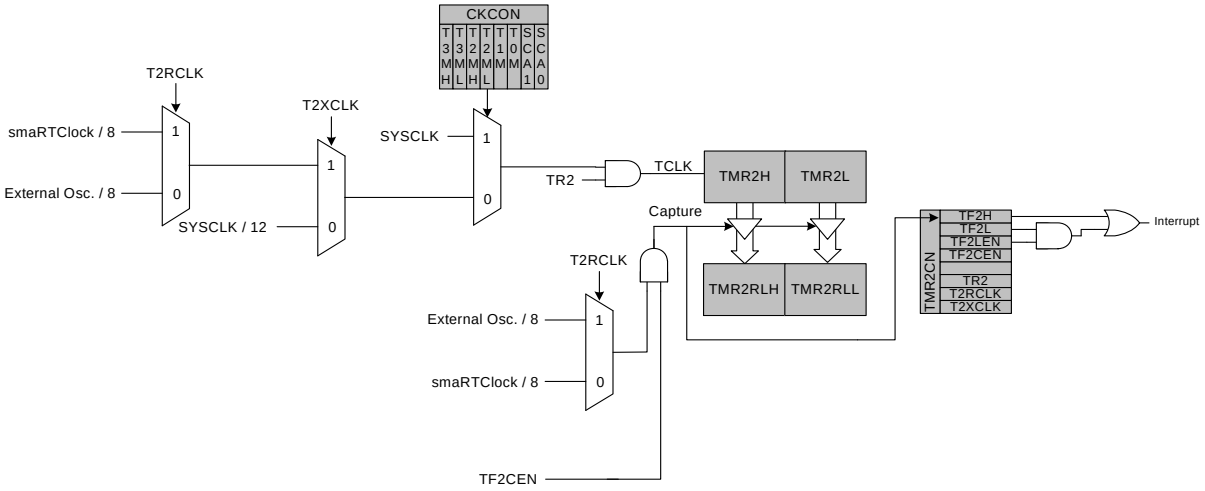

Figure 24.6. Timer 2 Capture Mode Block Diagram

SFR Definition 25.2. PCA0MD: PCA Mode

R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	Reset Value
CIDL	WDTE	WDLCK	-	CPS2	CPS1	CPS0	ECF	01000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xD9

Bit7: CIDL: PCA Counter/Timer Idle Control.
Specifies PCA behavior when CPU is in Idle Mode.
0: PCA continues to function normally while the system controller is in Idle Mode.
1: PCA operation is suspended while the system controller is in Idle Mode.

Bit6: WDTE: Watchdog Timer Enable
If this bit is set, PCA Module 5 is used as the watchdog timer.
0: Watchdog Timer disabled.
1: PCA Module 5 enabled as Watchdog Timer.

Bit5: WDLCK: Watchdog Timer Lock
This bit locks/unlocks the Watchdog Timer Enable. When WDLCK is set, the Watchdog Timer may not be disabled until the next system reset.
0: Watchdog Timer Enable unlocked.
1: Watchdog Timer Enable locked.

Bit4: UNUSED. Read = 0b, Write = don't care.

Bits3–1: CPS2–CPS0: PCA Counter/Timer Pulse Select.
These bits select the timebase source for the PCA counter.

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 clock divided by 8*
1	1	0	smaRTClock clock divided by 8*
1	1	1	Reserved

***Note:** External clock divided by 8 and smaRTClock clock divided by 8 are synchronized with the system clock.

Bit0: ECF: PCA Counter/Timer Overflow Interrupt Enable.
This bit sets the masking of the PCA Counter/Timer Overflow (CF) interrupt.
0: Disable the CF interrupt.
1: Enable a PCA Counter/Timer Overflow interrupt request when CF (PCA0CN.7) is set.

Note: When the WDTE bit is set to '1', the PCA0MD register cannot be modified. To change the contents of the PCA0MD register, the Watchdog Timer must first be disabled.