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	25MHz
Connectivity	LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, 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	256 x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.25V
Data Converters	A/D 16x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	20-TSSOP (0.173", 4.40mm Width)
Supplier Device Package	20-TSSOP
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f530-c-it

1.6. Programmable Comparator

C8051F52x/F52xA/F53x/F53xA devices include a software-configurable voltage comparator with an input multiplexer. The comparator offers programmable response time and hysteresis and an output that is optionally available at the Port pins: a synchronous “latched” output (CP0). The comparator interrupt 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. The Comparator may also be configured as a reset source. A block diagram of the comparator is shown in Figure 1.8.

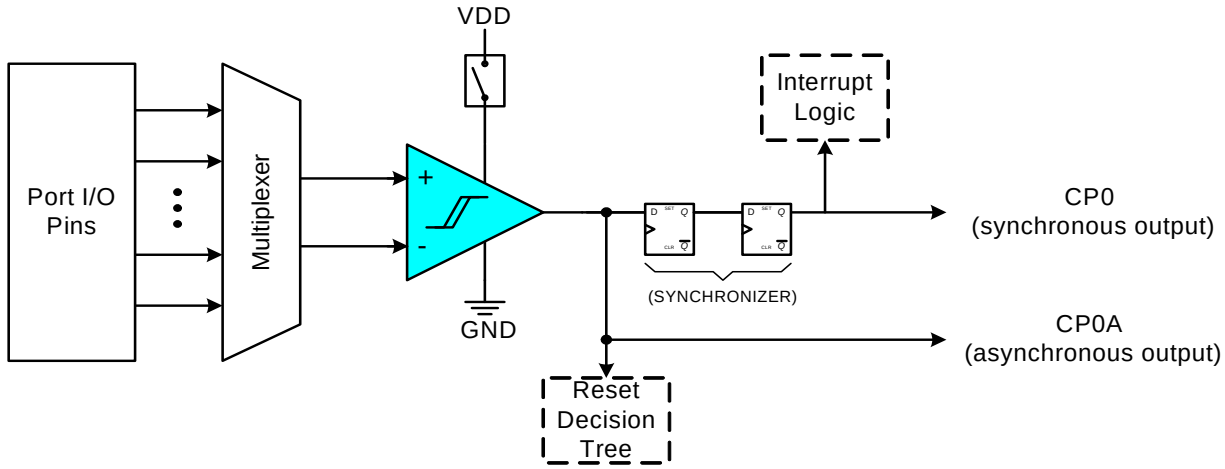

Figure 1.8. Comparator Block Diagram

1.7. Voltage Regulator

C8051F52x/F52xA/F53x/F53xA devices include an on-chip low dropout voltage regulator (REG0). The input to REG0 at the V_{REGIN} pin can be as high as 5.25 V. The output can be selected by software to 2.1 or 2.6 V. When enabled, the output of REG0 powers the device and drives the V_{DD} pin. The voltage regulator can be used to power external devices connected to V_{DD}.

1.8. Serial Port

The C8051F52x/F52xA/F53x/F53xA family includes 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.

Table 3.7. Pin Definitions for the C8051F53x and C805153xA (QFN 20) (Continued)

Name	Pin Numbers		Type	Description
	'F53xA 'F53x-C	'F53x		
P1.0/	13	13	D I/O or A In	Port 1.0. See Port I/O Section for a complete description.
XTAL2			D I/O	External Clock Output. For an external crystal or resonator, this pin is the excitation driver. This pin is the external clock input for CMOS, capacitor, or RC oscillator configurations. Section "14. Oscillators" on page 135.
P0.7/	14	14	D I/O or	Port 0.7. See Port I/O Section for a complete description.
XTAL1			A In	External Clock Input. This pin is the external oscillator return for a crystal or resonator. See Oscillator Section.
P0.6/	15	15	D I/O or A In	Port 0.6. See Port I/O Section for a complete description.
C2D			D I/O	Bi-directional data signal for the C2 Debug Interface.
P0.5/RX*	16	—	D I/O or A In	Port 0.5. See Port I/O Section for a complete description.
P0.5	—	16	D I/O or A In	Port 0.5. See Port I/O Section for a complete description.
P0.4/TX*	17	—	D I/O or A In	Port 0.4. See Port I/O Section for a complete description.
P0.4/RX*	—	17	D I/O or A In	Port 0.4. See Port I/O Section for a complete description.
P0.3	18	—	D I/O or A In	Port 0.3. See Port I/O Section for a complete description.
P0.3/TX*	—	18	D I/O or A In	Port 0.3. See Port I/O Section for a complete description.
P0.2	19	19	D I/O or A In	Port 0.2. See Port I/O Section for a complete description.
P0.1	20	20	D I/O or A In	Port 0.1. See Port I/O Section for a complete description.
Note: Please refer to Section "20. Device Specific Behavior" on page 210.				

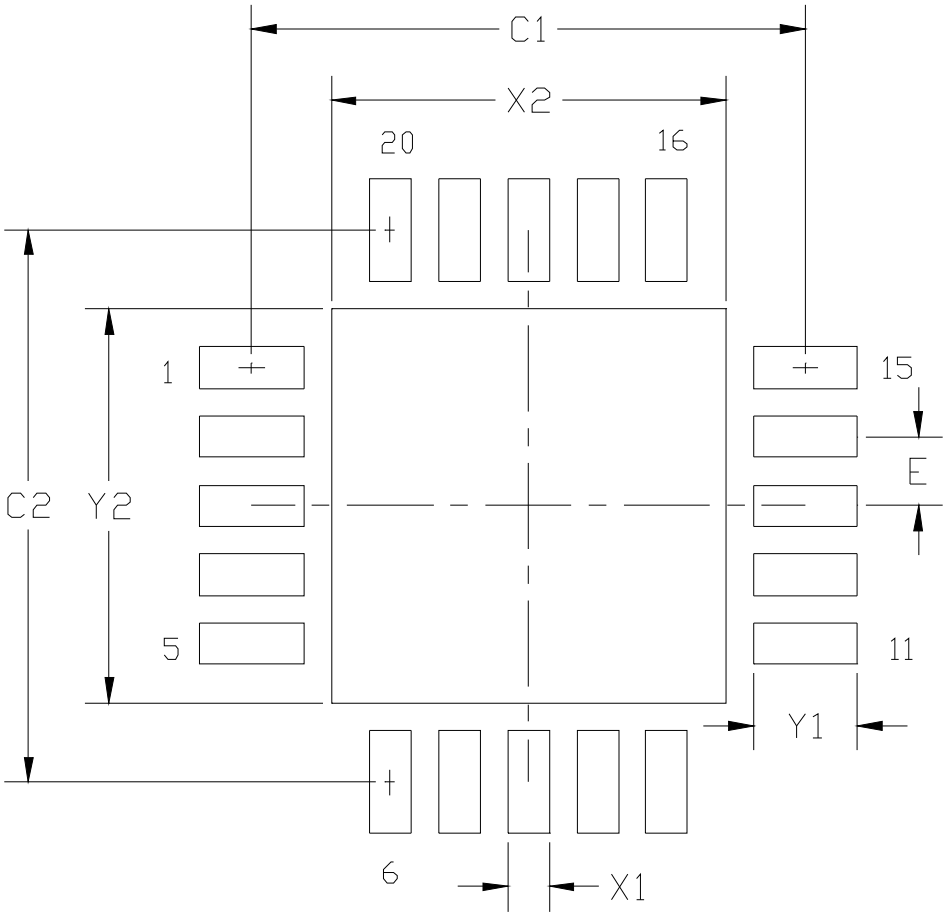

Figure 3.9. QFN-20 Landing Diagram*

Note: The Landing Dimensions are given in Table 3.9, “QFN-20 Landing Diagram Dimensions,” on page 51.

SFR Definition 4.6. ADC0H: ADC0 Data Word MSB

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:
								0xBE

Bits7–0: ADC0 Data Word High-Order Bits.
For AD0LJST = 0 and AD0RPT as follows:
00: Bits 3–0 are the upper 4 bits of the 12-bit result. Bits 7–4 are 0000b.
01: Bits 4–0 are the upper 5 bits of the 14-bit result. Bits 7–5 are 000b.
10: Bits 5–0 are the upper 6 bits of the 15-bit result. Bits 7–6 are 00b.
11: Bits 7–0 are the upper 8 bits of the 16-bit result.
For AD0LJST = 1 (AD0RPT must be '00'): Bits 7–0 are the most-significant bits of the ADC0 12-bit result.

SFR Definition 4.7. ADC0L: ADC0 Data Word LSB

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:
								0xBD

Bits7–0: ADC0 Data Word Low-Order Bits.
For AD0LJST = 0: Bits 7–0 are the lower 8 bits of the ADC0 Accumulated Result.
For AD0LJST = 1 (AD0RPT must be '00'): Bits 7–4 are the lower 4 bits of the 12-bit result. Bits 3–0 are 0000b.

SFR Definition 6.1. REG0CN: Regulator Control

R/W	R/W	R	R/W	R	R	R	R	Reset Value
REGDIS	Reserved	—	REG0MD	—	—	—	DROPOUT	01010000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xC9

Bit7: **REGDIS**: Voltage Regulator Disable Bit.
This bit disables/enables the Voltage Regulator.
0: Voltage Regulator Enabled.
1: Voltage Regulator Disabled.
Bit6: **RESERVED**. Read = 1b. Must write 1b.
Bit5: **UNUSED**. Read = 0b. Write = don't care.
Bit4: **REG0MD**: Voltage Regulator Mode Select Bit.
This bit selects the Voltage Regulator output voltage.
0: Voltage Regulator output is 2.1 V.
1: Voltage Regulator output is 2.6 V (default).
Bits3–1: **UNUSED**. Read = 000b. Write = don't care.
Bit0: **DROPOUT**: Voltage Regulator Dropout Indicator Bit.
0: Voltage Regulator is not in dropout.
1: Voltage Regulator is in or near dropout.

SFR Definition 7.2. CPT0MX: Comparator0 MUX Selection

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
CMX0N3	CMX0N2	CMX0N1	CMX0N0	CMX0P3	CMX0P2	CMX0P1	CMX0P0	01110111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0x9F

Bits7–4: **CMX0N3–CMX0N0**: Comparator0 Negative Input MUX Select.
These bits select which Port pin is used as the Comparator0 negative input.

CMX0N3	CMX0N2	CMX0N1	CMX0N0	Negative Input
0	0	0	0	P0.1
0	0	0	1	P0.3
0	0	1	0	P0.5
0	0	1	1	P0.7*
0	1	0	0	P1.1*
0	1	0	1	P1.3*
0	1	1	0	P1.5*
0	1	1	1	P1.7*

***Note:** Available only on the C8051F53x/53xA devices

Bits1–0: **CMX0P3–CMX0P0**: Comparator0 Positive Input MUX Select.
These bits select which Port pin is used as the Comparator0 positive input.

CMX0P3	CMX0P2	CMX0P1	CMX0P0	Positive Input
0	0	0	0	P0.0
0	0	0	1	P0.2
0	0	1	0	P0.4
0	0	1	1	P0.6*
0	1	0	0	P1.0*
0	1	0	1	P1.2*
0	1	1	0	P1.4*
0	1	1	1	P1.6*

***Note:** Available only on the C8051F53x/53xA devices.

8. 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 C8051F52x/F52xA/F53x/F53xA family has a superset of all the peripherals included with a standard 8051. See Section “1. System Overview” on page 13 for more information about the available peripherals. The CIP-51 includes on-chip debug hardware which 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 8.1 for a block diagram). The CIP-51 core includes the following features:

- Fully Compatible with MCS-51 Instruction Set
- 25 MIPS Peak Throughput
- 256 Bytes of Internal RAM
- Extended Interrupt Handler
- Reset Input
- Power Management Modes
- Integrated Debug Logic
- Program and Data Memory Security

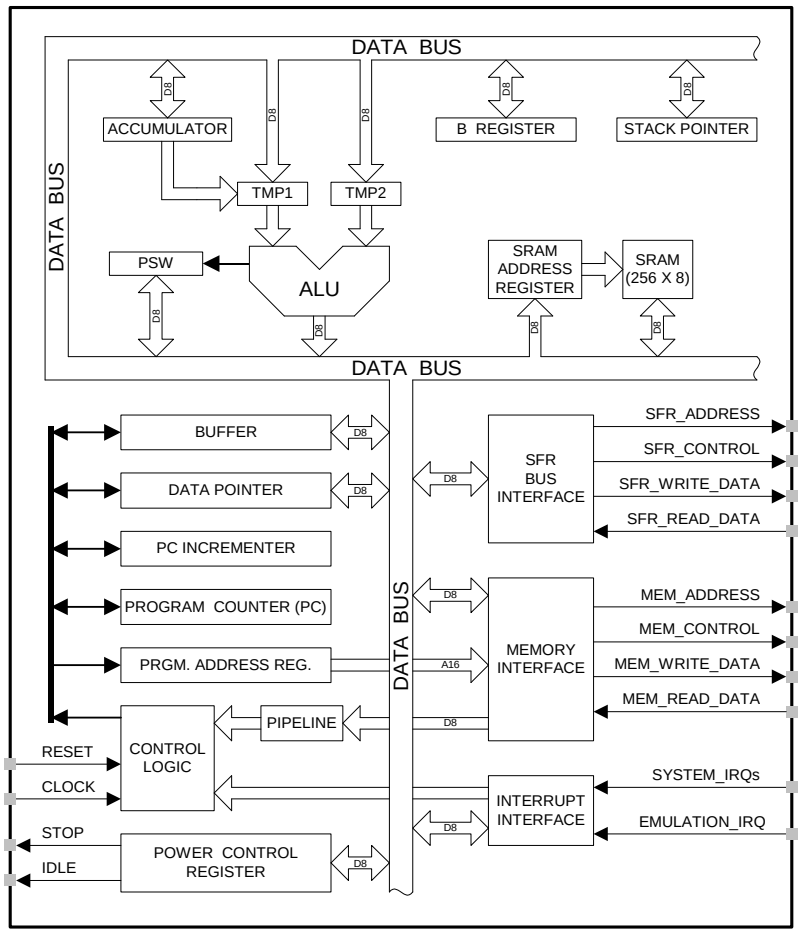

Figure 8.1. CIP-51 Block Diagram

Table 9.2. Special Function Registers (Continued)

SFRs are listed in alphabetical order. All undefined SFR locations are reserved

Register	Address	Description	Page
REF0CN	0xD1	Voltage Reference Control	73
REG0CN	0xC9	Voltage Regulator Control	75
RSTSRC	0xEF	Reset Source Configuration/Status	112
SBUF0	0x99	UART0 Data Buffer	150
SCON0	0x98	UART0 Control	149
SP	0x81	Stack Pointer	87
SPI0CFG	0xA1	SPI Configuration	157
SPI0CKR	0xA2	SPI Clock Rate Control	159
SPI0CN	0xF8	SPI Control	158
SPI0DAT	0xA3	SPI Data	160
TCON	0x88	Timer/Counter Control	186
TH0	0x8C	Timer/Counter 0 High	189
TH1	0x8D	Timer/Counter 1 High	189
TL0	0x8A	Timer/Counter 0 Low	189
TL1	0x8B	Timer/Counter 1 Low	189
TMOD	0x89	Timer/Counter Mode	187
TMR2CN	0xC8	Timer/Counter 2 Control	193
TMR2H	0xCD	Timer/Counter 2 High	194
TMR2L	0xCC	Timer/Counter 2 Low	194
TMR2RLH	0xCB	Timer/Counter 2 Reload High	194
TMR2RLL	0xCA	Timer/Counter 2 Reload Low	194
VDDMON	0xFF	V_{DD} Monitor Control	109
XBR0	0xE1	Port I/O Crossbar Control 0	127
XBR1	0xE2	Port I/O Crossbar Control 1	128

SFR Definition 10.3. EIE1: Extended Interrupt Enable 1

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
EMAT	EREG0	ELIN	ECPR	ECPF	EPCA0	EADC0	EWADC0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xE6

Bit7: **EMAT**: Enable Port Match Interrupt.
This bit sets the masking of the Port Match interrupt.
0: Disable the Port Match interrupt.
1: Enable the Port Match interrupt.

Bit6: **EREG0**: Enable Voltage Regulator Interrupt.
This bit sets the masking of the Voltage Regulator Dropout interrupt.
0: Disable the Voltage Regulator Dropout interrupt.
1: Enable the Voltage Regulator Dropout interrupt.

Bit5: **ELIN**: Enable LIN Interrupt.
This bit sets the masking of the LIN interrupt.
0: Disable LIN interrupts.
1: Enable LIN interrupt requests.

Bit4: **ECPR**: Enable Comparator 0 Rising Edge Interrupt
This bit sets the masking of the CP0 Rising Edge interrupt.
0: Disable CP0 Rising Edge Interrupt.
1: Enable CP0 Rising Edge Interrupt.

Bit3: **ECPF**: Enable Comparator 0 Falling Edge Interrupt
This bit sets the masking of the CP0 Falling Edge interrupt.
0: Disable CP0 Falling Edge Interrupt.
1: Enable CP0 Falling Edge Interrupt.

Bit2: **EPCA0**: Enable Programmable Counter Array (PCA0) Interrupt.
This bit sets the masking of the PCA0 interrupts.
0: Disable all PCA0 interrupts.
1: Enable interrupt requests generated by PCA0.

Bit1: **EADC0**: Enable ADC0 Conversion Complete Interrupt.
This bit sets the masking of the ADC0 Conversion Complete interrupt.
0: Disable ADC0 Conversion Complete interrupt.
1: Enable interrupt requests generated by the AD0INT flag.

Bit0: **EWADC0**: Enable ADC0 Window Comparison Interrupt.
This bit sets the masking of the ADC0 Window Comparison interrupt.
0: Disable ADC0 Window Comparison interrupt.
1: Enable interrupt requests generated by the AD0WINT flag.

11. 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 “14. Oscillators” on page 135 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 “19.3. Watchdog Timer Mode” on page 203 details the use of the Watchdog Timer). Program execution begins at location 0x0000.

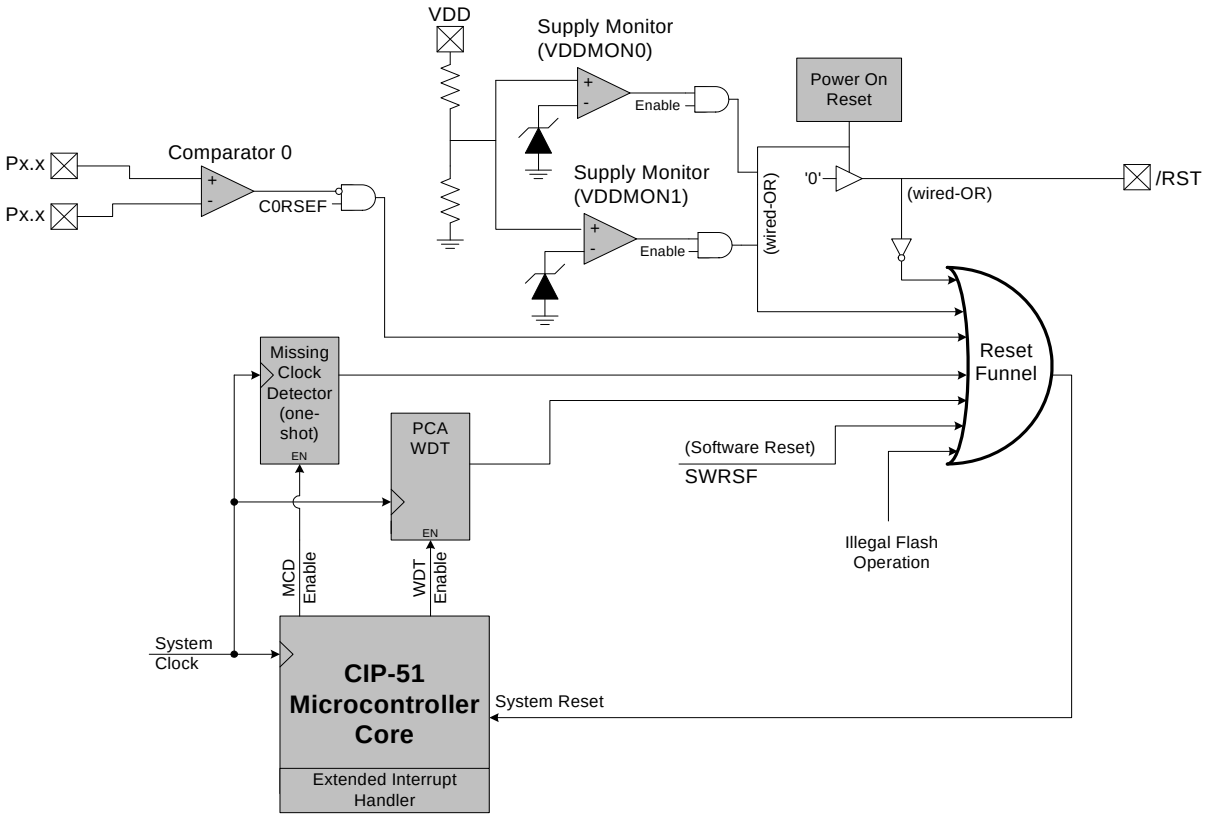

Figure 11.1. Reset Sources

11.2. Power-Fail Reset / V_{DD} Monitors (VDDMON0 and VDDMON1)

C8051F52x-C/F53x-C devices include two V_{DD} monitors: a standard V_{DD} monitor (VDDMON0) and a level-sensitive V_{DD} monitor (VDDMON1). VDDMON0 is primarily intended for setting a higher threshold to allow safe erase or write of Flash memory from firmware. VDDMON1 is used to hold the device in a reset state during power-up and brownout conditions.

Note: VDDMON1 is not present in older silicon revisions A and B. Please refer to Section “20.4. VDD Monitors and VDD Ramp Time” on page 211 for more details.

When a power-down transition or power irregularity causes V_{DD} to drop below V_{RST}, the power supply monitors (VDDMON0 and VDDMON1) will drive the $\overline{\text{RST}}$ pin low and hold the CIP-51 in a reset state (see Figure 11.2). When V_{DD} returns to a level above V_{RST}, the CIP-51 will be released from the reset state. Note that even though internal data memory contents are not altered by the power-fail reset, it is impossible to determine if V_{DD} dropped below the level required for data retention. If the PORSF flag reads 1, the data may no longer be valid.

VDDMON0 is enabled and is selected as a reset source after power-on resets; however its defined state (enabled/disabled) is not altered by any other reset source. For example, if VDDMON0 is disabled by software, and a software reset is performed, VDDMON0 will still be disabled after that reset.

VDDMON1 is enabled and is selected as a reset source after power-on reset and any other type of reset. There is no register setting that can disable this level-sensitive VDD monitor as a reset source.

To protect the integrity of Flash contents, the V_{DD} monitor (VDDMON0) must be enabled to the higher setting (VDMLVL = '1') and selected as a reset source if software contains routines which erase or write Flash memory. If the V_{DD} monitor is not enabled and set to the higher setting, any erase or write performed on Flash memory will cause a Flash Error device reset.

Note: Please refer to Section “20.5. VDD Monitor (VDDMON0) High Threshold Setting” on page 212 for important notes related to the VDD Monitor high threshold setting in older silicon revisions A and B.

The V_{DD} monitor (VDDMON0) must be enabled before it is selected as a reset source. Selecting the VDDMON0 as a reset source before it is enabled and stabilized may cause a system reset. The procedure for re-enabling the V_{DD} monitor and configuring the V_{DD} monitor as a reset source is shown below:

1. Enable the V_{DD} monitor (VDMEN bit in VDDMON = 1).
2. Wait for the V_{DD} monitor to stabilize (see Table 2.8 on page 32 for the V_{DD} Monitor turn-on time). **Note: This delay should be omitted if software contains routines which write or erase Flash memory.**
3. Select the V_{DD} monitor as a reset source (PORSF bit in RSTSRC = 1).

See Figure 11.2 for V_{DD} monitor timing; note that the reset delay is not incurred after a V_{DD} monitor reset. See Table 2.8 on page 32 for complete electrical characteristics of the V_{DD} monitor.

Note: Software should take care not to inadvertently disable the V_{DD} Monitor (VDDMON0) as a reset source when writing to RSTSRC to enable other reset sources or to trigger a software reset. All writes to RSTSRC should explicitly set PORSF to '1' to keep the V_{DD} Monitor enabled as a reset source.

11.2.1. VDD Monitor Thresholds and Minimum VDD

The minimum operating digital supply voltage (V_{DD}) is specified as 2.0 V in Table 2.2 on page 26. The voltage at which the MCU is released from reset (V_{RST}) can be as low as 1.65 V based on the V_{DD} Monitor thresholds that are specified in Table 2.8 on page 32. This could allow code execution during the power-up

SFR Definition 12.1. PSCTL: Program Store R/W Control

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

SFR Address: 0x8F

Bits7–2: **UNUSED**: Read = 000000b, Write = don't care.

Bit1: **PSEE**: Program Store Erase Enable
Setting this bit (in combination with PSWE) allows an entire page of Flash program memory to be erased. If this bit is logic 1 and Flash writes are enabled (PSWE is logic 1), 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.
0: Flash program memory erasure disabled.
1: Flash program memory erasure enabled.

Bit0: **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 Flash location should be erased before writing data.
0: Writes to Flash program memory disabled.
1: Writes to Flash program memory enabled; the MOVX write instruction targets Flash memory.

Note: See Section "12.1. Programming The Flash Memory" on page 113 for minimum V_{DD} and temperature requirements for flash erase and write operations.

SFR Definition 12.2. FLKEY: Flash Lock and Key

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: 0xB7

Bits7–0: **FLKEY**: Flash Lock and Key Register
Write:
This register provides a lock and key function for Flash erasures and writes. Flash writes and erases are enabled by writing 0xA5 followed by 0xF1 to the FLKEY register. Flash writes and erases are automatically disabled after the next write or erase is complete. If any writes to FLKEY are performed incorrectly, or if a Flash write or erase operation is attempted while these operations are disabled, the Flash will be permanently locked from writes or erasures until the next device reset. If an application never writes to Flash, it can intentionally lock the Flash by writing a non-0xA5 value to FLKEY from software.
Read:
When read, bits 1–0 indicate the current Flash lock state.
00: Flash is write/erase locked.
01: The first key code has been written (0xA5).
10: Flash is unlocked (writes/erases allowed).
11: Flash writes/erases disabled until the next reset.

The shift register contents are locked after the slave detects the first edge of SCK. Writes to SPI0DAT that occur after the first SCK edge will be held in the TX latch until the end of the current transfer.

When configured as a slave, SPI0 can be configured for 4-wire or 3-wire operation. The default, 4-wire slave mode, is active when NSSMD1 (SPI0CN.3) = 0 and NSSMD0 (SPI0CN.2) = 1. In 4-wire mode, the NSS signal is routed to a port pin and configured as a digital input. SPI0 is enabled when NSS is logic 0, and disabled when NSS is logic 1. The bit counter is reset on a falling edge of NSS. Note that the NSS signal must be driven low at least 2 system clocks before the first active edge of SCK for each byte transfer. Figure 16.4 shows a connection diagram between two slave devices in 4-wire slave mode and a master device.

3-wire slave mode is active when NSSMD1 (SPI0CN.3) = 0 and NSSMD0 (SPI0CN.2) = 0. NSS is not used in this mode, and is not mapped to an external port pin through the crossbar. Since there is not a way of uniquely addressing the device in 3-wire slave mode, SPI0 must be the only slave device present on the bus. It is important to note that in 3-wire slave mode there is no external means of resetting the bit counter that determines when a full byte has been received. The bit counter can only be reset by disabling and re-enabling SPI0 with the SPIEN bit. Figure 16.3 shows a connection diagram between a slave device in 3-wire slave mode and a master device.

16.4. SPI0 Interrupt Sources

When SPI0 interrupts are enabled, the following four flags will generate an interrupt when they are set to logic 1:

Note that all of the following interrupt bits must be cleared by software.

1. The SPI Interrupt Flag, SPIF (SPI0CN.7) is set to logic 1 at the end of each byte transfer. This flag can occur in all SPI0 modes.
2. The Write Collision Flag, WCOL (SPI0CN.6) is set to logic 1 if a write to SPI0DAT is attempted when the transmit buffer has not been emptied to the SPI shift register. When this occurs, the write to SPI0DAT will be ignored, and the transmit buffer will not be written.This flag can occur in all SPI0 modes.
3. The Mode Fault Flag MODF (SPI0CN.5) is set to logic 1 when SPI0 is configured as a master in multi-master mode and the NSS pin is pulled low. When a Mode Fault occurs, the MSTEN and SPIEN bits in SPI0CN are set to logic 0 to disable SPI0 and allow another master device to access the bus.
4. The Receive Overrun Flag RXOVRN (SPI0CN.4) is set to logic 1 when configured as a slave, and a transfer is completed while the receive buffer still holds an unread byte from a previous transfer. The new byte is not transferred to the receive buffer, allowing the previously received data byte to be read. The data byte which caused the overrun is lost.

SFR Definition 16.3. SPI0CKR: SPI0 Clock Rate

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
SCR7	SCR6	SCR5	SCR4	SCR3	SCR2	SCR1	SCR0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

SFR Address: 0xA2

Bits7–0: **SCR7–SCR0**: SPI0 Clock Rate.
These bits determine the frequency of the SCK output when the SPI0 module is configured for master mode operation. The SCK clock frequency is a divided version of the system clock, and is given in the following equation, where *SYSCLK* is the system clock frequency and *SPI0CKR* is the 8-bit value held in the SPI0CKR register.

$$f_{SCK} = \frac{SYSCLK}{2 \times (SPI0CKR + 1)}$$

for 0 <= SPI0CKR <= 255

Example: If SYSCLK = 2 MHz and SPI0CKR = 0x04,

$$f_{SCK} = \frac{2000000}{2 \times (4 + 1)}$$

$$f_{SCK} = 200kHz$$

17.3. LIN Master Mode Operation

The master node is responsible for the scheduling of messages and sends the header of each frame, containing the SYNCH BREAK FIELD, SYNCH FIELD and IDENTIFIER FIELD. The steps to schedule a message transmission or reception are listed below.

1. Load the 6-bit Identifier into the LIN0ID register.
2. Load the data length into the LIN0SIZE register. Set the value to the number of data bytes or "1111b" if the data length should be decoded from the identifier. Also, set the checksum type, classic or enhanced, in the same LIN0SIZE register.
3. Set the data direction by setting the TXRX bit (LIN0CTRL.5). Set the bit to 1 to perform a master transmit operation, or set the bit to 0 to perform a master receive operation.
4. If performing a master transmit operation, load the data bytes to transmit into the data buffer (LIN0DT1 to LIN0DT8).
5. Set the STREQ bit (LIN0CTRL.0) to start the message transfer. The LIN peripheral will schedule the message frame and request an interrupt if the message transfer is successfully completed or if an error has occurred.

This code segment shows the procedure to schedule a message in a transmission operation:

```
LINADDR   = 0x08;// Point to LIN0CTRL
LINDATA  |=  0x20;// Select to transmit data
LINADDR   = 0x0E;// Point to LIN0ID
LINDATA   = 0x11;// Load the ID, in this example 0x11
LINADDR   = 0x0B;// Point to LIN0SIZE
LINDATA   = ( LINDATA & 0xF0 ) | 0x08; // Load the size with 8

LINADDR = 0x00;// Point to Data buffer first byte
for (i=0; i<8; i++)
{
   LINDATA = i + 0x41;// Load the buffer with ‘A’, ‘B’, ...
   LINADDR++;// Increment the address to the next buffer
}
LINADDR   = 0x08;// Point to LIN0CTRL
LINDATA   = 0x01;// Start Request
```

The application should perform the following steps when an interrupt is requested.

1. Check the DONE bit (LIN0ST.0) and the ERROR bit (LIN0ST.2).
2. If performing a master receive operation and the transfer was successful, read the received data from the data buffer.
3. If the transfer was not successful, check the error register to determine the kind of error. Further error handling has to be done by the application.
4. Set the RSTINT (LIN0CTRL.3) and RSTERR bits (LIN0CTRL.2) to reset the interrupt request and the error flags.

SFR Definition 17.3. LINCF Control Mode Register

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

SFR Address: 0x95

Bit7: **LINEN**: LIN Interface Enable bit
0: LIN0 is disabled.
1: LIN0 is enabled.

Bit6: **MODE**: LIN Mode Selection
0: LIN0 operates in Slave mode.
1: LIN0 operates in Master mode.

Bit5: **ABAUD**: LIN Mode Automatic Baud Rate Selection (**slave mode only**).
0: Manual baud rate selection is enabled.
1: Automatic baud rate selection is enabled.

SFR Definition 17.13. LIN0ST: LIN0 STATUS Register

R	R	R	R	R/W	R	R	R	Reset Value
ACTIVE	IDLTOUT	ABORT	DTREQ	LININT	ERROR	WAKEUP	DONE	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

Address: 0x09 (indirect)

Bit7: **ACTIVE**: LIN Bus Activity Bit.
0: No transmission activity detected on the LIN bus.
1: Transmission activity detected on the LIN bus.

Bit6: **IDLTOUT**: Bus Idle Timeout Bit (**slave mode only**).
0: The bus has not been idle for four seconds.
1: No bus activity has been detected for four seconds, but the bus is not yet in Sleep mode.

Bit5: **ABORT**: Aborted transmission signal (**slave mode only**).
0: The current transmission has not been interrupted or stopped. This bit is reset to 0 after receiving a SYNCH BREAK that does not interrupt a pending transmission.
1: New SYNCH BREAK detected before the end of the last transmission or the STOP bit (LIN0CTRL.7) has been set.

Bit4: **DTREQ**: Data Request bit (**slave mode only**).
0: Data identifier has not been received.
1: Data identifier has been received.

Bit3: **LININT**: Interrupt Request bit.
0: An interrupt is not pending. This bit is cleared by setting RSTINT (LIN0CTRL.3)
1: There is a pending LIN0 interrupt.

Bit2: **ERROR**: Communication Error Bit.
0: No error has been detected. This bit is cleared by setting RSTERR (LIN0CTRL.2)
1: An error has been detected.

Bit1: **WAKEUP**: Wakeup Bit.
0: A wakeup signal is not being transmitted and has not been received.
1: A wakeup signal is being transmitted or has been received.

Bit0: **DONE**: Transmission Complete Bit.
0: A transmission is not in progress or has not been started. This bit is cleared at the start of a transmission.
1: The current transmission is complete.

18. Timers

Each MCU includes three counter/timers: two are 16-bit counter/timers compatible with those found in the standard 8051, and one is a 16-bit auto-reload timer for use with other device peripherals or for general purpose use. These timers can be used to measure time intervals, count external events and generate periodic interrupt requests. Timer 0 and Timer 1 are nearly identical and have four primary modes of operation. Timer 2 offer 16-bit and split 8-bit timer functionality with auto-reload.

Timer 0 and Timer 1 Modes	Timer 2 Modes
13-bit counter/timer	16-bit timer with auto-reload
16-bit counter/timer	
8-bit counter/timer with auto-reload	Two 8-bit timers with auto-reload
Two 8-bit counter/timers (Timer 0 only)	

Timers 0 and 1 may be clocked by one of five sources, determined by the Timer Mode Select bits (T1M–T0M) and the Clock Scale bits (SCA1–SCA0). The Clock Scale bits define a pre-scaled clock from which Timer 0 and/or Timer 1 may be clocked (See SFR Definition 18.3 for pre-scaled clock selection).

Timer 0/1 may then be configured to use this pre-scaled clock signal or the system clock. Timer 2 may be clocked by the system clock, the system clock divided by 12, or the external oscillator clock source divided by 8.

Timer 0 and Timer 1 may also be operated as counters. When functioning as a counter, a counter/timer register is incremented on each high-to-low transition at the selected input pin (T0 or T1). Events with a frequency of up to one-fourth the system clock's frequency can be counted. The input signal need not be periodic, but it must be held at a given level for at least two full system clock cycles to ensure the level is properly sampled.

18.1. Timer 0 and Timer 1

Each timer is implemented as a 16-bit register accessed as two separate bytes: a low byte (TL0 or TL1) and a high byte (TH0 or TH1). The Counter/Timer Control register (TCON) is used to enable Timer 0 and Timer 1 as well as indicate status. Timer 0 interrupts can be enabled by setting the ET0 bit in the IE register (Section “10.4. Interrupt Register Descriptions” on page 100); Timer 1 interrupts can be enabled by setting the ET1 bit in the IE register (Section 10.4). Both counter/timers operate in one of four primary modes selected by setting the Mode Select bits T1M1–T0M0 in the Counter/Timer Mode register (TMOD). Each timer can be configured independently. Each operating mode is described below.

18.1.1. Mode 0: 13-bit Counter/Timer

Timer 0 and Timer 1 operate as 13-bit counter/timers in Mode 0. The following describes the configuration and operation of Timer 0. However, both timers operate identically, and Timer 1 is configured in the same manner as described for Timer 0.

The TH0 register holds the eight MSBs of the 13-bit counter/timer. TL0 holds the five LSBs in bit positions TL0.4–TL0.0. The three upper bits of TL0 (TL0.7–TL0.5) are indeterminate and should be masked out or ignored when reading. As the 13-bit timer register increments and overflows from 0x1FFF (all ones) to 0x0000, the timer overflow flag TF0 (TCON.5) is set and an interrupt will occur if Timer 0 interrupts are enabled.

The C/T0 bit (TMOD.2) selects the counter/timer's clock source. When C/T0 is set to logic 1, high-to-low transitions at the selected Timer 0 input pin (T0) increment the timer register (Refer to Section “13.1. Priority Crossbar Decoder” on page 122 for information on selecting and configuring external I/O pins). Clearing C/T selects the clock defined by the T0M bit (CKCON.3). When T0M is set, Timer 0 is

19.2.4. Frequency Output Mode

Frequency Output Mode produces a programmable-frequency square wave on the module's associated CEXn pin. The capture/compare module high byte holds the number of PCA clocks to count before the output is toggled. The frequency of the square wave is then defined by Equation 19.1.

$$F_{CEXn} = \frac{F_{PCA}}{2 \times PCA0CPHn}$$

Note: A value of 0x00 in the PCA0CPHn register is equal to 256 for this equation.

Equation 19.1. Square Wave Frequency Output

Where F_{PCA} is the frequency of the clock selected by the CPS2-0 bits in the PCA mode register, PCA0MD. The lower byte of the capture/compare module is compared to the PCA counter low byte; on a match, CEXn is toggled and the offset held in the high byte is added to the matched value in PCA0CPLn. Frequency Output Mode is enabled by setting the ECOMn, TOGn, and PWMn bits in the PCA0CPMn register.

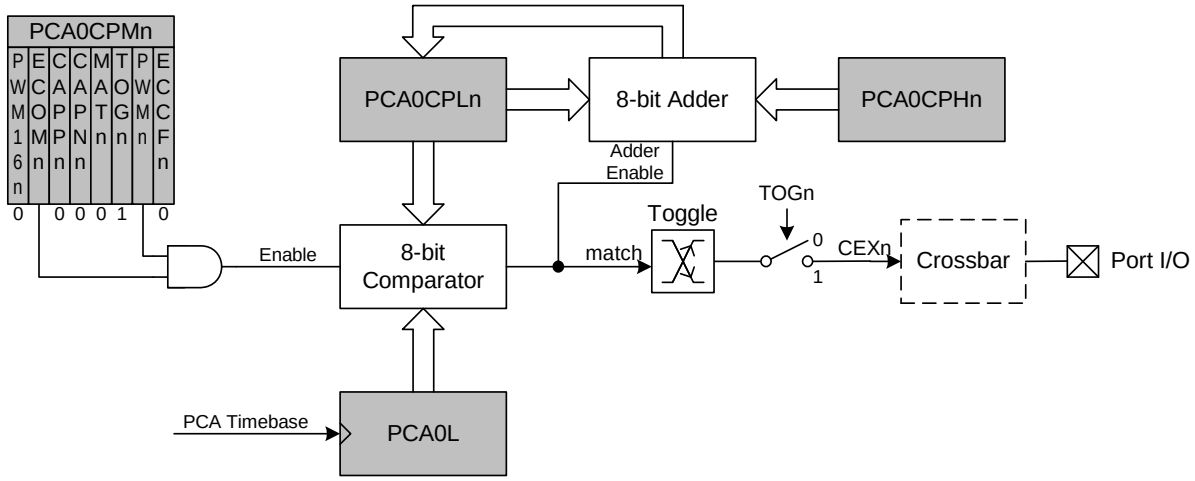

Figure 19.7. PCA Frequency Output Mode

19.2.5. 8-Bit Pulse Width Modulator Mode

Each module can be used independently to generate a pulse width modulated (PWM) output on its associated CEXn pin. The frequency of the output is dependent on the timebase for the PCA counter/timer. The duty cycle of the PWM output signal is varied using the module's PCA0CPHn capture/compare register. When the value in the low byte of the PCA counter/timer (PCA0L) is equal to the value in PCA0CPLn, the output on the CEXn pin will be set. When the count value in PCA0L overflows, the CEXn output will be reset (see Figure 19.8). Also, when the counter/timer low byte (PCA0L) overflows from 0xFF to 0x00, PCA0CPLn is reloaded automatically with the value stored in the module's capture/compare high byte (PCA0CPHn) without software intervention. Setting the ECOMn and PWMn bits in the PCA0CPMn register enables 8-Bit Pulse Width Modulator mode. The duty cycle for 8-Bit PWM Mode is given by Equation 19.2.

Important Note About Capture/Compare Registers: When writing a 16-bit value to the PCA0 Capture/Compare registers, the low byte should always be written first. Writing to PCA0CPLn clears the ECOMn bit to 0; writing to PCA0CPHn sets ECOMn to 1.

$$DutyCycle = \frac{(256 - PCA0CPHn)}{256}$$

Equation 19.2. 8-Bit PWM Duty Cycle

Using Equation 19.2, the largest duty cycle is 100% (PCA0CPHn = 0), and the smallest duty cycle is 0.39% (PCA0CPHn = 0xFF). A 0% duty cycle may be generated by clearing the ECOMn bit to 0.

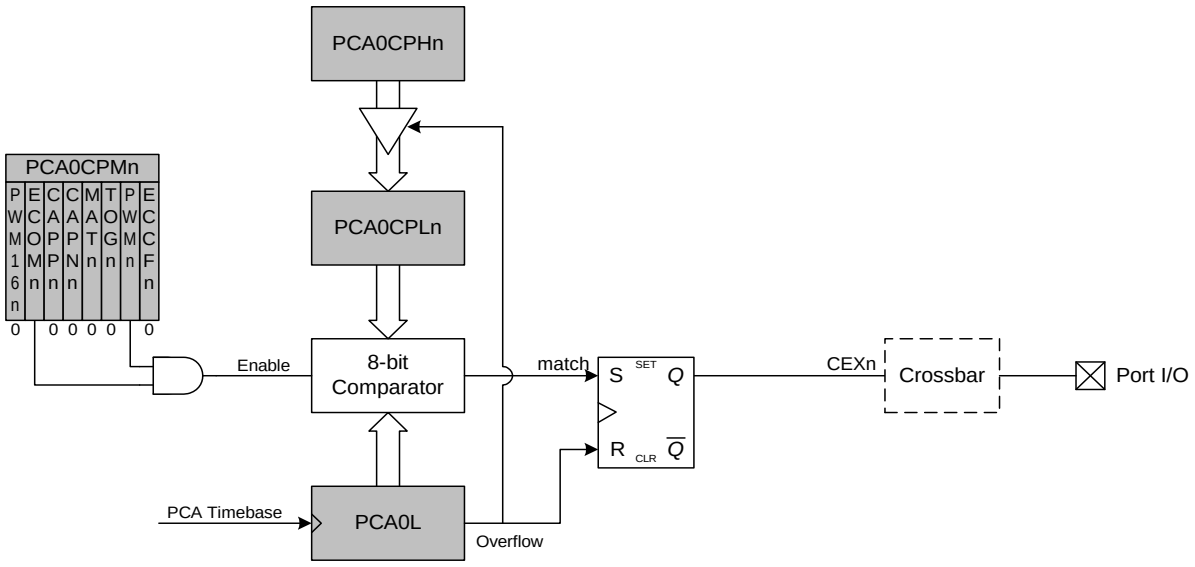

Figure 19.8. PCA 8-Bit PWM Mode Diagram