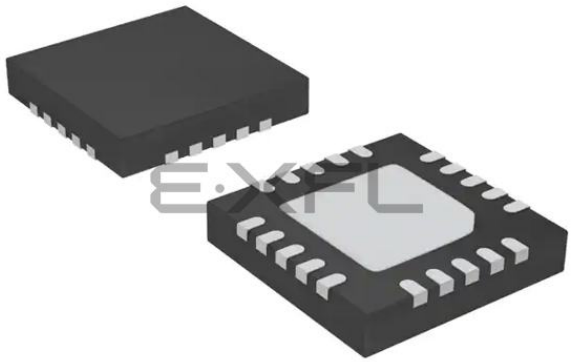

Welcome to E-XFL.COM

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	Last Time Buy
Core Processor	8051
Core Size	8-Bit
Speed	25MHz
Connectivity	SMBus (2-Wire/I²C), SPI, UART/USART
Peripherals	POR, PWM, Temp Sensor, WDT
Number of I/O	17
Program Memory Size	8KB (8K x 8)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	768 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	20-VFQFN Exposed Pad
Supplier Device Package	20-QFN (4x4)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051t630-gmr

List of Registers

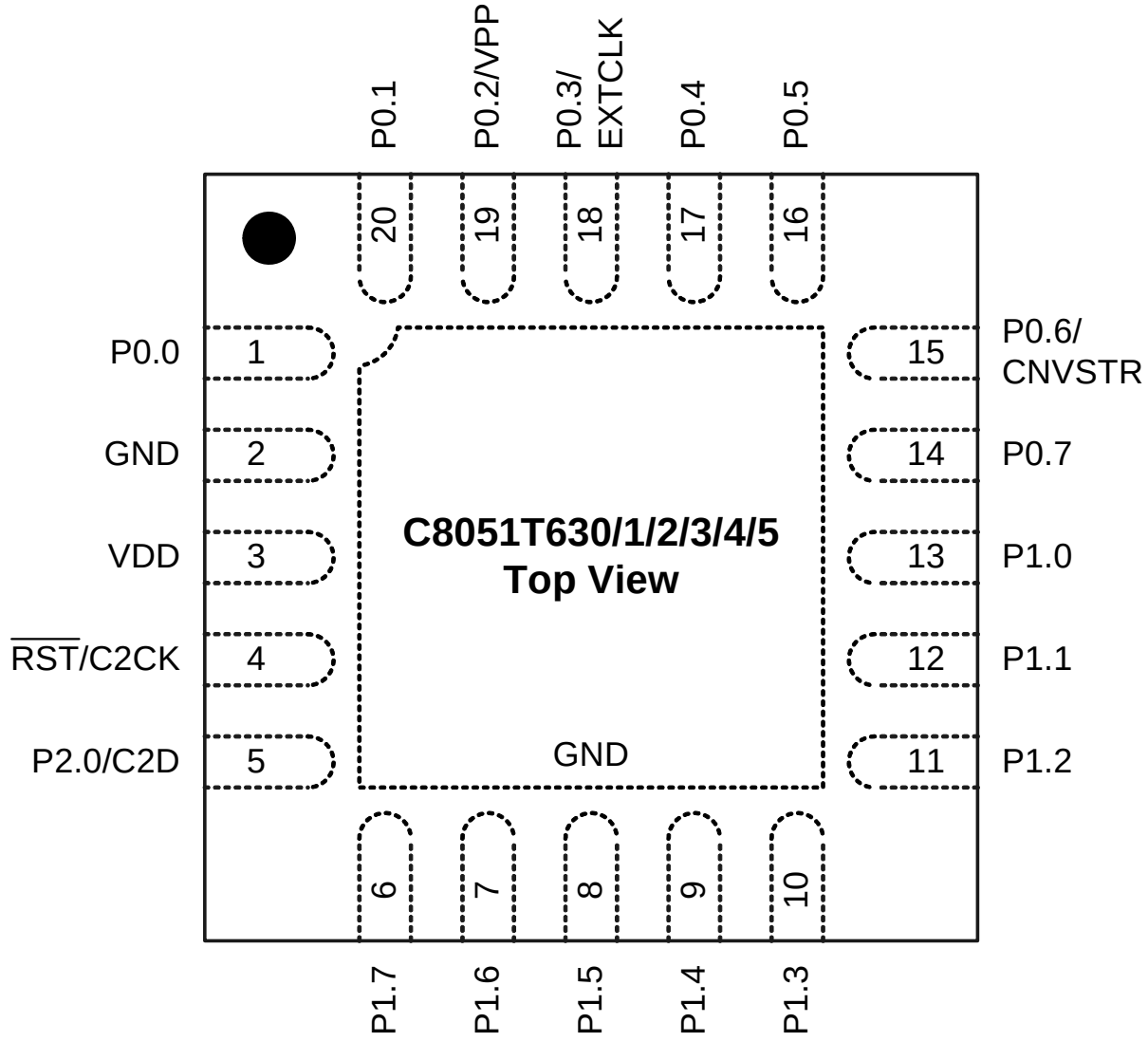

Figure 3.1. QFN-20 Pinout Diagram (Top View)

5.3. Typical Performance Curves

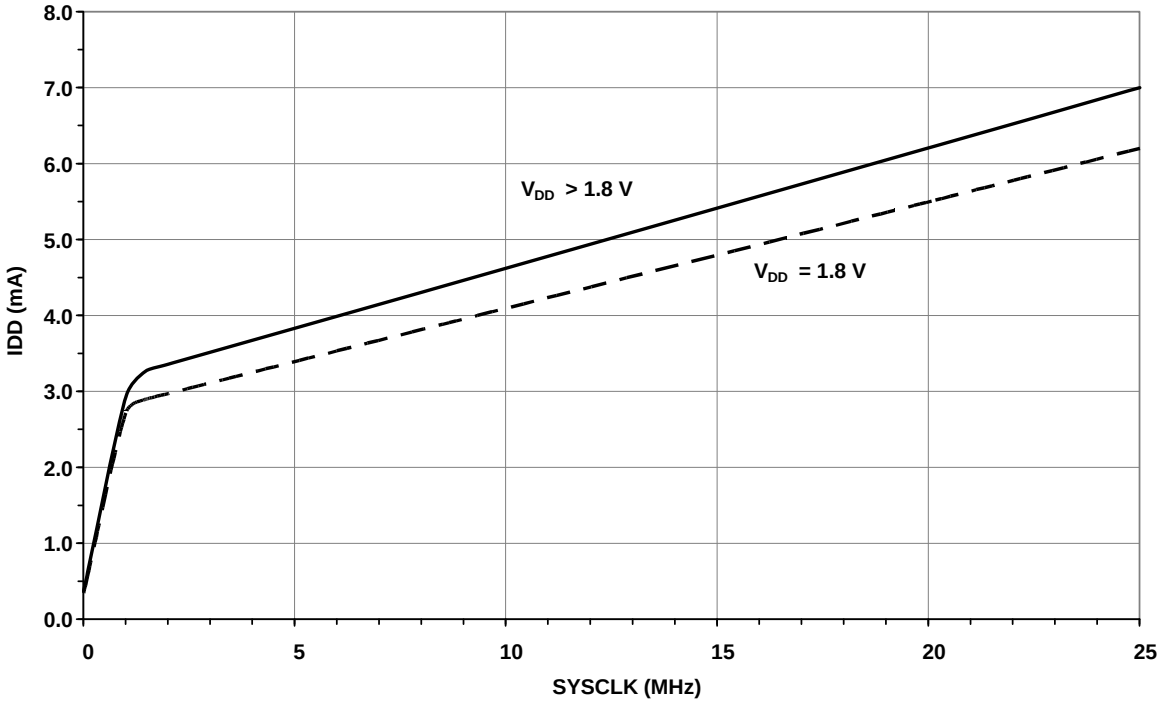

Figure 5.1. Normal Mode Digital Supply Current vs. Frequency (MPCE = 1)

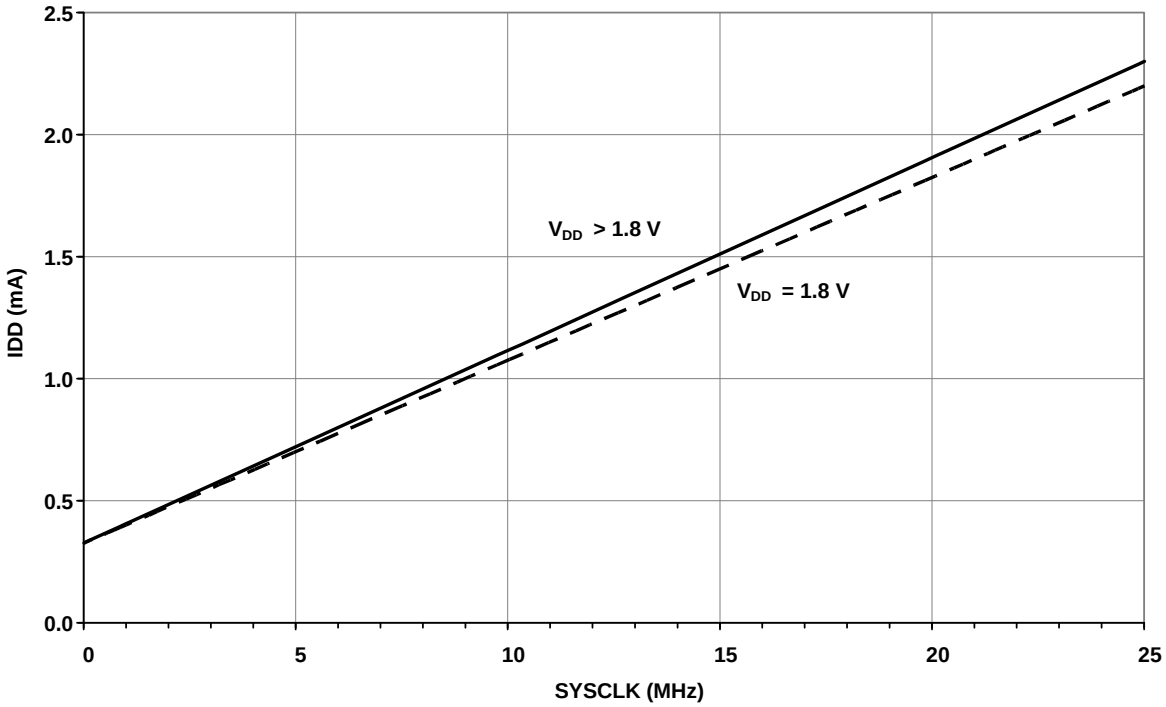

Figure 5.2. Idle Mode Digital Supply Current vs. Frequency (MPCE = 1)

6. 10-Bit ADC (ADC0, C8051T630/2/4 only)

ADC0 on the C8051T630/2/4 is a 500 ksps, 10-bit successive-approximation-register (SAR) ADC with integrated track-and-hold, a gain stage programmable to 1x or 0.5x, and a programmable window detector. The ADC is fully configurable under software control via Special Function Registers. The ADC may be configured to measure various different signals using the analog multiplexer described in Section “6.5. ADC0 Analog Multiplexer (C8051T630/2/4 only)” on page 43. The voltage reference for the ADC is selected as described in Section “9. Voltage Reference Options” on page 52. The ADC0 subsystem is enabled only when the AD0EN bit in the ADC0 Control register (ADC0CN) is set to logic 1. The ADC0 subsystem is in low power shutdown when this bit is logic 0.

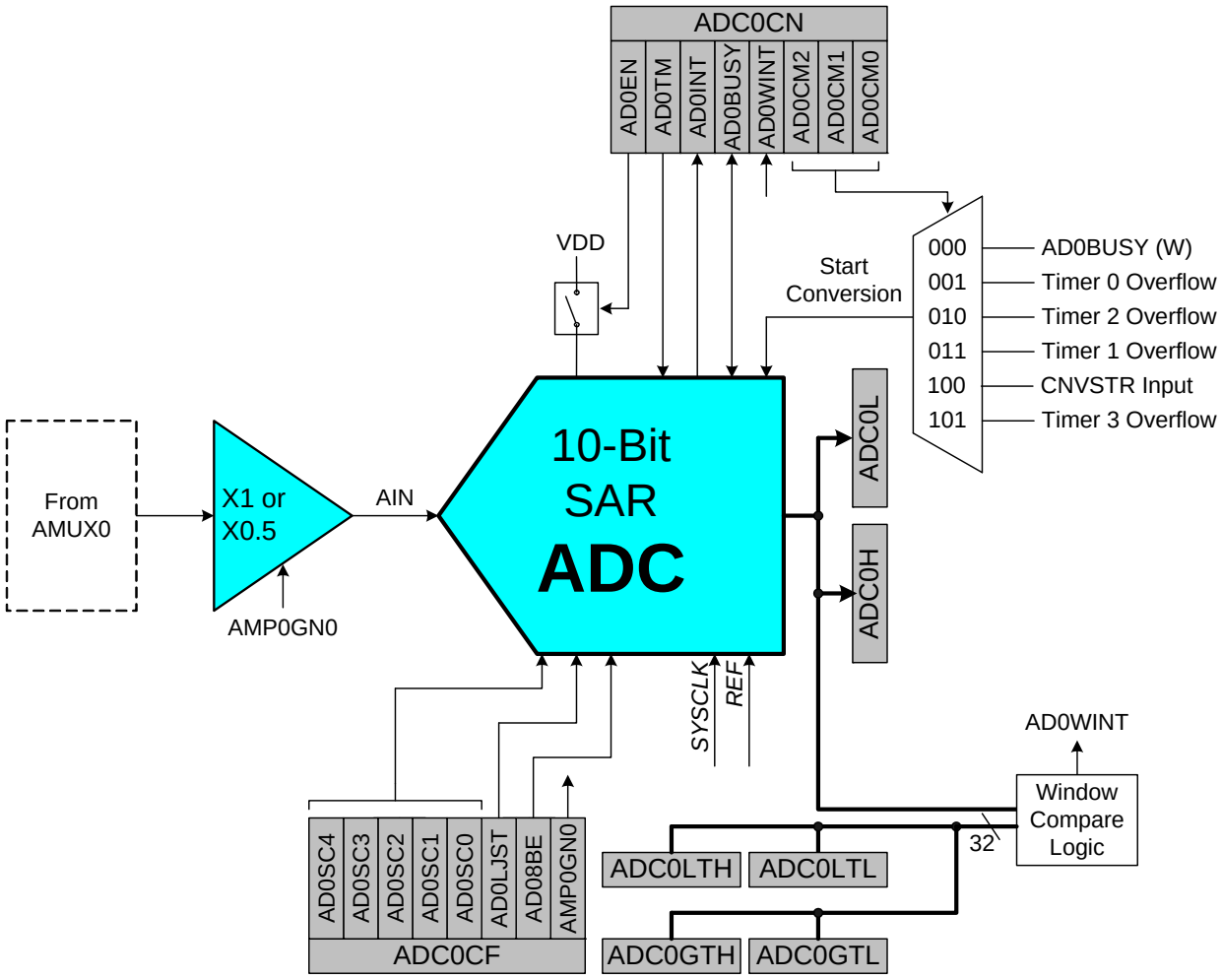

Figure 6.1. ADC0 Functional Block Diagram

SFR Definition 6.9. AMX0P: AMUX0 Positive Channel Select

Bit	7	6	5	4	3	2	1	0
Name				AMX0P[4:0]				
Type	R	R	R	R/W				
Reset	0	0	0	1	1	1	1	1

SFR Address = 0xBB

Bit	Name	Function
7:5	Unused	Unused. Read = 000b; Write = Don't Care.
4:0	AMX0P[4:0]	**AMUX0 Positive Input Selection.** 00000: P0.0 00001: P0.1 00010: P0.2 00011: P0.3 00100: P0.4 00101: P0.5 00110: P0.6 00111: P0.7 01000: P1.0 01001: P1.1 01010: P1.2 01011: P1.3 01100: P1.4 01101: P1.5 01110: P1.6 01111: P1.7 10000: Temp Sensor 10001: V_{DD} 10010 – 11111: no input selected

8.1.2. Update Output Based on Timer Overflow

Similar to the ADC operation, in which an ADC conversion can be initiated by a timer overflow independently of the processor, the IDAC outputs can use a Timer overflow to schedule an output update event. This feature is useful in systems where the IDAC is used to generate a waveform of a defined sampling rate by eliminating the effects of variable interrupt latency and instruction execution on the timing of the IDAC output. When the IDA0CM bits (IDA0CN.[6:4]) are set to 000, 001, 010 or 011, writes to both IDAC data registers (IDA0L and IDA0H) are held until an associated Timer overflow event (Timer 0, Timer 1, Timer 2 or Timer 3, respectively) occurs, at which time the IDA0H:IDA0L contents are copied to the IDAC input latches, allowing the IDAC output to change to the new value.

8.1.3. Update Output Based on CNVSTR Edge

The IDAC output can also be configured to update on a rising edge, falling edge, or both edges of the external CNVSTR signal. When the IDA0CM bits (IDA0CN.[6:4]) are set to 100, 101, or 110, writes to both IDAC data registers (IDA0L and IDA0H) are held until an edge occurs on the CNVSTR input pin. The particular setting of the IDA0CM bits determines whether IDAC outputs are updated on rising, falling, or both edges of CNVSTR. When a corresponding edge occurs, the IDA0H:IDA0L contents are copied to the IDAC input latches, allowing the IDAC output to change to the new value.

8.2. IDAC Output Mapping

The IDAC data registers (IDA0H and IDA0L) are left-justified, meaning that the eight MSBs of the IDAC output word are mapped to bits 7–0 of the IDA0H register, and the two LSBs of the IDAC output word are mapped to bits 7 and 6 of the IDA0L register. The data word mapping for the IDAC is shown in Figure 8.2.

IDA0H								IDA0L							
B9	B8	B7	B6	B5	B4	B3	B2	B1	B0						

Input Data Word (IDA09–IDA00)	Output Current IDA0OMD[1:0] = 1x	Output Current IDA0OMD[1:0] = 01	Output Current IDA0OMD[1:0] = 00
0x000	0 mA	0 mA	0 mA
0x001	1/1024 x 2 mA	1/1024 x 1 mA	1/1024 x 0.5 mA
0x200	512/1024 x 2 mA	512/1024 x 1 mA	512/1024 x 0.5 mA
0x3FF	1023/1024 x 2 mA	1023/1024 x 1 mA	1023/1024 x 0.5 mA

Figure 8.2. IDA0 Data Word Mapping

The full-scale output current of the IDAC is selected using the IDA0OMD bits (IDA0CN[1:0]). By default, the IDAC is set to a full-scale output current of 2 mA. The IDA0OMD bits can also be configured to provide full-scale output currents of 1 mA or 0.5 mA, as shown in SFR Definition 8.1.

9. Voltage Reference Options

The Voltage reference multiplexer for the ADC is configurable to use an externally connected voltage reference, the on-chip reference voltage generator routed to the VREF pin, the unregulated power supply voltage (V_{DD}), or the regulated 1.8 V internal supply (see Figure 9.1). The REFSL bit in the Reference Control register (REF0CN, SFR Definition 9.1) selects the reference source for the ADC. For an external source or the on-chip reference, REFSL should be set to 0 to select the VREF pin. To use V_{DD} as the reference source, REFSL should be set to 1. To override this selection and use the internal regulator as the reference source, the REGOVR bit can be set to 1.

The BIASE bit enables the internal voltage bias generator, which is used by many of the analog peripherals on the device. This bias is automatically enabled when any peripheral which requires it is enabled, and it does not need to be enabled manually. The bias generator may be enabled manually by writing a 1 to the BIASE bit in register REF0CN. The electrical specifications for the voltage reference circuit are given in Table 5.11.

The C8051T630/2/4 devices also include an on-chip voltage reference circuit which consists of a 1.2 V, temperature stable bandgap voltage reference generator and a selectable-gain output buffer amplifier. The buffer is configured for 1x or 2x gain using the REFBGS bit in register REF0CN. On the 1x gain setting the output voltage is nominally 1.2 V, and on the 2x gain setting the output voltage is nominally 2.4 V. The on-chip voltage reference can be driven on the VREF pin by setting the REFBE bit in register REF0CN to a 1. The maximum load seen by the VREF pin must be less than 200 µA to GND. Bypass capacitors of 0.1 µF and 4.7 µF are recommended from the VREF pin to GND, and a minimum of 0.1uF is required. If the on-chip reference is not used, the REFBE bit should be cleared to 0. Electrical specifications for the on-chip voltage reference are given in Table 5.11.

Important Note about the VREF Pin: When using either an external voltage reference or the on-chip reference circuitry, the VREF pin should be configured as an analog pin and skipped by the Digital Crossbar. Refer to Section “20. Port Input/Output” on page 109 for the location of the VREF pin, as well as details of how to configure the pin in analog mode and to be skipped by the crossbar.

SFR Definition 10.1. REG0CN: Voltage Regulator Control

Bit	7	6	5	4	3	2	1	0
Name	STOPCF	BYPASS						MPCE
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 Address = 0xC7

Bit	Name	Function
7	STOPCF	**Stop Mode Configuration.** This bit configures the regulator's behavior when the device enters STOP mode. 0: Regulator is still active in STOP mode. Any enabled reset source will reset the device. 1: Regulator is shut down in STOP mode. Only the $\overline{RST}$ pin or power cycle can reset the device.
6	BYPASS	**Bypass Internal Regulator.** This bit places the regulator in bypass mode, turning off the regulator, and allowing the core to run directly from the V_{DD} supply pin. 0: Normal Mode—Regulator is on. 1: Bypass Mode—Regulator is off, and the microcontroller core operates directly from the V_{DD} supply voltage. **IMPORTANT: Bypass mode is for use with an external regulator as the supply voltage only. Never place the regulator in bypass mode when the V_{DD} supply voltage is greater than the specifications given in Table 5.1 on page 23. Doing so may cause permanent damage to the device.**
5:1	Reserved	Reserved. Must Write 00000b
0	MPCE	**Memory Power Controller Enable.** This bit can help the system save power at slower system clock frequencies (about 2.0 MHz or less) by automatically shutting down the EPROM memory between clocks when information is not being fetched from the EPROM memory. 0: Normal Mode—Memory power controller disabled (EPROM memory is always on). 1: Low Power Mode—Memory power controller enabled (EPROM memory turns on/off as needed). **Note:** If an external clock source is used with the Memory Power Controller enabled, and the clock frequency changes from slow (<2.0 MHz) to fast (> 2.0 MHz), the EPROM power will turn on, and up to 20 clocks may be "skipped" to ensure that the EPROM power is stable before reading memory.

12. 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 26), 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 12.1 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

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.

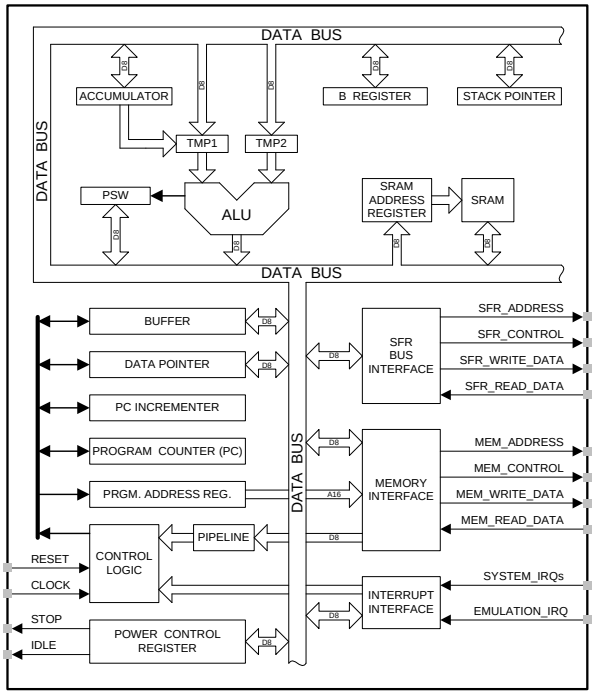

Figure 12.1. CIP-51 Block Diagram

SFR Definition 15.4. EIP1: Extended Interrupt Priority 1

Bit	7	6	5	4	3	2	1	0
Name	PT3	Reserved	PCP0	PPCA0	PADC0	PWADC0	PMAT	PSMB0
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 Address = 0xF6

Bit	Name	Function
7	PT3	**Timer 3 Interrupt Priority Control.** This bit sets the priority of the Timer 3 interrupt. 0: Timer 3 interrupts set to low priority level. 1: Timer 3 interrupts set to high priority level.
6	Reserved	Reserved. Must Write 0.
5	PCP0	**Comparator0 (CP0) Interrupt Priority Control.** This bit sets the priority of the CP0 interrupt. 0: CP0 interrupt set to low priority level. 1: CP0 interrupt set to high priority level.
4	PPCA0	**Programmable Counter Array (PCA0) Interrupt Priority Control.** This bit sets the priority of the PCA0 interrupt. 0: PCA0 interrupt set to low priority level. 1: PCA0 interrupt set to high priority level.
3	PADC0	**ADC0 Conversion Complete Interrupt Priority Control.** This bit sets the priority of the ADC0 Conversion Complete interrupt. 0: ADC0 Conversion Complete interrupt set to low priority level. 1: ADC0 Conversion Complete interrupt set to high priority level.
2	PWADC0	**ADC0 Window Comparator Interrupt Priority Control.** This bit sets the priority of the ADC0 Window interrupt. 0: ADC0 Window interrupt set to low priority level. 1: ADC0 Window interrupt set to high priority level.
1	PMAT	**Port Match Interrupt Priority Control.** This bit sets the priority of the Port Match Event interrupt. 0: Port Match interrupt set to low priority level. 1: Port Match interrupt set to high priority level.
0	PSMB0	**SMBus (SMB0) Interrupt Priority Control.** This bit sets the priority of the SMB0 interrupt. 0: SMB0 interrupt set to low priority level. 1: SMB0 interrupt set to high priority level.

software prior to entering the idle mode if the WDT was initially configured to allow this operation. This provides the opportunity for additional power savings, allowing the system to remain in the idle mode indefinitely, waiting for an external stimulus to wake up the system. Refer to Section “18.6. PCA Watchdog Timer Reset” on page 99 for more information on the use and configuration of the WDT.

17.2. Stop Mode

Setting the Stop Mode Select bit (PCON.1) causes the controller core to enter stop mode as soon as the instruction that sets the bit completes execution. In stop mode the internal oscillator, CPU, and all digital peripherals are stopped; the state of the external oscillator circuit is not affected. Each analog peripheral (including the external oscillator circuit) may be shut down individually prior to entering stop mode. Stop mode can only be terminated by an internal or external reset. On reset, the device performs the normal reset sequence and begins program execution at address 0x0000.

If enabled, the missing clock detector will cause an internal reset and thereby terminate the stop mode. The missing clock detector should be disabled if the CPU is to be put to in stop mode for longer than the MCD timeout.

By default, when in stop mode the internal regulator is still active. However, the regulator can be configured to shut down while in stop mode to save power. To shut down the regulator in stop mode, the STOPCF bit in register REG0CN should be set to 1 prior to setting the STOP bit (see SFR Definition 10.1). If the regulator is shut down using the STOPCF bit, only the RST pin or a full power cycle are capable of resetting the device.

17.3. Suspend Mode

Setting the SUSPEND bit (OSCICN.5) causes the hardware to halt the CPU and the high-frequency internal oscillator, and go into suspend mode as soon as the instruction that sets the bit completes execution. All internal registers and memory maintain their original data. Most digital peripherals are not active in suspend mode. The exception to this is the Port Match feature and Timer 3, when it is run from an external oscillator source or the internal low-frequency oscillator.

Suspend mode can be terminated by four types of events, a port match (described in Section “20.5. Port Match” on page 118), a Timer 3 overflow (described in Section “24.3. Timer 3” on page 185), a comparator low output (if enabled), or a device reset event. To run Timer 3 in suspend mode, the timer must be configured to clock from either the external clock source or the internal low-frequency oscillator source. When suspend mode is terminated, the device will continue execution on the instruction following the one that set the SUSPEND bit. If the wake event (port match or Timer 3 overflow) was configured to generate an interrupt, the interrupt will be serviced upon waking the device. If suspend mode is terminated by an internal or external reset, the CIP-51 performs a normal reset sequence and begins program execution at address 0x0000.

SFR Definition 18.2. RSTSRC: Reset Source

Bit	7	6	5	4	3	2	1	0
Name		MEMERR	C0RSEF	SWRSF	WDTRSF	MCDRSF	PORSF	PINRSF
Type	R	R	R/W	R/W	R	R/W	R/W	R
Reset	0	Varies	Varies	Varies	Varies	Varies	Varies	Varies

SFR Address = 0xEF

Bit	Name	Description	Write	Read
7	Unused	**Unused.**	Don't care.	0
6	MEMERR	**EPROM Error Reset Flag.**	N/A	Set to 1 if EPROM read/write error caused the last reset.
5	C0RSEF	**Comparator0 Reset Enable and Flag.**	Writing a 1 enables Comparator0 as a reset source (active-low).	Set to 1 if Comparator0 caused the last reset.
4	SWRSF	**Software Reset Force and Flag.**	Writing a 1 forces a system reset.	Set to 1 if last reset was caused by a write to SWRSF.
3	WDTRSF	**Watchdog Timer Reset Flag.**	N/A	Set to 1 if Watchdog Timer overflow caused the last reset.
2	MCDRSF	**Missing Clock Detector Enable and Flag.**	Writing a 1 enables the Missing Clock Detector. The MCD triggers a reset if a missing clock condition is detected.	Set to 1 if Missing Clock Detector timeout caused the last reset.
1	PORSF	**Power-On/V_{DD} Monitor Reset Flag, and V_{DD} monitor Reset Enable.**	Writing a 1 enables the V_{DD} monitor as a reset source. **Writing 1 to this bit before the V_{DD} monitor is enabled and stabilized may cause a system reset.**	Set to 1 anytime a power-on or V_{DD} monitor reset occurs. **When set to 1 all other RSTSRC flags are indeterminate.**
0	PINRSF	**HW Pin Reset Flag.**	N/A	Set to 1 if $\overline{\text{RST}}$ pin caused the last reset.

Note: Do not use read-modify-write operations on this register

19.4. External Oscillator Drive Circuit

The external oscillator circuit may drive an external capacitor or RC network. A CMOS clock may also provide a clock input. In RC, capacitor, or CMOS clock configuration, the clock source should be wired to the EXTCLK pin as shown in Figure 19.1. The type of external oscillator must be selected in the OSCXCN register, and the frequency control bits (XFCN) must be selected appropriately (see SFR Definition 19.5).

Important Note on External Oscillator Usage: Port pins must be configured when using the external oscillator circuit. When the external oscillator drive circuit is enabled in capacitor, RC, or CMOS clock mode, Port pin P0.3 is used as EXTCLK. The Port I/O Crossbar should be configured to skip the Port pin used by the oscillator circuit; see Section “20.3. Priority Crossbar Decoder” on page 114 for Crossbar configuration. Additionally, when using the external oscillator circuit in capacitor or RC mode, the associated Port pin should be configured as an **analog input**. In CMOS clock mode, the associated pin should be configured as a **digital input**. See Section “20.4. Port I/O Initialization” on page 116 for details on Port input mode selection.

SFR Definition 20.10. P0SKIP: Port 0 Skip

Bit	7	6	5	4	3	2	1	0
Name	P0SKIP[7:0]							
Type	R/W							
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD4

Bit	Name	Function
7:0	P0SKIP[7:0]	**Port 0 Crossbar Skip Enable Bits.** These bits select Port 0 pins to be skipped by the Crossbar Decoder. Port pins used for analog, special functions or GPIO should be skipped by the Crossbar. 0: Corresponding P0.n pin is not skipped by the Crossbar. 1: Corresponding P0.n pin is skipped by the Crossbar.

SFR Definition 20.11. P1: Port 1

Bit	7	6	5	4	3	2	1	0
Name	P1[7:0]							
Type	R/W							
Reset	1	1	1	1	1	1	1	1

SFR Address = 0x90; Bit-Addressable

Bit	Name	Description	Write	Read
7:0	P1[7:0]	**Port 1 Data.** Sets the Port latch logic value or reads the Port pin logic state in Port cells configured for digital I/O.	0: Set output latch to logic LOW. 1: Set output latch to logic HIGH.	0: P1.n Port pin is logic LOW. 1: P1.n Port pin is logic HIGH.

Table 21.1. SMBus Clock Source Selection

SMBCS1	SMBCS0	SMBus Clock Source
0	0	Timer 0 Overflow
0	1	Timer 1 Overflow
1	0	Timer 2 High Byte Overflow
1	1	Timer 2 Low Byte Overflow

The SMBCS1–0 bits select the SMBus clock source, which is used only when operating as a master or when the Free Timeout detection is enabled. When operating as a master, overflows from the selected source determine the absolute minimum SCL low and high times as defined in Equation 21.1. Note that the selected clock source may be shared by other peripherals so long as the timer is left running at all times. For example, Timer 1 overflows may generate the SMBus and UART baud rates simultaneously. Timer configuration is covered in Section “24. Timers” on page 169.

$$T_{HighMin} = T_{LowMin} = \frac{1}{f_{ClockSourceOverflow}}$$

Equation 21.1. Minimum SCL High and Low Times

The selected clock source should be configured to establish the minimum SCL High and Low times as per Equation 21.1. When the interface is operating as a master (and SCL is not driven or extended by any other devices on the bus), the typical SMBus bit rate is approximated by Equation 21.2.

$$BitRate = \frac{f_{ClockSourceOverflow}}{3}$$

Equation 21.2. Typical SMBus Bit Rate

Figure 21.4 shows the typical SCL generation described by Equation 21.2. Notice that T_{HIGH} is typically twice as large as T_{LOW}. The actual SCL output may vary due to other devices on the bus (SCL may be extended low by slower slave devices, or driven low by contending master devices). The bit rate when operating as a master will never exceed the limits defined by equation Equation 21.1.

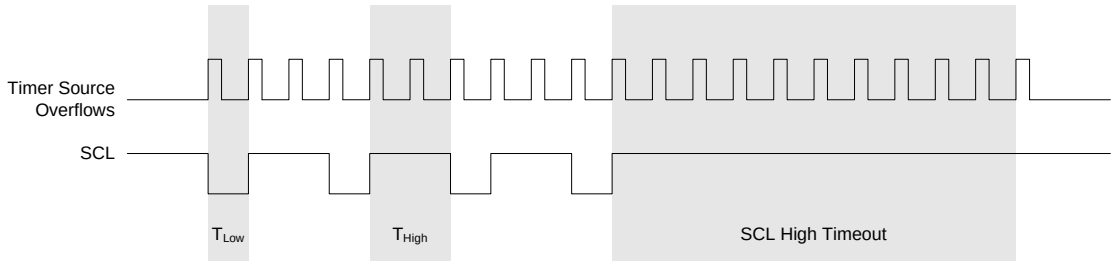

Figure 21.4. Typical SMBus SCL Generation

Setting the EXTHOLD bit extends the minimum setup and hold times for the SDA line. The minimum SDA setup time defines the absolute minimum time that SDA is stable before SCL transitions from low-to-high. The minimum SDA hold time defines the absolute minimum time that the current SDA value remains stable

21.5.2. Read Sequence (Master)

During a read sequence, an SMBus master reads data from a slave device. The master in this transfer will be a transmitter during the address byte, and a receiver during all data bytes. The SMBus interface generates the START condition and transmits the first byte containing the address of the target slave and the data direction bit. In this case the data direction bit (R/W) will be logic 1 (READ). Serial data is then received from the slave on SDA while the SMBus outputs the serial clock. The slave transmits one or more bytes of serial data.

If hardware ACK generation is disabled, the ACKRQ is set to 1 and an interrupt is generated after each received byte. Software must write the ACK bit at that time to ACK or NACK the received byte.

With hardware ACK generation enabled, the SMBus hardware will automatically generate the ACK/NACK, and then post the interrupt. **It is important to note that the appropriate ACK or NACK value should be set up by the software prior to receiving the byte when hardware ACK generation is enabled.**

Writing a 1 to the ACK bit generates an ACK; writing a 0 generates a NACK. Software should write a 0 to the ACK bit for the last data transfer, to transmit a NACK. The interface exits Master Receiver Mode after the STO bit is set and a STOP is generated. The interface will switch to Master Transmitter Mode if SMB0DAT is written while an active Master Receiver. Figure 21.6 shows a typical master read sequence. Two received data bytes are shown, though any number of bytes may be received. Notice that the ‘data byte transferred’ interrupts occur at different places in the sequence, depending on whether hardware ACK generation is enabled. The interrupt occurs **before** the ACK with hardware ACK generation disabled, and **after** the ACK when hardware ACK generation is enabled.

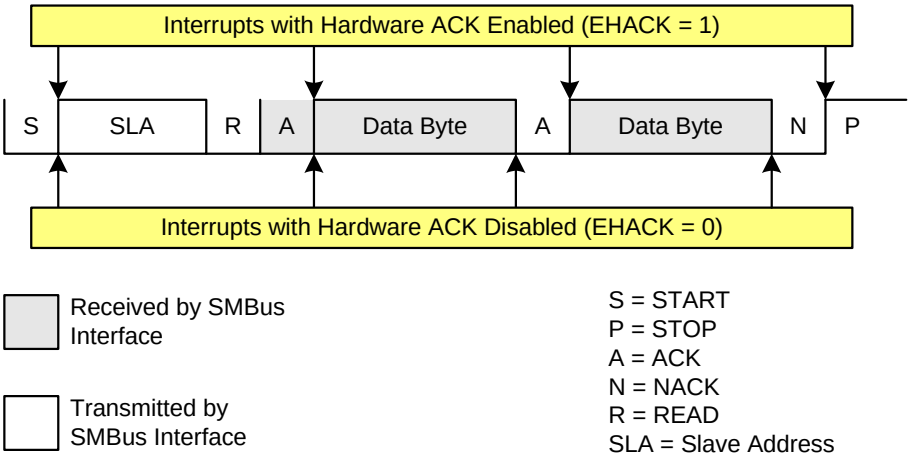

Figure 21.6. Typical Master Read Sequence

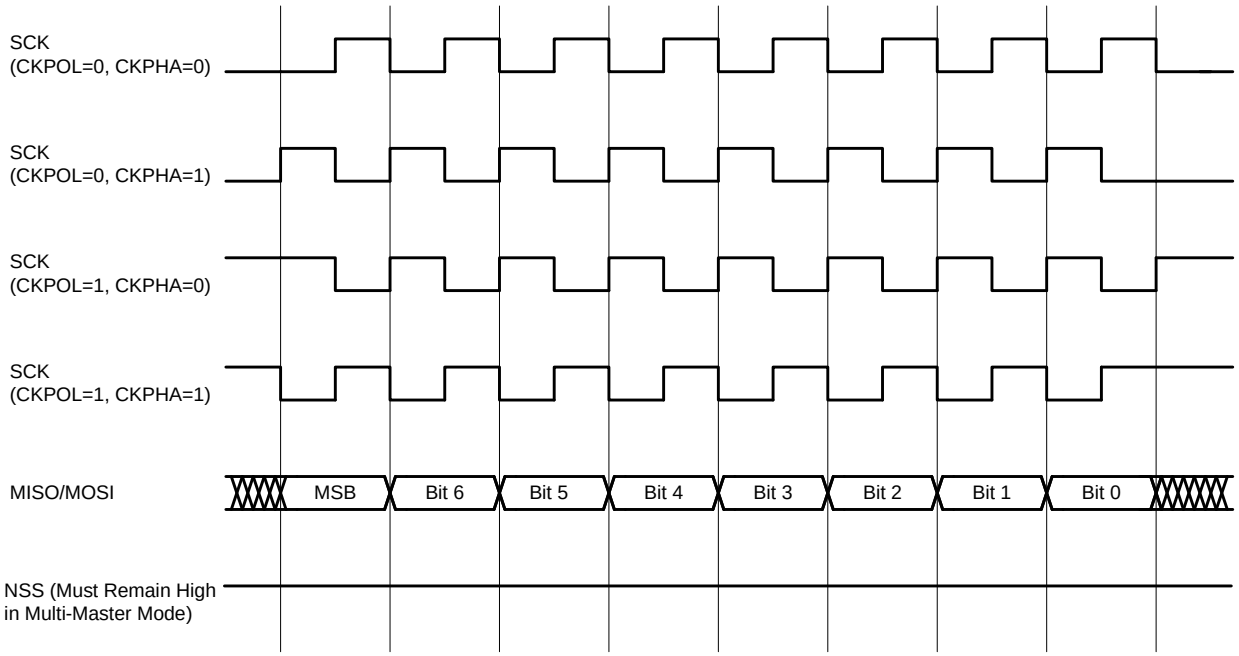

Figure 23.5. Master Mode Data/Clock Timing

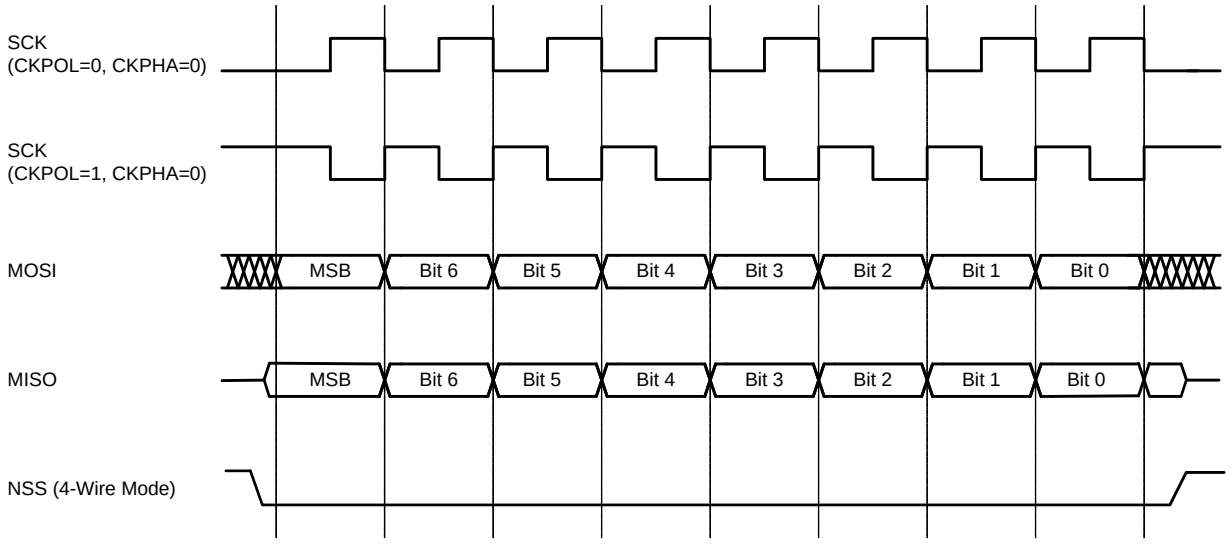

Figure 23.6. Slave Mode Data/Clock Timing (CKPHA = 0)

25.3.6. 16-Bit Pulse Width Modulator Mode

A PCA module may also be operated in 16-Bit PWM mode. 16-bit PWM mode is independent of the other (8/9/10/11-bit) PWM modes. In this mode, the 16-bit capture/compare module defines the number of PCA clocks for the low time of the PWM signal. When the PCA counter matches the module contents, the output on CEXn is asserted high; when the 16-bit counter overflows, CEXn is asserted low. To output a varying duty cycle, new value writes should be synchronized with PCA CCFn match interrupts. 16-Bit PWM Mode is enabled by setting the ECOMn, PWMn, and PWM16n bits in the PCA0CPMn register. For a varying duty cycle, match interrupts should be enabled (ECCFn = 1 AND MATn = 1) to help synchronize the capture/compare register writes. If the MATn bit is set to 1, the CCFn flag for the module will be set each time a 16-bit comparator match (rising edge) occurs. The CF flag in PCA0CN can be used to detect the overflow (falling edge). The duty cycle for 16-Bit PWM Mode is given by Equation 25.4.

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.

$$\text{Duty Cycle} = \frac{(65536 - PCA0CPn)}{65536}$$

Equation 25.4. 16-Bit PWM Duty Cycle

Using Equation 25.4, the largest duty cycle is 100% (PCA0CPn = 0), and the smallest duty cycle is 0.0015% (PCA0CPn = 0xFFFF). A 0% duty cycle may be generated by clearing the ECOMn bit to 0.

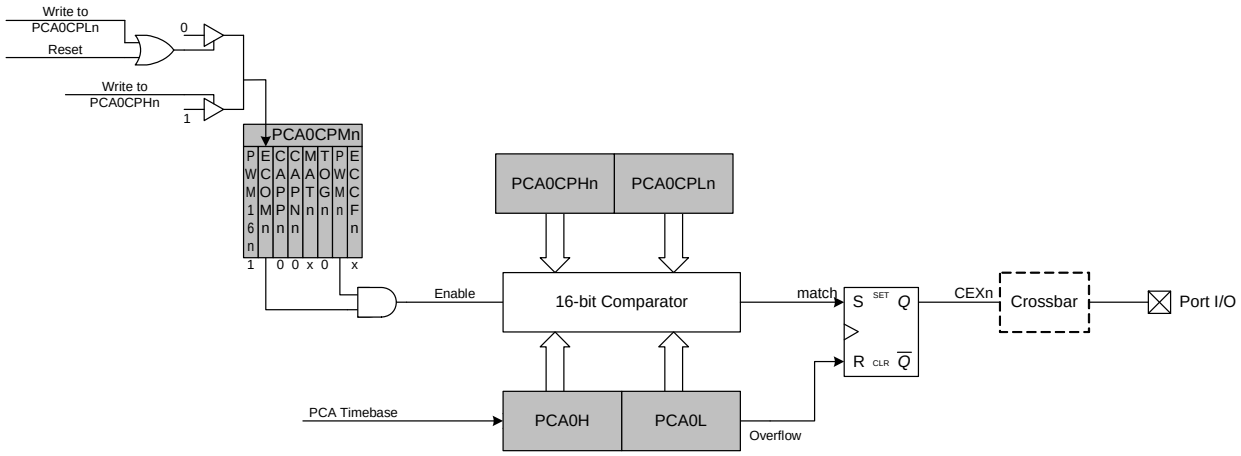

Figure 25.10. PCA 16-Bit PWM Mode