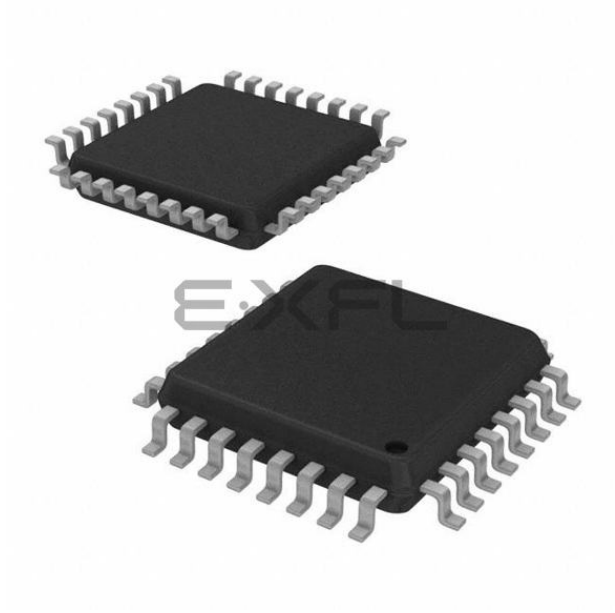

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Not For New Designs
Core Processor	8051
Core Size	8-Bit
Speed	100MHz
Connectivity	SMBus (2-Wire/I²C), SPI, UART/USART
Peripherals	POR, PWM, WDT
Number of I/O	29
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	-
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/c8051f364-c-gq

SFR Definition 6.1. IDA0CN: IDA0 Control

SFR Page: all pages
SFR Address: 0xB9

R/W	R/W	R/W	R/W	R	R	R/W	R/W	Reset Value
IDA0EN	IDA0CM			–	–	IDA0OMD		01110010
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

Bit 7: IDA0EN: IDA0 Enable.
0: IDA0 Disabled.
1: IDA0 Enabled.

Bits 6–4: IDA0CM[2:0]: IDA0 Update Source Select bits.
000: DAC output updates on Timer 0 overflow.
001: DAC output updates on Timer 1 overflow.
010: DAC output updates on Timer 2 overflow.
011: DAC output updates on Timer 3 overflow.
100: DAC output updates on rising edge of CNVSTR.
101: DAC output updates on falling edge of CNVSTR.
110: DAC output updates on any edge of CNVSTR.
111: DAC output updates on write to IDA0H. (default)

Bits 3–2: UNUSED. Read = 00b. Write = don't care.

Bits 1–0: IDA0OMD[1:0]: IDA0 Output Mode Select bits.
00: 0.5 mA full-scale output current.
01: 1.0 mA full-scale output current.
1x: 2.0 mA full-scale output current. (default)

SFR Definition 6.2. IDA0H: IDA0 Data Word MSB

SFR Page: all pages
SFR Address: 0x97

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	

Bits 7–0: IDA0 Data Word High-Order Bits.
Bits 7–0 are the most-significant bits of the 10-bit IDA0 Data Word.

7. Voltage Reference (C8051F360/1/2/6/7/8/9)

The Voltage reference MUX on the C8051F360/1/2/6/7/8/9 devices is configurable to use an externally connected voltage reference, the internal reference voltage generator, or the V_{DD} power supply voltage (see Figure 7.1). The REFSL bit in the Reference Control register (REF0CN) selects the reference source. For an external source or the internal reference, REFSL should be set to ‘0’. To use V_{DD} as the reference source, REFSL should be set to ‘1’.

The BIASE bit enables the internal voltage bias generator, which is used by the ADC, Temperature Sensor, internal oscillators, and Current DAC. This bias is enabled when any of the aforementioned peripherals are enabled. The bias generator may be enabled manually by writing a ‘1’ to the BIASE bit in register REF0CN; see SFR Definition 7.1 for REF0CN register details. The electrical specifications for the voltage reference circuit are given in Table 7.1.

The internal voltage reference circuit consists of a 1.2 V, temperature stable bandgap voltage reference generator and a gain-of-two output buffer amplifier. The internal voltage reference can be driven out on the VREF pin by setting the REFBE bit in register REF0CN to a ‘1’ (see SFR Definition 7.1). The maximum load seen by the VREF pin must be less than 200 µA to GND. When using the internal voltage reference, bypass capacitors of 0.1 µF and 4.7 µF are recommended from the VREF pin to GND. If the internal reference is not used, the REFBE bit should be cleared to ‘0’. Electrical specifications for the internal voltage reference are given in Table 7.1.

Important Note about the VREF Pin: Port pin P0.3 on the C8051F360 device and P0.0 on C8051F361/2/6/7/89 devices is used as the external VREF input and as an output for the internal VREF. When using either an external voltage reference or the internal reference circuitry, the port pin should be configured as an analog pin, and skipped by the Digital Crossbar. To configure the port pin as an analog pin, set the appropriate bit to ‘0’ in register P0MDIN. To configure the Crossbar to skip the VREF port pin, set the appropriate bit to ‘1’ in register P0SKIP. Refer to Section “17. Port Input/Output” on page 182 for

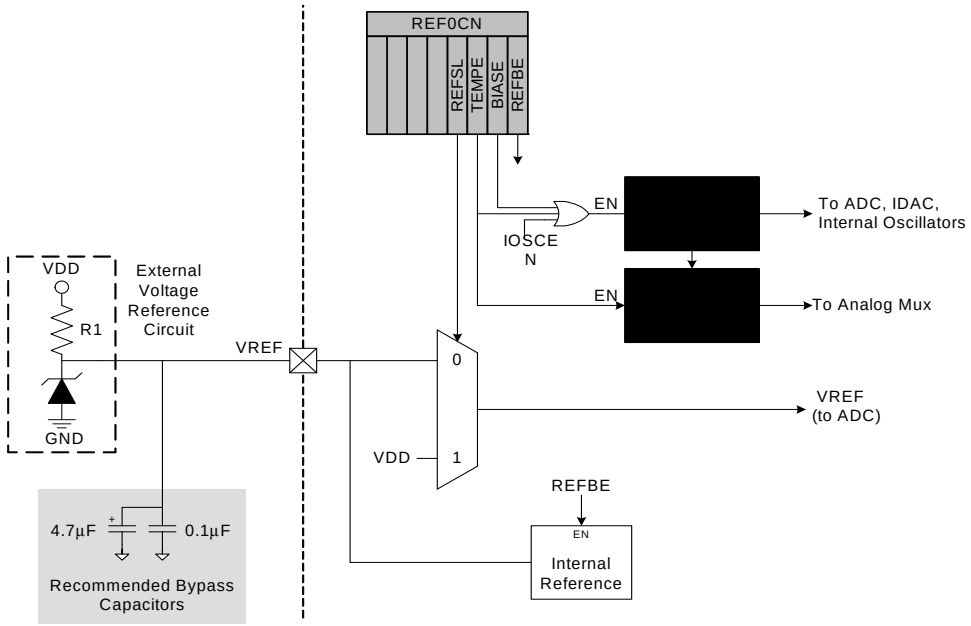

Figure 7.1. Voltage Reference Functional Block Diagram

11.6. Rounding and Saturation

A Rounding Engine is included, which can be used to provide a rounded result when operating on fractional numbers. MAC0 uses an unbiased rounding algorithm to round the data stored in bits 31–16 of the accumulator, as shown in Table 11.1. Rounding occurs during the third stage of the MAC0 pipeline, after any shift operation, or on a write to the LSB of the accumulator. The rounded results are stored in the rounding registers: MAC0RNDH (SFR Definition 11.12) and MAC0RNDL (SFR Definition 11.13). The accumulator registers are not affected by the rounding engine. Although rounding is primarily used for fractional data, the data in the rounding registers is updated in the same way when operating in integer mode.

Table 11.1. MAC0 Rounding (MAC0SAT = 0)

Accumulator Bits 15–0 (MAC0ACC1:MAC0ACC0)	Accumulator Bits 31–16 (MAC0ACC3:MAC0ACC2)	Rounding Direction	Rounded Results (MAC0RNDH:MAC0RNDL)
Greater Than 0x8000	Anything	Up	(MAC0ACC3:MAC0ACC2) + 1
Less Than 0x8000	Anything	Down	(MAC0ACC3:MAC0ACC2)
Equal To 0x8000	Odd (LSB = 1)	Up	(MAC0ACC3:MAC0ACC2) + 1
Equal To 0x8000	Even (LSB = 0)	Down	(MAC0ACC3:MAC0ACC2)

The rounding engine can also be used to saturate the results stored in the rounding registers. If the MAC0SAT bit is set to ‘1’ and the rounding register overflows, the rounding registers will saturate. When a positive overflow occurs, the rounding registers will show a value of 0x7FFF when saturated. For a negative overflow, the rounding registers will show a value of 0x8000 when saturated. If the MAC0SAT bit is cleared to ‘0’, the rounding registers will not saturate.

11.7. Usage Examples

This section details some software examples for using MAC0. Section 11.7.1 shows a series of two MAC operations using fractional numbers. Section 11.7.2 shows a single operation in Multiply Only mode with integer numbers. The last example, shown in Section 11.7.3, demonstrates how the left-shift and right-shift operations can be used to modify the accumulator. All of the examples assume that all of the flags in the MAC0STA register are initially set to ‘0’.

11.7.1. Multiply and Accumulate Example

The example below implements the equation:

$$(0.5 \times 0.25) + (0.5 \times -0.25) = 0.125 - 0.125 = 0.0$$

```
MOV   MAC0CF, #0Ah    ; Set to Clear Accumulator, Use fractional numbers
MOV   MAC0AH, #40h    ; Load MAC0A register with 4000 hex = 0.5 decimal
MOV   MAC0AL, #00h
MOV   MAC0BH, #20h    ; Load MAC0B register with 2000 hex = 0.25 decimal
MOV   MAC0BL, #00h    ; This line initiates the first MAC operation
MOV   MAC0BH, #E0h    ; Load MAC0B register with E000 hex = -0.25 decimal
MOV   MAC0BL, #00h    ; This line initiates the second MAC operation
NOP
NOP                   ; After this instruction, the Accumulator should be equal to 0,
                      ; and the MAC0STA register should be 0x04, indicating a zero
NOP                   ; After this instruction, the Rounding register is updated
```

Steps 3–8 must be repeated for each byte to be written

For block Flash writes, the Flash write procedure is only performed after the last byte of each block is written with the MOVX write instruction. When writing to addresses located in any of the four code banks, a Flash write block is four bytes long, from addresses ending in 00b to addresses ending in 11b. Writes must be performed sequentially (i.e. addresses ending in 00b, 01b, 10b, and 11b must be written in order). The Flash write will be performed following the MOVX write that targets the address ending in 11b. The Flash write will be performed following the MOVX write that targets the address ending in 1b. If any bytes in the block do not need to be updated in Flash, they should be written to 0xFF. The recommended procedure for writing Flash in blocks is as follows:

Step 1. Disable interrupts.
Step 2. Set CHBLKW (register CCH0CN) to select block write mode.
Step 3. Write the first key code to FLKEY: 0xA5.
Step 4. Write the second key code to FLKEY: 0xF1.
Step 5. Set PSWE (register PSCTL) to redirect MOVX commands to write to Flash.
Step 6. Clear the PSEE bit (register PSCTL).
Step 7. Using the MOVX instruction, write the first data byte to the first block location (ending in 00b).
Step 8. Clear the PSWE bit to redirect MOVX commands to the XRAM data space.
Step 9. Write the first key code to FLKEY: 0xA5.
Step 10. Write the second key code to FLKEY: 0xF1.
Step 11. Set PSWE (register PSCTL) to redirect MOVX commands to write to Flash.
Step 12. Clear the PSEE bit (register PSCTL).
Step 13. Using the MOVX instruction, write the second data byte to the second block location (ending in 01b).
Step 14. Clear the PSWE bit to redirect MOVX commands to the XRAM data space.
Step 15. Write the first key code to FLKEY: 0xA5.
Step 16. Write the second key code to FLKEY: 0xF1.
Step 17. Set PSWE (register PSCTL) to redirect MOVX commands to write to Flash.
Step 18. Clear the PSEE bit (register PSCTL).
Step 19. Using the MOVX instruction, write the third data byte to the third block location (ending in 10b).
Step 20. Clear the PSWE bit to redirect MOVX commands to the XRAM data space.
Step 21. Write the first key code to FLKEY: 0xA5.
Step 22. Write the second key code to FLKEY: 0xF1.
Step 23. Set PSWE (register PSCTL) to redirect MOVX commands to write to Flash.
Step 24. Clear the PSEE bit (register PSCTL).
Step 25. Using the MOVX instruction, write the fourth data byte to the last block location (ending in 11b).
Step 26. Clear the PSWE bit to redirect MOVX commands to the XRAM data space.
Step 27. Re-enable interrupts.

Steps 3-26 must be repeated for each block to be written.

13.1.4. Non-volatile Data Storage

The Flash memory can be used for non-volatile data storage as well as program code. This allows data such as calibration coefficients to be calculated and stored at run time. Data is written and erased using the MOVX write instruction (as described in Section 13.1.2 and Section 13.1.3) and read using the MOVC instruction. Note: MOVX read instructions always target XRAM.

13.3. Flash Write and Erase Guidelines

Any system which contains routines which write or erase Flash memory from software involves some risk that the write or erase routines will execute unintentionally if the CPU is operating outside its specified operating range of V_{DD}, system clock frequency, or temperature. This accidental execution of Flash modifying code can result in alteration of Flash memory contents causing a system failure that is only recoverable by re-Flashing the code in the device.

To help prevent the accidental modification of Flash by firmware, the V_{DD} Monitor must be enabled and enabled as a reset source on C8051F36x devices for the Flash to be successfully modified. **If either the V_{DD} Monitor or the V_{DD} Monitor reset source is not enabled, a Flash Error Device Reset will be generated when the firmware attempts to modify the Flash.**

The following guidelines are recommended for any system that contains routines which write or erase Flash from code.

13.3.1. V_{DD} Maintenance and the V_{DD} Monitor

1. If the system power supply is subject to voltage or current "spikes," add sufficient transient protection devices to the power supply to ensure that the supply voltages listed in the Absolute Maximum Ratings table are not exceeded.
2. Make certain that the minimum V_{DD} rise time specification of 1 ms is met. If the system cannot meet this rise time specification, then add an external V_{DD} brownout circuit to the /RST pin of the device that holds the device in reset until V_{DD} reaches V_{RST} and re-asserts /RST if V_{DD} drops below V_{RST}. Please see Table 12.1, “Reset Electrical Characteristics,” on page 134 for more information on the VDD Monitor Threshold voltage (V_{RST}).
3. Keep the on-chip V_{DD} Monitor enabled and enable the V_{DD} Monitor as a reset source as early in code as possible. This should be the first set of instructions executed after the Reset Vector. For 'C'-based systems, this will involve modifying the startup code added by the 'C' compiler. See your compiler documentation for more details. Make certain that there are no delays in software between enabling the V_{DD} Monitor and enabling the V_{DD} Monitor as a reset source. Code examples showing this can be found in AN201, "Writing to Flash from Firmware", available from the Silicon Laboratories web site.

   Note: On C8051F36x devices, both the V_{DD} Monitor and the V_{DD} Monitor reset source must be enabled to write or erase Flash without generating a Flash Error Device Reset.

4. As an added precaution, explicitly enable the V_{DD} Monitor and enable the V_{DD} Monitor as a reset source inside the functions that write and erase Flash memory. The V_{DD} Monitor enable instructions should be placed just after the instruction to set PSWE to a '1', but before the Flash write or erase operation instruction.
5. Make certain that all writes to the RSTSRC (Reset Sources) register use direct assignment operators and explicitly DO NOT use the bit-wise operators (such as AND or OR). For example, "RSTSRC = 0x02" is correct, but "RSTSRC |= 0x02" is incorrect.
6. Make certain that all writes to the RSTSRC register explicitly set the PORSF bit to a '1'. Areas to check are initialization code which enables other reset sources, such as the Missing Clock Detector or Comparator, for example, and instructions which force a Software Reset. A global search on "RSTSRC" can quickly verify this.

15.2. Configuring the External Memory Interface

Configuring the External Memory Interface consists of five steps:

1. Configure the Output Modes of the associated port pins as either push-pull or open-drain (push-pull is most common), and skip the associated pins in the crossbar.
2. Configure Port latches to “park” the EMIF pins in a dormant state (usually by setting them to logic ‘1’).
3. Select Multiplexed mode or Non-multiplexed mode.
4. Select the memory mode (on-chip only, split mode without bank select, split mode with bank select, or off-chip only).
5. Set up timing to interface with off-chip memory or peripherals.

Each of these five steps is explained in detail in the following sections. The Port selection, Multiplexed mode selection, and Mode bits are located in the EMI0CF register shown in SFR Definition 15.2.

15.3. Port Configuration

The External Memory Interface appears on Ports 1, 2 (non-multiplexed mode only), 3, and 4 when it is used for off-chip memory access. When the EMIF is used in multiplexed mode, the Crossbar should be configured to skip over the ALE control line (P0.0) using the P0SKIP register. The other control lines, /RD (P4.4) and /WR (P4.5), are not available on the Crossbar and do not need to be skipped. For more information about configuring the Crossbar, see Section “17.3. General Purpose Port I/O” on page 189. The EMIF pinout is shown in Table 15.1 on page 154.

The External Memory Interface claims the associated Port pins for memory operations ONLY during the execution of an off-chip MOVX instruction. Once the MOVX instruction has completed, control of the Port pins reverts to the Port latches or to the Crossbar settings for those pins. See Section “17. Port Input/Output” on page 182 for more information about the Crossbar and Port operation and configuration. **The Port latches should be explicitly configured to ‘park’ the External Memory Interface pins in a dormant state, most commonly by setting them to a logic ‘1’**.

During the execution of the MOVX instruction, the External Memory Interface will explicitly disable the drivers on all Port pins that are acting as Inputs (Data[7:0] during a READ operation, for example). The Output mode of the Port pins (whether the pin is configured as Open-Drain or Push-Pull) is unaffected by the External Memory Interface operation, and remains controlled by the PnMDOUT registers. In most cases, the output modes of all EMIF pins should be configured for push-pull mode.

15.6.1.3.8-bit MOVX with Bank Select: EMI0CF[4:2] = ‘110’.

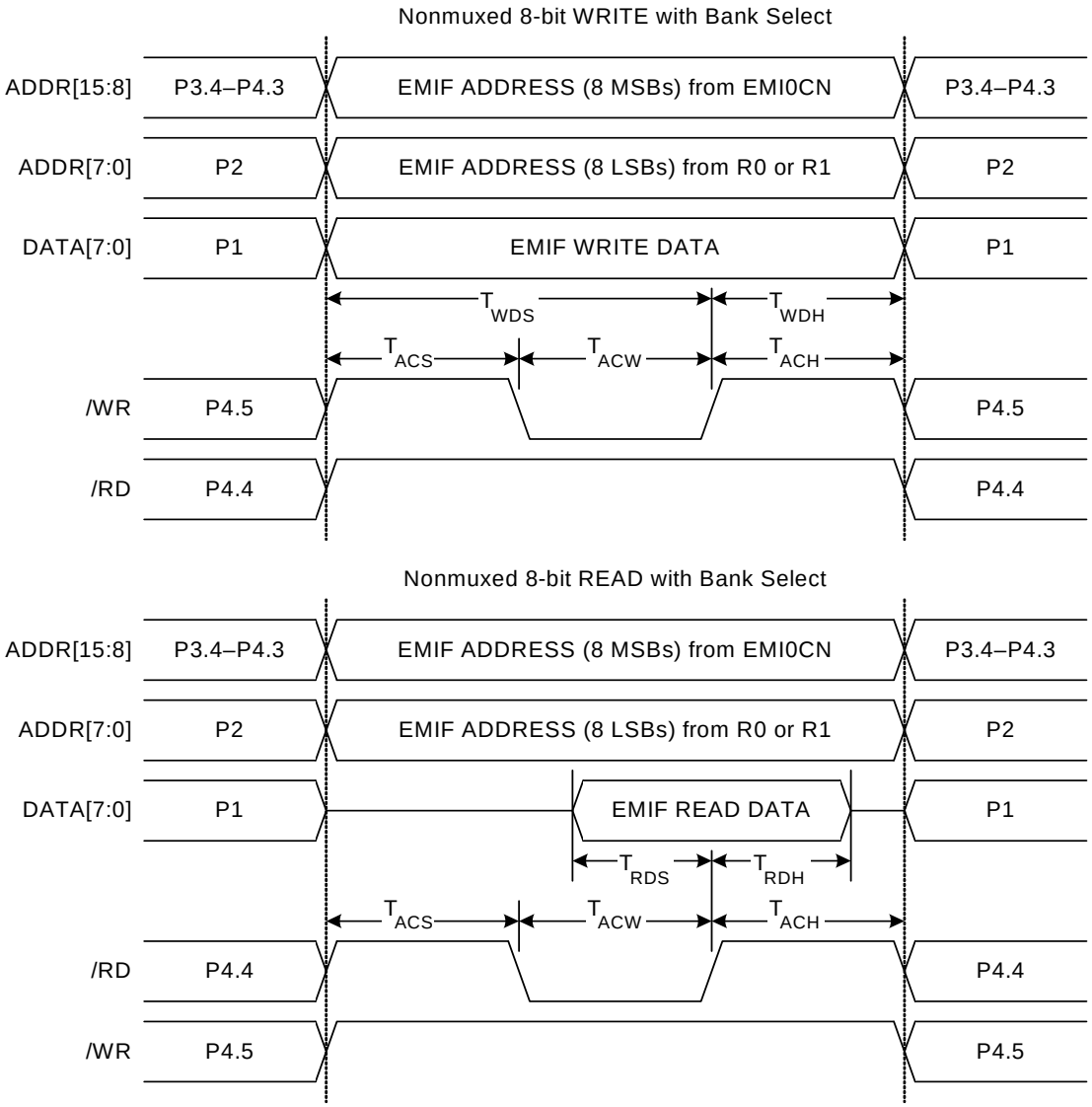

Figure 15.6. Non-multiplexed 8-bit MOVX with Bank Select Timing

Table 15.2. AC Parameters for External Memory Interface

Parameter	Description	Min*	Max*	Units
T_{ACS}	Address/Control Setup Time	0	3 x T_{SYSCLK}	ns
T_{ACW}	Address/Control Pulse Width	1 x T_{SYSCLK}	16 x T_{SYSCLK}	ns
T_{ACH}	Address/Control Hold Time	0	3 x T_{SYSCLK}	ns
T_{ALEH}	Address Latch Enable High Time	1 x T_{SYSCLK}	4 x T_{SYSCLK}	ns
T_{ALEL}	Address Latch Enable Low Time	1 x T_{SYSCLK}	4 x T_{SYSCLK}	ns
T_{WDS}	Write Data Setup Time	1 x T_{SYSCLK}	19 x T_{SYSCLK}	ns
T_{WDH}	Write Data Hold Time	0	3 x T_{SYSCLK}	ns
T_{RDS}	Read Data Setup Time	20		ns
T_{RDH}	Read Data Hold Time	0		ns

***Note:** T_{SYSCLK} is equal to one period of the device system clock (SYSCLK).

SFR Definition 16.4. CLKSEL: System Clock Selection

SFR Page: F
SFR Address: 0x8F

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
Reserved	Reserved	CLKDIV1	CLKDIV0	Reserved	CLKSL2	CLKSL1	CLKSL0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

Bits 7–6: RESERVED. Read = 00b. Must Write 00b.
Bits 5–4: CLKDIV1-0: Output SYSCLK Divide Factor.
These bits can be used to pre-divide SYSCLK before it is output to a port pin through the crossbar.
00: Output will be SYSCLK.
01: Output will be SYSCLK/2.
10: Output will be SYSCLK/4.
11: Output will be SYSCLK/8.
See Section “17. Port Input/Output” on page 182 for more details about routing this output to a port pin.
Bit 3: RESERVED. Read = 0b. Must Write 0b.
Bits 2–0: CLKSL2–0: System Clock Source Select Bits.
000: SYSCLK derived from the high-frequency Internal Oscillator, and scaled as per the IFCN bits in OSCICN.
001: SYSCLK derived from the External Oscillator circuit.
010: SYSCLK derived from the low-frequency Internal Oscillator, and scaled as per the OSCLD bits in OSCLCN.
011: RESERVED.
100: SYSCLK derived from the PLL.
101-11x: RESERVED.

SFR Definition 17.1. XBR0: Port I/O Crossbar Register 0

SFR Page: F
SFR Address: 0xE1

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
CP1AE	CP1E	CP0AE	CP0E	SYSCKE	SMB0E	SPI0E	URT0E	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

Bit 7: CP1AE: Comparator1 Asynchronous Output Enable
0: Asynchronous CP1 unavailable at Port pin.
1: Asynchronous CP1 routed to Port pin.

Bit 6: CP1E: Comparator1 Output Enable
0: CP1 unavailable at Port pin.
1: CP1 routed to Port pin.

Bit 5: CP0AE: Comparator0 Asynchronous Output Enable
0: Asynchronous CP0 unavailable at Port pin.
1: Asynchronous CP0 routed to Port pin.

Bit 4: CP0E: Comparator0 Output Enable
0: CP0 unavailable at Port pin.
1: CP0 routed to Port pin.

Bit 3: SYSCKE: /SYSCLK Output Enable
0: /SYSCLK unavailable at Port pin.
1: /SYSCLK (divided by 1, 2, 4, or 8) routed to Port pin. The divide factor is determined by the CLKDIV1–0 bits in register CLKSEL (See Section Section “16. Oscillators” on page 168).

Bit 2: SMB0E: SMBus I/O Enable
0: SMBus I/O unavailable at Port pins.
1: SMBus I/O routed to Port pins.

Bit 1: SPI0E: SPI I/O Enable
0: SPI I/O unavailable at Port pins.
1: SPI I/O routed to Port pins. Note that the SPI can be assigned either 3 or 4 GPIO pins.

Bit 0: URT0E: UART I/O Output Enable
0: UART I/O unavailable at Port pin.
1: UART TX0, RX0 routed to Port pins P0.1 and P0.2 (C8051F360/3) or P0.4 and P0.5 (C8051F361/2/4/5/6/7/8/9).

SFR Definition 17.21. P3: Port3

SFR Page: **all pages**
SFR Address: **0xB0** (bit addressable)

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
P3.7	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0	11111111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

Bits 7–0: P3.[7:0]
Write - Output appears on I/O pins per Crossbar Registers.
0: Logic Low Output.
1: Logic High Output (high impedance if corresponding P3MDOUT.n bit = 0).
Read - Always reads ‘0’ if selected as analog input in register P3MDIN. Directly reads Port pin when configured as digital input.
0: P3.n pin is logic low.
1: P3.n pin is logic high.

SFR Definition 17.22. P3MDIN: Port3 Input Mode

SFR Page: **F**
SFR Address: **0xF4**

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
								11111111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

Bits 7–0: Analog Input Configuration Bits for P3.7-P3.0 (respectively).
Port pins configured as analog inputs have their weak pullup, digital driver, and digital receiver disabled.
0: Corresponding P3.n pin is configured as an analog input.
1: Corresponding P3.n pin is not configured as an analog input.

SFR Definition 18.2. SMB0CN: SMBus Control

SFR Page: all pages
SFR Address: 0xC0 (bit addressable)

R	R	R/W	R/W	R	R	R/W	R/W	Reset Value
MASTER	TXMODE	STA	STO	ACKRQ	ARBLOST	ACK	SI	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

Bit 7: MASTER: SMBus Master/Slave Indicator.
This read-only bit indicates when the SMBus is operating as a master.
0: SMBus operating in Slave Mode.
1: SMBus operating in Master Mode.

Bit 6: TXMODE: SMBus Transmit Mode Indicator.
This read-only bit indicates when the SMBus is operating as a transmitter.
0: SMBus in Receiver Mode.
1: SMBus in Transmitter Mode.

Bit 5: STA: SMBus Start Flag.
Write:
0: No Start generated.
1: When operating as a master, a START condition is transmitted if the bus is free (If the bus is not free, the START is transmitted after a STOP is received or a timeout is detected). If STA is set by software as an active Master, a repeated START will be generated after the next ACK cycle.
Read:
0: No Start or repeated Start detected.
1: Start or repeated Start detected.

Bit 4: STO: SMBus Stop Flag.
Write:
0: No STOP condition is transmitted.
1: Setting STO to logic ‘1’ causes a STOP condition to be transmitted after the next ACK cycle. When the STOP condition is generated, hardware clears STO to logic ‘0’. If both STA and STO are set, a STOP condition is transmitted followed by a START condition.
Read:
0: No Stop condition detected.
1: Stop condition detected (if in Slave Mode) or pending (if in Master Mode).

Bit 3: ACKRQ: SMBus Acknowledge Request
This read-only bit is set to logic ‘1’ when the SMBus has received a byte and needs the ACK bit to be written with the correct ACK response value.

Bit 2: ARBLOST: SMBus Arbitration Lost Indicator.
This read-only bit is set to logic ‘1’ when the SMBus loses arbitration while operating as a transmitter. A lost arbitration while a slave indicates a bus error condition.

Bit 1: ACK: SMBus Acknowledge Flag.
This bit defines the out-going ACK level and records incoming ACK levels. It should be written each time a byte is received (when ACKRQ=1), or read after each byte is transmitted.
0: A "not acknowledge" has been received (if in Transmitter Mode) OR will be transmitted (if in Receiver Mode).
1: An "acknowledge" has been received (if in Transmitter Mode) OR will be transmitted (if in Receiver Mode).

Bit 0: SI: SMBus Interrupt Flag.
This bit is set by hardware under the conditions listed in Table 18.3. SI must be cleared by software. While SI is set, SCL is held low and the SMBus is stalled.

Table 18.3. Sources for Hardware Changes to SMB0CN

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

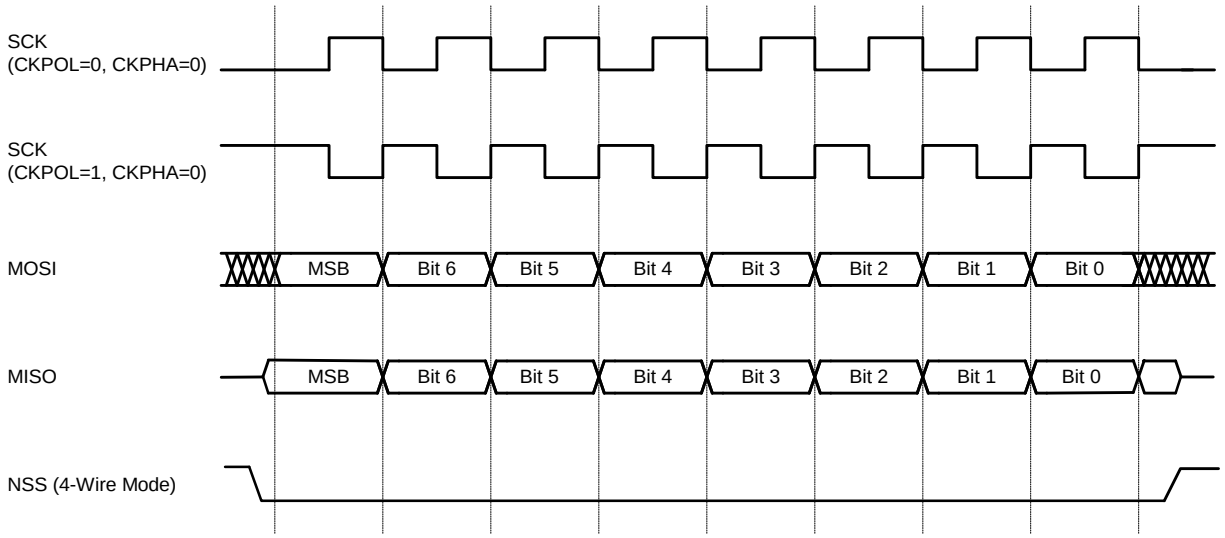

Figure 20.6. Slave Mode Data/Clock Timing (CKPHA = 0)

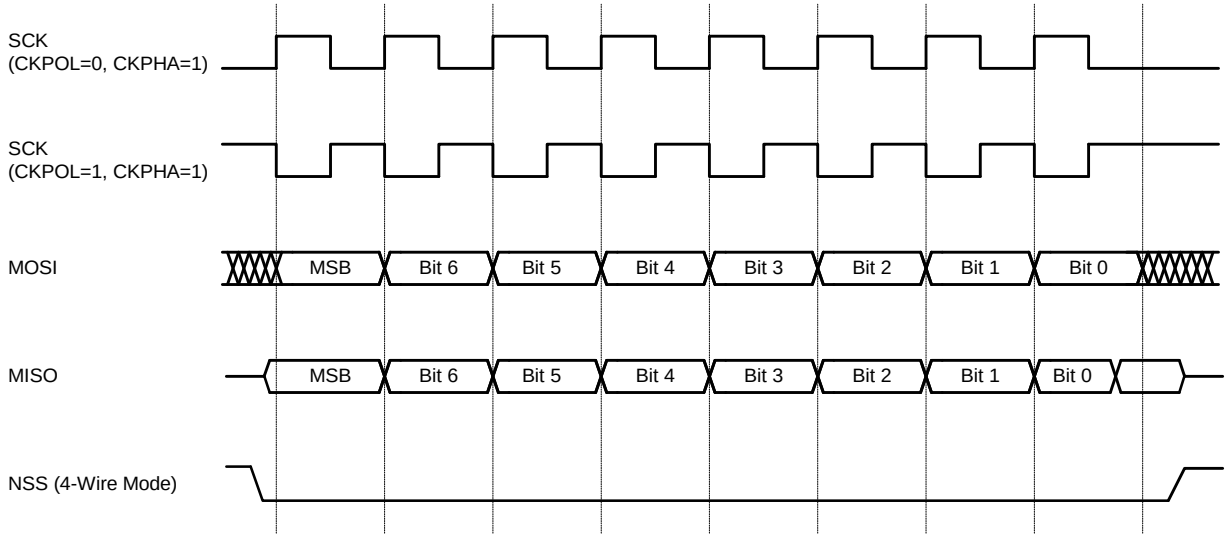

Figure 20.7. Slave Mode Data/Clock Timing (CKPHA = 1)

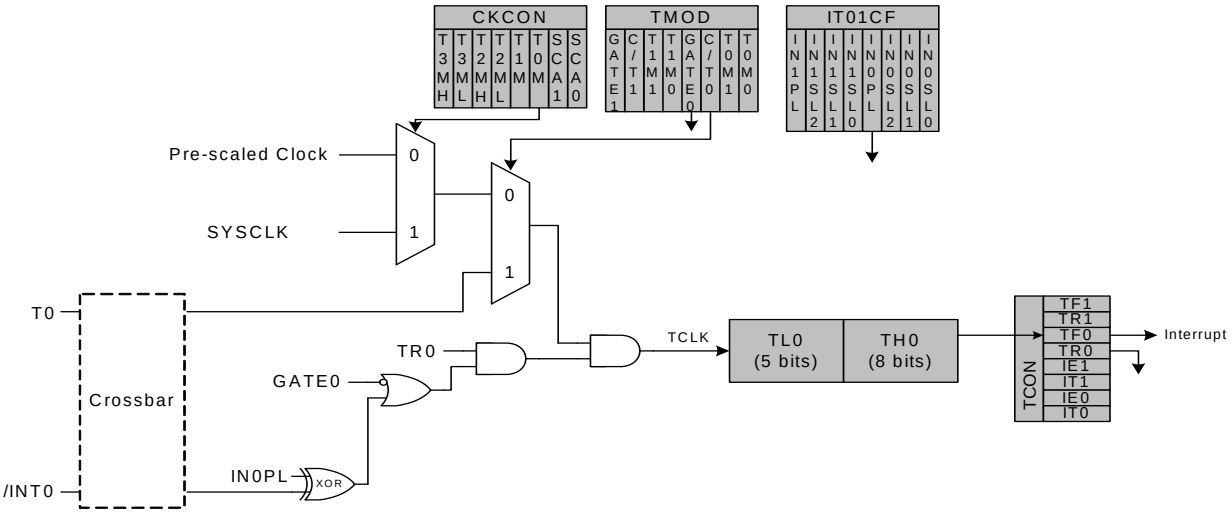

Figure 21.1. T0 Mode 0 Block Diagram

21.1.2. Mode 1: 16-bit Counter/Timer

Mode 1 operation is the same as Mode 0, except that the counter/timer registers use all 16 bits. The counter/timers are enabled and configured in Mode 1 in the same manner as for Mode 0.

21.1.3. Mode 2: 8-bit Counter/Timer with Auto-Reload

Mode 2 configures Timer 0 and Timer 1 to operate as 8-bit counter/timers with automatic reload of the start value. TL0 holds the count and TH0 holds the reload value. When the counter in TL0 overflows from all ones to 0x00, the timer overflow flag TF0 (TCON.5) is set and the counter in TL0 is reloaded from TH0. If Timer 0 interrupts are enabled, an interrupt will occur when the TF0 flag is set. The reload value in TH0 is not changed. TL0 must be initialized to the desired value before enabling the timer for the first count to be correct. When in Mode 2, Timer 1 operates identically to Timer 0.

Both counter/timers are enabled and configured in Mode 2 in the same manner as Mode 0. Setting the TR0 bit (TCON.4) enables the timer when either GATE0 (TMOD.3) is logic ‘0’ or when the input signal /INT0 is active as defined by bit IN0PL in register IT01CF (see Section “10.5. External Interrupts” on page 115 for details on the external input signals /INT0 and /INT1).

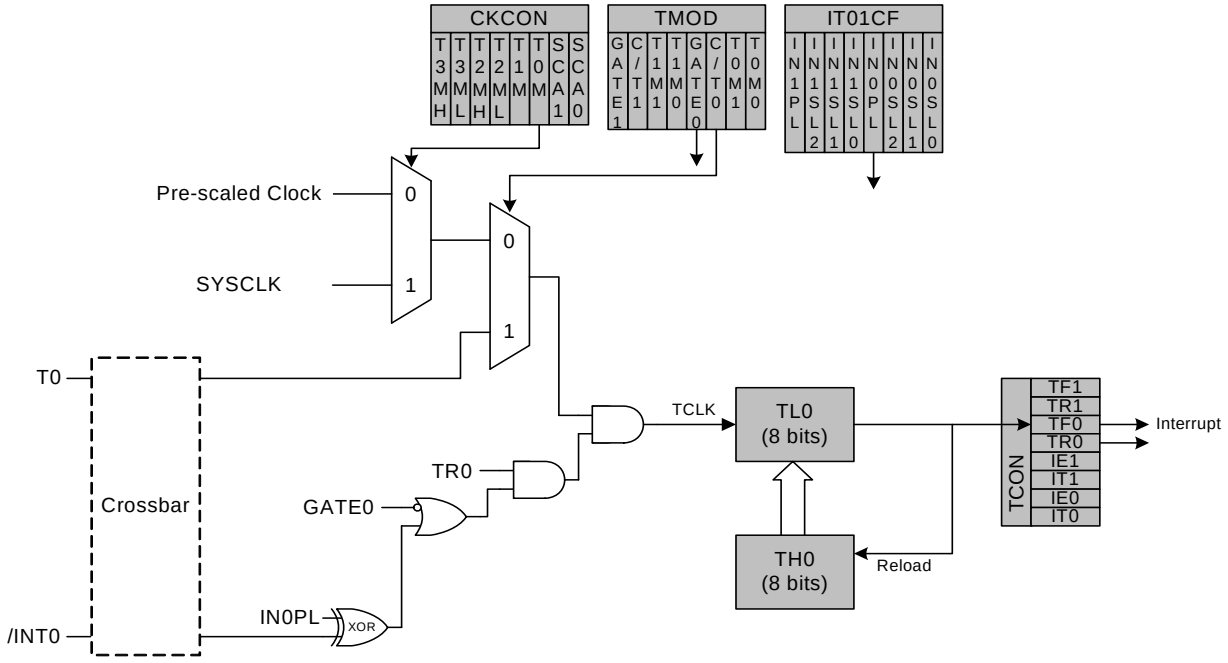

Figure 21.2. T0 Mode 2 Block Diagram

22.2.2. Software Timer (Compare) Mode

In Software Timer mode, the PCA0 counter/timer is compared to the module's 16-bit capture/compare register (PCA0CPHn and PCA0CPLn). When a match occurs, the Capture/Compare Flag (CCFn) in PCA0CN is set to logic ‘1’ and an interrupt request is generated if CCF interrupts are enabled. The CCFn bit is not automatically cleared by hardware when the CPU vectors to the interrupt service routine, and must be cleared by software. Setting the ECOMn and MATn bits in the PCA0CPMn register enables Software Timer mode.

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’.

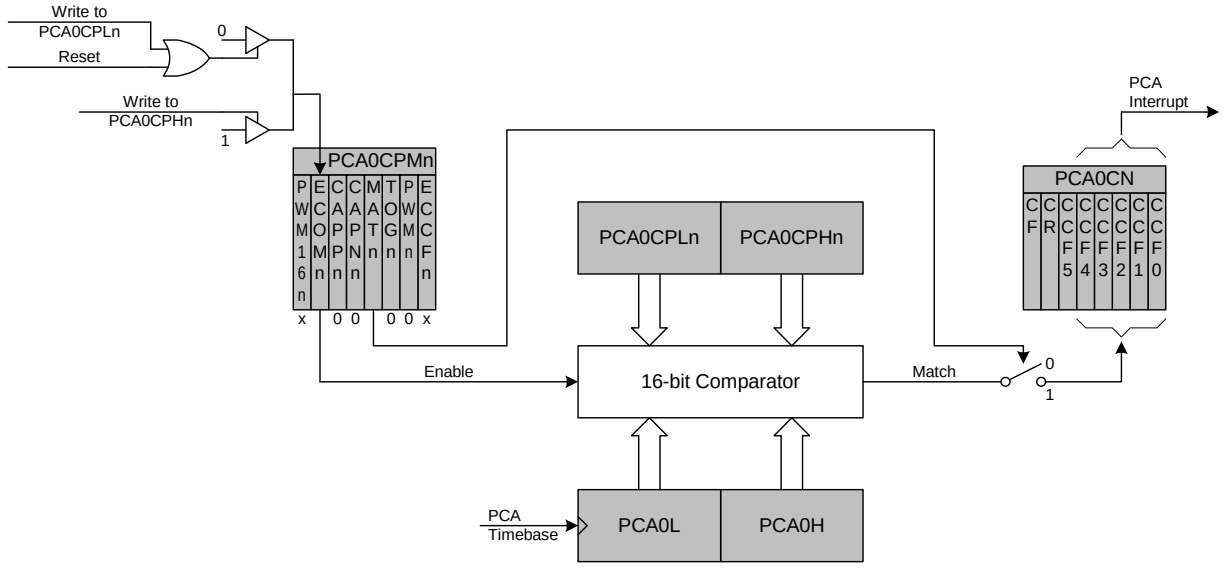

Figure 22.5. PCA Software Timer Mode Diagram

22.3.1. Watchdog Timer Operation

While the WDT is enabled:

- PCA counter is forced on.
- Writes to PCA0L and PCA0H are not allowed.
- PCA clock source bits (CPS2-CPS0) are frozen.
- PCA Idle control bit (CIDL) is frozen.
- Module 5 is forced into software timer mode.
- Writes to the Module 5 mode register (PCA0CPM5) are disabled.

While the WDT is enabled, writes to the CR bit will not change the PCA counter state; the counter will run until the WDT is disabled. The PCA counter run control (CR) will read zero if the WDT is enabled but user software has not enabled the PCA counter. If a match occurs between PCA0CPH5 and PCA0H while the WDT is enabled, a reset will be generated. To prevent a WDT reset, the WDT may be updated with a write of any value to PCA0CPH5. Upon a PCA0CPH5 write, PCA0H plus the offset held in PCA0CPL5 is loaded into PCA0CPH5 (See Figure 22.10).

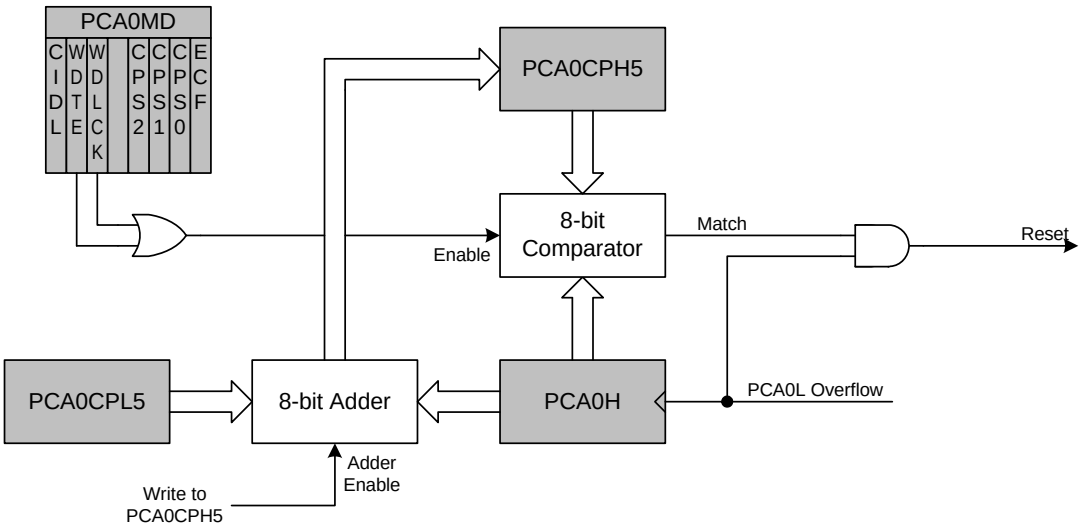

Figure 22.10. PCA Module 5 with Watchdog Timer Enabled

SFR Definition 22.3. PCA0CPMn: PCA0 Capture/Compare Mode

SFR Page: PCA0CPM0: all pages, PCA0CPM1: all pages, PCA0CPM2: all pages, PCA0CPM3: all pages, PCA0CPM4: all pages, PCA0CPM5: all pages

SFR Address: PCA0CPM0: 0xDA, PCA0CPM1: 0xDB, PCA0CPM2: 0xDC, PCA0CPM3: 0xDD, PCA0CPM4: 0xDE, PCA0CPM5: 0xDF

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
PWM16n	ECOMn	CAPPn	CAPNn	MATn	TOGn	PWMn	ECCFn	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

Bit 7: PWM16n: 16-bit Pulse Width Modulation Enable
This bit selects 16-bit mode when Pulse Width Modulation mode is enabled (PWMn = 1).
0: 8-bit PWM selected.
1: 16-bit PWM selected.

Bit 6: ECOMn: Comparator Function Enable.
This bit enables/disables the comparator function for PCA0 module n.
0: Disabled.
1: Enabled.

Bit 5: CAPPn: Capture Positive Function Enable.
This bit enables/disables the positive edge capture for PCA0 module n.
0: Disabled.
1: Enabled.

Bit 4: CAPNn: Capture Negative Function Enable.
This bit enables/disables the negative edge capture for PCA0 module n.
0: Disabled.
1: Enabled.

Bit 3: MATn: Match Function Enable.
This bit enables/disables the match function for PCA0 module n. When enabled, matches of the PCA0 counter with a module's capture/compare register cause the CCFn bit in PCA0MD register to be set to logic ‘1’.
0: Disabled.
1: Enabled.

Bit 2: TOGn: Toggle Function Enable.
This bit enables/disables the toggle function for PCA0 module n. When enabled, matches of the PCA0 counter with a module's capture/compare register cause the logic level on the CEXn pin to toggle. If the PWMn bit is also set to logic ‘1’, the module operates in Frequency Output Mode.
0: Disabled.
1: Enabled.

Bit 1: PWMn: Pulse Width Modulation Mode Enable.
This bit enables/disables the PWM function for PCA0 module n. When enabled, a pulse width modulated signal is output on the CEXn pin. 8-bit PWM is used if PWM16n is logic ‘0’; 16-bit mode is used if PWM16n logic ‘1’. If the TOGn bit is also set, the module operates in Frequency Output Mode.
0: Disabled.
1: Enabled.

Bit 0: ECCFn: Capture/Compare Flag Interrupt Enable.
This bit sets the masking of the Capture/Compare Flag (CCFn) interrupt.
0: Disable CCFn interrupts.
1: Enable a Capture/Compare Flag interrupt request when CCFn is set.