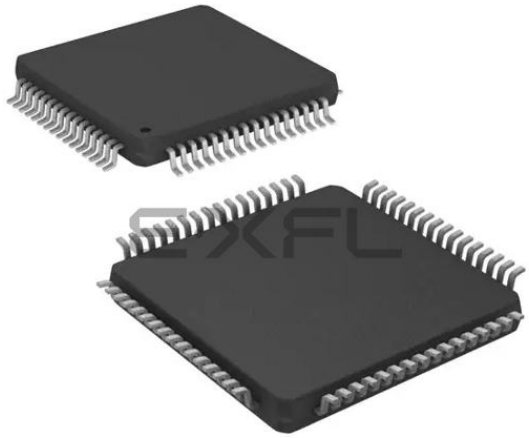


**Welcome to E-XFL.COM**

## What is "Embedded - Microcontrollers"?

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

## Applications of "Embedded - Microcontrollers"

| Details | |
|---|---|
| Product Status | Obsolete |
| Core Processor | 8051 |
| Core Size | 8-Bit |
| Speed | 100MHz |
| Connectivity | EBI/EMI, SMBus (2-Wire/I²C), SPI, UART/USART |
| Peripherals | Brown-out Detect/Reset, POR, PWM, Temp Sensor, WDT |
| Number of I/O | 32 |
| Program Memory Size | 128KB (128K x 8) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 8.25K x 8 |
| Voltage - Supply (Vcc/Vdd) | 3V ~ 3.6V |
| Data Converters | A/D 8x10b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 64-TQFP |
| Supplier Device Package | 64-TQFP (10x10) |
| Purchase URL | https://www.e-xfl.com/product-detail/silicon-labs/c8051f131 |

## 1.8. 12 or 10-Bit Analog to Digital Converter

All devices include either a 12 or 10-bit SAR ADC (ADC0) with a 9-channel input multiplexer and programmable gain amplifier. With a maximum throughput of 100 ksps, the 12 and 10-bit ADCs offer true 12-bit linearity with an INL of ±1LSB. The ADC0 voltage reference can be selected from an external VREF pin, or (on the C8051F12x devices) the DAC0 output. On the 100-pin TQFP devices, ADC0 has its own dedicated Voltage Reference input pin; on the 64-pin TQFP devices, the ADC0 shares a Voltage Reference input pin with the 8-bit ADC2. The on-chip voltage reference may generate the voltage reference for other system components or the on-chip ADCs via the VREF output pin.

The ADC is under full control of the CIP-51 microcontroller via its associated Special Function Registers. One input channel is tied to an internal temperature sensor, while the other eight channels are available externally. Each pair of the eight external input channels can be configured as either two single-ended inputs or a single differential input. The system controller can also put the ADC into shutdown mode to save power.

A programmable gain amplifier follows the analog multiplexer. The gain can be set in software from 0.5 to 16 in powers of 2. The gain stage can be especially useful when different ADC input channels have widely varied input voltage signals, or when it is necessary to "zoom in" on a signal with a large DC offset (in differential mode, a DAC could be used to provide the DC offset).

Conversions can be started in four ways; a software command, an overflow of Timer 2, an overflow of Timer 3, or an external signal input. This flexibility allows the start of conversion to be triggered by software events, external HW signals, or a periodic timer overflow signal. Conversion completions are indicated by a status bit and an interrupt (if enabled). The resulting 10 or 12-bit data word is latched into two SFRs upon completion of a conversion. The data can be right or left justified in these registers under software control.

Window Compare registers for the ADC data can be configured to interrupt the controller when ADC data is within or outside of a specified range. The ADC can monitor a key voltage continuously in background mode, but not interrupt the controller unless the converted data is within the specified window.

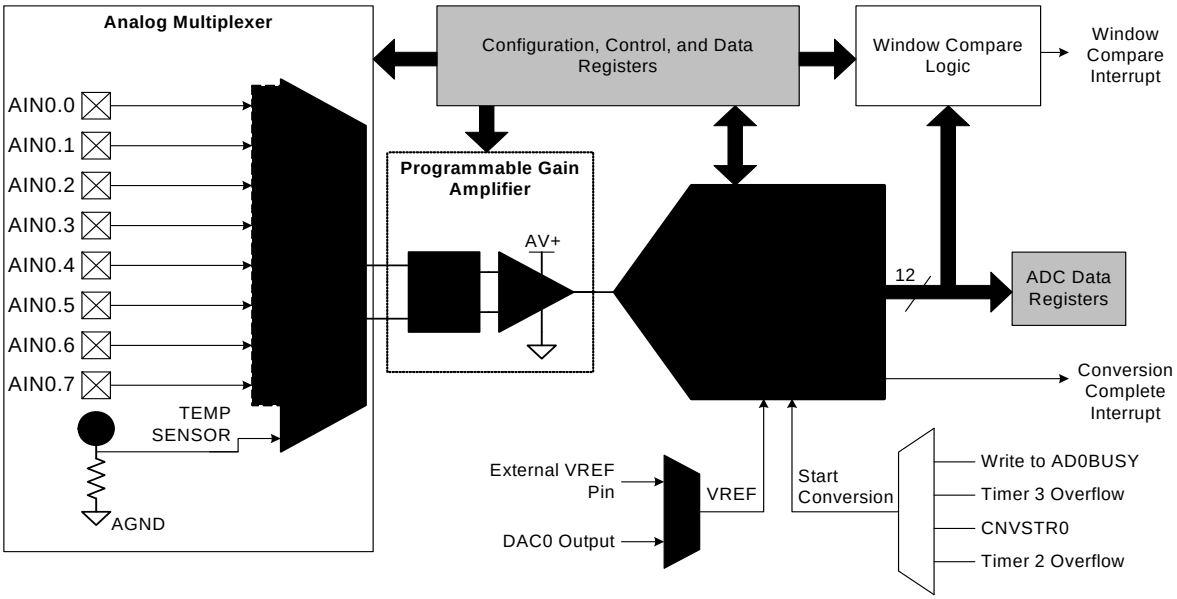



**Figure 1.13. 12-Bit ADC Block Diagram**

**Table 4.1. Pin Definitions (Continued)**

| Name | Pin Numbers | | | | Type | Description |
|---|---|---|---|---|---|---|
| | ‘F120 ‘F122 ‘F124 ‘F126 | ‘F121 ‘F123 ‘F125 ‘F127 | ‘F130 ‘F132 | ‘F131 ‘F133 | | |
| DAC1 | 99 | 63 | | | A Out | Digital to Analog Converter 1 Voltage Output. (See DAC Specification for complete description). |
| P0.0 | 62 | 55 | 62 | 55 | D I/O | Port 0.0. See Port Input/Output section for complete description. |
| P0.1 | 61 | 54 | 61 | 54 | D I/O | Port 0.1. See Port Input/Output section for complete description. |
| P0.2 | 60 | 53 | 60 | 53 | D I/O | Port 0.2. See Port Input/Output section for complete description. |
| P0.3 | 59 | 52 | 59 | 52 | D I/O | Port 0.3. See Port Input/Output section for complete description. |
| P0.4 | 58 | 51 | 58 | 51 | D I/O | Port 0.4. See Port Input/Output section for complete description. |
| ALE/P0.5 | 57 | 50 | 57 | 50 | D I/O | ALE Strobe for External Memory Address bus (multiplexed mode)<br>Port 0.5<br>See Port Input/Output section for complete description. |
| $\overline{RD}$/P0.6 | 56 | 49 | 56 | 49 | D I/O | /RD Strobe for External Memory Address bus<br>Port 0.6<br>See Port Input/Output section for complete description. |
| $\overline{WR}$/P0.7 | 55 | 48 | 55 | 48 | D I/O | /WR Strobe for External Memory Address bus<br>Port 0.7<br>See Port Input/Output section for complete description. |
| AIN2.0/A8/P1.0 | 36 | 29 | 36 | 29 | A In<br>D I/O | ADC2 Input Channel 0 (See ADC2 Specification for complete description).<br>Bit 8 External Memory Address bus (Non-multiplexed mode)<br>Port 1.0<br>See Port Input/Output section for complete description. |
| AIN2.1/A9/P1.1 | 35 | 28 | 35 | 28 | A In<br>D I/O | Port 1.1. See Port Input/Output section for complete description. |

The Temperature Sensor transfer function is shown in Figure 6.2. The output voltage ($V_{TEMP}$) is the PGA input when the Temperature Sensor is selected by bits AMX0AD3-0 in register AMX0SL; this voltage will be amplified by the PGA according to the user-programmed PGA settings. Typical values for the Slope and Offset parameters can be found in Table 6.1.

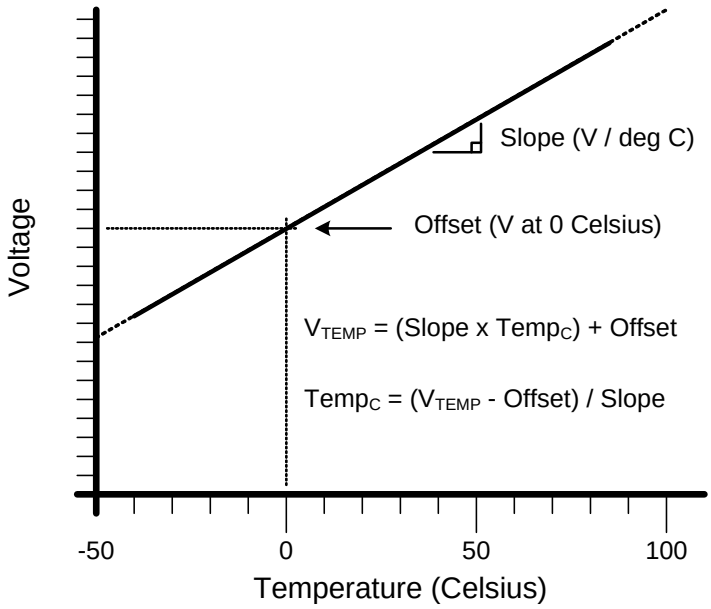



**Figure 6.2. Typical Temperature Sensor Transfer Function**

## 6.2. ADC Modes of Operation

ADC0 has a maximum conversion speed of 100 ksps. The ADC0 conversion clock is derived from the system clock divided by the value held in the ADCSC bits of register ADC0CF.

### 6.2.1. Starting a Conversion

A conversion can be initiated in one of four ways, depending on the programmed states of the ADC0 Start of Conversion Mode bits (AD0CM1, AD0CM0) in ADC0CN. Conversions may be initiated by:

1. Writing a ‘1’ to the AD0BUSY bit of ADC0CN;
2. A Timer 3 overflow (i.e. timed continuous conversions);
3. A rising edge detected on the external ADC convert start signal, CNVSTR0;
4. A Timer 2 overflow (i.e. timed continuous conversions).

The AD0BUSY bit is set to logic 1 during conversion and restored to logic 0 when conversion is complete. The falling edge of AD0BUSY triggers an interrupt (when enabled) and sets the AD0INT interrupt flag (ADC0CN.5). Converted data is available in the ADC0 data word MSB and LSB registers, ADC0H, ADC0L. Converted data can be either left or right justified in the ADC0H:ADC0L register pair (see example in Figure 6.5) depending on the programmed state of the AD0LJST bit in the ADC0CN register.

When initiating conversions by writing a ‘1’ to AD0BUSY, the AD0INT bit should be polled to determine when a conversion has completed (ADC0 interrupts may also be used). The recommended polling procedure is shown below.

Step 1. Write a ‘0’ to AD0INT;
Step 2. Write a ‘1’ to AD0BUSY;
Step 3. Poll AD0INT for ‘1’;
Step 4. Process ADC0 data.

When CNVSTR0 is used as a conversion start source, it must be enabled in the crossbar, and the corresponding pin must be set to open-drain, high-impedance mode (see **Section “18. Port Input/Output” on page 235** for more details on Port I/O configuration).

## SFR Definition 7.1. AMX2CF: AMUX2 Configuration

SFR Page: 2
SFR Address: 0xBA

| R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | Reset Value |
|---|---|---|---|---|---|---|---|---|
| - | - | - | - | PIN67IC | PIN45IC | PIN23IC | PIN01IC | 00000000 |
| Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | |

Bits7–4: UNUSED. Read = 0000b; Write = don't care.
Bit3: PIN67IC: AIN2.6, AIN2.7 Input Pair Configuration Bit.
0: AIN2.6 and AIN2.7 are independent single-ended inputs.
1: AIN2.6 and AIN2.7 are (respectively) +, – differential input pair.
Bit2: PIN45IC: AIN2.4, AIN2.5 Input Pair Configuration Bit.
0: AIN2.4 and AIN2.5 are independent single-ended inputs.
1: AIN2.4 and AIN2.5 are (respectively) +, – differential input pair.
Bit1: PIN23IC: AIN2.2, AIN2.3 Input Pair Configuration Bit.
0: AIN2.2 and AIN2.3 are independent single-ended inputs.
1: AIN2.2 and AIN2.3 are (respectively) +, – differential input pair.
Bit0: PIN01IC: AIN2.0, AIN2.1 Input Pair Configuration Bit.
0: AIN2.0 and AIN2.1 are independent single-ended inputs.
1: AIN2.0 and AIN2.1 are (respectively) +, – differential input pair.

**Note:** The ADC2 Data Word is in 2's complement format for channels configured as differential.

## SFR Definition 10.2. CPT0MD: Comparator0 Mode Selection

SFR Page: 1
SFR Address: 0x89

| R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | Reset Value |
|---|---|---|---|---|---|---|---|---|
| - | - | CP0RIE | CP0FIE | - | - | CP0MD1 | CP0MD0 | 00000010 |
| Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | |

Bits7–6: UNUSED. Read = 00b, Write = don't care.
Bit 5: CP0RIE: Comparator 0 Rising-Edge Interrupt Enable Bit.
0: Comparator 0 rising-edge interrupt disabled.
1: Comparator 0 rising-edge interrupt enabled.
Bit 4: CP0FIE: Comparator 0 Falling-Edge Interrupt Enable Bit.
0: Comparator 0 falling-edge interrupt disabled.
1: Comparator 0 falling-edge interrupt enabled.
Bits3–2: UNUSED. Read = 00b, Write = don't care.
Bits1–0: CP0MD1–CP0MD0: Comparator0 Mode Select
These bits select the response time for Comparator0.

| Mode | CP0MD1 | CP0MD0 | Notes |
|---|---|---|---|
| 0 | 0 | 0 | Fastest Response Time |
| 1 | 0 | 1 | — |
| 2 | 1 | 0 | — |
| 3 | 1 | 1 | Lowest Power Consumption |

## 11.3. Interrupt Handler

The CIP-51 includes an extended interrupt system supporting a total of 20 interrupt sources with two priority levels. The allocation of interrupt sources between on-chip peripherals and external input pins varies according to the specific version of the device. Each interrupt source has one or more associated interrupt-pending flag(s) located in an SFR. When a peripheral or external source meets a valid interrupt condition, the associated interrupt-pending flag is set to logic 1.

If interrupts are enabled for the source, an interrupt request is generated when the interrupt-pending flag is set. As soon as execution of the current instruction is complete, the CPU generates an LCALL to a predetermined address to begin execution of an interrupt service routine (ISR). Each ISR must end with an RETI instruction, which returns program execution to the next instruction that would have been executed if the interrupt request had not occurred. If interrupts are not enabled, the interrupt-pending flag is ignored by the hardware and program execution continues as normal. (The interrupt-pending flag is set to logic 1 regardless of the interrupt's enable/disable state.)

Each interrupt source can be individually enabled or disabled through the use of an associated interrupt enable bit in an SFR (IE, EIE1, or EIE2). However, interrupts must first be globally enabled by setting the EA bit (IE.7) to logic 1 before the individual interrupt enables are recognized. Setting the EA bit to logic 0 disables all interrupt sources regardless of the individual interrupt-enable settings.

**Note:** Any instruction that clears the EA bit should be immediately followed by an instruction that has two or more opcode bytes. For example:

```
// in 'C':
EA = 0; // clear EA bit.
EA = 0; // this is a dummy instruction with two-byte opcode.

; in assembly:
CLR EA ; clear EA bit.
CLR EA ; this is a dummy instruction with two-byte opcode.
```

If an interrupt is posted during the execution phase of a "CLR EA" opcode (or any instruction which clears the EA bit), and the instruction is followed by a single-cycle instruction, the interrupt may be taken. However, a read of the EA bit will return a '0' inside the interrupt service routine. When the "CLR EA" opcode is followed by a multi-cycle instruction, the interrupt will not be taken.

Some interrupt-pending flags are automatically cleared by the hardware when the CPU vectors to the ISR. However, most are not cleared by the hardware and must be cleared by software before returning from the ISR. If an interrupt-pending flag remains set after the CPU completes the return-from-interrupt (RETI) instruction, a new interrupt request will be generated immediately and the CPU will re-enter the ISR after the completion of the next instruction.

### 11.3.1. MCU Interrupt Sources and Vectors

The MCUs support 20 interrupt sources. Software can simulate an interrupt event by setting any interrupt-pending flag to logic 1. If interrupts are enabled for the flag, an interrupt request will be generated and the CPU will vector to the ISR address associated with the interrupt-pending flag. MCU interrupt sources, associated vector addresses, priority order and control bits are summarized in Table 11.4. Refer to the datasheet section associated with a particular on-chip peripheral for information regarding valid interrupt conditions for the peripheral and the behavior of its interrupt-pending flag(s).

## 12.3. Operating in Multiply and Accumulate Mode

MAC0 operates in Multiply and Accumulate (MAC) mode when the MAC0MS bit (MAC0CF.0) is cleared to '0'. When operating in MAC mode, MAC0 performs a 16-by-16 bit multiply on the contents of the MAC0A and MAC0B registers, and adds the result to the contents of the 40-bit MAC0 accumulator. Figure 12.4 shows the MAC0 pipeline. There are three stages in the pipeline, each of which takes exactly one SYSCLK cycle to complete. The MAC operation is initiated with a write to the MAC0BL register. After the MAC0BL register is written, MAC0A and MAC0B are multiplied on the first SYSCLK cycle. During the second stage of the MAC0 pipeline, the results of the multiplication are added to the current accumulator contents, and the result of the addition is stored in the MAC0 accumulator. The status flags in the MAC0STA register are set after the end of the second pipeline stage. During the second stage of the pipeline, the next multiplication can be initiated by writing to the MAC0BL register, if it is desired. The rounded (and optionally, saturated) result is available in the MAC0RNDH and MAC0RNDL registers at the end of the third pipeline stage. If the MAC0CA bit (MAC0CF.3) is set to '1' when the MAC operation is initiated, the accumulator and all MAC0STA flags will be cleared during the next cycle of the controller's clock (SYSCLK). The MAC0CA bit will clear itself to '0' when the clear operation is complete.

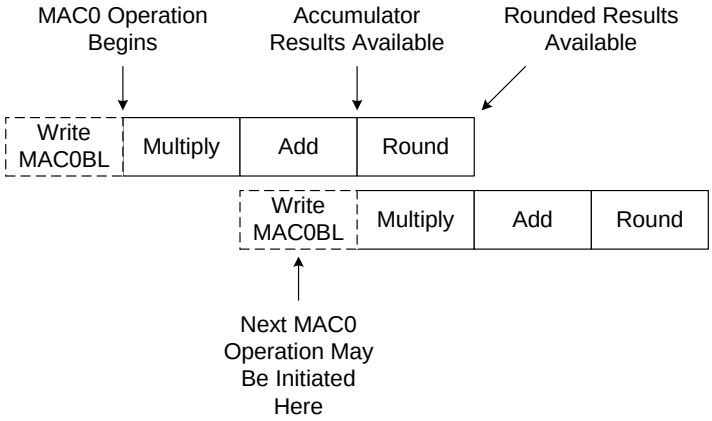



**Figure 12.4. MAC0 Pipeline**

## 12.4. Operating in Multiply Only Mode

MAC0 operates in Multiply Only mode when the MAC0MS bit (MAC0CF.0) is set to '1'. Multiply Only mode is identical to Multiply and Accumulate mode, except that the multiplication result is added with a value of zero before being stored in the MAC0 accumulator (i.e. it overwrites the current accumulator contents). The result of the multiplication is available in the MAC0 accumulator registers at the end of the second MAC0 pipeline stage (two SYSCLKs after writing to MAC0BL). As in MAC mode, the rounded result is available in the MAC0 Rounding Registers after the third pipeline stage. Note that in Multiply Only mode, the MAC0HO flag is not affected.

## 12.5. Accumulator Shift Operations

MAC0 contains a 1-bit arithmetic shift function which can be used to shift the contents of the 40-bit accumulator left or right by one bit. The accumulator shift is initiated by writing a '1' to the MAC0SC bit (MAC0CF.5), and takes one SYSCLK cycle (the rounded result is available in the MAC0 Rounding Registers after a second SYSCLK cycle, and MAC0SC is cleared to '0'). The direction of the arithmetic shift is controlled by the MAC0SD bit (MAC0CF.4). When this bit is cleared to '0', the MAC0 accumulator will shift left. When the MAC0SD bit is set to '1', the MAC0 accumulator will shift right. Right-shift operations are sign-extended with the current value of bit 39. Note that the status flags in the MAC0STA register are not affected by shift operations.

## SFR Definition 12.2. MAC0STA: MAC0 Status

| R | R | R | R | R/W | R/W | R/W | R/W | Reset Value |
|---|---|---|---|---|---|---|---|---|
| - | - | - | - | MAC0HO | MAC0Z | MAC0SO | MAC0N | 00000100 |
| Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | Bit Addressable |

SFR Address: 0xC0
SFR Page: 3

Bits 7–4: UNUSED: Read = 0000b, Write = don't care.

Bit 3: MAC0HO: Hard Overflow Flag.
This bit is set to '1' whenever an overflow out of the MAC0OVR register occurs during a MAC operation (i.e. when MAC0OVR changes from 0x7F to 0x80 or from 0x80 to 0x7F). The hard overflow flag must be cleared in software by directly writing it to '0', or by resetting the MAC logic using the MAC0CA bit in register MAC0CF.

Bit 2: MAC0Z: Zero Flag.
This bit is set to '1' if a MAC0 operation results in an Accumulator value of zero. If the result is non-zero, this bit will be cleared to '0'.

Bit 1: MAC0SO: Soft Overflow Flag.
This bit is set to '1' when a MAC operation causes an overflow into the sign bit (bit 31) of the MAC0 Accumulator. If the overflow condition is corrected after a subsequent MAC operation, this bit is cleared to '0'.

Bit 0: MAC0N: Negative Flag.
If the MAC Accumulator result is negative, this bit will be set to '1'. If the result is positive or zero, this flag will be cleared to '0'.

***Note:** The contents of this register should not be changed by software during the first two MAC0 pipeline stages.

## SFR Definition 12.3. MAC0AH: MAC0 A High Byte

| R | R | R | R | R | R | R | R | Reset Value |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | 00000000 |
| Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | |

SFR Address: 0xC2
SFR Page: 3

Bits 7–0: High Byte (bits 15–8) of MAC0 A Register.

### SFR Definition 12.10. MAC0ACC0: MAC0 Accumulator Byte 0

| R | R | R | R | R | R | R | R | Reset Value |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | 00000000 |
| Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | |

SFR Address: 0x93
SFR Page: 3

Bits 7–0: Byte 0 (bits 7–0) of MAC0 Accumulator.

***Note:** The contents of this register should not be changed by software during the first two MAC0 pipeline stages.

### SFR Definition 12.11. MAC0OVR: MAC0 Accumulator Overflow

| R | R | R | R | R | R | R | R | Reset Value |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | 00000000 |
| Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | |

SFR Address: 0x97
SFR Page: 3

Bits 7–0: MAC0 Accumulator Overflow Bits (bits 39–32).

***Note:** The contents of this register should not be changed by software during the first two MAC0 pipeline stages.

### SFR Definition 12.12. MAC0RNDH: MAC0 Rounding Register High Byte

| R | R | R | R | R | R | R | R | Reset Value |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | 00000000 |
| Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | |

SFR Address: 0xCF
SFR Page: 3

Bits 7–0: High Byte (bits 15–8) of MAC0 Rounding Register.

Electrical specifications for the precision internal oscillator are given in Table 14.1. Note that the system clock may be derived from the programmed internal oscillator divided by 1, 2, 4, or 8, as defined by the IFCN bits in register OSCICN.

### SFR Definition 14.1. OSCICL: Internal Oscillator Calibration.

| R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | Reset Value |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | Variable |
| Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | |

SFR Address: 0x8B
SFR Page: F

Bits 7–0: OSCICL: Internal Oscillator Calibration Register.
This register calibrates the internal oscillator period. The reset value for OSCICL defines the internal oscillator base frequency. The reset value is factory calibrated to generate an internal oscillator frequency of 24.5 MHz.

### SFR Definition 14.2. OSCICN: Internal Oscillator Control

| R/W | R | R/W | R | R/W | R/W | R/W | R/W | Reset Value |
|---|---|---|---|---|---|---|---|---|
| IOSCEN | IFRDY | - | - | - | - | IFCN1 | IFCN0 | 11000000 |
| Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | |

SFR Address: 0x8A
SFR Page: F

Bit 7: IOSCEN: Internal Oscillator Enable Bit.
0: Internal Oscillator Disabled.
1: Internal Oscillator Enabled.
Bit 6: IFRDY: Internal Oscillator Frequency Ready Flag.
0: Internal Oscillator not running at programmed frequency.
1: Internal Oscillator running at programmed frequency.
Bits 5–2: Reserved.
Bits 1–0: IFCN1-0: Internal Oscillator Frequency Control Bits.
00: Internal Oscillator is divided by 8.
01: Internal Oscillator is divided by 4.
10: Internal Oscillator is divided by 2.
11: Internal Oscillator is divided by 1.

## 15.2. Security Options

The CIP-51 provides security options to protect the Flash memory from inadvertent modification by software as well as prevent the viewing of proprietary program code and constants. The Program Store Write Enable (PSCTL.0), Program Store Erase Enable (PSCTL.1), and Flash Write/Erase Enable (FLACL.0) bits protect the Flash memory from accidental modification by software. These bits must be explicitly set to logic 1 before software can write or erase the Flash memory. Additional security features prevent proprietary program code and data constants from being read or altered across the JTAG interface or by software running on the system controller.

A set of security lock bytes protect the Flash program memory from being read or altered across the JTAG interface. Each bit in a security lock-byte protects one 16k-byte block of memory. Clearing a bit to logic 0 in the Read Lock Byte prevents the corresponding block of Flash memory from being read across the JTAG interface. Clearing a bit in the Write/Erase Lock Byte protects the block from JTAG erasures and/or writes. The Scratchpad area is read or write/erase locked when all bits in the corresponding security byte are cleared to logic 0.

On the C8051F12x and C8051F130/1, the security lock bytes are located at 0x1FBFE (Write/Erase Lock) and 0x1FBFF (Read Lock), as shown in Figure 15.2. On the C8051F132/3, the security lock bytes are located at 0x0FFFE (Write/Erase Lock) and 0x0FFFF (Read Lock), as shown in Figure 15.3. The 1024-byte sector containing the lock bytes can be written to, but not erased, by software. An attempted read of a read-locked byte returns undefined data. Debugging code in a read-locked sector is not possible through the JTAG interface. The lock bits can always be read from and written to logic 0 regardless of the security setting applied to the block containing the security bytes. This allows additional blocks to be protected after the block containing the security bytes has been locked.

**Important Note: To ensure protection from external access, the block containing the lock bytes must be Write/Erase locked. On the 128 kB devices (C8051F12x and C8051F130/1), the block containing the security bytes is 0x18000-0x1BFFF, and is locked by clearing bit 7 of the Write/Erase Lock Byte. On the 64 kB devices (C8051F132/3), the block containing the security bytes is 0x0C000-0x0FFFF, and is locked by clearing bit 3 of the Write/Erase Lock Byte. If the page containing the security bytes is not Write/Erase locked, it is still possible to erase this page of Flash memory through the JTAG port and reset the security bytes.**

**When the page containing the security bytes has been Write/Erase locked, a JTAG full device erase must be performed to unlock any areas of Flash protected by the security bytes. A JTAG full device erase is initiated by performing a normal JTAG erase operation on either of the security byte locations. This operation must be initiated through the JTAG port, and cannot be performed from firmware running on the device.**

The Flash Access Limit security feature (see SFR Definition 15.1) protects proprietary program code and data from being read by software running on the device. This feature provides support for OEMs that wish to program the MCU with proprietary value-added firmware before distribution. The value-added firmware can be protected while allowing additional code to be programmed in remaining program memory space later.

The Flash Access Limit (FAL) is a 17-bit address that establishes two logical partitions in the program memory space. The first is an upper partition consisting of all the program memory locations at or above the FAL address, and the second is a lower partition consisting of all the program memory locations starting at 0x00000 up to (but excluding) the FAL address. Software in the upper partition can execute code in the lower partition, but is prohibited from reading locations in the lower partition using the MOVC instruction. (Executing a MOVC instruction from the upper partition with a source address in the lower partition will return indeterminate data.) Software running in the lower partition can access locations in both the upper and lower partition without restriction.

The Value-added firmware should be placed in the lower partition. On reset, control is passed to the value-added firmware via the reset vector. Once the value-added firmware completes its initial execution, it branches to a predetermined location in the upper partition. If entry points are published, software running in the upper partition may execute program code in the lower partition, but it cannot read or change the contents of the lower partition. Parameters may be passed to the program code running in the lower partition either through the typical method of placing them on the stack or in registers before the call or by placing them in prescribed memory locations in the upper partition.

The FAL address is specified using the contents of the Flash Access Limit Register. The 8 MSBs of the 17-bit FAL address are determined by the setting of the FLACL register. Thus, the FAL can be located on 512-byte boundaries anywhere in program memory space. However, the 1024-byte erase sector size essentially requires that a 1024 boundary be used. The contents of a non-initialized FLACL security byte are 0x00, thereby setting the FAL address to 0x00000 and allowing read access to all locations in program memory space by default.

### SFR Definition 15.1. FLACL: Flash Access Limit

| 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: |

SFR Address: 0xB7
SFR Page: F

Bits 7–0: FLACL: Flash Access Limit.
This register holds the most significant 8 bits of the 17-bit program memory read/write/erase limit address. The lower 9 bits of the read/write/erase limit are always set to 0. A write to this register sets the Flash Access Limit. This register can only be written once after any reset. Any subsequent writes are ignored until the next reset. **To fully protect all addresses below this limit, bit 0 of FLACL should be set to '0' to align the FAL on a 1024-byte Flash page boundary.**

5. Select the memory mode (on-chip only, split mode without bank select, split mode with bank select, or off-chip only).
6. 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 17.2.

## 17.3. Port Selection and Configuration

The External Memory Interface can appear on Ports 3, 2, 1, and 0 (All Devices) or on Ports 7, 6, 5, and 4 (100-pin TQFP devices only), depending on the state of the PRTSEL bit (EMI0CF.5). If the lower Ports are selected, the EMIFLE bit (XBR2.1) must be set to a '1' so that the Crossbar will skip over P0.7 (/WR), P0.6 (/RD), and if multiplexed mode is selected P0.5 (ALE). For more information about the configuring the Crossbar, see **Section "18.1. Ports 0 through 3 and the Priority Crossbar Decoder" on page 238**.

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 (on Ports 3, 2, 1, and 0). See **Section "18. Port Input/ Output" on page 235** 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. See"Configuring the Output Modes of the Port Pins" on page 239.

### SFR Definition 17.1. EMI0CN: External Memory Interface Control

| R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | Reset Value |
|---|---|---|---|---|---|---|---|---|
| PGSEL7 | PGSEL6 | PGSEL5 | PGSEL4 | PGSEL3 | PGSEL2 | PGSEL1 | PGSEL0 | 00000000 |
| Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | |

SFR Address: 0xA2
SFR Page: 0

Bits7–0: PGSEL[7:0]: XRAM Page Select Bits.
The XRAM Page Select Bits provide the high byte of the 16-bit external data memory address when using an 8-bit MOVX command, effectively selecting a 256-byte page of RAM.
0x00: 0x0000 to 0x00FF
0x01: 0x0100 to 0x01FF
...
0xFE: 0xFE00 to 0xFEFF
0xFF: 0xFF00 to 0xFFFF

## SFR Definition 17.3. EMI0TC: External Memory Timing Control

| R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | Reset Value |
|---|---|---|---|---|---|---|---|---|
| EAS1 | EAS0 | ERW3 | EWR2 | EWR1 | EWR0 | EAH1 | EAH0 | 11111111 |
| Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | |

SFR Address: 0xA1
SFR Page: 0

Bits7–6: EAS1–0: EMIF Address Setup Time Bits.
00: Address setup time = 0 SYSCLK cycles.
01: Address setup time = 1 SYSCLK cycle.
10: Address setup time = 2 SYSCLK cycles.
11: Address setup time = 3 SYSCLK cycles.

Bits5–2: EWR3–0: EMIF /WR and /RD Pulse-Width Control Bits.
0000: /WR and /RD pulse width = 1 SYSCLK cycle.
0001: /WR and /RD pulse width = 2 SYSCLK cycles.
0010: /WR and /RD pulse width = 3 SYSCLK cycles.
0011: /WR and /RD pulse width = 4 SYSCLK cycles.
0100: /WR and /RD pulse width = 5 SYSCLK cycles.
0101: /WR and /RD pulse width = 6 SYSCLK cycles.
0110: /WR and /RD pulse width = 7 SYSCLK cycles.
0111: /WR and /RD pulse width = 8 SYSCLK cycles.
1000: /WR and /RD pulse width = 9 SYSCLK cycles.
1001: /WR and /RD pulse width = 10 SYSCLK cycles.
1010: /WR and /RD pulse width = 11 SYSCLK cycles.
1011: /WR and /RD pulse width = 12 SYSCLK cycles.
1100: /WR and /RD pulse width = 13 SYSCLK cycles.
1101: /WR and /RD pulse width = 14 SYSCLK cycles.
1110: /WR and /RD pulse width = 15 SYSCLK cycles.
1111: /WR and /RD pulse width = 16 SYSCLK cycles.

Bits1–0: EAH1–0: EMIF Address Hold Time Bits.
00: Address hold time = 0 SYSCLK cycles.
01: Address hold time = 1 SYSCLK cycle.
10: Address hold time = 2 SYSCLK cycles.
11: Address hold time = 3 SYSCLK cycles.

### SFR Definition 18.13. P4: Port4 Data

| R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | Reset Value |
|---|---|---|---|---|---|---|---|---|
| P4.7 | P4.6 | P4.5 | P4.4 | P4.3 | P4.2 | P4.1 | P4.0 | 11111111 |
| Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | Bit Addressable |

SFR Address: 0xC8
SFR Page: F

Bits7–0: P4.[7:0]: Port4 Output Latch Bits.
Write - Output appears on I/O pins.
0: Logic Low Output.
1: Logic High Output (Open-Drain if corresponding P4MDOUT.n bit = 0). See SFR Definition 18.14.
Read - Returns states of I/O pins.
0: P4.n pin is logic low.
1: P4.n pin is logic high.

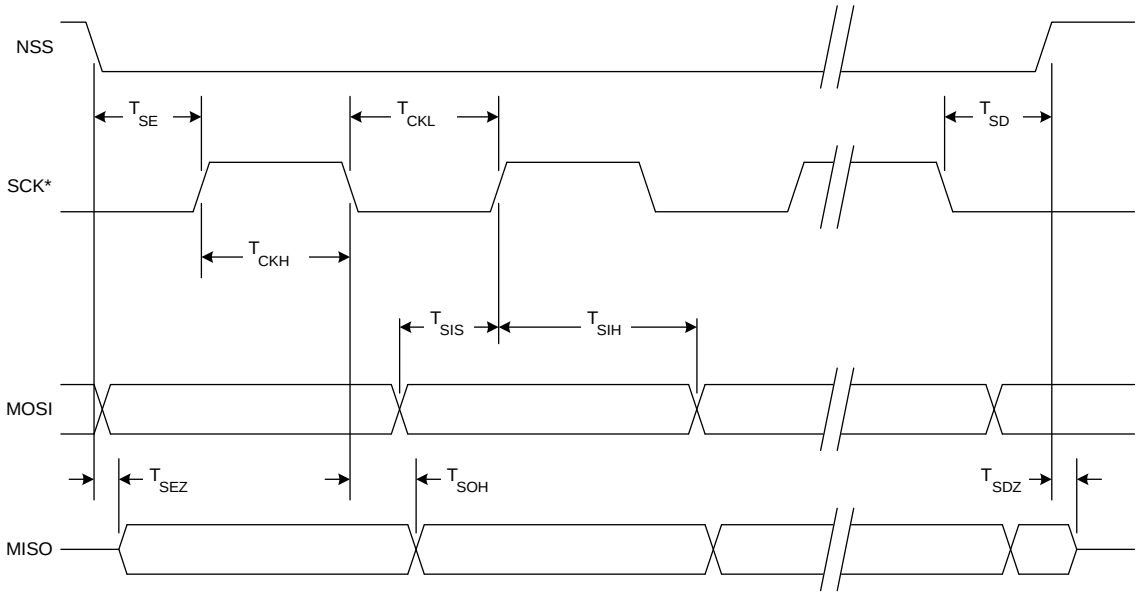
**Note:** P4.7 (/WR), P4.6 (/RD), and P4.5 (ALE) can be driven by the External Data Memory Interface. See **Section “17. External Data Memory Interface and On-Chip XRAM” on page 219** for more information.

### SFR Definition 18.14. P4MDOUT: Port4 Output Mode

| 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 | |

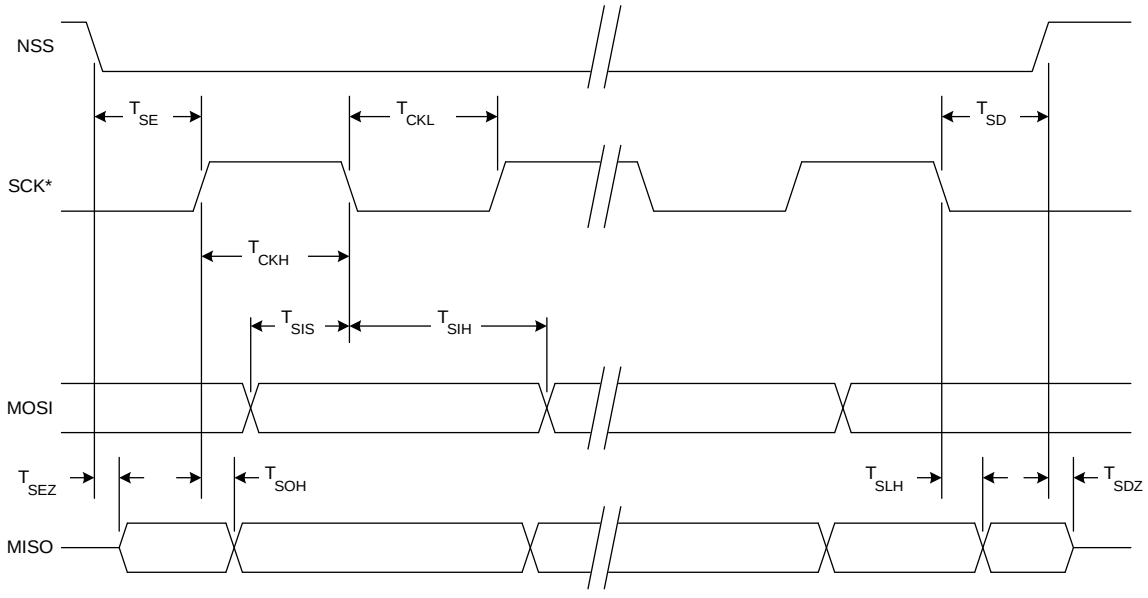
SFR Address: 0x9C
SFR Page: F

Bits7–0: P4MDOUT.[7:0]: Port4 Output Mode Bits.
0: Port Pin output mode is configured as Open-Drain.
1: Port Pin output mode is configured as Push-Pull.




* SCK is shown for CKPOL = 0. SCK is the opposite polarity for CKPOL = 1.

**Figure 20.10. SPI Slave Timing (CKPHA = 0)**




* SCK is shown for CKPOL = 0. SCK is the opposite polarity for CKPOL = 1.

**Figure 20.11. SPI Slave Timing (CKPHA = 1)**

### 24.2.5. 8-Bit Pulse Width Modulator Mode

Each module can be used independently to generate pulse width modulated (PWM) outputs on its associated CEXn pin. The frequency of the output is dependent on the timebase for the PCA0 counter/timer. The duty cycle of the PWM output signal is varied using the module's PCA0CPLn capture/compare register. When the value in the low byte of the PCA0 counter/timer (PCA0L) is equal to the value in PCA0CPLn, the output on the CEXn pin will be high. When the count value in PCA0L overflows, the CEXn output will be low (see Figure 24.8). Also, when the counter/timer low byte (PCA0L) overflows from 0xFF to 0x00, PCA0CPLn is reloaded automatically with the value stored in the counter/timer's high byte (PCA0H) 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 24.2.

**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’.

**Equation 24.2. 8-Bit PWM Duty Cycle**

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

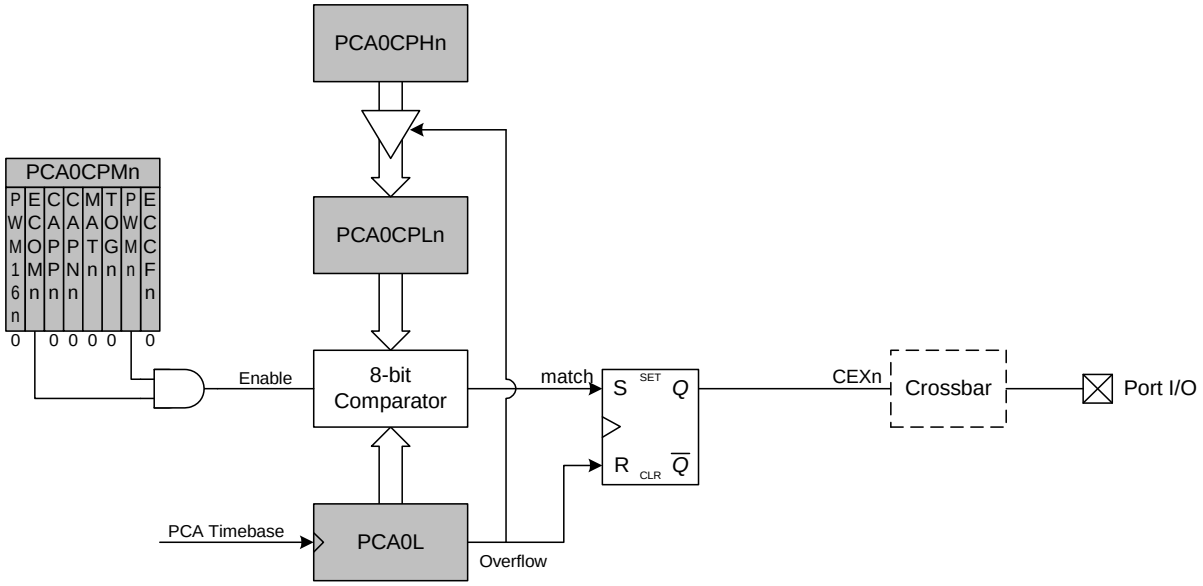



**Figure 24.8. PCA 8-Bit PWM Mode Diagram**

# DOCUMENT CHANGE LIST

## Revision 1.3 to Revision 1.4

- Added new paragraph tags: SFR Definition and JTAG Register Definition.
- Product Selection Guide Table 1.1: Added RoHS-compliant ordering information.
- Overview Chapter, Figure 1.8, “On-Chip Memory Map”: Corrected on-chip XRAM size to “8192 Bytes”.
- SAR8 Chapter: Table 7.1, “ADC2 Electrical Characteristics”: Track/Hold minimum spec corrected to “300 ns”.
- SAR8 Chapter: Table 7.1, “ADC2 Electrical Characteristics”: Total Harmonic Distortion typical spec corrected to “-51 dB”.
- Oscillators Chapter, Figure 14.1, “Oscillator Diagram”: Corrected location of IOSCEN arrow.
- CIP51 Chapter, **Section 11.3**: Added note describing EA change behavior when followed by single-cycle instruction.
- CIP51 Chapter, Interrupt Summary Table: Added “SFRPAGE” column and SFRPAGE value for each interrupt source.
- CIP-51 Chapter, Figure 11.2, “Memory Map”: Corrected on-chip XRAM size to “8192 Bytes”.
- Port I/O Chapter, Crossbar Priority Figures: Character formatting problem corrected.
- Port I/O Chapter, P7MDOUT Register Description: Removed references to UART and SMBus peripherals.
- Port I/O Chapter, P3MDOUT Register Description: Corrected text to read “P3MDOUT.[7:0]”.
- Timers Chapter: References to “TnCON” corrected to read “TMRnCN”.
- PCA0 Chapter, Section 24.1: Added note about PCA0CN Register and effects of read-modify-write instructions on the CF bit.