Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	32MHz
Connectivity	I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	18
Program Memory Size	28KB (16K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	2.3V ~ 5.5V
Data Converters	A/D 17x12b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SOIC (0.295", 7.50mm Width)
Supplier Device Package	20-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f18446-e-so

    - Multiple gate modes
    - Time base for capture/compare function
  - Timer2/4/6 with Hardware Limit Timer:
    - 8-bit timers
    - Programmable prescaler/postscaler
    - Time base for PWM function
    - Hardware Limit (HLT) and one-shot extensions
    - Selectable clock sources
  - Signal Measurement Timer (SMT)
    - 1 SMT(s)
    - 24-bit timer/counter with programmable prescaler

Analog Peripherals

- Analog-to-Digital Converter with Computation (ADC^2):
  - 12-bit with up to 17 external channels
  - Conversion available during Sleep
  - Automated post-processing
  - Automated math functions on input signals:
    - Averaging, filter calculations, oversampling and threshold comparison
  - Integrated charge pump for low-voltage operation
  - CVD support
- Zero-Cross Detect (ZCD):
  - AC high voltage zero-crossing detection for simplifying TRIAC control
  - Synchronized switching control and timing
- Temperature Sensor Circuit
- Comparator:
  - 2 Comparators
  - Fixed Voltage Reference at (non)inverting input(s)
  - Comparator outputs externally accessible
- Digital-to-Analog Converter (DAC):
  - 5-bit resolution, rail-to-rail
  - Positive Reference Selection
  - Unbuffered I/O pin output
  - Internal connections to ADCs and comparators
- Fixed Voltage Reference (FVR) module:
  - 1.024V, 2.048V and 4.096V output levels

Flexible Oscillator Structure

- High-Precision Internal Oscillator:
  - Software-selectable frequency range up to 32 MHz
  - ±2% at calibration (nominal)
- 4x PLL for use with external sources
  - up to 32 MHz (4-8 MHz input)

1.4.2.1 Short Bit Names

Short bit names are an abbreviation for the bit function. For example, some peripherals are enabled with the EN bit. The bit names shown in the registers are the short name variant.

Short bit names are useful when accessing bits in C programs. The general format for accessing bits by the short name is RegisterNamebits.ShortName. For example, the enable bit, EN, in the CM1CON0 register can be set in C programs with the instruction `CM1CON0bits.EN = 1`.

Short names are generally not useful in assembly programs because the same name may be used by different peripherals in different bit positions. When this occurs, during the include file generation, all instances of that short bit name are appended with an underscore plus the name of the register in which the bit resides to avoid naming contentions.

1.4.2.2 Long Bit Names

Long bit names are constructed by adding a peripheral abbreviation prefix to the short name. The prefix is unique to the peripheral thereby making every long bit name unique. The long bit name for the COG1 enable bit is the COG1 prefix, G1, appended with the enable bit short name, EN, resulting in the unique bit name G1EN.

Long bit names are useful in both C and assembly programs. For example, in C the COG1CON0 enable bit can be set with the `G1EN = 1` instruction. In assembly, this bit can be set with the `BSF COG1CON0,G1EN` instruction.

1.4.2.3 Bit Fields

Bit fields are two or more adjacent bits in the same register. Bit fields adhere only to the short bit naming convention. For example, the three Least Significant bits of the COG1CON0 register contain the mode control bits. The short name for this field is MD. There is no long bit name variant. Bit field access is only possible in C programs. The following example demonstrates a C program instruction for setting the COG1 to the Push-Pull mode:

COG1CON0bits.MD = 0x5;

Individual bits in a bit field can also be accessed with long and short bit names. Each bit is the field name appended with the number of the bit position within the field. For example, the Most Significant mode bit has the short bit name MD2 and the long bit name is G1MD2. The following two examples demonstrate assembly program sequences for setting the COG1 to Push-Pull mode:

Example 1:

```
MOVLW  ~(1<<G1MD1)
ANDWF  COG1CON0,F
MOVLW  1<<G1MD2 | 1<<G1MD0
IORWF  COG1CON0,F
```

Example 2:

```
BSF    COG1CON0,G1MD2
BCF    COG1CON0,G1MD1
BSF    COG1CON0,G1MD0
```

1.4.3 Register and Bit Naming Exceptions

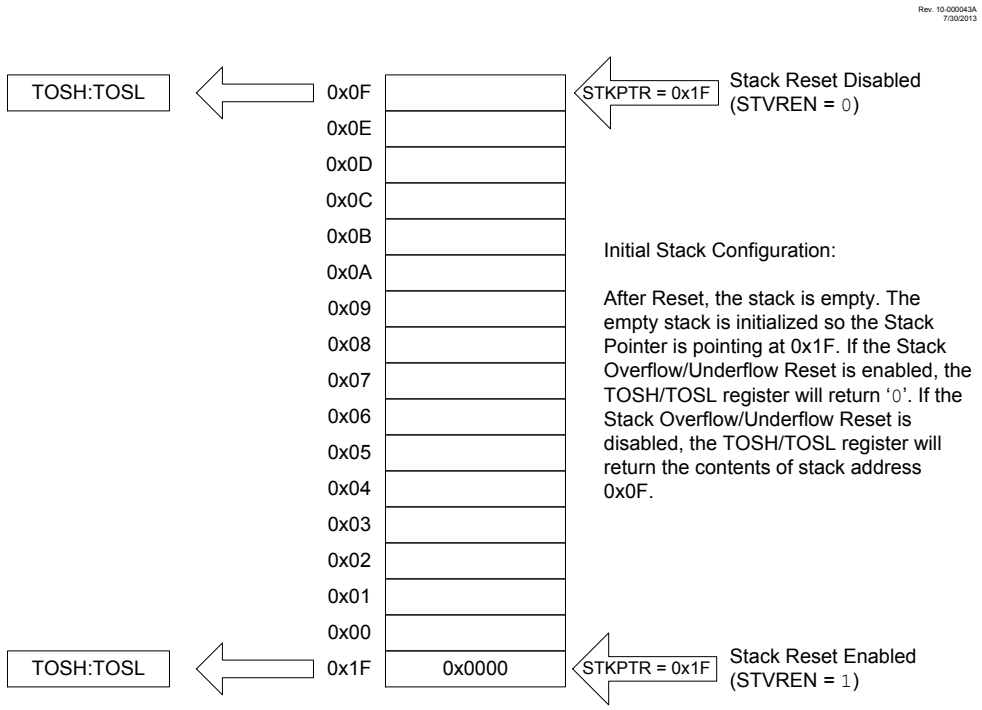
1.4.3.1 Status, Interrupt, and Mirror Bits

Status, interrupt enables, interrupt flags, and mirror bits are contained in registers that span more than one peripheral. In these cases, the bit name shown is unique so there is no prefix or short name variant.

memory left at any given time. The STKPTR always points at the currently used place on the stack. Therefore, a CALL or CALLW will increment the STKPTR and then write the PC, and a return will unload the PC value from the stack and then decrement the STKPTR.

Reference the following figures for examples of accessing the stack.

Figure 7-4. Accessing the Stack Example 1



7.10.3 BSR_SHAD

Name: BSR_SHAD
Address: 0x1FE6

Shadow of Bank Select Register

The BSR indicates the data memory bank by writing the bank number into the register. All data memory can be accessed directly via instructions, or indirectly via FSRs.

Bit	7	6	5	4	3	2	1	0
			BSR[5:0]					
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			x	x	x	x	x	x

Bits 5:0 – BSR[5:0]
Six Most Significant bits of the data memory address
Reset States: POR/BOR = xxxxxx
All Other Resets = uuuuuu

8. Resets

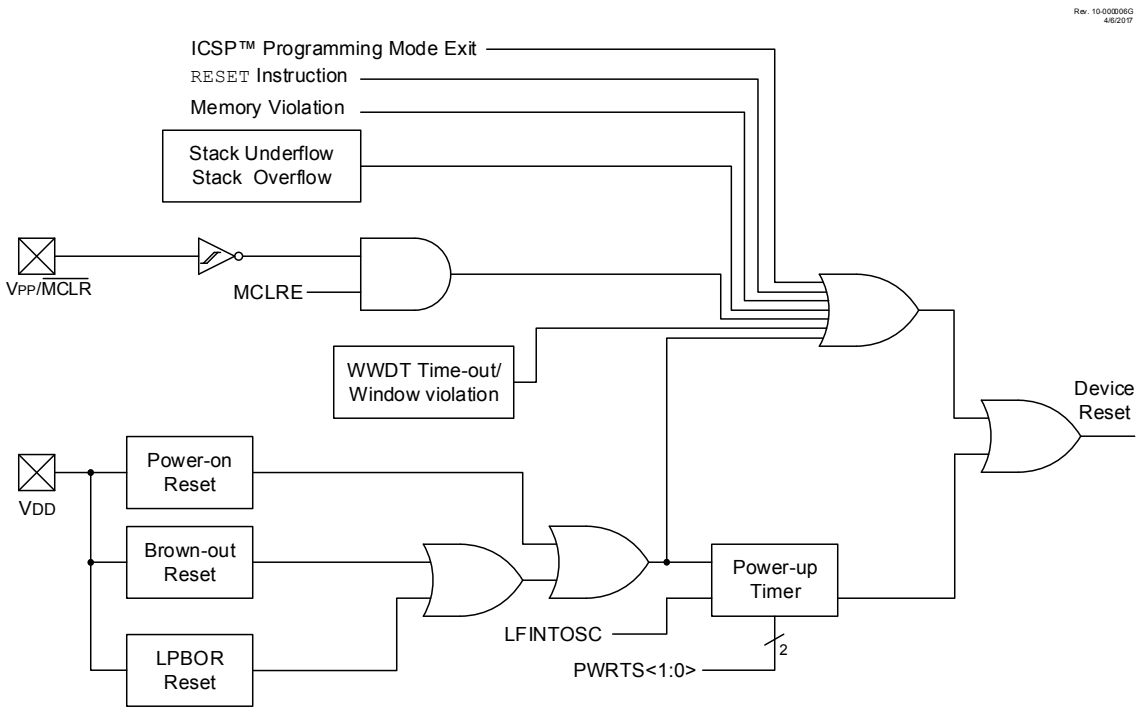
There are multiple ways to reset this device:

- Power-on Reset (POR)
- Brown-out Reset (BOR)
- Low-Power Brown-Out Reset (LPBOR)
- $\overline{\text{MCLR}}$ Reset
- WDT Reset
- `RESET` instruction
- Stack Overflow
- Stack Underflow
- Programming mode exit

To allow V_{DD} to stabilize, an optional Power-up Timer can be enabled to extend the Reset time after a BOR or POR event.

A simplified block diagram of the On-Chip Reset Circuit is shown in the block diagram below.

Figure 8-1. Simplified Block Diagram of On-Chip Reset Circuit



Note: See *"BOR Operating Conditions"* table for BOR active conditions.

Related Links

BOR Controlled by Software

The Low-Power Sleep mode is beneficial for applications that stay in Sleep mode for long periods of time. The Normal mode is beneficial for applications that need to wake from Sleep quickly and frequently.

11.2.3.2 Peripheral Usage in Sleep

Some peripherals that can operate in Sleep mode will not operate properly with the Low-Power Sleep mode selected. The Low-Power Sleep mode is intended for use with these peripherals:

- Brown-out Reset (BOR)
- Windowed Watchdog Timer (WWDT)
- External interrupt pin/Interrupt-On-Change pins
- Timer1 (with external clock source)

It is the responsibility of the end user to determine what is acceptable for their application when setting the VREGPM settings in order to ensure operation in Sleep.

Important: The LF devices do not have a configurable Low-Power Sleep mode. LFs are unregulated devices and are always in the lowest power state when in Sleep, with no wake-up time penalty. These devices have a lower maximum V_{DD} and I/O voltage than the F devices.

11.3 Idle Mode

When the Idle Enable (IDLEN) bit is clear (IDLEN = 0), the SLEEP instruction will put the device into full Sleep mode. When IDLEN is set (IDLEN = 1), the SLEEP instruction will put the device into IDLE mode. In IDLE mode, the CPU and memory operations are halted, but the peripheral clocks continue to run. This mode is similar to DOZE mode, except that in IDLE both the CPU and program memory are shut off.

Important: Peripherals using F_{OSC} will continue running while in Idle (but not in Sleep). Peripherals using HFINTOSC:LFINTOSC will continue running in both Idle and Sleep.

Important: If $\overline{\text{CLKOUTEN}}$ is enabled ($\overline{\text{CLKOUTEN}}$ = 0, Configuration Word 1), the output will continue operating while in Idle.

11.3.1 Idle and Interrupts

IDLE mode ends when an interrupt occurs (even if GIE = 0), but IDLEN is not changed. The device can re-enter IDLE by executing the SLEEP instruction.

If Recover-On-Interrupt is enabled (ROI = 1), the interrupt that brings the device out of Idle also restores full-speed CPU execution when doze is also enabled.

11.3.2 Idle and WWDT

When in Idle, the WWDT Reset is blocked and will instead wake the device. The WWDT wake-up is not an interrupt, therefore ROI does not apply.

```
BANKSEL     NVMADRL
MOVF        ADDRL,W
MOVWF       NVMADRL           ; Load lower 8 bits of erase address boundary
MOVF        ADDRH,W
MOVWF       NVMADRH           ; Load upper 6 bits of erase address boundary
BCF         NVMCON1,NVMREGS   ; Choose PFM memory area
BSF         NVMCON1,FREE      ; Specify an erase operation
BSF         NVMCON1,WREN      ; Enable writes
BCF         INTCON,GIE        ; Disable interrupts during unlock sequence

; ---------------------REQUIRED UNLOCK SEQUENCE:--------------------

MOVLW       55h               ; Load 55h to get ready for unlock sequence
MOVWF       NVMCON2           ; First step is to load 55h into NVMCON2
MOVLW       AAh               ; Second step is to load AAh into W
MOVWF       NVMCON2           ; Third step is to load AAh into NVMCON2
BSF         NVMCON1,WR        ; Final step is to set WR bit

; ---------------------------------------------------------------------

BSF         INTCON,GIE        ; Re-enable interrupts, erase is complete
BCF         NVMCON1,WREN      ; Disable writes
```

Table 13-1. NVM Organization and Access Information

Master Values				NVMREG Access			FSR Access	
Memory Function	Memory Type	Program Counter (PC)	ICSP Address	NVMREGS bit (NVMCON1)	NVMADR<14:0>	Allowed Operations	FSR Address	FSR Programming Access
RESET VECTOR	Program Flash Memory	0000h	0000h	0	0000h	READ/ WRITE	8000h	READ ONLY
USER MEMORY		0001h	0001h	0	0001h		8001h	
		0003h	0003h		0003h		8003h	
INT VECTOR		0004h	0004h	0	0004h		8004h	
USER MEMORY		0005h	0005h	0	0005h		8005h	
		7FFFh(1)	7FFFh(1)		7FFFh(1)		FFFFh	
USER ID	Program Flash Memory	NO PC ACCESS	8000h	1	0000h	READ	NO ACCESS	
			8003h		0003h			
Reserved	—		—	—	0004h	—		
REV ID	HC		8005h	1	0005h	READ		
DEVICE ID			8006h	1	0006h			
CONFIG1	FUSE		8007h	1	0007h	READ/ WRITE		
CONFIG2			8008h	1	0008h			
CONFIG3			8009h	1	0009h			
CONFIG4			800Ah	1	000Ah			
CONFIG5			800Bh	1	000Bh			
DIA and DCI	PFM		8100h	1	0100h	READ		
			82FFh	1	02FFh			
USER MEMORY	EEPROM		F000h	1	7000h	READ/ WRITE	7000h	READ ONLY
			F0FFh		70FFh		70FFh	

19. Temperature Indicator Module

This family of devices is equipped with a temperature circuit designed to measure the operating temperature of the silicon die. The main purpose of the temperature indicator module is to provide a temperature-dependent voltage that can be measured by the Analog-to-Digital Converter.

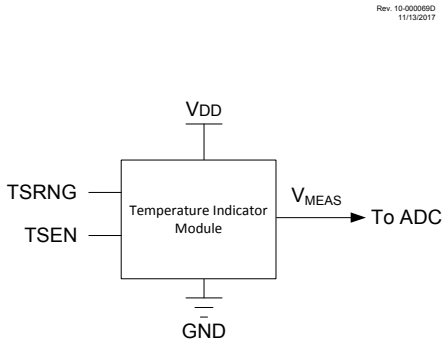
The circuit's range of operating temperature falls between -40°C and +125°C. The circuit may be used as a temperature threshold detector or a more accurate temperature indicator, depending on the level of calibration performed. A one-point calibration allows the circuit to indicate a temperature closely surrounding that point. A two-point calibration allows the circuit to sense the entire range of temperature more accurately.

19.1 Module Operation

The temperature indicator module consists of a temperature-sensing circuit that provides a voltage to the device ADC. The analog voltage output, V_{MEAS}, varies inversely to the device temperature. The output of the temperature indicator is referred to as V_{MEAS}.

The following figure shows a simplified block diagram of the temperature indicator module.

Figure 19-1. Temperature Indicator Module Block Diagram



The output of the circuit is measured using the internal Analog-to-Digital Converter. A channel is reserved for the temperature circuit output. Refer to the ADC link below for detailed information.

The ON/OFF bit for the module is located in the FVRCON register. The circuit is enabled by setting the TSEN bit of the FVRCON register. When the module is disabled, the circuit draws no current. Refer to the FVR link below for more information.

The circuit operates in either High or Low range. Refer to the *"Temperature Indicator Range"* for more details on the range settings.

Related Links

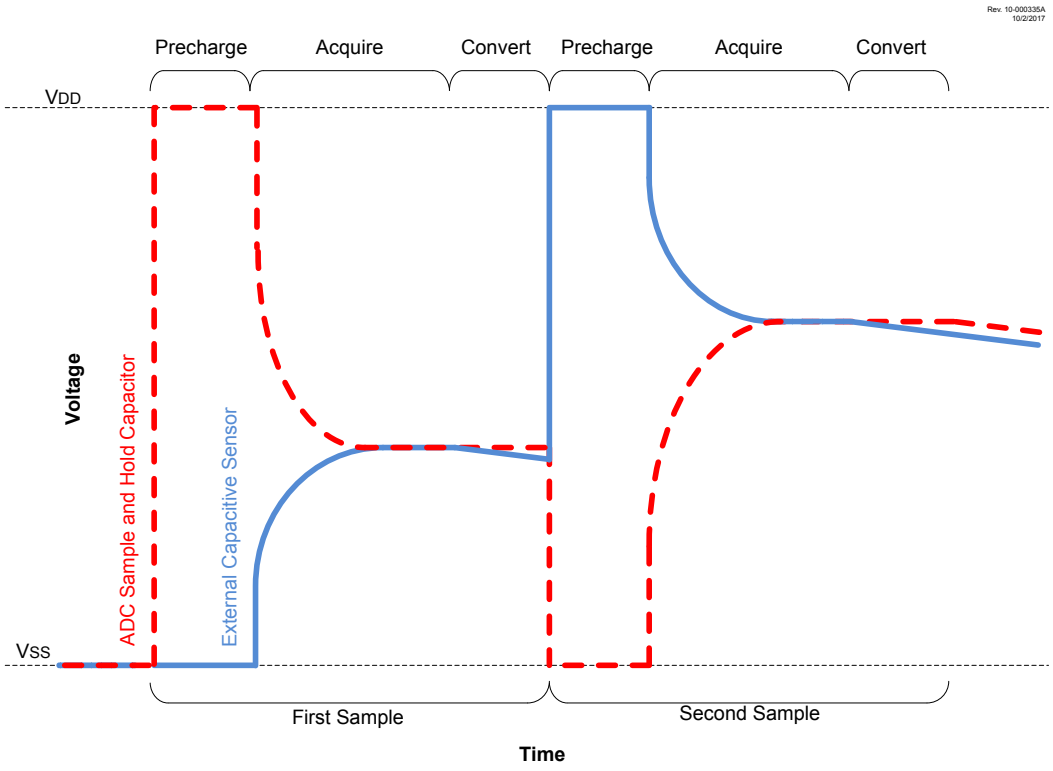
(FVR) Fixed Voltage Reference

Temperature Indicator Range

(ADC2) Analog-to-Digital Converter with Computation Module

C_{HOLD} which is determined by the capacitances and precharge levels of the two nodes. After acquisition, the ADC converts the voltage level on C_{HOLD}. This process is then repeated with the selected precharge levels inverted for both the C_{HOLD} and the sensor nodes. The waveform for two CVD measurements, which is known as differential CVD measurement, is shown in the following figure.

Figure 20-8. Differential CVD Measurement Waveform



20.5.2 Precharge Control

The Precharge stage is an optional period of time that brings the external channel and internal sample and hold capacitor to known voltage levels. Precharge is enabled by writing a non-zero value to the ADPRE register. This stage is initiated when an ADC conversion begins, either from setting the GO bit, a special event trigger, or a conversion restart from the computation functionality. If the ADPRE register is cleared when an ADC conversion begins, this stage is skipped.

During the precharge time, C_{HOLD} is disconnected from the outer portion of the sample path that leads to the external capacitive sensor and is connected to either V_{DD} or V_{SS}, depending on the value of the PPOL bit. At the same time, the port pin logic of the selected analog channel is overridden to drive a digital high or low out, in order to precharge the outer portion of the ADC's sample path, which includes the external sensor. The output polarity of this override is also determined by the PPOL bit. The amount of time that this charging needs is controlled by the ADPRE register.

Important: The external charging overrides the TRIS setting of the respective I/O pin. If there is a device attached to this pin, Precharge should not be used.

Related Links

ADCON1
ADPRE

20.5.3 Acquisition Control for CVD

The Acquisition stage allows time for the voltage on the internal sample and hold capacitor to charge or discharge from the selected analog channel. This acquisition time is controlled by the ADACQ register. If ADPRE = 0, acquisition starts at the beginning of conversion. When ADPRE ≠ 0, the acquisition stage begins when precharge ends.

At the start of the acquisition stage, the port pin logic of the selected analog channel is overridden to turn off the digital high/low output drivers so they do not affect the final result of the charge averaging. Also, the selected ADC channel is connected to C_{HOLD}. This allows charge averaging to proceed between the precharged channel and the C_{HOLD} capacitor.

Important: When ADPRE ≠ 0 setting ADACQ to '0' will set a maximum acquisition time (8191 ADC clock cycles). When ADPRE = 0, setting ADACQ to '0' will disable hardware acquisition time control.

20.5.4 Guard Ring Outputs

The following figure shows a typical guard ring circuit. C_{GUARD} represents the capacitance of the guard ring trace placed on the PCB board. The user selects values for R_A and R_B that will create a voltage profile on C_{GUARD}, which will match the selected acquisition channel.

The purpose of the guard ring is to generate a signal in phase with the CVD sensing signal to minimize the effects of the parasitic capacitance on sensing electrodes. It also can be used as a mutual drive for mutual capacitive sensing. For more information about active guard and mutual drive, see Application Note AN1478, "mTouch™ Sensing Solution Acquisition Methods Capacitive Voltage Divider".

The ADC has two guard ring drive outputs, ADGRDA and ADGRDB. These outputs can be routed through PPS controls to I/O pins (see *"Peripheral Pin Select (PPS) Module"* for details) and the polarity of these outputs are controlled by the GPOL and IPEN bits.

At the start of the first precharge stage, both outputs are set to match the GPOL bit. Once the acquisition stage begins, ADGRDA changes polarity, while ADGRDB remains unchanged. When performing a double sample conversion, setting the IPEN bit causes both guard ring outputs to transition to the opposite polarity of GPOL at the start of the second precharge stage, and ADGRDA toggles again for the second acquisition. For more information on the timing of the guard ring output, refer to the two following figures.

- Accurate long term time measurement
- Dimmer phase delayed drive
- Low EMI cycle switching

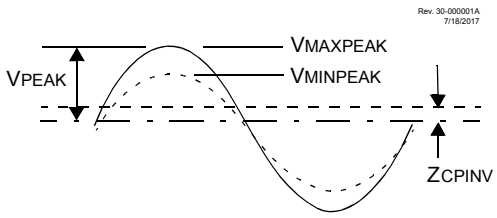
24.1 External Resistor Selection

The ZCD module requires a current-limiting resistor in series with the external voltage source. The impedance and rating of this resistor depends on the external source peak voltage. Select a resistor value that will drop all of the peak voltage when the current through the resistor is nominally 300 µA. Make sure that the ZCD I/O pin internal weak pull-up is disabled so it does not interfere with the current source and sink.

Equation 24-1. External Resistor

$$R_{SERIES} = \frac{V_{PEAK}}{3 \times 10^{-4}}$$

Figure 24-3. External Voltage Source



24.2 ZCD Logic Output

The ZCD module includes a Status bit, which can be read to determine whether the current source or sink is active. The OUT bit is set when the current sink is active, and cleared when the current source is active. The OUT bit is affected by the polarity bit.

The OUT signal can also be used as input to other modules. This is controlled by the registers of the corresponding module. OUT can be used as follows:

- Gate source for TMR1/3/5
- Clock source for TMR2/4/6
- Reset source for TMR2/4/6

24.3 ZCD Logic Polarity

The POL bit inverts the OUT bit relative to the current source and sink output. When the POL bit is set, a OUT high indicates that the current source is active, and a low output indicates that the current sink is active.

The POL bit affects the ZCD interrupts.

31.15.6 CWGxAS0

Name: CWGxAS0
Address: 0x612,0x61C

CWG Auto-Shutdown Control Register 0

Bit	7	6	5	4	3	2	1	0
	SHUTDOWN	REN	LSBD[1:0]		LSAC[1:0]			
Access	R/W/HS/HC	R/W	R/W	R/W	R/W	R/W		
Reset	0	0	0	1	0	1		

Bit 7 – SHUTDOWN Auto-Shutdown Event Status bit(1,2)

Value	Description
1	An auto-shutdown state is in effect
0	No auto-shutdown event has occurred

Bit 6 – REN Auto-Restart Enable bit

Value	Description
1	Auto-restart is enabled
0	Auto-restart is disabled

Bits 5:4 – LSBD[1:0] CWGxB and CWGxD Auto-Shutdown State Control bits

Value	Description
11	A logic '1' is placed on CWGxB/D when an auto-shutdown event occurs.
10	A logic '0' is placed on CWGxB/D when an auto-shutdown event occurs.
01	Pin is tri-stated on CWGxB/D when an auto-shutdown event occurs.
00	The inactive state of the pin, including polarity, is placed on CWGxB/D after the required dead-band interval when an auto-shutdown event occurs.

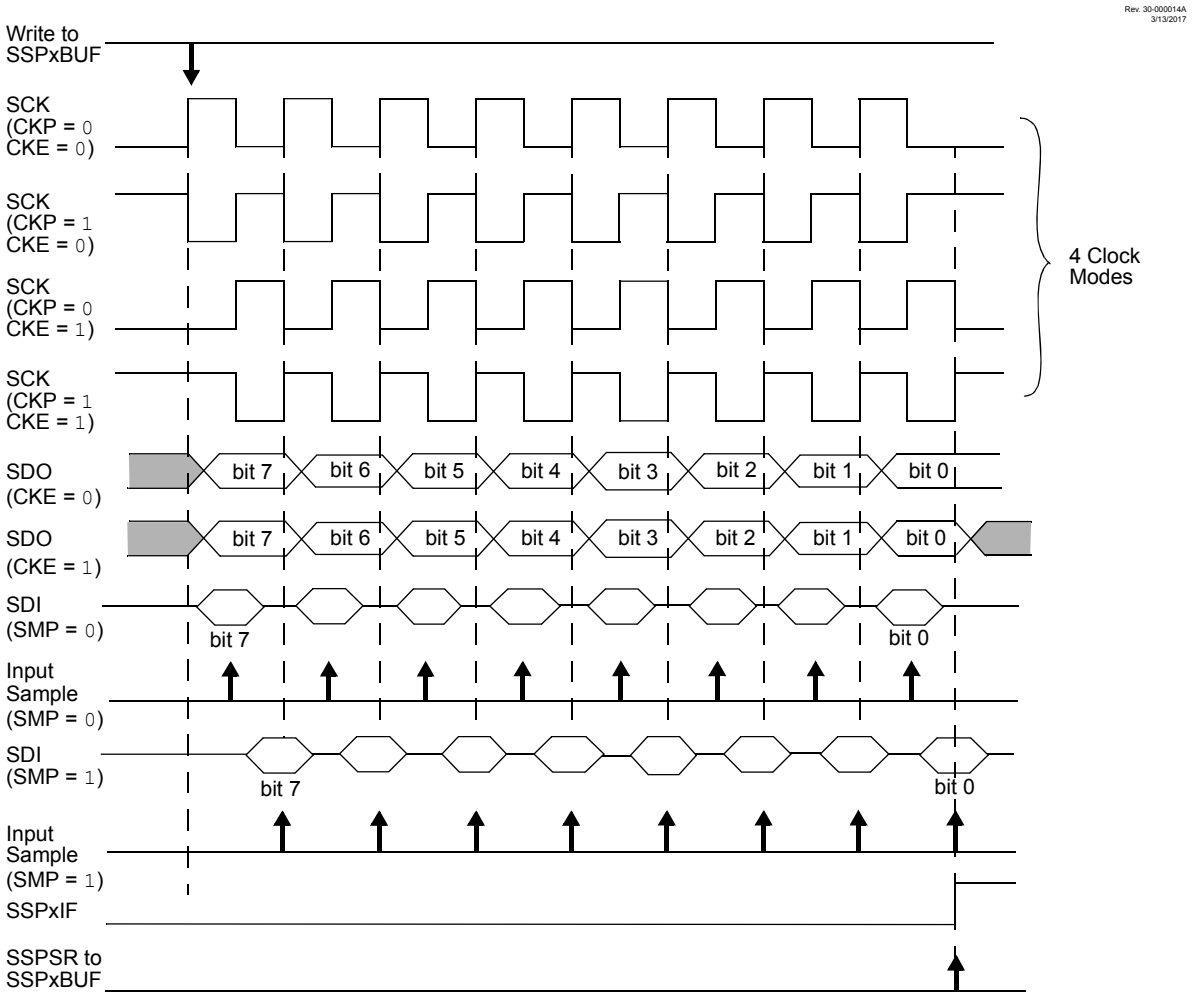
Bits 3:2 – LSAC[1:0] CWGxA and CWGxC Auto-Shutdown State Control bits

Value	Description
11	A logic '1' is placed on CWGxA/C when an auto-shutdown event occurs.
10	A logic '0' is placed on CWGxA/C when an auto-shutdown event occurs.
01	Pin is tri-stated on CWGxA/C when an auto-shutdown event occurs.
00	The inactive state of the pin, including polarity, is placed on CWGxA/C after the required dead-band interval when an auto-shutdown event occurs.

Note:

1. This bit may be written while EN = 0 (CWGxCON0), to place the outputs into the shutdown configuration.
2. The outputs will remain in auto-shutdown state until the next rising edge of the CWG data input after this bit is cleared.

Figure 35-4. SPI Mode Waveform (Master Mode)



35.2.2 SPI Slave Mode

In Slave mode, the data is transmitted and received as external clock pulses appear on SCK. When the last bit is latched, the SSPxIF interrupt flag bit is set.

Before enabling the module in SPI Slave mode, the clock line must match the proper Idle state. The clock line can be observed by reading the SCK pin. The Idle state is determined by the CKP bit.

While in Slave mode, the external clock is supplied by the external clock source on the SCK pin. This external clock must meet the minimum high and low times as specified in the electrical specifications.

While in Sleep mode, the slave can transmit/receive data. The shift register is clocked from the SCK pin input and when a byte is received, the device will generate an interrupt. If enabled, the device will wake-up from Sleep.

35.2.3 Daisy-Chain Configuration

The SPI bus can sometimes be connected in a daisy-chain configuration. The first slave output is connected to the second slave input, the second slave output is connected to the third slave input, and so on. The final slave output is connected to the master input. Each slave sends out, during a second group of clock pulses, an exact copy of what was received during the first group of clock pulses. The whole

35.5.6 Clock Stretching

Clock stretching occurs when a device on the bus holds the SCL line low, effectively pausing communication. The slave may stretch the clock to allow more time to handle data or prepare a response for the master device. A master device is not concerned with stretching as anytime it is active on the bus and not transferring data it is stretching. Any stretching done by a slave is invisible to the master software and handled by the hardware that generates SCL.

The CKP bit is used to control stretching in software. Any time the CKP bit is cleared, the module will wait for the SCL line to go low and then hold it. Setting CKP will release SCL and allow more communication.

35.5.6.1 Normal Clock Stretching

Following an $\overline{ACK}$ if the R/W bit is set, a read request, the slave hardware will clear CKP. This allows the slave time to update SSPxBUF with data to transfer to the master. If the SEN bit is set, the slave hardware will always stretch the clock after the $\overline{ACK}$ sequence. Once the slave is ready; CKP is set by software and communication resumes.

Important:

1. The BF bit has no effect on if the clock will be stretched or not. This is different than previous versions of the module that would not stretch the clock, clear CKP, if SSPxBUF was read before the ninth falling edge of SCL.
2. Previous versions of the module did not stretch the clock for a transmission if SSPxBUF was loaded before the ninth falling edge of SCL. It is now always cleared for read requests.

35.5.6.2 10-bit Addressing Mode

In 10-bit Addressing mode, when the UA bit is set, the clock is always stretched. This is the only time the SCL is stretched without CKP being cleared. SCL is released immediately after a write to SSPxADD.

Important: Previous versions of the module did not stretch the clock if the second address byte did not match.

35.5.6.3 Byte NACKing

When the AHEN bit is set; CKP is cleared by hardware after the eighth falling edge of SCL for a received matching address byte. When the DHEN bit is set; CKP is cleared after the eighth falling edge of SCL for received data.

Stretching after the eighth falling edge of SCL allows the slave to look at the received address or data and decide if it wants to ACK the received data.

35.5.7 Clock Synchronization and the CKP bit

Any time the CKP bit is cleared, the module will wait for the SCL line to go low and then hold it. However, clearing the CKP bit will not assert the SCL output low until the SCL output is already sampled low. Therefore, the CKP bit will not assert the SCL line until an external I^2C master device has already asserted the SCL line. The SCL output will remain low until the CKP bit is set and all other devices on the I^2C bus have released SCL. This ensures that a write to the CKP bit will not violate the minimum high time requirement for SCL (see the following figure).

Table 36-1. Baud Rate Formulas

Configuration Bits			BRG/EUSART Mode	Baud Rate Formula
SYNC	BRG16	BRGH		
0	0	0	8-bit/Asynchronous	$F_{OSC}/[64\ (n+1)]$
0	0	1	8-bit/Asynchronous	$F_{OSC}/[16\ (n+1)]$
0	1	0	16-bit/Asynchronous	
0	1	1	16-bit/Asynchronous	$F_{OSC}/[4\ (n+1)]$
1	0	x	8-bit/Synchronous	
1	1	x	16-bit/Synchronous	

Note: x = Don't care, n = value of SPxBRGH:SPxBRGL register pair.

Table 36-2. Sample Baud Rates for Asynchronous Modes

BAUD RATE	SYNC = 0, BRGH = 0, BRG16 = 0											
	F_{OSC} = 32.000 MHz			F_{OSC} = 20.000 MHz			F_{OSC} = 18.432 MHz			F_{OSC} = 11.0592 MHz		
	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)
300	—	—	—	—	—	—	—	—	—	—	—	—
1200	—	—	—	1221	1.73	255	1200	0.00	239	1200	0.00	143
2400	2404	0.16	207	2404	0.16	129	2400	0.00	119	2400	0.00	71
9600	9615	0.16	51	9470	-1.36	32	9600	0.00	29	9600	0.00	17
10417	10417	0.00	47	10417	0.00	29	10286	-1.26	27	10165	-2.42	16
19.2k	19.23k	0.16	25	19.53k	1.73	15	19.20k	0.00	14	19.20k	0.00	8
57.6k	55.55k	-3.55	3	—	—	—	57.60k	0.00	7	57.60k	0.00	2
115.2k	—	—	—	—	—	—	—	—	—	—	—	—

BAUD RATE	SYNC = 0, BRGH = 0, BRG16 = 0											
	Fosc = 8.000 MHz			Fosc = 4.000 MHz			Fosc = 3.6864 MHz			Fosc = 1.000 MHz		
	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)
300	—	—	—	300	0.16	207	300	0.00	191	300	0.16	51
1200	1202	0.16	103	1202	0.16	51	1200	0.00	47	1202	0.16	12
2400	2404	0.16	51	2404	0.16	25	2400	0.00	23	—	—	—
9600	9615	0.16	12	—	—	—	9600	0.00	5	—	—	—

used to clock both the counter and any synchronization logic used by the module. Refer the table below for possible clock source options.

The polarity of the clock source can be selected using the CPOL bit in the SMTxCON0 register.

Table 37-1. SMT Clock Source Selection

CSEL<2:0>	Clock Source
111	CLKREF output
110	SOSC
101	MFINTOSC (31.25kHz)
100	MFINTOSC (500kHz)
011	LFINTOSC
010	HFINTOSC
001	F_{OSC}
000	$F_{OSC}/4$

37.1.2 Signal and Window Source Selection

The SMT signal and window sources are selected by configuring the SSEL bits in the SMTxSIG register and the WSEL bits in the SMTxWIN register. Refer the tables below for the possible selections.

The polarity of the signal and window sources can be selected using the SPOL and WPOL bits in the SMTxCON0 register.

Table 37-2. SMT Signal Selection

SSEL<4:0>	SMT1 Signal Source
11111-10110	Reserved
10101	CLC4OUT
10100	CLC3OUT
10011	CLC2OUT
10010	CLC1OUT
10001	ZCDOUT
10000	C2OUT
01111	C1OUT
01110	NCO1OUT
01101	PWM7OUT
01100	PWM6OUT
01011	CCP4OUT
01010	CCP3OUT
01001	CCP2OUT

Offset	Name	Bit Pos.								
0x0581	INDF1	7:0	INDF1[7:0]							
0x0582	PCL	7:0	PCL[7:0]							
0x0583	STATUS	7:0				$\overline{TO}$	$\overline{PD}$	Z	DC	C
0x0584	FSR0	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x0586	FSR1	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x0588	BSR	7:0			BSR[5:0]					
0x0589	WREG	7:0	WREG[7:0]							
0x058A	PCLATH	7:0		PCLATH[6:0]						
0x058B	INTCON	7:0	GIE	PEIE						INTEDG
0x058C	NCO1ACC	7:0	ACCL[7:0]							
		15:8	ACCH[7:0]							
		23:16					ACCU[3:0]			
0x058F	NCO1INC	7:0	INCL[7:0]							
		15:8	INCH[7:0]							
		23:16					INCU[3:0]			
0x0592	NCO1CON	7:0	EN		OUT	POL				PFM
0x0593	NCO1CLK	7:0	PWS[2:0]				CKS[3:0]			
0x0594 ... 0x059B	Reserved									
0x059C	TMR0L	7:0	TMR0L[7:0]							
0x059D	TMR0H	7:0	TMR0H[7:0]							
0x059E	T0CON0	7:0	T0EN		T0OUT	T016BIT	T0OUTPS[3:0]			
0x059F	T0CON1	7:0	T0CS[2:0]			T0ASYNC	T0CKPS[3:0]			
0x05A0 ... 0x05FF	Reserved									
0x0600	INDF0	7:0	INDF0[7:0]							
0x0601	INDF1	7:0	INDF1[7:0]							
0x0602	PCL	7:0	PCL[7:0]							
0x0603	STATUS	7:0				$\overline{TO}$	$\overline{PD}$	Z	DC	C
0x0604	FSR0	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x0606	FSR1	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x0608	BSR	7:0			BSR[5:0]					
0x0609	WREG	7:0	WREG[7:0]							
0x060A	PCLATH	7:0		PCLATH[6:0]						
0x060B	INTCON	7:0	GIE	PEIE						INTEDG
0x060C	CWG1CLK	7:0								CS
0x060D	CWG1ISM	7:0					ISM[3:0]			
0x060E	CWG1DBR	7:0			DBR[5:0]					
0x060F	CWG1DBF	7:0			DBF[5:0]					
0x0610	CWG1CON0	7:0	EN	LD				MODE[2:0]		

- Conditional assembly for multipurpose source files
- Directives that allow complete control over the assembly process

41.4 MPLINK Object Linker/MPLIB Object Librarian

The MPLINK Object Linker combines relocatable objects created by the MPASM Assembler. It can link relocatable objects from precompiled libraries, using directives from a linker script.

The MPLIB Object Librarian manages the creation and modification of library files of precompiled code. When a routine from a library is called from a source file, only the modules that contain that routine will be linked in with the application. This allows large libraries to be used efficiently in many different applications.

The object linker/library features include:

- Efficient linking of single libraries instead of many smaller files
- Enhanced code maintainability by grouping related modules together
- Flexible creation of libraries with easy module listing, replacement, deletion and extraction

41.5 MPLAB Assembler, Linker and Librarian for Various Device Families

MPLAB Assembler produces relocatable machine code from symbolic assembly language for PIC24, PIC32 and dsPIC DSC devices. MPLAB XC Compiler uses the assembler to produce its object file. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. Notable features of the assembler include:

- Support for the entire device instruction set
- Support for fixed-point and floating-point data
- Command-line interface
- Rich directive set
- Flexible macro language
- MPLAB X IDE compatibility

41.6 MPLAB X SIM Software Simulator

The MPLAB X SIM Software Simulator allows code development in a PC-hosted environment by simulating the PIC MCUs and dsPIC DSCs on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a comprehensive stimulus controller. Registers can be logged to files for further run-time analysis. The trace buffer and logic analyzer display extend the power of the simulator to record and track program execution, actions on I/O, most peripherals and internal registers.

The MPLAB X SIM Software Simulator fully supports symbolic debugging using the MPLAB XC Compilers, and the MPASM and MPLAB Assemblers. The software simulator offers the flexibility to develop and debug code outside of the hardware laboratory environment, making it an excellent, economical software development tool.

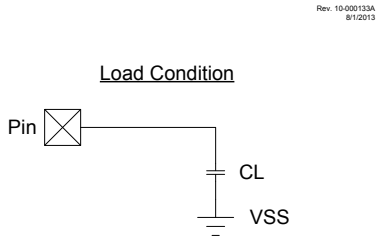
Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +125°C					
Param No.	**Sym.**	**Characteristic**	**Typ.**	**Units**	**Conditions**
TH04	PD	Power Dissipation	—	W	$PD=P_{INTERNAL}+P_{I/O}$
TH05	$P_{INTERNAL}$	Internal Power Dissipation	—	W	$P_{INTERNAL}=I_{DD}xV_{DD}$ (1)
TH06	$P_{I/O}$	I/O Power Dissipation	—	W	$P_{I/O}=\Sigma(I_{OL}*V_{OL})+\Sigma(I_{OH}*(V_{DD}-V_{OH}))$
TH07	P_{DER}	Derated Power	—	W	$P_{DER}=PD_{MAX}\ (T_J-T_A)/\theta_{JA}$ (2)

Note:

1. I_{DD} is current to run the chip alone without driving any load on the output pins.
2. T_A = Ambient Temperature, T_J = Junction Temperature.

42.4 AC Characteristics

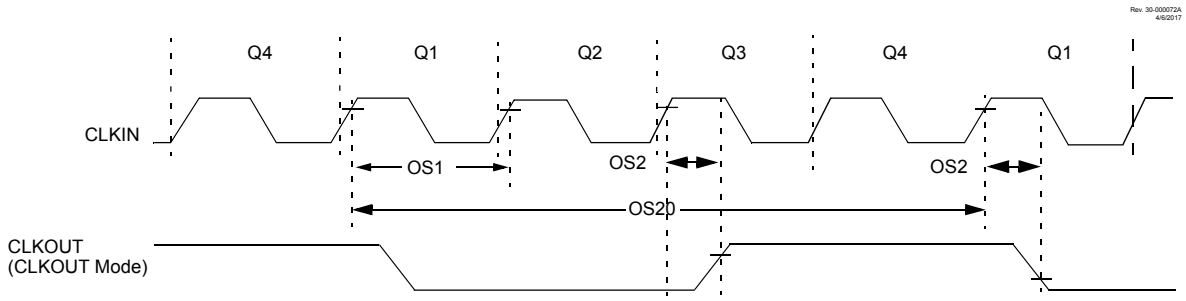
Figure 42-4. Load Conditions



Legend: CL=50 pF for all pins

42.4.1 External Clock/Oscillator Timing Requirements

Figure 42-5. Clock Timing



Note: See table below.

42.4.6 Temperature Indicator Requirements

Table 42-12.

Standard Operating Conditions (unless otherwise stated)								
Param. No.	**Sym.**	**Characteristic**		**Min.**		**Max.**	**Units**	**Conditions**
TS01	T_{ACQMIN}	Minimum ADC Acquisition Time Delay		—	25	—	µs	
TS02	Mv	Voltage Sensitivity	High Range	—	-3.684	—	mV/°C	TSRNG = 1
			Low Range	—	-3.456	—	mV/°C	TSRNG = 0

* - These parameters are characterized but not tested.

† Data in “Typ” column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

42.4.7 Analog-To-Digital Converter (ADC) Accuracy Specifications(1,2)

Table 42-13.

Standard Operating Conditions (unless otherwise stated) V_{DD} = 3.0V, T_A = 25°C, T_{AD} = 1µs							
Param No.	**Sym.**	**Characteristic**	**Min.**	**Typ. †**	**Max.**	**Units**	**Conditions**
AD01	N_R	Resolution	—	—	12	bit	
AD02	E_{IL}	Integral Error	—	±1.0	±0.2	LSb	ADC_{REF}+=3.0V, ADC_{REF}- = 0V
AD03	E_{DL}	Differential Error	—	±1.0	±1.0	LSb	ADC_{REF}+=3.0V, ADC_{REF}- = 0V
AD04	E_{OFF}	Offset Error	—	0.5	6.5	LSb	ADC_{REF}+=3.0V, ADC_{REF}- = 0V
AD05	E_{GN}	Gain Error	—	±0.2	±6.0	LSb	ADC_{REF}+=3.0V, ADC_{REF}- = 0V
AD06	V_{ADREF}	ADC Reference Voltage (AD_{REF}+ - AD_{REF}-)	1.8	—	V_{DD}	V	
AD07	V_{AIN}	Full-Scale Range	AD_{REF}-	—	AD_{REF}+	V	
AD08	Z_{AIN}	Recommended Impedance of Analog Voltage Source	—	10	—	kΩ	
AD09	R_{VREF}	ADC Voltage Reference Ladder Impedance	—	50	—	kΩ	

* - These parameters are characterized but not tested.