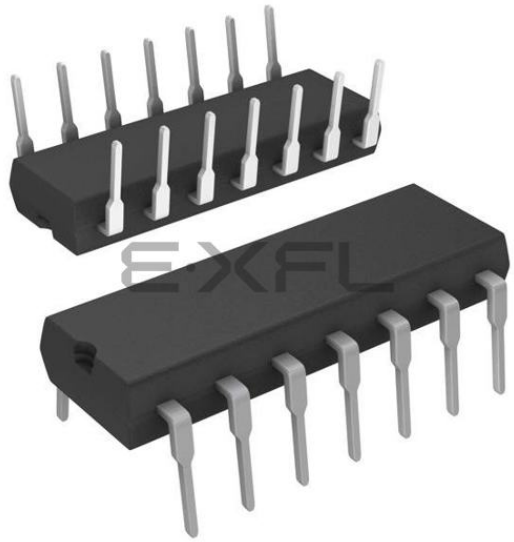


Welcome to E-XFL.COM

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	12
Program Memory Size	14KB (8K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 8x10b; D/A 1x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Through Hole
Package / Case	14-DIP (0.300", 7.62mm)
Supplier Device Package	14-PDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf1615-e-p

6.13 Power Control (PCON) Register

The Power Control (PCON) register contains flag bits to differentiate between a:

- Power-On Reset ($\overline{POR}$)
- Brown-Out Reset ($\overline{BOR}$)
- Reset Instruction Reset ($\overline{RI}$)
- $\overline{MCLR}$ Reset ($\overline{RMCLR}$)
- Watchdog Timer Reset ($\overline{RWDT}$)
- Stack Underflow Reset (STKUNF)
- Stack Overflow Reset (STKOVF)

The PCON register bits are shown in Register 6-2.

6.14 Register Definitions: Power Control

REGISTER 6-2: PCON: POWER CONTROL REGISTER

R/W/HS-0/q	R/W/HS-0/q	R/W/HC-1/q	R/W/HC-1/q	R/W/HC-1/q	R/W/HC-1/q	R/W/HC-q/u	R/W/HC-q/u
STKOVF	STKUNF	$\overline{WDTWV}$	$\overline{RWDT}$	$\overline{RMCLR}$	$\overline{RI}$	$\overline{POR}$	$\overline{BOR}$
bit 7							bit 0

Legend:		
HC = Bit is cleared by hardware		HS = Bit is set by hardware
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	q = Value depends on condition

bit 7 **STKOVF:** Stack Overflow Flag bit
1 = A Stack Overflow occurred
0 = A Stack Overflow has not occurred or cleared by firmware

bit 6 **STKUNF:** Stack Underflow Flag bit
1 = A Stack Underflow occurred
0 = A Stack Underflow has not occurred or cleared by firmware

bit 5 **$\overline{WDTWV}$:** WDT Window Violation Flag bit
1 = A WDT Window Violation Reset has not occurred or set by firmware
0 = A WDT Window Violation Reset has occurred (a CLRWDT instruction was executed either without arming the window or outside the window (cleared by hardware)

bit 4 **$\overline{RWDT}$:** Watchdog Timer Reset Flag bit
1 = A Watchdog Timer Reset has not occurred or set by firmware
0 = A Watchdog Timer Reset has occurred (cleared by hardware)

bit 3 **$\overline{RMCLR}$:** $\overline{MCLR}$ Reset Flag bit
1 = A $\overline{MCLR}$ Reset has not occurred or set by firmware
0 = A $\overline{MCLR}$ Reset has occurred (cleared by hardware)

bit 2 **$\overline{RI}$:** RESET Instruction Flag bit
1 = A RESET instruction has not been executed or set by firmware
0 = A RESET instruction has been executed (cleared by hardware)

bit 1 **$\overline{POR}$:** Power-On Reset Status bit
1 = No Power-on Reset occurred
0 = A Power-on Reset occurred (must be set in software after a Power-on Reset occurs)

bit 0 **$\overline{BOR}$:** Brown-Out Reset Status bit
1 = No Brown-out Reset occurred
0 = A Brown-out Reset occurred (must be set in software after a Power-on Reset or Brown-out Reset occurs)

7.3 Interrupts During Sleep

Some interrupts can be used to wake from Sleep. To wake from Sleep, the peripheral must be able to operate without the system clock. The interrupt source must have the appropriate Interrupt Enable bit(s) set prior to entering Sleep.

On waking from Sleep, if the GIE bit is also set, the processor will branch to the interrupt vector. Otherwise, the processor will continue executing instructions after the SLEEP instruction. The instruction directly after the SLEEP instruction will always be executed before branching to the ISR. Refer to **Section8.0 "Power-Down Mode (Sleep)"** for more details.

7.4 INT Pin

The INT pin can be used to generate an asynchronous edge-triggered interrupt. This interrupt is enabled by setting the INTE bit of the INTCON register. The INTEDG bit of the OPTION_REG register determines on which edge the interrupt will occur. When the INTEDG bit is set, the rising edge will cause the interrupt. When the INTEDG bit is clear, the falling edge will cause the interrupt. The INTF bit of the INTCON register will be set when a valid edge appears on the INT pin. If the GIE and INTE bits are also set, the processor will redirect program execution to the interrupt vector.

7.5 Automatic Context Saving

Upon entering an interrupt, the return PC address is saved on the stack. Additionally, the following registers are automatically saved in the shadow registers:

- W register
- STATUS register (except for $\overline{TO}$ and $\overline{PD}$)
- BSR register
- FSR registers
- PCLATH register

Upon exiting the Interrupt Service Routine, these registers are automatically restored. Any modifications to these registers during the ISR will be lost. If modifications to any of these registers are desired, the corresponding shadow register should be modified and the value will be restored when exiting the ISR. The shadow registers are available in Bank 31 and are readable and writable. Depending on the user's application, other registers may also need to be saved.

REGISTER 10-5: PMCON1: PROGRAM MEMORY CONTROL 1 REGISTER

U-1	R/W-0/0	R/W-0/0	R/W/HC-0/0	R/W/HC-x/q[2]	R/W-0/0	R/S/HC-0/0	R/S/HC-0/0
—[1]	CFGS	LWLO	FREE	WRERR	WREN	WR	RD
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
S = Bit can only be set	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	HC = Bit is cleared by hardware

bit 7 **Unimplemented:** Read as ‘1’

bit 6 **CFGS:** Configuration Select bit
- 1 = Access Configuration, User ID and Device ID Registers
- 0 = Access Flash program memory

bit 5 **LWLO:** Load Write Latches Only bit[3]
- 1 = Only the addressed program memory write latch is loaded/updated on the next WR command
- 0 = The addressed program memory write latch is loaded/updated and a write of all program memory write latches will be initiated on the next WR command

bit 4 **FREE:** Program Flash Erase Enable bit
- 1 = Performs an erase operation on the next WR command (hardware cleared upon completion)
- 0 = Performs a write operation on the next WR command

bit 3 **WRERR:** Program/Erase Error Flag bit
- 1 = Condition indicates an improper program or erase sequence attempt or termination (bit is set automatically on any set attempt (write ‘1’) of the WR bit)
- 0 = The program or erase operation completed normally

bit 2 **WREN:** Program/Erase Enable bit
- 1 = Allows program/erase cycles
- 0 = Inhibits programming/erasing of program Flash

bit 1 **WR:** Write Control bit
- 1 = Initiates a program Flash program/erase operation.
  The operation is self-timed and the bit is cleared by hardware once operation is complete.
  The WR bit can only be set (not cleared) in software.
- 0 = Program/erase operation to the Flash is complete and inactive

bit 0 **RD:** Read Control bit
- 1 = Initiates a program Flash read. Read takes one cycle. RD is cleared in hardware. The RD bit can only be set (not cleared) in software.
- 0 = Does not initiate a program Flash read

Note 1: Unimplemented bit, read as ‘1’.
2: The WRERR bit is automatically set by hardware when a program memory write or erase operation is started (WR = 1).
3: The LWLO bit is ignored during a program memory erase operation (FREE = 1).

12.3.7 PORTB FUNCTIONS AND OUTPUT PRIORITIES

Each pin defaults to the PORT latch data after Reset. Other functions are selected with the peripheral pin select logic. See **Section13.0 “Peripheral Pin Select (PPS) Module”** for more information. Analog input functions, such as ADC inputs, are not shown in the peripheral pin select lists. These inputs are active when the I/O pin is set for Analog mode using the ANSELB register. Digital output functions continue to may continue to control the pin when it is in Analog mode.

13.0 PERIPHERAL PIN SELECT (PPS) MODULE

The Peripheral Pin Select (PPS) module connects peripheral inputs and outputs to the device I/O pins. Only digital signals are included in the selections. All analog inputs and outputs remain fixed to their assigned pins. Input and output selections are independent as shown in the simplified block diagram Figure 13-1.

13.1 PPS Inputs

Each peripheral has a PPS register with which the inputs to the peripheral are selected. Inputs include the device pins.

Multiple peripherals can operate from the same source simultaneously. Port reads always return the pin level regardless of peripheral PPS selection. If a pin also has associated analog functions, the ANSEL bit for that pin must be cleared to enable the digital input buffer.

Although every peripheral has its own PPS input selection register, the selections are identical for every peripheral as shown in Register 13-1.

Note:	The notation “xxx” in the register name is a place holder for the peripheral identifier. For example, CLC1PPS.

13.2 PPS Outputs

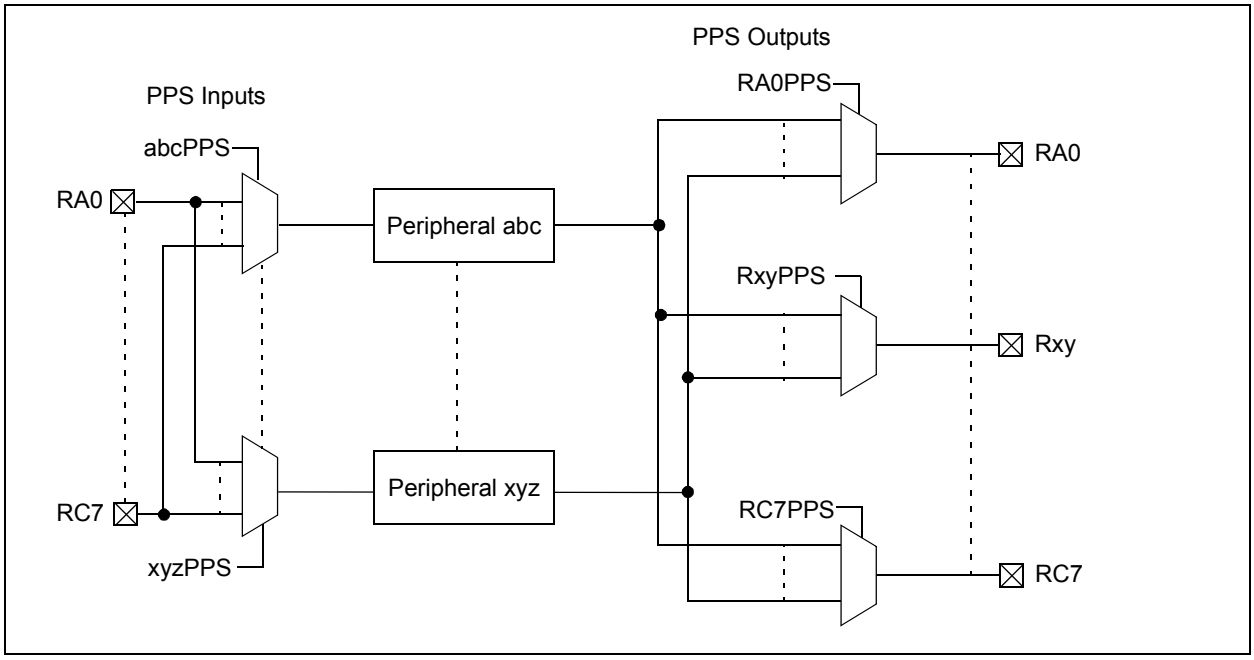
Each I/O pin has a PPS register with which the pin output source is selected. With few exceptions, the port TRIS control associated with that pin retains control over the pin output driver. Peripherals that control the pin output driver as part of the peripheral operation will override the TRIS control as needed. These peripherals include:

- EUSART (synchronous operation)
- MSSP (I^2C)
- CWG (auto-shutdown)

Although every pin has its own PPS peripheral selection register, the selections are identical for every pin as shown in Register 13-2.

Note:	The notation “Rxy” is a place holder for the pin identifier. For example, RA0PPS.

FIGURE 13-1: SIMPLIFIED PPS BLOCK DIAGRAM



23.5.9 EDGE-TRIGGERED MONOSTABLE MODES

The Edge-Triggered Monostable modes start the timer on an edge from the external Reset signal input, after the ON bit is set, and stop incrementing the timer when the timer matches the PRx period value. The following edges will start the timer:

- Rising edge (MODE<4:0> = 10001)
- Falling edge (MODE<4:0> = 10010)
- Rising or Falling edge (MODE<4:0> = 10011)

When an Edge-Triggered Monostable mode is used in conjunction with the CCP PWM operation the PWM drive goes active with the external Reset signal edge that starts the timer, but will not go active when the timer matches the PRx value. While the timer is incrementing, additional edges on the external Reset signal will not affect the CCP PWM.

REGISTER 23-3: TxHLT: TIMERx HARDWARE LIMIT CONTROL REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
PSYNC(1, 2)	CKPOL(3)	CKSYNC(4, 5)	MODE<4:0>(6, 7)				
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7 **PSYNC:** Timerx Prescaler Synchronization Enable bit(1, 2)
- 1 = TMRx Prescaler Output is synchronized to Fosc/4
- 0 = TMRx Prescaler Output is not synchronized to Fosc/4

bit 6 **CKPOL:** Timerx Clock Polarity Selection bit(3)
- 1 = Falling edge of input clock clocks timer/prescaler
- 0 = Rising edge of input clock clocks timer/prescaler

bit 5 **CKSYNC:** Timerx Clock Synchronization Enable bit(4, 5)
- 1 = ON register bit is synchronized to TMR2_clk input
- 0 = ON register bit is not synchronized to TMR2_clk input

bit 4-0 **MODE<4:0>:** Timerx Control Mode Selection bits(6, 7)
See Table 23-1.

Note 1: Setting this bit ensures that reading TMRx will return a valid value.
2: When this bit is '1', Timer2 cannot operate in Sleep mode.
3: CKPOL should not be changed while ON = 1.
4: Setting this bit ensures glitch-free operation when the ON is enabled or disabled.
5: When this bit is set then the timer operation will be delayed by two TMRx input clocks after the ON bit is set.
6: Unless otherwise indicated, all modes start upon ON = 1 and stop upon ON = 0 (stops occur without affecting the value of TMRx).
7: When TMRx = PRx, the next clock clears TMRx, regardless of the operating mode.

24.2.3 SPI MASTER MODE

The master can initiate the data transfer at any time because it controls the SCK line. The master determines when the slave (Processor 2, Figure 24-5) is to broadcast data by the software protocol.

In Master mode, the data is transmitted/received as soon as the SSPxBUF register is written to. If the SPI is only going to receive, the SDO output could be disabled (programmed as an input). The SSPSR register will continue to shift in the signal present on the SDI pin at the programmed clock rate. As each byte is received, it will be loaded into the SSPxBUF register as if a normal received byte (interrupts and Status bits appropriately set).

The clock polarity is selected by appropriately programming the CKP bit of the SSPxCON1 register and the CKE bit of the SSPxSTAT register. This then, would give waveforms for SPI communication as shown in Figure 24-6, Figure 24-8, Figure 24-9 and Figure 24-10, where the MSB is transmitted first. In Master mode, the SPI clock rate (bit rate) is user programmable to be one of the following:

- FOSC/4 (or TCY)
- FOSC/16 (or 4 * TCY)
- FOSC/64 (or 16 * TCY)
- Timer2 output/2
- FOSC/(4 * (SSPxADD + 1))

Figure 24-6 shows the waveforms for Master mode.

When the CKE bit is set, the SDO data is valid before there is a clock edge on SCK. The change of the input sample is shown based on the state of the SMP bit. The time when the SSPxBUF is loaded with the received data is shown.

Note:	In Master mode the clock signal output to the SCK pin is also the clock signal input to the peripheral. The pin selected for output with the RxyPPS register must also be selected as the peripheral input with the SSPCLKPPS register.

24.5.2 SLAVE RECEPTION

When the R/$\overline{W}$ bit of a matching received address byte is clear, the R/$\overline{W}$ bit of the SSPxSTAT register is cleared. The received address is loaded into the SSPxBUF register and acknowledged.

When the overflow condition exists for a received address, then not Acknowledge is given. An overflow condition is defined as either bit BF of the SSPxSTAT register is set, or bit SSPOV of the SSPxCON1 register is set. The BOEN bit of the SSPxCON3 register modifies this operation. For more information see Register 24-4.

An MSSP interrupt is generated for each transferred data byte. Flag bit, SSPxIF, must be cleared by software.

When the SEN bit of the SSPxCON2 register is set, SCL will be held low (clock stretch) following each received byte. The clock must be released by setting the CKP bit of the SSPxCON1 register, except sometimes in 10-bit mode. See **Section 24.5.6.2 "10-bit Addressing Mode"** for more detail.

24.5.2.1 7-bit Addressing Reception

This section describes a standard sequence of events for the MSSP module configured as an I^2C slave in 7-bit Addressing mode. Figure 24-14 and Figure 24-15 is used as a visual reference for this description.

This is a step by step process of what typically must be done to accomplish I^2C communication.

1. Start bit detected.
2. S bit of SSPxSTAT is set; SSPxIF is set if interrupt on Start detect is enabled.
3. Matching address with R/$\overline{W}$ bit clear is received.
4. The slave pulls SDA low sending an $\overline{ACK}$ to the master, and sets SSPxIF bit.
5. Software clears the SSPxIF bit.
6. Software reads received address from SSPxBUF clearing the BF flag.
7. If SEN = 1; Slave software sets CKP bit to release the SCL line.
8. The master clocks out a data byte.
9. Slave drives SDA low sending an $\overline{ACK}$ to the master, and sets SSPxIF bit.
10. Software clears SSPxIF.
11. Software reads the received byte from SSPxBUF clearing BF.
12. Steps 8-12 are repeated for all received bytes from the master.
13. Master sends Stop condition, setting P bit of SSPxSTAT, and the bus goes idle.

24.5.2.2 7-bit Reception with AHEN and DHEN

Slave device reception with AHEN and DHEN set operate the same as without these options with extra interrupts and clock stretching added after the eighth falling edge of SCL. These additional interrupts allow the slave software to decide whether it wants to $\overline{ACK}$ the receive address or data byte, rather than the hardware. This functionality adds support for PMBus™ that was not present on previous versions of this module.

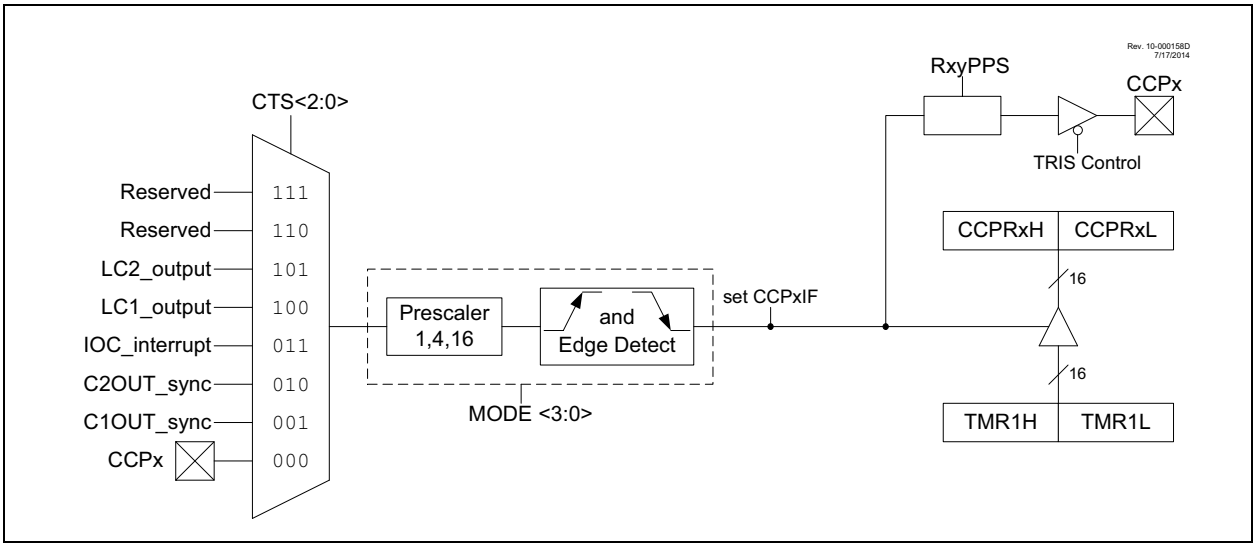
This list describes the steps that need to be taken by slave software to use these options for I^2C communication. Figure 24-16 displays a module using both address and data holding. Figure 24-17 includes the operation with the SEN bit of the SSPxCON2 register set.

1. S bit of SSPxSTAT is set; SSPxIF is set if interrupt on Start detect is enabled.
2. Matching address with R/$\overline{W}$ bit clear is clocked in. SSPxIF is set and CKP cleared after the eighth falling edge of SCL.
3. Slave clears the SSPxIF.
4. Slave can look at the ACKTIM bit of the SSPxCON3 register to determine if the SSPxIF was after or before the $\overline{ACK}$.
5. Slave reads the address value from SSPxBUF, clearing the BF flag.
6. Slave sets $\overline{ACK}$ value clocked out to the master by setting ACKDT.
7. Slave releases the clock by setting CKP.
8. SSPxIF is set after an $\overline{ACK}$, not after a NACK.
9. If SEN = 1 the slave hardware will stretch the clock after the $\overline{ACK}$.
10. Slave clears SSPxIF.

Note: SSPxIF is still set after the 9th falling edge of SCL even if there is no clock stretching and BF has been cleared. Only if NACK is sent to master is SSPxIF not set

11. SSPxIF set and CKP cleared after eighth falling edge of SCL for a received data byte.
12. Slave looks at ACKTIM bit of SSPxCON3 to determine the source of the interrupt.
13. Slave reads the received data from SSPxBUF clearing BF.
14. Steps 7-14 are the same for each received data byte.
15. Communication is ended by either the slave sending an $\overline{ACK}$ = 1, or the master sending a Stop condition. If a Stop is sent and Interrupt on Stop Detect is disabled, the slave will only know by polling the P bit of the SSTSTAT register.

FIGURE 26-1: CAPTURE MODE OPERATION BLOCK DIAGRAM



26.1.2 TIMER1 MODE RESOURCE

Timer1 must be running in Timer mode or Synchronized Counter mode for the CCP module to use the capture feature. In Asynchronous Counter mode, the capture operation may not work.

See **Section22.0 "Timer1/3/5 Module with Gate Control"** for more information on configuring Timer1.

26.1.3 SOFTWARE INTERRUPT MODE

When the Capture mode is changed, a false capture interrupt may be generated. The user should keep the CCPxIE interrupt enable bit of the PIEx register clear to avoid false interrupts. Additionally, the user should clear the CCPxIF interrupt flag bit of the PIRx register following any change in Operating mode.

Note:	Clocking Timer1 from the system clock (FOSC) should not be used in Capture mode. In order for Capture mode to recognize the trigger event on the CCPx pin, Timer1 must be clocked from the instruction clock (FOSC/4) or from an external clock source.

26.1.4 CCP PRESCALER

There are four prescaler settings specified by the MODE<3:0> bits of the CCPxCON register. Whenever the CCP module is turned off, or the CCP module is not in Capture mode, the prescaler counter is cleared. Any Reset will clear the prescaler counter.

Switching from one capture prescaler to another does not clear the prescaler and may generate a false interrupt. To avoid this unexpected operation, turn the module off by clearing the EN bit of the CCPxCON register before changing the prescaler.

26.1.5 CAPTURE DURING SLEEP

Capture mode depends upon the Timer1 module for proper operation. There are two options for driving the Timer1 module in Capture mode. It can be driven by the instruction clock (FOSC/4), or by an external clock source.

When Timer1 is clocked by FOSC/4, Timer1 will not increment during Sleep. When the device wakes from Sleep, Timer1 will continue from its previous state.

Capture mode will operate during Sleep when Timer1 is clocked by an external clock source.

26.1.6 CAPTURE OUTPUT

Whenever a capture occurs, the output of the CCP will go high for a period equal to one system clock period (1/FOSC). This output is available as an input signal to the CWG, as an auto-conversion trigger for the ADC, as an External Reset Signal for the TMR2 modules, as a window input to the SMT, and as an input to the CLC module. In addition, the CCPx pin output can be mapped to output pins through the use of PPS (see **13.2 "PPS Outputs"**).

REGISTER 26-2: CCPTMRS: PWM TIMER SELECTION CONTROL REGISTER 0

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
P4TSEL<1:0>		P3TSEL<1:0>		C2TSEL<1:0>		C1TSEL<1:0>	
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7-6 **P4TSEL<1:0>:** PWM4 Timer Selection bits
- 11 = Reserved
- 10 = PWM4 is based off Timer6 in PWM mode
- 01 = PWM4 is based off Timer4 in PWM mode
- 00 = PWM4 is based off Timer2 in PWM mode

bit 5-4 **P3TSEL<1:0>:** PWM3 Timer Selection bits
- 11 = Reserved
- 10 = PWM3 is based off Timer6 in PWM mode
- 01 = PWM3 is based off Timer4 in PWM mode
- 00 = PWM3 is based off Timer2 in PWM mode

bit 3-2 **C2TSEL<1:0>:** CCP2 (PWM2) Timer Selection bits
- 11 = Reserved
- 10 = CCP2 is based off Timer6 in PWM mode
- 01 = CCP2 is based off Timer4 in PWM mode
- 00 = CCP2 is based off Timer2 in PWM mode

bit 1-0 **C1TSEL<1:0>:** CCP1 (PWM1) Timer Selection bits
- 11 = Reserved
- 10 = CCP1 is based off Timer6 in PWM mode
- 01 = CCP1 is based off Timer4 in PWM mode
- 00 = CCP1 is based off Timer2 in PWM mode

REGISTER 26-3: CCPRxL: CCPx LOW BYTE REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
CCPR<7:0>							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Reset
'1' = Bit is set	'0' = Bit is cleared	

bit 7-0

MODE = Capture Mode
CCPRxL<7:0>: LSB of captured TMR1 value
MODE = Compare Mode
CCPRxL<7:0>: LSB compared to TMR1 value
MODE = PWM Mode && FMT = 0
CCPRxL<7:0>: CCPW<7:0> — Pulse width Least Significant eight bits
MODE = PWM Mode && FMT = 1
CCPRxL<7:6>: CCPW<1:0> — Pulse width Least Significant two bits
CCPRxL<5:0>: Not used

REGISTER 30-5: SMT1WIN: SMT1 WINDOW INPUT SELECT REGISTER

U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	WSEL<4:0>				
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	q = Value depends on condition

bit 7-5 **Unimplemented**: Read as ‘0’

bit 4-0 **WSEL<4:0>:** SMT1 Window Selection bits

11111 = Reserved
•
•
•
11000 = Reserved
10111 = MFINTOSC/16
10110 = AT1_perclk
10101 = LFINTOSC
10100 = PWM4_out
10011 = PWM3_out
10010 = SMT2_match
10001 = Reserved
10000 = TMR0_overflow
01111 = TMR5_overflow
01110 = TMR3_overflow
01101 = TMR1_overflow
01100 = LC4_out
01011 = LC3_out
01010 = LC2_out
01001 = LC1_out
01000 = TMR6_postscaled
00111 = TMR4_postscaled
00110 = TMR2_postscaled
00101 = ZCD1_out
00100 = CCP2_out
00011 = CCP1_out
00010 = C2OUT_sync
00001 = C1OUT_sync
00000 = SMTWINx pin

REGISTER 30-12: SMTxCPRL: SMT CAPTURED PERIOD REGISTER – LOW BYTE

R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x
SMTxCPR<7:0>							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7-0 **SMTxCPR<7:0>**: Significant bits of the SMT Period Latch – Low Byte

REGISTER 30-13: SMTxCPRH: SMT CAPTURED PERIOD REGISTER – HIGH BYTE

R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x
SMTxCPR<15:8>							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7-0 **SMTxCPR<15:8>**: Significant bits of the SMT Period Latch – High Byte

REGISTER 30-14: SMTxCPRU: SMT CAPTURED PERIOD REGISTER – UPPER BYTE

R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x
SMTxCPR<23:16>							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7-0 **SMTxCPR<23:16>**: Significant bits of the SMT Period Latch – Upper Byte

REGISTER 30-18: SMTxPRL: SMT PERIOD REGISTER – LOW BYTE

R/W-x/1	R/W-x/1	R/W-x/1	R/W-x/1	R/W-x/1	R/W-x/1	R/W-x/1	R/W-x/1
SMTxPR<7:0>							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7-0 **SMTxPR<7:0>**: Significant bits of the SMT Timer Value for Period Match – Low Byte

REGISTER 30-19: SMTxPRH: SMT PERIOD REGISTER – HIGH BYTE

R/W-x/1	R/W-x/1	R/W-x/1	R/W-x/1	R/W-x/1	R/W-x/1	R/W-x/1	R/W-x/1
SMTxPR<15:8>							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7-0 **SMTxPR<15:8>**: Significant bits of the SMT Timer Value for Period Match – High Byte

REGISTER 30-20: SMTxPRU: SMT PERIOD REGISTER – UPPER BYTE

R/W-x/1	R/W-x/1	R/W-x/1	R/W-x/1	R/W-x/1	R/W-x/1	R/W-x/1	R/W-x/1
SMTxPR<23:16>							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7-0 **SMTxPR<23:16>**: Significant bits of the SMT Timer Value for Period Match – Upper Byte

31.2.5 VALID BIT

Several values used by the AT module must be calculated from external signals. As such, these values may be inaccurate for a period of time after the angular timer starts up. Because of this, the module will not output signals or trigger interrupts for a period of time after the module is enabled, or under certain other conditions that might jeopardize accurate output values. This output inhibition is indicated by the read-only VALID bit of the ATxCON1 being clear.

The following cases will clear the VALID bit in hardware:

- Any write to ATxRES register pair
- Phase counter overflow (ATxPHS register pair) clocked beyond 0x3FF)
- In-Circuit Debugger halt
- EN = 0
- ATxPER register pair = 0
- Device Reset

As long as the VALID bit is cleared, the following occurs:

- Period clock is not output and associated interrupts do not trigger.
- Missed pulse is not output and associated interrupts do not trigger.
- Phase clock is not output and associated interrupts do not trigger.
- Phase counter does not increment.
- Capture logic does not function and associated interrupts do not trigger.
- Compare logic does not function and associated interrupts do not trigger.
- Every ATxsig edge latches the period counter into the ATxPER register pair, regardless of mode.

In single-pulse modes, the VALID bit becomes set upon the 3rd active input edge of the signal that latches the ATxPER register pair. In multi-pulse modes, a missing pulse trigger is also required, ensuring that at least one full revolution of the input has occurred.

An example of the VALID bit in Single-Pulse mode is shown in Figure 31-6.

31.2.6 DETERMINING ACCURACY

The ATxRES register pair determines the resolution of the period measurement and, by extension, the maximum value that the phase counter reaches at the end of each input signal period. The interim value, ATxPER, used to derive the phase counter is, by nature of the circuitry, an integer. The ratio of the integer value obtained by the circuit and the calculated floating point value is the inherent error of the measurement. When ATxRES is small then integer rounding results in large errors. Factors that contribute to large errors include:

- Large values for ATxRES
- Relatively low ATxclk frequency
- Relatively high ATxsig input frequency

The actual error can be determined with Equation 31-7.

EQUATION 31-7:

$$period = \frac{F(ATxclk)}{F(ATxsig) \bullet (ATxRES + 1)}$$

$$error\% = 100 \bullet \left(\frac{period - int(period + 1)}{period}\right)$$

31.3 Input and Clock Selection

The input clock for the AT module can come from either the FOSC system clock or the 16 MHz HFINTOSC, and is chosen by the CS0 bit of the ATxCLK register. In addition, the clock is run through a prescaler that can be /1, /2, /4, or /8, which is configured by the PS<1:0> bits of the ATxCON0 register. This prescaled clock is then used for all clock operations of the Angular Timer, and as such, should be used for all of the equations demonstrated above determining the Angular Timer's behavior.

The input signal for the AT module can come from a variety of sources. The source is selected by the SSEL bits of the ATxSIG register (Register 31-4).

31.4 Module Outputs

31.4.1 ANGLE/PHASE CLOCK OUTPUT

The angle/phase clock signal (ATx_phsclk) can be used by the CLC as an input signal to combinational logic. The polarity of this signal is configured by the PHP bit of the ATxCON1 register.

31.4.2 PERIOD CLOCK OUTPUT

The period clock signal (ATx_perclk) can be used as an input clock for the Timer2/4/6 and Signal Measurement module, as well as an input signal to the CLC for combinational logic. The polarity of this signal is configured by the PRP bit of the ATxCON1 register (Register 31-2).

31.4.3 MISSED PULSE OUTPUT

The missed pulse signal (ATx_missedpulse) can be used by the CLC as an input signal to combinational logic. The polarity of this signal is configured by the MPP bit of the ATxCON1 register.

REGISTER 31-15: ATxIE1: ANGULAR TIMER ENABLE 1 REGISTER

U-0	U-0	U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	—	—	CC3IE	CC2IE	CC1IE
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	q = Value depends on condition

bit 7-3 **Unimplemented:** Read as '0'

bit 2 **CC3IE:** Capture/Compare Interrupt 3 Enable bit
If CC3MODE = 1 (Capture)
1 = Capture interrupt 3 is enabled
0 = Capture interrupt 3 is disabled
If CC3MODE = 0 (Compare)
1 = Compare interrupt 3 is enabled
0 = Compare interrupt 3 is disabled

bit 1 **CC2IE:** Capture/Compare Interrupt 2 Enable bit
If CC2MODE = 1 (Capture)
1 = Capture interrupt 2 is enabled
0 = Capture interrupt 2 is disabled
If CC2MODE = 0 (Compare)
1 = Compare interrupt 2 is enabled
0 = Compare interrupt 2 is disabled

bit 0 **CC1IE:** Capture/Compare Interrupt 1 Enable bit
If CC1MODE = 1 (Capture)
1 = Capture interrupt 1 is enabled
0 = Capture interrupt 1 is disabled
If CC1MODE = 0 (Compare)
1 = Compare interrupt 1 is enabled
0 = Compare interrupt 1 is disabled

REGISTER 32-6: PIDxK1H: PID K1 HIGH REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
K1<15:8>							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	q = Value depends on condition

bit 7-0 **K1<15:8>:** K1 upper eight bits. K1 is the 16-bit user-controlled coefficient calculated from Kp + Ki + Kd

REGISTER 32-7: PIDxK1L: PID K1 LOW REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
K1<7:0>							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	q = Value depends on condition

bit 7-0 **K1<7:0>:** K1 lower eight bits. K1 is the 16-bit user-controlled coefficient calculated from Kp + Ki + Kd

REGISTER 32-8: PIDxK2H: PID K2 HIGH REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
K2<15:8>							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	q = Value depends on condition

bit 7-0 K2<15:8>: K2 upper eight bits. K2 is the 16-bit user-controlled coefficient calculated from -(Kp + 2Kd)

REGISTER 32-9: PIDxK2L: PID K2 LOW REGISTER

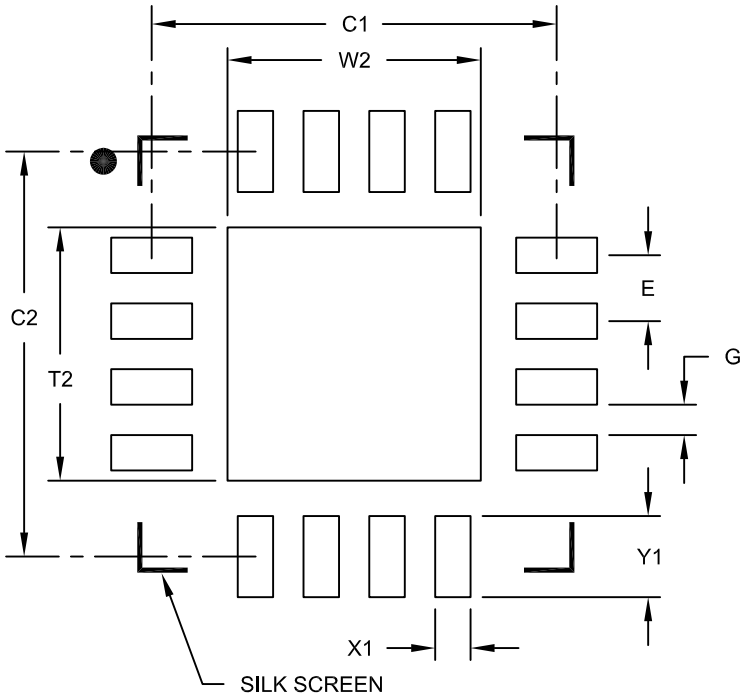
R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
K2<7:0>							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	q = Value depends on condition

bit 7-0 **K2<7:0>:** K2 lower eight bits. K2 is the 16-bit user-controlled coefficient calculated from -(Kp + 2Kd)

16-Lead Plastic Quad Flat, No Lead Package (ML) - 4x4x0.9mm Body [QFN]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E		0.65 BSC	
Optional Center Pad Width	W2			2.50
Optional Center Pad Length	T2			2.50
Contact Pad Spacing	C1		4.00	
Contact Pad Spacing	C2		4.00	
Contact Pad Width (X28)	X1			0.35
Contact Pad Length (X28)	Y1			0.80
Distance Between Pads	G	0.30		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2127A

APPENDIX A: DATA SHEET REVISION HISTORY

Revision A (10/2014)

Original release.

Revision B (4/2015)

Added High-Current pins.

Updated PIC12/16(L)F161X Family Types table and Packages Table

Added Figures 36-7 and 36-8 for V_{OH} vs. I_{OH} for high drive pins.

Deleted Figures 36-27 and 36-28.

Updated Example 3-2.

Updated Figures 5-1, 23-1, 23-2, 28-12, 31-1, 31-2, 31-4, 1, and 35-6.

Updated Registers 23-4, 27-2, 31-4 and 31-7.

Updated Sections 26.3, 25.4.2, 27.1, 30.0, 35.0, and 35.1.

Updated Table 1-2, 1-3, 3-5, 3-16, 13-2, 23-3, 23-4, 23-7, 28-1, 35-4, 35-8, and 35-17.

Updated Section 23 - added missing modes/mode summary table, reworded text to be more clear/descriptive.

Minor typos corrected.

Revision C (5/2016)

Added High endurance column to Table 1: PIC12/16(L)F161x Family Types.

Minor typos corrected.

Updated the High-Endurance Flash data memory information on the cover page. Updated Register 21-1.

Updated Section 19.6, 19.7. Updated Registers 19-2, 29-1, and 31-22. Updated Table 3: Pin Allocations Table and Table 5-1. Updated Figure 19-2.

Updated Package Drawings C04-127.