



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 "<u>Embedded -</u> <u>Microcontrollers</u>"

Details

Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	32MHz
Connectivity	I ² C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	25
Program Memory Size	7KB (4K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	2.3V ~ 5.5V
Data Converters	A/D 24x10b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-VQFN Exposed Pad
Supplier Device Package	28-QFN (6x6)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f15354-i-ml

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong

1.1 Register and Bit Naming Conventions

1.1.1 REGISTER NAMES

When there are multiple instances of the same peripheral in a device, the peripheral control registers will be depicted as the concatenation of a peripheral identifier, peripheral instance, and control identifier. The control registers section will show just one instance of all the register names with an 'x' in the place of the peripheral instance number. This naming convention may also be applied to peripherals when there is only one instance of that peripheral in the device to maintain compatibility with other devices in the family that contain more than one.

1.1.2 BIT NAMES

There are two variants for bit names:

- Short name: Bit function abbreviation
- Long name: Peripheral abbreviation + short name

1.1.2.1 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 *RegisterName*bits.*ShortName*. For example, the enable bit, EN, in the COG1CON0 register can be set in C programs with the instruction COG1CON0bits.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.

1.1.2.2 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.

1.1.2.3 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

1.1.3 REGISTER AND BIT NAMING EXCEPTIONS

1.1.3.1 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.

1.1.3.2 Legacy Peripherals

There are some peripherals that do not strictly adhere to these naming conventions. Peripherals that have existed for many years and are present in almost every device are the exceptions. These exceptions were necessary to limit the adverse impact of the new conventions on legacy code. Peripherals that do adhere to the new convention will include a table in the registers section indicating the long name prefix for each peripheral instance. Peripherals that fall into the exception category will not have this table. These peripherals include, but are not limited to, the following:

- EUSART
- MSSP

Name	Function	Input Type	Output Type	Description			
RB3/ANB3/C1IN2-/C2IN2-/IOCB3	RB3	TTL/ST	CMOS/OD	General purpose I/O.			
	ANB3	AN	_	ADC Channel B3 input.			
	C1IN2-	AN	_	Comparator 1 negative input.			
	C2IN2-	AN	_	Comparator 2 negative input.			
	IOCB3	TTL/ST	_	Interrupt-on-change input.			
RB4/ANB4/ADACT ⁽¹⁾ /IOCB4	RB4	TTL/ST	CMOS/OD	General purpose I/O.			
	ANB4	AN	_	ADC Channel B4 input.			
	ADACT ⁽¹⁾	TTL/ST	_	ADC Auto-Conversion Trigger input.			
	IOCB4	TTL/ST	_	Interrupt-on-change input.			
RB5/ANB5/T1G ⁽¹⁾ /IOCB5	RB5	TTL/ST	CMOS/OD	General purpose I/O.			
	ANB5	AN	_	ADC Channel B5 input.			
	T1G ⁽¹⁾	ST	_	Timer1 Gate input.			
	IOCB5	TTL/ST	_	Interrupt-on-change input.			
RB6/ANB6/CLCIN2 ⁽¹⁾ /IOCB6/TX2/	RB6	TTL/ST	CMOS/OD	General purpose I/O.			
UKZ MUSPULK	ANB6	AN	-	ADC Channel B6 input.			
	CLCIN2 ⁽¹⁾	TTL/ST	_	Configurable Logic Cell source input.			
	IOCB6	TTL/ST	_	Interrupt-on-change input.			
	TX2	—	CMOS	EUSART2 asynchronous.			
	CK2 ⁽³⁾	TTL/ST	CMOS/OD	EUSART2 synchronous mode clock input/output.			
	ICSPCLK	ST	_	In-Circuit Serial Programming™ and debugging clock input.			
RB7/ANB7/RX2/DT2/CLCIN3 ⁽¹⁾ /	RB7	TTL/ST	CMOS/OD	General purpose I/O.			
IOCB/IDACIOUIZ/ICSFDAI	ANB7	AN	_	ADC Channel B7 input.			
	CLCIN3 ⁽¹⁾	TTL/ST	_	Configurable Logic Cell source input.			
	IOCB7	TTL/ST	_	Interrupt-on-change input.			
	RX2 ⁽¹⁾	TTL/ST	-	EUSART2 Asynchronous mode receiver data input.			
	DT2 ⁽³⁾	TTL/ST	CMOS/OD	EUSART2 Synchronous mode data input/output.			
	DAC10UT2	—	AN	Digital-to-Analog Converter output.			
	ICSPDAT	ST	CMOS	In-Circuit Serial Programming™ and debugging data input/ output.			
RC0/ANC0/T1CKI(1)/IOCC0/SOSCO	RC0	TTL/ST	CMOS/OD	General purpose I/O.			
	ANC0	AN	_	ADC Channel C0 input.			
	T1CKI ⁽¹⁾	TTL/ST	_	Timer1 external digital clock input.			
	IOCC0	TTL/ST	_	Interrupt-on-change input.			
	SOSCO	_	AN	32.768 kHz secondary oscillator crystal driver output.			
Legend: AN = Analog input or output CMOS = CMOS compatible input or output OD = Open-Drain							

TABLE 1-2: PIC16(L)F15354/55 PINOUT DESCRIPTION (CONTINUED)

Note

TTL = TTL compatible input ST = Schmitt Trigger input with CMOS levels I^2C = Schmitt Trigger input with I^2C HV = High Voltage XTAL = Crystal levels This is a PPS remappable input signal. The input function may be moved from the default location shown to one of several other PORTx pins. Refer to Table 15-2 for details on which PORT pins may be used for this signal. All output signals shown in this row are PPS remappable. These signals may be mapped to output onto one of several PORTx pin

1:

2:

options as described in Table 15-3. This is a bidirectional signal. For normal module operation, the firmware should map this signal to the same pin in both the PPS input and 3: PPS output registers.

These pins are configured for I²C logic levels. The SCLx/SDAx signals may be assigned to any of the RB1/RB2/RC3/RC4 pins. PPS 4: assignments to the other pins (e.g., RA5) will operate, but input logic levels will be standard TTL/ST, as selected by the INLVL register, instead of the I²C specific or SMBus input buffer thresholds.

2.3 Master Clear (MCLR) Pin

The MCLR pin provides two specific device functions: Device Reset, and Device Programming and Debugging. If programming and debugging are not required in the end application, a direct connection to VDD may be all that is required. The addition of other components, to help increase the application's resistance to spurious Resets from voltage sags, may be beneficial. A typical configuration is shown in Figure 2-1. Other circuit designs may be implemented, depending on the application's requirements.

During programming and debugging, the resistance and capacitance that can be added to the pin must be considered. Device programmers and debuggers drive the MCLR pin. Consequently, specific voltage levels (VIH and VIL) and fast signal transitions must not be adversely affected. Therefore, specific values of R1 and C1 will need to be adjusted based on the application and PCB requirements. For example, it is recommended that the capacitor, C1, be isolated from the MCLR pin during programming and debugging operations by using a jumper (Figure 2-2). The jumper is replaced for normal run-time operations.

Any components associated with the $\overline{\text{MCLR}}$ pin should be placed within 0.25 inch (6 mm) of the pin.

FIGURE 2-2: EXAMPLE OF MCLR PIN CONNECTIONS



2.4 ICSP[™] Pins

The PGC and PGD pins are used for In-Circuit Serial ProgrammingTM (ICSPTM) and debugging purposes. It is recommended to keep the trace length between the ICSP connector and the ICSP pins on the device as short as possible. If the ICSP connector is expected to experience an ESD event, a series resistor is recommended, with the value in the range of a few tens of ohms, not to exceed 100 Ω .

Pull-up resistors, series diodes and capacitors on the PGC and PGD pins are not recommended as they will interfere with the programmer/debugger communications to the device. If such discrete components are an application requirement, they should be removed from the circuit during programming and debugging. Alternatively, refer to the AC/DC characteristics and timing requirements information in the respective device Flash programming specification for information on capacitive loading limits, and pin input voltage high (VIH) and input low (VIL) requirements.

For device emulation, ensure that the "Communication Channel Select" (i.e., PGCx/PGDx pins), programmed into the device, matches the physical connections for the ICSP to the Microchip debugger/emulator tool.

For more information on available Microchip development tools connection requirements, refer to **Section 39.0 "Development Support"**.

3.0 ENHANCED MID-RANGE CPU

This family of devices contains an enhanced mid-range 8-bit CPU core. The CPU has 48 instructions. Interrupt capability includes automatic context saving.

FIGURE 3-1: CORE DATA PATH DIAGRAM



The hardware stack is 16-levels deep and has

Overflow and Underflow Reset capability. Direct,

Indirect, and Relative Addressing modes are available.

Two File Select Registers (FSRs) provide the ability to

read program and data memory.

8.1 Power-on Reset (POR)

The POR circuit holds the device in Reset until VDD has reached an acceptable level for minimum operation. Slow rising VDD, fast operating speeds or analog performance may require greater than minimum VDD. The PWRT, BOR or MCLR features can be used to extend the start-up period until all device operation conditions have been met.

8.2 Brown-out Reset (BOR)

The BOR circuit holds the device in Reset when VDD reaches a selectable minimum level. Between the POR and BOR, complete voltage range coverage for execution protection can be implemented.

The Brown-out Reset module has four operating modes controlled by the BOREN<1:0> bits in Configuration Words. The four operating modes are:

- · BOR is always on
- · BOR is off when in Sleep
- BOR is controlled by software
- BOR is always off

Refer to Table 8-1 for more information.

The Brown-out Reset voltage level is selectable by configuring the BORV bit in Configuration Words.

A VDD noise rejection filter prevents the BOR from triggering on small events. If VDD falls below VBOR for a duration greater than parameter TBORDC, the device will reset. See Figure 8-2 for more information.

BOREN<1:0>	SBOREN	Device Mode	BOR Mode	Instruction Execution upon: Release of POR or Wake-up from Sleep
11	х	Х	Active	Wait for release of BOR ⁽¹⁾ (BORRDY = 1)
10	X	Awake	Active	Waits for release of BOR (BORRDY = 1)
		Sleep	Disabled	Waits for BOR Reset release
01	1	х	Active	Waits for BOR Reset release (BORRDY = 1)
UL	0	х	Disabled	Paging immediately (POPPDY =)
00	х	х	Disabled	begins inimediately (BORRDY = x)

TABLE 8-1: BOR OPERATING MODES

Note 1: In this specific case, "Release of POR" and "Wake-up from Sleep", there is no delay in start-up. The BOR ready flag, (BORRDY = 1), will be set before the CPU is ready to execute instructions because the BOR circuit is forced on by the BOREN<1:0> bits.

8.2.1 BOR IS ALWAYS ON

When the BOREN bits of Configuration Words are programmed to '11', the BOR is always on. The device start-up will be delayed until the BOR is ready and VDD is higher than the BOR threshold.

BOR protection is active during Sleep. The BOR does not delay wake-up from Sleep.

8.2.2 BOR IS OFF IN SLEEP

When the BOREN bits of Configuration Words are programmed to '10', the BOR is on, except in Sleep. The device start-up will be delayed until the BOR is ready and VDD is higher than the BOR threshold.

BOR protection is not active during Sleep. The device wake-up will be delayed until the BOR is ready.

8.14 Power Control (PCONx) Registers

The Power Control (PCONx) registers contain flag bits to differentiate between a:

- Power-on Reset (POR)
- Brown-out Reset (BOR)
- Reset Instruction Reset (RI)
- MCLR Reset (RMCLR)
- Watchdog Timer Reset (RWDT)
- Watchdog Timer Window Violation Reset
 (WDTWV)
- Stack Underflow Reset (STKUNF)
- Stack Overflow Reset (STKOVF)
- Memory Violation Reset (MEMV)

The PCON0 register bits are shown in Register 8-3. The PCON1 register bits are shown in Register 8-3.

Hardware will change the corresponding register bit during the Reset process; if the Reset was not caused by the condition, the bit remains unchanged (Table 8-4).

Software should reset the bit to the inactive state after the restart (hardware will not reset the bit).

Software may also set any PCON bit to the active state, so that user code may be tested, but no reset action will be generated.

9.0 OSCILLATOR MODULE (WITH FAIL-SAFE CLOCK MONITOR)

9.1 Overview

The oscillator module has a wide variety of clock sources and selection features that allow it to be used in a wide range of applications while maximizing performance and minimizing power consumption. Figure 9-1 illustrates a block diagram of the oscillator module.

Clock sources can be supplied from external oscillators, quartz-crystal resonators and ceramic resonators. In addition, the system clock source can be supplied from one of two internal oscillators and PLL circuits, with a choice of speeds selectable via software. Additional clock features include:

- Selectable system clock source between external or internal sources via software.
- Fail-Safe Clock Monitor (FSCM) designed to detect a failure of the external clock source (LP, XT, HS, ECH, ECM, ECL) and switch automatically to the internal oscillator.
- Oscillator Start-up Timer (OST) ensures stability of crystal oscillator sources.

The RSTOSC bits of Configuration Word 1 determine the type of oscillator that will be used when the device reset, including when it is first powered up.

The internal clock modes, LFINTOSC, HFINTOSC (set at 1 MHz), or HFINTOSC (set at 32 MHz) can be set through the RSTOSC bits.

If an external clock source is selected, the FEXTOSC bits of Configuration Word 1 must be used in conjunction with the RSTOSC bits to select the external clock mode.

The external oscillator module can be configured in one of the following clock modes, by setting the FEXTOSC<2:0> bits of Configuration Word 1:

- 1. ECL External Clock Low-Power mode ECL<= 500 kHz
- 2. ECM External Clock Medium Power mode ECM <= 8 MHz
- 3. ECH External Clock High-Power mode ECH <= 32 MHz
- 4. LP 32 kHz Low-Power Crystal mode.
- 5. XT Medium Gain Crystal or Ceramic Resonator Oscillator mode (between 100 kHz and 4 MHz)
- 6. HS High Gain Crystal or Ceramic Resonator mode (above 4 MHz)

The ECH, ECM, and ECL clock modes rely on an external logic level signal as the device clock source. The LP, XT, and HS clock modes require an external crystal or resonator to be connected to the device. Each mode is optimized for a different frequency range. The INTOSC internal oscillator block produces low and high-frequency clock sources, designated LFINTOSC and HFINTOSC. (see Internal Oscillator Block, Figure 9-1). A wide selection of device clock frequencies may be derived from these clock sources.

14.4 PORTB Registers

14.4.1 DATA REGISTER

PORTB is an 8-bit wide, bidirectional port. The corresponding data direction register is TRISB (Register 14-10). Setting a TRISB bit (= 1) will make the corresponding PORTB pin an input (i.e., disable the output driver). Clearing a TRISB bit (= 0) will make the corresponding PORTB pin an output (i.e., enables output driver and puts the contents of the output latch on the selected pin). Figure 14-1 shows how to initialize PORTB.

Reading the PORTB register (Register 14-9) reads the status of the pins, whereas writing to it will write to the PORT latch. All write operations are read-modify-write operations. Therefore, a write to a port implies that the port pins are read, this value is modified and then written to the PORT data latch (LATB).

The PORT data latch LATB (Register 14-11) holds the output port data, and contains the latest value of a LATB or PORTB write.

14.4.2 DIRECTION CONTROL

The TRISB register (Register 14-10) controls the PORTB pin output drivers, even when they are being used as analog inputs. The user should ensure the bits in the TRISB register are maintained set when using them as analog inputs. I/O pins configured as analog inputs always read '0'.

14.4.3 OPEN-DRAIN CONTROL

The ODCONB register (Register 14-14) controls the open-drain feature of the port. Open-drain operation is independently selected for each pin. When an ODCONB bit is set, the corresponding port output becomes an open-drain driver capable of sinking current only. When an ODCONB bit is cleared, the corresponding port output pin is the standard push-pull drive capable of sourcing and sinking current.

Note:	It is not necessary to set open-drain control when using the pin for I ² C; the I ² C
	module controls the pin and makes the pin open-drain.

14.4.4 SLEW RATE CONTROL

The SLRCONB register (Register 14-15) controls the slew rate option for each port pin. Slew rate control is independently selectable for each port pin. When an SLRCONB bit is set, the corresponding port pin drive is slew rate limited. When an SLRCONB bit is cleared, The corresponding port pin drive slews at the maximum rate possible.

14.4.5 INPUT THRESHOLD CONTROL

The INLVLB register (Register 14-8) controls the input voltage threshold for each of the available PORTB input pins. A selection between the Schmitt Trigger CMOS or the TTL Compatible thresholds is available. The input threshold is important in determining the value of a read of the PORTB register and also the level at which an interrupt-on-change occurs, if that feature is enabled. See Table 37-4 for more information on threshold levels.

Note: Changing the input threshold selection should be performed while all peripheral modules are disabled. Changing the threshold level during the time a module is active may inadvertently generate a transition associated with an input pin, regardless of the actual voltage level on that pin.

14.4.6 ANALOG CONTROL

The ANSELB register (Register 14-12) is used to configure the Input mode of an I/O pin to analog. Setting the appropriate ANSELA bit high will cause all digital reads on the pin to be read as '0' and allow analog functions on the pin to operate correctly.

The state of the ANSELB bits has no effect on digital output functions. A pin with its TRIS bit clear and its ANSEL bit set will still operate as a digital output, but the Input mode will be analog. This can cause unexpected behavior when executing read-modify-write instructions on the affected port.

Note:	The ANSELB bits default to the Analog									
	mode after Reset. To use any pins as									
	digital general purpose or peripheral									
	inputs, the corresponding ANSEL bits									
	must be initialized to '0' by user software.									

14.4.7 WEAK PULL-UP CONTROL

The WPUB register (Register 14-5) controls the individual weak pull-ups for each PORT pin.

14.4.8 PORTB FUNCTIONS AND OUTPUT PRIORITIES

Each PORTB pin is multiplexed with other functions.

Each pin defaults to the PORT latch data after Reset. Other output functions are selected with the peripheral pin select logic or by enabling an analog output, such as the DAC. See **Section 15.0** "**Peripheral Pin Select (PPS) Module**" for more information.

Analog input functions, such as ADC and comparator inputs are not shown in the peripheral pin select lists. Digital output functions may continue to control the pin when it is in Analog mode.

REGISTER 14-15: SLRCONB: PORTB SLEW RATE CONTROL REGISTER
--

| R/W-1/1 |
|---------|---------|---------|---------|---------|---------|---------|---------|
| SLRB7 | SLRB6 | SLRB5 | SLRB4 | SLRB3 | SLRB2 | SLRB1 | SLRB0 |
| bit 7 | | | | | | | bit 0 |
| | | | | | | | |
| Logond | | | | | | | |

Legena:		
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 **SLRB<7:0>:** PORTB Slew Rate Enable bits For RB<7:0> pins, respectively 1 = Port pin slew rate is limited 0 = Port pin slews at maximum rate

REGISTER 14-16: INLVLB: PORTB INPUT LEVEL CONTROL REGISTER

| R/W-1/1 |
|---------|---------|---------|---------|---------|---------|---------|---------|
| INLVLB7 | INLVLB6 | INLVLB5 | INLVLB4 | INLVLB3 | INLVLB2 | INLVLB1 | INLVLB0 |
| 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 INLVLB<7:0>: PORTB Input Level Select bits For RB<7:0> pins, respectively

 $\ensuremath{\mathtt{1}}$ = ST input used for PORT reads and interrupt-on-change

0 = TTL input used for PORT reads and interrupt-on-change

TABLE 14-3: SUMMARY OF REGISTERS ASSOCIATED WITH PORTB

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	178
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	178
LATB	LATB7	LATB6	LATB5	LATB4	LATB3	LATB2	LATB1	LATB0	179
ANSELB	ANSB7	ANSB6	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1	ANSB0	179
WPUB	WPUB7	WPUB6	WPUB5	WPUB4	WPUB3	WPUB2	WPUB1	WPUB0	180
ODCONB	ODCB7	ODCB6	ODCB5	ODCB4	ODCB3	ODCB2	ODCB1	ODCB0	180
SLRCONB	SLRB7	SLRB6	SLRB5	SLRB4	SLRB3	SLRB2	SLRB1	SLRB0	181
INLVLB	INLVLB7	INLVLB6	INLVLB5	INLVLB4	INLVLB3	INLVLB2	INLVLB1	INLVLB0	181

Legend: x = unknown, u = unchanged, – = unimplemented locations read as '0'. Shaded cells are not used by PORTB.

Note 1: Unimplemented, read as '1'.

TABLE 15-4: S	UMMARY OF REGISTERS ASSOCIA	TED WITH THE PPS MODULE (CONTINUED)
---------------	-----------------------------	---------------------------	------------

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
RC1PPS	—	-	—			RC1PPS<	4:0>		197
RC2PPS	—	_	—		RC2PPS<4:0>				
RC3PPS	—	_	—		RC3PPS<4:0>				
RC4PPS	—	_	—		RC4PPS<4:0>				
RC5PPS	—	_	—		RC5PPS<4:0>				197
RC6PPS	—	_	—	RC6PPS<4:0>				197	
RC7PPS	—	_	—			RC7PPS<	4:0>		197

Legend: — = unimplemented, read as '0'. Shaded cells are unused by the PPS module.

REGISTER 17-7: IOCO	CP: INTERRUPT-ON-CHANGE PORTC POSITIVE EDGE 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
IOCCP7	IOCCP6	IOCCP5	IOCCP4	IOCCP3	IOCCP2	IOCCP1	IOCCP0
bit 7							bit 0
Legend:							
R = Readable bit W = Writable bit		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				

bit 7-0

'1' = Bit is set

IOCCP<7:0>: Interrupt-on-Change PORTC Positive Edge Enable bits

- 1 = Interrupt-on-Change enabled on the pin for a positive-going edge. IOCCFx bit and IOCIF flag will be set upon detecting an edge.
- 0 = Interrupt-on-Change disabled for the associated pin

'0' = Bit is cleared

REGISTER 17-8: IOCCN: INTERRUPT-ON-CHANGE PORTC NEGATIVE EDGE REGISTER

| R/W-0/0 |
|---------|---------|---------|---------|---------|---------|---------|---------|
| IOCCN7 | IOCCN6 | IOCCN5 | IOCCN4 | IOCCN3 | IOCCN2 | IOCCN1 | IOCCN0 |
| 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

IOCCN<7:0>: Interrupt-on-Change PORTC Negative Edge Enable bits

- 1 = Interrupt-on-Change enabled on the pin for a negative-going edge. IOCCFx bit and IOCIF flag will be set upon detecting an edge.
- 0 = Interrupt-on-Change disabled for the associated pin

REGISTER 17-9: IOCCF: INTERRUPT-ON-CHANGE PORTC FLAG REGISTER

| R/W/HS-0/0 |
|------------|------------|------------|------------|------------|------------|------------|------------|
| IOCCF7 | IOCCF6 | IOCCF5 | IOCCF4 | IOCCF3 | IOCCF2 | IOCCF1 | IOCCF0 |
| 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	HS - Bit is set in hardware

bit 7-0

- IOCCF<7:0>: Interrupt-on-Change PORTC Flag bits
- 1 = An enabled change was detected on the associated pin
 - Set when IOCCPx = 1 and a rising edge was detected on RCx, or when IOCCNx = 1 and a falling edge was detected on RCx.
- 0 = No change was detected, or the user cleared the detected change

23.3 Comparator Hysteresis

A selectable amount of separation voltage can be added to the input pins of each comparator to provide a hysteresis function to the overall operation. Hysteresis is enabled by setting the CxHYS bit of the CMxCON0 register.

See Comparator Specifications in Table 37-14 for more information.

23.4 Timer1 Gate Operation

The output resulting from a comparator operation can be used as a source for gate control of Timer1. See **Section 26.6 "Timer Gate"** for more information. This feature is useful for timing the duration or interval of an analog event.

It is recommended that the comparator output be synchronized to Timer1. This ensures that Timer1 does not increment while a change in the comparator is occurring.

23.4.1 COMPARATOR OUTPUT SYNCHRONIZATION

The output from a comparator can be synchronized with Timer1 by setting the CxSYNC bit of the CMxCON0 register.

Once enabled, the comparator output is latched on the falling edge of the Timer1 source clock. If a prescaler is used with Timer1, the comparator output is latched after the prescaling function. To prevent a race condition, the comparator output is latched on the falling edge of the Timer1 clock source and Timer1 increments on the rising edge of its clock source. See the Comparator Block Diagram (Figure 23-2) and the Timer1 Block Diagram (Figure 26-1) for more information.

23.5 Comparator Interrupt

An interrupt can be generated upon a change in the output value of the comparator for each comparator, a rising edge detector and a falling edge detector are present.

When either edge detector is triggered and its associated enable bit is set (CxINTP and/or CxINTN bits of the CMxCON1 register), the Corresponding Interrupt Flag bit (CxIF bit of the PIR2 register) will be set.

To enable the interrupt, you must set the following bits:

- CxON, CxPOL and CxSP bits of the CMxCON0 register
- CxIE bit of the PIE2 register
- CxINTP bit of the CMxCON1 register (for a rising edge detection)
- CxINTN bit of the CMxCON1 register (for a falling edge detection)
- · PEIE and GIE bits of the INTCON register

The associated interrupt flag bit, CxIF bit of the PIR2 register, must be cleared in software. If another edge is detected while this flag is being cleared, the flag will still be set at the end of the sequence.

Note: Although a comparator is disabled, an interrupt can be generated by changing the output polarity with the CxPOL bit of the CMxCON0 register, or by switching the comparator on or off with the CxON bit of the CMxCON0 register.

23.6 Comparator Positive Input Selection

Configuring the CxPCH<2:0> bits of the CMxPSEL register directs an internal voltage reference or an analog pin to the noninverting input of the comparator:

- CxIN0+ analog pin
- DAC output
- FVR (Fixed Voltage Reference)
- · Vss (Ground)

See **Section 18.0** "**Fixed Voltage Reference (FVR)**" for more information on the Fixed Voltage Reference module.

See Section 21.0 "5-Bit Digital-to-Analog Converter (DAC1) Module" for more information on the DAC input signal.

Any time the comparator is disabled (CxON = 0), all comparator inputs are disabled.

23.7 Comparator Negative Input Selection

The CxNCH<2:0> bits of the CMxCON1 register direct an analog input pin and internal reference voltage or analog ground to the inverting input of the comparator:

- CxIN- pin
- FVR (Fixed Voltage Reference)
- Analog Ground

Note: To use CxINy+ and CxINy- pins as analog input, the appropriate bits must be set in the ANSEL register and the corresponding TRIS bits must also be set to disable the output drivers.

PIC16(L)F15354/55

REGISTER 23-3: CMxNSEL: COMPARATOR Cx NEGATIVE INPUT SELECT REGISTER

U-0	U-0	U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	_	—	—		NCH<2:0>	
bit 7							bit 0
Legend:							

Legenu.		
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 7-2	Unimplemented: Read as '0'
bit 2-0	NCH<2:0>: Comparator Negative Input Channel Select bits
	111 = CxVN connects to AVss
	110 = CxVN connects to FVR Buffer 2
	101 = CxVN unconnected
	100 = CxVN unconnected
	011 = CxVN connects to CxIN3- pin
	010 = CxVN connects to CxIN2- pin
	001 = CxVN connects to CxIN1- pin
	000 = CxVN connects to CxIN0- pin

REGISTER 23-4: CMxPSEL: COMPARATOR Cx POSITIVE INPUT SELECT REGISTER

U-0	U-0	U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0
—	_	—	—	—		PCH<2:0>	
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 7-2 Unimplemented: Read as '0'

bit 5-3 PCH<2:0>: Comparator Positive Input Channel Select bits

- 111 = CxVP connects to AVss
- 110 = CxVP connects to FVR Buffer 2
- 101 = CxVP connects to DAC output
- 100 = CxVP unconnected
- 011 = CxVP unconnected
- 010 = CxVP unconnected
- 001 = CxVP connects to CxIN1+ pin
- 000 = CxVP connects to CxIN0+ pin

24.9 Register Definitions: ZCD Control

R/W-q/q	U-0	R-x/x	R/W-0/0	U-0	U-0	R/W-0/0	R/W-0/0
SEN	—	OUT	POL	—	—	INTP	INTN
bit 7							bit 0
Legend:							
R = Readable	bit	W = Writable	bit	U = Unimpler	mented bit, read	as '0'	
u = Bit is uncha	anged	x = Bit is unkr	nown	-n/n = Value a	at POR and BO	R/Value at all o	ther Resets
'1' = Bit is set		'0' = Bit is cle	ared	q = value dep	ends on Config	uration bits	
bit 7	SEN: Zero-Cr	oss Detection	Enable bit				
	1 = Zero-cros	ss detect is ena	abled. ZCD pi	n is forced to o	utput to source	and sink currer	nt.
	0 = Zero-cros	ss detect is dis	abled. ZCD pi	n operates acc	ording to PPS a	and TRIS contro	ols.
bit 6	Unimplemen	ted: Read as '	0'				
bit 5	OUT: Zero-Cr	oss Detection	Logic Level bi	t			
	<u>POL bit = 1</u> :						
	1 = ZCD pin	is sourcing cur	rent				
	$0 = \angle CD pin$	is sinking curre	ent				
	1 = 7CD pin	is sinkina curre	ent				
	0 = ZCD pin is sourcing current						
bit 4	POL: Zero-Cr	oss Detection	Logic Output	Polarity bit			
	1 = ZCD logi	c output is inve	rted				
	0 = ZCD logi	c output is not	inverted				
bit 3-2	Unimplemen	ted: Read as '	0'				
bit 1	INTP: Zero-C	ross Positive E	dge Interrupt	Enable bit			
	1 = ZCDIF bit is set on low-to-high ZCDx_output transition						
	0 = ZCDIF bi	t is unaffected	by low-to-high	n ZCDx_output	transition		
bit 0	INTN: Zero-Cross Negative Edge Interrupt Enable bit						
	1 = ZCDIF bi	t is set on high	-to-low ZCDx	_output transiti	on		
	0 = ZCDIF bi	t is unaffected	by high-to-lov	v ZCDx_output	transition		

REGISTER 24-1: ZCDCON: ZERO-CROSS DETECTION CONTROL REGISTER

TABLE 24-1: SUMMARY OF REGISTERS ASSOCIATED WITH THE ZCD MODULE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
PIE3	RC2IE	TX2IE	RC1IE	TX1IE	BCL2IE	SSP2IE	BCL1IE	SSP1IE	123
PIR3	RC2IF	TX2IF	RC1IF	TX1IF	BCL2IF	SSP2IF	BCL1IF	SSP1IF	131
ZCDxCON	EN		OUT	POL	_		INTP	INTN	265

Legend: — = unimplemented, read as '0'. Shaded cells are unused by the ZCD module.

TABLE 24-2: SUMMARY OF CONFIGURATION WORD WITH THE ZCD MODULE

Name	Bits	Bit -/7	Bit -/6	Bit 13/5	Bit 12/4	Bit 11/3	Bit 10/2	Bit 9/1	Bit 8/0	Register on Page
	13:8	_	_	DEBUG	STVREN	PPS1WAY	ZCDDIS	BORV	_	76
CONFIG2	7:0	BOREN	N <1:0>	LPBOREN		_		PWRTE	MCLRE	

Legend: — = unimplemented location, read as '0'. Shaded cells are not used by the ZCD module.

PIC16(L)F15354/55



FIGURE 26-5: TIMER1 GATE SINGLE-PULSE MODE





FIGURE 27-12: RISING EDGE-TRIGGERED MONOSTABLE MODE TIMING DIAGRAM (MODE = 10001)

29.1.1 PWM CLOCK SELECTION

The PIC16(L)F15354/55 allows each individual CCP and PWM module to select the timer source that controls the module. Each module has an independent selection.

The CCPTMRS0 and CCPTMRS1 register are used to select which timer is used.

29.1.2 USING THE TMR2 WITH THE PWM MODULE

This device has a newer version of the TMR2 module that has many new modes, which allow for greater customization and control of the PWM signals than on older parts. Refer to **Section 27.5** "**Operation Examples**" for examples of PWM signal generation using the different modes of Timer2.

Note:	PWM operation requires that the timer
	used as the PWM time base has the
	FOSC/4 clock source selected.

29.1.3 PWM PERIOD

Referring to Figure 29-1, the PWM output has a period and a pulse width. The frequency of the PWM is the inverse of the period (1/period).

The PWM period is specified by writing to the PR2 register. The PWM period can be calculated using the following formula:

EQUATION 29-1: PWM PERIOD

 $PWM Period = [(PR2) + 1] \cdot 4 \cdot TOSC$ $\cdot (TMR2 Prescale Value)$

Note 1: Tosc = 1/Fosc

When TMR2 is equal to PR2, the following three events occur on the next increment cycle:

- TMR2 is cleared
- The PWMx pin is set (Exception: If the PWM duty cycle = 0%, the pin will not be set.)
- The PWM pulse width is latched from PWMxDC.

Note: If the pulse width value is greater than the period the assigned PWM pin(s) will remain unchanged.

29.1.4 PWM DUTY CYCLE

The PWM duty cycle is specified by writing a 10-bit value to the PWMxDC register. The PWMxDCH contains the eight MSbs and the PWMxDCL<7:6> bits contain the two LSbs.

The PWMDC register is double-buffered and can be updated at any time. This double buffering is essential for glitch-free PWM operation. New values take effect when TMR2 = PR2. Note that PWMDC is left-justified.

The 8-bit timer TMR2 register is concatenated with either the 2-bit internal system clock (FOSC), or two bits of the prescaler, to create the 10-bit time base. The system clock is used if the Timer2 prescaler is set to 1:1.

Equation 29-2 is used to calculate the PWM pulse width.

Equation 29-3 is used to calculate the PWM duty cycle ratio.

EQUATION 29-2: PULSE WIDTH

Pulse Width = (PWMxDC) · TOSC · (TMR2 Prescale Value)

EQUATION 29-3: DUTY CYCLE RATIO

 $Duty Cycle Ratio = \frac{(PWMxDC)}{4(PR2+1)}$

29.1.5 PWM RESOLUTION

The resolution determines the number of available duty cycles for a given period. For example, a 10-bit resolution will result in 1024 discrete duty cycles, whereas an 8-bit resolution will result in 256 discrete duty cycles.

The maximum PWM resolution is ten bits when PR2 is 255. The resolution is a function of the PR2 register value as shown by Equation 29-4.

EQUATION 29-4: PWM RESOLUTION

Resolution = $\frac{\log[4(PR2+1)]}{\log(2)}$ bits

PIC16(L)F15354/55

REGISTER 30-8: CWG1CLK: CWG1 CLOCK SELECTION REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0/0
—	—	—	—	—	_	—	CS
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 7-1 Unimplemented: Read as '0'

bit 0

bit 3-0

CS: CWG1 Clock Selection bit

- 1 = HFINTOSC 16 MHz is selected
 - 0 = Fosc is selected

REGISTER 30-9: CWG1ISM: CWG1 INPUT SELECTION REGISTER

U-0	U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	—		IS<	3:0>	
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 7-4 Unimplemented: Read as '0'

IS<3:0>:	CWG1 Input Selection bits
1111 =	Reserved. No channel connected.
1110 =	Reserved. No channel connected.
1101 =	LC4_out.
1100 =	LC3_out.
1011 =	LC2_out.
1010 =	LC1_out
1001 =	Comparator C2 out
1000 =	Comparator C1 out.
0111 =	NCO1 output.
0110 =	PWM6_out
0101 =	PWM5_out
0100 =	PWM4_out
0011 =	PWM3_out
0010 =	CCP2_out
0001 =	CCP1_out
0000 =	CWG11CLK

PIC16(L)F15354/55



39.6 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.

39.7 MPLAB REAL ICE In-Circuit Emulator System

The MPLAB REAL ICE In-Circuit Emulator System is Microchip's next generation high-speed emulator for Microchip Flash DSC and MCU devices. It debugs and programs all 8, 16 and 32-bit MCU, and DSC devices with the easy-to-use, powerful graphical user interface of the MPLAB X IDE.

The emulator is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with either a connector compatible with in-circuit debugger systems (RJ-11) or with the new high-speed, noise tolerant, Low-Voltage Differential Signal (LVDS) interconnection (CAT5).

The emulator is field upgradeable through future firmware downloads in MPLAB X IDE. MPLAB REAL ICE offers significant advantages over competitive emulators including full-speed emulation, run-time variable watches, trace analysis, complex breakpoints, logic probes, a ruggedized probe interface and long (up to three meters) interconnection cables.

39.8 MPLAB ICD 3 In-Circuit Debugger System

The MPLAB ICD 3 In-Circuit Debugger System is Microchip's most cost-effective, high-speed hardware debugger/programmer for Microchip Flash DSC and MCU devices. It debugs and programs PIC Flash microcontrollers and dsPIC DSCs with the powerful, yet easy-to-use graphical user interface of the MPLAB IDE.

The MPLAB ICD 3 In-Circuit Debugger probe is connected to the design engineer's PC using a highspeed USB 2.0 interface and is connected to the target with a connector compatible with the MPLAB ICD 2 or MPLAB REAL ICE systems (RJ-11). MPLAB ICD 3 supports all MPLAB ICD 2 headers.

39.9 PICkit 3 In-Circuit Debugger/ Programmer

The MPLAB PICkit 3 allows debugging and programming of PIC and dsPIC Flash microcontrollers at a most affordable price point using the powerful graphical user interface of the MPLAB IDE. The MPLAB PICkit 3 is connected to the design engineer's PC using a fullspeed USB interface and can be connected to the target via a Microchip debug (RJ-11) connector (compatible with MPLAB ICD 3 and MPLAB REAL ICE). The connector uses two device I/O pins and the Reset line to implement in-circuit debugging and In-Circuit Serial Programming[™] (ICSP[™]).

39.10 MPLAB PM3 Device Programmer

The MPLAB PM3 Device Programmer is a universal, CE compliant device programmer with programmable voltage verification at VDDMIN and VDDMAX for maximum reliability. It features a large LCD display (128 x 64) for menus and error messages, and a modular, detachable socket assembly to support various package types. The ICSP cable assembly is included as a standard item. In Stand-Alone mode, the MPLAB PM3 Device Programmer can read, verify and program PIC devices without a PC connection. It can also set code protection in this mode. The MPLAB PM3 connects to the host PC via an RS-232 or USB cable. The MPLAB PM3 has high-speed communications and optimized algorithms for quick programming of large memory devices, and incorporates an MMC card for file storage and data applications.