TB3275

Using the CCP Module

Introduction

The Capture/Compare/PWM (CCP) module is a peripheral that allows the measurement and control of different events and also generates pulse-width modulated (PWM) output signals. In Capture mode, the module can be used to measure the duration of an event. Compare mode allows the user to trigger an external event after a predetermined amount of time has expired. PWM mode is used to generate pulsed output signals that may vary in frequency and duty cycle. This technical brief describes each of the three CCP modes.

© 2020 Microchip Technology Inc.

Table of Contents

© 2020 Microchip Technology Inc.

1. CCP Module Overview

1.1 CCP Module Configuration

Each CCP module is associated with a CCP Control register (CCPxCON), a CCP Capture Input Selection register (CCPxCAP) and a CCP Data register pair (CCPRxH:CCPRxL). The CCPxCAP register is used only in Capture mode and selects the input signal that will be measured. The CCPRx register pair is used in each CCP mode of operation. In Capture mode, the CCPRx register pair stores the captured value from the specified signal. In Compare mode, the CCPRx register pair holds the value for comparison while in PWM mode, this register is used to configure the duty cycle of the output signal.

1.2 CCP Timer Resources

The CCP modules utilize both the 8-bit and 16-bit timers, depending on the CCP mode. Table 1-1 shows the timer resources available to the CCP in Capture, Compare and PWM modes.

All the CCP modules can be active at once and share the same timer resource, so long as they are configured to operate in the same mode at the same time. For example, if CCP1 and CCP2 are both operating in PWM mode, they both can use Timer2 as the timer resource.

Table 1-1. CCP Timer Resources

CCP Mode	Timer Resource
Capture	Timer0 (16-bit mode), Timer1, Timer3 or Timer5
Compare	
PWM	Timer2, Timer4 or Timer6

Important: Timer1 is the default timer resource for Capture and Compare modes while Timer2 is the default resource for PWM mode.

1.2.1 CCP Timer Selection

Many devices have multiple instances of both 16-bit and 8-bit timers. For those devices, each CCP module can be configured to use an independent timer resource via the CCP Timers Selection (CCPTMRSx) register. For example, if the device has two CCP modules, CCP1 operating in Compare mode may select Timer1 as the clock source, whereas CCP2 operating in Compare mode may select Timer3 as the clock source.

Important: Devices with a single 8/16-bit timer resource will not have a CCP Timer Selection register. Instead, Timer1 is the default resource for Capture and Compare modes while Timer2 is the resource for PWM mode.

1.3 Open-Drain Output Option

When operating in Compare or PWM modes, the output drivers for the CCPx pins can be configured optionally as open-drain outputs by setting the associated Open-Drain Control (ODCONx) register bit. The open-drain feature allows the voltage level on the pin to be pulled to a higher level through an external pull-up resistor, which removes the need for additional level shifters when communicating with external circuits.

1.4 Effects of Reset

Any Reset will force all ports to Input mode and the CCP registers to their Reset states.

2. Capture Mode

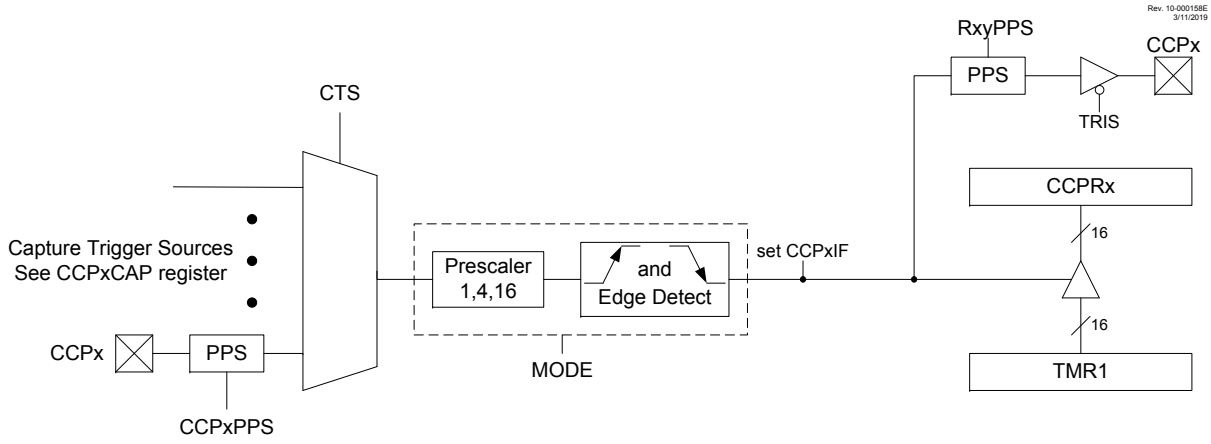
Capture mode uses the 16-bit odd numbered timer resources (Timer1, Timer3, etc.). When an event occurs on the capture source, the 16-bit CCPRx register captures and stores the 16-bit value of the TMRx register. The valid events that can be detected when using the CCP module in Capture mode are defined in the following list and are configured using the MODE bits:

- Every falling edge of CCPx input
- Every rising edge of CCPx input
- Every 4th rising edge of CCPx input
- Every 16th rising edge of CCPx input
- Every Edge of CCPx input (rising and falling)

When a capture is made, the CCP Interrupt Flag (CCPxIF) bit is set. The interrupt flag must be cleared in software. If another capture occurs before the value in the CCPRx register is read, the old captured value is overwritten by the new captured value. Figure 2-1 shows a simplified diagram of the capture operation.

Important: If an event occurs during a 2-byte read, the high and low-byte data will be from different events. It is recommended that, while reading the CCPRx register pair, to either disable the module or read the register pair twice for data integrity.

Figure 2-1. Capture Mode Operation Block Diagram



2.1 Capture Sources

The capture source is selected with the Capture Trigger Input Selection (CTS) bits. The capture source may come from an internal source, such as a CLC or comparator output, or from an external source routed through the CCPx pin via the CCPxPPS input selection register.

In Capture mode, the CCPx pin can be configured as an input by setting the associated TRIS control bit.

Important: If the CCPx pin is configured as an output, a write to the port can cause a capture condition.

2.2 Timer1 Mode for Capture

Timer1 must be operating synchronously to the system clock for the CCP module to use the capture feature. When Timer1 uses the internal instruction clock ($F_{OSC}/4$) as its clock source, clock synchronization is handled by module hardware automatically. When Timer1 uses an external clock source, the Timer External Clock Input Synchronization Control (SYNC) bit must be clear (SYNC = 0) so that the external source is synchronized with the system clock. If the Timer clock source is not synchronized, the capture operation may not work.

See the **"TMR1 - Timer1 Module with Gate Control"** chapter in the device data sheet for more information on configuring Timer1 for use with the CCP module.

Important: Do not clock Timer1 from the system clock (F_{OSC}) in Capture mode. In order for Capture mode to recognize the trigger event, Timer1 must be clocked from the instruction clock or from an external clock source.

2.3 Software Interrupt Mode

The CCP Interrupt Flag (CCPxIF) bit is set with every capture. If the device is in Sleep and the CCP Interrupt Enable (CCPxIE) bit is set, the device will wake up.

When the Capture mode is changed, a false capture interrupt may be generated. Software should keep the CCPxIE bit clear (CCPxIE = 0) to avoid false interrupts. Additionally, the user can clear the CCPxIF bit following any change in Operating mode.

2.4 Capture Mode Prescaler

There are four prescaler settings specified by the CCP Mode Select (MODE) bits where bits that determine which input edge triggers a Capture event. A Capture event can be configured to occur every:

- 16th rising edge of the CCP input
- 4th rising edge of the CCP input
- Rising or falling edge of the CCP input
- Edge (rising and falling) of the CCP input

Important: When capturing events on every edge (both rising and falling), the input trigger source must be operating at a frequency less than the instruction clock frequency and at least one instruction period must be between capture events. If the trigger source frequency is greater than the instruction clock frequency, a capture event may be missed, resulting in invalid data.

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 CCPxCON register before changing the prescaler. Example 2-1 demonstrates the code to perform this function.

Example 2-1. Changing Between Capture Prescalers

```
void UpdatePrescale(void)
{
    CCP1CONbits.EN = 0;              // Disable CCP
    CCP1CONbits.MODE = 0b0100;       // Update with new prescale
    CCP1CONbits.EN = 1;              // Re-enable CCP
}
```

2.5 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 ($F_{OSC}/4$) or by an external clock source.

When Timer1 is clocked by $F_{OSC}/4$, 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.

2.6 Capture Mode Configuration

Example 2-2 shows the CCP configured in Capture mode. The example contains the CCP Initialization and Capture Interrupt Service Routines, as well as the Timer1 Initialization routine. The software demonstrates how the CCP module can be configured to measure a continuous series of pulses using Timer1 as the time base. The measurement occurs on every edge of the incoming signal. Timer1 is configured to use the '$F_{OSC}/4$' clock source.

Example 2-2. Capture Mode Initialization and Interrupt Service Routines

```
static void (*CCP1_CallBack)(uint16_t);
static void CCP1_DefaultCallBack(uint16_t capturedValue)
{
    // Add your code here
}

void CCP1_Initialize(void)
{
    CCP1CON = 0x83;                     // MODE Every edge; FMT right;
    CCP1CAP = 0x00;                     // CCP1CTS CCP1 pin;
    CCPR1H = 0x00;
    CCPR1L = 0x00;
    CCP1_SetCallBack(CCP1_DefaultCallBack);  // Set call back function
    PIR1bits.CCP1IF = 0;                // Clear CCP1 interrupt flag
    PIE1bits.CCP1IE = 1;                // Enable the CCP1 interrupt
}

void CCP1_CaptureISR(void)
{
    CCP1_PERIOD_REG_T module;
    PIR1bits.CCP1IF = 0;                // Clear interrupt flag
    module.ccpr1l = CCPR1L;             // Copy captured value
    module.ccpr1h = CCPR1H;
    CCP1_CallBack(module.ccpr1_16Bit);  // Return 16bit captured value
}

void CCP1_SetCallBack(void (*customCallBack)(uint16_t))
{
    CCP1_CallBack = customCallBack;
}

void TMR1_Initialize(void)
{
    T1GCON = 0x00;                    // T1GE disabled;
    T1GATE = 0x00;
    T1CLK = 0x01;                     // CS FOSC/4;
    TMR1H = 0x00;
    TMR1L = 0x00;
    PIR1bits.TMR1IF = 0;              // Clear IF flag;
    T1CON = 0x33;                     // CKPS 1:8; ON enabled; RD16 enabled;
}
```

3. Compare Mode

Compare mode makes use of the 16-bit odd numbered Timer resources (Timer1, Timer3, etc.). The 16-bit value of the CCPRx register is constantly compared against the 16-bit value of the TMRx register. When a match occurs, one of the following events can occur based on the configuration of the MODE control bits:

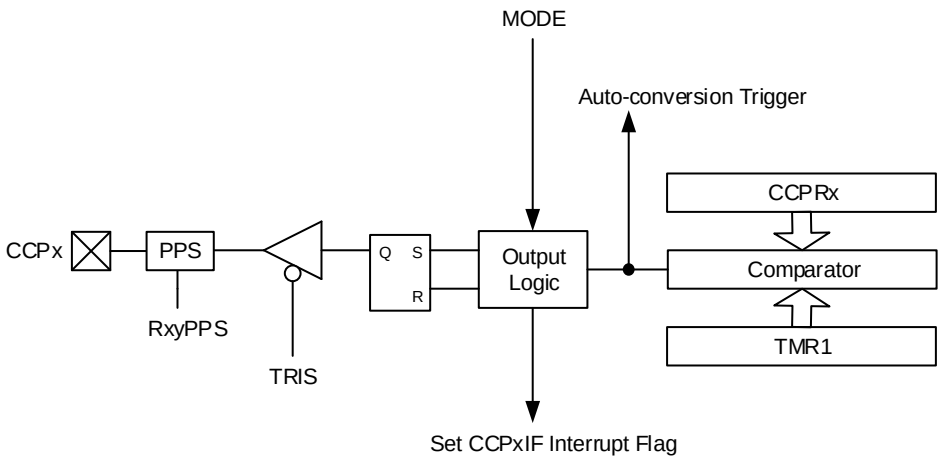
- Toggle the CCPx output and clear TMRx.
- Toggle the CCPx output without clearing TMRx.
- Set the CCPx output.
- Clear the CCPx output.
- Generate a pulse output.
- Generate a pulse output and clear TMRx.

In Compare mode, Timer1 acts as a timebase or as a counter. For example, if Timer1 is configured so that the comparison match point occurs every 50 ms, CCP hardware will generate an output event at a repeatable interval. If Timer1 is configured as a counter, CCP hardware will generate an output event based on the number of times Timer1 records an event.

Compare mode hardware can also clear Timer1 after certain events occur. This feature can be useful when operating Timer1 as a fixed time base.

Figure 3-1 shows a simplified diagram of the compare operation.

Figure 3-1. Compare Mode Operation Block Diagram



3.1 CCPx Pin Configuration

The CCPx pin must be configured as an output in software by clearing the associated TRIS bit and defining the appropriate output pin through the RxyPPS registers. See the **“PPS - Peripheral Pin Select Module”** chapter in the device data sheet for more details.

The CCP output can also be used as an input for other peripherals. For example, the CCP output can be used as an optional Timer1 Gate trigger input source.

Important: Clearing the CCPxCON register will force the CCPx compare output latch to the default low level. This is not the PORT I/O data latch.

3.2 Timer1 Mode for Compare

Timer1 must be operating synchronously to the system clock for the CCP module to use the compare feature. When Timer1 uses the internal instruction clock ($F_{OSC}/4$) as its clock source, clock synchronization is handled by module hardware automatically. When Timer1 uses an external clock source, the Timer External Clock Input Synchronization Control (SYNC) bit must be clear (SYNC = 0) so that the external source is synchronized with the system clock. If the Timer clock source is not synchronized, a clock collision may occur and the comparison may be invalid.

See the **"TMR1 - Timer1 Module with Gate Control"** chapter in the device data sheet for more information on configuring Timer1.

Important: Timer1 should not be clocked from the system clock (F_{OSC}) in Compare mode. In order for Compare mode to recognize the trigger event on the CCPx pin, Timer1 must be clocked from the instruction clock ($F_{OSC}/4$) or from an external clock source.

3.3 Compare Mode Interrupts

The CCP Interrupt Flag (CCPxIF) bit is set with every comparison match. If the device is in Sleep and the CCP Interrupt Enable (CCPxIE) bit is set, the device will wake up. CCPxIF must be cleared in software.

3.4 Compare During Sleep

Similar to Capture mode, Compare mode also depends upon the Timer1 module for proper operation. There are two options for driving the Timer1 module in Compare mode. It can be driven by the instruction clock ($F_{OSC}/4$) or by an external clock source.

When Timer1 is clocked by $F_{OSC}/4$, Timer1 will not increment during Sleep because the system clock is disabled. When the device wakes from Sleep, Timer1 will continue from its previous state.

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

3.5 Compare Mode Configuration

Example 3-1 shows the CCP and Timer1 Initialization routines for Compare mode. The CCPRx register pair is loaded with value 0x3CB0. When Timer1 reaches 0x3CB0, which occurs approximately every 50 ms, CCP hardware generates a pulsed output and clears Timer1.

Example 3-1. Compare Mode Initialization Routines

```
void CCP1_Initialize(void)
{
    CCP1CON = 0x8B;              // MODE Pulse-clear-timer
    CCPR1H = 0x3C;               // When TMR1 = 0x3CB0 = 50 ms
    CCPR1L = 0xB0;
}

void TMR1_Initialize(void)
{
    T1GCON = 0x00;
    T1GATE = 0x00;
    T1CLK = 0x01;                // CS FOSC/4
    TMR1H = 0x00;
    TMR1L = 0x00;
    PIR1bits.TMR1IF = 0;         // Clear IF flag
    T1CON = 0x01;                // CKPS 1:1; ON enabled
}
```

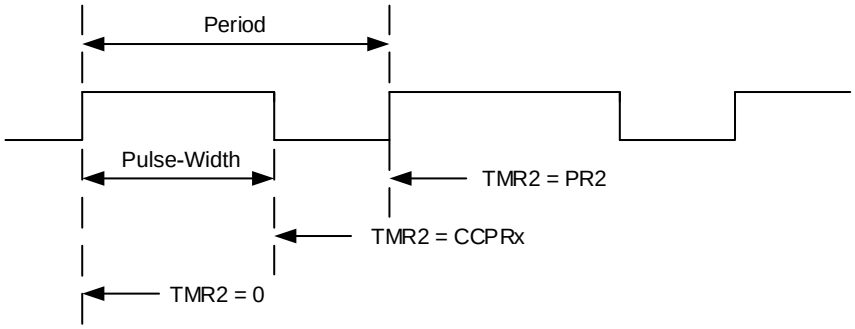
4. PWM Mode

Pulse-Width Modulation (PWM) is a scheme that controls power to a load by switching quickly between fully active and fully inactive states. The PWM signal resembles a square wave where the high portion of the signal is considered the active state while the low portion of the signal is considered the inactive state. The high portion, also known as the pulse-width, can vary in time and is defined in steps. A larger number of steps applied, which lengthens the pulse-width, also supplies more power to the load. Lowering the number of steps applied, which shortens the pulse-width, supplies less power. The PWM period is defined as the duration of one complete cycle or the total amount of active and inactive time combined.

PWM resolution defines the maximum number of steps present in a single PWM period. A higher resolution allows for more precise control of the pulse-width time and, in turn, the power that is applied to the load.

The term duty cycle describes the proportion of the active time to the inactive time and is expressed in percentages, where 0% is fully inactive and 100% is fully active. A lower duty cycle corresponds to less power applied while a higher duty cycle corresponds to more power applied. The figure below shows a typical waveform of the PWM signal.

Figure 4-1. CCP PWM Output Signal



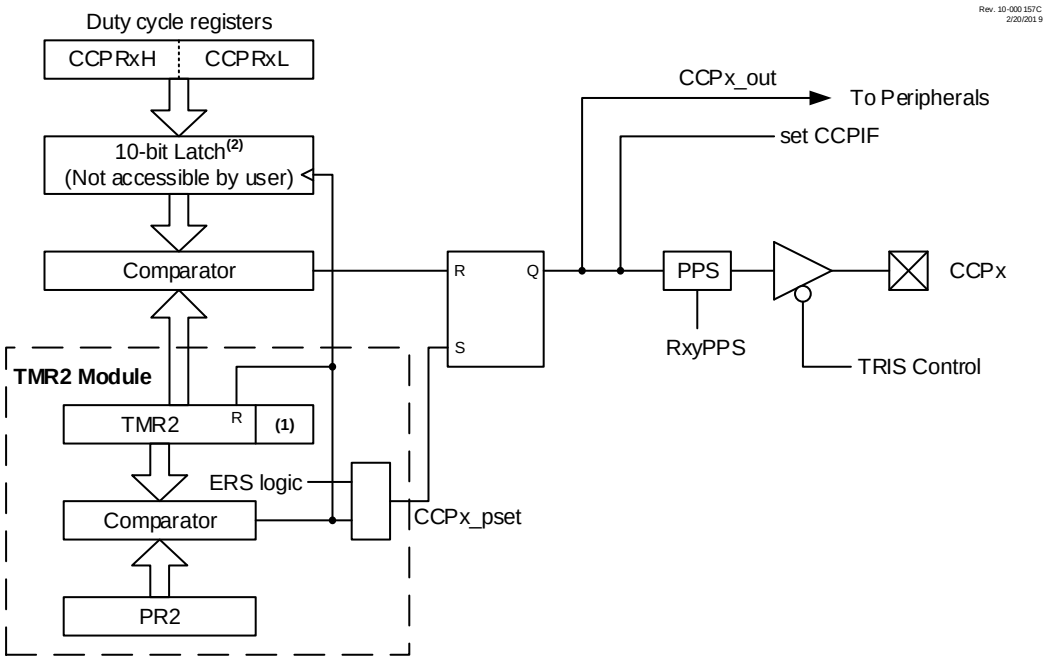
4.1 Standard PWM Operation

The standard PWM function described in this section is available and identical for all CCP modules. It generates a PWM signal on the CCPx pin with up to ten bits of resolution. The period, duty cycle and resolution are controlled by the following registers:

- Even numbered TxPR registers (T2PR, T4PR, etc.)
- Even numbered TxCON registers (T2CON, T4CON, etc.)
- 16-bit CCPRx registers
- CCPxCON registers

It is required to have $F_{OSC}/4$ as the clock input to the selected CCP timer resource for correct PWM operation. The following figure shows a simplified block diagram of PWM operation.

Figure 4-2. Simplified PWM Block Diagram



Notes: 1. 8-bit timer is concatenated with two bits generated by Fosc or two bits of the internal prescaler to create 10-bit time-base.
2. The alignment of the 10 bits from the CCPR register is determined by the CCPxFMT bit.

Important: The corresponding TRIS bit must be cleared and the CCPx pin must be configured through the appropriate RxyPPS register to enable the PWM output.

4.2 PWM Mode Timer Resource

PWM mode makes use of the even numbered 8-bit timers (Timer2, Timer4, etc.) to specify the PWM period. For correct PWM operation, Timer2 must use the instruction clock ($F_{OSC}/4$) as its clock source. To achieve 10-bit PWM resolution while using an 8-bit timer as the time base, module hardware combines the 8-bit timer with two additional bits that are either generated by the system clock (F_{OSC}) or copied from an internal prescaler, creating a 10-bit time base.

4.3 PWM Period

The PWM period is specified by the T2PR register of Timer2. The PWM period can be calculated using the formula in Equation 4-1.

Equation 4-1. PWM Period

$$PWM\ Period = [(T2PR + 1)] \bullet 4 \bullet T_{OSC} \bullet (TMR2\ Prescale\ Value)$$

where:

$$TOSC = \frac{1}{FOSC}$$

When T2TMR is equal to T2PR, the following three events occur on the next increment event:

- T2TMR is cleared.
- The CCPx pin is set. (Exception: If the PWM duty cycle = 0%, the pin will not be set.)
- The PWM duty cycle is transferred from the CCPRx register into a 10-bit buffer.

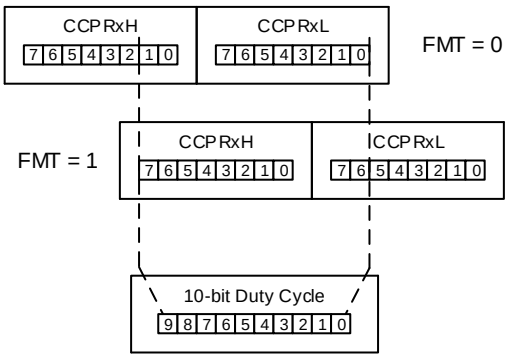
Important: The Timer postscaler is not used in the determination of the PWM frequency.

4.4 PWM Duty Cycle

The PWM duty cycle is specified by writing a 10-bit value to the CCPRx register. The alignment of the 10-bit value is determined by the CCP Pulse-Width Value Alignment (FMT) bit (see Figure 4-3). The CCPRx register can be written to at any time. However, the duty cycle value is not latched into the 10-bit buffer until after a match between T2PR and T2TMR.

Equation 4-2 and Equation 4-3 are used to calculate the PWM pulse-width and the PWM duty cycle ratio, respectively.

Figure 4-3. PWM 10-Bit Alignment



Equation 4-2. Pulse-Width

$$Pulse\ Width = (CCPRxH{:}CCPRxL\ \ register\ value) \bullet T_{OSC} \bullet (TMR2\ Prescale\ Value)$$

Equation 4-3. Duty Cycle

$$Duty\ Cycle\ Ratio = \frac{(CCPRxH{:}CCPRxL\ register\ value)}{4(T2PR + 1)}$$

The CCPRx register is used to double buffer the PWM duty cycle. This double buffering is essential for glitchless PWM operation.

The 8-bit timer T2TMR register is concatenated with either the 2-bit internal system clock (F_{OSC}) or two bits of the prescaler, creating the 10-bit time base. The system clock is used if the Timer2 prescaler is set to 1:1.

When the 10-bit time base matches the CCPRx register, the CCPx pin is cleared.

4.5 PWM Resolution

The resolution determines the number of available duty cycles for a given period. For example, a 10-bit resolution will result in 1,024 discrete duty cycles while an 8-bit resolution will result in 256 discrete duty cycles.

The maximum PWM resolution is ten bits when T2PR is 0xFF. The resolution is a function of the T2PR register value as shown in Equation 4-4.

Equation 4-4. PWM Resolution

$$Resolution = \frac{log\,[4\,(T2PR + 1)]}{log\,(2)}\ bits$$

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

Table 4-1. Example PWM Frequencies and Resolutions (F_{OSC} = 20 MHz)

PWM Frequency	1.22 kHz	4.88 kHz	19.53 kHz	78.12 kHz	156.3 kHz	208.3 kHz
Timer Prescale	16	4	1	1	1	1
T2PR Value	0xFF	0xFF	0xFF	0x3F	0x1F	0x17
Maximum Resolution (bits)	10	10	10	8	7	6

Table 4-2. Example PWM Frequencies and Resolutions (F_{OSC} = 8 MHz)

PWM Frequency	1.22 kHz	4.90 kHz	19.61 kHz	76.92 kHz	153.85 kHz	200.0 kHz
Timer Prescale	16	4	1	1	1	1
T2PR Value	0x65	0x65	0x65	0x19	0x0C	0x09
Maximum Resolution (bits)	8	8	8	6	5	5

4.6 Operation in Sleep Mode

In Sleep mode, the T2TMR register will not increment and the state of the module will not change since the timer's clock source is suspended during Sleep. If the CCPx pin is driving a value, it will continue to drive that value. When the device wakes up, T2TMR will continue from the previous state.

4.7 Changes in System Clock Frequency

The PWM frequency is derived from the system clock frequency. Any changes in the system clock frequency will result in changes to the PWM frequency. See the **"OSC - Oscillator Module (with Fail-Safe Clock Monitor)"** chapter in the device data sheet for additional details.

4.8 PWM Mode Configuration

Example 4-1 shows the CCP and Timer2 Initialization routines. In this example, the CCPRx register pair is loaded with a value, `0x63`, which will result in a 50% duty cycle. Timer2 is configured to generate a PWM frequency of 20 kHz.

Example 4-1. PWM Mode Initialization Routines

```
void PWM1_Initialize(void)
{
    CCP1CON = 0x8F;                 // MODE PWM; EN enabled; FMT right
    CCPR1H = 0x00;                  // Set for 50% duty cycle
    CCPR1L = 0x63;
}

void TMR2_Initialize(void)
{
    T2CLKCON = 0x01;                // T2CS FOSC/4
    T2HLT = 0x00;
    T2RST = 0x00;
    T2PR = 0x31;                    // Set PWM Period
    T2TMR = 0x00;
    PIR1bits.TMR2IF = 0;            // Clear IF flag
    T2CON = 0x80;                   // T2CKPS 1:1; T2OUTPS 1:1; TMR2ON on
}
```

5. Conclusion

The Capture/Compare/PWM module is a peripheral that can be used for a variety of functions. Capture mode can be used to measure the duration of an event, such as measuring the rotational speed of a motor. Compare mode can be used to create a specific event at a certain time, such as generating an output pulse to quickly turn on/off a switch every second. PWM mode is used to generate pulse-width modulated output signals that can be used in a wide variety of applications, such as motor control or lighting control.

The Microchip Website

Microchip provides online support via our website at www.microchip.com/. This website is used to make files and information easily available to customers. Some of the content available includes:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQs), technical support requests, online discussion groups, Microchip design partner program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

Product Change Notification Service

Microchip's product change notification service helps keep customers current on Microchip products. Subscribers will receive email notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, go to www.microchip.com/pcn and follow the registration instructions.

Customer Support

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Embedded Solutions Engineer (ESE)
- Technical Support

Customers should contact their distributor, representative or ESE for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in this document.

Technical support is available through the website at: www.microchip.com/support

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Legal Notice

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with

your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE. Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Trademarks

The Microchip name and logo, the Microchip logo, Adaptec, AnyRate, AVR, AVR logo, AVR Freaks, BesTime, BitCloud, chipKIT, chipKIT logo, CryptoMemory, CryptoRF, dsPIC, FlashFlex, flexPWR, HELDO, IGLOO, JukeBlox, KeeLoq, Kleer, LANCheck, LinkMD, maXStylus, maXTouch, MediaLB, megaAVR, Microsemi, Microsemi logo, MOST, MOST logo, MPLAB, OptoLyzer, PackeTime, PIC, picoPower, PICSTART, PIC32 logo, PolarFire, Prochip Designer, QTouch, SAM-BA, SenGenuity, SpyNIC, SST, SST Logo, SuperFlash, Symmetricom, SyncServer, Tachyon, TempTrackr, TimeSource, tinyAVR, UNI/O, Vectron, and XMEGA are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

APT, ClockWorks, The Embedded Control Solutions Company, EtherSynch, FlashTec, Hyper Speed Control, HyperLight Load, IntelliMOS, Libero, motorBench, mTouch, Powermite 3, Precision Edge, ProASIC, ProASIC Plus, ProASIC Plus logo, Quiet-Wire, SmartFusion, SyncWorld, Temux, TimeCesium, TimeHub, TimePictra, TimeProvider, Vite, WinPath, and ZL are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Adjacent Key Suppression, AKS, Analog-for-the-Digital Age, Any Capacitor, AnyIn, AnyOut, BlueSky, BodyCom, CodeGuard, CryptoAuthentication, CryptoAutomotive, CryptoCompanion, CryptoController, dsPICDEM, dsPICDEM.net, Dynamic Average Matching, DAM, ECAN, EtherGREEN, In-Circuit Serial Programming, ICSP, INICnet, Inter-Chip Connectivity, JitterBlocker, KleerNet, KleerNet logo, memBrain, Mindi, MiWi, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, MultiTRAK, NetDetach, Omniscient Code Generation, PICDEM, PICDEM.net, PICkit, PICtail, PowerSmart, PureSilicon, QMatrix, REAL ICE, Ripple Blocker, SAM-ICE, Serial Quad I/O, SMART-I.S., SQI, SuperSwitcher, SuperSwitcher II, Total Endurance, TSHARC, USBCheck, VariSense, ViewSpan, WiperLock, Wireless DNA, and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

The Adaptec logo, Frequency on Demand, Silicon Storage Technology, and Symmcom are registered trademarks of Microchip Technology Inc. in other countries.

GestIC is a registered trademark of Microchip Technology Germany II GmbH & Co. KG, a subsidiary of Microchip Technology Inc., in other countries.

All other trademarks mentioned herein are property of their respective companies.

© 2020, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

ISBN: 978-1-5224-6404-4

Quality Management System

For information regarding Microchip's Quality Management Systems, please visit www.microchip.com/quality.

Worldwide Sales and Service

AMERICAS	ASIA/PACIFIC	ASIA/PACIFIC	EUROPE
Corporate Office 2355 West Chandler Blvd. Chandler, AZ 85224-6199 Tel: 480-792-7200 Fax: 480-792-7277 Technical Support: www.microchip.com/support Web Address: www.microchip.com **Atlanta** Duluth, GA Tel: 678-957-9614 Fax: 678-957-1455 **Austin, TX** Tel: 512-257-3370 **Boston** Westborough, MA Tel: 774-760-0087 Fax: 774-760-0088 **Chicago** Itasca, IL Tel: 630-285-0071 Fax: 630-285-0075 **Dallas** Addison, TX Tel: 972-818-7423 Fax: 972-818-2924 **Detroit** Novi, MI Tel: 248-848-4000 **Houston, TX** Tel: 281-894-5983 **Indianapolis** Noblesville, IN Tel: 317-773-8323 Fax: 317-773-5453 Tel: 317-536-2380 **Los Angeles** Mission Viejo, CA Tel: 949-462-9523 Fax: 949-462-9608 Tel: 951-273-7800 **Raleigh, NC** Tel: 919-844-7510 **New York, NY** Tel: 631-435-6000 **San Jose, CA** Tel: 408-735-9110 Tel: 408-436-4270 **Canada - Toronto** Tel: 905-695-1980 Fax: 905-695-2078	**Australia - Sydney** Tel: 61-2-9868-6733 **China - Beijing** Tel: 86-10-8569-7000 **China - Chengdu** Tel: 86-28-8665-5511 **China - Chongqing** Tel: 86-23-8980-9588 **China - Dongguan** Tel: 86-769-8702-9880 **China - Guangzhou** Tel: 86-20-8755-8029 **China - Hangzhou** Tel: 86-571-8792-8115 **China - Hong Kong SAR** Tel: 852-2943-5100 **China - Nanjing** Tel: 86-25-8473-2460 **China - Qingdao** Tel: 86-532-8502-7355 **China - Shanghai** Tel: 86-21-3326-8000 **China - Shenyang** Tel: 86-24-2334-2829 **China - Shenzhen** Tel: 86-755-8864-2200 **China - Suzhou** Tel: 86-186-6233-1526 **China - Wuhan** Tel: 86-27-5980-5300 **China - Xian** Tel: 86-29-8833-7252 **China - Xiamen** Tel: 86-592-2388138 **China - Zhuhai** Tel: 86-756-3210040	**India - Bangalore** Tel: 91-80-3090-4444 **India - New Delhi** Tel: 91-11-4160-8631 **India - Pune** Tel: 91-20-4121-0141 **Japan - Osaka** Tel: 81-6-6152-7160 **Japan - Tokyo** Tel: 81-3-6880- 3770 **Korea - Daegu** Tel: 82-53-744-4301 **Korea - Seoul** Tel: 82-2-554-7200 **Malaysia - Kuala Lumpur** Tel: 60-3-7651-7906 **Malaysia - Penang** Tel: 60-4-227-8870 **Philippines - Manila** Tel: 63-2-634-9065 **Singapore** Tel: 65-6334-8870 **Taiwan - Hsin Chu** Tel: 886-3-577-8366 **Taiwan - Kaohsiung** Tel: 886-7-213-7830 **Taiwan - Taipei** Tel: 886-2-2508-8600 **Thailand - Bangkok** Tel: 66-2-694-1351 **Vietnam - Ho Chi Minh** Tel: 84-28-5448-2100	**Austria - Wels** Tel: 43-7242-2244-39 Fax: 43-7242-2244-393 **Denmark - Copenhagen** Tel: 45-4485-5910 Fax: 45-4485-2829 **Finland - Espoo** Tel: 358-9-4520-820 **France - Paris** Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79 **Germany - Garching** Tel: 49-8931-9700 **Germany - Haan** Tel: 49-2129-3766400 **Germany - Heilbronn** Tel: 49-7131-72400 **Germany - Karlsruhe** Tel: 49-721-625370 **Germany - Munich** Tel: 49-89-627-144-0 Fax: 49-89-627-144-44 **Germany - Rosenheim** Tel: 49-8031-354-560 **Israel - Ra'anana** Tel: 972-9-744-7705 **Italy - Milan** Tel: 39-0331-742611 Fax: 39-0331-466781 **Italy - Padova** Tel: 39-049-7625286 **Netherlands - Drunen** Tel: 31-416-690399 Fax: 31-416-690340 **Norway - Trondheim** Tel: 47-72884388 **Poland - Warsaw** Tel: 48-22-3325737 **Romania - Bucharest** Tel: 40-21-407-87-50 **Spain - Madrid** Tel: 34-91-708-08-90 Fax: 34-91-708-08-91 **Sweden - Gothenberg** Tel: 46-31-704-60-40 **Sweden - Stockholm** Tel: 46-8-5090-4654 **UK - Wokingham** Tel: 44-118-921-5800 Fax: 44-118-921-5820