Output Compare with Dedicated Timer

HIGHLIGHTS

This section of the manual contains the following topics:

1.0 Introduction .................................................................................................................... 2
2.0 Output Compare Registers ............................................................................................ 4
3.0 Modes of Operation ...................................................................................................... 11
4.0 Output Compare Operation with DMA ......................................................................... 39
5.0 Output Compare Operation in Power-Saving States .................................................. 40
6.0 I/O Pin Control .......................................................................................................... 41
7.0 Register Maps .......................................................................................................... 42
8.0 Design Tips .............................................................................................................. 43
9.0 Related Application Notes ....................................................................................... 44
10.0 Revision History ....................................................................................................... 45
The Output Compare (OC) module in dsPIC33/PIC24 devices compares the Output Compare Timer register value with the value of one or two Compare registers, depending on its mode of operation. The OC module on compare match events can generate a single output transition or a series of output pulses. Like most PIC® MCU peripherals, the OC module can also generate interrupts on a compare match event.

Each Output Compare timer can use one of the available six selectable time clocks. The clock is selected using the Output Compare x Clock Select (OCTSEL<2:0>) bits in the Output Compare x Control Register 1 (OCxCON1<12:10>). For more information on specific timers that can be used as a time base for the Output Compare timer, refer to the specific device data sheet.

Figure 1-1 shows the block diagram of the OC module.

Note: For more information on the number of available Output Compare channels, refer to the specific device data sheet.

All of the Output Compare channels are functionally identical. In this section, an ‘x’ in the pin, register or bit name denotes the specific Output Compare channel.

The OCx output must be assigned to an available RPn pin before use if the device supports Peripheral Pin Select (PPS). For more information, refer to the “Peripheral Pin Select (PPS)” section in the specific device data sheet.
Output Compare with Dedicated Timer

Figure 1-1: Output Compare x Module Block Diagram (Double-Buffered, 16-Bit PWM Mode)

Note 1: The Trigger/Sync source is enabled by default and is set to Timer2 as a source. This timer must be enabled for proper OCx module operation or the Trigger/Sync source must be changed to another source option. If a timer source other than Timer2 has to be selected, then the SYNCSEL<4:0> bits in the OCxCON2 register have to be set to the corresponding value, before enabling the OCx module.
2.0 OUTPUT COMPARE REGISTERS

This section outlines the specific functions of each register that controls the operation of the Output Compare with Dedicated Timer module. The registers are as follows:

- **OCxCON1**: Output Compare x Control Register 1 and **OCxCON2**: Output Compare x Control Register 2(4)
  These are the OCx Control registers for the Output Compare channel.

- **OCxR**: Output Compare x Register
  This is the OCx Data register for the Output Compare channel.

- **OCxRS**: Output Compare x Secondary Register
  This is the OCx Secondary Data register for the Output Compare channel.

- **OCxTMR**: Output Compare x Timer Register
  This is the OCx Internal Time Base register for the Output Compare channel.

All of the control registers have identical bit definitions and are represented by common register definitions as listed in Register 2-1 to Register 2-5.

**Note:** Each dsPIC33/PIC24 family device variant may have one or more Output Compare with Dedicated Timer modules. An ‘x’ used in the names of pins, control/status bits and registers denote the particular Output Compare channel number. Refer to the “Output Compare with Dedicated Timer” chapter in the specific device data sheet for more details.
### Output Compare with Dedicated Timer

**Register 2-1: OCxCON1: Output Compare x Control Register 1**

<table>
<thead>
<tr>
<th>Bit 15-14</th>
<th>Unimplemented: Read as ‘0’</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bit 13</td>
<td>OCSIDL: Output Compare x Stop in Idle Mode Control bit</td>
</tr>
<tr>
<td></td>
<td>1 = Output Compare x halts in CPU Idle mode</td>
</tr>
<tr>
<td></td>
<td>0 = Output Compare x continues to operate in CPU Idle mode</td>
</tr>
<tr>
<td>Bit 12-10</td>
<td>OCTSEL&lt;2:0&gt;: Output Compare x Clock Select bits</td>
</tr>
<tr>
<td></td>
<td>111 = Peripheral clock (FCY)</td>
</tr>
<tr>
<td></td>
<td>110 = Reserved</td>
</tr>
<tr>
<td></td>
<td>101 = Reserved for PTGOx(4)</td>
</tr>
<tr>
<td></td>
<td>100 = Timer1 clock (only the synchronous clock is supported)</td>
</tr>
<tr>
<td></td>
<td>011 = Timer5 clock</td>
</tr>
<tr>
<td></td>
<td>010 = Timer4 clock</td>
</tr>
<tr>
<td></td>
<td>001 = Timer3 clock</td>
</tr>
<tr>
<td></td>
<td>000 = Timer2 clock</td>
</tr>
<tr>
<td>Bit 9</td>
<td>ENFLT2/ENFLTC: Fault 2/C Input Enable bit(3)</td>
</tr>
<tr>
<td></td>
<td>1 = Fault inputs are enabled</td>
</tr>
<tr>
<td></td>
<td>0 = Fault inputs are disabled</td>
</tr>
<tr>
<td>Bit 8</td>
<td>ENFLT1/ENFLTB: Fault 1/B Input Enable bit(3)</td>
</tr>
<tr>
<td></td>
<td>1 = Fault inputs are enabled</td>
</tr>
<tr>
<td></td>
<td>0 = Fault inputs are disabled</td>
</tr>
<tr>
<td>Bit 7</td>
<td>ENFLT0/ENFLTA: Fault 0/A Input Enable bit (corresponds to the OCFA pin)</td>
</tr>
<tr>
<td></td>
<td>1 = Fault inputs are enabled</td>
</tr>
<tr>
<td></td>
<td>0 = Fault inputs are disabled</td>
</tr>
<tr>
<td>Bit 6</td>
<td>OCFLT2/OCFLTC: Output Compare x PWM Fault 2/C Condition Status bit(3)</td>
</tr>
<tr>
<td></td>
<td>1 = PWM Fault condition has occurred</td>
</tr>
<tr>
<td></td>
<td>0 = PWM Fault condition has not occurred</td>
</tr>
<tr>
<td>Bit 5</td>
<td>OCFLT1/OCFLTB: Output Compare x PWM Fault 1/B Condition Status bit(3)</td>
</tr>
<tr>
<td></td>
<td>1 = PWM Fault condition has occurred</td>
</tr>
<tr>
<td></td>
<td>0 = PWM Fault condition has not occurred</td>
</tr>
</tbody>
</table>

**Legend:**

- **HCS** = Hardware Clearable/Settable bit
- **R** = Readable bit
- **W** = Writable bit
- **U** = Unimplemented bit, read as ‘0’
- **-n** = Value at POR
- **‘1’** = Bit is set
- **‘0’** = Bit is cleared
- **x** = Bit is unknown

---

**Note 1:** The OCx output must also be configured to an available RPn pin if the device supports Peripheral Pin Select (PPS). For more information, refer to the specific device data sheet.

**Note 2:** The OCxR and OCxRS registers are double-buffered only in PWM modes.

**Note 3:** Refer to the specific device data sheet to find the Fault bits mapping.

**Note 4:** This mode is not available on all devices. Refer to the “Output Compare” chapter of the specific device data sheet.
Register 2-1:    OCxCON1: Output Compare x Control Register 1 (Continued)

bit 4  OCFLT0/OCFLT A: Output Compare x PWM Fault 0/A Condition Status bit
       1 = PWM Fault condition has occurred
       0 = PWM Fault condition has not occurred

bit 3  TRIGMODE: Trigger Status Mode Select bit
       1 = TRIGSTAT (OCxCON2<6>) bit is cleared when OCxRS = OCxTMR or in software
       0 = TRIGSTAT (OCxCON2<6>) bit is cleared only by software

bit 2-0  OCM<2:0>: Output Compare x Mode Select bits\(^{(1)}\)
       111 = Center-Aligned PWM mode: Output is set high when OCxTMR = OCxR and is set low when
            OCxTMR = OCxRS\(^{(2)}\)
       110 = Edge-Aligned PWM mode: Output is set high when OCxTMR = 0 and is set low when
            OCxTMR = OCxR\(^{(2)}\)
       101 = Double Compare Continuous Pulse mode: Initializes the OCx pin low and toggles the OCx state
            continuously on alternate matches of OCxR and OCxRS
       100 = Double Compare Single-Shot mode: Initializes the OCx pin low and toggles the OCx state on
            matches of OCxR and OCxRS for one cycle
       011 = Single Compare mode: Compares events with OCxR and continuously toggles the OCx pin
       010 = Single Compare Single-Shot mode: Initializes the OCx pin high, compares event with OCxR and
            forces OCx pin low
       001 = Single Compare Single-Shot mode: Initializes the OCx pin low, compares event with OCxR and
            forces OCx pin high
       000 = Output Compare channel is disabled

Note 1:  The OCx output must also be configured to an available RPn pin if the device supports Peripheral Pin
        Select (PPS). For more information, refer to the specific device data sheet.
2:  The OCxR and OCxRS registers are double-buffered only in PWM modes.
3:  Refer to the specific device data sheet to find the Fault bits mapping.
4:  This mode is not available on all devices. Refer to the "Output Compare" chapter of the specific device
     data sheet.
Output Compare with Dedicated Timer

Register 2-2:  OCxCON2: Output Compare x Control Register 2(4)

<table>
<thead>
<tr>
<th>Bit 15</th>
<th>Bit 14</th>
<th>Bit 13</th>
<th>Bit 12</th>
<th>Bit 11</th>
<th>Bit 10-9</th>
<th>Bit 8</th>
<th>Bit 7</th>
<th>Bit 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>FLTMD</td>
<td>FLTOUT</td>
<td>FLTTRIEN</td>
<td>OCINV</td>
<td>—-</td>
<td>DCB1(3)</td>
<td>DCB0(3)</td>
<td>OC32</td>
<td></td>
</tr>
</tbody>
</table>

Legend:  
HS = Hardware Settable bit  
R = Readable bit  
W = Writable bit  
U = Unimplemented bit, read as ‘0’  
-n = Value at POR ‘1’ = Bit is set ‘0’ = Bit is cleared  
x = Bit is unknown

bit 15  
**FLTMD**: Fault Mode Select bit  
1 = Fault mode is maintained until the Fault source is removed; the corresponding OCFLTx bit is cleared in software and a new PWM period starts  
0 = Fault mode is maintained until the Fault source is removed and a new PWM period starts

bit 14  
**FLTOUT**: Fault Out bit  
1 = PWM output is driven high on a Fault  
0 = PWM output is driven low on a Fault

bit 13  
**FLTTRIEN**: Fault Output State Select bit  
1 = OCx pin is tri-stated on a Fault condition  
0 = OCx pin I/O state is defined by the FLTOUT bit on a Fault condition

bit 12  
**OCINV**: OCMP Invert bit  
1 = OCx output is inverted  
0 = OCx output is not inverted

bit 11  
**Unimplemented**: Read as ‘0’

bit 10-9  
**DCB<1:0>**: PWM Duty Cycle Least Significant bits(3)  
These bits can be considered as the two Least Significant bits (LSbs) of the duty cycle in the Pulse Generation modes. They are also used to delay the falling edge of the OCx output in all other modes and the rising edge when the output inversion is active (OCINV bit (OCxCON2<12>) = 1).  
00 = OCx output falling edge transitions on the rising edge of the P1 clock (Legacy mode)  
01 = OCx output falling edge transitions on the rising edge of the P2 clock  
10 = OCx output falling edge transitions on the rising edge of the P3 clock  
11 = OCx output falling edge transitions on the rising edge of the P4 clock  
The DCB<1:0> bits can be used to generate LSBs of the PWM with a resolution of Peripheral Clock/2.

**Note 1:**  
Never use an OCx (SYNCSEL<4:0> = 11111, selecting the same OCx module) module as a Trigger/  
Synchronization source when the OCTTRIG bit = 1. The source selected by the SYNCSEL<4:0> bits must  
operate from the same clock source when the OCTTRIG bit = 0. When OCTTRIG = 1, the source selected by  
the SYNCSELx bits can be asynchronous. The trigger source needs to create an event pulse which will  
trigger the time base when a rising edge has been detected.

2:  
The inputs, SYNCSEL<4:0> = Input Capture x (ICx), should only be used as trigger sources and not as Sync  
sources.

3:  
The duty cycle Least Significant bits (DCB<1:0>) in the OCxCON2<10:9> register are double-buffered in  
PWM modes only (OCM<2:0> bits (OCxCON1<2:0>) = 111, 110). This feature is not implemented in all the  
devices; refer to the specific device data sheet for more information.

4:  
These bits are not available on all devices; refer to the specific device data sheet for availability.

5:  
When an Output Compare x module (OCx) is disabled, it sends a trigger out signal. If a second Output  
Compare y module (OCy) uses OCx as a Trigger or a Synchronization source, it must first deselect the OCx  
as its trigger source before OCx is disabled to avoid receiving an erroneous signal.
Register 2-2:  OCxCON2: Output Compare x Control Register 2\(^{(4)}\) (Continued)

bit 8  \textbf{OC32}: Cascade Two Output Compare x Modules Enable bit (32-bit operation)
\begin{itemize}
  \item 1 = Cascade module operation is enabled
  \item 0 = Cascade module operation is disabled
\end{itemize}

bit 7  \textbf{OCTRIG}: Output Compare x Trigger/Sync Select bit
\begin{itemize}
  \item 1 = Triggers the OCx from the source designated by the SYNCSSELx bits
  \item 0 = Synchronizes the OCx with the source designated by the SYNCSSELx bits
\end{itemize}

bit 6  \textbf{TRIGSTAT}: Timer Trigger Status bit
\begin{itemize}
  \item 1 = Timer source has been triggered and is running
  \item 0 = Timer source has not been triggered and is being held clear
\end{itemize}

bit 5  \textbf{OCTRIS}: Output Compare x Output Pin Direction Select bit
\begin{itemize}
  \item 1 = OCx is tri-stated
  \item 0 = OCx module drives the OCx pin
\end{itemize}

bit 4-0  \textbf{SYNCSEL<4:0>:} Trigger/Synchronization Source Selection bits\(^{(1,2,5)}\)
These bits select the CMPx, ADCx, ICx, OCx, PTGOx and Timerx inputs as Trigger/Synchronization sources. Refer to the “Output Compare” chapter in the specific device data sheet for availability.

\textbf{Note 1:}  Never use an OCx (SYNCSEL<4:0> = 11111, selecting the same OCx module) module as a Trigger/Synchronization source when the OCTTRIG bit = 1. The source selected by the SYNCSEL<4:0> bits must operate from the same clock source when the OCTRIG bit = 0. When OCTRIG = 1, the source selected by the SYNCSSELx bits can be asynchronous. The trigger source needs to create an event pulse which will trigger the time base when a rising edge has been detected.

\textbf{2:}  The inputs, SYNCSSEL<4:0> = Input Capture x (ICx), should only be used as trigger sources and not as Sync sources.

\textbf{3:}  The duty cycle Least Significant bits (DCB<1:0>) in the OCxCON2<10:9> register are double-buffered in PWM modes only (OCM<2:0> bits (OCxCON1<2:0>) = 111, 110). This feature is not implemented in all the devices; refer to the specific device data sheet for more information.

\textbf{4:}  These bits are not available on all devices; refer to the specific device data sheet for availability.

\textbf{5:}  When an Output Compare x module (OCx) is disabled, it sends a trigger out signal. If a second Output Compare y module (OCy) uses OCx as a Trigger or a Synchronization source, it must first deselect the OCx as its trigger source before OCx is disabled to avoid receiving an erroneous signal.
### Register 2-3: OCxR: Output Compare x Register

<table>
<thead>
<tr>
<th></th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>OCRB&lt;15:8&gt;</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bit 15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Register 2-4: OCxRS: Output Compare x Secondary Register

<table>
<thead>
<tr>
<th></th>
<th>R/W-0</th>
<th>R/W-0, HS</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>OCRSB&lt;7:0&gt;</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bit 7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Legend:**

- R = Readable bit
- W = Writable bit
- U = Unimplemented bit, read as '0'
- -n = Value at POR
- ‘1’ = Bit is set
- ‘0’ = Bit is cleared
- x = Bit is unknown

**bit 15-0 OCRB<15:0>: Output Compare x Primary Compare Register Value bits**

- When OCM<2:0> = 0b110, the register is used for the duty cycle in an Edge-Aligned PWM mode.
- When OCM<2:0> = 0b111, 0b101 or 0b100, the register is used for generating a positive edge.
- When OCM<2:0> = 0b001, 0b010 or 0b011, the register is used for generating all edges.

**Register 2-4: OCxRS: Output Compare x Secondary Register**

<table>
<thead>
<tr>
<th></th>
<th>R/W-0</th>
<th>R/W-0, HS</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>OCRSB&lt;15:8&gt;</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bit 15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Legend:

- R = Readable bit
- W = Writable bit
- U = Unimplemented bit, read as '0'
- -n = Value at POR
- ‘1’ = Bit is set
- ‘0’ = Bit is cleared
- x = Bit is unknown

**bit 15-0 OCSR<15:0>: Output Compare x Secondary Register Value bits**

- This is the Period Register:
  - If SYNCSEL<4:0> bits (OCxCON2<4:0>) = 0x1F.
  - If SYNCSEL<4:0> bits (OCxCON2<4:0>) = N (where 'N' is the alternate value to select this as the Period register).
  - If OCTRIG (OCxCON2<7>) = 1.
  - All Other Conditions:
    - The period is determined outside the OCx module. Used for generating a negative edge when the OCM<2:0> bits = 0b111, 0b101 or 0b100.
### Register 2-5: OCxTMR: Output Compare x Timer Register

<table>
<thead>
<tr>
<th>Bit 15</th>
<th>Bit 14</th>
<th>Bit 13</th>
<th>Bit 12</th>
<th>Bit 11</th>
<th>Bit 10</th>
<th>Bit 9</th>
<th>Bit 8</th>
<th>Bit 7</th>
<th>Bit 6</th>
<th>Bit 5</th>
<th>Bit 4</th>
<th>Bit 3</th>
<th>Bit 2</th>
<th>Bit 1</th>
<th>Bit 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td>R-0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Legend:**

- **R** = Readable bit
- **W** = Writable bit
- **U** = Unimplemented bit, read as ‘0’
- **-n** = Value at POR
- **‘1’** = Bit is set
- **‘0’** = Bit is cleared
- **x** = Bit is unknown

**bit 15-0 TMRB<15:0>: Output Compare x Timer bits**

The current value of the Output Compare x timer.
3.0 MODES OF OPERATION

The Output Compare module comprises the following operating modes:

- Single Compare Match mode
- Dual Compare Match mode which generates:
  - Single Output Pulse
  - Continuous Output Pulse
- Simple Pulse-Width Modulation mode with/without Fault Protection:
  - Edge-Aligned
  - Center-Aligned
- Cascade mode (32-bit operation)

Prior to understanding the modes, it is necessary to understand the Synchronization/Trigger mechanism. In Synchronous operation, the internal timer resets to zero when the source selected by the Trigger/Synchronization Source Selection (SYNCSEL<4:0>) bits (OCxCON2<4:0>) sends a Sync signal. In Trigger mode, the internal timer is held in the Reset state until the selected Trigger source sends a Sync signal.

The Synchronous or Trigger mode is selected by the OCx Trigger/Sync Select (OCTRIG) bit (OCxCON2<7>) and the Synchronization/Trigger source can be selected by the SYNCSEL<4:0> bits (OCxCON2<4:0>), as indicated in Section 2.0 “Output Compare Registers”.

3.1 Single Compare Match Mode

When control bits, OCM<2:0> (OCxCON1<2:0>) = 0b001, 0b010 or 0b011, the selected Output Compare channel is configured as:

- If the OCM<2:0> bits (OCxCON1<2:0>) = 0b001, the OCx pin is initially set low and a subsequent compare event with OCxR sets the pin high
- If the OCM<2:0> bits (OCxCON1<2:0>) = 0b010, the OCx pin is initially set high and a subsequent compare event with OCxR sets the pin low
- If the OCM<2:0> bits (OCxCON1<2:0>) = 0b011, the OCx pin is initially set low and a subsequent compare event with OCxR toggles the pin

In Single Compare mode, the OCxR register is used to generate the compare events. This register is loaded with a value and is compared with the Output Compare x Timer register. The interrupt is set on each compare event if there is a level change on the OCx pin.

3.1.1 SINGLE COMPARE MODE OUTPUT DRiven HIGH

To configure the OC module for the Single Compare mode output driven high, set control bits, OCM<2:0> (OCxCON1<2:0>) = 0b001. Once the Compare mode is enabled, the Output Compare x pin, OCx, will be initially driven low and remains low until a match between the timer and the OCxR/S registers occurs.

Note: When the SYNCSEL<4:0> bits (OCxCON2<4:0>) = 0b00000, they put the timer in a Free-Running mode with no synchronization.

When the SYNCSEL<4:0> bits (OCxCON2<4:0>) = 0b11111, they make the timer reset when it reaches the value of OCxRS, making the OCx module use its own Sync signal.

The OCx module sends out a Synchronization/Trigger signal when its timer matches the OCxRS register.

For more information on Synchronous/Trigger mode, refer to Section 3.3.7 “Synchronous Operation”.

3.2 Dual Compare Match Mode

When control bits, OCM<2:0> (OCxCON1<2:0>) = 0b100, 0b101 or 0b110, the selected Output Compare channel is configured as:

- If the OCM<2:0> bits (OCxCON1<2:0>) = 0b100, the OCx pin is driven low at the start of the Compare Window and is driven high at the end of the Compare Window
- If the OCM<2:0> bits (OCxCON1<2:0>) = 0b101, the OCx pin is driven high at the start of the Compare Window and is driven low at the end of the Compare Window
- If the OCM<2:0> bits (OCxCON1<2:0>) = 0b110, the OCx pin toggles state at the end of the Compare Window

In Dual Compare mode, the OCxR/S registers drive the compare events. This register is loaded with a value and is compared with the Output Compare x Timer register. The interrupt is set on each compare event if there is a level change on the OCx pin.

3.2.1 DUAL COMPARE MODE OUTPUT DRiven HIGH

To configure the OC module for the Dual Compare mode output driven high, set control bits, OCM<2:0> (OCxCON1<2:0>) = 0b100. Once the Compare mode is enabled, the Output Compare x pin, OCx, will be driven low at the start of the Compare Window and then driven high at the end of the Compare Window.

3.3 Simple Pulse-Width Modulation Mode

When control bits, OCM<2:0> (OCxCON1<2:0>) = 0b111, the selected Output Compare channel is configured as:

- If the Edges are aligned, the OCx pin is driven high when the timer matches the OCxR/S register and driven low when it resets
- If the Edges are center aligned, the OCx pin is driven high when the timer reaches the value of the OCxRS register and driven low when it resets

In Simple Pulse-Width Modulation mode, the OCxR/S registers generate the compare events. This register is loaded with a value and is compared with the Output Compare x Timer register. The interrupt is set on each compare event if there is a level change on the OCx pin.

3.3.1 SIMPLE PWM MODE OUTPUT DRiven HIGH

To configure the OC module for the Simple PWM mode output driven high, set control bits, OCM<2:0> (OCxCON1<2:0>) = 0b111. Once the Compare mode is enabled, the Output Compare x pin, OCx, will be driven high when the timer matches the OCxR/S register and then driven low when it resets.

3.4 Cascade Mode

When control bits, OCM<2:0> (OCxCON1<2:0>) = 0b111, the selected Output Compare channel is configured as:

- The OCx pin is driven high when the timer matches the OCxR/S register
- The OCx pin is driven low when the timer resets

In Cascade mode, the OCxR/S registers generate the compare events. This register is loaded with a value and is compared with the Output Compare x Timer register. The interrupt is set on each compare event if there is a level change on the OCx pin.

3.4.1 CASCADE MODE OUTPUT DRiven HIGH

To configure the OC module for the Cascade mode output driven high, set control bits, OCM<2:0> (OCxCON1<2:0>) = 0b111. Once the Compare mode is enabled, the Output Compare x pin, OCx, will be driven high when the timer matches the OCxR/S register and then driven low when it resets.
Figure 3-1 shows the following key timing events:

- The OCx pin is driven high, one instruction clock after a compare match between the timer and the OCxR register. The OCx pin remains high until a mode is changed or the module is turned off.
- The timer counts up until it rolls over, or until a Synchronization event occurs, and then resets to 0x0000 on the next instruction clock.
- The respective Output Compare x Channel Interrupt Flag, OCxIF, is asserted to two instruction clocks after the OCx pin is driven high.

Figure 3-1: Single Compare Mode – Sets OCx High on Compare Match Event

3.1.2 SINGLE COMPARE MODE OUTPUT DRIVEN LOW

To configure the OC module for the Single Compare mode output driven low, set control bits, OCM<2:0> (OCxCON1<2:0>) = 0b010. Once the Compare mode is enabled, the output pin, OCx, will be initially driven high and remains high until a match occurs between the timer and the OCxR/S registers. Figure 3-2 shows the following key timing events:

- The OCx pin is driven low, one instruction clock after a compare match event occurs between the timer and the OCxR register. The OCx pin remains low until a mode is changed or the module is turned off.
- The timer counts up until it rolls over, or until a Synchronization event occurs, and then resets to 0x0000 on the next instruction clock.
- The respective Output Compare x Channel Interrupt Flag, OCxIF, is asserted to two instruction clocks after the OCx pin is driven low.

Figure 3-2: Single Compare Mode – Forces OCx Low on Compare Match Event

Note: One instruction clock period comprises two Fosc cycles.
3.1.3 SINGLE COMPARE MODE TOGGLE OUTPUT

To configure the OC module for the Single Compare mode toggle output, set control bits, OCM<2:0> (OCxCON1<2:0>) = 0b011. Once the Single Compare mode is enabled, the output pin, OCx, toggles on every match event between the timer and the OCxR/S registers. Example 3-1 shows the sample code for the Single Compare mode toggle output.

Figure 3-3 shows the following key timing events.

- The OCx pin is toggled, one instruction clock after a compare match occurs between the Timer and the OCxR register. The OCx pin remains at this new state until the next toggle event, or until a mode is changed or the module is turned off.
- The timer counts up until it rolls over, or until a Synchronization occurs, and then resets to 0x0000 on the next instruction clock.
- The respective Output Compare x Channel Interrupt Flag, OCxF, is asserted to two instruction clocks after the OCx pin is toggled.
- The internal OCx pin output logic is set to a logic '0' on a device Reset. However, the operational OCx pin state in the Toggle mode can be set by the user software.

Example 3-1: Single Compare Mode Toggle Output

```
OC1CON1 = 0;  /* It is a good practice to clear off the control bits initially */
OC1CON2 = 0;
OC1CON1bits.OCTSEL = 0x07;  /* This selects the peripheral clock as the clock input to the OC module */
OC1R = 1000;   /* This is just a typical number, user must calculate based on the waveform requirements and the system clock */
OC1CON1bits.OCM = 3;  /* This selects and starts the toggle mode */
```
3.1.4 SPECIAL CASES OF SINGLE COMPARE MODE

Table 3-1 lists the special cases of Single Compare mode.

<table>
<thead>
<tr>
<th>Special Conditions</th>
<th>Operation</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>When the OCxR register is greater than the timer period as determined by the Sync source</td>
<td>No compare event occurs and the compare output remains at the initial condition.</td>
<td>No change in output level</td>
</tr>
<tr>
<td>When the OCxR register is equal to the timer period as determined by the Sync source</td>
<td>The compare output functions normally. Combining this with the Toggle mode can be used to generate a fixed frequency square wave, as illustrated in Figure 3-4.</td>
<td>Output level transition</td>
</tr>
<tr>
<td>When the module is enabled into Single Compare mode (OCxR = 0x0000) and the timer is held in Reset, the Sync source is active</td>
<td>The compare output remains in the initial condition.</td>
<td>No change in output level</td>
</tr>
<tr>
<td>If, after a compare event, the OCxR register is cleared and the Sync source becomes active</td>
<td>Output remains in the new state.</td>
<td>No further change in output level</td>
</tr>
</tbody>
</table>

Figure 3-4: Single Compare Mode – Toggle Output on Compare Match Event (OCxTMR = OCxR)

Note: One instruction clock period comprises two Fosc cycles.
3.2 Dual Compare Match Mode

When control bits, OCM<2:0> (OCxCON1<2:0>) = 0b100 or 0b101, the selected Output Compare channel is configured for one of the two following Dual Compare Match modes:

- Dual Compare Single Output Pulse mode
- Dual Compare Continuous Output Pulse mode

In the Dual Compare Match mode, the module uses both the OCxR and OCxRS registers for the compare match events. The OCxR register is compared with the incrementing timer count, OCxTMR and the rising (leading) edge of the pulse is generated at the OCx pin on a compare match event. The OCxRS register is then compared to the same incrementing timer count, OCxTMR, and the falling (trailing) edge of the pulse is generated at the OCx pin on a compare match event.

3.2.1 DUAL COMPARE SINGLE OUTPUT PULSE MODE

When control bits, OCM<2:0> (OCxCON1<2:0>) = 0b100, the selected Output Compare channel is configured so that the OCx pin is initialized low and a single output pulse is generated (see Figure 3-5 and Figure 3-6). Once the Dual Compare Single Output Pulse mode is enabled, the OCx pin will be driven low. The OCx pin will be driven high after a first timer compare match with the Output Compare x register, OCxR.

When the incrementing timer count matches the Output Compare x Secondary register, OCxRS, the second and trailing edge (high-to-low) of the pulse is driven onto the OCx pin. At this second compare, the OCxIF interrupt flag bit gets set. Example 3-2 shows the code for the Dual Compare Single Output Pulse mode. Table 3-2 shows the calculated examples for the Dual Compare Single Output Pulse mode. Equation 3-1 shows the formula for the calculations for the Dual Compare Single Output Pulse mode.

Note: If another write on the mode bits with the same value occurs on the same control bits, a new single output pulse sequence is generated, even if there is no change after the falling edge of the pulse.

The value of the OCxRS register must be greater than the OCxR register by a minimum of 2.
3.2.1.1 To Set Up Single Output Pulse Generation

To configure the module for the generation of a single output pulse, perform the following steps:

1. Determine the instruction cycle time, T_CY.
2. Calculate the desired pulse-width value based upon T_CY.
3. Calculate the time to start the pulse from the timer start value of 0x0000.
4. Write pulse-width start and stop times into the OCxR and OCxRS registers.
5. Select the SYNCSEL<4:0> bits (OCxCON2<4:0>) so that the synchronization is active after the timer is equal to, or greater than, the value in OCxRS.
6. Set the OCM<2:0> bits (OCxCON1<2:0>) = 0b100; the pulse will be generated.
7. Issue another write to set the OCM<2:0> bits (OCxCON1<2:0>) = 0b100 to initiate another single pulse with the same parameters.
8. Disable the OCx by writing the OCM<2:0> bits (OCxCON1<2:0>) = 0b000 to change the parameters and then enable the OCx by writing the OCM<2:0> bits (OCxCON1<2:0>) = 0b100 to initiate another single pulse with different parameters.
Output Compare with Dedicated Timer

Table 3-2 provides examples of single output pulse-width calculations.
Table 3-3 provides examples of the Dual Compare Match mode generating a single output pulse.

Table 3-2: Dual Compare Mode – Single Output Pulse-Width Calculation Examples

<table>
<thead>
<tr>
<th>Instruction Cycle Time (TCY)[1]</th>
<th>Desired On Time</th>
<th>Start Pulse Time from Timer = 0x0000</th>
<th>End Pulse Time (OCxRS) Register</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Time</td>
<td>Hex Value</td>
<td>Time</td>
</tr>
<tr>
<td>16.6 ns</td>
<td>1 μs</td>
<td>0x003C</td>
<td>10 μs</td>
</tr>
<tr>
<td>30 ns</td>
<td>1 μs</td>
<td>0x0021</td>
<td>10 μs</td>
</tr>
<tr>
<td>30 ns</td>
<td>2 μs</td>
<td>0x0042</td>
<td>10 μs</td>
</tr>
<tr>
<td>50 ns</td>
<td>3 μs</td>
<td>0x003C</td>
<td>10 μs</td>
</tr>
<tr>
<td>62.5 ns</td>
<td>3 μs</td>
<td>0x0030</td>
<td>10 μs</td>
</tr>
<tr>
<td>100 ns</td>
<td>5 μs</td>
<td>0x0032</td>
<td>50 μs</td>
</tr>
<tr>
<td>300 ns</td>
<td>10 μs</td>
<td>0x0021</td>
<td>100 μs</td>
</tr>
<tr>
<td>500 ns</td>
<td>20 μs</td>
<td>0x0028</td>
<td>500 μs</td>
</tr>
<tr>
<td>500 ns</td>
<td>30 μs</td>
<td>0x003C</td>
<td>2 ms</td>
</tr>
</tbody>
</table>

Note 1: Verify the specific device data sheet for the minimum TCY of operation.

Equation 3-1: Dual Compare Mode – Single Output Pulse Width

Value = Desired Time/Instruction Cycle Time (TCY)

Example 3-2: Dual Compare Mode – Single Output Pulse Width

OC1CON1 = 0; /* It is a good practice to clear off the control bits initially */
OC1CON2 = 0;
OC1CON1bits.OCTSEL = 0x07; /* This selects the peripheral clock as the clock input to the OC module */
OC1R = 1000; /* This is just a typical number, user must calculate based on the waveform requirements and the system clock */
OC1RS = 2000;
OC1CON1bits.OCM = 4; /* This selects and starts the Single Output Pulse mode */
<table>
<thead>
<tr>
<th>Special Conditions</th>
<th>Operation</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Synchronization occurs when the timer value is equal to OCxRS</td>
<td>Timer resets to zero in the next cycle, but the pulse is unaffected.</td>
<td>Pulse</td>
</tr>
<tr>
<td>Synchronization occurs before the timer value reaches OCxR</td>
<td>Timer resets to zero before any output transition.</td>
<td>Remains low</td>
</tr>
<tr>
<td>Synchronization occurs before the timer value reaches OCxRS, but after it reaches OCxR</td>
<td>Only a single transition (low-to-high) is generated (see Figure 3-7).</td>
<td>Low/High</td>
</tr>
<tr>
<td>OCxR = OCxRS = 0x0000 and Sync occurs</td>
<td>The output is initialized low and does not change. No interrupt is generated.</td>
<td>Remains low</td>
</tr>
<tr>
<td>OCxRS &lt; OCxR</td>
<td>The timer counts up to the first Output Compare x register (TMRx = OCxR) and the first rising edge is generated. The timer then continues to count and eventually resets when the synchronization occurs or rolls over. The timer then restarts from 0x0000 and counts up to the Output Compare x Secondary register (TMRx = OCxRS) and the second falling edge of the signal is generated. The falling edge of the output pulse generates an interrupt condition.</td>
<td>Pulse</td>
</tr>
<tr>
<td>OCxR = OCxRS</td>
<td>The timer counts up to the first Output Compare x register (Timer = OCxR) and the first rising edge is generated. The timer continues to count and eventually resets when the synchronization occurs or a rollover from 0xFFFF occurs. The timer then restarts from 0x0000 and counts up to the Output Compare x Secondary register (TMRx = OCxRS), and the second falling edge of the signal is generated. The falling edge of the output pulse generates an interrupt condition.</td>
<td>Pulse</td>
</tr>
<tr>
<td>OCxR = 0x0000 and OCxRS &gt; OCxR</td>
<td>The first cycle of the timer counts until the synchronization occurs or rolls over and the Output Compare x pin remains low. After the Timer register resets to zero, the Output Compare x pin goes high. In the next timer match with the register, OCxRS, the Output Compare x pin goes low and remains. The falling edge of the output pulse generates an interrupt condition (see Figure 3-8).</td>
<td>Pulse except for the first cycle</td>
</tr>
</tbody>
</table>
Output Compare with Dedicated Timer

Figure 3-7: Dual Compare Mode – Single Output Pulse (Sync Before Timer Reaches OCxRS)

Figure 3-8: Dual Compare Mode – Single Output Pulse (OCxR = 0x0000, OCxRS > OCxR)
3.2.2 DUAL COMPARE CONTINUOUS OUTPUT PULSE MODE

When the OCx Mode Select bits, OCM<2:0> (OCxCON1<2:0>) = 0b101, the selected Output Compare channel is configured so that the OCx pin is initialized low and continuous output pulses are generated. Figure 3-9 shows the Dual Compare Continuous Output Pulse mode. Once the Dual Compare Continuous Output Pulse mode is enabled, the pin state will be driven low. The OCx pin will be driven high after a first timer compare match with the Output Compare x register, OCxR.

When the incrementing timer count matches the Output Compare Secondary register, OCxRS, the second and trailing edges (high-to-low) of the pulse are driven onto the OCx pin. At this second compare, the OCxIF interrupt flag bit is set. Example 3-3 shows the sample code for the Dual Compare Continuous Output Pulse mode generation. Table 3-4 shows the calculated examples for the Dual Compare Continuous Output Pulse mode.

Note: Unlike the Dual Compare Single Output Pulse mode, the output pulses continue indefinitely until the mode is terminated by the user firmware or by a Reset. The falling edge of each output pulse sets the interrupt flag.

One way of generating a pulse with 50% duty cycle is by setting OCxR = OCxRS and self-synchronizing.

Figure 3-9: Dual Compare Mode – Continuous Output Pulses
3.2.3 SETUP FOR CONTINUOUS OUTPUT PULSE GENERATION

To configure the OCx module for the generation of a continuous stream of output pulses, perform the following steps:

1. Determine the instruction cycle time, TCY.
2. Calculate the timer to start the pulse width from the timer start value of 0x0000.
3. Calculate the timer to stop the pulse width from the timer start value of 0x0000.
4. Write the pulse-width start and stop values into the OCxR and OCxRS registers, respectively. The Sync signal should occur when OCxRS = timer or after.
5. Set the OCM<2:0> bits (OCxCON1<2:0>) = 0b101; the timer must be enabled.

Example 3-3: Continuous Output Pulse Generation

Table 3-4: Dual Compare Mode – Continuous Output Pulse-Width Calculation Examples

<table>
<thead>
<tr>
<th>Instruction Cycle Time (TCY)</th>
<th>Desired Pulse Width</th>
<th>Start Pulse Time from Timer = 0x0000</th>
<th>End Pulse Time from Timer = 0x0000</th>
<th>Pulse Period</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Time</td>
<td>Value</td>
<td>Time</td>
<td>Value</td>
</tr>
<tr>
<td></td>
<td>(OCxR)</td>
<td>Time</td>
<td>(OCxR)</td>
<td>Value</td>
</tr>
<tr>
<td>30 ns</td>
<td>1 µs</td>
<td>0x0021</td>
<td>10 µs</td>
<td>0x014D</td>
</tr>
<tr>
<td>30 ns</td>
<td>2 µs</td>
<td>0x0042</td>
<td>10 µs</td>
<td>0x014D</td>
</tr>
<tr>
<td>50 ns</td>
<td>3 µs</td>
<td>0x003C</td>
<td>10 µs</td>
<td>0x00C8</td>
</tr>
<tr>
<td>62.5 ns</td>
<td>3 µs</td>
<td>0x0030</td>
<td>10 µs</td>
<td>0x00A0</td>
</tr>
<tr>
<td>100 ns</td>
<td>5 µs</td>
<td>0x0032</td>
<td>10 µs</td>
<td>0x0064</td>
</tr>
<tr>
<td>300 ns</td>
<td>10 µs</td>
<td>0x0021</td>
<td>100 µs</td>
<td>0x014D</td>
</tr>
<tr>
<td>500 ns</td>
<td>20 µs</td>
<td>0x0028</td>
<td>500 µs</td>
<td>0x03E8</td>
</tr>
<tr>
<td>500 ns</td>
<td>30 µs</td>
<td>0x003C</td>
<td>2 ms</td>
<td>0x0FA0</td>
</tr>
</tbody>
</table>

Note 1: Verify the device data sheet for the minimum TCY of operation.

Equation 3-2:

Value = Desired Time/Instruction Cycle Time (TCY)

Note: The timer module with the same clock as OCx is used as the Sync source in Table 3-4.
Table 3-5: Special Cases for Dual Compare Match Mode Generating Continuous Output Pulse

<table>
<thead>
<tr>
<th>Special Condition</th>
<th>Operation</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Synchronization occurs when the timer value is equal to OCxRS</td>
<td>Timer resets to zero in the next cycle, but the pulse is unaffected. (see Figure 3-10).</td>
<td>Pulses</td>
</tr>
<tr>
<td>Synchronization occurs before the timer value reaches OCxR</td>
<td>Timer resets to zero before any output transition.</td>
<td>Remains low</td>
</tr>
<tr>
<td>Synchronization occurs before the timer value reaches OCxRS, but after it reaches OCxR</td>
<td>Only a single transition, low-to-high, is generated. (see Figure 3-11).</td>
<td>Low/High</td>
</tr>
<tr>
<td>OCxR = OCxRS = 0x0000 and synchronization occurs</td>
<td>The output is initialized low and does not change. No interrupt is generated.</td>
<td>Remains low</td>
</tr>
<tr>
<td>OCxRS &lt; OCxR</td>
<td>The timer counts up to the first compare (TMRx = OCxR) and the first rising edge is generated. The timer then continues to count and eventually resets when synchronization occurs or rolls over. The timer then restarts from 0x0000 and counts up to the second compare (TMRx = OCxRS) and the second falling edge of the signal is generated. The falling edge of the output pulse generates an interrupt condition. The sequence repeats until the module is disabled.</td>
<td>Pulses</td>
</tr>
<tr>
<td>OCxR = OCxRS</td>
<td>The timer counts up to the first compare (TMRx = OCxR) and the first rising edge is generated. The timer continues to count and eventually resets when synchronization occurs or a rollover from FFFFh occurs. The timer then restarts from 0x0000 and counts up to the second compare (TMRx = OCxRS), and the second falling edge of the signal is generated. The falling edge of the output pulse generates an interrupt condition. The sequence repeats until the module is disabled.</td>
<td>Pulses</td>
</tr>
<tr>
<td>OCxR = 0x0000 and OCxRS &gt; OCxR</td>
<td>The first cycle of the timer counts until synchronization occurs or rolls over; the Output Compare x pin remains low. After the Timer register resets to zero, the Output Compare x pin goes high. In the next timer match with the Output Compare x Secondary register, OCxRS, the Output Compare x pin goes low and remains low. The falling edge of the output pulse generates an interrupt condition. (see Figure 3-12). The sequence repeats until the module is disabled.</td>
<td>Pulses except for the first cycle</td>
</tr>
</tbody>
</table>
Output Compare with Dedicated Timer

Figure 3-10: Dual Compare Mode – Continuous Output Pulse (Sync Occurs When Timer = OCxRS)

![Diagram of I/O pair showing continuous output pulse with sync occurring when timer equals OCxRS. The diagram includes labels for OCxTMR, OCx Sync, OCxR, OCxRS, OCx Pin, OCxIF, and a note that one instruction clock period comprises two FOSC cycles.]

Note: One instruction clock period comprises two FOSC cycles.

Figure 3-11: Dual Compare Mode – Continuous Output Pulse (Sync Before Timer Reaches OCxRS)

![Diagram of I/O pair showing continuous output pulse with sync occurring before timer reaches OCxRS. The diagram includes labels for OCxTMR, OCx Sync, OCxR, OCxRS, OCx Pin, OCxIF, and a note that one instruction clock period comprises two FOSC cycles.]

Note: One instruction clock period comprises two FOSC cycles.
Figure 3-12: Dual Compare Mode – Continuous Output Pulse (OCxR = 0x0000 (SYNCSEL<4:0> = 0x1F))
Output Compare with Dedicated Timer

3.3 Pulse-Width Modulation Mode

When control bits, OCM<2:0> (OCxCON1<2:0>) = 0b110 or 0b111, the Pulse-Width Modulation (PWM) mode is selected. The registers, OCxR and OCxRS, are double-buffered in these modes, that is, the changes on these registers will be reflected only after a timer rollover from 0xFFFF or after a Sync event occurs. As a result, any changes in the OCxR and OCxRS registers during operation occurs only with the next pulse. Furthermore, in PWM mode, the Fault input is supported as described in the following sections.

3.3.1 EDGE-ALIGNED PWM MODE

When control bits, OCM<2:0> (OCxCON1<2:0>) = 0b110, the Edge-Aligned PWM mode of operation is selected. The OCxR register contains the current duty cycle and the SYNCSELx bits determine the period. The OCxRS register can be made to determine the period by setting the SYNCSEL<4:0> bits (OCxCON2<4:0>) = 0x1F.

**Note:** This is a migration issue for applications. In the OC module without dedicated timers (see “Output Compare” chapter in the specific device data sheet); the OCxRS register served as a double-buffer to OCxR. In this version, both the registers are double-buffered.

Figure 3-13 and Figure 3-14 show the PWM mode of operation.

Edge-Aligned PWM Mode Operation:
- When synchronization occurs, the following four events occur on the next increment cycle:
  - The timer is reset to zero and resumes counting
  - The OCx pin is set high (if OCxRS = 0b0000, the OCx pin may not be set)
  - The OCxR and OCxRS Buffered registers are updated from OCxR and OCxRS
  - The interrupt flag, OCxIF, is set
- When the timer and OCxR match, the pin may be set low. This match does not generate the interrupts.

**Figure 3-13: PWM Output Timing**

**Figure 3-14: PWM Output Timing**

*Note:* In this example, one instruction clock period comprises two Fosc cycles.
3.3.2 EDGE-ALIGNED PWM MODE INITIALIZATION

Once the PWM mode is enabled by setting the OCM<2:0> bits (OCxCON1<2:0>) = 0b110, the OCx pin would be driven low if OCxR = 0x0000. If OCxR is not equal to zero, then the OCx pin will be set high (see Figure 3-15 and Figure 3-16).

When OCxR is not equal to zero and the pin state is set to high, then the first match between the OCxR and the timer clears the OCx pin. The OCx pin will remain low until a valid compare between synchronization occurs or until a rollover occurs (see Figure 3-17).

Figure 3-15: Edge-Aligned PWM Mode with OCxR = 0 – At Module Initialization, OCxR = 0x0000, OCxRS = 0x5000

Figure 3-16: Edge-Aligned PWM Mode with OCR > 0 – At Module Initialization, OCxR = 0x1000, OCxRS = 0x5000
Output Compare with Dedicated Timer

3.3.3 USER SETUP FOR PWM OPERATION

To configure the OCx module for PWM operation, perform the following steps (Example 3-4 shows the example code for the PWM):

1. Determine the instruction cycle time, Tcy.
2. Calculate the desired pulse on time value based upon Tcy and write it into OCxR.
3. Calculate the period value based upon Tcy and write it into OCxRS.
4. Write 0xF0 to the SYNCSEL<4:0> bits (OCxCON2<4:0>) to select self-synchronization.
5. Set the required clock source.
6. Set the OCM<2:0> bits (OCxCON1<2:0>) = 0b110 to select and start Edge-Aligned PWM mode.

Example 3-4: PWM Mode

```
OC1CON1 = 0;  /* It is a good practice to clear off the control bits initially */
OC1CON2 = 0;
OC1CON1bits.OCTSEL = 0x07; /* This selects the peripheral clock as the clock input to the OC module */
OC1R = 1000;   /* This is just a typical number, user must calculate based on the waveform requirements and the system clock */
OC1RS = 2000;  /* Determines the Period */
OC1CON2bits.SYNCSEL = 0x1F; /* This selects the synchronization source as itself */
OC1CON1bits.OCM = 6; /* This selects and starts the Edge Aligned PWM mode */
```

3.3.4 PWM SPECIAL COMPARE MODE CONDITIONS

Table 3-6 lists the PWM Special Compare mode conditions.

<table>
<thead>
<tr>
<th>Special Condition</th>
<th>Operation</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>OCxR = 0</td>
<td>The OCx pin would be set low</td>
<td>Low</td>
</tr>
<tr>
<td>OCxR &gt; OCxRS</td>
<td>The OCx pin would be set high</td>
<td>High</td>
</tr>
<tr>
<td>OCxR = OCxTMR and synchronization occurs</td>
<td>The OCx pin would remain high</td>
<td>High</td>
</tr>
</tbody>
</table>

Figure 3-17: PWM Output Timing (0% Duty Cycle, OCxR = 0x0000)

Note: One instruction clock period comprises two Fosc cycles.
Figure 3-18: PWM Output Timing (100% Duty Cycle, OCxR > OCxRS (SYNCSEL<4:0> = 0x1F))

**Note:** One instruction clock period comprises two Fosc cycles.

---

Figure 3-19: PWM Output Timing (OCxR = OCxRS (SYNCSEL<4:0> = 0x1F))

**Note:** One instruction clock period comprises two Fosc cycles.
3.3.5 CENTER-ALIGNED PWM MODE

In this mode, the OCM<2:0> bits (OCxCON1<2:0>) = 0b111 functions are the same as Continuous Pulse mode, OCM<2:0> bits (OCxCON1<2:0>) = 0b101, and the only differences are:

- The OCxR and OCxRS registers are double-buffered, which means that the new register value would be effective only after a timer rollover or synchronization.
- Fault control and pins are used.

| Note: | Center-alignment does not mean the pulse is exactly aligned to the center of the pulse width. It indicates that the on time of the pulse can be positioned anywhere within the period. |

3.3.6 FAULT INPUT AND CONTROL

When operating in either a Center-Aligned PWM mode or in an Edge-Aligned PWM mode (OCM<2:0> bits (OCxCON1<2:0>) = 0b111 or 0b110), the Fault pin and its controls can be activated. The Fault pin, OCFA, is always available and controls all of the OCx modules. However, another Fault pin, OCFB, may also be available. The Fault pin is controlled by the register bits, ENFLTx (OCxCON1<9:7>). If these bits are set to zero, the corresponding Fault input pins (OCFA, OCFB, etc., refer to the specific device data sheet for Fault signal mappings) are ignored. The status of the Fault input can be observed in the corresponding OCFLTx (OCxCON1<6:4>) register bits.

| Note: | The Output Compare x Fault pins, OCFA and OCFB, are active-low signals. |

When a Fault occurs (OCFx = 0), the OCx pin output level is determined by the FLTOUT bit (OCxCON2<14>). The tri-stating of the OCx pin during a Fault condition is controlled by the FLTTRIEN bit (OCxCON2<13>).

| Note: | For more information on how Fault pins are assigned to the various OCx peripherals, refer to the specific device data sheet. |

The Fault control can operate in the following two modes based on the FLTMD bit (OCxCON2<15>):

- Inactive mode
- Cycle-by-Cycle mode

3.3.6.1 Inactive Mode

When the FLTMD bit (OCxCON2<15>) = 1, the Fault inputs operate in the Inactive mode (see Figure 3-20). If a Fault input goes into an active (‘0’) mode, the OCFLTx bits (OCxCON1<6:4>) are set, and the OCx module will remain in the Fault condition until:

- The Fault input goes into an inactive mode.
- The OCFLTx bits (OCxCON1<6:4>) are cleared in software.
- A new timer cycle is started (timer goes to 0000h).

3.3.6.2 Cycle-by-Cycle Mode

When the FLTMD bit (OCxCON2<15>) = 0, the Fault inputs operate in the Cycle-by-Cycle mode (see Figure 3-21). If a Fault input goes into an active (‘0’) mode, the OCFLTx bits (OCxCON1<6:4>) are set and the OCx module will remain in the Fault condition until:

- The Fault input goes into an inactive mode.
- A new timer cycle is started (timer goes to 0000h).
Figure 3-20: Fault Input Pin Timing, Inactive Mode

Figure 3-21: Fault Input Pin Timing, Cycle-by-Cycle Mode
3.3.7 SYNCHRONOUS OPERATION
The synchronous operation of the timer is enabled when the OCTRIG bit (OCxCON2<7>) = 0.
In synchronous operation, the TRIGSTAT bit (OCxCON2<6>) has no function. The timer can be
synchronized with the other modules using the synchronization/trigger inputs (see Register 2-2).
Whenever the selected module receives a synchronization signal, the timer will roll over to
0x0000 on the next positive edge of the selected clock.

3.3.8 USE OF THE MODULE TIMER IN A SYNCHRONIZED APPLICATION
Figure 3-22 shows the connections for synchronization and Figure 3-23 shows the timing for
multiple modules being synchronized. The OC2 module is being synchronized to the OC1
module. The synchronization signal from OC1 is selected for synchronization by both OC1 and
OC2 using the SYNCSEL<4:0> bits (OCxCON2<4:0>). The OC1RS register now becomes the
Period register for both OC1 and OC2.
When the OC1RS register matches the OC1 timer value, the OC1 module produces the synchro-
nization signal. This causes the timers in both OC1 and OC2 to go to zero on the next positive
clock edge.

Note: Synchronized modules should select the same clock source to ensure proper
operation.

Figure 3-22: Synchronous Operation Integration (TRIGEN = 0)
When initializing the synchronized modules, the module being used as the source of synchronization should be enabled last. As shown in Figure 3-23, OC2 should be initialized first and OC1 should be initialized last. This ensures that the timers of all synchronized modules are maintained in a Reset condition until the last module is initialized.

### 3.3.9 TRIGGER OPERATION

Trigger operation of the timer is enabled when the OCTRIG bit (OCxCON2<7>) = 1. When configured for trigger operation, the module timer is held in Reset until a trigger event occurs. After the trigger event occurs, the timer begins to count. The trigger source is selected by the SYNCSELx bits.

### 3.3.10 OCxCON2 TRIGGER FUNCTION

The TRIGSTAT bit (OCxCON2<6>) holds the timer in Reset or releases it to count. It controls the timer in the following manner:

- **TRIGSTAT = 0**
  - Timer is held in Reset

- **TRIGSTAT = 1**
  - Timer is released from Reset
  - Timer increments on every positive clock

There are two types of trigger conditions when operating in a Trigger mode:

- Hardware/software, TRIGSTAT bit is set
- Software only, TRIGSTAT bit is set

In both cases, the trigger is always cleared in software.
Output Compare with Dedicated Timer

3.3.10.1 Hardware/Software TRIGSTAT Set
The TRIGSTAT bit (OCxCON2<6>) can be set by hardware or software when:
• The SYNCSELx bits (OCxCON2<4:0>) are not equal to '0b00000' (see Section 3.3.12 “Illegal Settings”)
When the module is enabled for a triggered response, the timer would be held in a cleared state. It remains in this cleared state until a trigger event occurs, which sets the TRIGSTAT bit. Additionally, the timer can be released from Reset by writing to the TRIGSTAT bit and setting it.

3.3.10.2 Software Only TRIGSTAT Set
The TRIGSTAT bit can be set only by software when the SYNCSEL<4:0> bits = 0b00000.

3.3.11 CLEARING TRIGSTAT BIT
The TRIGSTAT bit can only be cleared in software by writing a '0' to it. When the TRIGSTAT bit is cleared in software, the timer is reset to 0x0000 on the next timer clock’s rising edge and is ready for another trigger.

3.3.12 ILLEGAL SETTINGS
It is illegal for the module to select itself as a trigger source. Therefore, two possible values of the SYNCSEL<4:0> bits in Trigger mode are not allowed:
• SYNCSEL<4:0> = 0xF
• SYNCSEL<4:0> = N, where N is the second setting that selects the same module (see Register 2-2)

A Sync/Trigger with Timer module occurs when the corresponding TMRx register value matches with the PRx register. In Trigger mode, the OCxTMR register is held in Reset and starts counting after a match between the TMRx and PRx register occurs. In Synchronization mode, the OCxTMR and TMRx registers will count together after a match between the TMRx and PRx registers occurs.

A Sync/Trigger with an Input Capture (IC) module occurs when a capture event occurs and an IC interrupt is generated. In Trigger mode, the OCxTMR is held in Reset and starts counting after the IC interrupt is generated. In Synchronization mode, the OCxTMR and ICxTMR registers will count together after the interrupt occurs.

A Sync/Trigger between the two OC modules occurs when the OCxTMR value of the triggering OC module matches with its period value. In Trigger mode, the OCxTMR register is held in Reset and starts counting after an OC interrupt is generated for the triggering OC module. In Synchronization mode, both OCxTMR registers will count together after the interrupt occurs.

A Sync/Trigger with a comparator module occurs when the comparator module is enabled and the compare event occurs. In Trigger mode, the OCxTMR register will be held in Reset until the comparator compare event occurs and starts counting after the compare event. In Synchronization mode, the OCxTMR register will be reset and does not count as there is no comparator timer to synchronize.

A Sync/Trigger with an Analog-to-Digital Controller (ADC) module occurs when the ADC generates an interrupt after a successful conversion. In Trigger mode, the OCxTMR register will be held in Reset until the ADC interrupt occurs and starts counting after the interrupt event. In Synchronization mode, the OCxTMR register will be reset and does not count as there is no ADC timer to synchronize.

Note: The TRIGSTAT bit cannot be changed in software when operating in One-Shot mode (see Section 3.3.13.2 “One-Shot Functionality”).

The trigger source will be synchronized to the OCx clock. There should be measures to prevent these illegal conditions in the user software.
3.3.13 USE OF THE OCx MODULE IN A TRIGGERED APPLICATION

Figure 3-24 shows a typical application of the module timer in a triggered application. In this application, a trigger event can be generated by another OC module, timer module, IC module, analog comparator or other peripheral function. Refer to the specific device data sheet for a list of trigger sources.

**Note:** When OCx is switched off, it sends a trigger out signal. If any other module is using OCx as a trigger source, it must disable the Trigger mode before switching off the OCx module.

3.3.13.1 Initialization of the OCx Module in a Triggered Application

The user misses any trigger event that occurs before the OCx module is initialized. Therefore, to avoid missing a trigger, the OCx module is enabled before the trigger source.

![Figure 3-24: Trigger Operation Integration (TRIGEN = 1)](image)

3.3.13.2 One-Shot Functionality

While operating as a trigger, the timer can operate in One-Shot mode. This produces one pulse for every trigger. The One-Shot mode is enabled by setting the TRIGMODE bit (OCxCON1<3>). In One-Shot mode, the timer remains in Reset until a trigger event occurs. This event sets the TRIGSTAT bit and the timer begins to count. When the timer rolls over to 0000h, the TRIGSTAT bit will be cleared by hardware if the TRIGMODE bit = 1. This holds the timer in Reset until the next trigger event, creating a one-shot timer.
3.4 Cascade Mode

When 16-bit timers are not enough, the OCx modules can be grouped in pairs to cascade them into 32-bit timers (see Figure 3-25). They are grouped as odd and even pairs (1-2, 3-4, 5-6, etc.). When cascading, the odd OCx module forms the Least Significant 16 bits of the timer/compare and the even module forms the Most Significant 16 bits. The OCx pin of the even module is the output of the cascaded timers.

**Note:** When OCx is configured for PWM Cascade mode, the Even Duty Cycle register should contain a non-zero value.

![Cascade Operation Diagram](image)

3.5 Setting Up Modules for Cascade

In this section, read OC1 as the odd OC module and OC2 as the even OC module.

The odd OC module is set up as follows:
- OC32 bit (OC1CON2<8>) = 1
- OCTRIG bit (OC1CON2<7>) can either be ‘1’ or ‘0’ as the timer can either be synchronized or triggered
- OCTRIS bit (OC1CON2<5>) = 1 (since the OC1 pin is not used, the output should be tri-stated)

The even OC module is set up as follows:
- OC32 bit (OC2CON1<8>) = 1
- OCTRIG bit (OC2CON2<7>) = 0 (even the timer must be operated in Synchronized mode when cascaded)
- OCTRIS bit (OC2CON2<5>) = 0 (since OC2 will be used, the output should be enabled)

3.5.1 Initialization of the Modules in a Cascade Application

When initializing cascaded modules, the even module should be initialized first and the odd module should be initialized last. Example 3-5 shows the example code for the Output Compare module in Cascade mode.
3.5.2 TIMER CLOCK SELECTION
The timer clock should be selected before the module is enabled and should not be changed during the operation. The waveform for the cascade operation is shown in Figure 3-26.

**Note:** The even and odd OC modules must have the same clock.

**Figure 3-26: Cascade Operation in Dual Compare Mode**

3.5.3 CASCADE OPERATION WITH THE ODD OC MODULE TRIGGERED
When the two modules are cascaded to form a 32-bit timer, the timer can be triggered by setting the odd module, OCTRIG bit (OCxCON2 <7>) = 1. The odd module remains in Reset until a trigger event occurs. Once a trigger event occurs, the odd and even modules count as usual.

3.5.4 SYNCHRONIZING MULTIPLE CASCADED MODULE PAIRS
The following examples show that multiple 32-bit pairs can be synchronized:

- To synchronize the OC3 + OC4 pair with the OC1 + OC2 pair:
  - OC1 and OC2 are set up as defined in Section 3.5 “Setting Up Modules for Cascade”.
  - OC3 and OC4 are set up in the same way, but the SYNCSEL bit = 1 (synchronization out from OC1); this allows the sync out from OC1 to hold OC3 in sync.
Output Compare with Dedicated Timer

3.5.5 PWM IN CASCA DING MODE

A PWM pulse can be generated by cascading two OC modules. Cascading can be used if the compare values are more than 16 bits. A proper PWM pulse is not generated when the compare values are 16-bit or less and the Cascade mode is selected. If the compare values are 16-bit or less, the OC module has to configure for 16-bit operation.

Example 3-6: PWM In Cascade Mode

In Example 3-6, the compare value for the OC cascaded timer is 21000 (OC2R-OC1R cascaded) and the period is 32000 (OC2RS-OC1RS cascaded). For a match between OC2TMR-OC1TMR (cascaded) and OC2R-OC1R (cascaded), the OCx pin goes high. The pin is set low for a match of OC2RS-OC1RS (cascaded). The OC2TMR always resets for a match of OC2RS and the OC1TMR counts up to 0xFFFF.
The following are the cases when the compare value is 16 bits or less:

- If OC2RS = 0, then in the first match between OC2TMR-OC1TMR (cascaded) and OC2R-OC1R (cascaded), the OCx pin goes high. Since OC2RS, which determines the period is '0', both the OC2TMR and OC1TMR keep incrementing and reset on overflow. Therefore, the output remains high and no PWM pulse is obtained.

- If OC2R = 0, then in the first match between OC2TMR-OC1TMR (cascaded) and OC2R-OC1R (cascaded) – or OC1R, since OC2R = 0, the OCx pin goes high. The OC2TMR increments by 1 when the OC1TMR reaches 0xFFFF. The OC2TMR clears for a match with OC2RS. The pin remains high and no PWM pulse is obtained.

- If OC2R = 0 and OC2RS = 0, then in the first match between OC2TMR-OC1TMR (cascaded) and OC2R-OC1R (cascaded) – or OC1R and OC1TMR, since OC2R, OC2TMR = 0, the OCx pin goes high. The OC1TMR clears for the first match to OC1RS and OC2TMR increments by 1 for this match, and both the timers keep incrementing until they reach 0xFFFF. The pin remains high and no PWM pulse is obtained.

3.5.6 EFFECTS OF DCB<1:0> SETTINGS

The DCB<1:0> bits (OCxCON2 <10:9>) setting can be used to achieve a finer resolution of a duty cycle. If the DCBx bits = 0b00, then there would be no effect of the DCBx bits but for the settings, 0b01, 0b10 and 0b11, and the second edge of the pulse is delayed until the rising edge of the P2, P3 and P4 clocks, respectively. Since these quadrature clocks may not have a 50% duty cycle, selecting between adjacent quadrature clocks may not yield a 25% difference, but selecting between the P1 and P3 clocks will yield a 50% difference (see Figure 3-27).

Note: A duty cycle setting of 0 (without DCBx bits) will not have any effect because of the DCBx bits. The DCBx bits will have their effect only if there is a pulse.

A prescaler will not scale the effects of the DCB<1:0> bits; that is, even if the prescaler is used to divide the clock fed to the OCx module, the effect of the DCB<1:0> bits will be as if no prescaler is used.
4.0 OUTPUT COMPARE OPERATION WITH DMA

Some Output Compare with Dedicated Timer family devices include a Direct Memory Access (DMA) module, which allows data transfer from data memory to the OCx module without CPU intervention.

The DMA channel must be initialized with the following:

- Initialize the DMAx Channel Peripheral Address (DMAxPAD) register to the address of the Output Compare x (OCxR) register or the Output Compare x Secondary (OCxRS) register.
- Set the Transfer Direction (DIR) bit in the DMAx Control (DMAxCON<13>) register. In this condition, data is read from the dual port DMA memory and written to the peripheral's Special Function Register (SFR).
- The DMA Request Source Selection (IRQSEL<7:0>) bits in the DMAx Request (DMAxREQ<7:0>) register must select the DMA transfer request source.

Example 4-1 provides sample code that modulates the PWM duty cycle without CPU intervention. The duty cycle values stored in an array are transferred to the OCxRS register on every timer interrupt.

**Example 4-1: Code to Modulate the PWM Duty Cycle Without CPU Intervention**

```c
// Define Buffer in RAM as global variable:
unsigned int BufferA[256] __attribute__((space(xmemory)));

// Initialize buffer with duty cycle values
int i;
for(i=0;i<256;i++)
    BufferA[i] = i;

// Initialize Output Compare Module in PWM mode
OC1CON1bits.OCM = 0b000; // Disable Output Compare Module
OC1R = 100; // Write the duty cycle for the PWM pulse
OC1RS = 255; // Write the PWM frequency
OC1CON1bits.OCTSEL = 0; // Select Timer2 as output compare time base
OC1CON1bits.OCM = 0b110; // Select the Output Compare mode
OC1CON2bits.SYNCSEL = 31; // OC2RS compare event is used for synchronization

// Initialize Timer2
T2CONbits.TON = 0; // Disable Timer
T2CONbits.TCS = 0; // Select internal instruction cycle clock
T2CONbits.TGATE = 0; // Disable Gated Timer mode
T2CONbits.TCKPS = 0b00; // Select 1:1 Prescaler
TMR2 = 0x00; // Clear timer register
PR2 = 500; // Load the period value

// Set up and Enable DMA Channel
DMA0CONbits.AMODE = 0b00; // Register indirect with post increment
DMA0CONbits.NODE = 0b00; // Continuous, Ping-Pong mode Disabled
DMA0CONbits.DIR = 1; // Peripheral to RAM
DMA0PAD = (int)&OC1R; // Address of the output compare register
DMA0REQ = 7; // Select Timer2 interrupt as DMA request source
DMA0CNT = 255; // Number of words to buffer.
DMA0STA = (unsigned int)&BufferA;
DMA0STAL = (unsigned int)&BufferA;

IF0bits.DMA0IF = 0; // Clear the DMA interrupt flag
IE0bits.DMA0IE = 1; // Enable DMA interrupt
DMA0CONbits.CHEN = 1; // Enable DMA channel

// Enable Timer
T2CONbits.TON = 1; // Start Timer

// Set up DMA Interrupt Handler:
void __attribute__((__interrupt__,no_auto_psv)) _DMA0Interrupt(void)
{
    // Process the captured values
    IFS0bits.DMA0IF = 0; // Clear the DMA0 Interrupt Flag
}
```

**Note:** The DMA module is not available on all devices. For more information, refer to the specific device data sheet.
5.0 OUTPUT COMPARE OPERATION IN POWER-SAVING STATES

5.1 Output Compare Operation in Sleep Mode

When the device enters Sleep mode, the system clock is disabled. During Sleep, the Output Compare x channel drives the pin to the same active state as it was driven prior to entering the Sleep state. The OCx module then halts at this state.

For example, if the pin was high and the CPU enters the Sleep state, the pin stays high. Likewise, if the pin was low and the CPU enters the Sleep state, the pin stays low. In both cases, when the device awakes, the OCx module resumes operation.

5.2 Sleep with PWM Fault Mode

When the OCx module is in PWM Fault mode, the asynchronous portions of the Fault circuit remain active. If a Fault is detected, the output of OCx is determined by the FLTOUT and OCTRIS bits setting in the OCxCON2 register. The FLTOUT bit will be set. An interrupt will not be generated at a Fault occurrence. However, the interrupt will be queued and will occur at the time the part wakes up.

5.3 Output Compare Operation in Idle Mode

When the device enters Idle mode, the system clock sources remain functional and the CPU stops executing code. The Output Compare x Stop in Idle Mode Control (OCSIDL) bit in the (OCxCON1<13>) register selects if the output capture module stops in Idle mode or continues operation in Idle mode.

- If the bit, OCSIDL = 1, then the module discontinues the operation in Idle mode. The module performs the same procedures when stopped in Idle mode (OCSIDL = 1) as it does for Sleep mode.
- If the bit, OCSIDL = 0, then the Output Compare x channel(s) operate during the CPU Idle mode. Furthermore, the time base must be enabled with the respective TSIDL bit set to a logic '0'; if internal, the timer is used as the clock source.

Note: The external Fault pins, if enabled for use, continue to control the associated OCx output pins while the device is in Sleep mode or in Idle mode.

5.4 Doze Mode

Output Compare operation in Doze mode is the same as in normal mode. When the device enters Doze mode, the system clock sources remain functional and the CPU may run at a slower clock rate.

5.5 Selective Peripheral Module Control

The Peripheral Module Disable x (PMDx) registers provide a method to disable the OCx module by stopping all the clock sources supplied to it. When the module is disabled through the appropriate PMDx control bit, it is in a minimum power consumption state. The control and status registers associated with the module will be disabled. Therefore, a write to these registers will have no effect, and the read values will be invalid and return to zero.
6.0 I/O PIN CONTROL

When the OCx module is enabled, the Input/Output (I/O) pin direction is controlled by the Output Compare x module. The Output Compare x module returns the I/O pin control back to the appropriate LATx and TRISx control bits when it is disabled. When the Simple PWM with Fault Protection Input mode is enabled, the OCFA/OCFB Fault pins must be configured as inputs by setting the respective TRISx bits. Enabling this special PWM mode does not configure the OCFA/OCFB Fault pins as inputs.

<table>
<thead>
<tr>
<th>Note:</th>
<th>Refer to the specific device data sheet for the availability of the Output Compare x module and Fault pins.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>If the PPS feature is present, the OCx module I/Os must be assigned to the required remappable pins before enabling the module.</td>
</tr>
</tbody>
</table>
## 7.0 REGISTER MAPS

A summary of the registers associated with the Output Compare with Dedicated Timer module is provided in Table 7-1.

### Table 7-1: Output Compare x with Dedicated Timer Register Map

<table>
<thead>
<tr>
<th>File Name</th>
<th>Bit 15</th>
<th>Bit 14</th>
<th>Bit 13</th>
<th>Bit 12</th>
<th>Bit 11</th>
<th>Bit 10</th>
<th>Bit 9</th>
<th>Bit 8</th>
<th>Bit 7</th>
<th>Bit 6</th>
<th>Bit 5</th>
<th>Bit 4</th>
<th>Bit 3</th>
<th>Bit 2</th>
<th>Bit 1</th>
<th>Bit 0</th>
<th>All Resets</th>
</tr>
</thead>
<tbody>
<tr>
<td>OCxCON1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>00000</td>
</tr>
<tr>
<td>OCxCON2</td>
<td>FLTMD</td>
<td>FLTOUT</td>
<td>FLTRIEN</td>
<td>OCINV</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>00000</td>
</tr>
<tr>
<td></td>
<td>OCxRS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>00000</td>
</tr>
<tr>
<td>OCxR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>00000</td>
</tr>
<tr>
<td>OCxTMR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>xxxxx</td>
</tr>
</tbody>
</table>

**Legend:** — = unimplemented, read as ‘0’. Reset values are shown in hexadecimal.

**Note:** For details on the Output Compare with Dedicated Timer map, refer to the specific device data sheet.
8.0 DESIGN TIPS

**Question 1:** The Output Compare x pin stops functioning even when the OCSIDL bit is not set. Why?

**Answer:** This is most likely to occur when the TSIDL (TxCON<13>) bit of the associated timer source is set if it is used as the clock source. Therefore, it is the timer that actually goes into Idle mode when the PWRSAV instruction is executed and the clock is not generated.

**Question 2:** Can I cascade OC2 and OC3?

**Answer:** No; cascading can be done only in pairs of OC1-OC2, OC3-OC4, etc., and OC9 cannot be paired.

**Question 3:** My device has PPS and I have mapped the OCx pins to the remappable pins. Still, it doesn’t work. Why?

**Answer:** Verify if the required values are actually written to the appropriate registers. Writing into PPS registers might require unlocking and locking sequences.
9.0 RELATED APPLICATION NOTES

This section lists application notes that are related to this section of the manual. These application notes may not be written specifically for the dsPIC33/PIC24F device families, but the concepts are pertinent and could be used with modification and possible limitations. The current application notes related to the Output Compare with Dedicated Timer module are:

<table>
<thead>
<tr>
<th>Title</th>
<th>Application Note #</th>
</tr>
</thead>
<tbody>
<tr>
<td>No related application notes at this time.</td>
<td></td>
</tr>
</tbody>
</table>

**Note:** Please visit the Microchip web site (www.microchip.com) for additional application notes and code examples for the dsPIC33/PIC24 families of devices.
10.0 REVISION HISTORY

Revision A (March 2014)
This is the initial released version of this document.
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.

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.

Trademarks
The Microchip name and logo, the Microchip logo, dsPIC, FlashFlex, KEEL00, KEEL00 logo, MPLAB, PIC, PICmicro, PICSTART, PIC18C logo, rPIC, SST, SST Logo, SuperFlash and UNI/O are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.
FilterLab, Hampshire, HI-TECH C, Linear Active Thermistor, MTP, SEEVAL and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A.
Silicon Storage Technology is a registered trademark of Microchip Technology Incorporated in other countries.
Analog-for-the-Digital Age, Application Maestro, BodyCom, chipKIT, chipKIT logo, CodeGuard, dsPICDEM, dsPICDEM.net, dsPICworks, dsSPEAK, ECAN, ECONOMONITOR, FanSense, Hi-TIDE, In-Circuit Serial Programming, ICSP, Mindi, MiWi, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, mTouch, Omniscient Code Generation, PICC, PICC-18, PICDEM, PICDEM.net, PICkit, PICtail, REAL ICE, rFLAB, Select Mode, SQI, Serial Quad I/O, Total Endurance, TSHARC, UniWinDriver, WiperLock, ZENA and Z-Scale 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.
GestIC and ULPP are registered trademarks 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.
© 2014, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.
Printed on recycled paper.
ISBN: 978-1-62077-958-3

QUALITY MANAGEMENT SYSTEM CERTIFIED BY DNV

ISO/TS 16949

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company’s quality system processes and procedures are for its PIC™ MCUs and dsPIC™ DSCs, KEELOQ™ code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip’s quality system for the design and manufacture of development systems is ISO 9001:2000 certified.
<table>
<thead>
<tr>
<th>Region</th>
<th>City</th>
<th>Contact Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>AMERICAS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Atlanta</td>
<td>Duluth, GA</td>
<td>Tel: 678-957-9614 Fax: 678-957-1455</td>
</tr>
<tr>
<td>Austin, TX</td>
<td>Tel: 512-257-3370</td>
<td></td>
</tr>
<tr>
<td>Boston</td>
<td>Westborough, MA</td>
<td>Tel: 774-760-0087 Fax: 774-760-0088</td>
</tr>
<tr>
<td>Chicago</td>
<td>Itasca, IL</td>
<td>Tel: 630-285-0071 Fax: 630-285-0075</td>
</tr>
<tr>
<td>Cleveland</td>
<td>Independence, OH</td>
<td>Tel: 216-447-0464 Fax: 216-447-0643</td>
</tr>
<tr>
<td>Dallas</td>
<td>Addison, TX</td>
<td>Tel: 972-818-7423 Fax: 972-818-2924</td>
</tr>
<tr>
<td>Detroit</td>
<td>Novi, MI</td>
<td>Tel: 248-848-4000</td>
</tr>
<tr>
<td>Houston, TX</td>
<td>Tel: 281-894-5983</td>
<td></td>
</tr>
<tr>
<td>Indianapolis</td>
<td>Noblesville, IN</td>
<td>Tel: 317-773-8323 Fax: 317-773-5453</td>
</tr>
<tr>
<td>Los Angeles</td>
<td>Mission Viejo, CA</td>
<td>Tel: 949-462-9523 Fax: 949-462-9608</td>
</tr>
<tr>
<td>New York, NY</td>
<td>Tel: 631-435-6000</td>
<td></td>
</tr>
<tr>
<td>San Jose, CA</td>
<td>Tel: 408-735-9110</td>
<td></td>
</tr>
<tr>
<td>Canada - Toronto</td>
<td>Tel: 905-673-0699 Fax: 905-673-6509</td>
<td></td>
</tr>
<tr>
<td>ASIA/PACIFIC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ASIA/PACIFIC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>India - Bangalore Tel: 91-80-3090-4444 Fax: 91-80-3090-4123 India - New Delhi Tel: 91-11-4160-8631 Fax: 91-11-4160-8632 India - Pune Tel: 91-20-3019-1500 Japan - Osaka Tel: 81-6-6152-7160 Fax: 81-6-6152-9310 Japan - Tokyo Tel: 81-3-6880-3770 Fax: 81-3-6880-3771 Korea - Daegu Tel: 82-53-744-4301 Fax: 82-53-744-4302 Korea - Seoul Tel: 82-2-554-7200 Fax: 82-2-558-5932 or 82-2-558-5934 Malaysia - Kuala Lumpur Tel: 60-3-6201-9857 Fax: 60-3-6201-9859 Malaysia - Penang Tel: 60-4-227-8870 Fax: 60-4-227-4068 Philippines - Manila Tel: 63-2-634-9065 Fax: 63-2-634-9069 Singapore Tel: 65-6334-8870 Fax: 65-6334-8850 Taiwan - Hsin Chu Tel: 886-3-5778-366 Fax: 886-3-5770-955 Taiwan - Kaohsiung Tel: 886-7-213-7830 Taiwan - Taipei Tel: 886-2-2508-8600 Fax: 886-2-2508-0102 Thailand - Bangkok Tel: 66-2-694-1351 Fax: 66-2-694-1350</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ASIA/PACIFIC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EU EUROPE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Austria - Wels</td>
<td>Tel: 43-7242-2244-39 Fax: 43-7242-2244-393</td>
<td></td>
</tr>
<tr>
<td>Denmark - Copenhagen</td>
<td>Tel: 45-4450-2828 Fax: 45-4485-2829</td>
<td></td>
</tr>
<tr>
<td>France - Paris</td>
<td>Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79</td>
<td></td>
</tr>
<tr>
<td>Germany - Dusseldorf</td>
<td>Tel: 49-2129-3766400</td>
<td></td>
</tr>
<tr>
<td>Germany - Munich</td>
<td>Tel: 49-89-627-144-0 Fax: 49-89-627-144-44</td>
<td></td>
</tr>
<tr>
<td>Germany - Pforzheim</td>
<td>Tel: 49-7231-424750</td>
<td></td>
</tr>
<tr>
<td>Italy - Milan</td>
<td>Tel: 39-0331-742611 Fax: 39-0331-466781</td>
<td></td>
</tr>
<tr>
<td>Italy - Venice</td>
<td>Tel: 39-049-7625286</td>
<td></td>
</tr>
<tr>
<td>Netherlands - Drunen</td>
<td>Tel: 31-416-690399 Fax: 31-416-690340</td>
<td></td>
</tr>
<tr>
<td>Poland - Warsaw</td>
<td>Tel: 48-22-3325737</td>
<td></td>
</tr>
<tr>
<td>Spain - Madrid</td>
<td>Tel: 34-91-708-08-90 Fax: 34-91-708-08-91</td>
<td></td>
</tr>
<tr>
<td>Sweden - Stockholm</td>
<td>Tel: 46-8-5090-4654</td>
<td></td>
</tr>
<tr>
<td>UK - Wokingham</td>
<td>Tel: 44-118-921-5800 Fax: 44-118-921-5820</td>
<td></td>
</tr>
</tbody>
</table>