Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	20MHz
Connectivity	-
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	11
Program Memory Size	3.5KB (2K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	128 x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5V
Data Converters	A/D 8x10b; D/A 1x9b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	14-SOIC (0.154", 3.90mm Width)
Supplier Device Package	14-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16hv753-e-sl

2.3.5 PIE2 REGISTER

The PIE2 register contains the Peripheral Interrupt Enable bits, as shown in Register 2-5.

Note: Bit PEIE of the INTCON register must be set to enable any peripheral interrupt.

REGISTER 2-5: PIE2: PERIPHERAL INTERRUPT ENABLE REGISTER 1

U-0	U-0	R/W-0	R/W-0	U-0	R/W-0	U-0	R/W-0
—	—	C2IE	C1IE	—	COG1IE	—	CCP1IE
bit 7							bit 0

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 7-6 **Unimplemented:** Read as '0'

bit 5 **C2IE:** Comparator 2 Interrupt Enable bit
1 = Enables the Comparator 2 interrupt
0 = Disables the Comparator 2 interrupt

bit 4 **C1IE:** Comparator 1 Interrupt Enable bit
1 = Enables the Comparator 1 interrupt
0 = Disables the Comparator 1 interrupt

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

bit 2 **COG1IE:** COG 1 Interrupt Flag bit
1 = COG1 interrupt enabled
0 = COG1 interrupt disabled

bit 1 **Unimplemented:** Read as '0'

bit 0 **CCP1IE:** CCP1 Interrupt Enable bit
1 = Enables the CCP1 interrupt
0 = Disables the CCP1 interrupt

REGISTER 5-7: IOCAP: INTERRUPT-ON-CHANGE POSITIVE EDGE REGISTER

U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	IOCAP5	IOCAP4	IOCAP3	IOCAP2	IOCAP1	IOCAP0
bit 7							bit 0

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

bit 7-6 **Unimplemented**: Read as ‘0’

bit 5-0 **IOCAP<5:0>:** Interrupt-on-Change Positive Edge Enable bits
1 = Interrupt-on-Change enabled on the pin for a positive going edge. Associated Status bit and interrupt flag will be set upon detecting an edge.
0 = Interrupt-on-Change disabled for the associated pin.

REGISTER 5-8: IOCAN: INTERRUPT-ON-CHANGE NEGATIVE EDGE REGISTER

U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	IOCAN5	IOCAN4	IOCAN3	IOCAN2	IOCAN1	IOCAN0
bit 7							bit 0

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

bit 7-6 **Unimplemented**: Read as ‘0’

bit 5-0 **IOCAN<5:0>:** Interrupt-on-Change Negative Edge Enable bits
1 = Interrupt-on-Change enabled on the pin for a negative going edge. Associated Status bit and interrupt flag will be set upon detecting an edge.
0 = Interrupt-on-Change disabled for the associated pin.

REGISTER 5-9: IOCAF: INTERRUPT-ON-CHANGE FLAG REGISTER

U-0	U-0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0
—	—	IOCAF5	IOCAF4	IOCAF3	IOCAF2	IOCAF1	IOCAF0
bit 7							bit 0

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

bit 7-6 **Unimplemented**: Read as ‘0’

bit 5-0 **IOCAF<5:0>:** Interrupt-on-Change Flag bits
1 = An enabled change was detected on the associated pin.
Set when IOCAPx = 1 and a rising edge was detected on RBx, or when IOCANx = 1 and a falling edge was detected on RAx.
0 = No change was detected, or the user cleared the detected change.

TABLE 0-1: SUMMARY OF REGISTERS ASSOCIATED WITH PORTC

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ADCON0	ADFM	—	CHS<3:0>				GO/$\overline{\text{DONE}}$	ADON	109
ADCON1	—	ADCS<2:0>			—	—	—	ADPREF1	110
ANSELC	—	—	—	—	ANSC3	ANSC2	ANSC1	ANSC0	44
APFCON	—	—	—	T1GSEL	—	—	—	—	40
CM1CON0	C1ON	C1OUT	C1OE	C1POL	C1ZLF	C1SP	C1HYS	C1SYNC	129
CM2CON0	C2ON	C2OUT	C2OE	C2POL	C2ZLF	C2SP	C2HYS	C2SYNC	129
CM1CON1	C1INTP	C1INTN	C1PCH<2:0>			C1NCH<2:0>			130
CM2CON1	C2NTP	C2INTN	C2PCH<2:0>			C2NCH<2:0>			130
DAC1CON0	DACEN	DACFM	DACOE	—	DACPSS1	DACPSS0	—	—	120
IOCCF	—	—	IOCCF5	IOCCF4	IOCCF3	IOCCF2	IOCCF1	IOCCF0	45
IOCCN	—	—	IOCCN5	IOCCN4	IOCCN3	IOCCN2	IOCCN1	IOCCN0	45
IOCCP	—	—	IOCCP5	IOCCP4	IOCCP3	IOCCP2	IOCCP1	IOCCP0	45
LATC	—	—	LATC5	LATC4	LATC3	LATC2	LATC1	LATC0	43
OPTION_REG	$\overline{\text{RAPU}}$	INTEDG	T0CS	T0SE	PSA	PS<2:0>			16
PORTC	—	—	RC5	RC4	RC3	RC2	RC1	RC0	43
SLRCONC	—	—	SLRC5	SLRC4	—	—	—	—	50
TRISC	—	—	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	43

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

6.0 TIMER0 MODULE

The Timer0 module is an 8-bit timer/counter with the following features:

- 8-bit timer/counter register (TMR0)
- 8-bit prescaler (shared with Watchdog Timer)
- Programmable internal or external clock source
- Programmable external clock edge selection
- Interrupt on overflow

Figure 6-1 is a block diagram of the Timer0 module.

6.1 Timer0 Operation

When used as a timer, the Timer0 module can be used as either an 8-bit timer or an 8-bit counter.

6.1.1 8-BIT TIMER MODE

When used as a timer, the Timer0 module will increment every instruction cycle (without prescaler). Timer mode is selected by clearing the T0CS bit of the OPTION register to '0'.

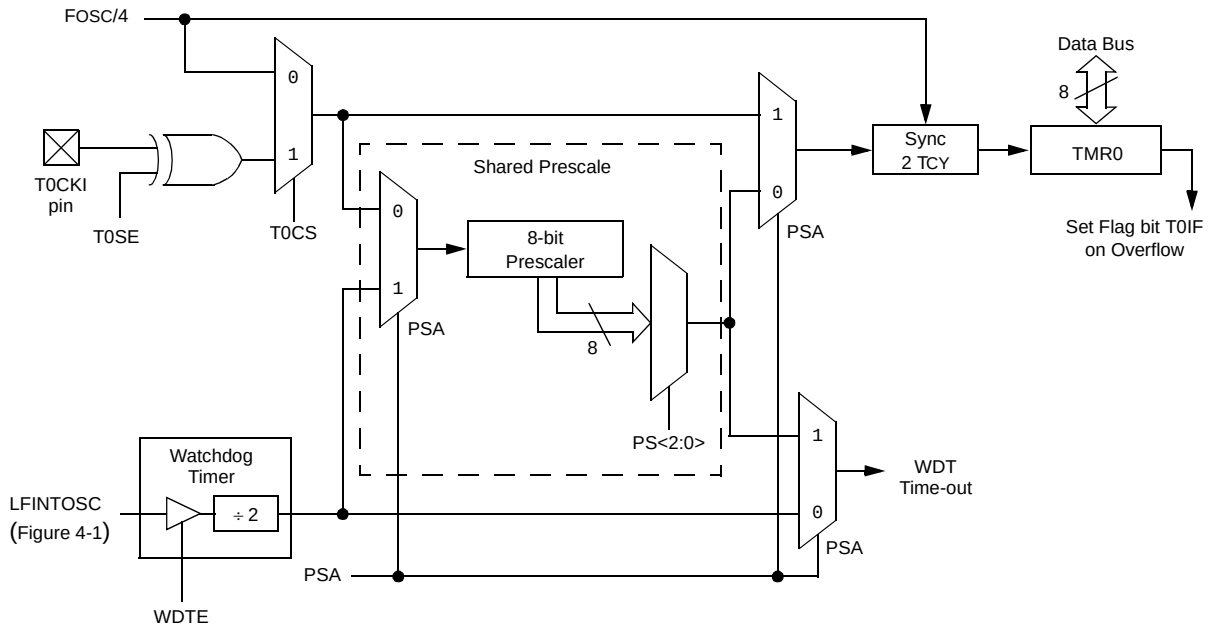
When TMR0 is written, the increment is inhibited for two instruction cycles immediately following the write.

Note:	The value written to the TMR0 register can be adjusted, in order to account for the two instruction cycle delay when TMR0 is written.

6.1.2 8-BIT COUNTER MODE

When used as a counter, the Timer0 module will increment on every rising or falling edge of the T0CKI pin. The incrementing edge is determined by the T0SE bit of the OPTION_REG register. Counter mode is selected by setting the T0CS bit of the OPTION register to '1'.

FIGURE 6-1: TIMER0 WITH SHARED PRESCALE BLOCK DIAGRAM



Note 1: T0SE, T0CS, PSA, PS<2:0> are bits in Register 6-1.

2: WDTE bit is in Register 19-1.

7.5.2.1 T1G Pin Gate Operation

The T1G pin is one source for Timer1 gate control. It can be used to supply an external source to the Timer1 gate circuitry.

7.5.2.2 Timer0 Overflow Gate Operation

When Timer0 increments from FFh to 00h, a low-to-high pulse will automatically be generated and internally supplied to the Timer1 gate circuitry.

7.5.2.3 C1OUT/C2OUT Gate Operation

The outputs from the Comparator C1 and C2 modules can be used as gate sources for the Timer1 module.

7.5.3 TIMER1 GATE TOGGLE MODE

When Timer1 Gate Toggle mode is enabled, it is possible to measure the full-cycle length of a Timer1 gate signal, as opposed to the duration of a single-level pulse.

The Timer1 gate source is routed through a flip-flop that changes state on every incrementing edge of the signal. See Figure 7-4 for timing details.

Timer1 Gate Toggle mode is enabled by setting the T1GTM bit of the T1GCON register. When the T1GTM bit is cleared, the flip-flop is cleared and held clear. This is necessary in order to control which edge is measured.

Note:	Enabling Toggle mode at the same time as changing the gate polarity may result in indeterminate operation.

7.5.4 TIMER1 GATE SINGLE-PULSE MODE

When Timer1 Gate Single-Pulse mode is enabled, it is possible to capture a single-pulse gate event. Timer1 Gate Single-Pulse mode is first enabled by setting the T1GSPM bit in the T1GCON register. Next, the T1GGO/$\overline{\text{DONE}}$ bit in the T1GCON register must be set. The Timer1 will be fully enabled on the next incrementing edge. On the next trailing edge of the pulse, the T1GGO/$\overline{\text{DONE}}$ bit will automatically be cleared. No other gate events will be allowed to increment Timer1 until the T1GGO/$\overline{\text{DONE}}$ bit is once again set in software.

Clearing the T1GSPM bit of the T1GCON register will also clear the T1GGO/$\overline{\text{DONE}}$ bit. See Figure 7-5 for timing details.

Enabling the Toggle mode and the Single-Pulse mode simultaneously will permit both sections to work together. This allows the cycle times on the Timer1 gate source to be measured. See Figure 7-6 for timing details.

7.5.5 TIMER1 GATE VALUE STATUS

When Timer1 gate value status is utilized, it is possible to read the most current level of the gate control value. The value is stored in the T1GVAL bit in the T1GCON register. The T1GVAL bit is valid even when the Timer1 gate is not enabled (TMR1GE bit is cleared).

7.5.6 TIMER1 GATE EVENT INTERRUPT

When Timer1 gate event interrupt is enabled, it is possible to generate an interrupt upon the completion of a gate event. When the falling edge of T1GVAL occurs, the TMR1GIF flag bit in the PIR1 register will be set. If the TMR1GIE bit in the PIE1 register is set, then an interrupt will be recognized.

The TMR1GIF flag bit operates even when the Timer1 gate is not enabled (TMR1GE bit is cleared).

7.6 Timer1 Interrupt

The Timer1 register pair (TMR1H:TMR1L) increments to FFFFh and rolls over to 0000h. When Timer1 rolls over, the Timer1 interrupt flag bit of the PIR1 register is set. To enable the interrupt on rollover, you must set these bits:

- TMR1ON bit of the T1CON register
- TMR1IE bit of the PIE1 register
- PEIE bit of the INTCON register
- GIE bit of the INTCON register

The interrupt is cleared by clearing the TMR1IF bit in the Interrupt Service Routine.

Note:	The TMR1H:TMR1L register pair and the TMR1IF bit should be cleared before enabling interrupts.

7.7 Timer1 Operation During Sleep

Timer1 can only operate during Sleep when set up in Asynchronous Counter mode or with the internal watchdog clock source. In this mode, the clock source can be used to increment the counter. To set up the timer to wake the device:

- TMR1ON bit of the T1CON register must be set
- TMR1IE bit of the PIE1 register must be set
- PEIE bit of the INTCON register must be set
- $\overline{\text{T1SYNC}}$ bit of the T1CON register must be set
- TMR1CS bits of the T1CON register must be configured
- TMR1GE bit of the T1GCON register must be configured

The device will wake-up on an overflow and execute the next instructions. If the GIE bit of the INTCON register is set, the device will call the Interrupt Service Routine (0004h).

7.8 CCP Capture/Compare Time Base

The CCP module uses the TMR1H:TMR1L register pair as the time base when operating in Capture or Compare mode.

In Capture mode, the value in the TMR1H:TMR1L register pair is copied into the CCPR1H:CCPR1L register pair on a configured event.

In Compare mode, an event is triggered when the value CCPR1H:CCPR1L register pair matches the value in the TMR1H:TMR1L register pair. This event can be a Special Event Trigger.

For more information, see **Section 10.0 “Capture/ Compare/PWM Modules”**.

7.9 CCP Special Event Trigger

When the CCP is configured to trigger a special event, the trigger will clear the TMR1H:TMR1L register pair. This special event does not cause a Timer1 interrupt. The CCP module may still be configured to generate a CCP interrupt.

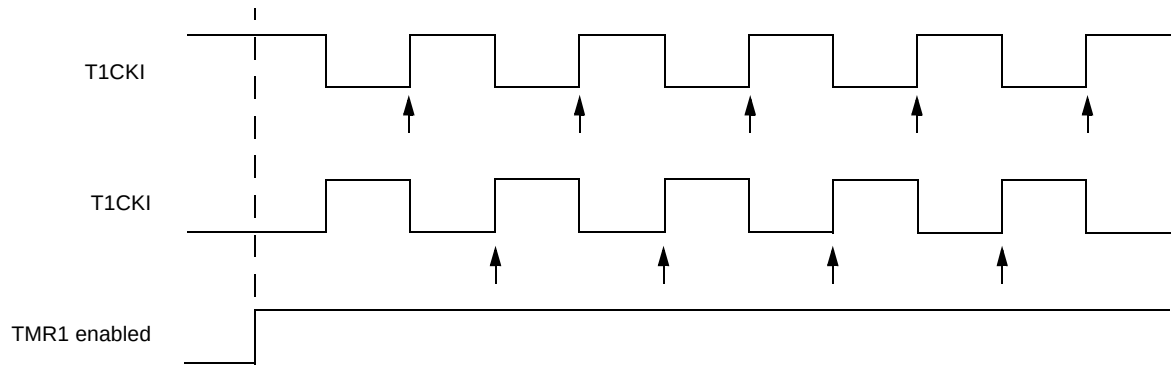
In this mode of operation, the CCPR1H:CCPR1L register pair becomes the period register for Timer1.

Timer1 should be synchronized to the FOSC/4 to utilize the Special Event Trigger. Asynchronous operation of Timer1 can cause a Special Event Trigger to be missed.

In the event that a write to TMR1H or TMR1L coincides with a Special Event Trigger from the CCP, the write will take precedence.

For more information, see **Section 12.2.5 “Special Event Trigger”**.

FIGURE 7-2: TIMER1 INCREMENTING EDGE



Note 1: Arrows indicate counter increments.

2: In Counter mode, a falling edge must be registered by the counter prior to the first incrementing rising edge of the clock.

FIGURE 7-3: TIMER1 GATE COUNT ENABLE MODE

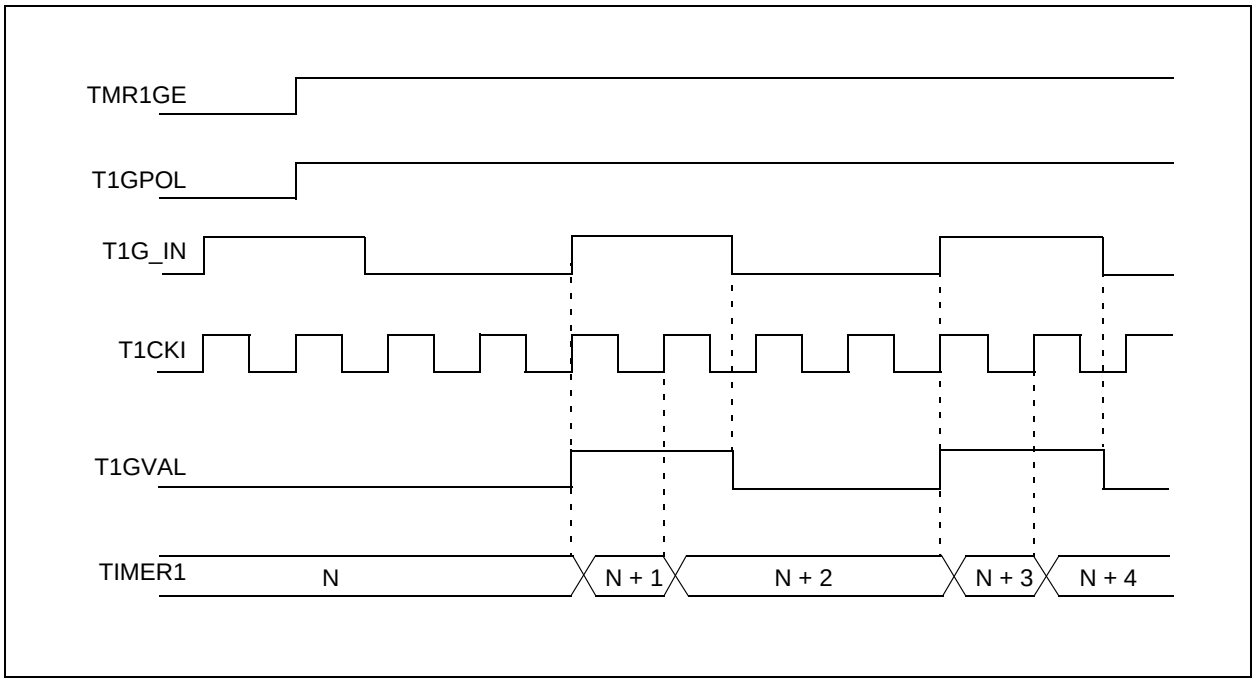


FIGURE 7-4: TIMER1 GATE TOGGLE MODE

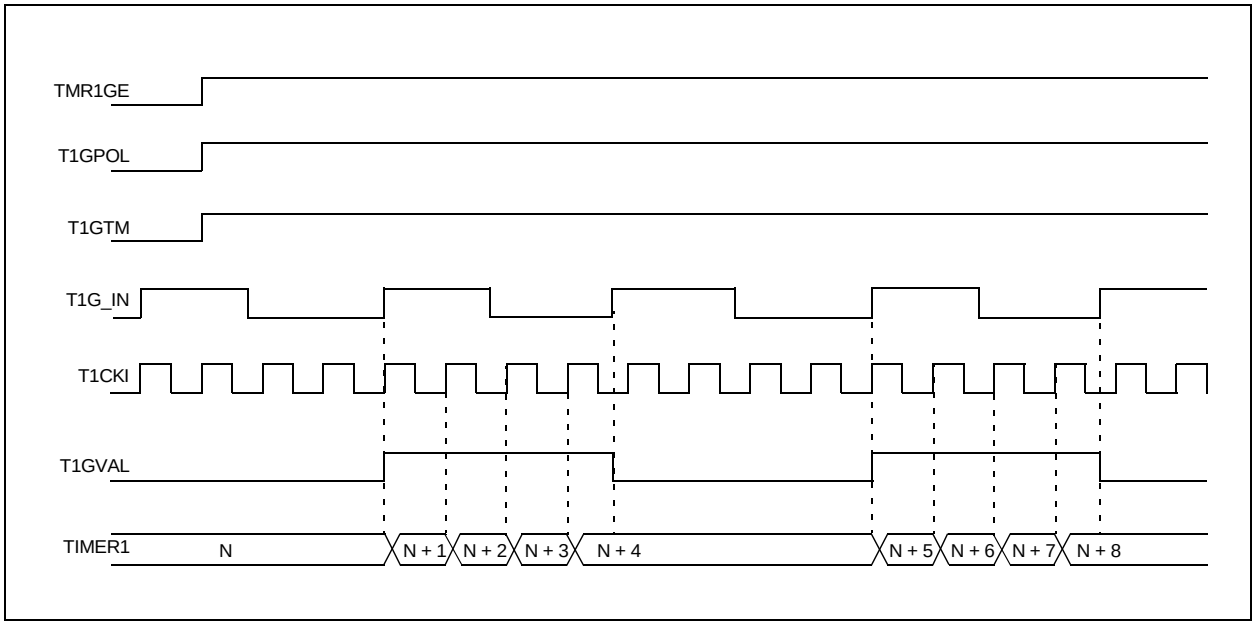
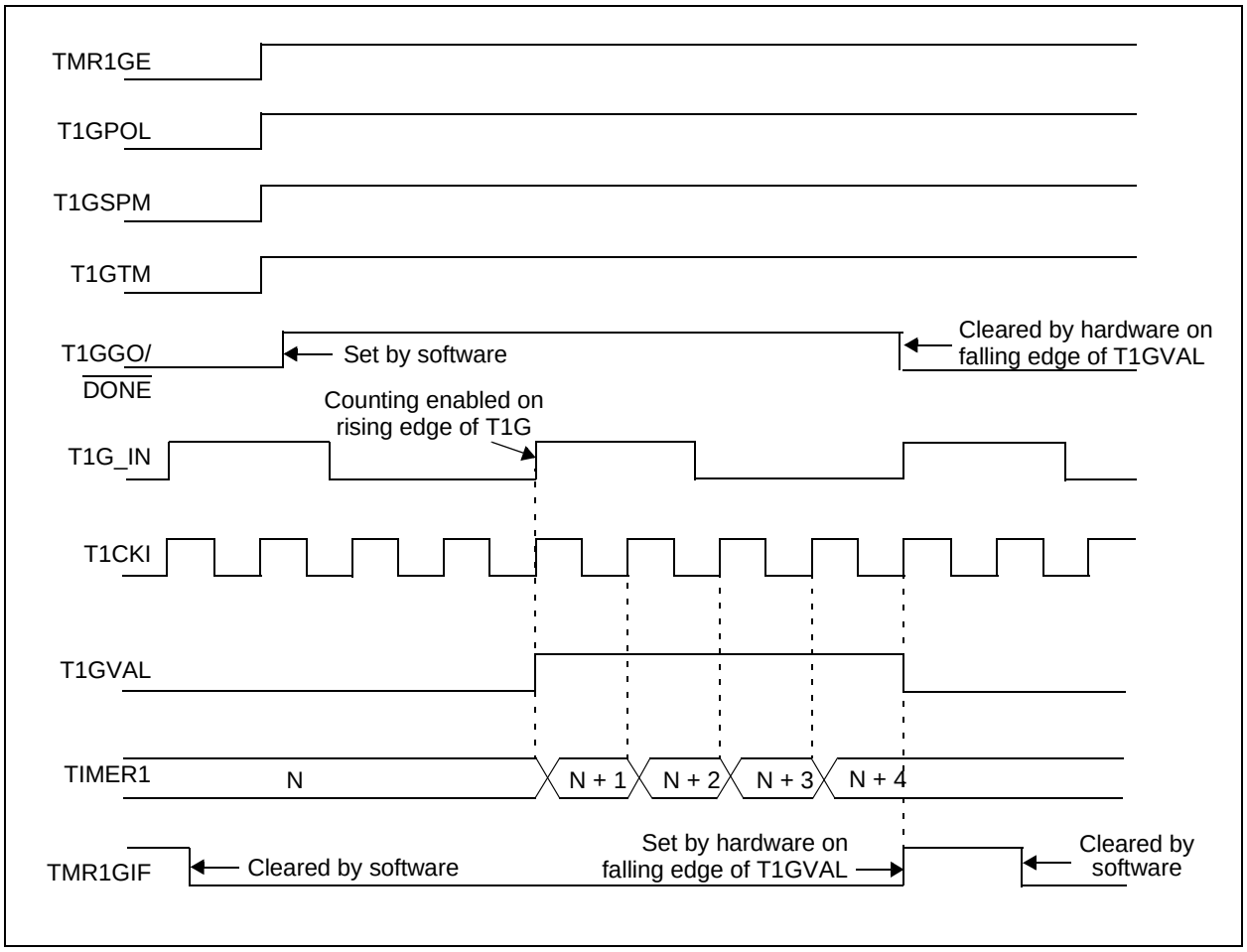


FIGURE 7-6: TIMER1 GATE SINGLE-PULSE AND TOGGLE COMBINED MODE



10.4 Register Definitions: CCP Control

REGISTER 10-1: CCP1CON: CCP1 CONTROL REGISTER

U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	DC1B<1:0>		CCP1M<3:0>			
bit 7							bit 0

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

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-4 **DC1B<1:0>:** PWM Duty Cycle Least Significant bits

Capture mode:
Unused

Compare mode:
Unused

PWM mode:
These bits are the two LSbs of the PWM duty cycle. The eight MSbs are found in CCPR1L.

bit 3-0 **CCP1M<3:0>:** CCP1 Mode Select bits

0000 = Capture/Compare/PWM off (resets CCP1 module)
0001 = Reserved
0010 = Compare mode: toggle output on match
0011 = Reserved

0100 = Capture mode: every falling edge
0101 = Capture mode: every rising edge
0110 = Capture mode: every 4th rising edge
0111 = Capture mode: every 16th rising edge

1000 = Compare mode: initialize CCP1 pin low; set output on compare match (set CCP1IF)
1001 = Compare mode: initialize CCP1 pin high; clear output on compare match (set CCP1IF)
1010 = Compare mode: generate software interrupt only; CCP1 pin reverts to I/O state
1011 = Compare mode: Special Event Trigger (CCP1 resets Timer, sets CCP1IF bit, and starts A/D conversion if A/D module is enabled)
11xx = PWM mode

FIGURE 11-4: TYPICAL COG OPERATION WITH CCP1

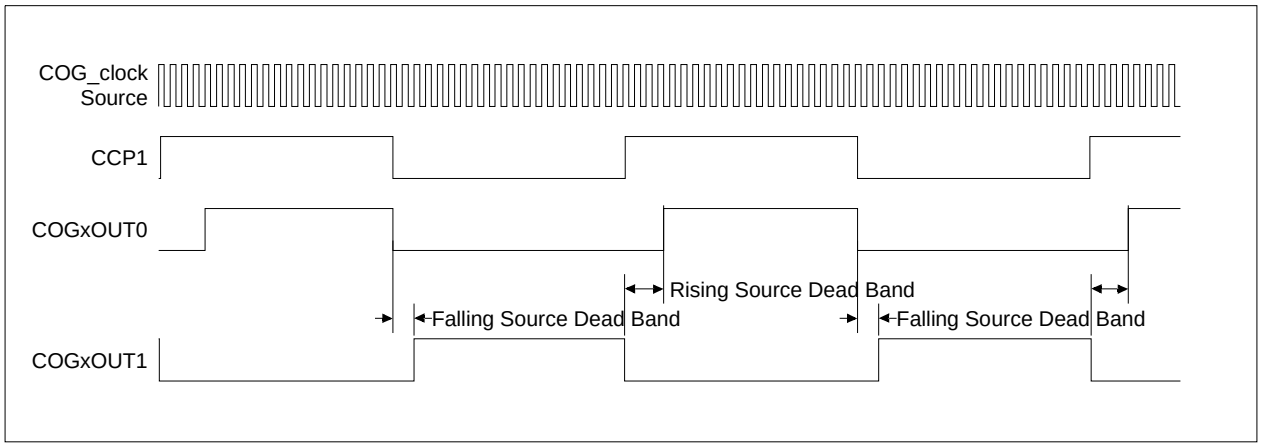


FIGURE 11-5: COG OPERATION WITH CCP1 AND PHASE DELAY

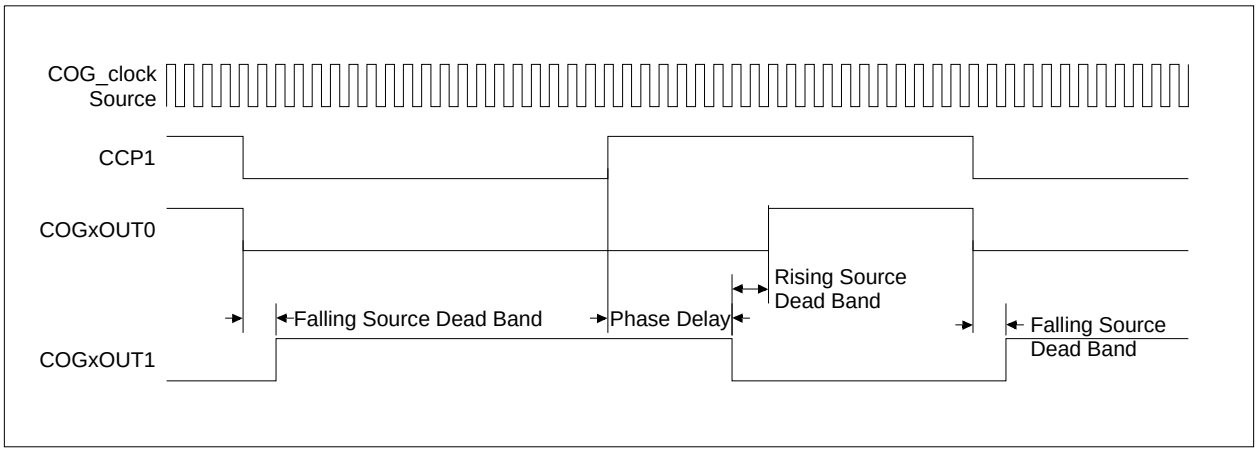
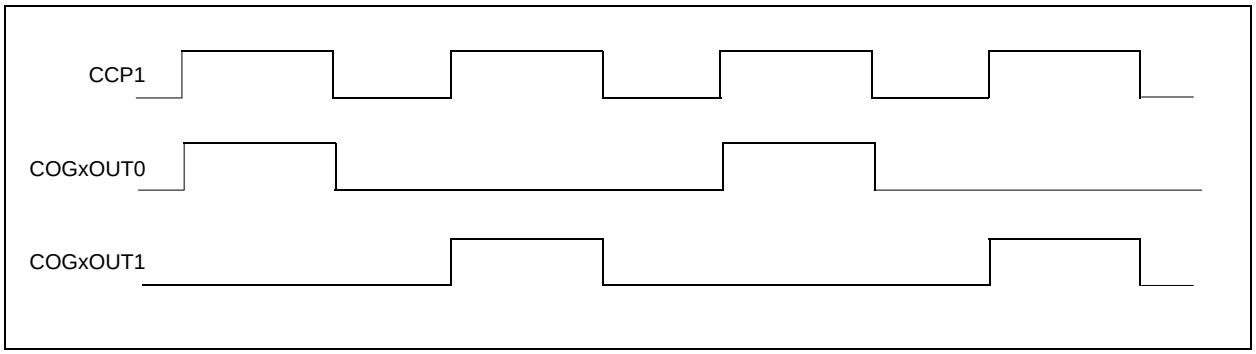


FIGURE 11-6: COG OPERATION IN PUSH-PULL MODE WITH CCP1



REGISTER 11-7: COGxASD0: COG AUTO-SHUTDOWN CONTROL REGISTER 0

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	U-0
GxASDE	GxARSEN	GxASD1L<1:0>		GxASD0L<1:0>		—	—
bit 7							bit 0

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

bit 7 **GxASDE:** Auto-Shutdown Event Status bit
1 = COG is in the shutdown state
0 = COG is either not in the shutdown state or will exit the shutdown state on the next rising event

bit 6 **GxARSEN:** Auto-Restart Enable bit
1 = Auto-restart is enabled
0 = Auto-restart is disabled

bit 5-4 **GxASD1L<1:0>:** COGxOUT1 Auto-Shutdown Override Level Select bits
11 = COGxOUT1 is tri-stated when shutdown is active
10 = The inactive state of the pin, including polarity, is placed on COGxOUT1 when shutdown is active
01 = A logic ‘1’ is placed on COGxOUT1 when shutdown is active
00 = A logic ‘0’ is placed on COGxOUT1 when shutdown is active

bit 3-2 **GxASD0L<1:0>:** COGxOUT0 Auto-Shutdown Override Level Select bits
11 = COGxOUT0 is tri-stated when shutdown is active
10 = The inactive state of the pin, including polarity, is placed on COGxOUT0 when shutdown is active
01 = A logic ‘1’ is placed on COGxOUT0when shutdown is active
00 = A logic ‘0’ is placed on COGxOUT0when shutdown is active

bit 1-0 **Unimplemented:** Read as ‘0’

TABLE 11-1: SUMMARY OF REGISTERS ASSOCIATED WITH COG

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ANSELA	—	—	—	ANSA4	—	ANSA2	ANSA1	ANSA0	44
APFCON	—	—	—	T1GSEL	—	—	—	—	40
COG1PHR	—	—	—	—	G1PHR<3:0>				102
COG1PHF	—	—	—	—	G1PHF<3:0>				102
COG1BKR	—	—	—	—	G1BKR<3:0>				101
COG1BKF	—	—	—	—	G1BKF<3:0>				101
COG1DBR	—	—	—	—	G1DBR<3:0>				100
COG1DBF	—	—	—	—	G1DBF<3:0>				100
COG1RIS	—	G1RIHLT2	G1RIHLT1	G1RIT2M	G1RIFLT	G1RICCP1	G1RIC2	G1RIC1	94
COG1RSIM	—	G1RMHLT2	G1RMHLT1	G1RMT2M	G1RMFLT	G1RMCCP1	G1RMC2	G1RMC1	95
COG1FIS	—	G1FIHLT2	G1FIHLT1	G1FIT2M	G1FIFLT	G1FICCP1	G1FIC2	G1FIC1	96
COG1FSIM	—	G1FMHLT2	G1FMHLT1	G1FMT2M	G1FMFLT	G1FMCCP1	G1FMC2	G1FMC1	97
COG1CON0	G1EN	G1OE1	G1OE0	G1POL1	G1POL0	G1LD	—	G1MD	92
COG1CON1	G1RDBTS	G1FDBTS	—	—	—	—	G1CS<1:0>		93
COG1ASD0	G1ASDE	G1ARSEN	G1ASD1L<1:0>		G1ASD0L<1:0>		—	—	98
COG1ASD1	—	—	—	G1ASDSHLT2	G1ASDSHLT1	G1ASDSC2	G1ASDSC1	G1ASDSFLT	99
INTCON	GIE	PEIE	T0IE	INTE	IOCIE	T0IF	INTF	IOCIF	17
LATA	—	—	LATA5	LATA4	—	LATA2	LATA1	LATA0	43
PIE2	—	—	C2IE	C1IE	—	COG1IE	—	CCP1IE	19
PIR2	—	—	C2IF	C1IF	—	COG1IF	—	CCP1IF	21
TRISA	—	—	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	43

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

15.2 Comparator Control

Each comparator has two control registers: CMxCON0 and CMxCON1.

The CMxCON0 registers (see Register 15-1) contain Control and Status bits for the following:

- Enable
- Output selection
- Output pin enable
- Output polarity
- Speed/Power selection
- Hysteresis enable
- Output synchronization

The CMxCON1 registers (see Register 15-2) contain Control bits for the following:

- Interrupt edge polarity (rising and/or falling)
- Positive input channel selection
- Negative input channel selection

15.2.1 COMPARATOR ENABLE

Setting the CxON bit of the CMxCON0 register enables the comparator for operation. Clearing the CxON bit disables the comparator resulting in minimum current consumption.

15.2.2 COMPARATOR OUTPUT SELECTION

The output of the comparator can be monitored by reading either the CxOUT bit of the CMxCON0 register or the MCOUTx bit of the CMOUT register. In order to make the output available for an external connection, the following conditions must be true:

- CxOE bit of the CMxCON0 register must be set
- Corresponding TRIS bit must be cleared
- CxON bit of the CMxCON0 register must be set

Note 1:	The CxOE bit of the CMxCON0 register overrides the PORT data latch. Setting the CxON bit of the CMxCON0 register has no impact on the port override.
2:	The internal output of the comparator is latched with each instruction cycle. Unless otherwise specified, external outputs are not latched.

15.2.3 COMPARATOR OUTPUT POLARITY

Inverting the output of the comparator is functionally equivalent to swapping the comparator inputs. The polarity of the comparator output can be inverted by setting the CxPOL bit of the CMxCON0 register. Clearing the CxPOL bit results in a non-inverted output.

Table 15-1 shows the output state versus input conditions, including polarity control.

TABLE 15-1: COMPARATOR OUTPUT STATE VS. INPUT CONDITIONS

Input Condition	CxPOL	CxOUT
CxVN > CxVP	0	0
CxVN < CxVP	0	1
CxVN > CxVP	1	1
CxVN < CxVP	1	0

15.2.4 COMPARATOR SPEED/POWER SELECTION

The trade-off between speed or power can be optimized during program execution with the CxSP control bit. The default state for this bit is '1' which selects the normal speed mode. Device power consumption can be optimized at the cost of slower comparator propagation delay by clearing the CxSP bit to '0'.

15.3 Comparator Hysteresis

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

See **Section 22.0 "Electrical Specifications"** for more information.

15.4 Timer1 Gate Operation

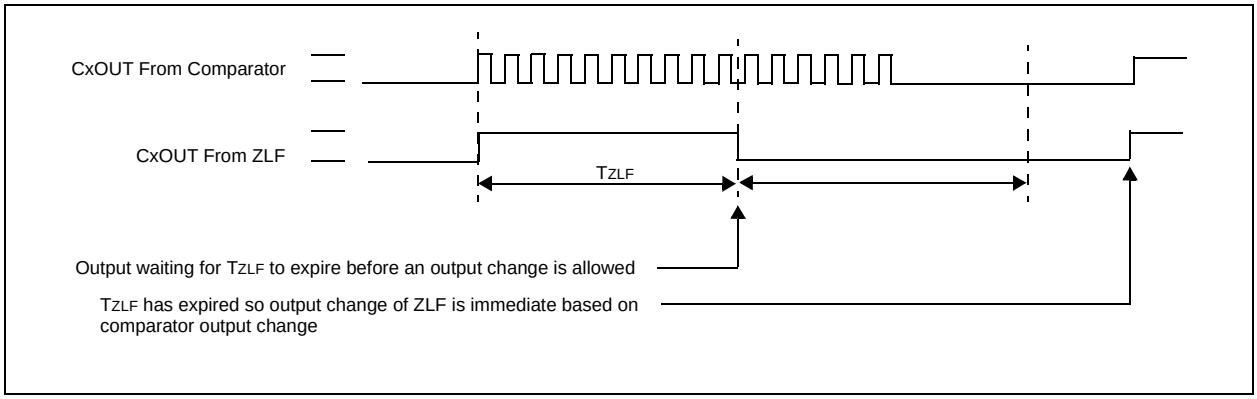
The output resulting from a comparator operation can be used as a source for gate control of Timer1. See **Section 7.5 "Timer1 Gate"** for more information. This feature is useful for timing the duration or interval of an analog event.

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

15.10 Zero Latency Filter

In high-speed operation, and under proper circuit conditions, it is possible for the comparator output to oscillate. This oscillation can have adverse effects on the hardware and software relying on this signal. Therefore, a digital filter has been added to the comparator output to suppress the comparator output oscillation. Once the comparator output changes, the output is prevented from reversing the change for a nominal time of 20 ns. This allows the comparator output to stabilize without affecting other dependent devices. Refer to Figure 15-3.

FIGURE 15-3: COMPARATOR ZERO LATENCY FILTER OPERATION



19.2 Calibration Bits

The 8 MHz internal oscillator is factory-calibrated. These calibration values are stored in fuses located in the Calibration Word (2008h). The Calibration Word is not erased when using the specified bulk erase sequence in the *PIC16F753/HV753 Flash Memory Programming Specification* (DS41686) and thus, does not require reprogramming.

19.3 Reset

The PIC16F753/HV753 device differentiates between various kinds of Reset:

a) Power-on Reset (POR)
b) WDT Reset during normal operation
c) WDT Reset during Sleep
d) $\overline{\text{MCLR}}$ Reset during normal operation
e) $\overline{\text{MCLR}}$ Reset during Sleep
f) Brown-out Reset (BOR)

Some registers are not affected in any Reset condition; their status is unknown on POR and unchanged in any other Reset. Most other registers are reset to a "Reset state" on:

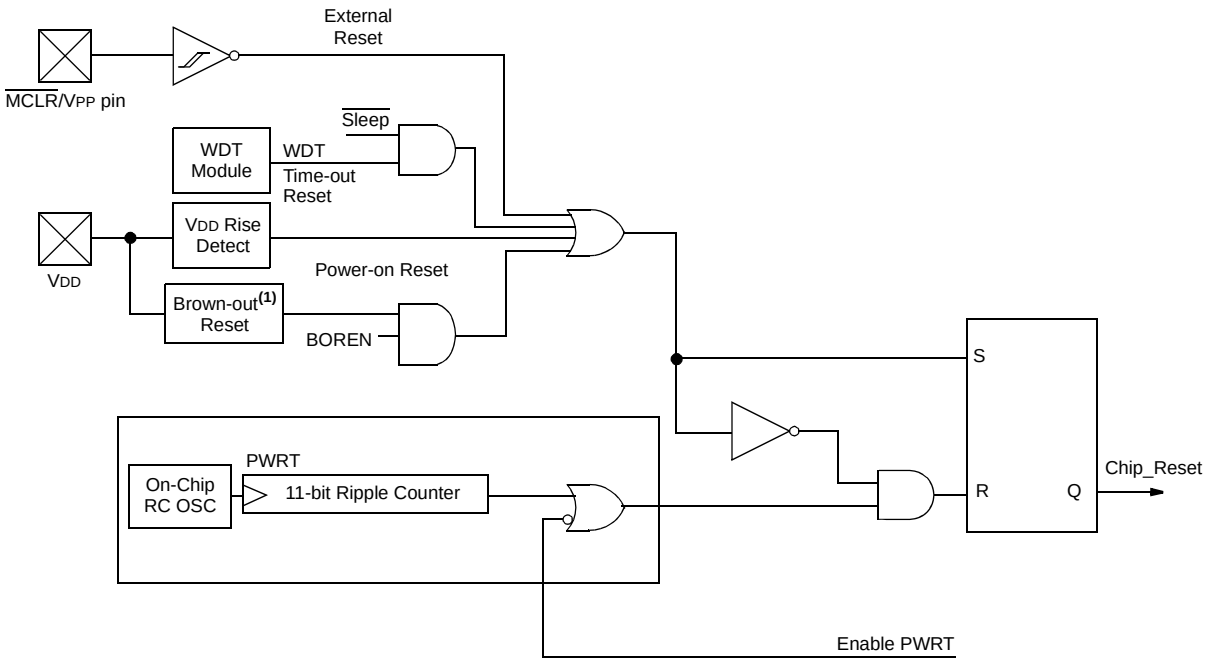
- Power-on Reset
- $\overline{\text{MCLR}}$ Reset
- $\overline{\text{MCLR}}$ Reset during Sleep
- WDT Reset
- Brown-out Reset (BOR)

WDT wake-up does not cause register Resets in the same manner as a WDT Reset since wake-up is viewed as the resumption of normal operation. $\overline{\text{TO}}$ and $\overline{\text{PD}}$ bits are set or cleared differently in different Reset situations, as indicated in Table 19-2. Software can use these bits to determine the nature of the Reset. See Table 19-4 for a full description of Reset states of all registers.

A simplified block diagram of the On-Chip Reset Circuit is shown in Figure 19-1.

The $\overline{\text{MCLR}}$ Reset path has a noise filter to detect and ignore small pulses. See **Section 22.0 "Electrical Specifications"** for pulse-width specifications.

FIGURE 19-1: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT



Note 1: Refer to the Configuration Word register (Register 19-1).

19.4 Interrupts

The PIC16F753/HV753 has multiple sources of interrupt:

- External Interrupt (INT pin)
- Interrupt-On-Change (IOC) Interrupts
- Timer0 Overflow Interrupt
- Timer1 Overflow Interrupt
- Timer2 Match Interrupt
- Hardware Limit Timer (HLT) Interrupt
- Comparator Interrupt (C1/C2)
- ADC Interrupt
- Complementary Output Generator (COG)
- CCP1 Interrupt
- Flash Memory Self-Write

The Interrupt Control register (INTCON) and Peripheral Interrupt Request Registers (PIRx) record individual interrupt requests in flag bits. The INTCON register also has individual and global interrupt enable bits.

The Global Interrupt Enable bit, GIE of the INTCON register, enables (if set) all unmasked interrupts, or disables (if cleared) all interrupts. Individual interrupts can be disabled through their corresponding enable bits in the INTCON register and PIEx registers. GIE is cleared on Reset.

When an interrupt is serviced, the following actions occur automatically:

- The GIE is cleared to disable any further interrupt.
- The return address is pushed onto the stack.
- The PC is loaded with 0004h.

The Return from Interrupt instruction, `RETFIE`, exits the interrupt routine, as well as sets the GIE bit, which re-enables unmasked interrupts.

The following interrupt flags are contained in the INTCON register:

- INT Pin Interrupt
- Interrupt-On-Change (IOC) Interrupts
- Timer0 Overflow Interrupt

The peripheral interrupt flags are contained in the PIR1 and PIR2 registers. The corresponding interrupt enable bit is contained in the PIE1 and PIE2 registers.

The following interrupt flags are contained in the PIR1 register:

- A/D Interrupt
- Comparator Interrupt
- Timer1 Overflow Interrupt
- Timer2 Match Interrupt
- Enhanced CCP Interrupt

For external interrupt events, such as the INT pin or PORTA change interrupt, the interrupt latency will be three or four instruction cycles. The exact latency depends upon when the interrupt event occurs (see Figure 19-8). The latency is the same for one or two-cycle instructions. Once in the Interrupt Service Routine, the source(s) of the interrupt can be determined by polling the interrupt flag bits. The interrupt flag bit(s) must be cleared in software before re-enabling interrupts to avoid multiple interrupt requests.

Note 1:	Individual interrupt flag bits are set, regardless of the status of their corresponding mask bit or the GIE bit.
2:	When an instruction that clears the GIE bit is executed, any interrupts that were pending for execution in the next cycle are ignored. The interrupts, which were ignored, are still pending to be serviced when the GIE bit is set again.

For additional information on Timer1, Timer2, comparators, ADC, Enhanced CCP modules, refer to the respective peripheral section.

19.4.1 RA2/INT INTERRUPT

The external interrupt on the RA2/INT pin is edge-triggered; either on the rising edge if the INTEDG bit of the OPTION register is set, or the falling edge, if the INTEDG bit is clear. When a valid edge appears on the RA2/INT pin, the INTF bit of the INTCON register is set. This interrupt can be disabled by clearing the INTE control bit of the INTCON register. The INTF bit must be cleared by software in the Interrupt Service Routine before re-enabling this interrupt. The RA2/INT interrupt can wake-up the processor from Sleep, if the INTE bit was set prior to going into Sleep. See **Section 19.7 "Power-Down Mode (Sleep)"** for details on Sleep and Figure 19-10 for timing of wake-up from Sleep through RA2/INT interrupt.

Note:	The ANSEL register must be initialized to configure an analog channel as a digital input. Pins configured as analog inputs will read '0' and cannot generate an interrupt.

21.2 MPLAB XC Compilers

The MPLAB XC Compilers are complete ANSI C compilers for all of Microchip's 8, 16, and 32-bit MCU and DSC devices. These compilers provide powerful integration capabilities, superior code optimization and ease of use. MPLAB XC Compilers run on Windows, Linux or MAC OS X.

For easy source level debugging, the compilers provide debug information that is optimized to the MPLAB X IDE.

The free MPLAB XC Compiler editions support all devices and commands, with no time or memory restrictions, and offer sufficient code optimization for most applications.

MPLAB XC Compilers include an assembler, linker and utilities. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. MPLAB XC Compiler uses the assembler to produce its object file. Notable features of the assembler include:

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

21.3 MPASM Assembler

The MPASM Assembler is a full-featured, universal macro assembler for PIC10/12/16/18 MCUs.

The MPASM Assembler generates relocatable object files for the MPLINK Object Linker, Intel® standard HEX files, MAP files to detail memory usage and symbol reference, absolute LST files that contain source lines and generated machine code, and COFF files for debugging.

The MPASM Assembler features include:

- Integration into MPLAB X IDE projects
- User-defined macros to streamline assembly code
- Conditional assembly for multipurpose source files
- Directives that allow complete control over the assembly process

21.4 MPLINK Object Linker/ MPLIB Object Librarian

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

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

The object linker/library features include:

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

21.5 MPLAB Assembler, Linker and Librarian for Various Device Families

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

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

TABLE 22-8: OSCILLATOR PARAMETERS

Standard Operating Conditions (unless otherwise stated)								
Param No.	**Sym.**	**Characteristic**	**Freq. Tolerance**	**Min.**	**Typ†**	**Max.**	**Units**	**Conditions**
OS06	TWARM	Internal Oscillator Switch when running	—	—	—	2	TOSC	
OS07	INTOSC	Internal Calibrated INTOSC Frequency(1) (4 MHz)	±1%	3.96	4.0	4.04	MHz	VDD = 3.5V, TA = 25°C
			±2%	3.92	4.0	4.08	MHz	2.5V ≤ VDD ≤ 5.5V, 0°C ≤ TA ≤ +85°C
			±5%	3.80	4.0	4.20	MHz	2.0V ≤ VDD ≤ 5.5V, -40°C ≤ TA ≤ +85°C (Ind.), -40°C ≤ TA ≤ +125°C (Ext.)
OS08	HFOSC	Internal Calibrated HFINTOSC Frequency(1)	±1%	7.92	8	8.08	MHz	VDD = 3.5V, TA = 25°C
			±2%	7.84	8	8.16	MHz	2.5V ≤ VDD ≤ 5.5V, 0°C ≤ TA ≤ +85°C
			±5%	7.60	8	8.40	MHz	2.0V ≤ VDD ≤ 5.5V, -40°C ≤ TA ≤ +85°C (Ind.), -40°C ≤ TA ≤ +125°C (Ext.)
OS09	LFOSC	Internal LFINTOSC Frequency	—	—	31	—	kHz	
OS10*	TIOSC ST	HFINTOSC Wake-up from Sleep Start-up Time	—	—	12	24	µs	VDD = 2.0V -40°C ≤ TA ≤ +85°C
				—	7	14	µs	VDD = 3.0V -40°C ≤ TA ≤ +85°C
				—	6	11	µs	VDD = 5.0V -40°C ≤ TA ≤ +85°C

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: To ensure these oscillator frequency tolerances, VDD and VSS must be capacitively decoupled as close to the device as possible. 0.1 µF and 0.01 µF values in parallel are recommended.

FIGURE 22-5: CLKOUT AND I/O TIMING

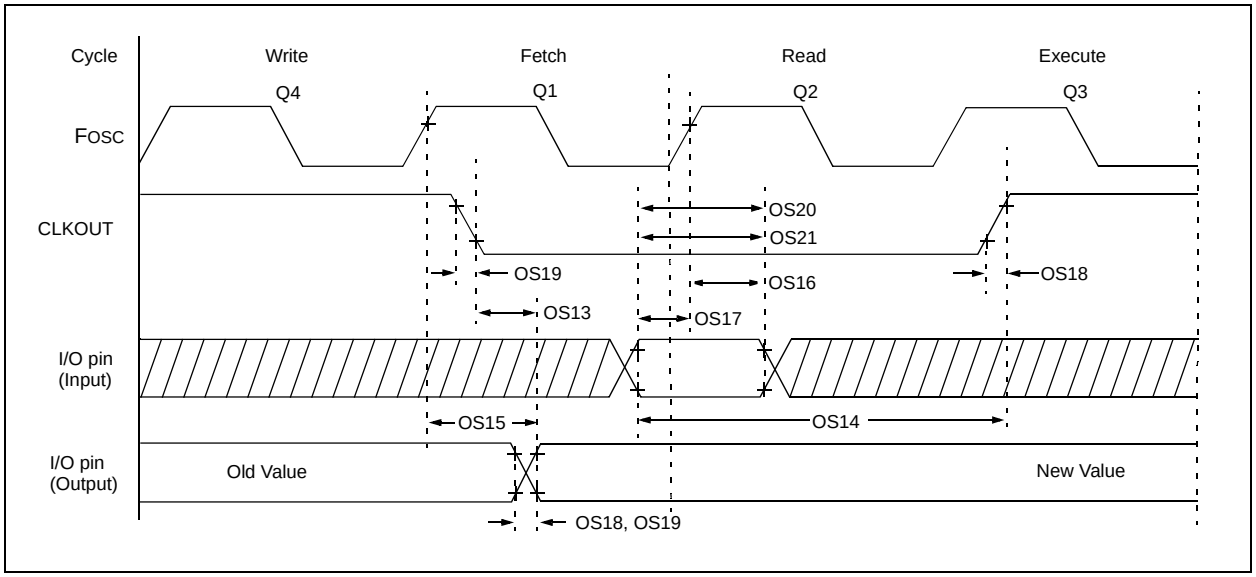


TABLE 22-9: CLKOUT AND I/O TIMING PARAMETERS

Standard Operating Conditions (unless otherwise stated)							
Param No.	**Sym.**	**Characteristic**	**Min.**	**Typ †**	**Max.**	**Units**	**Conditions**
OS13	TCKL2IOV	CLKOUT↓ to Port out valid(1)	—	—	20	ns	
OS14	TIOV2CKH	Port input valid before CLKOUT↑(1)	TOSC + 200 ns	—	—	ns	
OS15	TOSH2IOV	FOSC↑ (Q1 cycle) to Port out valid	—	50	70*	ns	VDD = 5.0V
OS16	TOSH2IOI	FOSC↑ (Q2 cycle) to Port input invalid (I/O in setup time)	50	—	—	ns	VDD = 5.0V
OS17	TIOV2OSH	Port input valid to FOSC↑ (Q2 cycle) (I/O in setup time)	20	—	—	ns	
OS18	TIOR	Port output rise time	— —	40 15	72 32	ns ns	VDD = 2.0V VDD = 5.0V
OS19	TIOF	Port output fall time	— —	28 15	55 30	ns ns	VDD = 2.0V VDD = 5.0V
OS20*	TINP	INT pin input high or low time	25	—	—	ns	
OS21*	TIOC	Interrupt-on-change new input level time	TCY	—	—	ns	

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated.

Note 1: Measurements are taken in RC mode where CLKOUT output is 4 x TOSC.

FIGURE 23-7: IDD TYPICAL, EC OSCILLATOR, HIGH-POWER MODE, PIC16F753 ONLY

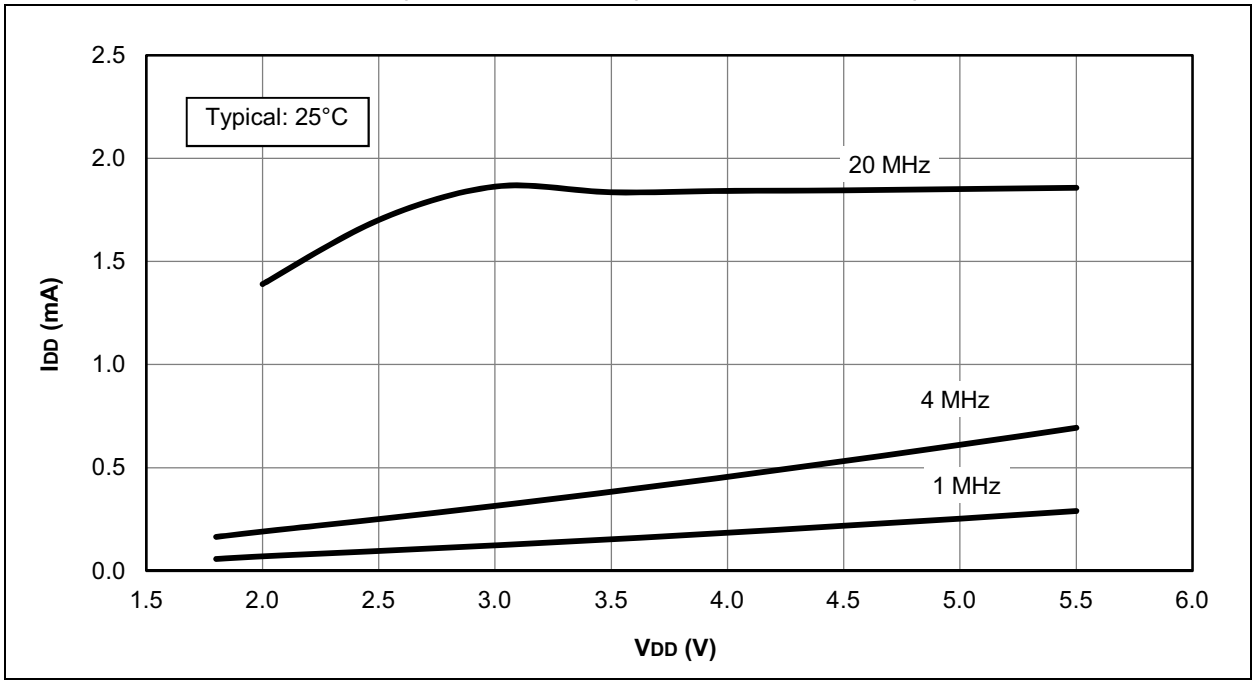


FIGURE 23-8: IDD MAXIMUM, EC OSCILLATOR, HIGH-POWER MODE, PIC16F753 ONLY