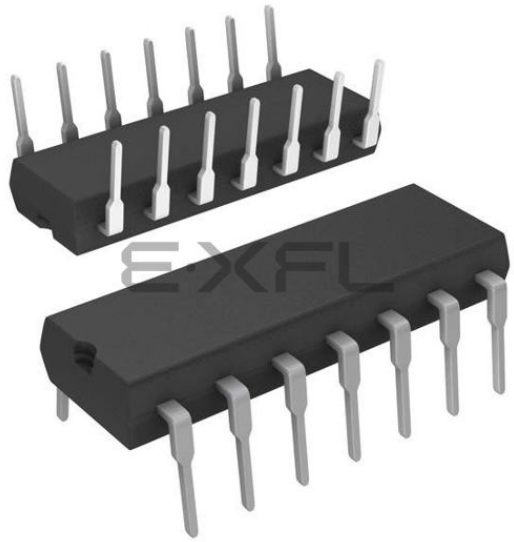


Welcome to E-XFL.COM

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 ~ 5.5V
Data Converters	A/D 8x10b; D/A 1x9b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Through Hole
Package / Case	14-DIP (0.300", 7.62mm)
Supplier Device Package	14-PDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f753-e-p

180h	INDF	INDF<7:0>								xxxx xxxx	uuuu uuuu
181h	OPTION_REG	$\overline{\text{RAPU}}$	INTEDG	T0CS	T0SE	PSA	PS<2:0>			1111 1111	1111 1111
182h	PCL	PCL<7:0>								0000 0000	0000 0000
183h	STATUS	IRP	RP1	RP0	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C	0001 1xxx	000q quuu
184h	FSR	FSR<7:0>								xxxx xxxx	uuuu uuuu
185h	ANSELA	—	—	—	ANSA4	—	ANSA2	ANSA1	ANSA0	---1 -111	---1 -111
186h	—	Unimplemented								—	—
187h	ANSELC	—	—	—	—	ANSC3	ANSC2	ANSC1	ANSC0	---- 0000	---- 0000
188h	APFCON	—	—	—	T1GSEL	—	—	—	—	---0 ----	---0 ----
189h	OSCTUNE	—	—	—	TUN<4:0>					---0 0000	---0 0000
18Ah	PCLATH	—	—	—	PCLATH<4:0>					---0 0000	---0 0000
18Bh	INTCON	GIE	PEIE	T0IE	INTE	IOCIE	T0IF	INTF	IOCIF	0000 0000	0000 0000
18Ch	PMCON1	—	—	—	—	—	WREN	WR	RD	---- -000	---- -000
18Dh	PMCON2	Program Memory Control Register 2								---- ----	---- ----
18Eh	PMADRL	PMADRL<7:0>								0000 0000	0000 0000
18Fh	PMADRH	—	—	—	—	—	—	PMADRH<1:0>		---- --00	---- --00
190h	PMDATL	PMDATL<7:0>								0000 0000	0000 0000
191h	PMDATH	—	—	PMDATH<5:0>						--00 0000	--00 0000
192h	COG1PHR	—	—	—	—	G1PHR<3:0>				---- xxxx	---- uuuu
193h	COG1PHF	—	—	—	—	G1PHF<3:0>				---- xxxx	---- uuuu
194h	COG1BKR	—	—	—	—	G1BKR<3:0>				---- xxxx	---- uuuu
195h	COG1BKF	—	—	—	—	G1BKF<3:0>				---- xxxx	---- uuuu
196h	COG1DBR	—	—	—	—	G1DBR<3:0>				---- xxxx	---- uuuu
197h	COG1DBF	—	—	—	—	G1DBF<3:0>				---- xxxx	---- uuuu
198h	COG1CON0	G1EN	G1OE1	G1OE0	G1POL1	G1POL0	G1LD	—	G1MD	0000 00-0	0000 00-0
199h	COG1CON1	G1RDBTS	G1FDBTS	—	—	—	—	G1CS<1:0>		00-- --00	00-- --00
19Ah	COG1RIS	—	G1RIHLT2	G1RIHLT1	G1RIT2M	G1RIFLT	G1RICCP1	G1RIC2	G1RIC1	0000 0000	0000 0000
19Bh	COG1RSIM	—	G1RMHLT2	G1RMHLT1	G1RMT2M	G1RMFLT	G1RMCCP1	G1RMC2	G1RMC1	0000 0000	0000 0000
19Ch	COG1FIS	—	G1FIHLT2	G1FIHLT1	G1FIT2M	G1FIFLT	G1FICCP1	G1FIC2	G1FIC1	0000 0000	0000 0000
19Dh	COG1FSIM	—	G1FMHLT2	G1FMHLT1	G1FMT2M	G1FMFLT	G1FMCCP1	G1FMC2	G1FMC1	0000 0000	0000 0000
19Eh	COG1ASD0	C1ASDE	C1ARSEN	G1ASD1L<1:0>		G1ASD0L<1:0>		—	—	0000 00--	0000 00--
19Fh	COG1ASD1	—	—	—	G1ASDSHLT2	G1ASDSHLT1	G1ASDSC2	G1ASDSC1	G1ASDSFLT	0000 0000	0000 0000

— = Unimplemented locations read as '0', u = unchanged, x = unknown, q = value depends on condition shaded = unimplemented

To read a program memory location, the user must write two bytes of the address to the PMADRL and PMADRH registers, and then set control bit RD (PMCON1<0>). Once the read control bit is set, the program memory Flash controller will use the second instruction cycle after to read the data. This causes the second instruction immediately following the "BSF PMCON1,RD" instruction to be ignored. The data is available in the very next cycle in the PMDATL and PMDATH registers; it can be read as two bytes in the following instructions. PMDATL and PMDATH registers will hold this value until another read or until it is written to by the user (during a write operation).

```
BANKSEL  PM_ADR              ; Change STATUS bits RP1:0 to select bank with PMADRL
MOVLW    MS_PROG_PM_ADDR     ;
MOVWF    PMADRH              ; MS Byte of Program Address to read
MOVLW    LS_PROG_PM_ADDR     ;
MOVWF    PMADRL              ; LS Byte of Program Address to read
BANKSEL  PMCON1              ; Bank to containing PMCON1
BSF      PMCON1, RD          ; PM Read

NOP                          ; First instruction after BSF PMCON1,RD executes normally

NOP                          ; Any instructions here are ignored as program
                             ; memory is read in second cycle after BSF PMCON1,RD
                             ;
BANKSEL  PMDATL              ; Bank to containing PMADRL
MOVF     PMDATL, W           ; W = LS Byte of Program PMDATL
MOVF     PMDATH, W           ; W = MS Byte of Program PMDATL
```

The CLKOUT pin is available for general purpose I/O or system clock output. The $\overline{\text{CLKOUTEN}}$ bit of the Configuration Word controls the function of the CLKOUT pin.

When the $\overline{\text{CLKOUTEN}}$ bit is cleared, the CLKOUT pin is driven by the selected internal oscillator frequency divided by 4. The corresponding I/O pin always reads '0' in this configuration.

The CLKOUT signal may be used to provide a clock for external circuitry, synchronization, calibration, test or other application requirements.

When the $\overline{\text{CLKOUTEN}}$ bit is set, the system clock out function is disabled and the CLKOUT pin is available for general purpose I/O.

In applications where the OSCTUNE register is used to shift the HFINTOSC frequency, the application should not expect the frequency to stabilize immediately. In this case, the frequency may shift gradually toward the new value. The time for this frequency shift is less than eight cycles of the base frequency.

A short delay is invoked upon power-up and when waking from sleep to allow the memory bias circuitry to stabilize. Table 4-1 shows examples where the oscillator delay is invoked.

Sleep/POR	INTOSC	31 kHz to 8 MHz	10 µs internal delay to allow memory bias to stabilize.
Sleep/POR	EC	DC – 20 MHz	

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

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 : Read as ‘0’

bit 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.

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

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 : Read as ‘0’

bit 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.

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

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 : Read as ‘0’

bit 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.

U-0	U-0	R/W-x/u	R/W-x/u	R-x/x	R/W-x/u	R/W-x/u	R/W-x/u
—	—	RC5	RC4	RC3	RC2	RC1	RC0
bit 7							bit 0

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 Read as ‘0’

bit 5-0 : PORTC I/O Value bits

1 = Port pin is $\geq V_{IH}$

0 = Port pin is $\leq V_{IL}$

Writes to PORTC are actually written to corresponding LATC register. Reads from PORTC register is return of actual I/O pin values.

U-0	U-0	R/W-1/1	R/W-1/1	R-1/1	R/W-1/1	R/W-1/1	R/W-1/1
—	—	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0
bit 7							bit 0

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 Read as ‘0’

bit 5-0 PORTC Tri-State Control bits

1 = PORTC pin configured as an input (tri-stated)

0 = PORTC pin configured as an output

U-0	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
—	—	LATC5	LATC4	LATC3	LATC2	LATC1	LATC0
bit 7							bit 0

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 Read as ‘0’

bit 5-0 : PORTC Output Latch Value bits

Writes to PORTC are actually written to corresponding LATC register. Reads from PORTC register is return of actual I/O pin values.

Timer1 has four prescaler options allowing one, two, four or eight divisions of the clock input. The T1CKPS bits of the T1CON register control the prescale counter. The prescale counter is not directly readable or writable; however, the prescaler counter is cleared upon a write to TMR1H or TMR1L.

If control bit $\overline{\text{T1SYNC}}$ of the T1CON register is set, the external clock input is not synchronized. The timer increments asynchronously to the internal phase clocks. If external clock source is selected then the timer will continue to run during Sleep and can generate an interrupt on overflow, which will wake-up the processor. However, special precautions in software are needed to read/write the timer (see).

	When switching from synchronous to asynchronous operation, it is possible to skip an increment. When switching from asynchronous to synchronous operation, it is possible to produce an additional increment.

7.4.1 READING AND WRITING TIMER1 IN ASYNCHRONOUS COUNTER MODE

Reading TMR1H or TMR1L while the timer is running from an external asynchronous clock will ensure a valid read (taken care of in hardware). However, the user should keep in mind that reading the 16-bit timer in two 8-bit values itself, poses certain problems, since the timer may overflow between the reads.

For writes, it is recommended that the user simply stop the timer and write the desired values. A write contention may occur by writing to the timer registers, while the register is incrementing. This may produce an unpredictable value in the TMR1H:TMR1L register pair.

Timer1 can be configured to count freely or the count can be enabled and disabled using Timer1 gate circuitry. This is also referred to as Timer1 gate count enable.

Timer1 gate can also be driven by multiple selectable sources.

7.5.1 TIMER1 GATE COUNT ENABLE

The Timer1 gate is enabled by setting the TMR1GE bit of the T1GCON register. The polarity of the Timer1 gate is configured using the T1GPOL bit of the T1GCON register.

When Timer1 Gate (T1G) input is active, Timer1 will increment on the rising edge of the Timer1 clock source. When Timer1 gate input is inactive, no incrementing will occur and Timer1 will hold the current count. See Figure 7-3 for timing details.

n	0	0	Counts
n	0	1	Holds Count
n	1	0	Holds Count
n	1	1	Counts

7.5.2 TIMER1 GATE SOURCE SELECTION

The Timer1 gate source can be selected from one of four different sources. Source selection is controlled by the T1GSS bits of the T1GCON register. The polarity for each available source is also selectable. Polarity selection is controlled by the T1GPOL bit of the T1GCON register.

11	SYNCC2OUT
10	SYNCC1OUT
01	Overflow of Timer0 (TMR0 increments from FFh to 00h)
00	Timer1 Gate Pin

The Timer2 module is an 8-bit timer with the following features:

- 8-bit timer register (TMR2)
- 8-bit period register (PR2)
- Interrupt on TMR2 match with PR2
- Software programmable prescaler (1:1, 1:4, 1:16, 1:64)
- Software programmable postscaler (1:1 to 1:16)

See Figure 8-1 for a block diagram of Timer2.

The clock input to the Timer2 module is the system instruction clock (FOSC/4). The clock is fed into the Timer2 prescaler, which has prescale options of 1:1, 1:4 or 1:16. The output of the prescaler is then used to increment the TMR2 register.

The values of TMR2 and PR2 are constantly compared to determine when they match. TMR2 will increment from 00h until it matches the value in PR2. When a match occurs, two things happen:

- TMR2 is reset to 00h on the next increment cycle.
- The Timer2 postscaler is incremented

The match output of the Timer2/PR2 comparator is then fed into the Timer2 postscaler. The postscaler has postscale options of 1:1 to 1:16 inclusive. The output of the Timer2 postscaler is used to set the TMR2IF interrupt flag bit in the PIR1 register.

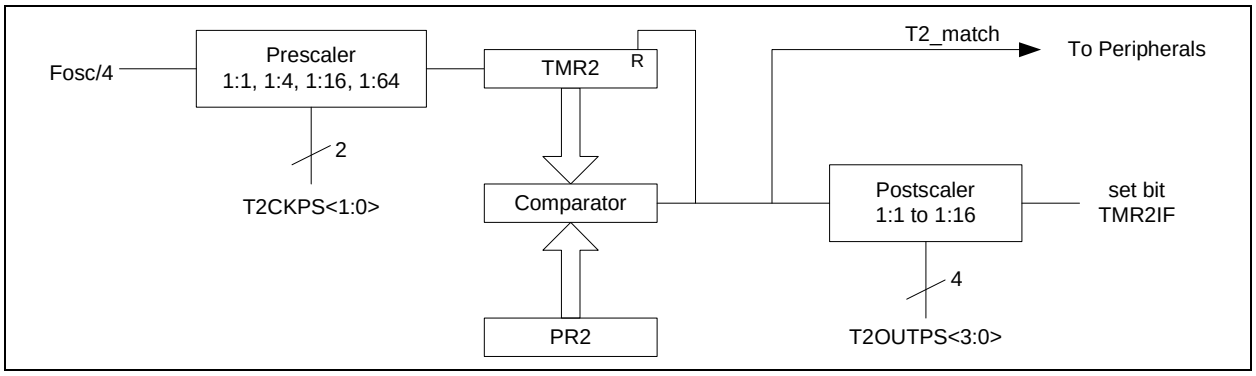
The TMR2 and PR2 registers are both fully readable and writable. On any Reset, the TMR2 register is set to 00h and the PR2 register is set to FFh.

Timer2 is turned on by setting the TMR2ON bit in the T2CON register to a '1'. Timer2 is turned off by clearing the TMR2ON bit to a '0'.

The Timer2 prescaler is controlled by the T2CKPS bits in the T2CON register. The Timer2 postscaler is controlled by the T2OUTPS bits in the T2CON register. The prescaler and postscaler counters are cleared when:

- A write to TMR2 occurs.
- A write to T2CON occurs.
- Any device Reset occurs (Power-on Reset, $\overline{\text{MCLR}}$ Reset, Watchdog Timer Reset, or Brown-out Reset).

	TMR2 is not cleared when T2CON is written.



10.2.5 COMPARE DURING SLEEP

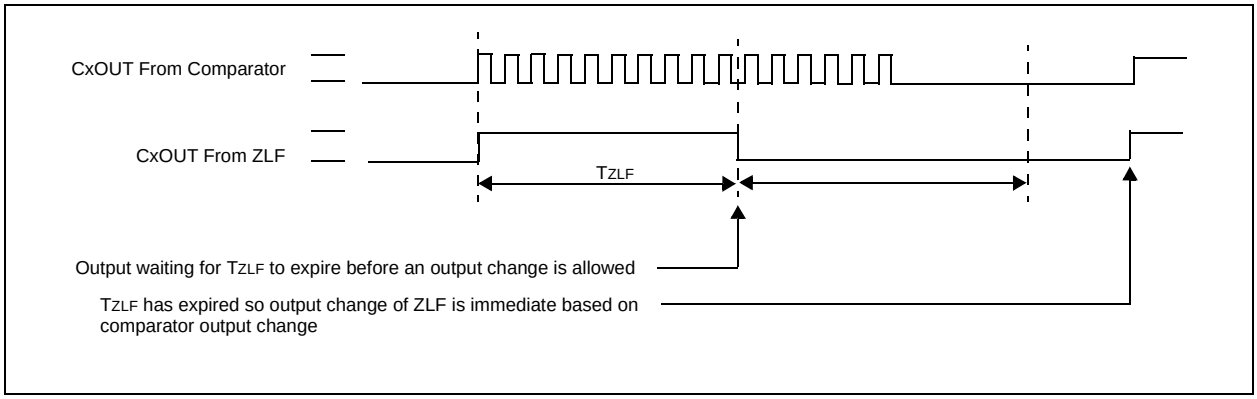
The Compare mode is dependent upon the system clock (FOSC) for proper operation. Since FOSC is shut down during Sleep mode, the Compare mode will not function properly during Sleep.

CCP1CON	—	—	DC1B<1:0>		CCP1M<3:0>				80
CCPR1L	CCPR1L<7:0>								74
CCPR1H	CCPR1H<7:0>								74
INTCON	GIE	PEIE	T0IE	INTE	IOCIE	T0IF	INTF	IOCIF	17
PIE1	TMR1-GIE	ADIE	—	—	HLTMR2IE	HLTMR1IE	TMR2IE	TMR1IE	18
PIE2	—	—	C2IE	C1IE	—	COG1IE	—	CCP1IE	19
PIR1	TMR1-GIF	ADIF	—	—	HLTMR2IF	HLTMR1IF	TMR2IF	TMR1IF	20
PIR2	—	—	C2IF	C1IF	—	COG1IF	—	CCP1IF	21
T1CON	TMR1CS<1:0>		T1CKPS<1:0>		T1OSCEN	$\overline{\text{T1SYNC}}$	—	TMR1ON	65
T1GCON	TMR1GE	T1GPOL	T1GTM	T1GSPM	T1GGO/$\overline{\text{DONE}}$	T1GVAL	T1GSS<1:0>		66
TMR1H	TMR1H<7:0>								57*
TMR1L	TMR1L<7:0>								57*
TRISA	—	—	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	43

— = Unimplemented location, read as ‘0’. Shaded cells are not used by Compare mode.

* Page provides register information.
TRISA3 always reads ‘1’.

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.



R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
CxINTP	CxINTN	CxPCH<1:0>			CxNCH<2:0>		
bit 7							bit 0

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 Comparator Interrupt on Positive Going Edge Enable bit
1 = The CxIF interrupt flag will be set upon a positive going edge of the CxOUT bit
0 = No interrupt flag will be set on a positive going edge of the CxOUT bit

bit 6 Comparator Interrupt on Negative Going Edge Enable bit
1 = The CxIF interrupt flag will be set upon a negative going edge of the CxOUT bit
0 = No interrupt flag will be set on a negative going edge of the CxOUT bit

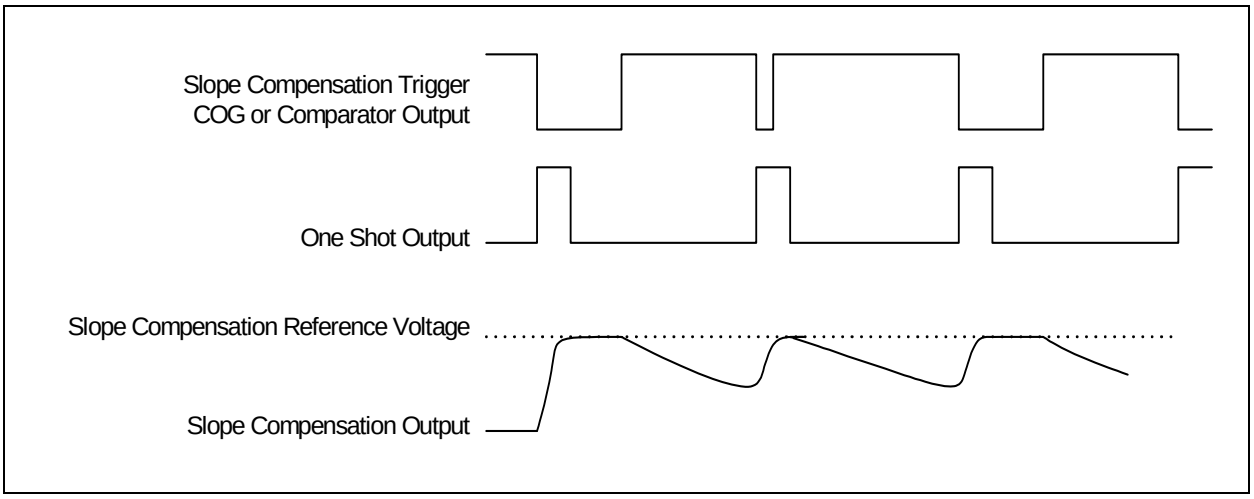
bit 5-3 Comparator Positive Input Channel Select bits
000 = CxVP connects to CxIN+ pin
001 = CxVP connects to dac_out
010 = CxVP connects to FVR
011 = CxVP connects to Slope Compensator Output
1xx = CxVP connects to AGND

bit 2-0 Comparator Negative Input Channel Select bits
000 = CxVN connects to CxIN0- pin
001 = CxVN connects to CxIN1- pin
010 = CxVN connects to CxIN2- pin
011 = CxVN connects to CxIN3- pin
1xx = CxVN connects to Slope Compensator Output

U-0	U-0	U-0	U-0	U-0	U-0	R-0/0	R-0/0
—	—	—	—	—	—	MCOUT2	MCOUT1
bit 7							bit 0

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

bit 7-2 Read as ‘0’
bit 1 Mirror Copy of C2OUT bit
bit 0 Mirror Copy of C1OUT bit



The slope compensator input reference voltage should be set to the target circuit peak current sense voltage. The slope compensator output voltage starts at the input reference voltage and should fall at a rate less than half the target circuit current sense voltage rate of rise. Therefore, the compensator slope expressed as volts per µs can be computed as shown in Equation 17-2.

$$\frac{V}{R} \text{ t } \frac{\frac{V_{REF}}{2}}{PWM\ Period\ (\ R}$$

For example, when the circuit is using a 1 : current sense resistor and the peak current is 1A, then the peak current expressed as a voltage (V_{REF}) is 1V. If your power supply is running at 1 MHz, then the period is 1 Rs. Therefore, the desired slope is:

$$\frac{\frac{V_{REF}}{2}}{PWM\ Period\ (\ R} = \frac{\frac{1}{2}}{1\ R} = 0.5V\ eR$$

: The setting for 0.5V/ Rs is SCxISET<3:0> = 6 and SCxRNG = 0.

R/W-0/0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	R/W-0/0
SCxEN	—	—	SCxPOL	SCxTSS<1:0>		—	SCxINS
bit 7							bit 0

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 configuration bits

bit 7 Slope Compensation Enable bit
1 = Slope compensation is enabled
0 = Slope compensation is disabled

bit 6-5 Read as ‘0’

bit 4 Slope Compensation Input Polarity bit
1 = Signal is inverted polarity (active-low)
0 = Signal is normal polarity (active-high)

bit 3-2 Slope Compensation Timing Select bits
11 = C2OUT_sync
10 = C1OUT_sync
01 = COG1_output1
00 = COG1_output0

bit 1 Read as ‘0’

bit 0 Slope Compensation Input Select bit
1 = FVR_buffer1 is selected
0 = SLPC1IN pin is selected

U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	SCxRNG	SCxISET<3:0>			
bit 7							bit 0

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 configuration bits

bit 7-5 Read as ‘0’

bit 4 Slope Compensator Range bit
1 = Range setting is SCxISET +1.0V/ Rs
0 = Range setting is SCxISET * 0.75/15 +0.2V/ Rs

bit 3-0 Slope Compensator Current Sink Set bits
xxxxx = SC module Slope Selection

Syntax: [*label*] SUBWF f,d

Operands: 0 df d127
d [0,1]

Operation: (f) - (W) o destination)

Status Affected: C, DC, Z

Description: Subtract (2's complement method) W register from register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f'.

C = 0	W ! f
C = 1	W df
DC = 0	W<3:0> ! f<3:0>
DC = 1	W<3:0> df<3:0>

Syntax: [*label*] SWAPF f,d

Operands: 0 df d127
d [0,1]

Operation: (f<3:0>) o (destination<7:4>),
(f<7:4>) o (destination<3:0>)

Status Affected: None

Description: The upper and lower nibbles of register 'f' are exchanged. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed in register 'f'.

Syntax: [*label*] XORLW k

Operands: 0 dk d255

Operation: (W) .XOR. k o W)

Status Affected: Z

Description: The contents of the W register are XOR'ed with the 8-bit literal 'k'. The result is placed in the W register.

Syntax: [*label*] XORWF f,d

Operands: 0 df d127
d [0,1]

Operation: (W) .XOR. (f) o destination)

Status Affected: Z

Description: Exclusive OR the contents of the W register with register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f'.

The PIC16F753/HV753 has a host of features intended to maximize system reliability, minimize cost through elimination of external components, provide power-saving features and offer code protection.

These features are:

- Reset
  - Power-on Reset (POR)
  - Power-up Timer (PWRT)
  - Brown-out Reset (BOR)
- Interrupts
- Watchdog Timer (WDT)
- Oscillator selection
- Sleep
- Code protection
- ID Locations
- In-Circuit Serial Programming™

The Power-up Timer (PWRT), which provides a fixed delay of 64 ms (nominal) on power-up only, is designed to keep the part in Reset while the power supply stabilizes. There is also circuitry to reset the device if a brown-out occurs, which can use the Power-up Timer to provide at least a 64 ms Reset. With these functions-on-chip, most applications need no external Reset circuitry.

The Sleep mode is designed to offer a very low-current Power-Down mode. The user can wake up from Sleep through:

- External Reset
- Watchdog Timer Wake-up
- An interrupt

Oscillator selection options are available to allow the part to fit the application. The INTOSC options save system cost, while the External Clock (EC) option provides a means for specific frequency and accurate clock sources. Configuration bits are used to select various options (see Register 19-1).

The Configuration bits can be programmed (read as '0'), or left unprogrammed (read as '1') to select various device configurations as shown in Register 19-1. These bits are mapped in program memory location 2007h.

Address 2007h is beyond the user program memory space. It belongs to the special configuration memory space (2000h-3FFFh), which can be accessed only during programming. See the *PIC16F753/HV753 Flash Memory Programming Specification* (DS41686) for more information.

The PIC® microcontrollers (MCU) and dsPIC® digital signal controllers (DSC) are supported with a full range of software and hardware development tools:

- Integrated Development Environment
  - MPLAB® X IDE Software
- Compilers/Assemblers/Linkers
  - MPLAB XC Compiler
  - MPASM™ Assembler
  - MPLINK™ Object Linker/ MPLIB™ Object Librarian
  - MPLAB Assembler/Linker/Librarian for Various Device Families
- Simulators
  - MPLAB X SIM Software Simulator
- Emulators
  - MPLAB REAL ICE™ In-Circuit Emulator
- In-Circuit Debuggers/Programmers
  - MPLAB ICD 3
  - PICkit™ 3
- Device Programmers
  - MPLAB PM3 Device Programmer
- Low-Cost Demonstration/Development Boards, Evaluation Kits and Starter Kits
- Third-party development tools

The MPLAB X IDE is a single, unified graphical user interface for Microchip and third-party software, and hardware development tool that runs on Windows®, Linux and Mac OS® X. Based on the NetBeans IDE, MPLAB X IDE is an entirely new IDE with a host of free software components and plug-ins for high-performance application development and debugging. Moving between tools and upgrading from software simulators to hardware debugging and programming tools is simple with the seamless user interface.

With complete project management, visual call graphs, a configurable watch window and a feature-rich editor

- Complementary Output Generator (COG):
  - Complementary Waveforms from selectable sources
  - Two I/O (50 mA) for direct MOSFET drive
  - Rising and/or Falling edge dead-band control
  - Phase control, Blanking control
  - Auto-shutdown
  - Slope Compensation Circuit for use with SMPS power supplies

TABLE 1: PIC16F753/HV753 FAMILY TYPES

Device	Data Sheet Index	Program Memory Flash (words)	Self-Read/Write Flash Memory	Data SRAM (bytes)	I/Os(2)	10-bit ADC (ch)	Comparators	Timers (8/16-bit)	CCP	Complementary Output Generator (COG)	DAC	Op Amp	Shunt Regulator	Debug(1)	XLP
PIC12F752	(1)	1K	Y	64	6	4	2	3/1	1	Y	5-bit	N	N	H	Y
PIC12HV752	(1)	1K	Y	64	6	4	2	3/1	1	Y	5-bit	N	Y	H	Y
PIC16F753	(2)	2K	Y	128	12	8	2	3/1	1	Y	9-bit	Y	N	I/H	Y
PIC16HV753	(2)	2K	Y	128	12	8	2	3/1	1	Y	9-bit	Y	Y	I/H	Y

Note 1: I - Debugging, Integrated on Chip; H - Debugging, Requires Debug Header.

2: One pin is input-only.

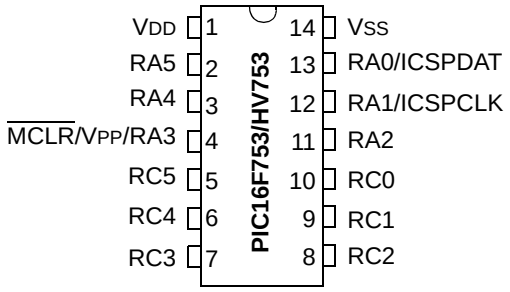
Data Sheet Index: (Unshaded devices are described in this document.)

1: DS40001576 PIC12F752/HV752 Data Sheet, 8-Pin Flash-Based, 8-Bit CMOS Microcontrollers.

2: DS40001709 PIC16F753/HV753 Data Sheet, 14/16-Pin Flash-based, 8-Bit CMOS Microcontrollers.

Note: For other small form-factor package availability and marking information, please visit **http://www.microchip.com/packaging** or contact your local sales office.

FIGURE 1: 14-PIN PDIP, SOIC, TSSOP DIAGRAM



Note: See Table 2 for location of all peripheral functions.