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	13
Program Memory Size	3.5KB (2K x 14)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	128 x 8
Voltage - Supply (Vcc/Vdd)	4V ~ 5.5V
Data Converters	A/D 8x8b; D/A 1x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SSOP (0.209", 5.30mm Width)
Supplier Device Package	20-SSOP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16c782t-i-ss

Peripheral Features (Continued):

- Dual Analog Comparator module with:
  - Individual enable and interrupt bits
  - Programmable speed and output polarity
  - Fully configurable inputs and outputs
  - Reference from DAC, or VREF1/VREF2
  - Low input offset voltage.
- VR voltage reference module:
  - 3.072V +/- 0.7% @25°C, AVDD = 5V
  - Configurable output to ADC reference, DAC reference, and VR pin
  - 5 mA sink/source
- Programmable Switch Mode Controller module:
  - PWM and PSM modes
  - Programmable switching frequency
  - Configurable for either single or dual feedback inputs
  - Configurable single or dual outputs
  - Slope compensation output available in single output mode

Key Features PIC® Mid-Range Reference Manual (DS33023)	PIC16C781	PIC16C782
Operating Frequency	DC - 20 MHz	DC - 20 MHZ
RESETS (and Delays)	POR, BOR, $\overline{\text{MCLR}}$, WDT (PWRT, OST)	POR, BOR, $\overline{\text{MCLR}}$, WDT (PWRT, OST)
Program Memory (14 bit words)	1K	2K
Data Memory (bytes)	128	128
Interrupts	8	8
I/O Ports	13 + 3 Input only	13 + 3 Input only
Timers	2	2
Programmable Switch Mode Controller	1	1
8-bit Analog-to-Digital Module	1	1
ADC channels	8 External, 2 Internal	8 External, 2 Internal
8-bit Digital-to-Analog Module	1	1
Comparators	2	2
Comparator Channels	4 (AN<7:4>)	4 (AN<7:4>)
Operational Amplifier	1	1
Voltage Reference	1	1
Brown-out Reset	Yes	Yes
Programmable Low Voltage Detect	Yes	Yes
Instruction Set	35 Instructions	35 Instructions

FIGURE 1-3: ANALOG SIGNAL MULTIPLEXING DIAGRAM

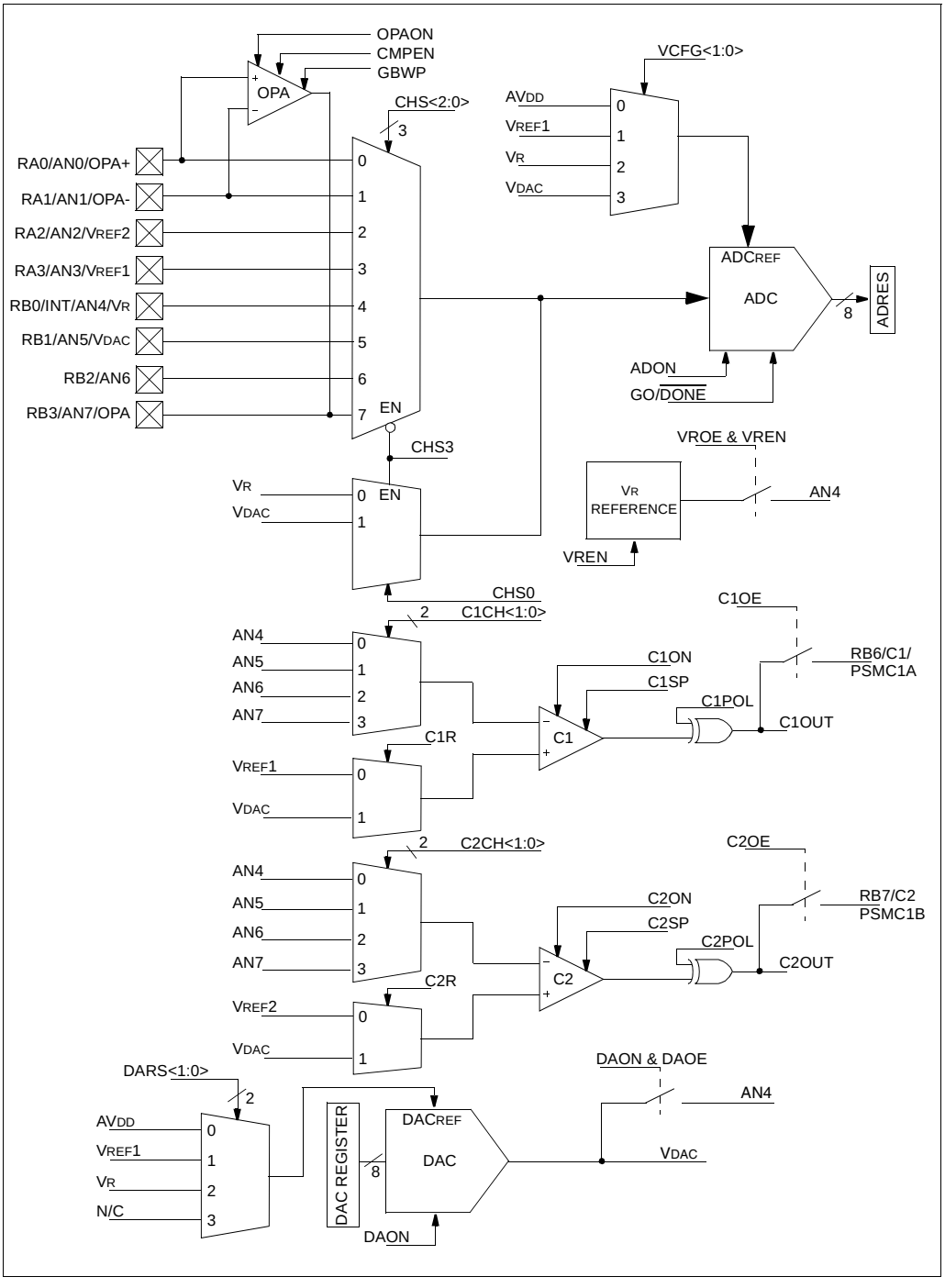


TABLE 1-1: PIC16C781/782 PINOUT DESCRIPTION (CONTINUED)

Name	Function	Input Type	Output Type	Description
RB7/C2/PSMC1B/$\overline{T1G}$	RB7	TTL	CMOS	Bi-directional I/O
	C2	—	CMOS	Comparator 2 Output
	PSMC1B	—	CMOS	PSMC Output 1B
	$\overline{T1G}$	ST	—	Timer 1 Gate Input
AVDD	AVDD	Power	—	Positive Supply for Analog
AVSS	AVSS	Power	—	Ground Reference for Analog
VDD	VDD	Power	—	Positive Supply for Logic and I/O pins
VSS	VSS	Power	—	Ground Reference for Logic and I/O pins

Legend: ST = Schmitt Trigger AN = Analog OD = open drain TTL = Logic Level
XTAL = Crystal CMOS = CMOS Output Power = Power Supply

REGISTER 3-2: WEAK PULL-UP PORTB REGISTER (WPUB: 95h)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
WPUB7	WPUB6	WPUB5	WPUB4	WPUB3	WPUB2	WPUB1	WPUB0
bit7							bit0

bit 7-0 **WPUB<7:0>:** PORTB Weak Pull-Up Control bits

1 = Weak pull-up enabled for corresponding pin

0 = Weak pull-up disabled for corresponding pin

Note 1: For the WPUB register setting to take effect, the $\overline{\text{RBPU}}$ bit in the OPTION_REG register must be cleared.

2: The weak pull-up device is automatically disabled if the pin is in output mode, i.e., (TRISB = 0) for corresponding pin.

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

REGISTER 3-3: INTERRUPT-ON-CHANGE PORTB REGISTER (IOCB: 96h)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0
IOCB7	IOCB6	IOCB5	IOCB4	IOCB3	IOCB2	IOCB1	IOCB0
bit7							bit0

bit 7-0 **IOCB<7:0>:** Interrupt-on-Change PORTB Control bits

1 = Interrupt-on-change enabled for corresponding pin

0 = Interrupt-on-change disabled for corresponding pin

Note 1: The interrupt enable bits, GIE and RBIE in the INTCON register, must be set for individual interrupts to be recognized.

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

3.3.3 TRISB, ANSEL, AND CONTROL PRECEDENCE

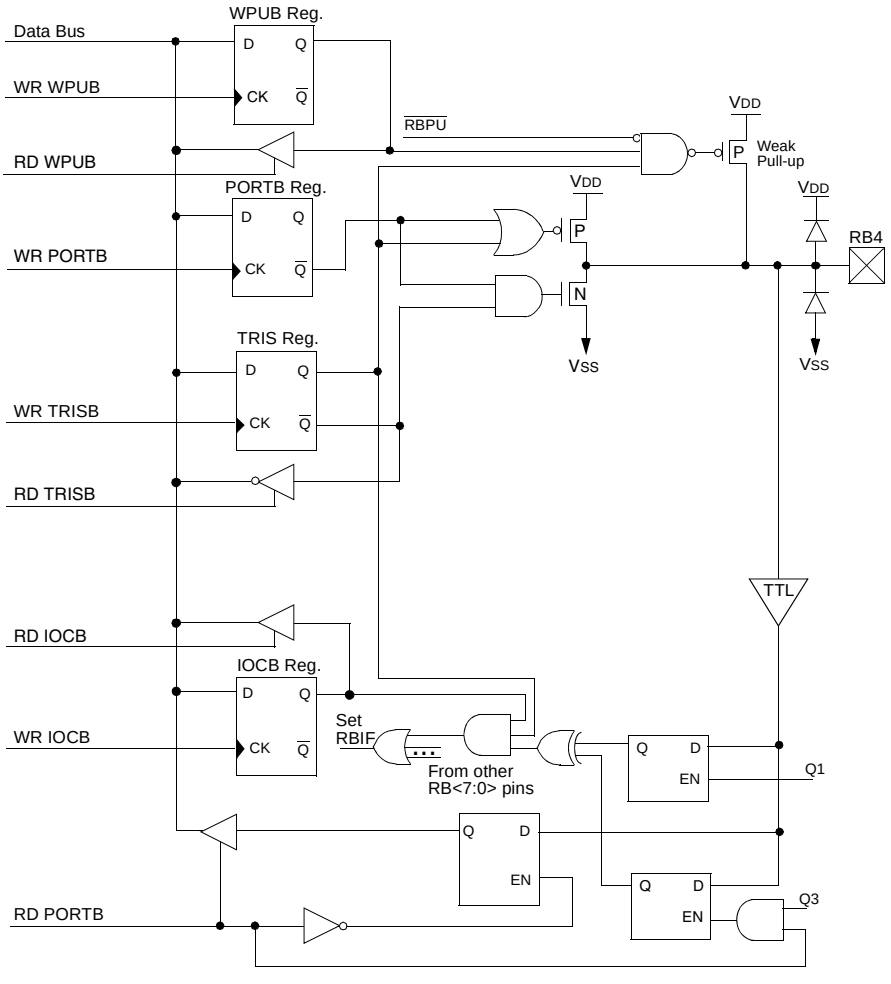
The ANSEL and TRISB registers are the primary controls for the configuration of PORTB pins. TRISB tri-states the output drivers of PORTB, and the ANSEL register disables the input buffers. It is important to program both registers when configuring a port pin, since most peripherals do not have precedence over the TRISB and ANSEL registers' control of the pin. Even if a peripheral has the ability to override the control of the TRISB and ANSEL registers, it is good practice to program both registers appropriately.

Note 1: Upon RESET, the ANSEL register configures the RB<3:0> pins as analog inputs.

2: When programmed as analog inputs, RB<3:0> pins will read as '0'.

3: There are specific cases in which the functions of the TRISB and ANSEL registers can be overridden by a peripheral or configuration word (see Figure 3-9 through Figure 3-16 for details).

FIGURE 3-13: BLOCK DIAGRAM OF RB4 PIN



TRISB<4>	FUNCTION	PORTB<4> READ
0	Digital Out	Pin
1	Digital In	Pin

REGISTER 4-3: PROGRAM MEMORY DATA LOW (PMDATL: 10Ch)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PMD7	PMD6	PMD5	PMD4	PMD3	PMD2	PMD1	PMD0
bit7							bit0

bit 7-0 **PMD<7:0>:** Program Memory Data bits
The value of the program memory word pointed to by PMADRH and PMADRL after a program memory read command.

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

REGISTER 4-4: PROGRAM MEMORY ADDRESS HIGH (PMADRH: 10Fh)

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	Reserved	Reserved	PMA10	PMA9	PMA8
bit7							bit0

bit 7-5 **Unimplemented:** Read as '0'

bit 4-3 **Reserved:** Read state is not guaranteed

bit 2-0 **PMA<10:8>**: PMR Address bits

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

REGISTER 4-5: PROGRAM MEMORY ADDRESS LOW (PMADRL: 10Dh)

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
PMA7	PMA6	PMA5	PMA4	PMA3	PMA2	PMA1	PMA0
bit7							bit0

bit 7-0 **PMA<7:0>:** PMR Address bits

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

6.3 Timer1 Oscillator for the PIC16C781/782

When the microcontroller is using INTRC w/o CLKOUT, Timer1 can enable and use the LP oscillator as the Timer1 oscillator. When enabled, Timer1 oscillator operation is solely controlled by the T1OSCEN bit. The oscillator will operate independently of the TMR1ON bit, allowing the programmer to start and stop the Timer/Counter using the TMR1ON bit. The oscillator will also operate during SLEEP, allowing continuous timekeeping with Timer1. The electrical requirements for the LP oscillator, when used as the Timer1 oscillator, are the same as when the oscillator is used in LP mode.

Note:	The oscillator requires a startup and stabilization time before use. Therefore, T1OSCEN should be set, and a suitable delay observed, prior to enabling Timer1 (see Section 14.2).

6.4 Timer1 Interrupt

The TMR1 register pair (TMR1H and TMR1L) increments from 0000h to FFFFh and then rolls over to 0000h. When Timer1 rolls over, the TMR1IF bit (PIR1<0>) is set. To enable an interrupt, the TMR1IE bit (PIE1<0>), the GIE (INTCON<7>) and the PEIE bit (INTCON<6>) must be set prior to rollover. To clear the interrupt, the TMR1IF must be cleared by software prior to re-enabling interrupts.

Note:	When enabling the Timer1 interrupt, the user should clear both TMR1 registers and the TMR1IF prior to enabling interrupts.

6.5 Effects of RESET

Only POR and BOR Resets clear T1CON, disabling Timer1. All other RESETS do not affect Timer1.

TABLE 6-1: SUMMARY OF REGISTERS ASSOCIATED WITH TIMER1

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other RESETS
0Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000X	0000 000u
0Ch	PIR1	LVDIF	ADIF	C2IF	C1IF	—	—	—	TMRIF	0000 ---0	0000 ---0
8Ch	PIE1	LVDIE	ADIE	C2IE	C1IE	—	—	—	TMRIE	0000 ---0	0000 ---0
0Eh	TMR1L	Least Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	uuuu uuuu
0Fh	TMR1H	Most Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	uuuu uuuu
10h	T1CON	—	TMR1GE	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	TMR1CS	TMR1ON	-000 0000	-uuu uuuu

Legend: x = unknown, u = unchanged, - = unimplemented, read as '0'. Shaded cells are not used by Timer1.

NOTES:

9.2 Configuring the ADC Module

9.2.1 CONFIGURING ANALOG PORT PINS

The ANSEL and TRISB registers control the operation of the ADC port pins. The port pins to be used as analog inputs must have their corresponding TRISB bits set (= 1). The proper ANSEL bits must also be set (analog input) to disable the digital input buffer.

Note 1: The ADC operation is independent of the state of the TRISB or ANSEL bits. These bits must be configured by the firmware prior to initiation of an ADC conversion.

2: When reading the PORTA or PORTB registers, all pins configured as analog input channels will read as a '0'.

3: Analog levels on any pin that is defined as a digital input, including AN<7:0>, may cause the input buffer to consume excess supply current.

9.2.2 CONFIGURING THE REFERENCE VOLTAGES

The VCFG<5:4> bits in the ADCON1 register configure the ADC module reference voltage input, ADCREF. The reference input can come from any of the following:

- Internal voltage reference (VR)
- External comparator C1 reference (VREF1)
- DAC output (VDAC)
- Analog positive supply (AVDD)

If an external reference is chosen for the ADCREF input, the port pin that multiplexes with the incoming external reference must also be configured as an analog input.

9.2.3 SELECTING THE ADC CONVERSION CLOCK

The ADC conversion cycle requires 9.5TAD. The source of the ADC conversion clock is software selectable. The four possible options for ADC clock are:

- FOSC/2
- FOSC/8
- FOSC/32
- ADRC (clock derived from a dedicated internal RC oscillator)

For correct ADC conversion, the ADC conversion clock (TAD) must be selected to ensure a minimum TAD time of 1.6 μsec. Table 9-1 shows the resultant TAD times derived from the device operating frequencies and the ADC clock source selected.

TABLE 9-1: TAD vs. DEVICE OPERATING FREQUENCIES: PIC16C781/782

ADC Clock Source (TAD)		Device Frequency			
Operation	**ADCS1:ADCS0**	**20 MHz**	**5 MHz**	**1.25 MHz**	**333.33kHz**
2 TOSC	00	100 ns(2)	400 ns(2)	1.6 s	6 μs
8 TOSC	01	400 ns	1.6 μs	6.4 μs	24 μs(3)
32 TOSC	10	1.6 μs	6.4 μs	25.6 μs(3)	96 μs(3)
RC	11	2 - 6 μs(1,4)	2 - 6 μs(1,4)	2 - 6 μs(1,4)	2 - 6 μs(1)

Legend: Shaded cells are outside of recommended range.

Note 1: The RC source has a typical TAD time of 4 μs.

2: These values violate the minimum required TAD time.

3: For faster conversion times, the selection of another clock source is recommended.

4: When device frequency is greater than 1 MHz, the RC ADC conversion clock source is recommended for SLEEP operation only.

12.0 COMPARATOR MODULE

The comparator module has two separate voltage comparators: Comparator C1 and Comparator C2 (see Figure 12-1).

Each comparator offers the following list of features:

- Control and configuration register
- Comparator output available externally
- Programmable output polarity
- Interrupt-on-change flags
- Wake-up from SLEEP
- Configurable as feedback input to the PSMC
- Programmable four input multiplexer
- Programmable reference selections
- Programmable speed
- Output synchronization to Timer1 clock input (Comparator C2 only)

12.1 Control Registers

Both comparators have separate control and configuration registers: CM1CON0 for C1 and CM2CON0 for C2. In addition, Comparator C2 has a second control register, CM2CON1, for synchronization control and simultaneous reading of both comparator outputs.

12.1.1 COMPARATOR C1 CONTROL REGISTER

The CM1CON0 register (shown in Register 12-1) contains the control and status bits for the following:

- Comparator enable
- Comparator input selection
- Comparator reference selection
- Output mode
- Comparator speed

Setting C1ON (CM1CON0<7>) enables Comparator C1 for operation.

Bits C1CH<1:0> (CM1CON0<1:0>) select the comparator input from the four analog pins AN<7:4>.

Note:	To use AN<7:4> as analog inputs, the appropriate bits must be programmed in the ANSEL register.

Setting C1R (CM1CON0<2>) selects the output of the DAC module as the reference voltage for the comparator. Clearing C1R selects the VREF1 input on the RA3/AN3/VREF1 pin.

The output of the comparator is available internally via the C1OUT flag (CM1CON0<6>). To make the output available for an external connection, the C1OE flag (CM1CON0<5>) must be set. If the module is disabled with C1OE set, the output will be driven as shown in Table 12-2:

The polarity of the comparator output can be inverted by setting the C1POL flag (CM1CON0<4>). Clearing C1POL results in a non-inverted output. A complete table showing the output state versus input conditions and the polarity bit is shown in Table 12-2.

TABLE 12-1: OUTPUT STATE VERSUS INPUT CONDITIONS

Input Condition	C1POL	C1OUT
C1VN > C1VP	0	0
C1VN < C1VP	0	1
C1VN > C1VP	1	1
C1VN < C1VP	1	0

Note 1:	The internal output of the comparator is latched at the end of each instruction cycle. External outputs are not latched.
2:	The C1 interrupt will operate correctly with C1OE set or cleared.
3:	For the output of C1 on RB6/C1/PSMC1A, the PSMC must be disabled and TRISB<6> must be '0'.

C1SP (CM1CON0<3>) configures the speed of the comparator. When C1SP is set, the comparator operates at its normal speed. Clearing C1SP operates the comparator in a slower, low power mode.

REGISTER 12-2: COMPARATOR C2 CONTROL REGISTER0 (CM2CON0: 11Ah)

R/W-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
C2ON	C2OUT	C2OE	C2POL	C2SP	C2R	C2CH1	C2CH0
bit 7							bit 0

bit 7 **C2ON:** Comparator C2 Enable bit
1 = C2 Comparator is enabled
0 = C2 Comparator is disabled

bit 6 **C2OUT:** Comparator C2 Output bit
If C2POL = 1 (inverted polarity):
C2OUT = 1, C2VP < C2VN
C2OUT = 0, C2VP > C2VN
If C2POL = 0 (non-inverted polarity):
C2OUT = 1, C2VP > C2VN
C2OUT = 0, C2VP < C2VN

bit 5 **C2OE:** Comparator C2 Output Enable bit
1 = C2OUT is present on RB7/C2/PSMC1B/$\overline{T1G}$(1)
0 = C2OUT is internal only

bit 4 **C2POL:** Comparator C2 Output Polarity Select bit
1 = C2OUT logic is inverted
0 = C2OUT logic is not inverted

bit 3 **C2SP:** Comparator C2 Speed Select bit
1 = C2 operates in normal speed mode
0 = C2 operates in low power, slow speed mode.

bit 2 **C2R:** Comparator C2 Reference Select bits (non-inverting input)
1 = C2VP connects to VDAC
0 = C2VP connects to VREF2

bit 1-0 **C2CH<1:0>:** Comparator C2 Channel Select bits
00 = C2VN of C2 connects to AN4
01 = C2VN of C2 connects to AN5
10 = C2VN of C2 connects to AN6
11 = C2VN of C2 connects to AN7

Note 1: C2OUT will only drive RB7/C2/PSMC1B/$\overline{T1G}$ if:
(C2OE = 1) & (C2ON = 1) & (TRISB<7> = 0) & ((SMCON = 0) or
((SMCOM = 0) & (SCEN = 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

TABLE 13-4: PSMC1A OUTPUT SEQUENCE IN PSM MODE USING C1 AND C2 COMPARATORS

Time	C1OUT	C2OUT	PSMC1A Output Signal
Beginning of PSM cycle	H	H	0 → 1
	L	x	0
	x	L	0
During Pulse Duty Cycle	x	x	No Change
	x	x	No Change
After Pulse Duty Cycle	x	x	1 → 0

Legend: x = Don't Care 0 = Inactive 1 = Active H = High L = Low

13.1.2 SINGLE OR DUAL OUTPUT

The PSMC has the capability to operate with either a single output, or dual alternating outputs. In the single output mode, the PSMC generates an output pulse on PSMC1A output only. The pulses are at the programmed frequency, and are variable between the programmed minimum and maximum duty cycle limits. In the dual output mode, the PSMC generates output pulses which alternate between PSMC1A and PSMC1B. The pulses generated at each output are generated at one half of the programmed frequency, and between 50% of the programmed minimum. and 50% maximum of the output duty cycle. The maximum duty cycle for either output is 50%.

13.1.3 SLOPE COMPENSATION

An optional feature of the PSMC single output mode is the ability to configure the PSMC1B output for use as a slope compensation ramp generator. In this mode, the PSMC1B output is pulled low for the last 1/16 of each pulse cycle. Connecting the PSMC1B output to an RC network, similar to Figure 13-4, results in a positive going pseudo ramp function. This pseudo ramp function is useful as an offset function for the loop error signal in unstable conditions at a duty cycle of greater than 50%.

Note:	When the Slope Compensation switch is enabled (SMCOM = 0, and SCEN = 1), the S1BPOL bit has no effect (see RC Network on next page for more detail).

FIGURE 13-4: SLOPE COMPENSATION (SC) SWITCH OPERATION

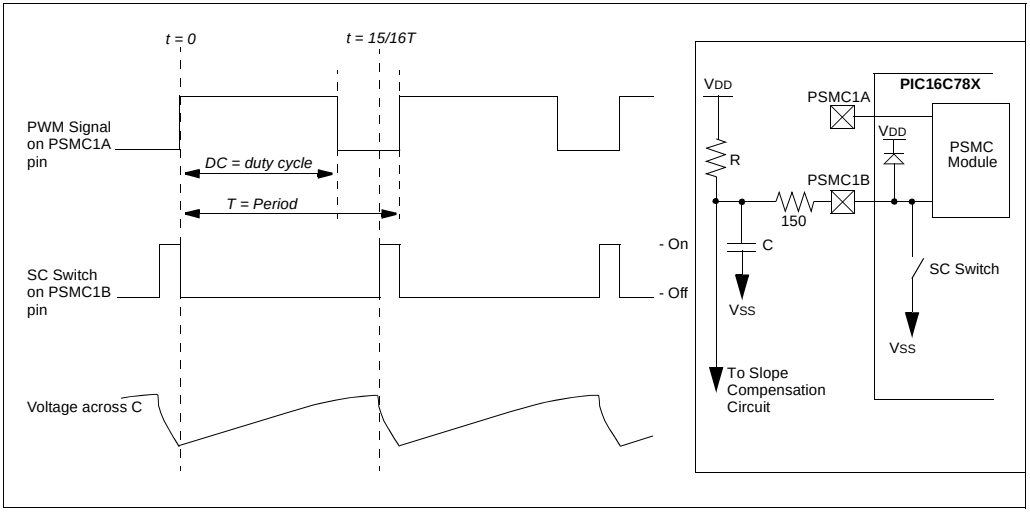
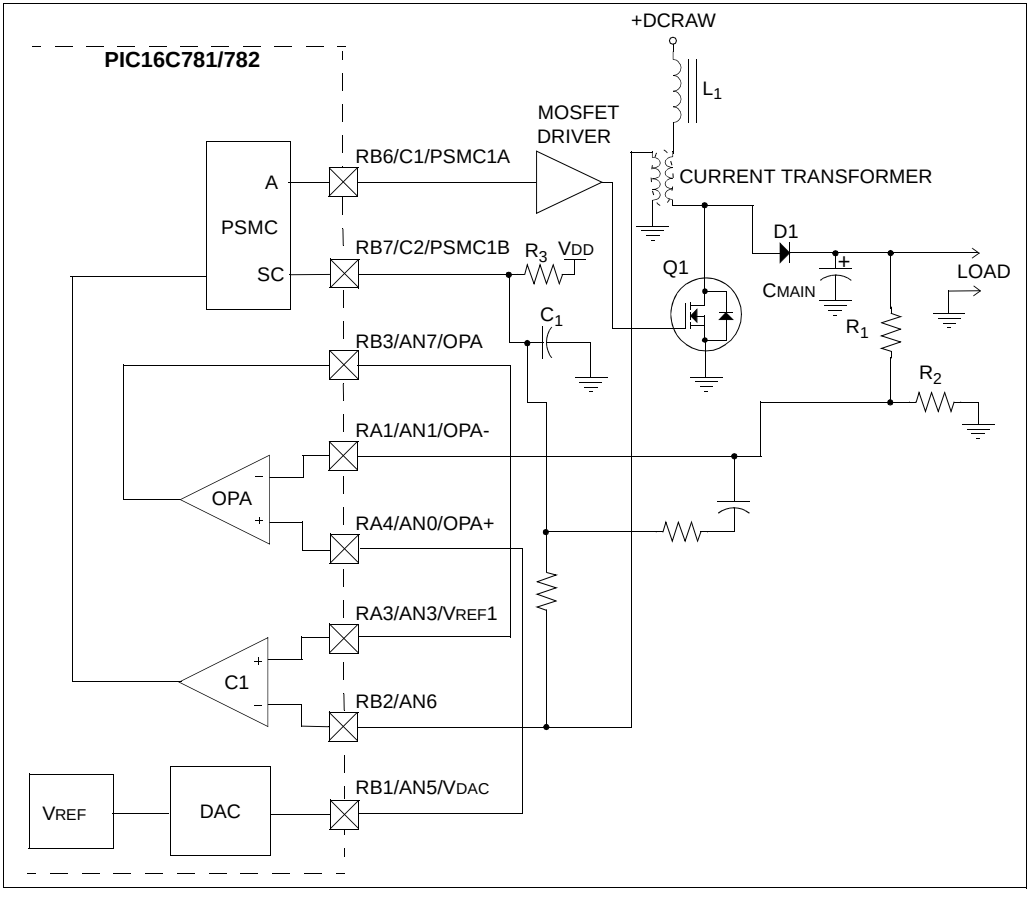


FIGURE 13-5: EXAMPLE BOOST CONFIGURATION LC SWITCHING POWER SUPPLY



Note:	The OPAMP, Comparator and DAC must be configured, prior to enabling the PSMC to prevent unpredictable operation which may stress the power MOSFET transistors.

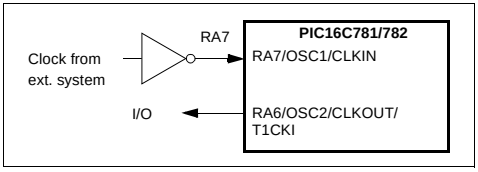
EXAMPLE 13-1: PSMC CONFIGURATION EXAMPLE

```
;* This code block will configure the PSMC and
;*   all additional peripherals for a boost mode
;*   switching power supply.
;*
;*  Order of configuration
;*  1.  PORTA/B I/O and analog configured
;*  2.  DAC enabled, configured, and preset
;*  3.  Op Amp enabled and configured
;*  4.  Comparator C1 enabled and configured
;*  5.  PSMC configured
;*  6.  PSMC enabled


************************************************************
;*  This code block will configure all analog ports.

    BANKSEL     TRISA               ; Select Bank 1
    MOVLW       B'00001011'
    MOVWF       TRISA               ; Set RA0,1,& 3 as inputs
    MOVLW       B'11001110'
    MOVWF       TRISB               ; Set RB1,2,3,6 & 7 as inputs
    MOVLW       B'11101011'         ; Configure RA0, RA1, RA3,
    MOVWF       ANSEL               ; RB1, RB2, RB3 as analog

;*********************************************************************
;*  This code block will configure the DAC for VDD as
;*  DACREF, and RB1/AN5/VDAC as an output

    BANKSEL     DACON0              ; Select Bank 2

    CLRF        DAC                 ; Set DAC to safe value
    MOVLW       B'11000000'         ; Enable DAC, output
    MOVWF       DACON0              ;   and set DACREF = VDD

    MOVLW       OUTPUT_VALUE
    MOVWF       DAC                 ; Set DAC output level
;*********************************************************************
;*  This code block will configure the OPA module as an
;*  Op Amp, with a 2MHz GBWP

    MOVLW       B'10000001'         ; Set Op Amp mode and
    MOVWF       OPACON              ;  2MHz GBWP

;*********************************************************************************
;*  This code block will configure Comparator C1
;*  for normal speed and output polarity,
;*  input on AN6, and Reference from the VREF1

    MOVLW       B'10001010          ; Set C1, no ext out, norm
    MOVWF       CM1CON0             ;   speed & pol, VREF1, AN6

;*********************************************************************
;*  This code block will configure the PSMC module
;*  for PWM, FOSC/128, Single in, Single pulse out, slope comp out
;*  Non-inverting out, DC min = 0%, DC max = 75%

    MOVLW       B'00001000'
    MOVWF       PSMCCON0            ; Set DCmin 0, DCmax 75, FOSC/128
    MOVLW       B'00001010'         ; Set PWM Sngl in, Sngl out non-invert
    MOVWF       PSMCCON1            ; Slope comp
    BSF         PSMCCON1,SMCON      ; Enable PSMC
```

EXAMPLE 13-3: PERIPHERAL CONFIGURATION EXAMPLE

```
;***********************************************
;*  This code block will configure the PSMC and
;*  all additional peripherals for a motor speed
;*  control.
;*
;*  Order of configuration
;*  1.  PORTA/B I/O and analog configured
;*  2.  DAC enabled, configured, and preset
;*  3.  Op Amp enabled and configured
;*  4.  Comparator C1 enabled and configured
;*  5.  PSMC configured
;*  6.  PSMC enabled
;*
;***********************************************
;*  This code block will configure all analog ports.

    BANKSEL     TRISA               ; Select Bank 1
    MOVLW       B'01000011'
    MOVWF       TRISA               ; Set RA0,1 & 6 as inputs
    MOVLW       B'00001100'
    MOVWF       TRISB               ; Set RB2 & 3 as inputs
    MOVLW       B'11000011'
    MOVWF       ANSEL               ; Set AN0,1,6,& 7 as analog

;***********************************************
;*  This code block will configure the DAC for VR as
;*  DACREF, and no output.

    BANKSEL     REFCON
    BSF         REFCON, VREN        ; Enable VR
    BANKSEL     DACON0              ; Select Bank 2

    CLRF        DAC                 ; Set DAC to safe value
    MOVLW       B'10000010'         ; Enable DAC, no output
    MOVWF       DACON0              ;  and set DACREF = VR

    MOVLW       OUTPUT_VALUE
    MOVWF       DAC                 ; Set DAC output level

;***********************************************
;*  This code block will configure the OPA module
;*  as an Op Amp, with a 2 MHz GBWP

    MOVLW       B'10000001'         ; Set Op Amp mode and
    MOVWF       OPACON              ;  2 MHz GBWP

;***********************************************
*   This code block will configure Comparator C1
*   for normal speed and output polarity,
*   input on AN6, and Reference from the VDAC

    MOVLW       B'10001110'         ; Set C1; no ext out, norm
    MOVWF       CM1CON0             ;  speed & pol, VDAC, AN6

;***********************************************
;*  This code block will configure the PSMC module
;*  for PWM, Fosc/16, Single input, Single output
;*  Non-inverting out, DC min = 0%, DC max = 94%

    MOVLW       B'11001100'
    MOVWF       PSMCCON0            ; Set DCmin 0, DCmax 94, Fosc/16
    MOVLW       B'00000010'
    MOVWF       PSMCCON1            ; Set PWM, Sngl in/out, noninvert
    BSF         PSMCCON1,SMCON      ; Enable PSMC
```

FIGURE 14-2: EC OSC CONFIGURATION



14.2.4 RC MODE

For timing insensitive applications, the RC device option offers additional cost savings. The RC oscillator frequency is a function of:

- supply voltage
- resistor (REXT) and capacitor (CEXT) values
- operating temperature

In addition, the oscillator frequency varies from unit to unit due to normal process variation. The difference in lead frame capacitance between package types also affects the oscillation frequency, especially for low CEXT values. The user should allow for variations due to tolerance of external R and C components used. Figure 14-3 shows how the RC combination is connected to the PIC16C781/782. For REXT values below 2.2 kΩ, the oscillator operation may become unstable or stop completely. For very high REXT values (e.g., 1 MΩ or greater), the oscillator becomes sensitive to:

- noise
- humidity
- leakage

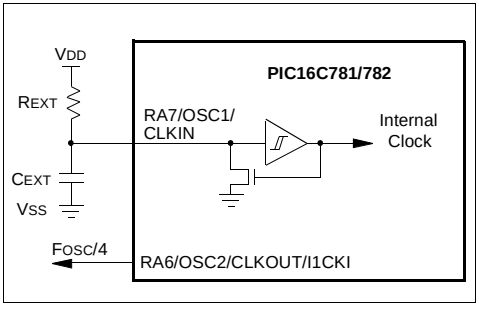
Microchip recommends keeping REXT between 3 kΩ and 100 kΩ.

Although the oscillator will operate with no external capacitor (CEXT = 0 pF), we recommend using values above 20 pF for noise and stability reasons. With no or small external capacitance, the oscillation frequency can vary dramatically due to changes in external capacitances, such as board trace capacitance or package lead frame capacitance.

See Section 18.0 for RC frequency variation from part to part due to normal process variation. The variation is greater for large values of R (since leakage current variations affect RC frequency more for large R) and for small values of C (since variations of input capacitance affect RC frequency more).

See Section 18.0 for variation of oscillator frequency due to VDD for given REXT and CEXT values (or for frequency variation due to operating temperature for given R, C, and VDD values).

FIGURE 14-3: RC OSCILLATOR MODE



14.2.5 INTRC MODE

The internal RC oscillator provides a fixed 4 MHz/37 kHz (nominal) system clock at VDD = 5V and 25°C. See Section 18.0 for information on variations over voltage and temperature ranges. The INTRC oscillator does not run during RESET.

14.2.6 DUAL SPEED OPERATION FOR INTRC MODE

A software programmable slow speed mode is available with the INTRC oscillator. This feature allows the firmware to dynamically toggle the oscillator speed between normal and slow frequencies. The nominal slow frequency is 37 kHz. Applications that require low current power savings, but cannot tolerate putting the part into SLEEP, may use this mode.

The OSCF bit (PCON<3>) is used to control dual speed mode. See the PCON Register, Register 2-6, for details.

When changing the INTRC internal oscillator speed, there is a brief period of time when the processor is inactive. When the speed changes from fast to slow, the processor inactive period is in the range of 100 μS to 300 μS. For a speed change from slow to fast, the processor is inactive between 1.25 μS and 3.25 μS, nominal.

14.2.7 CLKOUT

In the INTRC and RC modes, the PIC16C781/782 can be configured to provide a clock out signal by programming the configuration word. The oscillator frequency, divided by 4, can be used for test purposes or to synchronize other logic.

In the INTRC and RC modes, if the CLKOUT output is enabled, CLKOUT is held low during RESET.

TABLE 14-6: INITIALIZATION CONDITION FOR ALL REGISTERS

Register	Power-On Reset or Brown-Out Reset	$\overline{\text{MCLR}}$ Reset or WDT Reset	Wake-up via WDT or Interrupt
W (not a mapped register)	xxxx xxxx	uuuu uuuu	uuuu uuuu
INDF	0000 0000	uuuu uuuu	uuuu uuuu
TMR0	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCL	0000 0000	0000 0000	PC + 1(1)
STATUS	0001 1xxx	000q quuu(2)	uuuq quuu(2)
FSR	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTA	xxxx 0000	uuuu 0000	uuuu uuuu
PORTB	xxxx xx00	uuuu uu00	uuuu uu00
PCLATH	---0 0000	---0 0000	---u uuuu
INTCON	0000 000x	0000 000u	uuuu uuqq
PIR1	0000 ---0	0000 ---0	0000 ---u
CALCON	000- ----	000- ----	uuu- ----
TMR1L	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR1H	xxxx xxxx	uuuu uuuu	uuuu uuuu
T1CON	-000 0000	-uuu uuuu	-uuu uuuu
PSMCCON0	0000 0000	0000 0000	uuuu uuuu
PSMCCON1	000- 0000	000- 0000	uuu- uuuu
CM1CON0	0000 0000	0000 0000	uuuu uuuu
CM2CON0	0000 0000	0000 0000	uuuu uuuu
CM2CON1	00-- ---0	00-- ---0	uu-- ---u
OPACON	00-- ---0	00-- ---0	uu-- ---u
ADRES	xxxx xxxx	uuuu uuuu	uuuu uuuu
ADCON0	0000 0000	0000 0000	uuuu uuuu
OPTION_REG	1111 1111	1111 1111	uuuu uuuu
TRISA	1111 1111	1111 1111	uuuu uuuu
TRISB	1111 1111	1111 1111	uuuu uuuu
PIE1	0000 ---0	0000 ---0	uuuu ---u
PCON	---0 1-qq	---0 1-uu	---u u-uu
DAC	0000 0000	0000 0000	uuuu uuuu
DACON0	00-- --00	00-- --00	uu-- --uu
WPUB	1111 1111	1111 1111	uuuu uuuu
IOCB	1111 0000	1111 0000	uuuu uuuu
REFCON	---- 00--	---- 00--	---- uu--
LVDCON	--00 0101	--00 0101	--uu uuuu
ANSEL	1111 1111	1111 1111	uuuu uuuu
ADCON1	--00 ----	--00 ----	--uu ----
PMDATL	xxxx xxxx	uuuu uuuu	uuuu uuuu
PMADRL	xxxx xxxx	uuuu uuuu	uuuu uuuu
PMDATH	--xx xxxx	--uu uuuu	--uu uuuu
PMADRH	---- xxxx	---- uuuu	---- uuuu
PMCON1	1--- ---0	1--- ---0	1--- ---0

Legend: u = unchanged, x = unknown, - = unimplemented bit, read as '0', q = value depends on condition

Note 1: When the wake-up is due to an interrupt and the GIE bit is set, the PC is loaded with the interrupt vector (0004h).

2: See Table 14-5 for RESET value for specific condition.

XORWF	Exclusive OR W with f
Syntax:	[*label*] XORWF f,d
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$
Operation:	(W) .XOR. (f) $\rightarrow$ (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'.

17.3 AC Characteristics: PIC16C781/782 (Industrial)

17.3.1 TIMING PARAMETER SYMBOLOGY

The timing parameter symbols have been created following one of the following formats:

1. TppS2ppS
2. TppS

T			
F	Frequency	T	Time

Lowercase letters (pp) and their meanings:

pp			
		osc	OSC1
ck	CLKOUT		
dt	Data in	t0	T0CKI
io	I/O port	t1	T1CKI
mc	$\overline{\text{MCLR}}$		

Uppercase letters and their meanings:

S			
F	Fall	P	Period
H	High	R	Rise
I	Invalid (Hi-impedance)	V	Valid
L	Low	Z	Hi-impedance
		High	High
		Low	Low

FIGURE 17-3: LOAD CONDITIONS

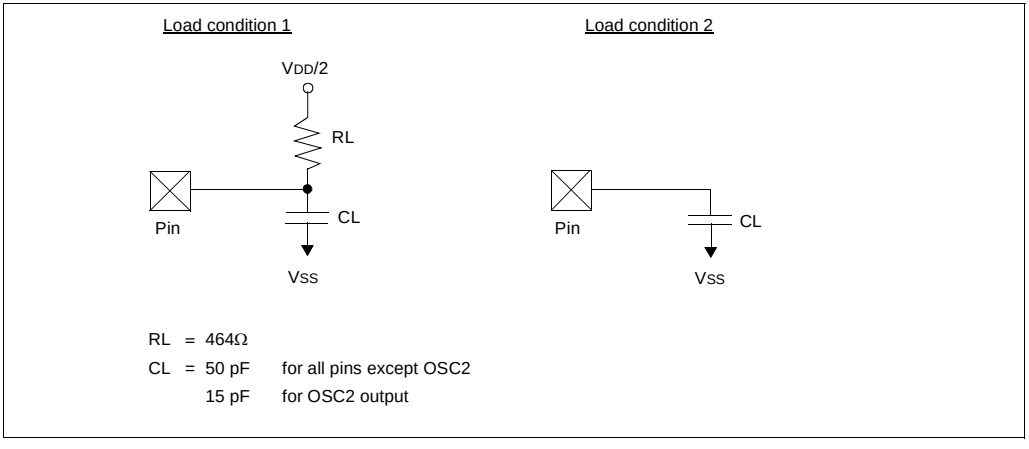


TABLE 17-7: TIMER0 AND TIMER1 EXTERNAL CLOCK REQUIREMENTS

Param No.	Sym	Characteristic			Min	Typ†	Max	Units	Conditions
40*	TTOH	T0CKI High Pulse Width		No Prescaler	0.5TCY + 20	—	—	ns	Must also meet parameter 42
				With Prescaler	10	—	—	ns	
41*	TTOL	T0CKI Low Pulse Width		No Prescaler	0.5TCY + 20	—	—	ns	Must also meet parameter 42
				With Prescaler	10	—	—	ns	
42*	TTOP	T0CKI Period		No Prescaler	TCY + 40	—	—	ns	
				With Prescaler	Greater of: 20 or TCY + 40 / N	—	—	ns	N = prescale value (2, 4, ..., 256)
45*	TT1H	T1CKI High Time	Synchronous, Prescaler = 1		0.5TCY + 20	—	—	ns	Must also meet parameter 47
			Synchronous, Prescaler = 2,4,8	PIC16**C**781/782	15	—	—	ns	
			Asynchronous	PIC16**C**781/782	30	—	—	ns	
45*	TT1H	T1CKI High Time	Synchronous, Prescaler = 1		0.5Tcy + 20	—	—	ns	Must also meet parameter 47
			Synchronous, Prescaler = 2,4,8	PIC16**LC**781/782	15	—	—	ns	
			Asynchronous	PIC16**LC**781/782	30	—	—	ns	
46*	TT1L	T1CKI Low Time	Synchronous, Prescaler = 1		0.5TCY + 20	—	—	ns	Must also meet parameter 47
			Synchronous, Prescaler = 2,4,8	PIC16**C**781/782	15	—	—	ns	
			Asynchronous	PIC16**C**781/782	30	—	—	ns	
46*	TT1L	T1CKI Low Time	Synchronous, Prescaler = 1		0.5TCY + 20	—	—	ns	Must also meet parameter 47
			Synchronous, Prescaler = 2,4,8	PIC16**LC**781/782	15	—	—	ns	
			Asynchronous	PIC16**LC**781/782	30	—	—	ns	
47*	TT1P	T1CKI input period	Synchronous	PIC16**C**781/782	Greater of: 30 OR TCY + 40 / N	—	—	ns	N = prescale value (1, 2, 4, 8)
			Asynchronous	PIC16**C**781/782	60	—	—	ns	
47*	TT1P	T1CKI input period	Synchronous	PIC16**LC**781/782	Greater of: 30 OR TCY + 40 / N	—	—	ns	N = prescale value (1, 2, 4, 8)
			Asynchronous	PIC16**C**781/782	60	—	—	ns	
	FT1	Timer1 oscillator input frequency range (oscillator enabled by setting bit T1OSCEN)			DC	—	50	kHz	
48*	Tcke2tmrl	Delay from external clock edge to timer increment			2TOSC	—	7TOSC	—	

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.