



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 "<u>Embedded -</u> <u>Microcontrollers</u>"

Details

Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	32MHz
Connectivity	I ² C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PSMC, PWM, WDT
Number of I/O	35
Program Memory Size	14KB (8K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	2.3V ~ 5.5V
Data Converters	A/D 14x12b; D/A 1x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	40-UFQFN Exposed Pad
Supplier Device Package	40-UQFN (5x5)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f1787-i-mv

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong

3.2 Data Memory Organization

The data memory is partitioned in 32 memory banks with 128 bytes in a bank. Each bank consists of (Figure 3-3):

- 12 core registers
- 20 Special Function Registers (SFR)
- Up to 80 bytes of General Purpose RAM (GPR)
- 16 bytes of common RAM

The active bank is selected by writing the bank number into the Bank Select Register (BSR). Unimplemented memory will read as '0'. All data memory can be accessed either directly (via instructions that use the file registers) or indirectly via the two File Select Registers (FSR). See **Section 3.6** "**Indirect Addressing**" for more information.

Data memory uses a 12-bit address. The upper 5 bits of the address define the Bank address and the lower 7 bits select the registers/RAM in that bank.

3.2.1 CORE REGISTERS

The core registers contain the registers that directly affect the basic operation. The core registers occupy the first 12 addresses of every data memory bank (addresses x00h/x08h through x0Bh/x8Bh). These registers are listed below in Table 3-2. For detailed information, see Table 3-11.

TABLE 3-2: CORE REGISTER	S
--------------------------	---

Addresses	BANKx
x00h or x80h	INDF0
x01h or x81h	INDF1
x02h or x82h	PCL
x03h or x83h	STATUS
x04h or x84h	FSR0L
x05h or x85h	FSR0H
x06h or x86h	FSR1L
x07h or x87h	FSR1H
x08h or x88h	BSR
x09h or x89h	WREG
x0Ah or x8Ah	PCLATH
x0Bh or x8Bh	INTCON

FIGURE 3-6: ACCESSING THE STACK EXAMPLE 2



4.6 Device ID and Revision ID

The memory location 8006h is where the Device ID and Revision ID are stored. The upper nine bits hold the Device ID. The lower five bits hold the Revision ID. See **Section 12.5 "User ID, Device ID and Configuration Word Access**" for more information on accessing these memory locations.

Development tools, such as device programmers and debuggers, may be used to read the Device ID and Revision ID.

4.7 Register Definitions: Device and Revision

REGISTER 4-3: DEVID: DEVICE ID REGISTER



Legend:

R = Readable bit '1' = Bit is set

bit 13-5 **DEV<8:0>:** Device ID bits

Device	DEVICEID<13	:0> Values
Device	DEV<8:0>	REV<4:0>
PIC16F1784	10 1010 010	x xxxx
PIC16LF1784	10 1010 111	x xxxx
PIC16F1786	10 1010 011	x xxxx
PIC16LF1786	10 1011 000	x xxxx
PIC16F1787	10 1010 100	x xxxx
PIC16LF1787	10 1011 001	x xxxx

'0' = Bit is cleared

bit 4-0 R

REV<4:0>: Revision ID bits

These bits are used to identify the revision (see Table under DEV<8:0> above).

6.0 OSCILLATOR MODULE (WITH FAIL-SAFE CLOCK MONITOR)

6.1 Overview

The oscillator module has a wide variety of clock sources and selection features that allow it to be used in a wide range of applications while maximizing performance and minimizing power consumption. Figure 6-1 illustrates a block diagram of the oscillator module.

Clock sources can be supplied from external oscillators, quartz crystal resonators, ceramic resonators and Resistor-Capacitor (RC) circuits. In addition, the system clock source can be supplied from one of two internal oscillators and PLL circuits, with a choice of speeds selectable via software. Additional clock features include:

- Selectable system clock source between external or internal sources via software.
- Two-Speed Start-up mode, which minimizes latency between external oscillator start-up and code execution.
- Fail-Safe Clock Monitor (FSCM) designed to detect a failure of the external clock source (LP, XT, HS, EC or RC modes) and switch automatically to the internal oscillator.
- Oscillator Start-up Timer (OST) ensures stability of crystal oscillator sources

The oscillator module can be configured in one of eight clock modes.

- 1. ECL External Clock Low-Power mode (0 MHz to 0.5 MHz)
- 2. ECM External Clock Medium-Power mode (0.5 MHz to 4 MHz)
- 3. ECH External Clock High-Power mode (4 MHz to 32 MHz)
- 4. LP 32 kHz Low-Power Crystal mode.
- 5. XT Medium Gain Crystal or Ceramic Resonator Oscillator mode (up to 4 MHz)
- 6. HS High Gain Crystal or Ceramic Resonator mode (4 MHz to 20 MHz)
- 7. RC External Resistor-Capacitor (RC).
- 8. INTOSC Internal oscillator (31 kHz to 32 MHz).

Clock Source modes are selected by the FOSC<2:0> bits in the Configuration Words. The FOSC bits determine the type of oscillator that will be used when the device is first powered.

The EC clock mode relies on an external logic level signal as the device clock source. The LP, XT, and HS clock modes require an external crystal or resonator to be connected to the device. Each mode is optimized for a different frequency range. The RC clock mode requires an external resistor and capacitor to set the oscillator frequency.

The INTOSC internal oscillator block produces low, medium, and high-frequency clock sources, designated LFINTOSC, MFINTOSC and HFINTOSC. (see Internal Oscillator Block, Figure 6-1). A wide selection of device clock frequencies may be derived from these three clock sources.

6.2.1.4 4x PLL

The oscillator module contains a 4x PLL that can be used with both external and internal clock sources to provide a system clock source. The input frequency for the 4x PLL must fall within specifications. See the PLL Clock Timing Specifications in **Section 30.0 "Electrical Specifications**".

The 4x PLL may be enabled for use by one of two methods:

- 1. Program the PLLEN bit in Configuration Words to a '1'.
- Write the SPLLEN bit in the OSCCON register to a '1'. If the PLLEN bit in Configuration Words is programmed to a '1', then the value of SPLLEN is ignored.

6.2.1.5 TIMER1 Oscillator

The Timer1 oscillator is a separate crystal oscillator that is associated with the Timer1 peripheral. It is optimized for timekeeping operations with a 32.768 kHz crystal connected between the T1OSO and T1OSI device pins.

The Timer1 oscillator can be used as an alternate system clock source and can be selected during run-time using clock switching. Refer to **Section 6.3 "Clock Switching"** for more information.

FIGURE 6-5: QUARTZ CRYSTAL OPERATION (TIMER1



- Note 1: Quartz crystal characteristics vary according to type, package and manufacturer. The user should consult the manufacturer data sheets for specifications and recommended application.
 - **2:** Always verify oscillator performance over the VDD and temperature range that is expected for the application.
 - **3:** For oscillator design assistance, reference the following Microchip Applications Notes:
 - AN826, "Crystal Oscillator Basics and Crystal Selection for rfPIC[®] and PIC[®] Devices" (DS00826)
 - AN849, "Basic PIC[®] Oscillator Design" (DS00849)
 - AN943, "Practical PIC[®] Oscillator Analysis and Design" (DS00943)
 - AN949, "Making Your Oscillator Work" (DS00949)
 - TB097, "Interfacing a Micro Crystal MS1V-T1K 32.768 kHz Tuning Fork Crystal to a PIC16F690/SS" (DS91097)
 - AN1288, "Design Practices for Low-Power External Oscillators" (DS01288)

8.3 Interrupts During Sleep

Some interrupts can be used to wake from Sleep. To wake from Sleep, the peripheral must be able to operate without the system clock. The interrupt source must have the appropriate Interrupt Enable bit(s) set prior to entering Sleep.

On waking from Sleep, if the GIE bit is also set, the processor will branch to the interrupt vector. Otherwise, the processor will continue executing instructions after the SLEEP instruction. The instruction directly after the SLEEP instruction will always be executed before branching to the ISR. Refer to **Section 9.0** "**Power-Down Mode (Sleep)**" for more details.

8.4 INT Pin

The INT pin can be used to generate an asynchronous edge-triggered interrupt. This interrupt is enabled by setting the INTE bit of the INTCON register. The INTEDG bit of the OPTION_REG register determines on which edge the interrupt will occur. When the INTEDG bit is set, the rising edge will cause the interrupt. When the INTEDG bit is clear, the falling edge will cause the interrupt. The INTF bit of the INTCON register will be set when a valid edge appears on the INT pin. If the GIE and INTE bits are also set, the processor will redirect program execution to the interrupt vector.

8.5 Automatic Context Saving

Upon entering an interrupt, the return PC address is saved on the stack. Additionally, the following registers are automatically saved in the shadow registers:

- W register
- STATUS register (except for TO and PD)
- · BSR register
- FSR registers
- PCLATH register

Upon exiting the Interrupt Service Routine, these registers are automatically restored. Any modifications to these registers during the ISR will be lost. If modifications to any of these registers are desired, the corresponding shadow register should be modified and the value will be restored when exiting the ISR. The shadow registers are available in Bank 31 and are readable and writable. Depending on the user's application, other registers may also need to be saved.

U-0	U-0	U-0	R/W-0/0	U-0	U-0	U-0	U-0
_	—	—	CCP3IF	—			_
bit 7							bit 0
Legend:							
R = Readable I	bit	W = Writable I	oit	U = Unimpler	nented bit, read	as '0'	
u = Bit is uncha	anged	x = Bit is unkn	own	-n/n = Value a	at POR and BO	R/Value at all o	ther Resets
'1' = Bit is set		'0' = Bit is clea	ared				
bit 7-5	Unimplement	ted: Read as 'o)'				
bit 4	CCP3IF: CCF	3 Interrupt Flag	g bit				
	1 = Interrupt is	s pending					
	0 = Interrupt is	s not pending					
bit 3-0	Unimplement	ted: Read as '()'				

REGISTER 8-8: PIR3: PERIPHERAL INTERRUPT REQUEST REGISTER 3

13.3.7 PORTA FUNCTIONS AND OUTPUT PRIORITIES

Each PORTA pin is multiplexed with other functions. The pins, their combined functions and their output priorities are shown in Table 13-2.

When multiple outputs are enabled, the actual pin control goes to the peripheral with the highest priority.

Analog input functions, such as ADC, and comparator inputs, are not shown in the priority lists. These inputs are active when the I/O pin is set for Analog mode using the ANSELx registers. Digital output functions may control the pin when it is in Analog mode with the priority shown in the priority list.

Pin Name	Function Priority ⁽¹⁾
RA0	RA0
RA1	OPA1OUT RA1
RA2	DAC1OUT1 RA2
RA3	RA3
RA4	C1OUT RA4
RA5	C2OUT RA5
RA6	CLKOUT C2OUT RA6
RA7	RA7

TABLE 13-2: PORTA OUTPUT PRIORITY

Note 1: Priority listed from highest to lowest.

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ANSELB	—	ANSB6	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1	ANSB0	138
INLVLB	INLVLB7	INLVLB6	INLVLB5	INLVLB4	INLVLB3	INLVLB2	INLVLB1	INLVLB0	139
LATB	LATB7	LATB6	LATB5	LATB4	LATB3	LATB2	LATB1	LATB0	137
ODCONB	ODB7	ODB6	ODB5	ODB4	ODB3	ODB2	ODB1	ODB0	139
PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	137
SLRCONB	SLRB7	SLRB6	SLRB5	SLRB4	SLRB3	SLRB2	SLRB1	SLRB0	139
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	137
WPUB	WPUB7	WPUB6	WPUB5	WPUB4	WPUB3	WPUB2	WPUB1	WPUB0	138

TABLE 13-6: SUMMARY OF REGISTERS ASSOCIATED WITH PORTB

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

17.3 Register Definitions: ADC Control

REGISTER 17-1: ADCON0: ADC 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	R/W-0/0	R/W-0/0
ADRMD			CHS<4:0>			GO/DONE	ADON
bit 7							bit 0
Legend:							
R = Readab	le bit	W = Writable bi	t	U = Unimpleme	ented bit, read a	s '0'	
u = Bit is un	changed	x = Bit is unkno	wn	-n/n = Value at	POR and BOR/	Value at all other	Resets
'1' = Bit is se	et	'0' = Bit is clear	ed				
bit 7	ADRMD: ADC 1 = ADRESL 0 = ADRESL See Figure 17	Result Mode bit and ADRESH pro and ADRESH pro -3 for details	ovide data form ovide data form	atted for a 10-bit atted for a 12-bit	result result		
bit 6-2	CHS<4:0>: Po 11111 = FVR 11110 = DAO 11101 = Tem 11100 = Res 10110 = Res 10101 = AN2 10100 = Res	psitive Differential (Fixed Voltage R C_output ⁽²⁾ aperature Indicato erved. No channe erved. No channe 21 ⁽¹⁾ erved. No channe	Input Channel Reference) Buffe r ⁽⁴⁾ el connected. el connected el connected	Select bits er 1 Output ⁽³⁾			
	•						
bit 1	01110 = Res 01101 = AN1 01100 = AN1 01011 = AN1 01010 = AN1 01001 = AN9 01000 = AN8 00111 = AN7 00110 = AN6 00101 = AN5 00100 = AN4 00011 = AN3 00010 = AN2 00010 = AN2 00001 = AN2	erved. No channe 3 2 1 0 (1) (1) (1) (1) (2) DC Conversion St	el connected.				
	1 = ADC conve This bit is a 0 = ADC conve	ersion cycle in pro automatically clea ersion completed	ogress. Setting ared by hardwar /not in progress	this bit starts an <i>I</i> e when the ADC	ADC conversion conversion has	cycle. completed.	
bit 0	ADON: ADC E 1 = ADC is en: 0 = ADC is dis	Enable bit abled abled and consur	nes no operatir	ig current			
Note 1: 2: 3: 4:	PIC16(L)F1784/7 or See Section 19.0 " See Section 15.0 " See Section 16.0 "	nly. Digital-to-Analog Fixed Voltage Re Temperature Ind	g Converter (D eference (FVR) licator Module	AC) Module" for " for more inform " for more informa	more informatio ation. ation.	n.	

FIGURE 17-4: ANALOG INPUT MODEL







21.0 TIMER0 MODULE

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

- 8-bit timer/counter register (TMR0)
- 8-bit prescaler (independent of Watchdog Timer)
- Programmable internal or external clock source
- Programmable external clock edge selection
- · Interrupt on overflow
- TMR0 can be used to gate Timer1

Figure 21-1 is a block diagram of the Timer0 module.

21.1 Timer0 Operation

The Timer0 module can be used as either an 8-bit timer or an 8-bit counter.

21.1.1 8-BIT TIMER MODE

The Timer0 module will increment every instruction cycle, if used without a prescaler. 8-bit Timer mode is selected by clearing the TMR0CS bit of the OPTION_REG register.

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.

FIGURE 21-1: BLOCK DIAGRAM OF THE TIMER0



In 8-Bit Counter mode, the Timer0 module will increment on every rising or falling edge of the T0CKI pin.

8-Bit Counter mode using the T0CKI pin is selected by setting the TMR0CS bit in the OPTION_REG register to '1'.

The rising or falling transition of the incrementing edge for either input source is determined by the TMR0SE bit in the OPTION_REG register.



Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
CCP2CON	—	—	DC2B	<1:0>		CCP2	/ <3:0>		280
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	98
PR2	Timer2 Mod	dule Period	Register						210*
T2CON	—		T2OUTF	PS<3:0>		TMR2ON	T2CKP	S<1:0>	212
TMR2	Holding Re	gister for the	e 8-bit TMR2	2 Register					210*

TABLE 23-1: SUMMARY OF REGISTERS ASSOCIATED WITH TIMER2

Legend: — = unimplemented location, read as '0'. Shaded cells are not used for Timer2 module.

* Page provides register information.

24.6 PSMC Modulation (Burst Mode)

PSMC modulation is a method to stop/start PWM operation of the PSMC without having to disable the module. It also allows other modules to control the operational period of the PSMC. This is also referred to as Burst mode.

This is a method to implement PWM dimming.

24.6.1 MODULATION ENABLE

The modulation function is enabled by setting the PxMDLEN bit of PSMC Modulation Control (PSMCxMDL) register (Register 24-2).

When modulation is enabled, the modulation source controls when the PWM signals are active and inactive.

When modulation is disabled, the PWM signals operate continuously, regardless of the selected modulation source.

24.6.2 MODULATION SOURCES

There are multiple sources that can be used for modulating the PSMC. However, unlike the PSMC input sources, only one modulation source can be selected at a time. Modulation sources include:

- PSMCxIN pin
- Any CCP output
- · Any Comparator output
- · PxMDLBIT of the PSMCxMDL register



FIGURE 24-19: PSMC MODULATION WAVEFORM

24.6.2.1 PxMDLBIT Bit

The PxMDLBIT bit of the PSMC Modulation Control (PSMCxMDL) register (Register 24-2) allows for software modulation control without having to enable/disable other module functions.

24.6.3 MODULATION EFFECT ON PWM SIGNALS

When modulation starts, the PSMC begins operation on a new period, just as if it had rolled over from one period to another during continuous operation.

When modulation stops, its operation depends on the type of waveform being generated.

In operation modes other than Fixed Duty Cycle, the PSMC completes its current PWM period and then freezes the module. The PSMC output pins are forced into the default inactive state ready for use when modulation starts.

In Fixed Duty Cycle mode operation, the PSMC continues to operate until the period event changes the PWM to its inactive state, at which point the PSMC module is frozen. The PSMC output pins are forced into the default inactive state ready for use when modulation starts.

26.6.7 I²C MASTER MODE RECEPTION

Master mode reception is enabled by programming the Receive Enable bit, RCEN bit of the SSPCON2 register.

Note:	The MSSP module must be in an Idle
	state before the RCEN bit is set or the
	RCEN bit will be disregarded.

The Baud Rate Generator begins counting and on each rollover, the state of the SCL pin changes (high-to-low/low-to-high) and data is shifted into the SSPSR. After the falling edge of the eighth clock, the receive enable flag is automatically cleared, the contents of the SSPSR are loaded into the SSPBUF, the BF flag bit is set, the SSP1IF flag bit is set and the Baud Rate Generator is suspended from counting, holding SCL low. The MSSP is now in Idle state awaiting the next command. When the buffer is read by the CPU, the BF flag bit is automatically cleared. The user can then send an Acknowledge bit at the end of reception by setting the Acknowledge Sequence Enable, ACKEN bit of the SSPCON2 register.

26.6.7.1 BF Status Flag

In receive operation, the BF bit is set when an address or data byte is loaded into SSPBUF from SSPSR. It is cleared when the SSPBUF register is read.

26.6.7.2 SSPOV Status Flag

In receive operation, the SSPOV bit is set when 8 bits are received into the SSPSR and the BF flag bit is already set from a previous reception.

26.6.7.3 WCOL Status Flag

If the user writes the SSPBUF when a receive is already in progress (i.e., SSPSR is still shifting in a data byte), the WCOL bit is set and the contents of the buffer are unchanged (the write does not occur). 26.6.7.4 Typical Receive Sequence:

- 1. The user generates a Start condition by setting the SEN bit of the SSPCON2 register.
- 2. SSP1IF is set by hardware on completion of the Start.
- 3. SSP1IF is cleared by software.
- 4. User writes SSPBUF with the slave address to transmit and the R/W bit set.
- 5. Address is shifted out the SDA pin until all 8 bits are transmitted. Transmission begins as soon as SSPBUF is written to.
- 6. The MSSP module shifts in the ACK bit from the slave device and writes its value into the ACKSTAT bit of the SSPCON2 register.
- 7. The MSSP module generates an interrupt at the end of the ninth clock cycle by setting the SSP1IF bit.
- 8. User sets the RCEN bit of the SSPCON2 register and the master clocks in a byte from the slave.
- 9. After the 8th falling edge of SCL, SSP1IF and BF are set.
- 10. User clears the SSP1IF bit and reads the received byte from SSPUF, which clears the BF flag.
- 11. The user either clears the SSPCON2 register's ACKDT bit to receive another byte or sets the ADKDT bit to suppress further data and then initiates the acknowledge sequence by setting the ACKEN bit.
- 12. Master's ACK or ACK is clocked out to the slave and SSP1IF is set.
- 13. User clears SSP1IF.
- 14. Steps 8-13 are repeated for each received byte from the slave.
- 15. If the ACKST bit was set in step 11 then the user can send a STOP to release the bus.

Standa	rd Operating	g Conditions (unless otherwise stated)					
Param No.	Sym.	Characteristic	Min.	Тур†	Max.	Units	Conditions
OS11	TosH2ckL	Fosc↑ to CLKOUT↓ ⁽¹⁾	—	_	70	ns	VDD = 3.3-5.0V
OS12	TosH2ckH	Fosc↑ to CLKOUT↑ ⁽¹⁾	—	_	72	ns	VDD = 3.3-5.0V
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 = 3.3-5.0V
OS16	TosH2iol	Fosc↑ (Q2 cycle) to Port input invalid (I/O in hold time)	50	_	_	ns	VDD = 3.3-5.0V
OS17	TioV2osH	Port input valid to Fosc↑ (Q2 cycle) (I/O in setup time)	20	_	_	ns	
OS18*	TioR	Port output rise time	—	40	72	ns	VDD = 1.8V
				15	32		VDD = 3.3-5.0V
OS19*	TioF	Port output fall time	—	28	55	ns	VDD = 1.8V
			_	15	30		VDD = 3.3-5.0V
OS20*	Tinp	INT pin input high or low time	25	_	_	ns	
OS21*	Tioc	Interrupt-on-change new input level time	25	_	_	ns	

TABLE 30-9: CLKOUT AND I/O TIMING PARAMETERS

* These parameters are characterized but not tested.

† Data in "Typ" column is at 3.0V, 25°C unless otherwise stated.

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

FIGURE 30-10: TIMER0 AND TIMER1 EXTERNAL CLOCK TIMINGS



|--|

Standard Operating Conditions (unless otherwise stated)												
Param No.	Sym.	Characteristic			Min.	Тур†	Max.	Units	Conditions			
40*	T⊤0H	T0CKI High Pulse Width		No Prescaler	0.5 Tcy + 20	—	_	ns				
				With Prescaler	10	_		ns				
41*	T⊤0L	T0CKI Low Pulse Width		No Prescaler	0.5 Tcy + 20	—		ns				
				With Prescaler	10	—		ns				
42*	TT0P	T0CKI Period			Greater of: 20 or <u>Tcy + 40</u> N	_		ns	N = prescale value (2, 4,, 256)			
45*	T⊤1H	T1CKI High Time	Synchronous, No Prescaler		0.5 Tcy + 20	_	_	ns				
			Synchronous, with Prescaler		15	—		ns				
			Asynchronous		30	_	_	ns				
46*	TT1L	T1CKI Low Time	Synchronous, No Prescaler		0.5 Tcy + 20	_	_	ns				
			Synchronous, with Prescaler		15	—	_	ns				
			Asynchronous		30	_		ns				
47*	TT1P	T1CKI Input Period	Synchronous		Greater of: 30 or <u>Tcy + 40</u> N	_		ns	N = prescale value (1, 2, 4, 8)			
			Asynchronous		60	—	_	ns				
48	F⊤1	Timer1 Oscillator Input Frequency Range (oscillator enabled by setting bit T1OSCEN)			32.4	32.768	33.1	kHz				
49*	TCKEZTMR1	Delay from External Clock Edge to Timer Increment			2 Tosc	_	7 Tosc	—	Timers in Sync mode			

* These parameters are characterized but not tested.

† Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note: Unless otherwise noted, VIN = 5V, Fosc = 300 kHz, CIN = 0.1 μ F, TA = 25°C.







FIGURE 31-69: PWRT Period, PIC16F1784/6/7 Only.









FIGURE 31-72: POR Rearm Voltage, NP Mode (VREGPM = 0), PIC16F1784/6/7 Only.

32.0 DEVELOPMENT SUPPORT

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

32.1 MPLAB X Integrated Development Environment Software

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 that includes code completion and context menus, MPLAB X IDE is flexible and friendly enough for new users. With the ability to support multiple tools on multiple projects with simultaneous debugging, MPLAB X IDE is also suitable for the needs of experienced users.

Feature-Rich Editor:

- Color syntax highlighting
- Smart code completion makes suggestions and provides hints as you type
- Automatic code formatting based on user-defined rules
- · Live parsing

User-Friendly, Customizable Interface:

- Fully customizable interface: toolbars, toolbar buttons, windows, window placement, etc.
- · Call graph window
- Project-Based Workspaces:
- Multiple projects
- Multiple tools
- Multiple configurations
- · Simultaneous debugging sessions
- File History and Bug Tracking:
- Local file history feature
- · Built-in support for Bugzilla issue tracker

28-Lead Plastic Small Outline (SO) - Wide, 7.50 mm Body [SOIC]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



RECOMMENDED LAND PATTERN

	MILLIMETERS				
Dimension	MIN	NOM	MAX		
Contact Pitch	E		1.27 BSC		
Contact Pad Spacing	С		9.40		
Contact Pad Width (X28)	Х			0.60	
Contact Pad Length (X28)	Y			2.00	
Distance Between Pads	Gx	0.67			
Distance Between Pads	G	7.40			

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2052A