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	32MHz
Connectivity	LINbus, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	18
Program Memory Size	14KB (8K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 12x10b; D/A 1x5b
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/pic16lf1579-i-ss

TABLE 3-3: PIC16(L)F1574 MEMORY MAP, BANKS 0-7

	BANK0		BANK1		BANK2		BANK3		BANK4		BANK5		BANK6		BANK7
000h	Core Registers (Table 3-2)	080h	Core Registers (Table 3-2)	100h	Core Registers (Table 3-2)	180h	Core Registers (Table 3-2)	200h	Core Registers (Table 3-2)	280h	Core Registers (Table 3-2)	300h	Core Registers (Table 3-2)	380h	Core Registers (Table 3-2)
00Bh		08Bh		10Bh		18Bh		20Bh		28Bh		30Bh		38Bh	
00Ch	PORTA	08Ch	TRISA	10Ch	LATA	18Ch	ANSELA	20Ch	WPUA	28Ch	ODCONA	30Ch	SLRCONA	38Ch	INLVLA
00Dh	—	08Dh	—	10Dh	—	18Dh	—	20Dh	—	28Dh	—	30Dh	—	38Dh	—
00Eh	PORTC	08Eh	TRISC	10Eh	LATC	18Eh	ANSELC	20Eh	WPUC	28Eh	ODCONC	30Eh	SLRCONC	38Eh	INLVLC
00Fh	—	08Fh	—	10Fh	—	18Fh	—	20Fh	—	28Fh	—	30Fh	—	38Fh	—
010h	—	090h	—	110h	—	190h	—	210h	—	290h	—	310h	—	390h	—
011h	PIR1	091h	PIE1	111h	CM1CON0	191h	PMADRL	211h	—	291h	—	311h	—	391h	IOCAP
012h	PIR2	092h	PIE2	112h	CM1CON1	192h	PMADRH	212h	—	292h	—	312h	—	392h	IOCAN
013h	PIR3	093h	PIE3	113h	CM2CON0	193h	PMDATL	213h	—	293h	—	313h	—	393h	IOCAF
014h	—	094h	—	114h	CM2CON1	194h	PMDATH	214h	—	294h	—	314h	—	394h	—
015h	TMR0	095h	OPTION_REG	115h	CMOUT	195h	PMCON1	215h	—	295h	—	315h	—	395h	—
016h	TMR1L	096h	PCON	116h	BORCON	196h	PMCON2	216h	—	296h	—	316h	—	396h	—
017h	TMR1H	097h	WDTCON	117h	FVRCON	197h	VREGCON(1)	217h	—	297h	—	317h	—	397h	IOCCP
018h	T1CON	098h	OSCTUNE	118h	DACCON0	198h	—	218h	—	298h	—	318h	—	398h	IOCCN
019h	T1GCON	099h	OSCCON	119h	DACCON1	199h	RCREG	219h	—	299h	—	319h	—	399h	IOCCF
01Ah	TMR2	09Ah	OSCSTAT	11Ah	—	19Ah	TXREG	21Ah	—	29Ah	—	31Ah	—	39Ah	—
01Bh	PR2	09Bh	ADRESL	11Bh	—	19Bh	SPBRGL	21Bh	—	29Bh	—	31Bh	—	39Bh	—
01Ch	T2CON	09Ch	ADRESH	11Ch	—	19Ch	SPBRGH	21Ch	—	29Ch	—	31Ch	—	39Ch	—
01Dh	—	09Dh	ADCON0	11Dh	—	19Dh	RCSTA	21Dh	—	29Dh	—	31Dh	—	39Dh	—
01Eh	—	09Eh	ADCON1	11Eh	—	19Eh	TXSTA	21Eh	—	29Eh	—	31Eh	—	39Eh	—
01Fh	—	09Fh	ADCON2	11Fh	—	19Fh	BAUDCON	21Fh	—	29Fh	—	31Fh	—	39Fh	—
020h	General Purpose Register 80 Bytes	0A0h	General Purpose Register 80 Bytes	120h	General Purpose Register 80 Bytes	1A0h	General Purpose Register 80 Bytes	220h	General Purpose Register 80 Bytes	2A0h	General Purpose Register 80 Bytes	320h	General Purpose Register 16 Bytes	3A0h	Unimplemented Read as '0'
06Fh		0EFh		16Fh		1EFh		26Fh		2EFh		32Fh		3EFh	
070h	Common RAM	0F0h	Accesses 70h – 7Fh	170h	Accesses 70h – 7Fh	1F0h	Accesses 70h – 7Fh	270h	Accesses 70h – 7Fh	2F0h	Accesses 70h – 7Fh	330h	Unimplemented Read as '0'		
07Fh		0FFh		17Fh		1FFh		27Fh		2FFh		36Fh			
												370h	Accesses 70h – 7Fh	3F0h	Accesses 70h – 7Fh
												37Fh		3FFh	

Legend: = Unimplemented data memory locations, read as '0'.

Note 1: Unimplemented on PIC16LF1574.

TABLE 3-10: PIC16(L)F1574/5/8/9 MEMORY MAP, BANKS 24-31

BANK 24		BANK 25		BANK 26		BANK 27		BANK 28		BANK 29		BANK 30		BANK 31	
C00h	Core Registers (Table 3-2)	C80h	Core Registers (Table 3-2)	D00h	Core Registers (Table 3-2)	D80h	Core Registers (Table 3-2)	E00h	Core Registers (Table 3-2)	E80h	Core Registers (Table 3-2)	F00h	Core Registers (Table 3-2)	F80h	Core Registers (Table 3-2)
C0Bh		C8Bh		D0Bh		D8Bh		E0Bh		E8Bh		F0Bh		F8Bh	
C0Ch	—	C8Ch	—	D0Ch	—	D8Ch	See Table 3-11	E0Ch	See Table 3-12	E8Ch	See Table 3-12	F0Ch	—	F8Ch	See Table 3-13
C0Dh	—	C8Dh	—	D0Dh	—							F0Dh	—		
C0Eh	—	C8Eh	—	D0Eh	—							F0Eh	—		
C0Fh	—	C8Fh	—	D0Fh	—							F0Fh	—		
C10h	—	C90h	—	D10h	—							F10h	—		
C11h	—	C91h	—	D11h	—							F11h	—		
C12h	—	C92h	—	D12h	—							F12h	—		
C13h	—	C93h	—	D13h	—							F13h	—		
C14h	—	C94h	—	D14h	—							F14h	—		
C15h	—	C95h	—	D15h	—							F15h	—		
C16h	—	C96h	—	D16h	—							F16h	—		
C17h	—	C97h	—	D17h	—							F17h	—		
C18h	—	C98h	—	D18h	—							F18h	—		
C19h	—	C99h	—	D19h	—							F19h	—		
C1Ah	—	C9Ah	—	D1Ah	—							F1Ah	—		
C1Bh	—	C9Bh	—	D1Bh	—							F1Bh	—		
C1Ch	—	C9Ch	—	D1Ch	—							F1Ch	—		
C1Dh	—	C9Dh	—	D1Dh	—							F1Dh	—		
C1Eh	—	C9Eh	—	D1Eh	—							F1Eh	—		
C1Fh	—	C9Fh	—	D1Fh	—							F1Fh	—		
C20h	Unimplemented Read as ‘0’	CA0h	Unimplemented Read as ‘0’	D20h	Unimplemented Read as ‘0’							F20h	Unimplemented Read as ‘0’		
C6Fh		CEFh		D6Fh		DEFh		E6Fh		EEFh		F6Fh		FEFh	
C70h	Accesses 70h – 7Fh	CF0h	Accesses 70h – 7Fh	D70h	Accesses 70h – 7Fh	DF0h	Accesses 70h – 7Fh	E70h	Accesses 70h – 7Fh	EF0h	Accesses 70h – 7Fh	F70h	Accesses 70h – 7Fh	FF0h	Accesses 70h – 7Fh
CFFh		CFFh		D7Fh		DFFh		E7Fh		EFFh		F7Fh		FFFh	

Legend: = Unimplemented data memory locations, read as ‘0’

TABLE 3-11: PIC16(L)F1574/5/8/9 MEMORY MAP, BANK 27

	Bank 27
D8Ch	—
D8Dh	—
D8Eh	PWMEN
D8Fh	PWMLD
D90h	PWMOUT
D91h	PWM1PHL
D92h	PWM1PHH
D93h	PWM1DCL
D94h	PWM1DCH
D95h	PWM1PRL
D96h	PWM1PRH
D97h	PWM1OFL
D98h	PWM1OFH
D99h	PWM1TMRL
D9Ah	PWM1TMRH
D9Bh	PWM1CON
D9Ch	PWM1INTE
D9Dh	PWM1INTF
D9Eh	PWM1CLKCON
D9Fh	PWM1LDCON
DA0h	PWM1OFCON
DA1h	PWM2PHL
DA2h	PWM2PHH
DA3h	PWM2DCL
DA4h	PWM2DCH
DA5h	PWM2PRL
DA6h	PWM2PRH
DA7h	PWM2OFL
DA8h	PWM2OFH
DA9h	PWM2TMRL
DAAh	PWM2TMRH
DABh	PWM2CON
DACh	PWM2INTE
DADh	PWM2INTF
DAEh	PWM2CLKCON
DAFh	PWM2LDCON
DB0h	PWM2OFCON
DB1h	PWM3PHL
DB2h	PWM3PHH
DB3h	PWM3DCL
DB4h	PWM3DCH
DB5h	PWM3PRL
DB6h	PWM3PRH
DB7h	PWM3OFL
DB8h	PWM3OFH
DB9h	PWM3TMRL
DBAh	PWM3TMRH
DBBh	PWM3CON
DBCh	PWM3INTE
DBDh	PWM3INTF
DBEh	PWM3CLKCON
DBFh	PWM3LDCON
DC0h	PWM3OFCON
DC1h	PWM4PHL
DC2h	PWM4PHH
DC3h	PWM4DCL
DC4h	PWM4DCH
DC5h	PWM4PRL
DC6h	PWM4PRH
DC7h	PWM4OFL
DC8h	PWM4OFH
DC9h	PWM4TMRL
DCAh	PWM4TMRH
DCBh	PWM4CON
DCCh	PWM4INTE
DCDh	PWM4INTF
DCEh	PWM4CLKCON
DCFh	PWM4LDCON
DD0h	PWM4OFCON
DD1h	
DEFh	—

Legend: = Unimplemented data memory locations, read as '0'.

TABLE 3-12: PIC16(L)F1574/5/8/9 MEMORY MAP, BANK 28-29

	Bank 28		Bank 29
E0Ch	—	E8Ch	—
E0Dh	—	E8Dh	—
E0Eh	—	E8Eh	—
E0Fh	PPSLOCK	E8Fh	—
E10h	INTPPS	E90h	RA0PPS
E11h	T0CKIPPS	E91h	RA1PPS
E12h	T1CKIPPS	E92h	RA2PPS
E13h	T1GPPS	E93h	—
E14h	CWG1PPS	E94h	RA4PPS
E15h	RXPPS	E95h	RA5PPS
E16h	CKPPS	E96h	—
E17h	ADCACTPPS	E97h	—
E18h	—	E98h	—
E19h	—	E99h	—
E1Ah	—	E9Ah	—
E1Bh	—	E9Bh	—
E1Ch	—	E9Ch	RB4PPS(1)
E1Dh	—	E9Dh	RB5PPS(1)
E1Eh	—	E9Eh	RB6PPS(1)
E1Fh	—	E9Fh	RB7PPS(1)
E20h	—	EA0h	RC0PPS
E21h	—	EA1h	RC1PPS
E22h	—	EA2h	RC2PPS
E23h	—	EA3h	RC3PPS
E24h	—	EA4h	RC4PPS
E25h	—	EA5h	RC5PPS
E26h	—	EA6h	RC6PPS(1)
E27h	—	EA7h	RC7PPS(1)
E28h	—	EA8h	—
E29h	—	EA9h	—
E2Ah	—	EAAh	—
E2Bh	—	EABh	—
E2Ch	—	EACh	—
E2Dh	—	EADh	—
E2Eh	—	EAEh	—
E2Fh	—	EAFh	—
E30h	—	EB0h	—
E31h	—	EB1h	—
E32h	—	EB2h	—
E33h	—	EB3h	—
E34h	—	EB4h	—
E35h	—	EB5h	—
E36h	—	EB6h	—
E37h	—	EB7h	—
E38h	—	EB8h	—
E39h	—	EB9h	—
E3Ah	—	EBAh	—
E3Bh	—	EBBh	—
E3Ch	—	EBCh	—
E3Dh	—	EBDh	—
E3Eh	—	EBEh	—
E3Fh	—	EBFh	—
E40h	—	EC0h	—
E6Fh		EEFh	

Legend: = Unimplemented data memory locations, read as '0'

Note 1: Unimplemented on PIC16(L)F1574/5.

TABLE 3-15: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

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
Bank 2											
10Ch	LATA	—	—	LATA5	LATA4	—	LATA2	LATA1	LATA0	--xx -xxx	--uu -uuu
10Dh	LATB(1)	LATB7	LATB6	LATB5	LATB4	—	—	—	—	xxxx ----	xxxx ----
10Eh	LATC	LATC7(1)	LATC6(1)	LATC5	LATC4	LATC3	LATC2	LATC1	LATC0	xxxx xxxx	xxxx xxxx
10Fh	—	Unimplemented								—	—
110h	—	Unimplemented								—	—
111h	CM1CON0	C1ON	C1OUT	—	C1POL	—	C1SP	C1HYS	C1SYNC	00-0 -100	00-0 -100
112h	CM1CON1	C1INTP	C1INTN	C1PCH<1:0>		—	C1NCH<2:0>			0000 -000	0000 -000
113h	CM2CON0	C2ON	C2OUT	—	C2POL	—	C2SP	C2HYS	C2SYNC	00-0 -100	00-0 -100
114h	CM2CON1	C2INTP	C2INTN	C2PCH<1:0>		—	C2NCH<2:0>			0000 -000	0000 -000
115h	CMOUT	—	—	—	—	—	—	MC2OUT	MC1OUT	---- --00	---- --00
116h	BORCON	SBOREN	BORFS	—	—	—	—	—	BORRDY	10-- ---q	uu-- ---u
117h	FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFVR<1:0>		ADFVR<1:0>		0q00 0000	0q00 0000
118h	DACCON0	DACEN	—	DACOE	—	DACPSS<1:0>		—	—	0-0- 00--	0-0- 00--
119h	DACCON1	—	—	—	DACR<4:0>					---0 0000	---0 0000
11Ah to 11Fh	—	Unimplemented								—	—

Legend: x = unknown, u = unchanged, q = value depends on condition, - = unimplemented, r = reserved. Shaded locations are unimplemented, read as '0'.

Note 1: PIC16(L)F1578/9 only.
2: PIC16F1574/5/8/9 only.
3: Unimplemented, read as '1'.

4.0 DEVICE CONFIGURATION

Device configuration consists of Configuration Words, Code Protection and Device ID.

4.1 Configuration Words

There are several Configuration Word bits that allow different oscillator and memory protection options. These are implemented as Configuration Word 1 at 8007h and Configuration Word 2 at 8008h.

Note:	The $\overline{\text{DEBUG}}$ bit in Configuration Words is managed automatically by device development tools including debuggers and programmers. For normal device operation, this bit should be maintained as a '1'.

REGISTER 4-2: CONFIGURATION WORD 2

R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
LVP(1)	$\overline{\text{DEBUG}}$(2)	$\overline{\text{LPBOREN}}$	BORV(3)	STVREN	PLLEN
bit 13					bit 8

U-1	U-1	U-1	U-1	U-1	R/P-1	R/P-1	R/P-1
—	—	—	—	—	PPS1WAY	WRT<1:0>	
bit 7							bit 0

Legend:		
R = Readable bit	P = Programmable bit	U = Unimplemented bit, read as '1'
'0' = Bit is cleared	'1' = Bit is set	n = Value when blank or after Bulk Erase

bit 13 **LVP:** Low-Voltage Programming Enable bit(1)
- 1 = ON – Low-voltage programming enabled. $\overline{\text{MCLR}}$/VPP pin function is $\overline{\text{MCLR}}$. MCLRE Configuration bit is ignored.
- 0 = OFF – High Voltage on $\overline{\text{MCLR}}$/VPP must be used for programming

bit 12 **$\overline{\text{DEBUG}}$:** Debugger Mode bit(2)
- 1 = OFF – In-Circuit Debugger disabled; ICSPCLK and ICSPDAT are general purpose I/O pins.
- 0 = ON – In-Circuit Debugger enabled; ICSPCLK and ICSPDAT are dedicated to the debugger.

bit 11 **$\overline{\text{LPBOREN}}$:** Low-Power Brown-out Reset Enable bit
- 1 = OFF – Low-power Brown-out Reset is disabled
- 0 = ON – Low-power Brown-out Reset is enabled

bit 10 **BORV:** Brown-out Reset Voltage Selection bit(3)
- 1 = LOW – Brown-out Reset voltage (VBOR), low trip point selected
- 0 = HIGH – Brown-out Reset voltage (VBOR), high trip point selected

bit 9 **STVREN:** Stack Overflow/Underflow Reset Enable bit
- 1 = ON – Stack Overflow or Underflow will cause a Reset
- 0 = OFF – Stack Overflow or Underflow will not cause a Reset

bit 8 **PLLEN:** PLL Enable bit
- 1 = ON – 4xPLL enabled
- 0 = OFF – 4xPLL disabled

bit 7-3 **Unimplemented:** Read as '1'

bit 2 **PPS1WAY**: PPSLOCK Bit One-Way Set Enable bit
- 1 = ON The PPSLOCK bit can only be set once after an unlocking sequence is executed; once PPSLOCK is set, all future changes to PPS registers are prevented
- 0 = OFF The PPSLOCK bit can be set and cleared as needed (provided an unlocking sequence is executed)

Note 1: This bit cannot be programmed to '0' when programming mode is entered via LVP.

2: The $\overline{\text{DEBUG}}$ bit in Configuration Words is managed automatically by device development tools including debuggers and programmers. For normal device operation, this bit should be maintained as a '1'.

3: See VBOR parameter for specific trip point voltages.

5.0 OSCILLATOR MODULE

5.1 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 5-1 illustrates a block diagram of the oscillator module.

Clock sources can be supplied from external logic level clocks. 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.

The oscillator module can be configured in one of the following 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. INTOSC – Internal oscillator (31 kHz to 32 MHz).

Clock Source modes are selected by the FOSC<1:0> bits in the Configuration Words. The FOSC bits determine the type of oscillator that will be used when the device is first powered.

The ECH, ECM, and ECL clock modes rely on an external logic level signal as the device clock source.

The INTOSC internal oscillator block produces low, medium, and high-frequency clock sources, designated LFINTOSC, MFINTOSC and HFINTOSC. (see Internal Oscillator Block, Figure 5-1). A wide selection of device clock frequencies may be derived from these three clock sources.

12.8 Register Definitions: PPS Input Selection

REGISTER 12-1: xxxPPS: PERIPHERAL xxx INPUT SELECTION

U-0	U-0	U-0	R/W-q/u	R/W-q/u	R/W-q/u	R/W-q/u	R/W-q/u
—	—	—	xxxPPS<4:0>				
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 peripheral

bit 7-5 **Unimplemented:** Read as ‘0’

bit 4-3 **xxxPPS<4:3>:** Peripheral xxx Input PORT Selection bits
11 = Reserved. Do not use.
10 = Peripheral input is PORTC
01 = Peripheral input is PORTB(2)
00 = Peripheral input is PORTA

bit 2-0 **xxxPPS<2:0>:** Peripheral xxx Input Bit Selection bits (1)
111 = Peripheral input is from PORTx Bit 7 (Rx7)
110 = Peripheral input is from PORTx Bit 6 (Rx6)
101 = Peripheral input is from PORTx Bit 5 (Rx5)
100 = Peripheral input is from PORTx Bit 4 (Rx4)
011 = Peripheral input is from PORTx Bit 3 (Rx3)
010 = Peripheral input is from PORTx Bit 2 (Rx2)
001 = Peripheral input is from PORTx Bit 1 (Rx1)
000 = Peripheral input is from PORTx Bit 0 (Rx0)

Note 1: See Table 12-1 for xxxPPS register list and Reset values.
2: PIC16(L)F1578/9 only.

REGISTER 12-2: RxyPPS: PIN Rxy OUTPUT SOURCE SELECTION REGISTER

U-0	U-0	U-0	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u
—	—	—	RxyPPS<4:0>				
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	

bit 7-5 **Unimplemented:** Read as ‘0’

bit 4-0 **RxyPPS<4:0>:** Pin Rxy Output Source Selection bits
Selection code determines the output signal on the port pin.
See Table 12-2 for the selection codes

REGISTER 16-6: ADRESH: ADC RESULT REGISTER HIGH (ADRESH) ADFM = 1

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
—	—	—	—	—	—	ADRES<9:8>	
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 7-2 **Reserved**: Do not use.

bit 1-0 **ADRES<9:8>**: ADC Result Register bits
Upper two bits of 10-bit conversion result

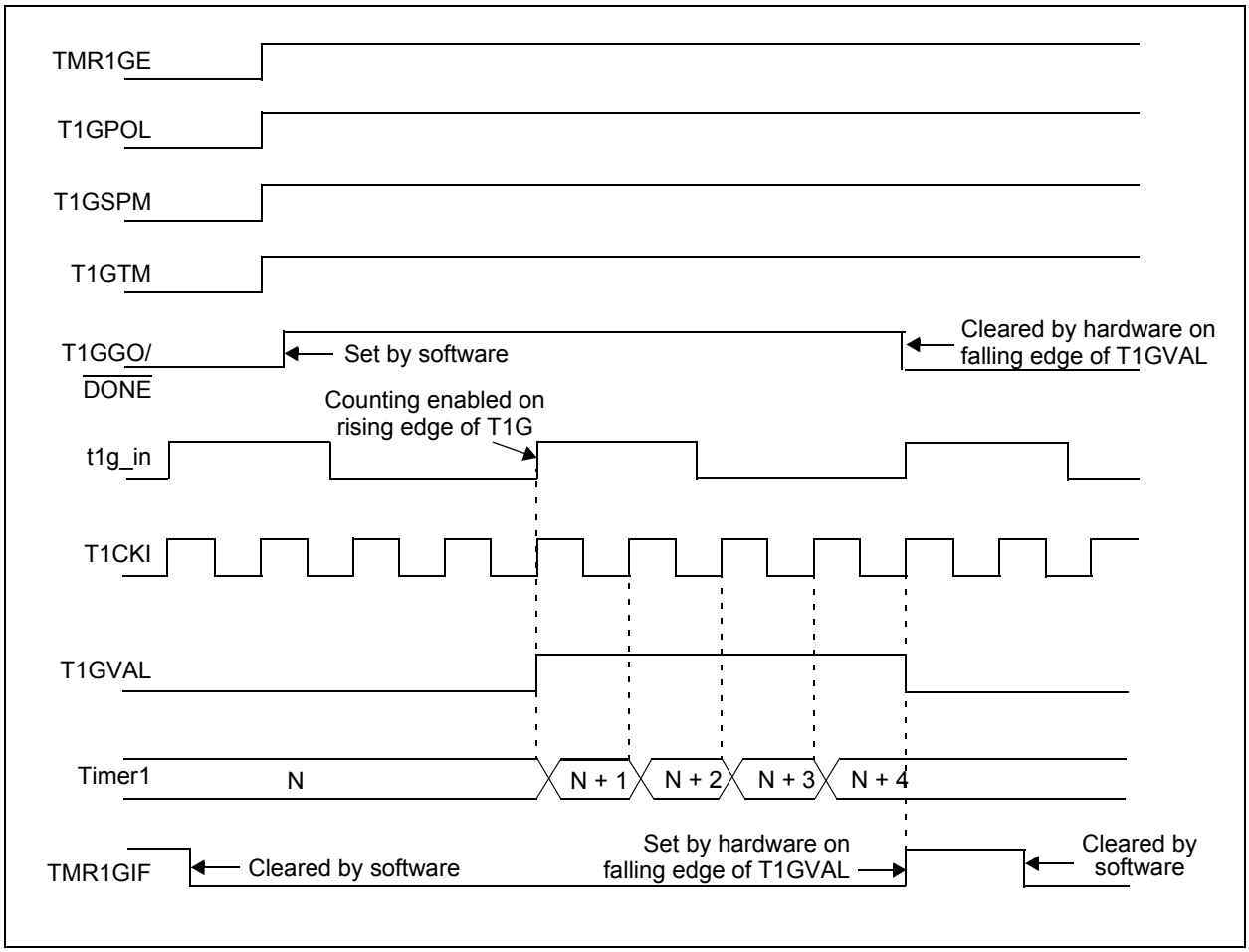
REGISTER 16-7: ADRESL: ADC RESULT REGISTER LOW (ADRESL) ADFM = 1

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
ADRES<7:0>							
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	

bit 7-0 **ADRES<7:0>**: ADC Result Register bits
Lower eight bits of 10-bit conversion result

FIGURE 20-6: TIMER1 GATE SINGLE-PULSE AND TOGGLE COMBINED MODE



21.1 Timer2 Operation

The clock input to the Timer2 module is the system instruction clock (FOSC/4).

TMR2 increments from 00h on each clock edge.

A 4-bit counter/prescaler on the clock input allows direct input, divide-by-4 and divide-by-16 prescale options. These options are selected by the prescaler control bits, T2CKPS<1:0> of the T2CON register. The value of TMR2 is compared to that of the Period register, PR2, on each clock cycle. When the two values match, the comparator generates a match signal as the timer output. This signal also resets the value of TMR2 to 00h on the next cycle and drives the output counter/ postscaler (see **Section 21.2 "Timer2 Interrupt"**).

The TMR2 and PR2 registers are both directly readable and writable. The TMR2 register is cleared on any device Reset, whereas the PR2 register initializes to FFh. Both the prescaler and postscaler counters are cleared on the following events:

- a write to the TMR2 register
- a write to the T2CON register
- Power-On Reset (POR)
- Brown-Out Reset (BOR)
- $\overline{\text{MCLR}}$ Reset
- Watchdog Timer (WDT) Reset
- Stack Overflow Reset
- Stack Underflow Reset
- RESET Instruction

Note:	TMR2 is not cleared when T2CON is written.

21.2 Timer2 Interrupt

Timer2 can also generate an optional device interrupt. The Timer2 output signal (T2_match) provides the input for the 4-bit counter/postscaler. This counter generates the TMR2 match interrupt flag which is latched in TMR2IF of the PIR1 register. The interrupt is enabled by setting the TMR2 Match Interrupt Enable bit, TMR2IE of the PIE1 register.

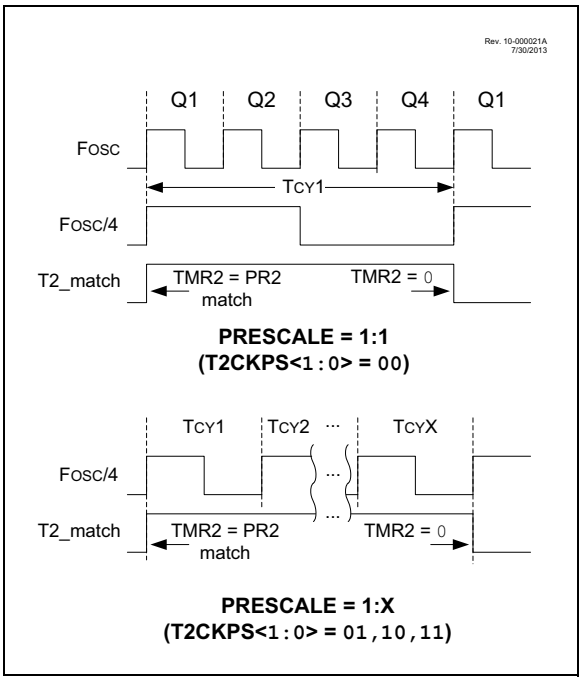
A range of 16 postscale options (from 1:1 through 1:16 inclusive) can be selected with the postscaler control bits, T2OUTPS<3:0>, of the T2CON register.

21.3 Timer2 Output

The output of TMR2 is T2_match.

The T2_match signal is synchronous with the system clock. Figure 21-3 shows two examples of the timing of the T2_match signal relative to FOSC and prescale value, T2CKPS<1:0>. The upper diagram illustrates 1:1 prescale timing and the lower diagram, 1:X prescale timing.

FIGURE 21-3: T2_MATCH TIMING DIAGRAM



21.4 Timer2 Operation During Sleep

Timer2 cannot be operated while the processor is in Sleep mode. The contents of the TMR2 and PR2 registers will remain unchanged while the processor is in Sleep mode.

22.1 EUSART Asynchronous Mode

The EUSART transmits and receives data using the standard non-return-to-zero (NRZ) format. NRZ is implemented with two levels: a VOH Mark state which represents a '1' data bit, and a VOL Space state which represents a '0' data bit. NRZ refers to the fact that consecutively transmitted data bits of the same value stay at the output level of that bit without returning to a neutral level between each bit transmission. An NRZ transmission port idles in the Mark state. Each character transmission consists of one Start bit followed by eight or nine data bits and is always terminated by one or more Stop bits. The Start bit is always a space and the Stop bits are always marks. The most common data format is eight bits. Each transmitted bit persists for a period of 1/(Baud Rate). An on-chip dedicated 8-bit/16-bit Baud Rate Generator is used to derive standard baud rate frequencies from the system oscillator. See Table 22-5 for examples of baud rate configurations.

The EUSART transmits and receives the LSb first. The EUSART's transmitter and receiver are functionally independent, but share the same data format and baud rate. Parity is not supported by the hardware, but can be implemented in software and stored as the ninth data bit.

22.1.1 EUSART ASYNCHRONOUS TRANSMITTER

The EUSART transmitter block diagram is shown in Figure 22-1. The heart of the transmitter is the serial Transmit Shift Register (TSR), which is not directly accessible by software. The TSR obtains its data from the transmit buffer, which is the TXREG register.

22.1.1.1 Enabling the Transmitter

The EUSART transmitter is enabled for asynchronous operations by configuring the following three control bits:

- TXEN = 1
- SYNC = 0
- SPEN = 1

All other EUSART control bits are assumed to be in their default state.

Setting the TXEN bit of the TXSTA register enables the transmitter circuitry of the EUSART. Clearing the SYNC bit of the TXSTA register configures the EUSART for asynchronous operation. Setting the SPEN bit of the RCSTA register enables the EUSART and automatically configures the TX/CK I/O pin as an output. If the TX/CK pin is shared with an analog peripheral, the analog I/O function must be disabled by clearing the corresponding ANSEL bit.

Note:	The TXIF Transmitter Interrupt flag is set when the TXEN enable bit is set.

22.1.1.2 Transmitting Data

A transmission is initiated by writing a character to the TXREG register. If this is the first character, or the previous character has been completely flushed from the TSR, the data in the TXREG is immediately transferred to the TSR register. If the TSR still contains all or part of a previous character, the new character data is held in the TXREG until the Stop bit of the previous character has been transmitted. The pending character in the TXREG is then transferred to the TSR in one TCY immediately following the Stop bit transmission. The transmission of the Start bit, data bits and Stop bit sequence commences immediately following the transfer of the data to the TSR from the TXREG.

22.1.1.3 Transmit Data Polarity

The polarity of the transmit data can be controlled with the SCKP bit of the BAUDCON register. The default state of this bit is '0' which selects high true transmit idle and data bits. Setting the SCKP bit to '1' will invert the transmit data resulting in low true idle and data bits. The SCKP bit controls transmit data polarity in Asynchronous mode only. In Synchronous mode, the SCKP bit has a different function. See **Section 22.5.1.2 "Clock Polarity"**.

22.1.1.4 Transmit Interrupt Flag

The TXIF interrupt flag bit of the PIR1 register is set whenever the EUSART transmitter is enabled and no character is being held for transmission in the TXREG. In other words, the TXIF bit is only clear when the TSR is busy with a character and a new character has been queued for transmission in the TXREG. The TXIF flag bit is not cleared immediately upon writing TXREG. TXIF becomes valid in the second instruction cycle following the write execution. Polling TXIF immediately following the TXREG write will return invalid results. The TXIF bit is read-only, it cannot be set or cleared by software.

The TXIF interrupt can be enabled by setting the TXIE interrupt enable bit of the PIE1 register. However, the TXIF flag bit will be set whenever the TXREG is empty, regardless of the state of TXIE enable bit.

To use interrupts when transmitting data, set the TXIE bit only when there is more data to send. Clear the TXIE interrupt enable bit upon writing the last character of the transmission to the TXREG.

22.5.2.3 EUSART Synchronous Slave Reception

The operation of the Synchronous Master and Slave modes is identical (**Section 22.5.1.5 "Synchronous Master Reception"**), with the following exceptions:

- Sleep
- CREN bit is always set, therefore the receiver is never idle
- SREN bit, which is a "don't care" in Slave mode

A character may be received while in Sleep mode by setting the CREN bit prior to entering Sleep. Once the word is received, the RSR register will transfer the data to the RCREG register. If the RCIE enable bit is set, the interrupt generated will wake the device from Sleep and execute the next instruction. If the GIE bit is also set, the program will branch to the interrupt vector.

22.5.2.4 Synchronous Slave Reception Set-up:

1. Set the SYNC and SPEN bits and clear the CSRC bit.
2. Clear the ANSEL bit for both the CK and DT pins (if applicable).
3. If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
4. If 9-bit reception is desired, set the RX9 bit.
5. Set the CREN bit to enable reception.
6. The RCIF bit will be set when reception is complete. An interrupt will be generated if the RCIE bit was set.
7. If 9-bit mode is enabled, retrieve the Most Significant bit from the RX9D bit of the RCSTA register.
8. Retrieve the eight Least Significant bits from the receive FIFO by reading the RCREG register.
9. If an overrun error occurs, clear the error by either clearing the CREN bit of the RCSTA register or by clearing the SPEN bit which resets the EUSART.

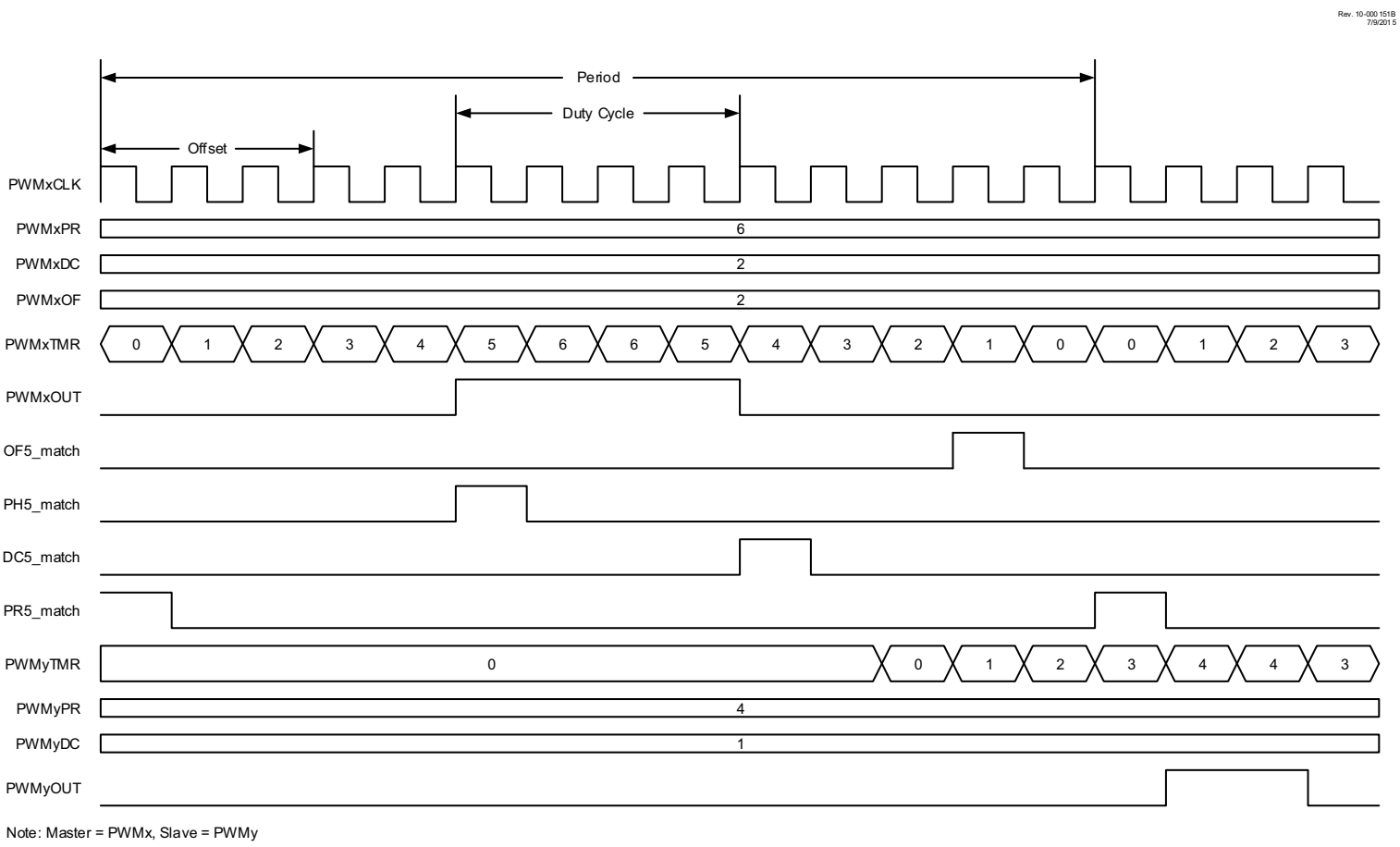
TABLE 22-10: SUMMARY OF REGISTERS ASSOCIATED WITH SYNCHRONOUS SLAVE RECEPTION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
BAUDCON	ABDOVF	RCIDL	—	SCKP	BRG16	—	WUE	ABDEN	204
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	86
PIE1	TMR1GIE	ADIE	RCIE	TXIE	—	—	TMR2IE	TMR1IE	87
PIR1	TMR1GIF	ADIF	RCIF	TXIF	—	—	TMR2IF	TMR1IF	90
RCREG	EUSART Receive Data Register								197*
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	203
TXSTA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	202

Legend: — = unimplemented location, read as '0'. Shaded cells are not used for synchronous slave reception.

* Page provides register information.

FIGURE 23-13: OFFSET MATCH ON DECREMENTING TIMER TIMING DIAGRAM



REGISTER 23-11: PWMxPRH: PWMx PERIOD COUNT HIGH REGISTER

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
PR<15:8>							
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 7-0 **PR<15:8>**: PWM Period High bits
Upper eight bits of PWM period count

REGISTER 23-12: PWMxPRL: PWMx PERIOD COUNT LOW REGISTER

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
PR<7:0>							
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	

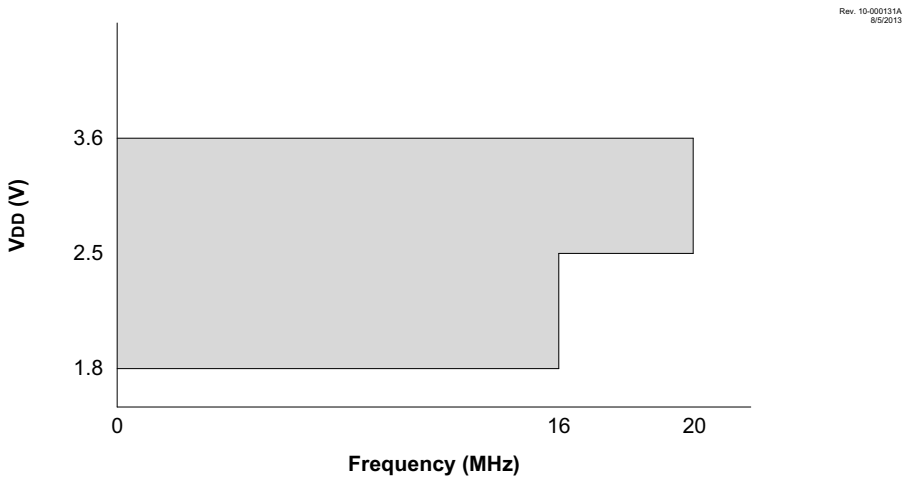
bit 7-0 **PR<7:0>**: PWM Period Low bits
Lower eight bits of PWM period count

TABLE 23-2: SUMMARY OF REGISTERS ASSOCIATED WITH PWM

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
OSCCON	SPLLEN	IRCF<3:0>				—	SCS<1:0>		69
PIE3	PWM4IE	PWM3IE	PWM2IE	PWM1IE	—	—	—	—	89
PIR3	PWM4IF	PWM3IF	PWM2IF	PWM1IF	—	—	—	—	92
PWMEN	—	—	—	—	PWM4EN_A	PWM3EN_A	PWM2EN_A	PWM1EN_A	243
PWMLD	—	—	—	—	PWM4LDA_A	PWM3LDA_A	PWM2LDA_A	PWM1LDA_A	243
PWMOUT	—	—	—	—	PWM4OUT_A	PWM3OUT_A	PWM2OUT_A	PWM1OUT_A	243
PWM1PHL	PH<7:0>								238
PWM1PHH	PH<15:8>								238
PWM1DCL	DC<7:0>								239
PWM1DCH	DC<15:8>								239
PWM1PRL	PR<7:0>								240
PWM1PRH	PR<15:8>								240
PWM1OFL	OF<7:0>								241
PWM1OFH	OF<15:8>								241
PWM1TMRL	TMR<7:0>								242
PWM1TMRH	TMR<15:8>								242
PWM1CON	EN	—	OUT	POL	MODE<1:0>		—	—	233
PWM1INTE	—	—	—	—	OFIE	PHIE	DCIE	PRIE	234
PWM1INTF	—	—	—	—	OFIF	PHIF	DCIF	PRIF	234
PWM1CLKCON	—	PS<2:0>			—	—	CS<1:0>		235
PWM1LDCON	LDA	LDT	—	—	—	—	LDS<1:0>		236
PWM1OFCON	—	OFM<1:0>		OFO	—	—	OFS<1:0>		237
PWM2PHL	PH<7:0>								238
PWM2PHH	PH<15:8>								238
PWM2DCL	DC<7:0>								239
PWM2DCH	DC<15:8>								239
PWM2PRL	PR<7:0>								240
PWM2PRH	PR<15:8>								240
PWM2OFL	OF<7:0>								241
PWM2OFH	OF<15:8>								241
PWM2TMRL	TMR<7:0>								242
PWM2TMRH	TMR<15:8>								242
PWM2CON	EN	—	OUT	POL	MODE<1:0>		—	—	233
PWM2INTE	—	—	—	—	OFIE	PHIE	DCIE	PRIE	234
PWM2INTF	—	—	—	—	OFIF	PHIF	DCIF	PRIF	234
PWM2CLKCON	—	PS<2:0>			—	—	CS<1:0>		235
PWM2LDCON	LDA	LDT	—	—	—	—	LDS<1:0>		236
PWM2OFCON	—	OFM<1:0>		OFO	—	—	OFS<1:0>		237
PWM3PHL	PH<7:0>								238
PWM3PHH	PH<15:8>								238
PWM3DCL	DC<7:0>								239
PWM3DCH	DC<15:8>								239
PWM3PRL	PR<7:0>								240
PWM3PRH	PR<15:8>								240
PWM3OFL	OF<7:0>								241
PWM3OFH	OF<15:8>								241
PWM3TMRL	TMR<7:0>								242
PWM3TMRH	TMR<15:8>								242
PWM3CON	EN	—	OUT	POL	MODE<1:0>		—	—	233
PWM3INTE	—	—	—	—	OFIE	PHIE	DCIE	PRIE	234
PWM3INTF	—	—	—	—	OFIF	PHIF	DCIF	PRIF	234

Legend: — = unimplemented location, read as '0'. Shaded cells are not used by PWM.

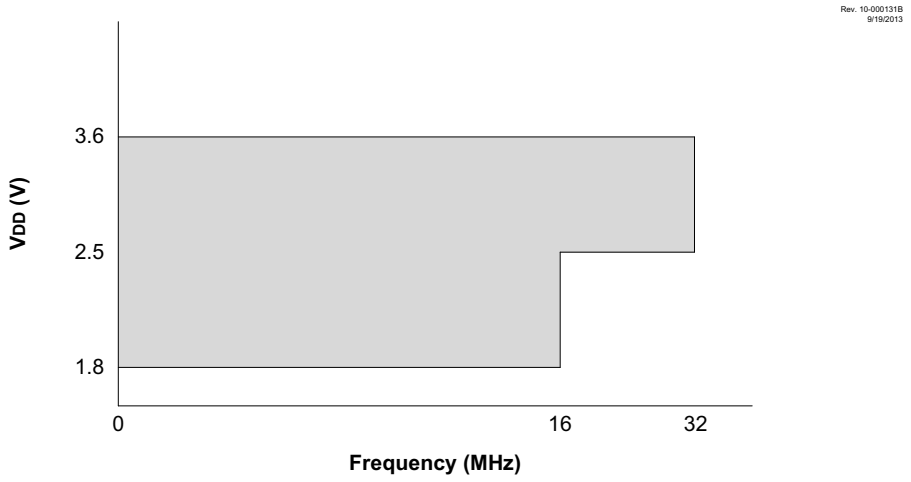
FIGURE 27-1: VOLTAGE FREQUENCY GRAPH, -40°C ≤ TA ≤ +125°C, PIC16F1574/5/8/9 ONLY



Note 1: The shaded region indicates the permissible combinations of voltage and frequency.

2: Refer to Table 27-7 for each Oscillator mode's supported frequencies.

FIGURE 27-2: VOLTAGE FREQUENCY GRAPH, -40°C ≤ TA ≤ +125°C, PIC16LF1574/5/8/9 ONLY



Note 1: The shaded region indicates the permissible combinations of voltage and frequency.

2: Refer to Table 27-7 for each Oscillator mode's supported frequencies.

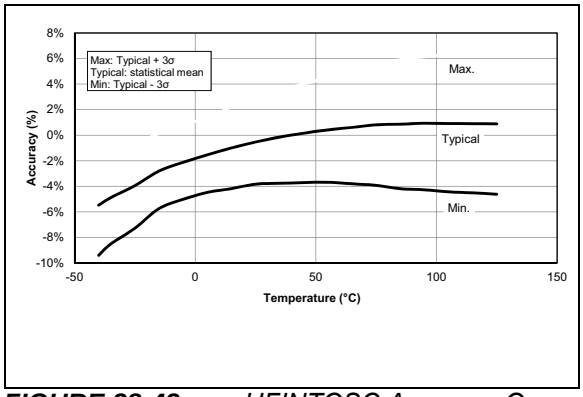


FIGURE 28-43: *HFINTOSC Accuracy Over Temperature, VDD = 1.8V, LF Devices Only.*

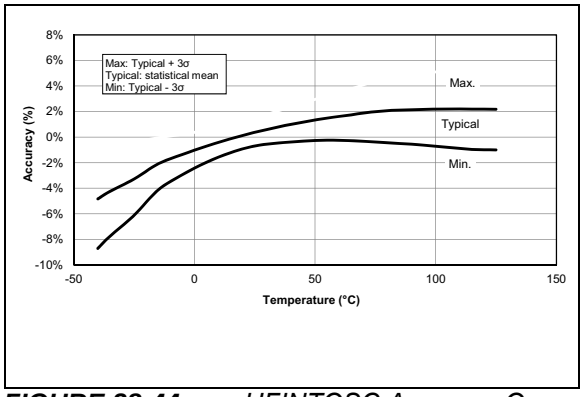


FIGURE 28-44: *HFINTOSC Accuracy Over Temperature, 2.3V ≤ VDD ≤ 5.5V.*

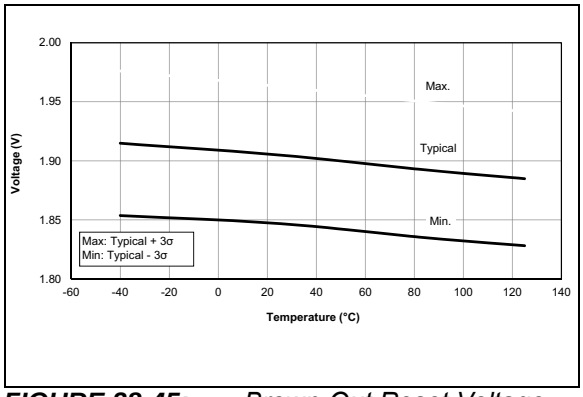


FIGURE 28-45: *Brown-Out Reset Voltage, BORV = 1, PIC16LF1574/5/8/9 Only.*

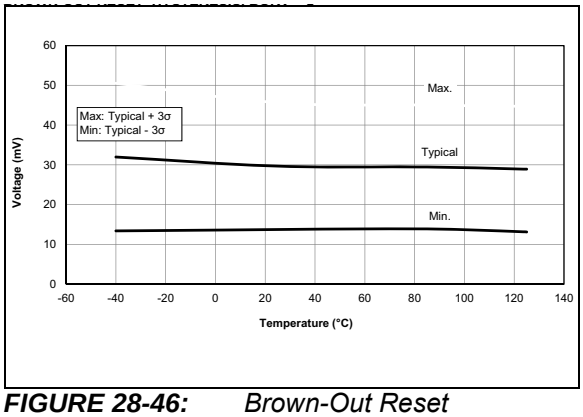


FIGURE 28-46: *Brown-Out Reset Hysteresis, BORV = 1, PIC16LF1574/5/8/9 Only.*

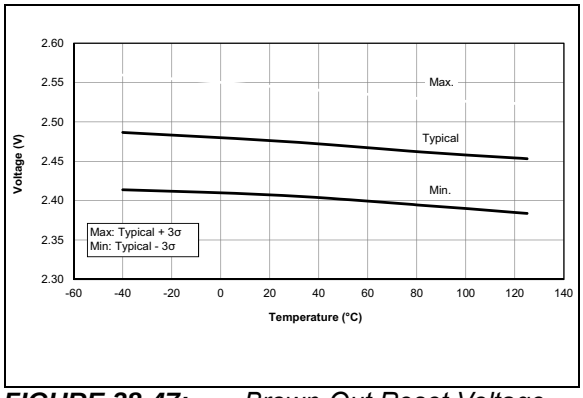


FIGURE 28-47: *Brown-Out Reset Voltage, BORV = 1, PIC16F1574/5/8/9 Only.*

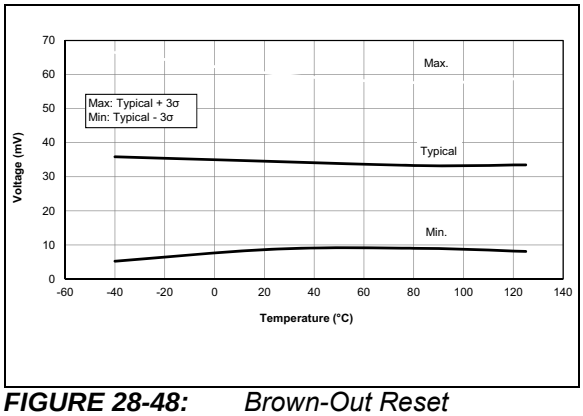


FIGURE 28-48: *Brown-Out Reset Hysteresis, BORV = 1, PIC16F1574/5/8/9 Only.*

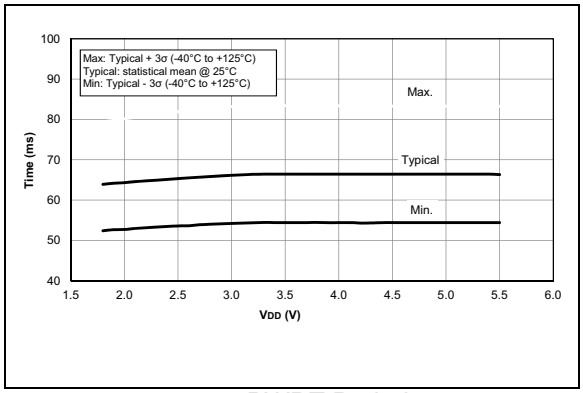


FIGURE 28-55: *PWRT Period.*

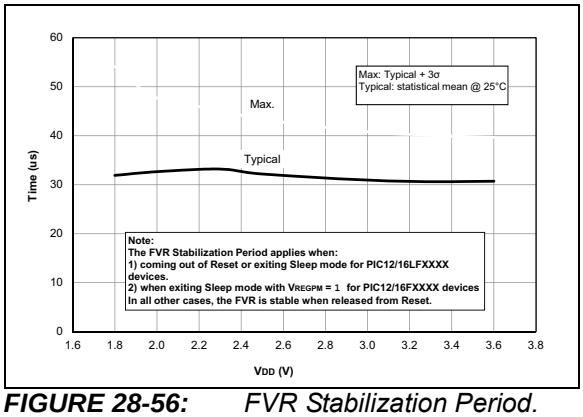


FIGURE 28-56: *FVR Stabilization Period.*

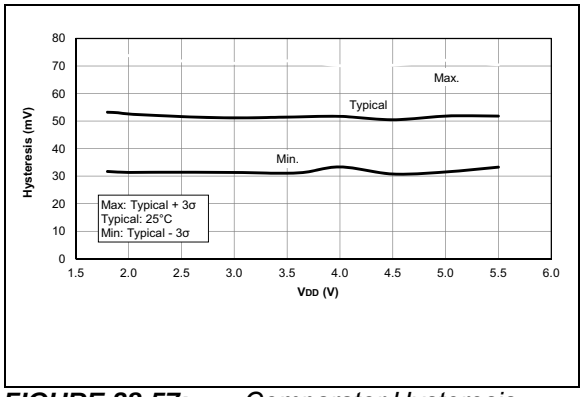


FIGURE 28-57: *Comparator Hysteresis, Normal Power Mode (CxSP = 1, CxHYS = 1).*

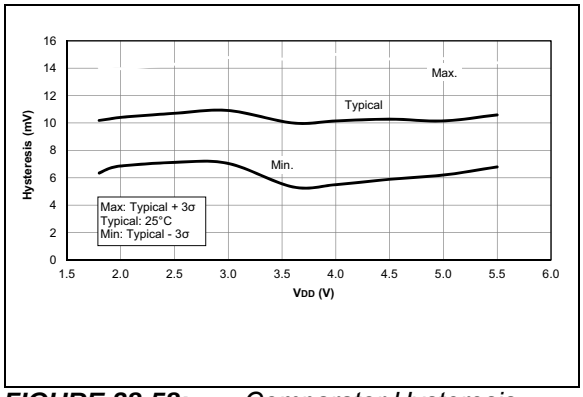


FIGURE 28-58: *Comparator Hysteresis, Low-Power Mode (CxSP = 0, CxHYS = 1).*

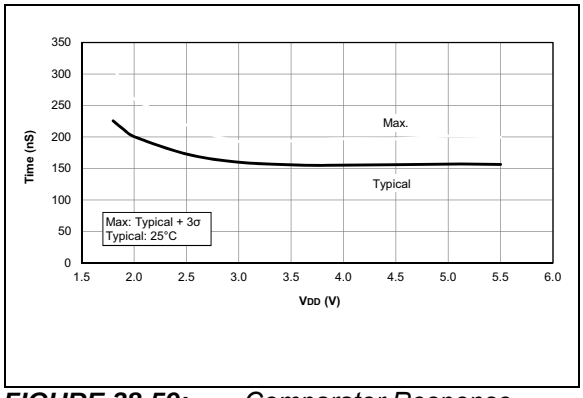


FIGURE 28-59: *Comparator Response Time, Normal Power Mode, (CxSP = 1).*

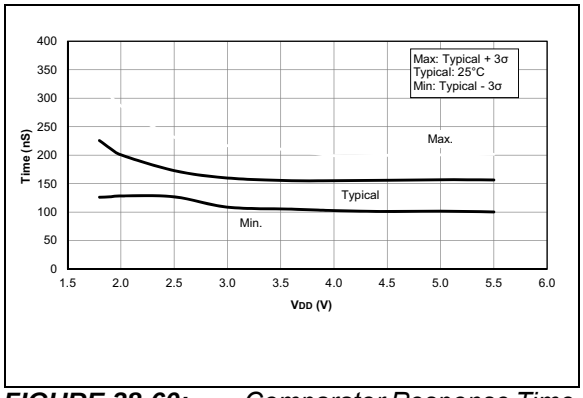
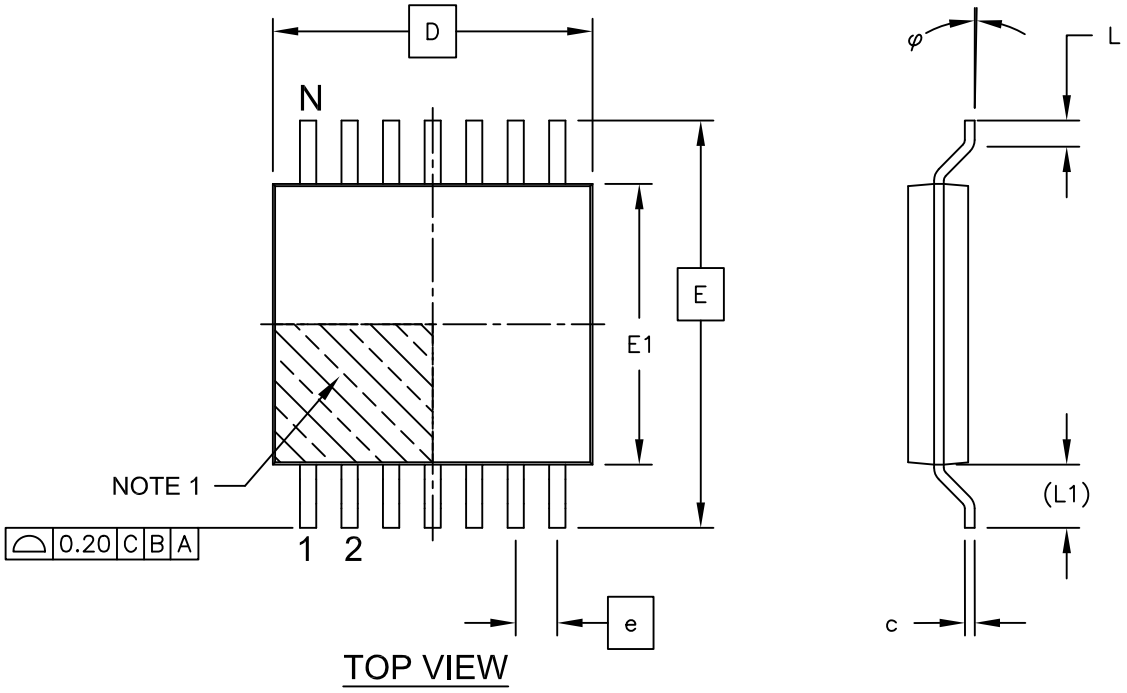


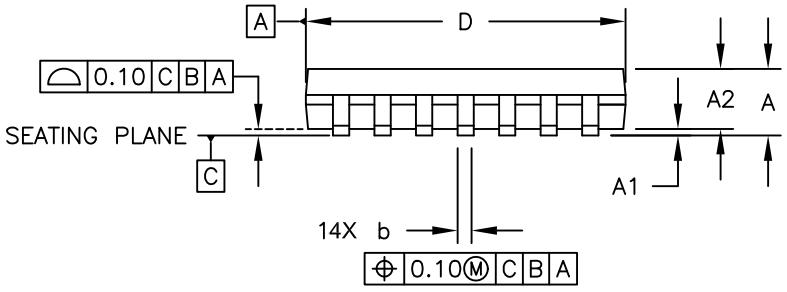
FIGURE 28-60: *Comparator Response Time Over Temperature, Normal Power Mode, (CxSP = 1).*

14-Lead Plastic Thin Shrink Small Outline (ST) - 4.4 mm Body [TSSOP]

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



TOP VIEW



SIDE VIEW

Microchip Technology Drawing C04-087C Sheet 1 of 2