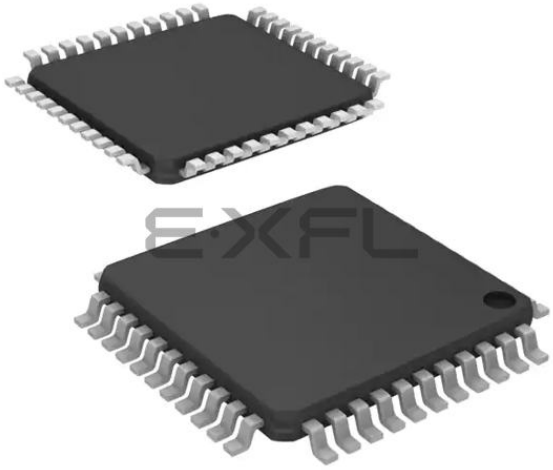


Welcome to E-XFL.COM

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	I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	36
Program Memory Size	14KB (8K x 14)
Program Memory Type	FLASH
EEPROM Size	128 x 8
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	2.3V ~ 5.5V
Data Converters	A/D 28x10b; D/A 4x5b, 4x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-TQFP
Supplier Device Package	44-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f1777-i-pt

TABLE 3-18: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Addr	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 5											
28Ch	ODCONA	ODA7	ODA6	ODA5	ODA4	ODA3	ODA2	ODA1	ODA0	0000 0000	0000 0000
28Dh	ODCONB	ODB7	ODB6	ODB5	ODB4	ODB3	ODB2	ODB1	ODB0	0000 0000	0000 0000
28Eh	ODCONC	ODC7	ODC6	ODC5	ODC4	ODC3	ODC2	ODC1	ODC0	0000 0000	0000 0000
28Fh	ODCOND(3)	ODD7	ODD6	ODD5	ODD4	ODD3	ODD2	ODD1	ODD0	0000 0000	0000 0000
290h	ODCONE(3)	—	—	—	—	—	ODE2	ODE1	ODE0	---- -000	--- -000
291h	CCPR1L	Capture/Compare/PWM Register 1 (LSB)								xxxx xxxx	uuuu uuuu
292h	CCPR1H	Capture/Compare/PWM Register 1 (MSB)								xxxx xxxx	uuuu uuuu
293h	CCP1CON	EN	—	OUT	FMT	MODE<3:0>				0-00 0000	0-00 0000
294h	CCP1CAP	—	—	—	—	CTS<3:0>				---- 0000	---- 0000
295h	CCPR2L	Capture/Compare/PWM Register 2 (LSB)								xxxx xxxx	uuuu uuuu
296h	CCPR2H	Capture/Compare/PWM Register 2 (MSB)								xxxx xxxx	uuuu uuuu
297h	CCP2CON	EN	—	OUT	FMT	MODE<3:0>				0-00 0000	0-00 0000
298h	CCP2CAP	—	—	—	—	CTS<3:0>				---- 0000	---- 0000
299h	CCPR7L	Capture/Compare/PWM Register 7 (LSB)								xxxx xxxx	uuuu uuuu
29Ah	CCPR7H	Capture/Compare/PWM Register 7 (MSB)								xxxx xxxx	uuuu uuuu
29Bh	CCP7CON	EN	—	OUT	FMT	MODE<3:0>				0-00 0000	0-00 0000
29Ch	CCP7CAP	—	—	—	—	CTS<3:0>				---- 0000	---- 0000
29Dh	—	Unimplemented								—	—
29Eh	CCPTMRS1	C8TSEL<1:0>(3)		C7TSEL<1:0>		C2TSEL<1:0>		C1TSEL<1:0>		--00 0000	--00 0000
29Fh	CCPTMRS2	P10TSEL<1:0>(3)		P9TSEL<1:0>		P4TSEL<1:0>		P3TSEL<1:0>		--00 0000	--00 0000

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

Note 1: Unimplemented, read as '1'.
2: Unimplemented on PIC16LF1777/8/9.
3: Unimplemented on PIC16(L)F1778.

TABLE 3-18: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Addr	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 28											
E0Ch — E0Bh	—	Unimplemented								—	—
E0Ch	PPSLOCK	—	—	—	—	—	—	—	PPSLOCKED	---- ---0	---- ---0
E0Dh	INTPPS	—	—	INTPPS<5:0>						--00 1000	--uu uuuu
E0Eh	T0CKIPPS	—	—	T0CKIPPS<5:0>						--00 0100	--uu uuuu
E0Fh	T1CKIPPS	—	—	T1CKIPPS<5:0>						--01 0000	--uu uuuu
E10h	T1GPPS	—	—	T1GPPS<5:0>						--00 1101	--uu uuuu
E11h	T3CKIPPS	—	—	T3CKIPPS<5:0>						--01 0000	--uu uuuu
E12h	T3GPPS	—	—	T3GPPS<5:0>						--01 0000	--uu uuuu
E13h	T5CKIPPS	—	—	T5CKIPPS<5:0>						--01 0000	--uu uuuu
E14h	T5GPPS	—	—	T5GPPS<5:0>						--00 1100	--uu uuuu
E15h	T2INPPS	—	—	T2INPPS<5:0>						--01 0011	--uu uuuu
E16h	T4INPPS	—	—	T4INPPS<5:0>						--01 0101	--uu uuuu
E17h	T6INPPS	—	—	T6INPPS<5:0>						--01 0100	--uu uuuu
E18h	T8INPPS	—	—	T8INPPS<5:0>						--01 0010	--uu uuuu
E19h	CCP1PPS	—	—	CCP1PPS<5:0>						--01 0010	--uu uuuu
E1Ah	CCP2PPS	—	—	CCP2PPS<5:0>						--01 0001	--uu uuuu
E1Bh	CCP7PPS	—	—	CCP7PPS<5:0>						--00 1101	--uu uuuu
E1Ch	CCP8PPS(3)	—	—	CCP8PPS<5:0>						--00 1101	--uu uuuu
E1Dh	COGIN1PPS	—	—	COG1PPS<5:0>						--00 1000	--uu uuuu
E1Eh	COG2INPPS	—	—	COG2PPS<5:0>						--00 1001	--uu uuuu
E1Fh	COG3INPPS	—	—	COG3PPS<5:0>						--00 1010	--uu uuuu
E20h	COG4INPPS(3)	—	—	COG4PPS<5:0>						--00 1010	--uu uuuu
E21h	MD1CLPPS	—	—	MD1CLPPS<5:0>						--00 0100	--uu uuuu
E22h	MD1CHPPS	—	—	MD1CHPPS<5:0>						--00 0011	--uu uuuu
E23h	MD1MODPPS	—	—	MD1MODPPS<5:0>						--00 0101	--uu uuuu
E24h	MD2CLPPS	—	—	MD2CLPPS<5:0>						--00 0100	--uu uuuu
E25h	MD2CHPPS	—	—	MD2CHPPS<5:0>						--00 0011	--uu uuuu
E26h	MD2MODPPS	—	—	MD2MODPPS<5:0>						--00 0101	--uu uuuu

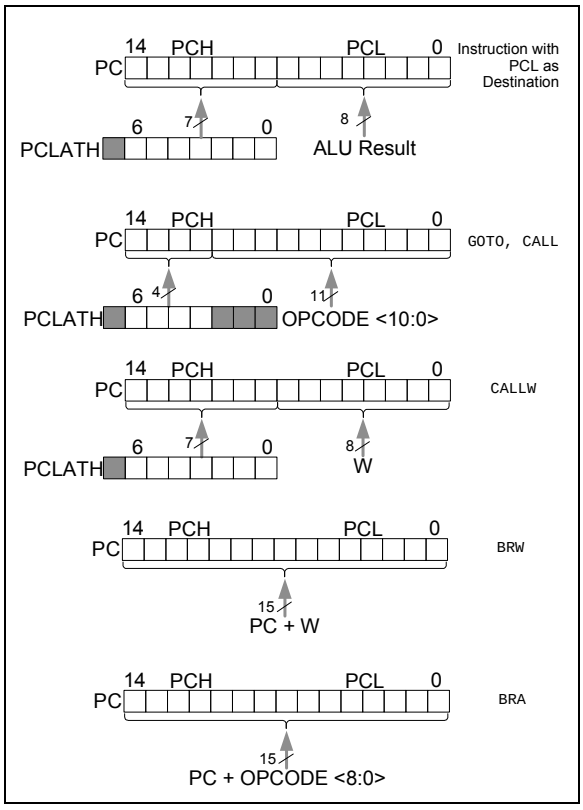
Legend: x = unknown, u = unchanged, q = value depends on condition, - = unimplemented, read as '0', r = reserved.
Shaded locations are unimplemented, read as '0'.

Note 1: Unimplemented, read as '1'.
2: Unimplemented on PIC16LF1777/8/9.
3: Unimplemented on PIC16(L)F1778.

3.5 PCL and PCLATH

The Program Counter (PC) is 15 bits wide. The low byte comes from the PCL register, which is a readable and writable register. The high byte (PC<14:8>) is not directly readable or writable and comes from PCLATH. On any Reset, the PC is cleared. Figure 3-4 shows the five situations for the loading of the PC.

FIGURE 3-4: LOADING OF PC IN DIFFERENT SITUATIONS



3.5.1 MODIFYING PCL

Executing any instruction with the PCL register as the destination simultaneously causes the Program Counter PC<14:8> bits (PCH) to be replaced by the contents of the PCLATH register. This allows the entire contents of the program counter to be changed by writing the desired upper seven bits to the PCLATH register. When the lower eight bits are written to the PCL register, all 15 bits of the program counter will change to the values contained in the PCLATH register and those being written to the PCL register.

3.5.2 COMPUTED GOTO

A computed GOTO is accomplished by adding an offset to the program counter (ADDWF PCL). When performing a table read using a computed GOTO method, care should be exercised if the table location crosses a PCL memory boundary (each 256-byte block). Refer to Application Note AN556, *"Implementing a Table Read"* (DS00556).

3.5.3 COMPUTED FUNCTION CALLS

A computed function CALL allows programs to maintain tables of functions and provide another way to execute state machines or look-up tables. When performing a table read using a computed function CALL, care should be exercised if the table location crosses a PCL memory boundary (each 256-byte block).

If using the CALL instruction, the PCH<2:0> and PCL registers are loaded with the operand of the CALL instruction. PCH<6:3> is loaded with PCLATH<6:3>.

The CALLW instruction enables computed calls by combining PCLATH and W to form the destination address. A computed CALLW is accomplished by loading the W register with the desired address and executing CALLW. The PCL register is loaded with the value of W and PCH is loaded with PCLATH.

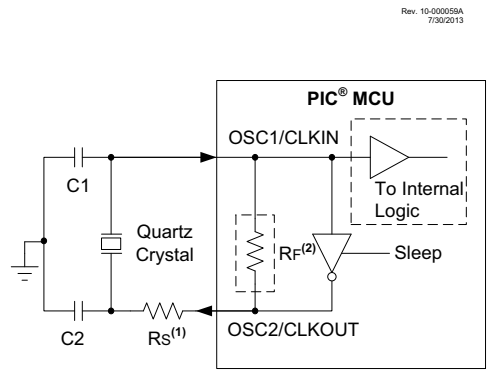
3.5.4 BRANCHING

The branching instructions add an offset to the PC. This allows relocatable code and code that crosses page boundaries. There are two forms of branching, BRW and BRA. The PC will have incremented to fetch the next instruction in both cases. When using either branching instruction, a PCL memory boundary may be crossed.

If using BRW, load the W register with the desired unsigned address and execute BRW. The entire PC will be loaded with the address PC + 1 + W.

If using BRA, the entire PC will be loaded with PC + 1 +, the signed value of the operand of the BRA instruction.

FIGURE 5-3: QUARTZ CRYSTAL OPERATION (LP, XT OR HS MODE)



Note 1: A series resistor (Rs) may be required for quartz crystals with low drive level.

2: The value of RF varies with the Oscillator mode selected (typically between 2 MΩ and 10 MΩ).

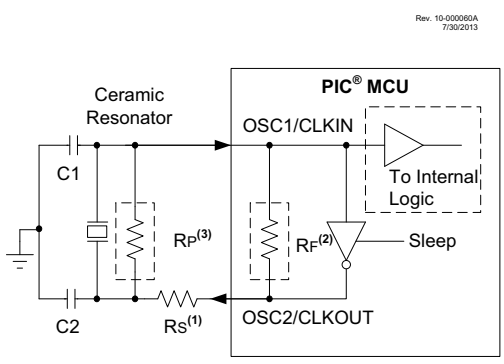
Note 1: Quartz crystal characteristics vary according to type, package and manufacturer. The user should consult the manufacturer data sheets for specifications and recommended application.

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 Application 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)

FIGURE 5-4: CERAMIC RESONATOR OPERATION (XT OR HS MODE)



Note 1: A series resistor (Rs) may be required for ceramic resonators with low drive level.

2: The value of RF varies with the Oscillator mode selected (typically between 2 MΩ and 10 MΩ).

3. An additional parallel feedback resistor (RP) may be required for proper ceramic resonator operation.

5.2.1.3 Oscillator Start-up Timer (OST)

If the oscillator module is configured for LP, XT or HS modes, the Oscillator Start-up Timer (OST) counts 1024 oscillations from OSC1. This occurs following a Power-on Reset (POR) and when the Power-up Timer (PWRT) has expired (if configured), or a wake-up from Sleep. During this time, the program counter does not increment and program execution is suspended, unless either FSCM or Two-Speed Start-Up are enabled. In this case, code will continue to execute at the selected INTOSC frequency while the OST is counting. The OST ensures that the oscillator circuit, using a quartz crystal resonator or ceramic resonator, has started and is providing a stable system clock to the oscillator module.

In order to minimize latency between external oscillator start-up and code execution, the Two-Speed Clock Start-up mode can be selected (see **Section 5.4 "Two-Speed Clock Start-up Mode"**).

REGISTER 7-8: PIR1: PERIPHERAL INTERRUPT REQUEST REGISTER 1

R/W-0/0	R/W-0/0	R-0/0	R-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF
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 **TMR1GIF:** Timer1 Gate Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

bit 6 **ADIF:** Analog-to-Digital Converter (ADC) Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

bit 5 **RCIF:** EUSART Receive Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

bit 4 **TXIF:** EUSART Transmit Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

bit 3 **SSP1IF:** Synchronous Serial Port (MSSP) Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

bit 2 **CCP1IF:** CCP1 Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

bit 1 **TMR2IF:** Timer2 to T2PR Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

bit 0 **TMR1IF:** Timer1 Overflow Interrupt Flag bit
1 = Interrupt is pending
0 = Interrupt is not pending

Note:	Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the Global Enable bit, GIE, of the INTCON register. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 12-3: PPSLOCK: PPS LOCK REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0/0
—	—	—	—	—	—	—	PPSLOCKED
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 7-1 **Unimplemented:** Read as ‘0’

bit 0 **PPSLOCKED:** PPS Locked bit

1 = PPS is locked. PPS selections can not be changed.

0 = PPS is not locked. PPS selections can be changed.

16.2 ADC Operation

16.2.1 STARTING A CONVERSION

To enable the ADC module, the ADON bit of the ADCON0 register must be set to a '1'. Setting the GO/$\overline{\text{DONE}}$ bit of the ADCON0 register to a '1' will start the Analog-to-Digital conversion.

Note:	The GO/$\overline{\text{DONE}}$ bit should not be set in the same instruction that turns on the ADC. Refer to **Section 16.2.6 "ADC Conversion Procedure"**.

16.2.2 COMPLETION OF A CONVERSION

When the conversion is complete, the ADC module will:

- Clear the GO/$\overline{\text{DONE}}$ bit
- Set the ADIF Interrupt Flag bit
- Update the ADRESH and ADRESL registers with new conversion result

16.2.3 TERMINATING A CONVERSION

If a conversion must be terminated before completion, the GO/$\overline{\text{DONE}}$ bit can be cleared in software. The ADRESH and ADRESL registers will be updated with the partially complete Analog-to-Digital conversion sample. Incomplete bits will match the last bit converted.

Note:	A device Reset forces all registers to their Reset state. Thus, the ADC module is turned off and any pending conversion is terminated.

16.2.4 ADC OPERATION DURING SLEEP

The ADC module can operate during Sleep. This requires the ADC clock source to be set to the FRC option. When the FRC oscillator source is selected, the ADC waits one additional instruction before starting the conversion. This allows the SLEEP instruction to be executed, which can reduce system noise during the conversion. If the ADC interrupt is enabled, the device will wake-up from Sleep when the conversion completes. If the ADC interrupt is disabled, the ADC module is turned off after the conversion completes, although the ADON bit remains set.

When the ADC clock source is something other than FRC, a SLEEP instruction causes the present conversion to be aborted and the ADC module is turned off, although the ADON bit remains set.

16.2.5 AUTO-CONVERSION TRIGGER

The Auto-conversion Trigger allows periodic ADC measurements without software intervention. When a rising edge of the selected source occurs, the GO/$\overline{\text{DONE}}$ bit is set by hardware.

The Auto-conversion Trigger source is selected with the TRIGSEL<5:0> bits of the ADCON2 register.

Using the Auto-conversion Trigger does not assure proper ADC timing. It is the user's responsibility to ensure that the ADC timing requirements are met.

See Table 16-2 for auto-conversion sources.

TABLE 16-2: AUTO-CONVERSION SOURCES

Source Peripheral	Signal Name
CCP1	CCP1_trigger
CCP2	CCP2_trigger
CCP7	CCP7_trigger
CCP8(1)	CCP8_trigger
Timer0	T0_overflow
Timer1	T1_overflow
Timer3	T3_overflow
Timer5	T5_overflow
Timer2	T2_postscaled
Timer4	T4_postscaled
Timer6	T6_postscaled
Timer8	T8_postscaled
Comparator C1	sync_C1OUT
Comparator C2	sync_C2OUT
Comparator C3	sync_C3OUT
Comparator C4	sync_C4OUT
Comparator C5	sync_C5OUT
Comparator C6	sync_C6OUT
Comparator C7(1)	sync_C7OUT
Comparator C8(1)	sync_C8OUT
CLC1	LC1_out
CLC2	LC2_out
CLC3	LC3_out
CLC4	LC4_out
PWM3	PWM3OUT
PWM4	PWM4OUT
PWM9	PWM9OUT
PWM9	PR/PH/OF/DC9_match
PWM5	PR/PH/OF/DC5_match
PWM6	PR/PH/OF/DC6_match
PWM10(1)	PR/PH/OF/DC10_match
PWM11	PR/PH/OF/DC11_match
PWM12(1)	PR/PH/OF/DC12_match
ADCACT	ADCACTPPS Pin

Note 1: PIC16(L)F1777/9 only.

REGISTER 19-4: CMxPSEL: COMPARATOR Cx POSITIVE CHANNEL SELECT REGISTER 1

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

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

bit 7-4 **Unimplemented:** Read as '0'

bit 3-0 **PCH<3:0>:** Comparator Positive Input Channel Select bits
CxVP connects to input source indicated by Table 19-5: Positive Input Sources

TABLE 19-5: POSITIVE INPUT SOURCES

PCH<3:0>	C1, C2, C3, C4	C5, C6, C7(1), C8(1)
1111	Reserved. Do not use	Reserved. Do not use
1110	Reserved. Do not use	Reserved. Do not use
1101	Reserved. Do not use	Reserved. Do not use
1100	Reserved. Do not use	Reserved. Do not use
1011	Reserved. Do not use	Reserved. Do not use
1010	Reserved. Do not use	Reserved. Do not use
1001	AGND	AGND
1000	DAC4_out	DAC8_out(1)
0111	DAC3_out	DAC7_out
0110	DAC2_out	DAC6_out(1)
0101	DAC1_out	DAC5_out
0100	PRG2_out	PRG4_out(1)
0011	PRG1_out	PRG3_out
0010	FVR_Buffer2	FVR_Buffer2
0001	CxIN1+ pin	CxIN1+ pin
0000	CxIN0+ pin	CxIN0+ pin

Note 1: PIC16(L)F1777/9 only.

22.0 TIMER1/3/5 MODULE WITH GATE CONTROL

The Timer1 module is a 16-bit timer/counter with the following features:

- 16-bit timer/counter register pair (TMR1H:TMR1L)
- Programmable internal or external clock source
- 2-bit prescaler
- Dedicated 32 kHz oscillator circuit
- Optionally synchronized comparator out
- Multiple Timer1 gate (count enable) sources
- Interrupt-on-overflow
- Wake-up on overflow (external clock, Asynchronous mode only)
- Time base for the Capture/Compare function
- Auto-conversion Trigger (with CCP)
- Selectable Gate Source Polarity
- Gate Toggle mode
- Gate Single-pulse mode
- Gate Value Status
- Gate Event Interrupt

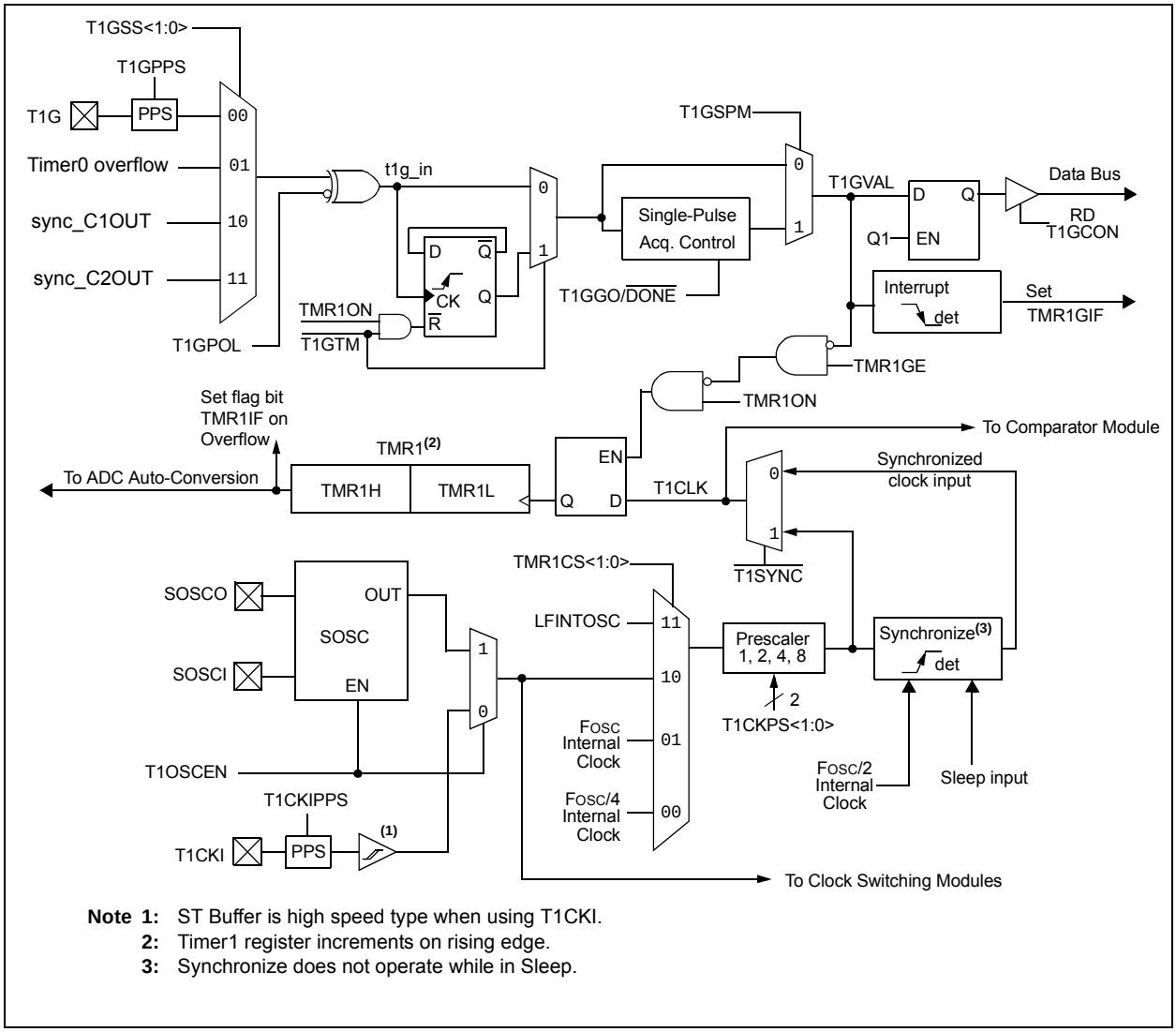
Figure 22-1 is a block diagram of the Timer1 module.

This device has three instances of Timer1 type modules. They include:

- Timer1
- Timer3
- Timer5

Note:	All references to Timer1 and Timer1 Gate apply to Timer3 and Timer5.

FIGURE 22-1: TIMER1 BLOCK DIAGRAM



Note 1: ST Buffer is high speed type when using T1CKI.
2: Timer1 register increments on rising edge.
3: Synchronize does not operate while in Sleep.

REGISTER 23-4: TxRST: TIMERx EXTERNAL RESET SIGNAL SELECTION REGISTER

U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	RSEL<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 **RSEL<4:0>:** TimerX External Reset Signal Source Selection bits
See Table 23-4.

TABLE 23-4: EXTERNAL RESET SOURCES

RSEL<4:0>	Timer2	Timer4	Timer6	Timer8
11111	Reserved	Reserved	Reserved	Reserved
11110	Reserved	Reserved	Reserved	Reserved
11101	LC4_out	LC4_out	LC4_out	LC4_out
11100	LC3_out	LC3_out	LC3_out	LC3_out
11011	LC2_out	LC2_out	LC2_out	LC2_out
11010	LC1_out	LC1_out	LC1_out	LC1_out
11001	ZCD_out	ZCD_out	ZCD_out	ZCD_out
11000(1)	sync_C8OUT	sync_C8OUT	sync_C8OUT	sync_C8OUT
10111(1)	sync_C7OUT	sync_C7OUT	sync_C7OUT	sync_C7OUT
10110	sync_C6OUT	sync_C6OUT	sync_C6OUT	sync_C6OUT
10101	sync_C5OUT	sync_C5OUT	sync_C5OUT	sync_C5OUT
10100	sync_C4OUT	sync_C4OUT	sync_C4OUT	sync_C4OUT
10011	sync_C3OUT	sync_C3OUT	sync_C3OUT	sync_C3OUT
10010	sync_C2OUT	sync_C2OUT	sync_C2OUT	sync_C2OUT
10001	sync_C1OUT	sync_C1OUT	sync_C1OUT	sync_C1OUT
10000(1)	PWM12_out	PWM12_out	PWM12_out	PWM12_out
01111	PWM11_out	PWM11_out	PWM11_out	PWM11_out
01110	PWM6_out	PWM6_out	PWM6_out	PWM6_out
01101	PWM5_out	PWM5_out	PWM5_out	PWM5_out
01100(1)	PWM10_out	PWM10_out	PWM10_out	PWM10_out
01011	PWM9_out	PWM9_out	PWM9_out	PWM9_out
01010	PWM4_out	PWM4_out	PWM4_out	PWM4_out
01001	PWM3_out	PWM3_out	PWM3_out	PWM3_out
01000(1)	CCP8_out	CCP8_out	CCP8_out	CCP8_out
00111	CCP7_out	CCP7_out	CCP7_out	CCP7_out
00110	CCP2_out	CCP2_out	CCP2_out	CCP2_out
00101	CCP1_out	CCP1_out	CCP1_out	CCP1_out
00100	TMR8_postscaled	TMR8_postscaled	TMR8_postscaled	Reserved
00011	TMR6_postscaled	TMR6_postscaled	Reserved	TMR6_postscaled
00010	TMR4_postscaled	Reserved	TMR4_postscaled	TMR4_postscaled
00001	Reserved	TMR2_postscaled	TMR2_postscaled	TMR2_postscaled
00000	Pin selected byT2INPPS	Pin selected by T4INPPS	Pin selected by T6INPPS	Pin selected by T6INPPS

Note 1: PIC16LF1777/9 only.

25.0 10-BIT PULSE-WIDTH MODULATION (PWM) MODULE

The 10-bit PWM module generates a Pulse-Width Modulated signal determined by the duty cycle, period, and resolution that are configured by the following registers:

- T2PR
- T2CON
- PWMxDCH
- PWMxDCL
- PWMxCON

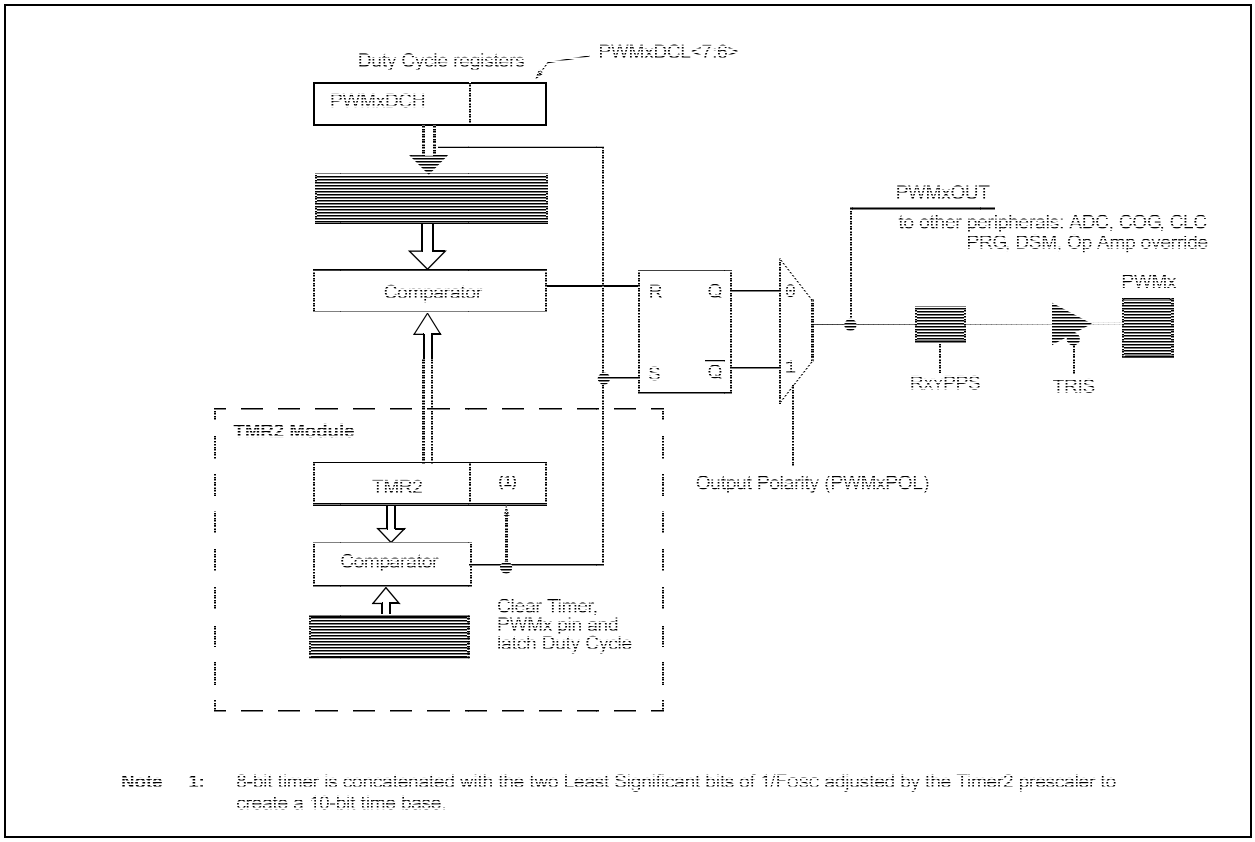
Figure 25-1 shows a simplified block diagram of PWM operation.

Figure 25-2 shows a typical waveform of the PWM signal.

TABLE 25-1: AVAILABLE 10-BIT PWM MODULES

Device	PWM3	PWM4	PWM9	PWM10
PIC16(L)F1778	•	•	•	
PIC16(L)F1777/9	•	•	•	•

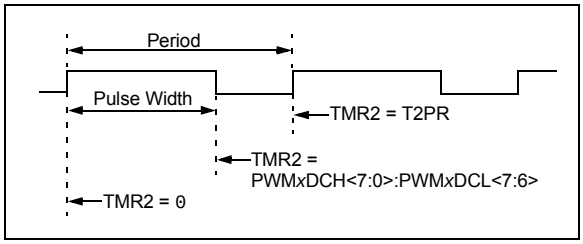
FIGURE 25-1: SIMPLIFIED PWM BLOCK DIAGRAM



Note 1: 8-bit timer is concatenated with the two Least Significant bits of 1/Fosc adjusted by the Timer2 prescaler to create a 10-bit time base.

For a step-by-step procedure on how to set up this module for PWM operation, refer to **Section 25.1.9 "Setup for PWM Operation using PWMx Output Pins"**.

FIGURE 25-2: PWM OUTPUT



25.2 Register Definitions: 10-Bit PWM Control

Long bit name prefixes for the DSM peripherals are shown in Table 25-4. Refer to **Section 1.1.2.2 "Long Bit Names"** for more information

TABLE 25-4:

Peripheral	Bit Name Prefix
PWM3	PWM3
PWM4	PWM4
PWM9	PWM9
PWM10(1)	PWM10

Note 1: PIC16(L)F1777/9 only.

REGISTER 25-1: PWMxCON: PWM CONTROL REGISTER

R/W-0/0	U-0	R-0/0	R/W-0/0	U-0	U-0	U-0	U-0
EN	—	OUT	POL	—	—	—	—
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 **EN:** PWM Module Enable bit
1 = PWM module is enabled
0 = PWM module is disabled

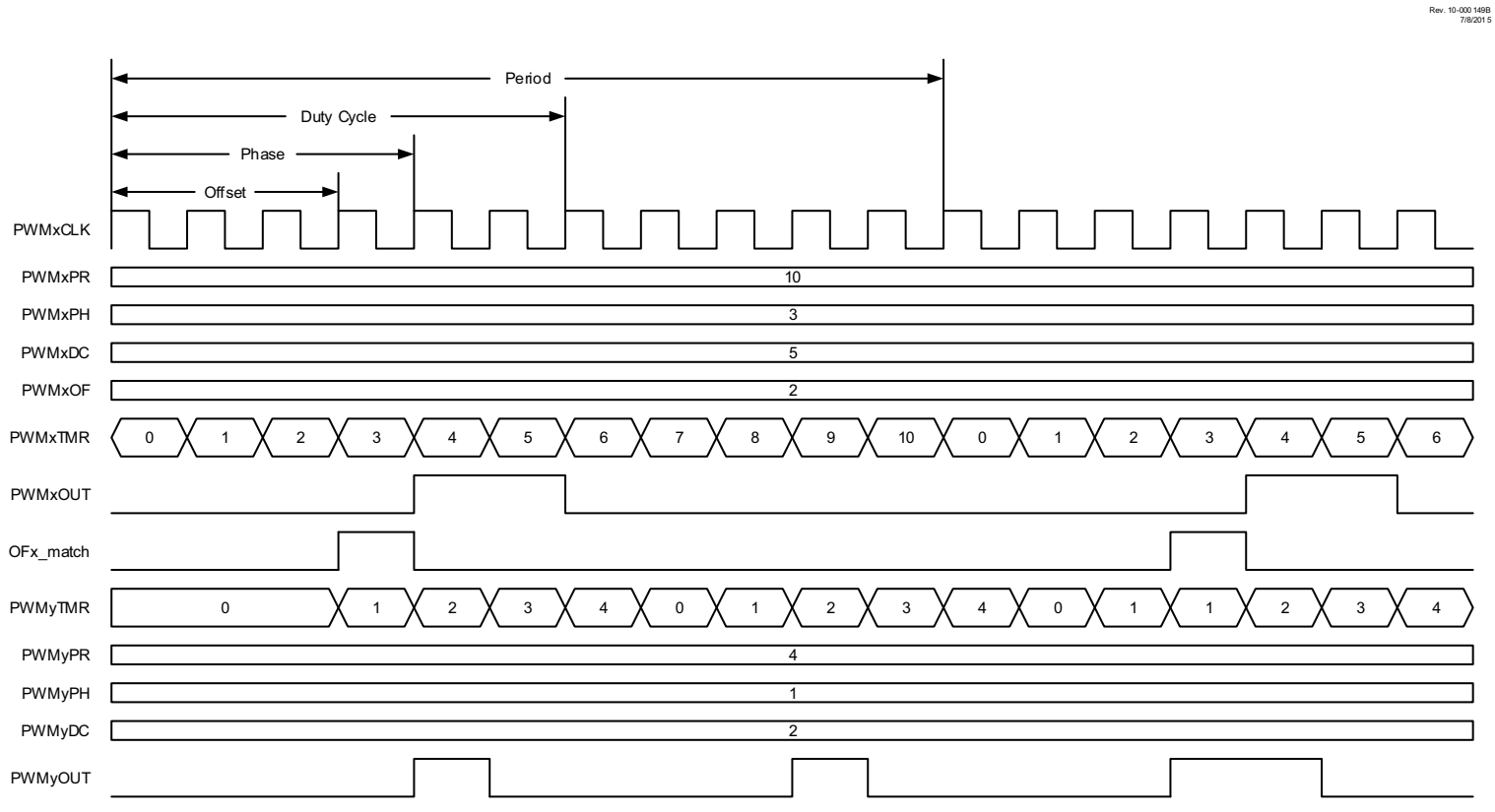
bit 6 **Unimplemented:** Read as '0'

bit 5 **OUT:** PWM module output level when bit is read.

bit 4 **POL:** PWMx Output Polarity Select bit
1 = PWM output is active-low
0 = PWM output is active-high

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

FIGURE 26-11: CONTINUOUS SLAVE RUN MODE WITH IMMEDIATE RESET AND SYNC START TIMING DIAGRAM



30.9 Slope Compensation Application

An example slope compensation circuit is shown in Figure 30-6. The PRG input voltage is PRGxIN which shares an I/O pin with the op amp output. The op amp output is designed to operate at the expected peak current sense voltage (i.e., VREF). The PRG output voltage starts at VREF and should fall at a rate less than half the target circuit current sense voltage rate of rise. Therefore, the compensator slope expressed as volts per µs can be computed by Equation 30-1.

EQUATION 30-1:

$$\frac{V}{\mu s} \geq \frac{\frac{V_{REF}}{2}}{PWM\ Period\ (\mu s)}$$

For example, when the circuit is using a 1Ω current sense resistor and the peak current is 1A, then the peak current expressed as a voltage is 1V. Therefore, for this example, the op amp output should be designed to operate at 1V. If the power supply PWM frequency is 1 MHz, then the period is 1 µs. Therefore, the desired slope is 0.5 V/µs, which is computed as shown in Equation 30-2.

EQUATION 30-2:

$$\frac{\frac{V_{REF}}{2}}{PWM\ Period\ (\mu s)} = \frac{\frac{1}{2}}{1\mu s} = 0.5V/\mu s$$

Note: The setting for 0.5V/µs is ISET<4:0> = 6

FIGURE 30-6: EXAMPLE SLOPE COMPENSATION CIRCUIT

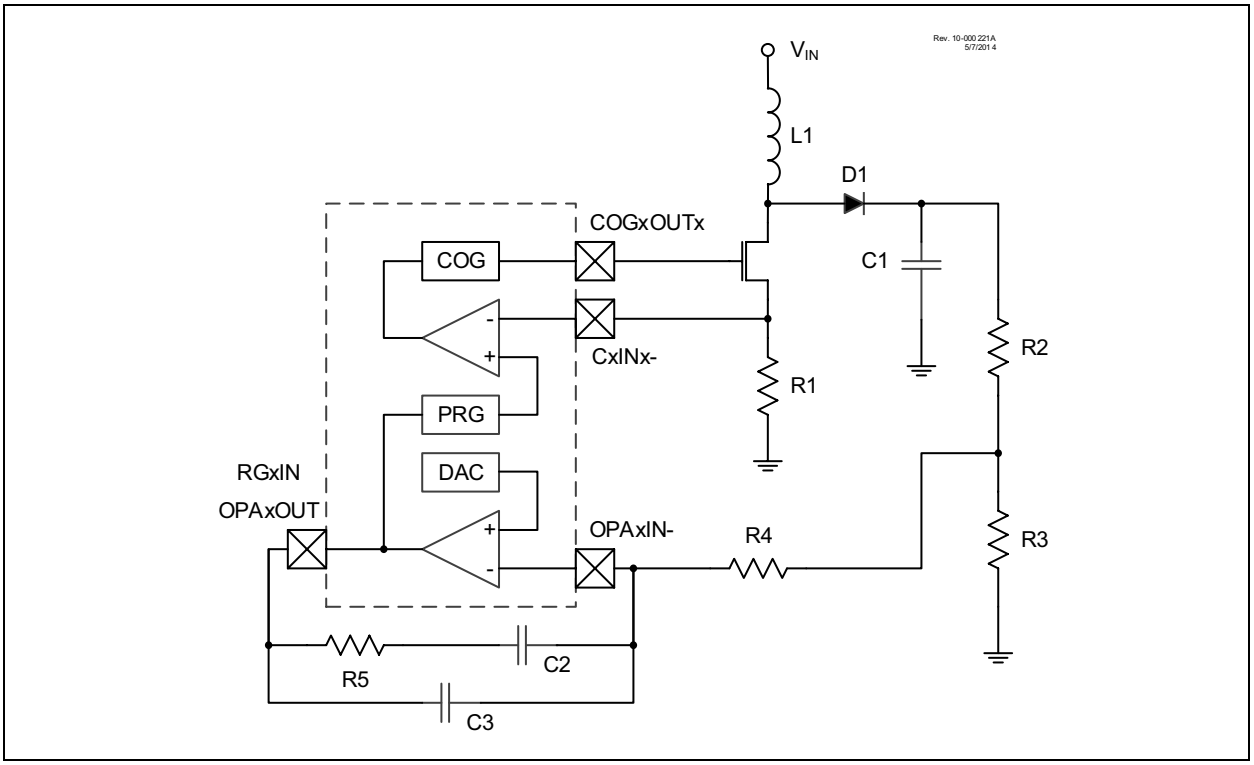
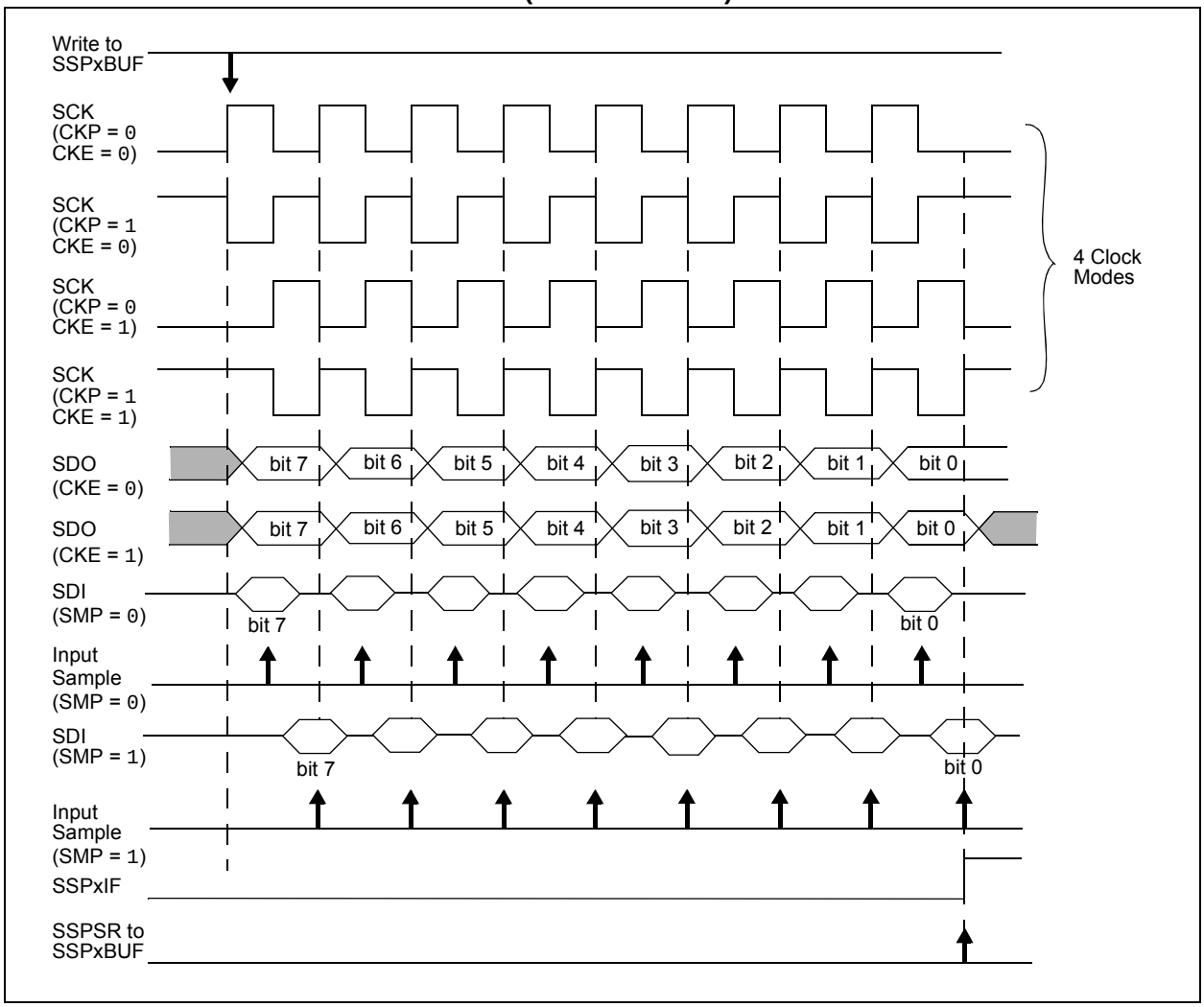


FIGURE 32-6: SPI MODE WAVEFORM (MASTER MODE)



32.2.4 SPI SLAVE MODE

In Slave mode, the data is transmitted and received as external clock pulses appear on SCK. When the last bit is latched, the SSPxIF interrupt flag bit is set.

Before enabling the module in SPI Slave mode, the clock line must match the proper Idle state. The clock line can be observed by reading the SCK pin. The Idle state is determined by the CKP bit of the SSPxCON1 register.

While in Slave mode, the external clock is supplied by the external clock source on the SCK pin. This external clock must meet the minimum high and low times as specified in the electrical specifications.

While in Sleep mode, the slave can transmit/receive data. The shift register is clocked from the SCK pin input and when a byte is received, the device will generate an interrupt. If enabled, the device will wake-up from Sleep.

32.2.4.1 Daisy-Chain Configuration

The SPI bus can sometimes be connected in a daisy-chain configuration. The first slave output is connected to the second slave input, the second slave output is connected to the third slave input, and so on. The final slave output is connected to the master input. Each slave sends out, during a second group of clock pulses, an exact copy of what was received during the first group of clock pulses. The whole chain acts as one large communication shift register. The daisy-chain feature only requires a single Slave Select line from the master device.

Figure 32-7 shows the block diagram of a typical daisy-chain connection when operating in SPI mode.

In a daisy-chain configuration, only the most recent byte on the bus is required by the slave. Setting the BOEN bit of the SSPxCON3 register will enable writes to the SSPxBUF register, even if the previous byte has not been read. This allows the software to ignore data that may not apply to it.

32.4 I²C MODE OPERATION

All MSSP I²C communication is byte oriented and shifted out MSb first. Six SFR registers and two interrupt flags interface the module with the PIC® microcontroller and user software. Two pins, SDA and SCL, are exercised by the module to communicate with other external I²C devices.

32.4.1 BYTE FORMAT

All communication in I²C is done in 9-bit segments. A byte is sent from a master to a slave or vice-versa, followed by an Acknowledge bit sent back. After the eighth falling edge of the SCL line, the device outputting data on the SDA changes that pin to an input and reads in an acknowledge value on the next clock pulse.

The clock signal, SCL, is provided by the master. Data is valid to change while the SCL signal is low, and sampled on the rising edge of the clock. Changes on the SDA line while the SCL line is high define special conditions on the bus, explained below.

32.4.2 DEFINITION OF I²C TERMINOLOGY

There is language and terminology in the description of I²C communication that have definitions specific to I²C. That word usage is defined below and may be used in the rest of this document without explanation. This table was adapted from the Philips I²C specification.

32.4.3 SDA AND SCL PINS

Selection of any I²C mode with the SSPEN bit set, forces the SCL and SDA pins to be open-drain. These pins should be set by the user to inputs by setting the appropriate TRIS bits.

Note 1: Data is tied to output zero when an I²C mode is enabled.

2: Any device pin can be selected for SDA and SCL functions with the PPS peripheral. These functions are bidirectional. The SDA input is selected with the SSPDATPPS registers. The SCL input is selected with the SSPCLKPPS registers. Outputs are selected with the RxyPPS registers. It is the user's responsibility to make the selections so that both the input and the output for each function is on the same pin.

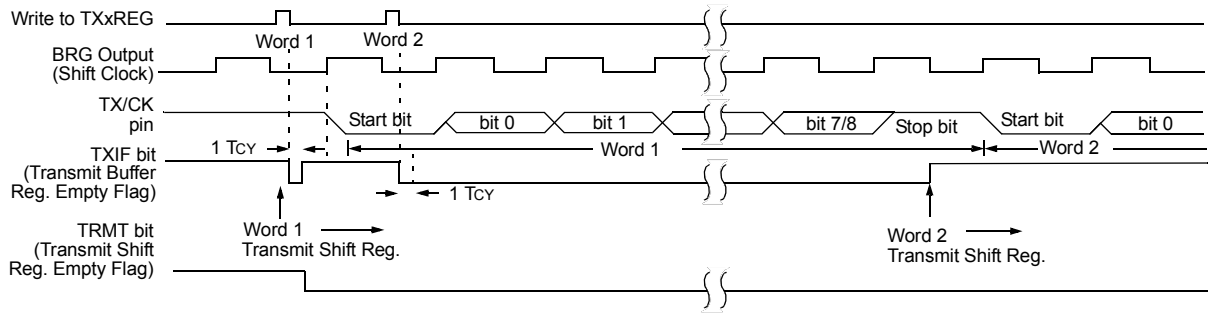
32.4.4 SDA HOLD TIME

The hold time of the SDA pin is selected by the SDAHT bit of the SSPxCON3 register. Hold time is the time SDA is held valid after the falling edge of SCL. Setting the SDAHT bit selects a longer 300 ns minimum hold time and may help on buses with large capacitance.

TABLE 32-2: I²C BUS TERMS

TERM	Description
Transmitter	The device which shifts data out onto the bus.
Receiver	The device which shifts data in from the bus.
Master	The device that initiates a transfer, generates clock signals and terminates a transfer.
Slave	The device addressed by the master.
Multi-master	A bus with more than one device that can initiate data transfers.
Arbitration	Procedure to ensure that only one master at a time controls the bus. Winning arbitration ensures that the message is not corrupted.
Synchronization	Procedure to synchronize the clocks of two or more devices on the bus.
Idle	No master is controlling the bus, and both SDA and SCL lines are high.
Active	Any time one or more master devices are controlling the bus.
Addressed Slave	Slave device that has received a matching address and is actively being clocked by a master.
Matching Address	Address byte that is clocked into a slave that matches the value stored in SSPxADD.
Write Request	Slave receives a matching address with R/$\overline{W}$ bit clear, and is ready to clock in data.
Read Request	Master sends an address byte with the R/$\overline{W}$ bit set, indicating that it wishes to clock data out of the Slave. This data is the next and all following bytes until a Restart or Stop.
Clock Stretching	When a device on the bus hold SCL low to stall communication.
Bus Collision	Any time the SDA line is sampled low by the module while it is outputting and expected high state.

FIGURE 33-4: ASYNCHRONOUS TRANSMISSION (BACK-TO-BACK)



Note: This timing diagram shows two consecutive transmissions.

TABLE 33-1: SUMMARY OF REGISTERS ASSOCIATED WITH ASYNCHRONOUS TRANSMISSION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ANSELA	—	—	ANSA5	ANSA4	ANSA3	ANSA2	ANSA1	ANSA0	177
ANSELB	—	—	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1	ANSB0	182
ANSELC	ANSC7	ANSC6	ANSC5	ANSC4	ANSC3	ANSC2	—	—	187
BAUD1CON	ABDOVF	RCIDL	—	SCKP	BRG16	—	WUE	ABDEN	505
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	132
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	133
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	139
RC1STA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	504
RxyPPS	—	—	RxyPPS<5:0>						205, 207
SP1BRGL	SP1BRG<7:0>								506*
SP1BRGH	SP1BRG<15:8>								506*
TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	176
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	181
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	186
TX1REG	EUSART Transmit Data Register								495*
TX1STA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	503

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

* Page provides register information.

REGISTER 33-2: RC1STA: RECEIVE STATUS AND CONTROL REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R-0/0	R-0/0	R-0/0
SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D
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 **SPEN:** Serial Port Enable bit
1 = Serial port enabled
0 = Serial port disabled (held in Reset)

bit 6 **RX9:** 9-Bit Receive Enable bit
1 = Selects 9-bit reception
0 = Selects 8-bit reception

bit 5 **SREN:** Single Receive Enable bit
Asynchronous mode:
Don't care
Synchronous mode – Master:
1 = Enables single receive
0 = Disables single receive
This bit is cleared after reception is complete.
Synchronous mode – Slave
Don't care

bit 4 **CREN:** Continuous Receive Enable bit
Asynchronous mode:
1 = Enables receiver
0 = Disables receiver
Synchronous mode:
1 = Enables continuous receive until enable bit CREN is cleared (CREN overrides SREN)
0 = Disables continuous receive

bit 3 **ADDEN:** Address Detect Enable bit
Asynchronous mode 9-bit (RX9 = 1):
1 = Enables address detection, enable interrupt and load the receive buffer when RSR<8> is set
0 = Disables address detection, all bytes are received and ninth bit can be used as parity bit
Asynchronous mode 8-bit (RX9 = 0):
Don't care

bit 2 **FERR:** Framing Error bit
1 = Framing error (can be updated by reading RCxREG register and receive next valid byte)
0 = No framing error

bit 1 **OERR:** Overrun Error bit
1 = Overrun error (can be cleared by clearing bit CREN)
0 = No overrun error

bit 0 **RX9D:** Ninth bit of Received Data
This can be address/data bit or a parity bit and must be calculated by user firmware.

Note: Unless otherwise noted, VIN = 5V, FOSC = 300 kHz, CIN = 0.1 µF, TA = 25°C.

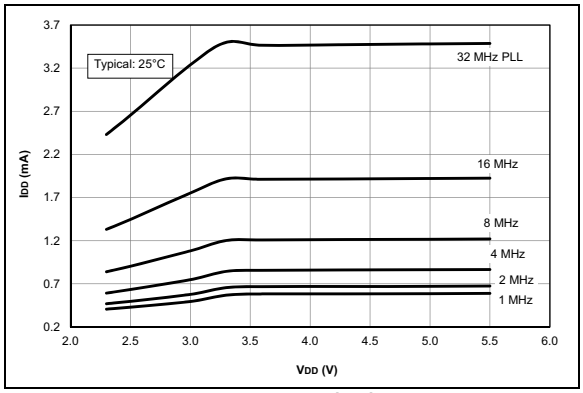


FIGURE 37-25: *IDD Typical, HFINTOSC Mode, PIC16F1777/8/9 Only.*

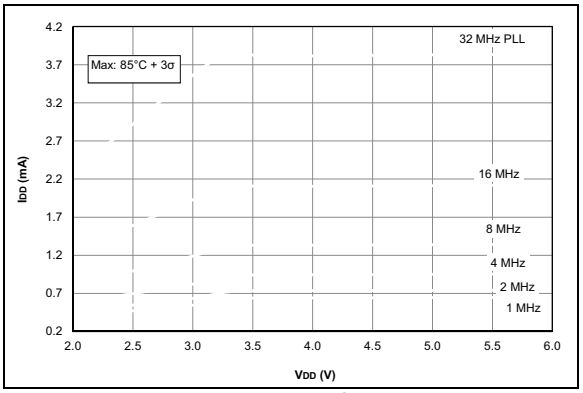


FIGURE 37-26: *IDD Maximum, HFINTOSC Mode, PIC16F1777/8/9 Only.*

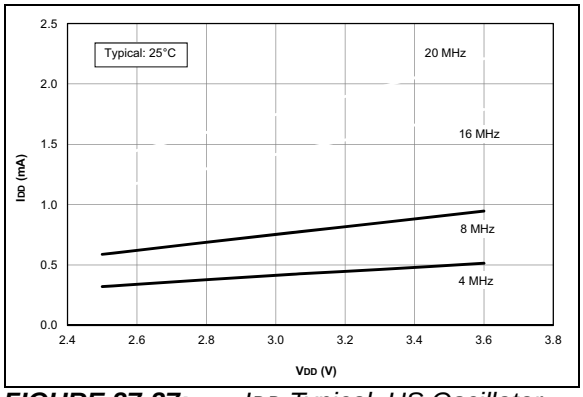


FIGURE 37-27: *IDD Typical, HS Oscillator, 25°C, PIC16LF1777/8/9 Only.*

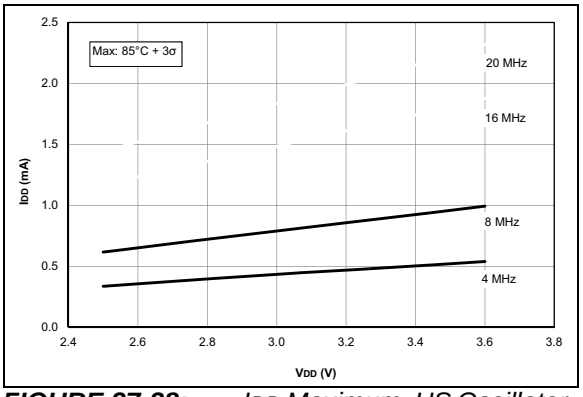


FIGURE 37-28: *IDD Maximum, HS Oscillator, PIC16LF1777/8/9 Only.*

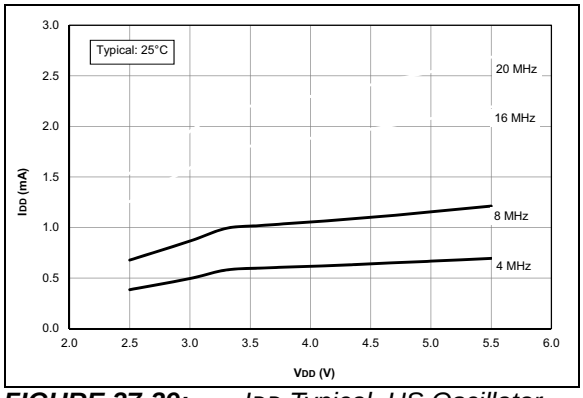


FIGURE 37-29: *IDD Typical, HS Oscillator, 25°C, PIC16F1777/8/9 Only.*

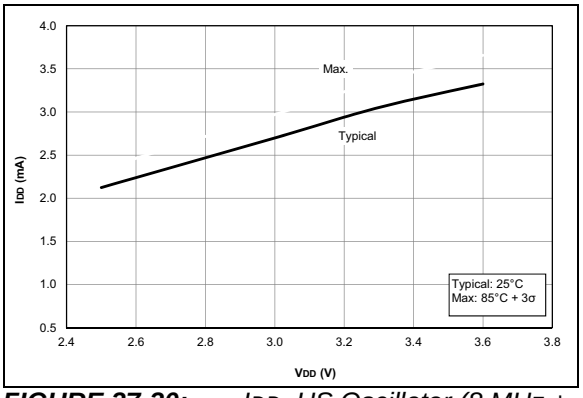
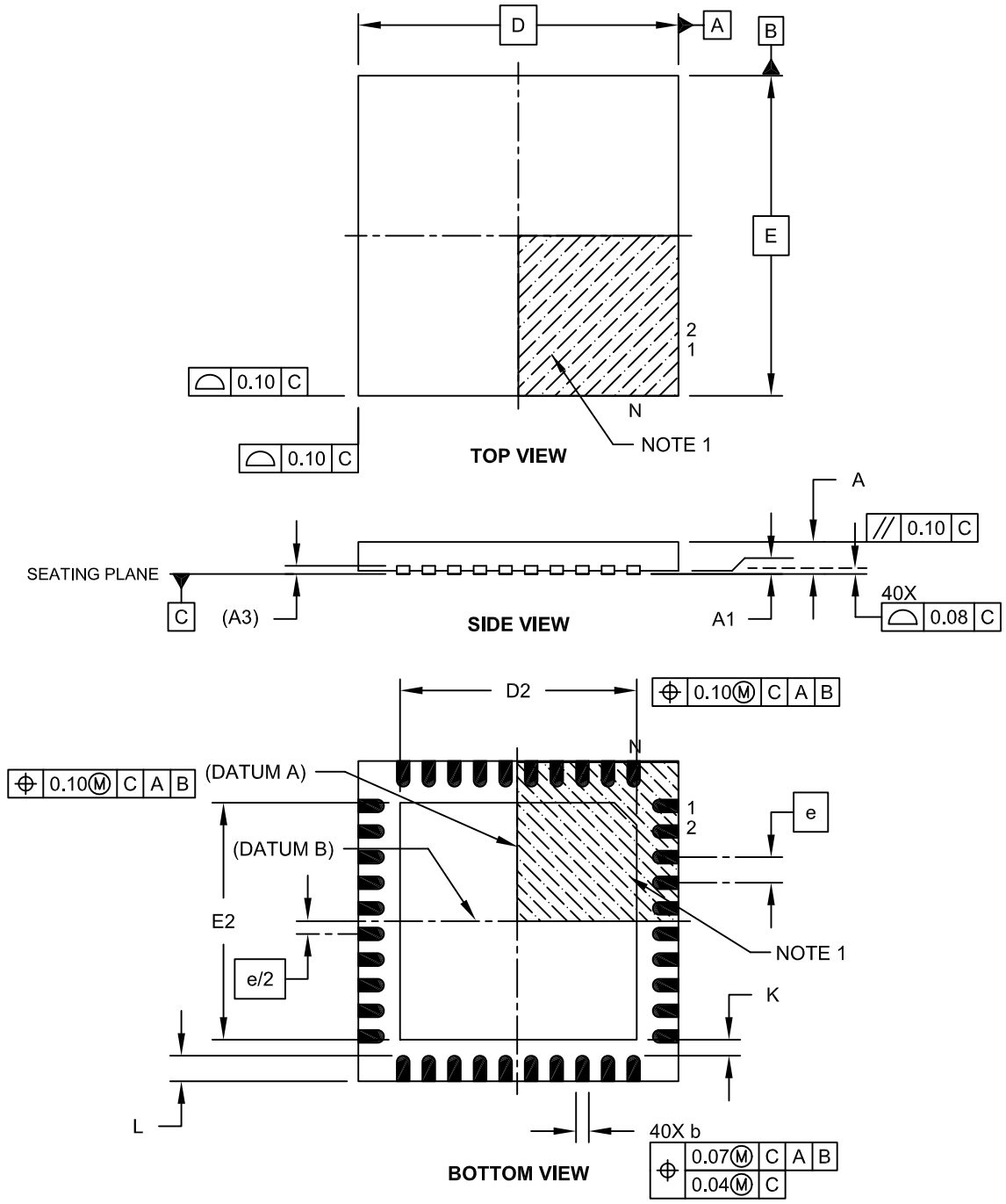


FIGURE 37-30: *IDD, HS Oscillator (8 MHz + 4x PLL), PIC16LF1777/8/9 Only.*

40-Lead Ultra Thin Plastic Quad Flat, No Lead Package (MV) – 5x5x0.5 mm Body [UQFN]

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



Microchip Technology Drawing C04-156A Sheet 1 of 2