



Welcome to [E-XFL.COM](https://www.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	I ² C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	36
Program Memory Size	28KB (16K x 14)
Program Memory Type	FLASH
EEPROM Size	128 x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 28x10b; D/A 4x5b, 4x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°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/pic16lf1779-e-mv

Table of Contents

1.0	Device Overview	15
2.0	Enhanced Mid-Range CPU	34
3.0	Memory Organization	36
4.0	Device Configuration	94
5.0	Oscillator Module (with Fail-Safe Clock Monitor)	101
6.0	Resets	119
7.0	Interrupts	127
8.0	Power-Down Mode (Sleep)	146
9.0	Watchdog Timer (WDT)	151
10.0	Flash Program Memory Control	156
11.0	I/O Ports	173
12.0	Peripheral Pin Select (PPS) Module	203
13.0	Interrupt-On-Change	213
14.0	Fixed Voltage Reference (FVR)	221
15.0	Temperature Indicator Module	224
16.0	Analog-to-Digital Converter (ADC) Module	226
17.0	5-Bit Digital-to-Analog Converter (DAC) Module	241
18.0	10-bit Digital-to-Analog Converter (DAC) Module	246
19.0	Comparator Module	252
20.0	Zero-Cross Detection (ZCD) Module	265
21.0	Timer0 Module	272
22.0	Timer1/3/5 Module with Gate Control	275
23.0	Timer2/4/6/8 Module	286
24.0	Capture/Compare/PWM Modules	311
25.0	10-Bit Pulse-Width Modulation (PWM) Module	325
26.0	16-bit Pulse-Width Modulation (PWM) Module	332
27.0	Complementary Output Generator (COG) Modules	359
28.0	Configurable Logic Cell (CLC)	391
29.0	Operational Amplifier (OPA) Modules	406
30.0	Programmable Ramp Generator (PRG) Module	413
31.0	Data Signal Modulator (DSM)	427
32.0	Master Synchronous Serial Port (MSSP) Module	440
33.0	Enhanced Universal Synchronous Asynchronous Receiver Transmitter (EUSART)	493
34.0	In-Circuit Serial Programming™ (ICSP™)	524
35.0	Instruction Set Summary	526
36.0	Electrical Specifications	540
37.0	DC and AC Characteristics Graphs and Charts	575
38.0	Development Support	599
39.0	Packaging Information	603
	Appendix A: Data Sheet Revision History	625

3.3 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 3.7 “Indirect Addressing”** for more information.

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

3.3.1 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 3-17.

TABLE 3-2: CORE REGISTERS

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

3.3.1.1 STATUS Register

The STATUS register, shown in Register 3-1, contains:

- the arithmetic status of the ALU
- the Reset status

The STATUS register can be the destination for any instruction, like any other register. If the STATUS register is the destination for an instruction that affects the Z, DC or C bits, then the write to these three bits is disabled. These bits are set or cleared according to the device logic. Furthermore, the \overline{TO} and \overline{PD} bits are not writable. Therefore, the result of an instruction with the STATUS register as destination may be different than intended.

For example, `CLRF STATUS` will clear the upper three bits and set the Z bit. This leaves the STATUS register as '000u u1uu' (where u = unchanged).

It is recommended, therefore, that only `BCF`, `BSF`, `SWAPF` and `MOVWF` instructions are used to alter the STATUS register, because these instructions do not affect any Status bits. For other instructions not affecting any Status bits (Refer to **Section 35.0 “Instruction Set Summary”**).

Note: The C and DC bits operate as Borrow and Digit Borrow out bits, respectively, in subtraction.

11.1 PORTA Registers

11.1.1 DATA REGISTER

PORTA is a 6-bit wide, bidirectional port. The corresponding data direction register is TRISA (Register 11-2). Setting a TRISA bit (= 1) will make the corresponding PORTA pin an input (i.e., disable the output driver). Clearing a TRISA bit (= 0) will make the corresponding PORTA pin an output (i.e., enables output driver and puts the contents of the output latch on the selected pin). The exception is RA3, which is input-only and its TRIS bit will always read as '1'. Example 11-1 shows how to initialize PORTA.

Reading the PORTA register (Register 11-1) reads the status of the pins, whereas writing to it will write to the PORT latch. All write operations are read-modify-write operations. Therefore, a write to a port implies that the port pins are read, this value is modified and then written to the PORT data latch (LATA).

11.1.2 DIRECTION CONTROL

The TRISA register (Register 11-2) controls the PORTA pin output drivers, even when they are being used as analog inputs. The user should ensure the bits in the TRISA register are maintained set when using them as analog inputs. I/O pins configured as analog inputs always read '0'.

11.1.3 OPEN-DRAIN CONTROL

The ODCONA register (Register 11-6) controls the open-drain feature of the port. Open-drain operation is independently selected for each pin. When an ODCONA bit is set, the corresponding port output becomes an open-drain driver capable of sinking current only. When an ODCONA bit is cleared, the corresponding port output pin is the standard push-pull drive capable of sourcing and sinking current.

11.1.4 SLEW RATE CONTROL

The SLRCONA register (Register 11-7) controls the slew rate option for each port pin. Slew rate control is independently selectable for each port pin. When an SLRCONA bit is set, the corresponding port pin drive is slew rate limited. When an SLRCONA bit is cleared, The corresponding port pin drive slews at the maximum rate possible.

11.1.5 INPUT THRESHOLD CONTROL

The INLVLA register (Register 11-8) controls the input voltage threshold for each of the available PORTA input pins. A selection between the Schmitt Trigger CMOS or the TTL Compatible thresholds is available. The input threshold is important in determining the value of a read of the PORTA register and also the level at which an interrupt-on-change occurs, if that feature is enabled. See Table 36-4: I/O Ports for more information on threshold levels.

Note: Changing the input threshold selection should be performed while all peripheral modules are disabled. Changing the threshold level during the time a module is active may inadvertently generate a transition associated with an input pin, regardless of the actual voltage level on that pin.

11.1.6 ANALOG CONTROL

The ANSELA register (Register 11-4) is used to configure the Input mode of an I/O pin to analog. Setting the appropriate ANSELA bit high will cause all digital reads on the pin to be read as '0' and allow analog functions on the pin to operate correctly.

The state of the ANSELA bits has no effect on digital output functions. A pin with TRIS clear and ANSEL set will still operate as a digital output, but the Input mode will be analog. This can cause unexpected behavior when executing read-modify-write instructions on the affected port.

Note: The ANSELA bits default to the Analog mode after reset. To use any pins as digital general purpose or peripheral inputs, the corresponding ANSEL bits must be initialized to '0' by user software.

EXAMPLE 11-1: INITIALIZING PORTA

```
; This code example illustrates
; initializing the PORTA register. The
; other ports are initialized in the same
; manner.

BANKSEL PORTA      ;
CLRF PORTA         ;Init PORTA
BANKSEL LATA        ;Data Latch
CLRF LATA          ;
BANKSEL ANSELA      ;
CLRF ANSELA        ;digital I/O
BANKSEL TRISA       ;
MOVLW B'00111000'  ;Set RA<5:3> as inputs
MOVWF TRISA        ;and set RA<2:0> as
                  ;outputs
```

PIC16(L)F1777/8/9

REGISTER 13-3: IOCAF: INTERRUPT-ON-CHANGE PORTA FLAG REGISTER

R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0
IOCAF7	IOCAF6	IOCAF5	IOCAF4	IOCAF3	IOCAF2	IOCAF1	IOCAF0
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 HS - Bit is set in hardware

bit 7-0 **IOCAF<7:0>:** Interrupt-on-Change PORTA Flag bits
1 = An enabled change was detected on the associated pin.
Set when IOCAPx = 1 and a rising edge was detected on RAX, or when IOCANx = 1 and a falling edge was detected on RAX.
0 = No change was detected, or the user cleared the detected change.

REGISTER 13-4: IOCBP: INTERRUPT-ON-CHANGE PORTB POSITIVE EDGE REGISTER

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
IOCBP7	IOCBP6	IOCBP5	IOCBP4	IOCBP3	IOCBP2	IOCBP1	IOCBP0
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 **IOCBP<7:0>:** Interrupt-on-Change PORTB Positive Edge Enable bits
1 = Interrupt-on-Change enabled on the pin for a positive going edge. IOCBFx bit and IOCIF flag will be set upon detecting an edge.
0 = Interrupt-on-Change disabled for the associated pin.

REGISTER 13-5: IOCBN: INTERRUPT-ON-CHANGE PORTB NEGATIVE EDGE REGISTER

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
IOCBN7	IOCBN6	IOCBN5	IOCBN4	IOCBN3	IOCBN2	IOCBN1	IOCBN0
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 **IOCBN<7:0>:** Interrupt-on-Change PORTB Negative Edge Enable bits
1 = Interrupt-on-Change enabled on the pin for a negative going edge. IOCBFx bit and IOCIF flag will be set upon detecting an edge.
0 = Interrupt-on-Change disabled for the associated pin.

REGISTER 16-1: ADCON0: ADC CONTROL REGISTER 0 (CONTINUED)

- bit 1 **GO/DONE:** ADC Conversion Status bit
1 = ADC conversion cycle in progress. Setting this bit starts an ADC conversion cycle.
This bit is automatically cleared by hardware when the ADC conversion has completed.
0 = ADC conversion completed/not in progress
- bit 0 **ADON:** ADC Enable bit
1 = ADC is enabled
0 = ADC is disabled and consumes no operating current
- Note**
- 1: See Section 17.0 “5-Bit Digital-to-Analog Converter (DAC) Module” for more information.
 - 2: See Section 14.0 “Fixed Voltage Reference (FVR)” for more information.
 - 3: See Section 15.0 “Temperature Indicator Module” for more information.
 - 4: See Section 18.0 “10-bit Digital-to-Analog Converter (DAC) Module” for more information.
 - 5: Input source is switched off when op amp override is forcing tri-state. See Section 29.3 “Override Control”
 - 6: PIC16(L)F1777/9 only.

PIC16(L)F1777/8/9

REGISTER 16-2: ADCON1: ADC CONTROL REGISTER 1

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	R/W-0/0	R/W-0/0	R/W-0/0
ADFM	ADCS<2:0>			—	ADNREF	ADPREF<1:0>	
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 **ADFM:** ADC Result Format Select bit
1 = Right justified. Six Most Significant bits of ADRESH are set to '0' when the conversion result is loaded.
0 = Left justified. Six Least Significant bits of ADRESL are set to '0' when the conversion result is loaded.
- bit 6-4 **ADCS<2:0>:** ADC Conversion Clock Select bits
111 = FRC (clock supplied from an internal RC oscillator)
110 = Fosc/64
101 = Fosc/16
100 = Fosc/4
011 = FRC (clock supplied from an internal RC oscillator)
010 = Fosc/32
001 = Fosc/8
000 = Fosc/2
- bit 3 **Unimplemented:** Read as '0'
- bit 2 **ADNREF:** ADC Negative Voltage Reference Configuration bit
1 = VREF- is connected to external VREF- pin
0 = VREF- is connected to VSS
- bit 1-0 **ADPREF<1:0>:** ADC Positive Voltage Reference Configuration bits
11 = VREF+ is connected to internal Fixed Voltage Reference (FVR) module⁽¹⁾
10 = VREF+ is connected to external VREF+ pin⁽¹⁾
01 = Reserved
00 = VREF+ is connected to VDD

Note 1: When selecting the VREF+ pin as the source of the positive reference, be aware that a minimum voltage specification exists. See Table 36-16: ADC Conversion Requirements for details.

PIC16(L)F1777/8/9

REGISTER 19-3: CMxNSEL: COMPARATOR Cx NEGATIVE CHANNEL SELECT REGISTER

U-0	U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	—	NCH<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 **NCH<3:0>:** Comparator Negative Input Channel Select bits

CxVN connects to input source indicated by Table 19-4: Negative Input Sources

TABLE 19-4: NEGATIVE INPUT SOURCES

NCH<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	OPA2IN- pin	OPA4IN- pin ⁽¹⁾
1001	OPA1IN- pin	OPA3IN- pin
1000	AGND	AGND
0111	PRG2_out	PRG4_out ⁽¹⁾
0110	PRG1_out	PRG3_out
0101	FVR_Buffer2	FVR_Buffer2
0100	CxIN4- pin	CxIN4- (OPA4OUT) pin ⁽¹⁾
0011	CxIN3- (OPA2OUT) pin	CxIN3- pin
0010	CxIN2- pin	CxIN2- pin
0001	CxIN1- (OPA1OUT) pin	CxIN1- (OPA3OUT) pin
0000	CxIN0- pin	CxIN0- pin

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

TABLE 19-6: SUMMARY OF REGISTERS ASSOCIATED WITH COMPARATOR MODULE

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
CM1CON0	ON	OUT	—	POL	ZLF	Reserved	HYS	SYNC	258
CM2CON0	ON	OUT	—	POL	ZLF	Reserved	HYS	SYNC	258
CM3CON0	ON	OUT	—	POL	ZLF	Reserved	HYS	SYNC	258
CM4CON0	ON	OUT	—	POL	ZLF	Reserved	HYS	SYNC	258
CM5CON0	ON	OUT	—	POL	ZLF	Reserved	HYS	SYNC	258
CM6CON0	ON	OUT	—	POL	ZLF	Reserved	HYS	SYNC	258
CM1CON1	—	—	—	—	—	—	INTP	INTN	259
CM2CON1	—	—	—	—	—	—	INTP	INTN	259
CM3CON1	—	—	—	—	—	—	INTP	INTN	259
CM4CON1	—	—	—	—	—	—	INTP	INTN	259
CM5CON1	—	—	—	—	—	—	INTP	INTN	259
CM6CON1	—	—	—	—	—	—	INTP	INTN	259
CM7CON1 ⁽¹⁾	—	—	—	—	—	—	INTP	INTN	259
CM8CON1 ⁽¹⁾	—	—	—	—	—	—	INTP	INTN	259
CM1NSEL	—	—	—	—	NCH<3:0>				260
CM2NSEL	—	—	—	—	NCH<3:0>				260
CM3NSEL	—	—	—	—	NCH<3:0>				260
CM4NSEL	—	—	—	—	NCH<3:0>				260
CM5NSEL	—	—	—	—	NCH<3:0>				260
CM6NSEL	—	—	—	—	NCH<3:0>				260
CM7NSEL ⁽¹⁾	—	—	—	—	NCH<3:0>				260
CM8NSEL ⁽¹⁾	—	—	—	—	NCH<3:0>				260
CM1PSEL	—	—	—	—	PCH<3:0>				261
CM2PSEL	—	—	—	—	PCH<3:0>				261
CM3PSEL	—	—	—	—	PCH<3:0>				261
CM4PSEL	—	—	—	—	PCH<3:0>				261
CM5PSEL	—	—	—	—	PCH<3:0>				261
CM6PSEL	—	—	—	—	PCH<3:0>				261
CM7PSEL ⁽¹⁾	—	—	—	—	PCH<3:0>				261
CM8PSEL ⁽¹⁾	—	—	—	—	PCH<3:0>				261
CMOUT	MC8OUT ⁽¹⁾	MC7OUT ⁽¹⁾	MC6OUT	MC5OUT	MC4OUT	MC3OUT	MC2OUT	MC1OUT	262
FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFVR<1:0>		ADFVR<1:0>		223
DAC1CON0	EN	FM	OE1	OE2	PSS<1:0>		NSS<1:0>		249
DAC2CON0	EN	FM	OE1	OE2	PSS<1:0>		NSS<1:0>		249
DAC5CON0	EN	FM	OE1	OE2	PSS<1:0>		NSS<1:0>		249
DAC6CON0 ⁽¹⁾	EN	FM	OE1	OE2	PSS<1:0>		NSS<1:0>		249
DAC3CON0	EN	—	OE1	OE2	PSS<1:0>		NSS<1:0>		244
DAC4CON0	EN	—	OE1	OE2	PSS<1:0>		NSS<1:0>		244
DAC7CON0	EN	—	OE1	OE2	PSS<1:0>		NSS<1:0>		244
DAC8CON0 ⁽¹⁾	EN	—	OE1	OE2	PSS<1:0>		NSS<1:0>		244
DAC3REF	---	---	---	REF<4:0>					245
DAC4REF	---	---	---	REF<4:0>					245
DAC7REF	---	---	---	REF<4:0>					245
DAC8REF ⁽¹⁾	---	---	---	REF<4:0>					245
DAC1REFH	REF<9:x> (x Depends on FM bit)								250

Legend: — = unimplemented location, read as '0'. Shaded cells are unused by the comparator module.

Note 1: PIC16LF1777/9 only.

22.3 Timer1 Prescaler

Timer1 has four prescaler options allowing 1, 2, 4 or 8 divisions of the clock input. The CKPS bits of the T1CON register control the prescale counter. The prescale counter is not directly readable or writable; however, the prescaler counter is cleared upon a write to TMR1H or TMR1L.

22.4 Timer1 (Secondary) Oscillator

A dedicated low-power 32.768 kHz oscillator circuit is built-in between pins SOSC1 (input) and SOSCO (amplifier output). This internal circuit is to be used in conjunction with an external 32.768 kHz crystal.

The oscillator circuit is enabled by setting the OSCEN bit of the T1CON register. The oscillator will continue to run during Sleep.

Note: The oscillator requires a start-up and stabilization time before use. Thus, OSCEN should be set and a suitable delay observed prior to using Timer1. A suitable delay similar to the OST delay can be implemented in software by clearing the TMR1IF bit then presetting the TMR1H:TMR1L register pair to FC00h. The TMR1IF flag will be set when 1024 clock cycles have elapsed, thereby indicating that the oscillator is running and reasonably stable.

22.5 Timer1 Operation in Asynchronous Counter Mode

If the control bit $\overline{\text{SYNC}}$ of the T1CON register is set, the external clock input is not synchronized. The timer increments asynchronously to the internal phase clocks. If the external clock source is selected then the timer will continue to run during Sleep and can generate an interrupt-on-overflow, which will wake-up the processor. However, special precautions in software are needed to read/write the timer (see **Section 22.5.1 “Reading and Writing Timer1 in Asynchronous Counter Mode”**).

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

22.5.1 READING AND WRITING TIMER1 IN ASYNCHRONOUS COUNTER MODE

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

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

22.6 Timer1 Gate

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

Timer1 gate can also be driven by multiple selectable sources.

22.6.1 TIMER1 GATE ENABLE

The Timer1 Gate Enable mode is enabled by setting the GE bit of the T1GCON register. The polarity of the Timer1 Gate Enable mode is configured using the GPOL bit of the T1GCON register.

When Timer1 Gate Enable mode is enabled, Timer1 will increment on the rising edge of the Timer1 clock source. When Timer1 Gate Enable mode is disabled, no incrementing will occur and Timer1 will hold the current count. See Figure 22-3 for timing details.

TABLE 22-3: TIMER1 GATE ENABLE SELECTIONS

T1CLK	T1GPOL	T1G	Timer1 Operation
↑	1	1	Counts
↑	1	0	Holds Count
↑	0	1	Holds Count
↑	0	0	Counts

23.6 Operation Examples

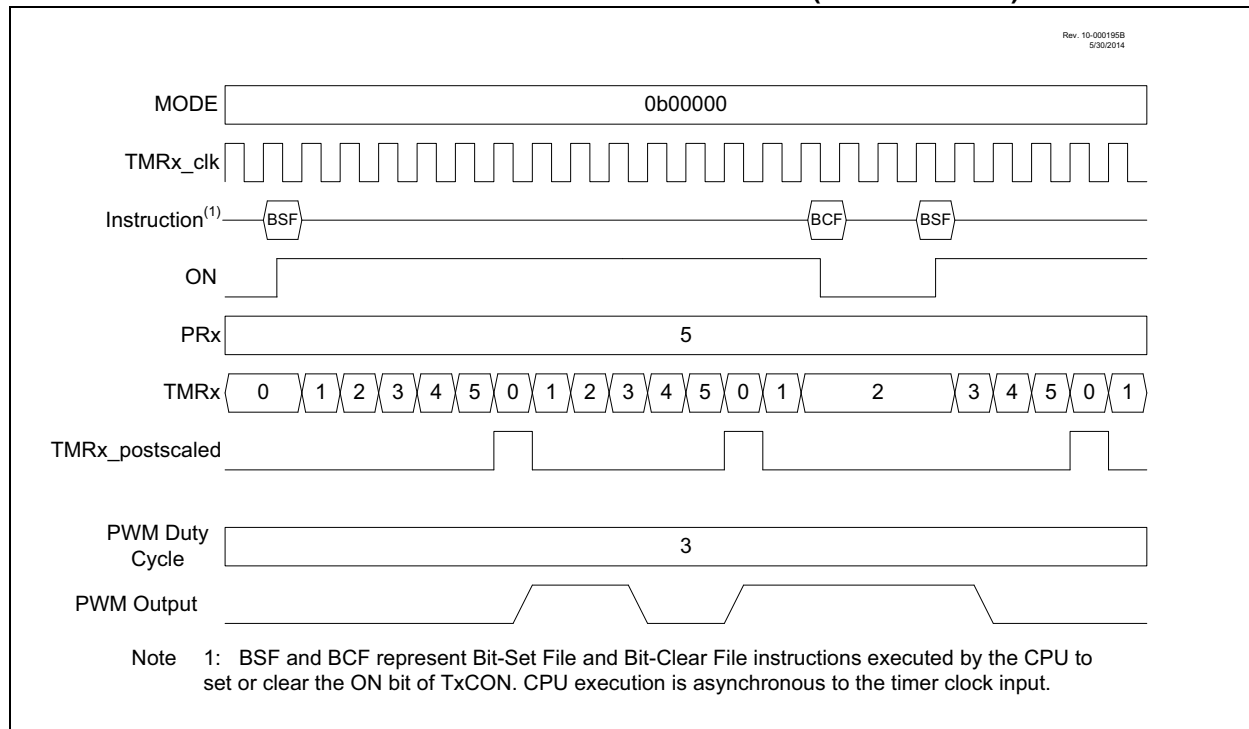
Unless otherwise specified, the following notes apply to the following timing diagrams:

- Both the prescaler and postscaler are set to 1:1 (both the CKPS and OUTPS bits in the TxCON register are cleared).
- The diagrams illustrate any clock except $F_{osc}/4$ and show clock-sync delays of at least two full cycles for both ON and Timer2_ers. When using $F_{osc}/4$, the clock-sync delay is at least one instruction period for Timer2_ers; ON applies in the next instruction period.
- The PWM Duty Cycle and PWM output are illustrated assuming that the timer is used for the PWM function of the CCP module as described in **Section 24.6 “CCP/PWM Clock Selection”**. The signals are not a part of the Timer2 module.

23.6.1 SOFTWARE GATE MODE

This mode corresponds to legacy Timer2 operation. The timer increments with each clock input when $ON = 1$ and does not increment when $ON = 0$. When the TMRx count equals the PRx period count the timer resets on the next clock and continues counting from 0. Operation with the ON bit software controlled is illustrated in Figure 23-4. With $PRx = 5$, the counter advances until $TMRx = 5$, and goes to zero with the next clock.

FIGURE 23-4: SOFTWARE GATE MODE TIMING DIAGRAM (MODE = 00000)



23.7 PR2 Period Register

The PR2 period register (T2PR) is double-buffered. Software reads and writes the PR2 register. However, the timer uses a buffered PR2 register for operation. Software does not have direct access to the buffered PR2 register. The contents of the PR2 register are transferred to the buffer by any of the following events:

- A write to the TMR2 register
- A write to the TMR2CON register
- When TMR2 = PR2 buffer and the prescaler rolls over
- An external Reset event

23.8 Timer2 Operation During Sleep

When PSYNC = 1, Timer2 cannot be operated while the processor is in Sleep mode. The contents of the TMR2 and T2PR registers will remain unchanged while processor is in Sleep mode.

When PSYNC = 0, Timer2 will operate in Sleep as long as the clock source selected is also still running. Selecting the LFINTOSC, MFINTOSC, or HFINTOSC oscillator as the timer clock source will keep the selected oscillator running during Sleep.

26.2 PWM Modes

PWM modes are selected with MODE<1:0> bits of the PWMxCON register (Register 26-1).

In all PWM modes an offset match event can also be used to synchronize the PWMxTMR in three offset modes. See **Section 26.3 “Offset Modes”** for more information.

26.2.1 STANDARD MODE

The Standard mode (MODE = 00) selects a single phase PWM output. The PWM output in this mode is determined by when the period, duty cycle, and phase counts match the PWMxTMR value. The start of the duty cycle occurs on the phase match and the end of the duty cycle occurs on the duty cycle match. The period match resets the timer. The offset match can also be used to synchronize the PWMxTMR in the offset modes. See **Section 26.3 “Offset Modes”** for more information.

Equation 26-1 is used to calculate the PWM period in Standard mode.

Equation 26-2 is used to calculate the PWM duty cycle ratio in Standard mode.

EQUATION 26-1: PWM PERIOD IN STANDARD MODE

$$Period = \frac{(PWMxPR + 1) \cdot Prescale}{PWM_clock}$$

EQUATION 26-2: PWM DUTY CYCLE IN STANDARD MODE

$$Duty\ Cycle = \frac{(PWMxDC - PWMxPH)}{PWMxPR + 1}$$

A detailed timing diagram for Standard mode is shown in Figure 26-4.

26.2.2 SET ON MATCH MODE

The Set On Match mode (MODE = 01) generates an active output when the phase count matches the PWMxTMR value. The output stays active until the OUT bit of the PWMxCON register is cleared or the PWM module is disabled. The duty cycle count has no effect in this mode. The period count only determines the maximum PWMxTMR value above which no phase matches can occur.

The PWMxOUT bit can be used to set or clear the output of the PWM in this mode. Writes to this bit will take place on the next rising edge of the PWM_clock after the bit is written.

A detailed timing diagram for Set On Match is shown in Figure 26-5.

26.2.3 TOGGLE ON MATCH MODE

The Toggle On Match mode (MODE = 10) generates a 50% duty cycle PWM with a period twice as long as that computed for the standard PWM mode. Duty cycle count has no effect in this mode. The phase count determines how many PWMxTMR periods after a period event the output will toggle.

Writes to the OUT bit of the PWMxCON register will have no effect in this mode.

A detailed timing diagram for Toggle On Match is shown in Figure 26-6.

26.2.4 CENTER ALIGNED MODE

The Center Aligned mode (MODE = 11) generates a PWM waveform that is centered in the period. In this mode the period is two times the PWMxPR count. The PWMxTMR counts up to the period value then counts back down to 0. The duty cycle count determines both the start and end of the active PWM output. The start of the duty cycle occurs at the match event when PWMxTMR is incrementing and the duty cycle ends at the match event when PWMxTMR is decrementing. The incrementing match value is the period count minus the duty cycle count. The decrementing match value is the incrementing match value plus 1.

Equation 26-3 is used to calculate the PWM period in Center Aligned mode.

EQUATION 26-3: PWM PERIOD IN CENTER ALIGNED MODE

$$Period = \frac{(PWMxPR + 1) \cdot 2 \cdot Prescale}{PWM_clock}$$

Equation 26-4 is used to calculate the PWM duty cycle ratio in Center Aligned mode

EQUATION 26-4: PWM DUTY CYCLE IN CENTER ALIGNED MODE

$$Duty\ Cycle = \frac{PWMxDC \cdot 2}{(PWMxPR + 1) \cdot 2}$$

Writes to PWMxOUT will have no effect in this mode.

A detailed timing diagram for Center Aligned mode is shown in Figure 26-7.

27.0 COMPLEMENTARY OUTPUT GENERATOR (COG) MODULES

The primary purpose of the Complementary Output Generator (COG) is to convert a single-output PWM signal into a two-output complementary PWM signal. The COG can also convert two separate input events into a single or complementary PWM output.

TABLE 27-1: AVAILABLE COG MODULES

Device	COG1	COG2	COG3	COG4
PIC16(L)F1778	•	•	•	
PIC16(L)F1777/9	•	•	•	•

The COG PWM frequency and duty cycle are determined by a rising event input and a falling event input. The rising event and falling event may be the same source. Sources may be synchronous or asynchronous to the COG_clock.

The rate at which the rising event occurs determines the PWM frequency. The time from the rising event to the falling event determines the duty cycle.

A selectable clock input is used to generate the phase delay, blanking, and dead-band times. Dead-band time can also be generated with a programmable delay chain, which is independent from all clock sources.

Simplified block diagrams of the various COG modes are shown in Figure 27-2 through Figure 27-6.

The COG module has the following features:

- Six modes of operation:
 - Steered PWM mode
 - Synchronous Steered PWM mode
 - Forward Full-Bridge mode
 - Reverse Full-Bridge mode
 - Half-Bridge mode
 - Push-Pull mode
- Selectable COG_clock clock source
- Independently selectable rising event sources
- Independently selectable falling event sources
- Independently selectable edge or level event sensitivity
- Independent output polarity selection
- Phase delay with independent rising and falling delay times
- Dead-band control with:
 - independent rising and falling event dead-band times
 - Synchronous and asynchronous timing
- Blanking control with independent rising and falling event blanking times
- Auto-shutdown control with:
 - Independently selectable shutdown sources
 - Auto-restart enable
 - Auto-shutdown pin override control (high, low, off, and High-Z)

27.1 Output to Pins (all modes)

The COG peripheral has four outputs: COGA, COGB, COGC, and COGD.

The operating mode, selected with the MD<2:0> bits of the COGxCON0 register, determine the waveform available at each output. An individual peripheral source control for each device pin selects the pin or pins at which the outputs will appear. Please refer to the RxyPPS register (Register 12-2) for more information.

27.2 Event-Driven PWM (All Modes)

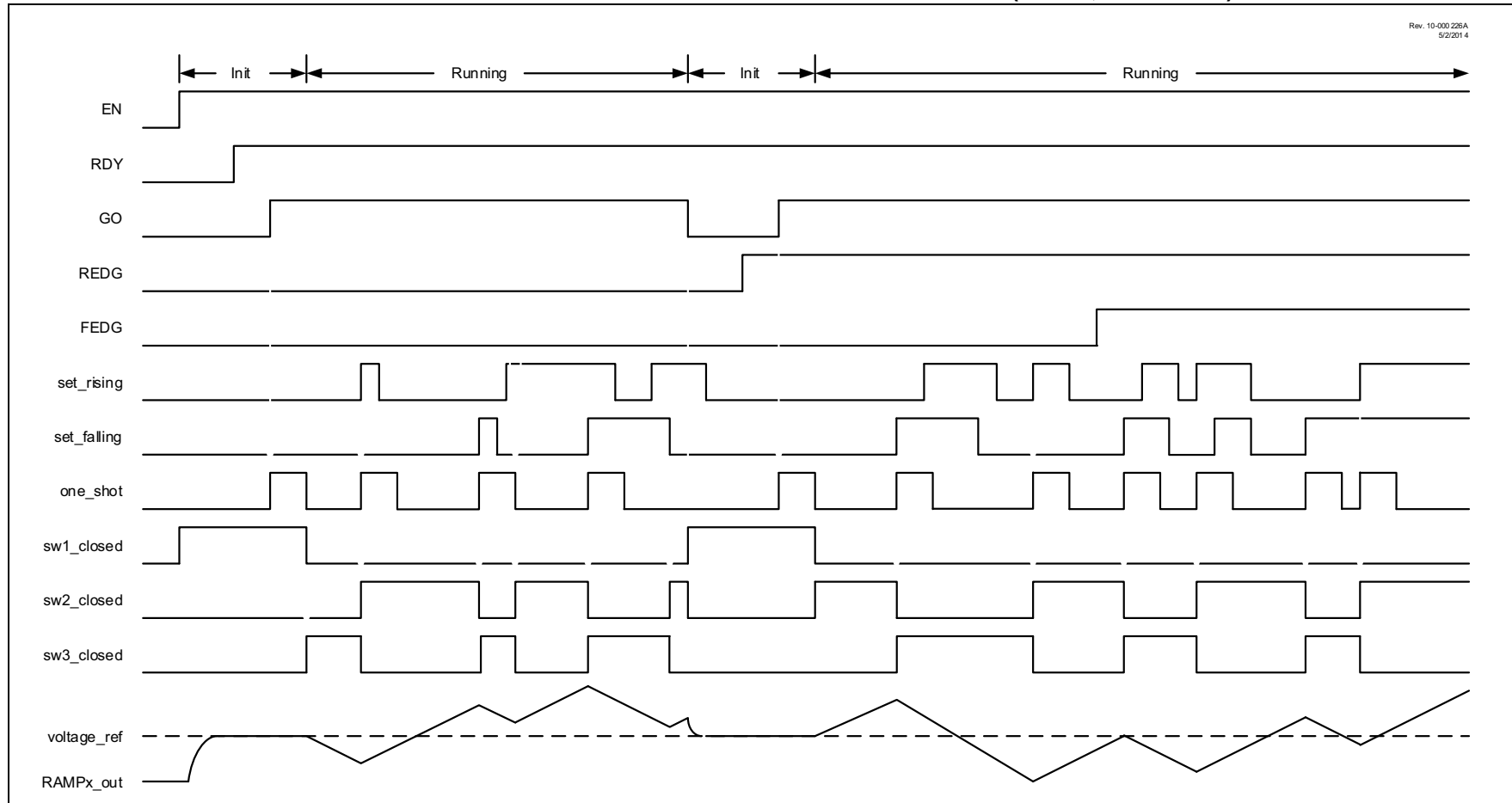
Besides generating PWM and complementary outputs from a single PWM input, the COG can also generate PWM waveforms from a periodic rising event and a separate falling event. In this case, the falling event is usually derived from analog feedback within the external PWM driver circuit. In this configuration, high-power switching transients may trigger a false falling event that needs to be blanked out. The COG can be configured to blank falling (and rising) event inputs for a period of time immediately following the rising (and falling) event drive output. This is referred to as input blanking and is covered in **Section 27.8 “Blanking Control”**.

It may be necessary to guard against the possibility of external circuit faults. In this case, the active drive must be terminated before the Fault condition causes damage. This is referred to as auto-shutdown and is covered in **Section 27.10 “Auto-Shutdown Control”**.

The COG can be configured to operate in phase delayed conjunction with another PWM. The active drive cycle is delayed from the rising event by a phase delay timer. Phase delay is covered in more detail in **Section 27.9 “Phase Delay”**.

A typical operating waveform, with phase delay and dead band, generated from a single CCP1 input is shown in Figure 27-10.

FIGURE 30-4: ALTERNATING RISING/FALLING RAMP GENERATION TIMING DIAGRAM (OS = 1, MODE = 01)



32.6.7 I²C MASTER MODE RECEPTION

Master mode reception (Figure 32-29) is enabled by programming the Receive Enable bit, RCEN bit of the SSPxCON2 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 SSPxBUF, the BF flag bit is set, the SSPxIF 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 SSPxCON2 register.

32.6.7.1 BF Status Flag

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

32.6.7.2 SSPOV Status Flag

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

32.6.7.3 WCOL Status Flag

If the user writes the SSPxBUF 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).

32.6.7.4 Typical Receive Sequence:

1. The user generates a Start condition by setting the SEN bit of the SSPxCON2 register.
2. SSPxIF is set by hardware on completion of the Start.
3. SSPxIF is cleared by software.
4. User writes SSPxBUF with the slave address to transmit and the R/W bit set.
5. Address is shifted out the SDA pin until all eight bits are transmitted. Transmission begins as soon as SSPxBUF is written to.
6. The MSSP module shifts in the $\overline{\text{ACK}}$ bit from the slave device and writes its value into the ACKSTAT bit of the SSPxCON2 register.
7. The MSSP module generates an interrupt at the end of the ninth clock cycle by setting the SSPxIF bit.
8. User sets the RCEN bit of the SSPxCON2 register and the master clocks in a byte from the slave.
9. After the eighth falling edge of SCL, SSPxIF and BF are set.
10. Master clears SSPxIF and reads the received byte from SSPxBUF, clears BF.
11. Master sets $\overline{\text{ACK}}$ value sent to slave in ACKDT bit of the SSPxCON2 register and initiates the $\overline{\text{ACK}}$ by setting the ACKEN bit.
12. Master's $\overline{\text{ACK}}$ is clocked out to the slave and SSPxIF is set.
13. User clears SSPxIF.
14. Steps 8-13 are repeated for each received byte from the slave.
15. Master sends a not $\overline{\text{ACK}}$ or Stop to end communication.

PIC16(L)F1777/8/9

35.0 INSTRUCTION SET SUMMARY

Each instruction is a 14-bit word containing the operation code (opcode) and all required operands. The opcodes are broken into three broad categories.

- Byte Oriented
- Bit Oriented
- Literal and Control

The literal and control category contains the most varied instruction word format.

Table 35-3 lists the instructions recognized by the MPASM™ assembler.

All instructions are executed within a single instruction cycle, with the following exceptions, which may take two or three cycles:

- Subroutine takes two cycles (CALL, CALLW)
- Returns from interrupts or subroutines take two cycles (RETURN, RETLW, RETFIE)
- Program branching takes two cycles (GOTO, BRA, BRW, BTFSS, BTFSC, DECFSZ, INCSFZ)
- One additional instruction cycle will be used when any instruction references an indirect file register and the file select register is pointing to program memory.

One instruction cycle consists of four oscillator cycles; for an oscillator frequency of 4 MHz, this gives a nominal instruction execution rate of 1 MHz.

All instruction examples use the format '0xhh' to represent a hexadecimal number, where 'h' signifies a hexadecimal digit.

35.1 Read-Modify-Write Operations

Any instruction that specifies a file register as part of the instruction performs a Read-Modify-Write (R-M-W) operation. The register is read, the data is modified, and the result is stored according to either the instruction, or the destination designator 'd'. A read operation is performed on a register even if the instruction writes to that register.

TABLE 35-1: OPCODE FIELD DESCRIPTIONS

Field	Description
f	Register file address (0x00 to 0x7F)
W	Working register (accumulator)
b	Bit address within an 8-bit file register
k	Literal field, constant data or label
x	Don't care location (= 0 or 1). The assembler will generate code with x = 0. It is the recommended form of use for compatibility with all Microchip software tools.
d	Destination select; d = 0: store result in W, d = 1: store result in file register f. Default is d = 1.
n	FSR or INDF number. (0-1)
mm	Pre-post increment-decrement mode selection

TABLE 35-2: ABBREVIATION DESCRIPTIONS

Field	Description
PC	Program Counter
\overline{TO}	Time-Out bit
C	Carry bit
DC	Digit Carry bit
Z	Zero bit
\overline{PD}	Power-Down bit

BCF	Bit Clear f
Syntax:	[<i>label</i>] BCF f,b
Operands:	$0 \leq f \leq 127$ $0 \leq b \leq 7$
Operation:	$0 \rightarrow (f < b >)$
Status Affected:	None
Description:	Bit 'b' in register 'f' is cleared.

BTFSC	Bit Test f, Skip if Clear
Syntax:	[<i>label</i>] BTFSC f,b
Operands:	$0 \leq f \leq 127$ $0 \leq b \leq 7$
Operation:	skip if $(f < b >) = 0$
Status Affected:	None
Description:	If bit 'b' in register 'f' is '1', the next instruction is executed. If bit 'b', in register 'f', is '0', the next instruction is discarded, and a NOP is executed instead, making this a 2-cycle instruction.

BRA	Relative Branch
Syntax:	[<i>label</i>] BRA label [<i>label</i>] BRA \$+k
Operands:	$-256 \leq \text{label} - \text{PC} + 1 \leq 255$ $-256 \leq k \leq 255$
Operation:	$(\text{PC}) + 1 + k \rightarrow \text{PC}$
Status Affected:	None
Description:	Add the signed 9-bit literal 'k' to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be $\text{PC} + 1 + k$. This instruction is a 2-cycle instruction. This branch has a limited range.

BTFSS	Bit Test f, Skip if Set
Syntax:	[<i>label</i>] BTFSS f,b
Operands:	$0 \leq f \leq 127$ $0 \leq b < 7$
Operation:	skip if $(f < b >) = 1$
Status Affected:	None
Description:	If bit 'b' in register 'f' is '0', the next instruction is executed. If bit 'b' is '1', then the next instruction is discarded and a NOP is executed instead, making this a 2-cycle instruction.

BRW	Relative Branch with W
Syntax:	[<i>label</i>] BRW
Operands:	None
Operation:	$(\text{PC}) + (W) \rightarrow \text{PC}$
Status Affected:	None
Description:	Add the contents of W (unsigned) to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be $\text{PC} + 1 + (W)$. This instruction is a 2-cycle instruction.

BSF	Bit Set f
Syntax:	[<i>label</i>] BSF f,b
Operands:	$0 \leq f \leq 127$ $0 \leq b \leq 7$
Operation:	$1 \rightarrow (f < b >)$
Status Affected:	None
Description:	Bit 'b' in register 'f' is set.

FIGURE 36-17: SPI MASTER MODE TIMING (CKE = 0, SMP = 0)

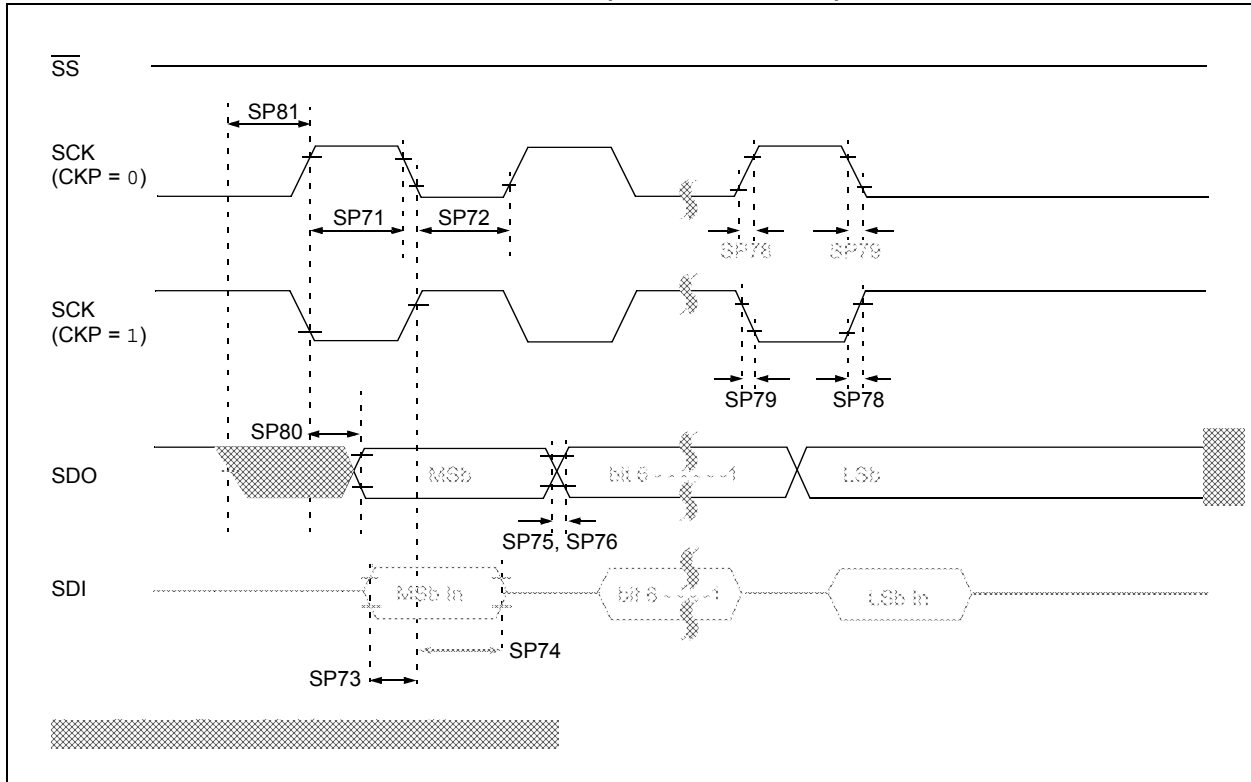
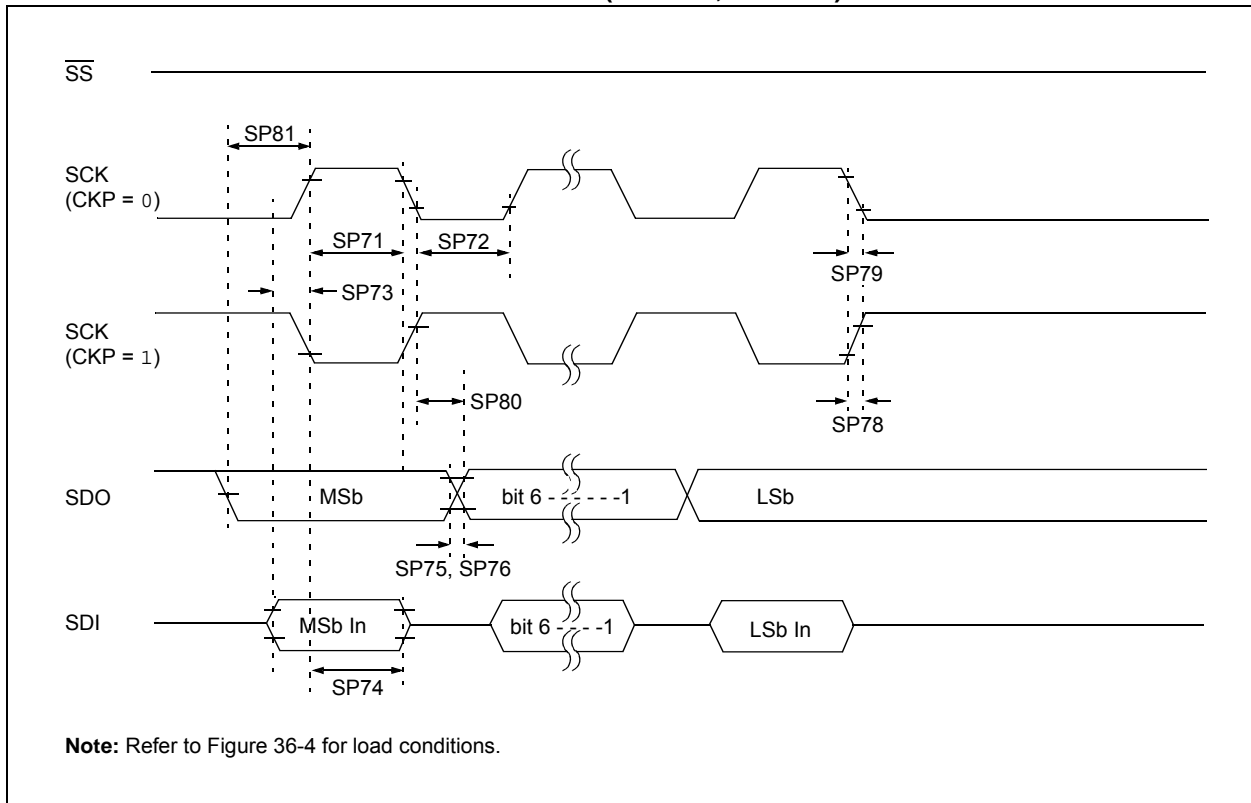


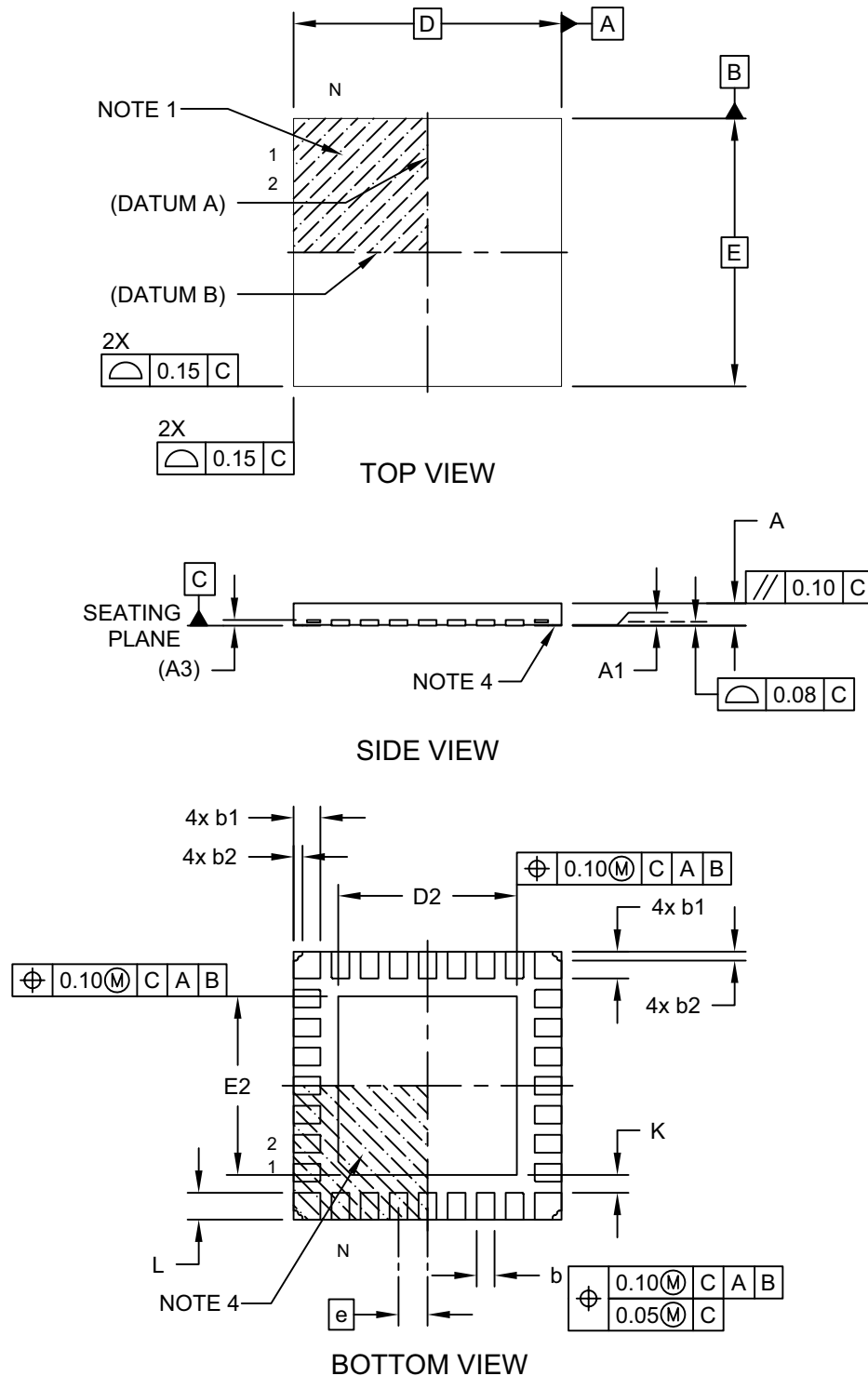
FIGURE 36-18: SPI MASTER MODE TIMING (CKE = 1, SMP = 1)



PIC16(L)F1777/8/9

28-Lead Plastic Quad Flat, No Lead Package (MX) - 6x6x0.5mm Body [UQFN] Ultra-Thin with 0.40 x 0.60 mm Terminal Width/Length and Corner Anchors

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

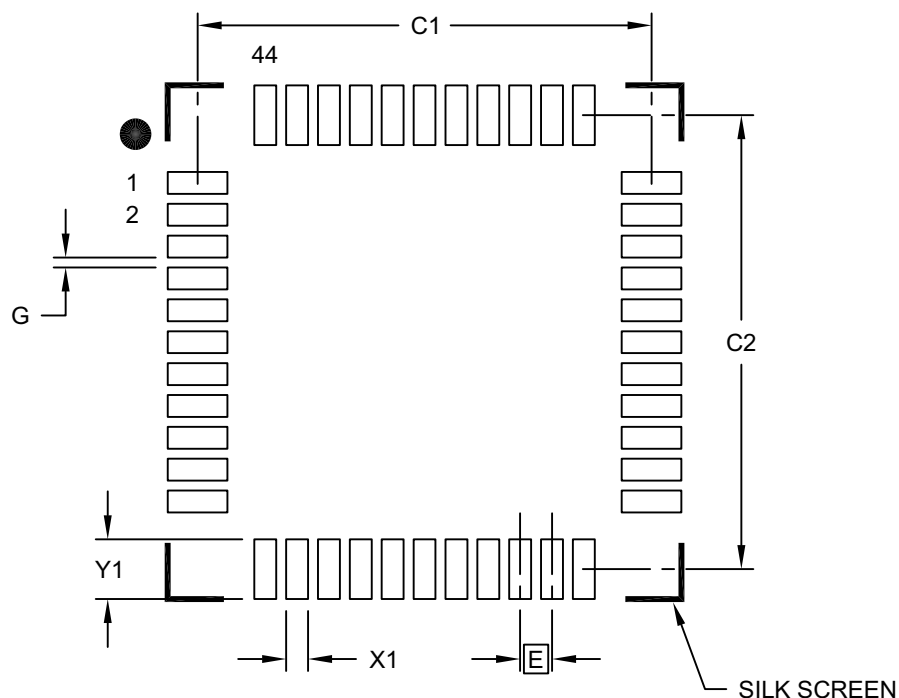


Microchip Technology Drawing C04-0209 Rev C Sheet 1 of 2

PIC16(L)F1777/8/9

44-Lead Plastic Thin Quad Flatpack (PT) - 10X10X1 mm Body, 2.00 mm Footprint [TQFP]

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



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E	0.80 BSC		
Contact Pad Spacing	C1		11.40	
Contact Pad Spacing	C2		11.40	
Contact Pad Width (X44)	X1			0.55
Contact Pad Length (X44)	Y1			1.50
Distance Between Pads	G	0.25		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2076B