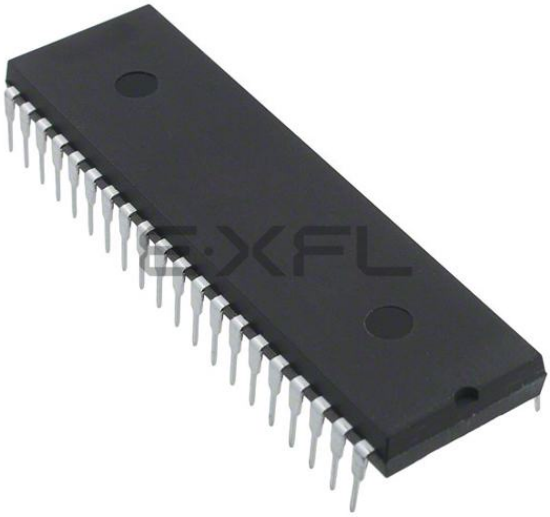


Welcome to E-XFL.COM

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	28KB (16K x 14)
Program Memory Type	FLASH
EEPROM Size	128 x 8
RAM Size	2K 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 ~ 125°C (TA)
Mounting Type	Through Hole
Package / Case	40-DIP (0.600", 15.24mm)
Supplier Device Package	40-PDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f1779-e-p

TABLE 4: 40/44-PIN ALLOCATION TABLE (PIC16(L)F1777/9) (CONTINUED)

I/O	40-Pin PDIP	40-Pin (U)QFN	44-Pin TQFP	44-Pin QFN	ADC	VREF	DAC	Op Amp	Comparator	ZCD	PRG	Timers	PWM	CCP	COG	CLC	Modulator	EUSART	MSSP	Interrupt	Pull-ups	High Current	Basic
RE0	8	23	25	25	AN5	DAC6REF1+ DAC8REF1+	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Y	—	—
RE1	9	24	26	26	AN6	DAC6REF1- DAC8REF1-	DAC6OUT1	—	—	—	—	—	—	—	—	—	—	—	—	—	Y	—	—
RE2	10	25	27	27	AN7	—	DAC8OUT1	—	—	—	—	—	—	—	—	—	—	—	—	—	Y	—	—
RE3	1	16	18	18	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	IOC	Y	—	$\overline{MCLR}$ V_{PP}
V_{DD}	11	7	7	7,8	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	V_{DD}
V_{DD}	32	26	28	28	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	V_{DD}
V_{SS}	12	6	6	6	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	V_{SS}
V_{SS}	31	27	29	30	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	V_{SS}
OUT(2)	—	—	—		—	—	—	—	C1OUT C2OUT C3OUT C4OUT C5OUT C6OUT C7OUT C8OUT	—	—	—	PWM3 PWM4 PWM5 PWM6 PWM9 PWM10 PWM11 PWM12	CCP1 CCP2 CCP7 CCP8	COG1A COG1B COG1C COG1D COG2A COG2B COG2C COG2D COG3A COG3B COG3C COG3D COG4A COG4B COG4C COG4D	CLC1OUT CLC2OUT CLC3OUT CLC4OUT	MD1OUT MD2OUT MD3OUT MD4OUT	DT(3) TX CK	SDO SDA(3) SCK SCL(3)	—	—	—	—

Note 1: Default peripheral input. Input can be moved to any other pin with the PPS input selection register.
2: All pin outputs default to PORT latch data. Any pin can be selected as a digital peripheral output with the PPS output selection registers.
3: These peripheral functions are bidirectional. The output pin selections must be the same as the input pin selections.

TABLE 1-3: PIC16(L)F1777/9 PINOUT DESCRIPTION (CONTINUED)

Name	Function	Input Type	Output Type	Description
RB2/AN8/OPA2IN0-/ DAC3OUT1/PRG4F/COG3IN/ MD4MOD	RB2	TTL/ST	CMOS	General purpose I/O.
	AN8	AN	—	ADC Channel 8 input.
	OPA2IN0-	AN	—	Operational amplifier 2 inverting input.
	DAC3OUT1	—	AN	DAC3 voltage output.
	PRG4F(1)	TTL/ST	—	Ramp generator set_falling input.
	COG3IN(1)	TTL/ST	—	Complementary output generator 3 input.
	MD4MOD(1)	TTL/ST	—	Data signal modulator modulation input.
RB3/AN9/C1IN2-/C2IN2-/ C3IN2-/OPA2IN0+/MD3CL	RB3	TTL/ST	CMOS	General purpose I/O.
	AN9	AN	—	ADC Channel 9 input.
	C1IN2-	AN	—	Comparator 1 negative input.
	C2IN2-	AN	—	Comparator 2 negative input.
	C3IN2-	AN	—	Comparator 3 negative input.
	OPA2IN0+	AN		Operational amplifier 2 non-inverting input.
	MD3CL(1)	TTL/ST	—	Data signal modulator 3 low carrier input.
RB4/AN11/C3IN1+/MD3CH	RB4	TTL/ST	CMOS	General purpose I/O.
	AN11	AN	—	ADC Channel 11 input.
	C3IN1+	AN	—	Comparator 3 positive input.
	MD3CH(1)	TTL/ST	—	Data signal modulator 3 high carrier input.
RB5/AN13/DAC5REF1-/ DAC7REF1-/C4IN2-/CCP7/ MD3MOD	RB5	TTL/ST	CMOS	General purpose I/O.
	AN13	AN	—	ADC Channel 11 input.
	DAC5REF1-	AN	—	DAC5 negative reference.
	DAC7REF1-	AN	—	DAC7 negative reference.
	C4IN2-	AN	—	Comparator 4 negative input.
	CCP7(1)	TTL/ST	—	CCP7 capture input.
	MD3MOD(1)	TTL/ST	—	Data signal modulator modulation input.
RB6/DAC5REF1+/DAC7REF1+/ C4IN1+/CLCIN2/ICSPCLK	RB6	TTL/ST	CMOS	General purpose I/O.
	DAC5REF1+	AN	—	DAC5 positive reference.
	DAC7REF1+	AN	—	DAC7 positive reference.
	C4IN1+	AN	—	Comparator 2 positive input.
	CLCIN2(1)	TTL/ST	—	CLC input 2.
	ICSPCLK	ST	—	Serial Programming Clock.

Legend: AN = Analog input or output CMOS = CMOS compatible input or output OD = Open-Drain
TTL = TTL compatible input ST = Schmitt Trigger input with CMOS levels I^2C = Schmitt Trigger input with I^2C
HP = High Power XTAL = Crystal levels

Note 1: Default peripheral input. Alternate pins can be selected as the peripheral input with the PPS input selection registers.
2: All pin digital outputs default to PORT latch data. Alternate outputs can be selected as the peripheral digital output with the PPS output selection registers.
3: These peripheral functions are bidirectional. The output pin selections must be the same as the input pin selections.

TABLE 3-7: PIC16(L)F1779 MEMORY MAP, BANK 8-15

BANK 8		BANK 9		BANK 10		BANK 11		BANK 12		BANK 13		BANK 14		BANK 15	
400h	Core Registers (Table 3-2)	480h	Core Registers (Table 3-2)	500h	Core Registers (Table 3-2)	580h	Core Registers (Table 3-2)	600h	Core Registers (Table 3-2)	680h	Core Registers (Table 3-2)	700h	Core Registers (Table 3-2)	780h	Core Registers (Table 3-2)
40Bh		48Bh		50Bh		58Bh		60Bh		68Bh		70Bh		78Bh	
40Ch	—	48Ch	—	50Ch	—	58Ch	—	60Ch	DAC8CON0	68Ch	—	70Ch	—	78Ch	—
40Dh	HIDRVB	48Dh	—	50Dh	—	58Dh	DACLD	60Dh	DAC8REFL	68Dh	COG1PHR	70Dh	COG2PHR	78Dh	—
40Eh	—	48Eh	ADRESL	50Eh	—	58Eh	DAC1CON0	60Eh	PRG4RTSS	68Eh	COG1PHF	70Eh	COG2PHF	78Eh	PRG1RTSS
40Fh	TMR5L	48Fh	ADRESH	50Fh	OPA1NCHS	58Fh	DAC1REFL	60Fh	PRG4FTSS	68Fh	COG1BLKR	70Fh	COG2BLKR	78Fh	PRG1FTSS
410h	TMR5H	490h	ADCON0	510h	OPA1PCHS	590h	DAC1REFH	610h	PRG4INS	690h	COG1BLKF	710h	COG2BLKF	790h	PRG1INS
411h	T5CON	491h	ADCON1	511h	OPA1CON	591h	DAC2CON0	611h	PRG4CON0	691h	COG1DBR	711h	COG2DBR	791h	PRG1CON0
412h	T5GCON	492h	ADCON2	512h	OPA1ORS	592h	DAC2REFL	612h	PRG4CON1	692h	COG1DBF	712h	COG2DBF	792h	PRG1CON1
413h	T4TMR	493h	T2TMR	513h	OPA2NCHS	593h	DAC2REFH	613h	PRG4CON2	693h	COG1CON0	713h	COG2CON0	793h	PRG1CON2
414h	T4PR	494h	T2PR	514h	OPA2PCHS	594h	DAC3CON0	614h	PWM3DCL	694h	COG1CON1	714h	COG2CON1	794h	PRG2RTSS
415h	T4CON	495h	T2CON	515h	OPA2CON	595h	DAC3REF	615h	PWM3DCH	695h	COG1RIS0	715h	COG2RIS0	795h	PRG2FTSS
416h	T4HLT	496h	T2HLT	516h	OPA2ORS	596h	DAC4CON0	616h	PWM3CON	696h	COG1RIS1	716h	COG2RIS1	796h	PRG2INS
417h	T4CLKCON	497h	T2CLKCON	517h	OPA3NCHS	597h	DAC4REF	617h	PWM4DCL	697h	COG1RSIM0	717h	COG2RSIM0	797h	PRG2CON0
418h	T4RST	498h	T2RST	518h	OPA3PCHS	598h	DAC5CON0	618h	PWM4DCH	698h	COG1RSIM1	718h	COG2RSIM1	798h	PRG2CON1
419h	—	499h	—	519h	OPA3CON	599h	DAC5REFL	619h	PWM4CON	699h	COG1FIS0	719h	COG2FIS0	799h	PRG2CON2
41Ah	T6TMR	49Ah	T8TMR	51Ah	OPA3ORS	59Ah	DAC5REFH	61Ah	PWM9DCL	69Ah	COG1FIS1	71Ah	COG2FIS1	79Ah	PRG3RTSS
41Bh	T6PR	49Bh	T8PR	51Bh	OPA4NCHS	59Bh	DAC6CON0	61Bh	PWM9DCH	69Bh	COG1FSIM0	71Bh	COG2FSIM0	79Bh	PRG3FTSS
41Ch	T6CON	49Ch	T8CON	51Ch	OPA4PCHS	59Ch	DAC6REFL	61Ch	PWM9CON	69Ch	COG1FSIM1	71Ch	COG2FSIM1	79Ch	PRG3INS
41Dh	T6HLT	49Dh	T8HLT	51Dh	OPA4CON	59Dh	DAC6REFH	61Dh	PWM10DCL	69Dh	COG1ASD0	71Dh	COG2ASD0	79Dh	PRG3CON0
41Eh	T6CLKCON	49Eh	T8CLKCON	51Eh	OPA4ORS	59Eh	DAC7CON0	61Eh	PWM10DCH	69Eh	COG1ASD1	71Eh	COG2ASD1	79Eh	PRG3CON1
41Fh	T6RST	49Fh	T8RST	51Fh	—	59Fh	DA73REF	61Fh	PWM10CON	69Fh	COG1STR	71Fh	COG2STR	79Fh	PRG3CON2
420h	General Purpose Register 80 Bytes	4A0h	General Purpose Register 80 Bytes	520h	General Purpose Register 80 Bytes	5A0h	General Purpose Register 80 Bytes	620h	General Purpose Register 80 Bytes	6A0h	General Purpose Register 80 Bytes	720h	General Purpose Register 80 Bytes	7A0h	General Purpose Register 80 Bytes
46Fh		4EFh		56Fh		5EFh		66Fh		6EFh		76Fh		7EFh	
470h	Accesses 70h – 7Fh	4F0h	Accesses 70h – 7Fh	570h	Accesses 70h – 7Fh	5F0h	Accesses 70h – 7Fh	670h	Accesses 70h – 7Fh	6F0h	Accesses 70h – 7Fh	770h	Accesses 70h – 7Fh	7F0h	Accesses 70h – 7Fh
47Fh		4FFh		57Fh		5FFh		67Fh		6FFh		77Fh		7FFh	

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

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 12											
60Ch to 613h	—	Unimplemented								—	—
60Ch	DAC8CON0(3)	EN	—	OE1	OE2	PSS<1:0>		NSS1	NSS0	0-00 0000	0-00 0000
60Dh	DAC8REF(3)	—	—	—	REF<4:0>					---0 0000	0000 0000
60Eh	PRG4RTSS(3)	—	—	—	—	RTSS<3:0>				---- 0000	---- 0000
60Fh	PRG4FTSS(3)	—	—	—	—	FTSS<3:0>				---- 0000	---- 0000
610h	PRG4INS(3)	—	—	—	—	INS<3:0>				---- 0000	---- 0000
611h	PRG4CON0(3)	EN	—	FEDG	REDG	MODE<1:0>		OS	GO	0-000 0000	0-00 0000
612h	PRG4CON1(3)	—	—	—	—	—	RDY	FPOL	RPOL	---- -000	---- -000
613h	PRG4CON2(3)	—	—	—	ISET<4:0>					---0 0000	---0 0000
614h	PWM3DCL	DC<1:0>		—	—	—	—	—	—	xx-- ----	uu-- ----
615h	PWM3DCH	DC<9:2>								xxxx xxxx	uuuu uuuu
616h	PWM3CON	EN	—	OUT	POL	—	—	—	—	0-00 ----	0-00 ----
617h	PWM4DCL	DC<1:0>		—	—	—	—	—	—	xx-- ----	uu-- ----
618h	PWM4DCH	DC<9:2>								xxxx xxxx	uuuu uuuu
619h	PWM4CON	EN	—	OUT	POL	—	—	—	—	0-00 ----	0-00 ----
61Ah	PWM9DCL	DC<1:0>		—	—	—	—	—	—	xx-- ----	uu-- ----
61Bh	PWM9DCH	DC<9:2>								xxxx xxxx	uuuu uuuu
61Ch	PWM9CON	EN	—	OUT	POL	—	—	—	—	0-00 ----	0-00 ----
61Dh	PWM10DCL(3)	DC<1:0>		—	—	—	—	—	—	xx-- ----	uu-- ----
61Eh	PWM10DCH(3)	DC<9:2>								xxxx xxxx	uuuu uuuu
61Fh	PWM10CON(3)	EN	—	OUT	POL	—	—	—	—	0-00 ----	0-00 ----

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 13											
68Ch	—	Unimplemented								—	—
68Dh	COG1PHR	—	—	COG Rising Edge Phase Delay Count Register						--00 0000	--00 0000
68Eh	COG1PHF	—	—	COG Falling Edge Phase Delay Count Register						--00 0000	--00 0000
68Fh	COG1BLKR	—	—	COG Rising Edge Blanking Count Register						--00 0000	--00 0000
690h	COG1BLKF	—	—	COG Falling Edge Blanking Count Register						--00 0000	--00 0000
691h	COG1DBR	—	—	COG Rising Edge Dead-band Count Register						--00 0000	--00 0000
692h	COG1DBF	—	—	COG Falling Edge Dead-band Count Register						--00 0000	--00 0000
693h	COG1CON0	EN	LD	—	CS<1:0>		MD<2:0>			00-0 0000	00-0 0000
694h	COG1CON1	RDBS	FDBS	—	—	POLD	POLC	POLB	POLA	00-- 0000	00-- 0000
695h	COG1RIS0	RIS7	RIS6	RIS5	RIS4	RIS3	RIS2	RIS1	RIS0	0000 0000	0000 0000
696h	COG1RIS1	RIS15	RIS14	RIS13	RIS12	RIS11	RIS10	RIS9	RIS8	0000 0000	0000 0000
697h	COG1RSIM0	RSIM7	RSIM6	RSIM5	RSIM4	RSIM3	RSIM2	RSIM1	RSIM0	0000 0000	0000 0000
698h	COG1RSIM1	RSIM15	RSIM14	RSIM13	RSIM12	RSIM11	RSIM10	RSIM9	RSIM8	0000 0000	0000 0000
699h	COG1FIS0	FIS7	FIS6	FIS5	FIS4	FIS3	FIS2	FIS1	FIS0	0000 0000	0000 0000
69Ah	COG1FIS1	FIS15	FIS14	FIS13	FIS12	FIS11	FIS10	FIS9	FIS8	0000 0000	0000 0000
69Bh	COG1FSIM0	FSIM7	FSIM6	FSIM5	FSIM4	FSIM3	FSIM2	FSIM1	FSIM0	0000 0000	0000 0000
69Ch	COG1FSIM1	FSIM15	FSIM14	FSIM13	FSIM12	FSIM11	FSIM10	FSIM9	FSIM8	0000 0000	0000 0000
69Dh	COG1ASD0	ASE	ARSEN	ASDBD<1:0>		ASDAC<1:0>		—	—	0001 01--	0001 01--
69Eh	COG1ASD1	AS7E	AS6E	AS5E	AS4E	AS3E	AS2E	AS1E	AS0E	0000 0000	0000 0000
69Fh	COG1STR	SDATD	SDATC	SDATB	SDATA	STRD	STRC	STRB	STRA	0000 0000	0000 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 10-1: FLASH MEMORY ORGANIZATION BY DEVICE

Device	Row Erase (words)	Write Latches (words)
PIC16(L)F1778	32	32
PIC16(L)F1777/9		

10.2.1 READING THE FLASH PROGRAM MEMORY

To read a program memory location, the user must:

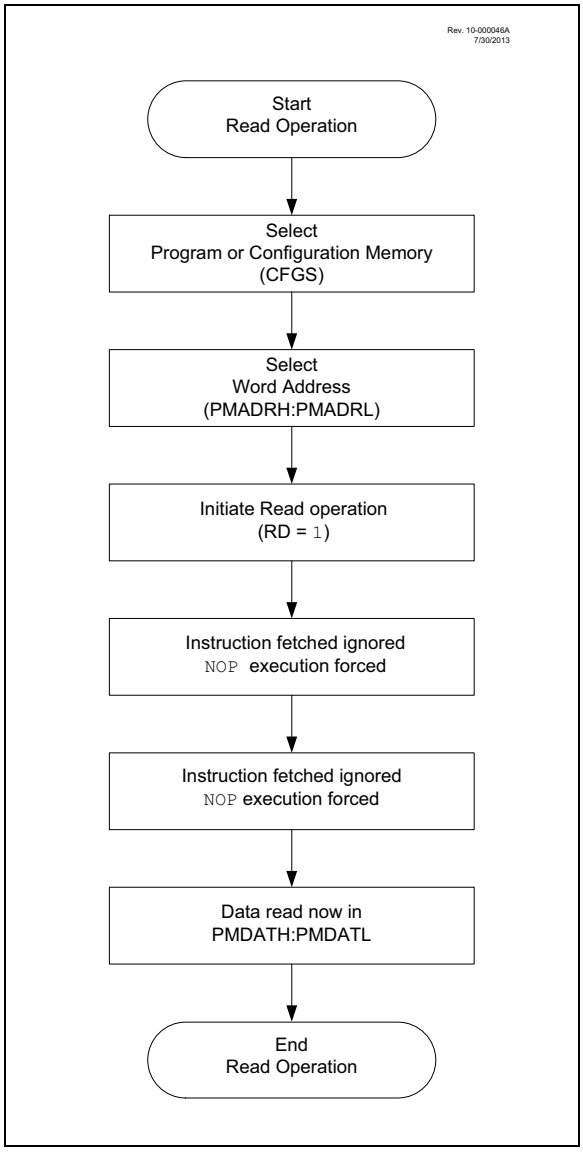
1. Write the desired address to the PMADRH:PMADRL register pair.
2. Clear the CFGS bit of the PMCON1 register.
3. Then, set control bit RD of the PMCON1 register.

Once the read control bit is set, the program memory Flash controller will use the second instruction cycle to read the data. This causes the second instruction immediately following the "`BSF PMCON1,RD`" instruction to be ignored. The data is available in the very next cycle, in the PMDATH:PMDATL register pair; therefore, it can be read as two bytes in the following instructions.

The PMDATH:PMDATL register pair will hold this value until another read or until it is written to by the user.

Note:	The two instructions following a program memory read are required to be `NOP`s. This prevents the user from executing a 2-cycle instruction on the next instruction after the RD bit is set.

FIGURE 10-1: FLASH PROGRAM MEMORY READ FLOWCHART



12.8 Register Definitions: PPS Input Selection

REGISTER 12-1: xxxPPS: PERIPHERAL xxx INPUT SELECTION

U-0	U-0	R/W-q/u	R/W-q/u	R/W-q/u	R/W-q/u	R/W-q/u	R/W-q/u
—	—	xxxPPS<5:0>					
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 7-6 **Unimplemented:** Read as '0'

bit 5-3 **xxxPPS<5:3>:** Peripheral xxx Input PORT Selection bits
100 = Peripheral input is PORTE
011 = Peripheral input is PORTD(1)
010 = Peripheral input is PORTC
001 = Peripheral input is PORTB
000 = 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)F1777/9 only.

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

U-0	U-0	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u
—	—	RxyPPS<5:0>					
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 7-6 **Unimplemented:** Read as '0'

bit 5-0 **RxyPPS<5:0>:** Pin Rxy Output Source Selection bits
Selection code determines the output signal on the port pin.
See Table 12-2 for the selection codes

TABLE 12-1: PPS INPUT REGISTER RESET VALUES (CONTINUED)

Peripheral	xxxPPS Register	Default Pin Selection	Reset Value (xxxPPS<5:0>)	Port Selection PIC16(L)F1777/9					Port Selection PIC16(L)F1778		
		PIC16(L)F1777/8/9	PIC16(L)F1777/8/9	A	B	C	D	E	A	B	C
SPI and I^2C clock	SSPCLKPPS	RC3	010011		•	•				•	•
SPI and I^2C data	SSPDATPPS	RC4	010100		•	•				•	•
SPI slave select	SSPSSPPS	RA5	000101	•			•		•		•
EUSART RX	RXPPS	RC7	010111		•	•				•	•
EUSART CK	CKPPS	RC6	010110		•	•				•	•
All CLCs	CLCIN0PPS	RA0	000000	•		•			•		•
All CLCs	CLCIN1PPS	RA1	000001	•		•			•		•
All CLCs	CLCIN2PPS	RB6	001110		•		•			•	•
All CLCs	CLCIN3PPS	RB7	001111		•		•			•	•

Example: CCP1PPS = 0x13 selects RC3 as the CCP1 input.
Note 1: PIC16(L)F1777/9 only

17.6 Register Definitions: DAC Control

Long bit name prefixes for the 5-bit DAC peripherals are shown in Table 17-2. Refer to **Section 1.1 "Register and Bit naming conventions"** for more information

TABLE 17-2:

Peripheral	Bit Name Prefix
DAC3	DAC3
DAC4	DAC4
DAC7	DAC7
DAC8(1)	DAC8

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

REGISTER 17-1: DACxCON0: DACx CONTROL REGISTER 0

R/W-0/0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	R/W-0/0
EN	—	OE1	OE2	PSS<1:0>		NSS<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 **EN:** DAC Enable bit
1 = DAC is enabled
0 = DAC is disabled

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

bit 5 **OE1:** DAC Voltage Output Enable bit
1 = DAC voltage level is also an output on the DACxOUT1 pin
0 = DAC voltage level is disconnected from the DACxOUT1 pin

bit 4 **OE2:** DAC Voltage Output Enable bit
1 = DAC voltage level is also an output on the DACxOUT2 pin
0 = DAC voltage level is disconnected from the DACxOUT2 pin

bit 3-2 **PSS<1:0>:** DAC Positive Source Select bits
11 = Reserved, do not use
10 = FVR Buffer2 output
01 = V_{REF+} pin
00 = V_{DD}

bit 1-0 **NSS<1:0>:** DAC Negative Source Select bits
11 = Reserved, do not use
10 = DACxREF1- (DAC7/8) or Reserved (DAC3/4)
01 = DACxREF0-
00 = AG_{ND} (AV_{SS})

EQUATION 20-2: R-C CALCULATIONS

V_{peak} = external voltage source peak voltage
f = external voltage source frequency
C = series capacitor
R = series resistor
V_C = Peak capacitor voltage
ϕ = Capacitor induced zero crossing phase advance in radians
T_ϕ = Time ZC event occurs before actual zero crossing

$$Z = \frac{V_{PEAK}}{3x10^{-4}}$$

$$X_C = \frac{1}{(2\pi fC)}$$

$$R = \sqrt{Z^2 - X_C}$$

$$V_C = X_C(3x10^{-4})$$

$$\phi = Tan^{-1}\left(\frac{X_C}{R}\right)$$

$$T_\phi = \frac{\phi}{(2\pi f)}$$

V_{rms} = 120

EQUATION 20-3: R-C CALCULATIONS EXAMPLE

$$V_{peak} = V_{rms} \cdot \sqrt{2} = 169.7$$

f = 60 Hz
C = 0.1 μf

$$Z = \frac{V_{peak}}{3 \times 10^{-4}} = \frac{169.7}{3 \times 10^{-4}} = 565.7\ kOhms$$

$$X_C = \frac{1}{(2\pi fC)} = \frac{1}{(2\pi \cdot 60 \cdot 1 \cdot 10^{-7})} = 26.53\ \text{kOhms}$$

R = 560 kOhms

$$Z_R = \sqrt{(R^2 + X_C^2)} = 560.6\ kOhm\ \text{(using actual resistor)}$$

$$I_{peak} = \frac{V_{peak}}{Z_R} = 302.7 \cdot 10^{-6}$$

$$V_C = X_C \cdot I_{peak} = 8.0\ \text{V}$$

$$\phi = Tan^{-1}\left(\frac{X_C}{R}\right) = 0.047\ radians$$

$$T_\phi = \frac{\phi}{(2\pi f)} = 125.6\ \mu s$$

FIGURE 22-6: TIMER1 GATE SINGLE-PULSE AND TOGGLE COMBINED MODE

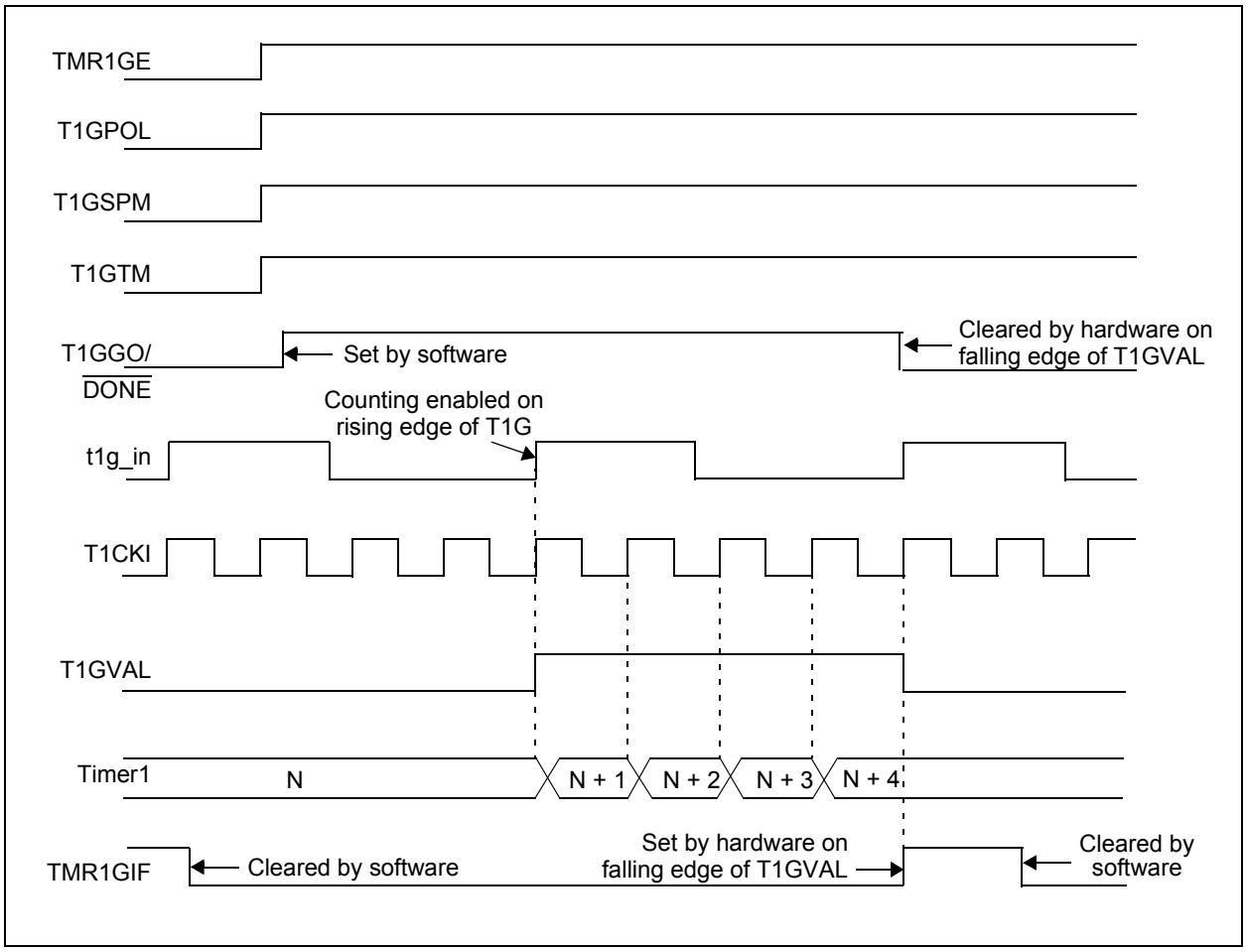
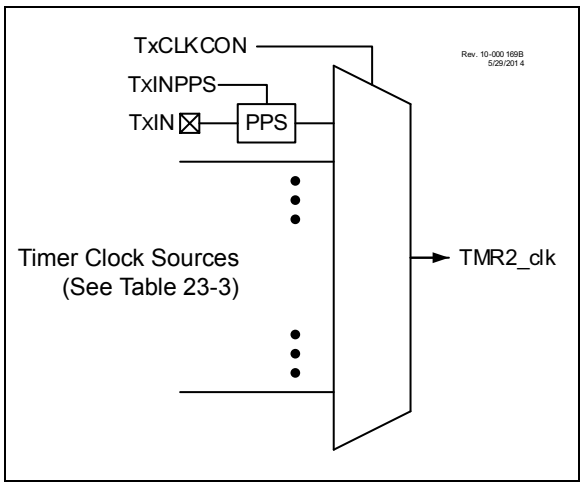


FIGURE 23-2: TIMER2 CLOCK SOURCE BLOCK DIAGRAM



23.1 Timer2 Operation

Timer2 operates in three major modes:

- Free Running Period
- One-shot
- Monostable

Within each mode there are several options for starting, stopping, and reset. Table 23-1 lists the options.

In all modes, the TMR2 count register is incremented on the rising edge of the clock signal from the programmable prescaler. When TMR2 equals T2PR, a high level is output to the postscaler counter. TMR2 is cleared on the next clock input.

An external signal from hardware can also be configured to gate the timer operation or force a TMR2 count Reset. In Gate modes the counter stops when the gate is disabled and resumes when the gate is enabled. In Reset modes the TMR2 count is reset on either the level or edge from the external source.

The TMR2 and T2PR registers are both directly readable and writable. The TMR2 register is cleared and the T2PR register initializes to FFh on any device Reset. Both the prescaler and postscaler counters are cleared on the following events:

- a write to the TMR2 register
- a write to the T2CON register
- any device Reset
- External Reset Source event that resets the timer.

Note:	TMR2 is not cleared when T2CON is written.

23.1.1 FREE RUNNING PERIOD MODE

The value of TMR2 is compared to that of the Period register, T2PR, on each clock cycle. When the two values match, the comparator resets the value of TMR2 to 00h on the next cycle and increments the output postscaler counter. When the postscaler count equals the value in the OUTPS<4:0> bits of the TMRxCON1 register then a one clock period wide pulse occurs on the TMR2_postscaled output, and the postscaler count is cleared.

23.1.2 ONE-SHOT MODE

The One-Shot mode is identical to the Free Running Period mode except that the ON bit is cleared and the timer is stopped when TMR2 matches T2PR and will not restart until the T2ON bit is cycled off and on. Postscaler OUTPS<4:0> values other than 0 are meaningless in this mode because the timer is stopped at the first period event and the postscaler is reset when the timer is restarted.

23.1.3 MONOSTABLE MODE

Monostable modes are similar to One-Shot modes except that the ON bit is not cleared and the timer can be restarted by an external Reset event.

23.2 PRx Period Register

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

- A write to the TMRx register
- A write to the TMRxCON register
- When TMRx = PRx buffer and the prescaler rolls over
- An external Reset event

23.3 Timer2 Output

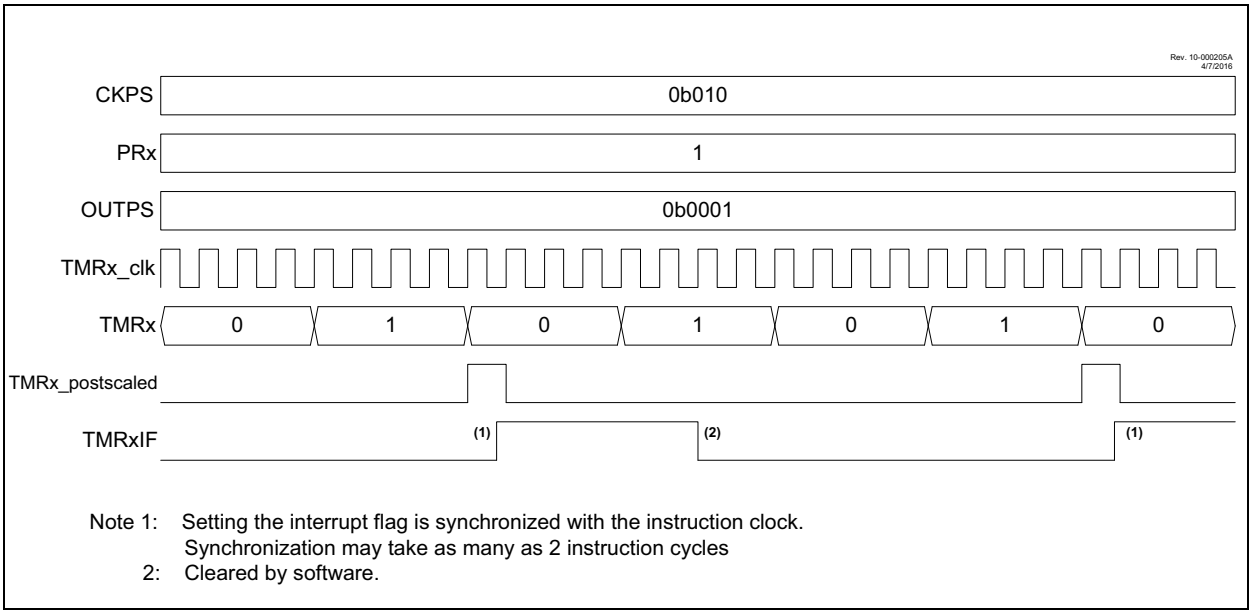
The Timer2 module's primary output is TMR2_postscaled, which pulses for a single TMR2_clk period when the postscaler counter matches the value in the OUTPS bits of the TMR2xCON register. The T2PR postscaler is incremented each time the TMR2 value matches the T2PR value. This signal can be selected as an input to several other input modules:

- The ADC module, as an Auto-conversion Trigger
- COG, as an auto-shutdown source

23.5 Timer2 Interrupt

Timer2 can also generate a device interrupt. The interrupt is generated when the postscaler counter matches one of 16 postscale options (from 1:1 through 1:16), which are selected with the postscaler control bits, OUTPS<3:0> of the T2CON register. The interrupt is enabled by setting the TMR2IE interrupt enable bit of the PIE1 register. Interrupt timing is illustrated in Figure 23-3.

FIGURE 23-3: TIMER2 PRESCALER, POSTSCALER, AND INTERRUPT TIMING DIAGRAM



Note 1: Setting the interrupt flag is synchronized with the instruction clock. Synchronization may take as many as 2 instruction cycles
2: Cleared by software.

24.4.4 PWM RESOLUTION

The resolution determines the number of available duty cycles for a given period. For example, a 10-bit resolution will result in 1024 discrete duty cycles, whereas an 8-bit resolution will result in 256 discrete duty cycles.

The maximum PWM resolution is ten bits when T2PR/T4PR/T6PR/T8PR is 255. The resolution is a function of the T2PR/T4PR/T6PR/T8PR register value as shown by Equation 24-4.

EQUATION 24-4: PWM RESOLUTION

$$Resolution = \frac{log[4(PR2 + 1)]}{log(2)} \ bits$$

Note:	If the pulse-width value is greater than the period, the assigned PWM pin(s) will remain unchanged.

TABLE 24-2: EXAMPLE PWM FREQUENCIES AND RESOLUTIONS (FOSC = 20 MHz)

PWM Frequency	1.22 kHz	4.88 kHz	19.53 kHz	78.12 kHz	156.3 kHz	208.3 kHz
Timer Prescale	16	4	1	1	1	1
T2PR Value	0xFF	0xFF	0xFF	0x3F	0x1F	0x17
Maximum Resolution (bits)	10	10	10	8	7	6

TABLE 24-3: EXAMPLE PWM FREQUENCIES AND RESOLUTIONS (FOSC = 8 MHz)

PWM Frequency	1.22 kHz	4.90 kHz	19.61 kHz	76.92 kHz	153.85 kHz	200.0 kHz
Timer Prescale	16	4	1	1	1	1
T2PR Value	0x65	0x65	0x65	0x19	0x0C	0x09
Maximum Resolution (bits)	8	8	8	6	5	5

24.4.5 CHANGES IN SYSTEM CLOCK FREQUENCY

The PWM frequency is derived from the system clock frequency. Any changes in the system clock frequency will result in changes to the PWM frequency. See **Section 5.0 "Oscillator Module (with Fail-Safe Clock Monitor)"** for additional details.

24.4.6 EFFECTS OF RESET

Any Reset will force all ports to Input mode and the CCP registers to their Reset states.

24.4.7 PWM OUTPUT

The output of the CCP in PWM mode is the PWM signal generated by the module and described above. This output is available to the following peripherals:

- ADC Trigger
- COG
- PRG
- DSM
- CLC
- Op Amp override
- Timer2/4/6/8 Reset
- Any device pins

REGISTER 26-6: PWMxOFCON: PWM OFFSET TRIGGER SOURCE SELECT REGISTER

U-0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	U-0	R/W-0/0	R/W-0/0
—	OFM<1:0>		OFO(1)	—	—	OFS<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 **Unimplemented:** Read as '0'

bit 6-5 **OFM<1:0>:** Offset Mode Select bits
11 = Continuous Run Slave mode with offset triggered timer Reset and synchronized start
10 = One-shot Slave mode with offset triggered synchronized start
01 = Slave Run mode with offset triggered synchronized start
00 = Independent Run mode

bit 4 **OFO:** Offset Match Output Control bit(1)
If MODE<1:0> = 11 (PWM center aligned mode):
1 = OFx_match occurs when the PWMxTMR is counting up
0 = OFx_match occurs when the PWMxTMR is counting down
If MODE<1:0> = 00, 01, or 10 (all other modes):
this bit is ignored

bit 3-2 **Unimplemented:** Read as '0'

bit 1-0 **OFS<1:0>:** Offset Trigger Source Select bit
10 = OF11_match
01 = OF6_match
00 = OF5_match

Note 1: The source corresponding to the PWM module's own OFx_match is reserved.

REGISTER 28-10: CLCxGLS3: GATE 4 LOGIC SELECT 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
G4D4T	G4D4N	G4D3T	G4D3N	G4D2T	G4D2N	G4D1T	G4D1N
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 **G4D4T:** Gate 4 Data 4 True (non-inverted) bit
1 = d4T is gated into g4
0 = d4T is not gated into g4

bit 6 **G4D4N:** Gate 4 Data 4 Negated (inverted) bit
1 = d4N is gated into g4
0 = d4N is not gated into g4

bit 5 **G4D3T:** Gate 4 Data 3 True (non-inverted) bit
1 = d3T is gated into g4
0 = d3T is not gated into g4

bit 4 **G4D3N:** Gate 4 Data 3 Negated (inverted) bit
1 = d3N is gated into g4
0 = d3N is not gated into g4

bit 3 **G4D2T:** Gate 4 Data 2 True (non-inverted) bit
1 = d2T is gated into g4
0 = d2T is not gated into g4

bit 2 **G4D2N:** Gate 4 Data 2 Negated (inverted) bit
1 = d2N is gated into g4
0 = d2N is not gated into g4

bit 1 **G4D1T:** Gate 4 Data 1 True (non-inverted) bit
1 = d1T is gated into g4
0 = d1T is not gated into g4

bit 0 **G4D1N:** Gate 4 Data 1 Negated (inverted) bit
1 = d1N is gated into g4
0 = d1N is not gated into g4

32.2.5 SLAVE SELECT SYNCHRONIZATION

The Slave Select can also be used to synchronize communication. The Slave Select line is held high until the master device is ready to communicate. When the Slave Select line is pulled low, the slave knows that a new transmission is starting.

If the slave fails to receive the communication properly, it will be reset at the end of the transmission, when the Slave Select line returns to a high state. The slave is then ready to receive a new transmission when the Slave Select line is pulled low again. If the Slave Select line is not used, there is a risk that the slave will eventually become out of sync with the master. If the slave misses a bit, it will always be one bit off in future transmissions. Use of the Slave Select line allows the slave and master to align themselves at the beginning of each transmission.

The $\overline{SS}$ pin allows a Synchronous Slave mode. The SPI must be in Slave mode with $\overline{SS}$ pin control enabled (SSPxCON1<3:0> = 0100).

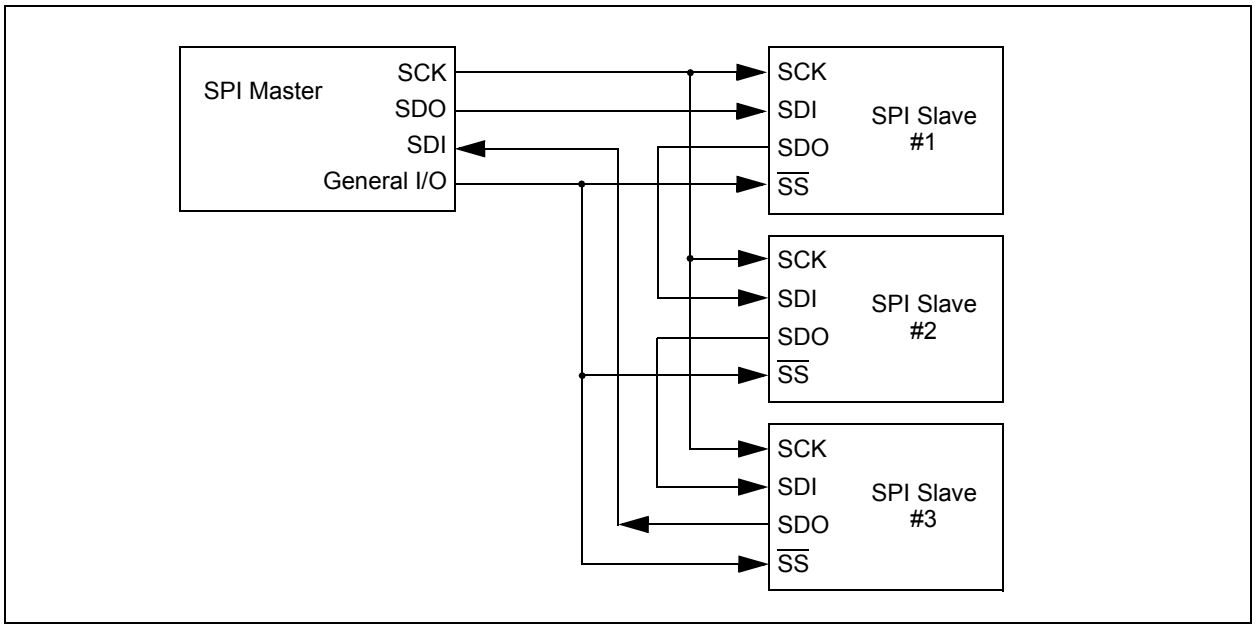
When the $\overline{SS}$ pin is low, transmission and reception are enabled and the SDO pin is driven.

When the $\overline{SS}$ pin goes high, the SDO pin is no longer driven, even if in the middle of a transmitted byte and becomes a floating output. External pull-up/pull-down resistors may be desirable depending on the application.

Note 1:	When the SPI is in Slave mode with $\overline{SS}$ pin control enabled (SSPxCON1<3:0> = 0100), the SPI module will reset if the $\overline{SS}$ pin is set to V_{DD}.
2:	When the SPI is used in Slave mode with CKE set; the user must enable $\overline{SS}$ pin control.
3:	While operated in SPI Slave mode the SMP bit of the SSPxSTAT register must remain clear.

When the SPI module resets, the bit counter is forced to '0'. This can be done by either forcing the $\overline{SS}$ pin to a high level or clearing the SSPEN bit.

FIGURE 32-7: SPI DAISY-CHAIN CONNECTION



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

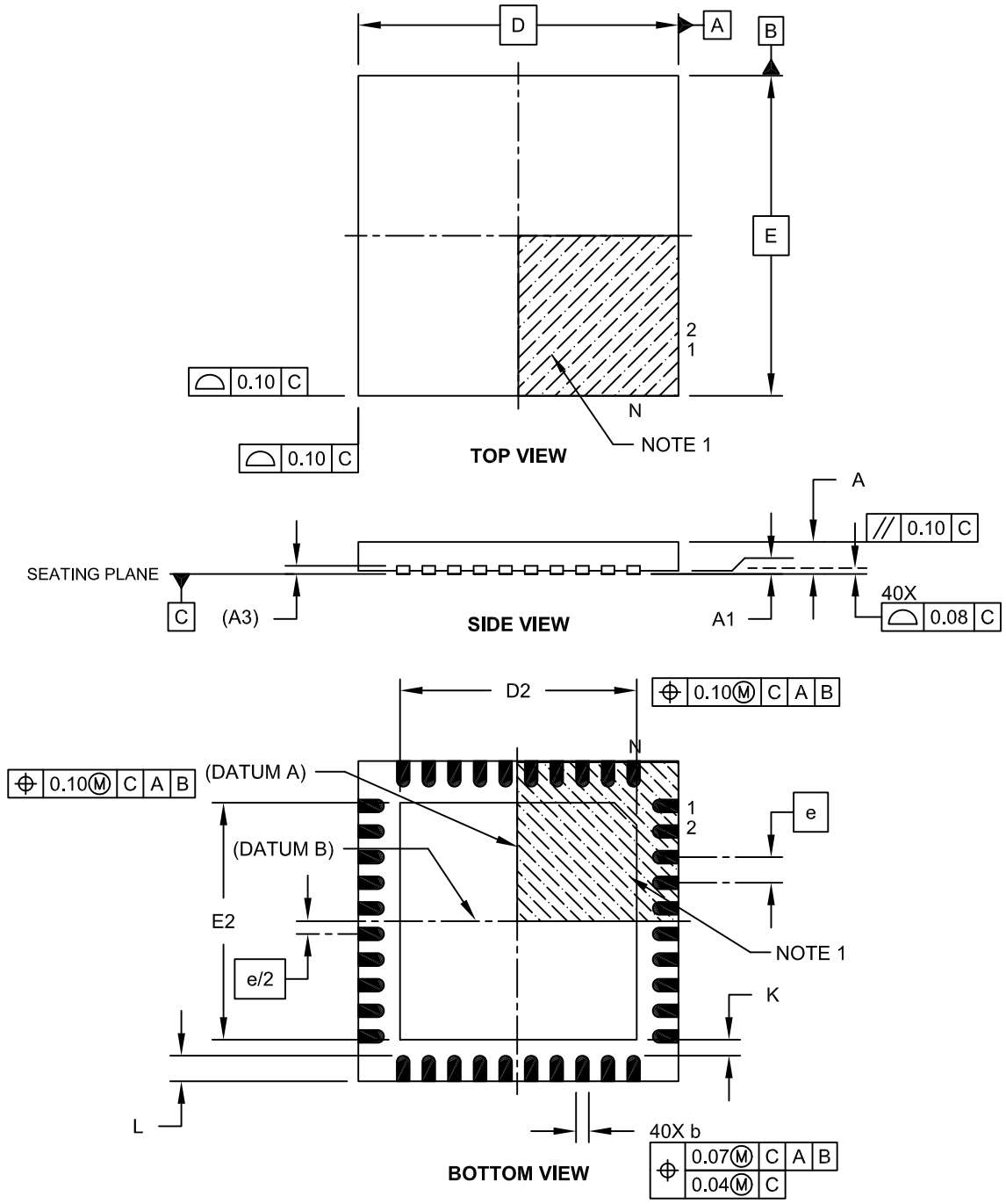
The hold time of the SDA pin is selected by the SDAHT bit of the SSPxCON3 register. Hold time is the time SDA is held valid after the falling edge of SCL. Setting the SDAHT bit selects a longer 300 ns minimum hold time and may help on buses with large capacitance.

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.

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

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
═ ISO/TS 16949 ═

Trademarks

ISBN: 978-1-5224-1016-4