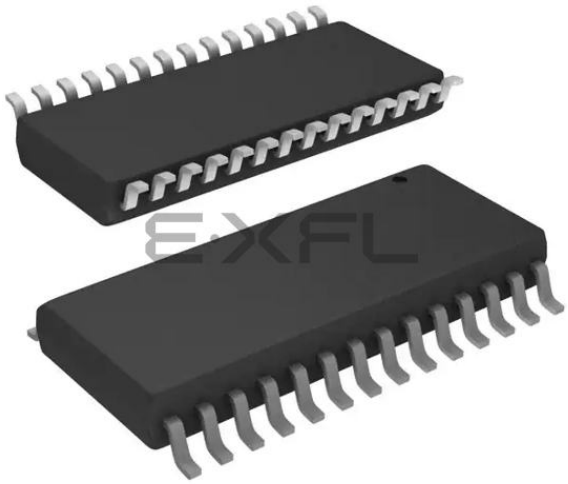


Welcome to E-XFL.COM

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	25
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 17x10b; D/A 3x5b, 3x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SOIC (0.295", 7.50mm Width)
Supplier Device Package	28-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf1778-e-so

3.2.1 READING PROGRAM MEMORY AS DATA

There are two methods of accessing constants in program memory. The first method is to use tables of RETLW instructions. The second method is to set an FSR to point to the program memory.

3.2.1.1 RETLW Instruction

The RETLW instruction can be used to provide access to tables of constants. The recommended way to create such a table is shown in Example 3-1.

EXAMPLE 3-1: RETLW INSTRUCTION

```
constants
    BRW                 ;Add Index in W to
                        ;program counter to
                        ;select data
    RETLW DATA0         ;Index0 data
    RETLW DATA1         ;Index1 data
    RETLW DATA2
    RETLW DATA3


my_function
    ;… LOTS OF CODE…
    MOVLW       DATA_INDEX
    call constants
    ;… THE CONSTANT IS IN W
```

The BRW instruction makes this type of table very simple to implement. If your code must remain portable with previous generations of microcontrollers, then the BRW instruction is not available so the older table read method must be used.

3.2.1.2 Indirect Read with FSR

The program memory can be accessed as data by setting bit 7 of the FSRxH register and reading the matching INDFx register. The MOVIW instruction will place the lower eight bits of the addressed word in the W register. Writes to the program memory cannot be performed via the INDF registers. Instructions that access the program memory via the FSR require one extra instruction cycle to complete. Example 3-2 demonstrates accessing the program memory via an FSR.

The high directive will set bit<7> if a label points to a location in program memory.

EXAMPLE 3-2: ACCESSING PROGRAM MEMORY VIA FSR

```
constants
  DW DATA0              ;First constant
  DW DATA1              ;Second constant
  DW DATA2
  DW DATA3
my_function
  ;… LOTS OF CODE…
  MOVLW  DATA_INDEX
  ADDLW  LOW constants
  MOVWF  FSR1L
  MOVLW  HIGH constants;MSb sets
                        automatically
  MOVWF  FSR1H
  BTFSC  STATUS, C      ;carry from ADDLW?
  INCF   FSR1H, f       ;yes
  MOVIW  0[FSR1]
;THE PROGRAM MEMORY IS IN W
```

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 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 14											
70Ch	—	Unimplemented								—	—
70Dh	COG2PHR	—	—	COG Rising Edge Phase Delay Count Register						--00 0000	--00 0000
70Eh	COG2PHF	—	—	COG Falling Edge Phase Delay Count Register						--00 0000	--00 0000
70Fh	COG2BLKR	—	—	COG Rising Edge Blanking Count Register						--00 0000	--00 0000
710h	COG2BLKF	—	—	COG Falling Edge Blanking Count Register						--00 0000	--00 0000
711h	COG2DBR	—	—	COG Rising Edge Dead-band Count Register						--00 0000	--00 0000
712h	COG2DBF	—	—	COG Falling Edge Dead-band Count Register						--00 0000	--00 0000
713h	COG2CON0	EN	LD	—	CS<1:0>		MD<2:0>			00-0 0000	00-0 0000
714h	COG2CON1	RDBS	FDBS	—	—	POLD	POLC	POLB	POLA	00-- 0000	00-- 0000
715h	COG2RIS0	RIS7	RIS6	RIS5	RIS4	RIS3	RIS2	RIS1	RIS0	0000 0000	0000 0000
716h	COG2RIS1	RIS15	RIS14	RIS13	RIS12	RIS11	RIS10	RIS9	RIS8	0000 0000	0000 0000
717h	COG2RSIM0	RSIM7	RSIM6	RSIM5	RSIM4	RSIM3	RSIM2	RSIM1	RSIM0	0000 0000	0000 0000
718h	COG2RSIM1	RSIM15	RSIM14	RSIM13	RSIM12	RSIM11	RSIM10	RSIM9	RSIM8	0000 0000	0000 0000
719h	COG2FIS0	FIS7	FIS6	FIS5	FIS4	FIS3	FIS2	FIS1	FIS0	0000 0000	0000 0000
71Ah	COG2FIS1	FIS15	FIS14	FIS13	FIS12	FIS11	FIS10	FIS9	FIS8	0000 0000	0000 0000
71Bh	COG2FSIM0	FSIM7	FSIM6	FSIM5	FSIM4	FSIM3	FSIM2	FSIM1	FSIM0	0000 0000	0000 0000
71Ch	COG2FSIM1	FSIM15	FSIM14	FSIM13	FSIM12	FSIM11	FSIM10	FSIM9	FSIM8	0000 0000	0000 0000
71Dh	COG2ASD0	ASE	ARSEN	ASDBD<1:0>		ASDAC<1:0>		—	—	0001 01--	0001 01--
71Eh	COG2ASD1	AS7E	AS6E	AS5E	AS4E	AS3E	AS2E	AS1E	AS0E	0000 0000	0000 0000
71Fh	COG2STR	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.

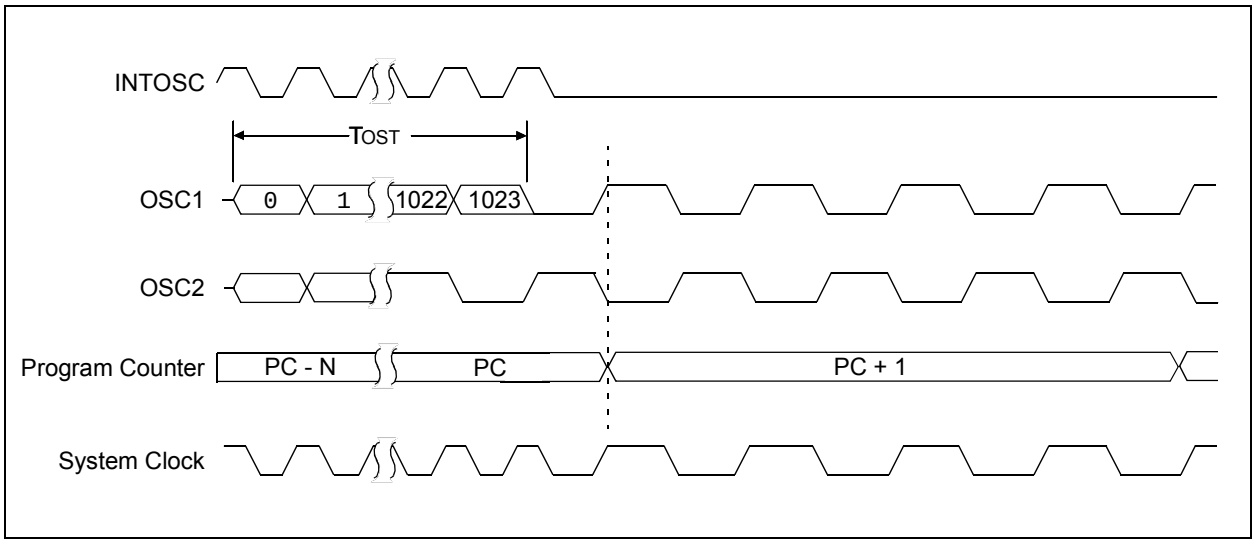
5.4.2 TWO-SPEED START-UP SEQUENCE

1. Wake-up from Power-on Reset or Sleep.
2. Instructions begin execution by the internal oscillator at the frequency set in the IRCF<3:0> bits of the OSCCON register.
3. OST enabled to count 1024 clock cycles.
4. OST timed out, wait for falling edge of the internal oscillator.
5. OSTS is set.
6. System clock held low until the next falling edge of new clock (LP, XT or HS mode).
7. System clock is switched to external clock source.

5.4.3 CHECKING TWO-SPEED CLOCK STATUS

Checking the state of the OSTS bit of the OSCSTAT register will confirm if the microcontroller is running from the external clock source, as defined by the FOSC<2:0> bits in the Configuration Words, or the internal oscillator.

FIGURE 5-8: TWO-SPEED START-UP



10.2.4 WRITING TO FLASH PROGRAM MEMORY

Program memory is programmed using the following steps:

1. Load the address in PMADRH:PMADRL of the row to be programmed.
2. Load each write latch with data.
3. Initiate a programming operation.
4. Repeat steps 1 through 3 until all data is written.

Before writing to program memory, the word(s) to be written must be erased or previously unwritten. Program memory can only be erased one row at a time. No automatic erase occurs upon the initiation of the write.

Program memory can be written one or more words at a time. The maximum number of words written at one time is equal to the number of write latches. See Figure 10-5 (row writes to program memory with 32 write latches) for more details.

The write latches are aligned to the Flash row address boundary defined by the upper ten bits of PMADRH:PMADRL, (PMADRH<6:0>:PMADRL<7:5>) with the lower five bits of PMADRL, (PMADRL<4:0>) determining the write latch being loaded. Write operations do not cross these boundaries. At the completion of a program memory write operation, the data in the write latches is reset to contain 0x3FFF.

The following steps should be completed to load the write latches and program a row of program memory. These steps are divided into two parts. First, each write latch is loaded with data from the PMDATH:PMDATL using the unlock sequence with LWLO = 1. When the last word to be loaded into the write latch is ready, the LWLO bit is cleared and the unlock sequence executed. This initiates the programming operation, writing all the latches into Flash program memory.

Note:	The special unlock sequence is required to load a write latch with data or initiate a Flash programming operation. If the unlock sequence is interrupted, writing to the latches or program memory will not be initiated.

1. Set the WREN bit of the PMCON1 register.
2. Clear the CFGS bit of the PMCON1 register.
3. Set the LWLO bit of the PMCON1 register. When the LWLO bit of the PMCON1 register is '1', the write sequence will only load the write latches and will not initiate the write to Flash program memory.
4. Load the PMADRH:PMADRL register pair with the address of the location to be written.
5. Load the PMDATH:PMDATL register pair with the program memory data to be written.
6. Execute the unlock sequence (**Section 10.2.2 "Flash Memory Unlock Sequence"**). The write latch is now loaded.
7. Increment the PMADRH:PMADRL register pair to point to the next location.
8. Repeat steps 5 through 7 until all but the last write latch has been loaded.
9. Clear the LWLO bit of the PMCON1 register. When the LWLO bit of the PMCON1 register is '0', the write sequence will initiate the write to Flash program memory.
10. Load the PMDATH:PMDATL register pair with the program memory data to be written.
11. Execute the unlock sequence (**Section 10.2.2 "Flash Memory Unlock Sequence"**). The entire program memory latch content is now written to Flash program memory.

Note:	The program memory write latches are reset to the blank state (0x3FFF) at the completion of every write or erase operation. As a result, it is not necessary to load all the program memory write latches. Unloaded latches will remain in the blank state.

An example of the complete write sequence is shown in Example 10-3. The initial address is loaded into the PMADRH:PMADRL register pair; the data is loaded using indirect addressing.

FIGURE 10-5: BLOCK WRITES TO FLASH PROGRAM MEMORY WITH 32 WRITE LATCHES

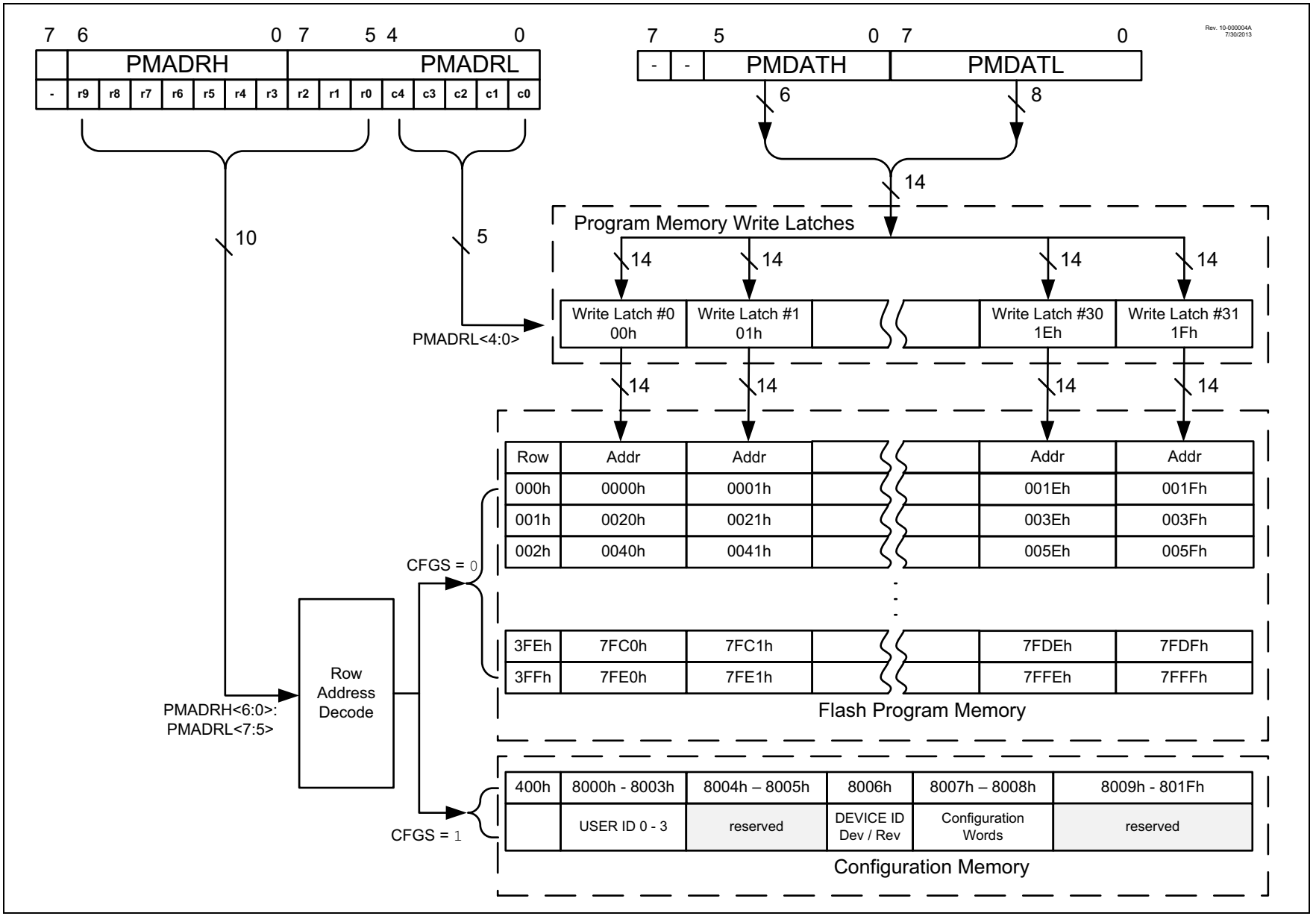


TABLE 12-2: AVAILABLE PORTS FOR OUTPUT BY PERIPHERAL(2)

RxyPPS<5:0>	Output Signal	PIC16(L)F1778			PIC16(L)F1777/9				
		A	B	C	A	B	C	D	E
110001	MD4_out(3)				—	●	—	●	
110000	MD3_out		●	●	—	●	—	●	—
101111	MD2_out	●		●	●	—	—	●	—
101110	MD1_out	●		●	●	—	—	●	—
101101	sync_C8OUT(3)				—	●	—	●	—
101100	sync_C7OUT(3)				—	●	—	●	—
101011	sync_C6OUT	●		●	●	—	—	—	●
101010	sync_C5OUT	●		●	●	—	—	●	—
101001	sync_C4OUT		●	●	—	●	—	●	—
101000	sync_C3OUT		●	●	—	●	—	●	—
100111	sync_C2OUT	●		●	●	—	—		●
100110	sync_C1OUT	●		●	●	—	—	●	—
100101	DT		●	●	—	●	●	—	—
100100	TX/CK		●	●	—	●	●	—	—
100011	SDO		●	●	—	●	●	—	—
100010	SDA		●	●	—	●	●	—	—
100001	SCK/SCL(1)		●	●	—	●	●	—	—
100000	PWM12_out(3)				—	●	—	●	—
011111	PWM11_out		●	●	—	●	—	●	—
011110	PWM6_out	●		●	●	—	—	●	—
011101	PWM5_out	●		●	●	—		●	—
011100	PWM10_out(3)				●	—	●	—	—
011011	PWM9_out	●		●	●	—	●		—
011010	PWM4_out	●		●	●	—		●	—
011001	PWM3_out	●		●	●	—		—	●
011000	CCP8_out(3)				—	●	—	●	—
010111	CCP7_out		●	●	—	●	—	●	—
010110	CCP2_out		●	●	—	●	●	—	—
010101	CCP1_out		●	●	—	●	●	—	—
010100	COG4D(1,3)				●	—		●	—
010011	COG4C(1,3)				●	—		●	—
010010	COG4B(1,3)				●	—		—	●
010001	COG4A(1,3)					●	●	—	—
010000	COG3D(1)	●		●	●	—	—	●	—
001111	COG3C(1)	●		●	●	—	—	●	—
001110	COG3B(1)	●		●	●	—	—	—	●
001101	COG3A(1)	●		●	—	●	●	—	—
001100	COG2D(1)		●	●	—	●	—	●	—
001011	COG2C(1)		●	●	—	●	—	●	—
001010	COG2B(1)		●	●		●		●	

Note 1: TRIS control is overridden by the peripheral as required.
2: Unsupported peripherals will output a '0'.
3: PIC16(L)F1777/9 only.

TABLE 12-3: SUMMARY OF REGISTERS ASSOCIATED WITH THE PPS MODULE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
PPSLOCK	—	—	—	—	—	—	—	PPSLOCKED	206
INTPPS	—	—	INTPPS<5:0>						205
T0CKIPPS	—	—	T0CKIPPS<5:0>						205
T1CKIPPS	—	—	T1CKIPPS<5:0>						205
T1GPPS	—	—	T1GPPS<5:0>						205
T3CKIPPS	—	—	T3CKIPPS<5:0>						205
T3GPPS	—	—	T3GPPS<5:0>						205
T5CKIPPS	—	—	T5CKIPPS<5:0>						205
T5GPPS	—	—	T5GPPS<5:0>						205
T2INPPS	—	—	T2INPPS<5:0>						205
T4INPPS	—	—	T4INPPS<5:0>						205
T6INPPS	—	—	T6INPPS<5:0>						205
T8INPPS	—	—	T8INPPS<5:0>						205
CCP1PPS	—	—	CCP1PPS<5:0>						205
CCP2PPS	—	—	CCP2PPS<5:0>						205
CCP7PPS	—	—	CCP7PPS<5:0>						205
CCP8PPS(1)	—	—	CCP8PPS<5:0>						205
COGIN1PPS	—	—	COG1PPS<5:0>						205
COG2INPPS	—	—	COG2PPS<5:0>						205
COG3INPPS	—	—	COG3PPS<5:0>						205
COG4INPPS(1)	—	—	COG4PPS<5:0>						205
MD1CLPPS	—	—	MD1CLPPS<5:0>						205
MD1CHPPS	—	—	MD1CHPPS<5:0>						205
MD1MODPPS	—	—	MD1MODPPS<5:0>						205
MD2CLPPS	—	—	MD2CLPPS<5:0>						205
MD2CHPPS	—	—	MD2CHPPS<5:0>						205
MD2MODPPS	—	—	MD2MODPPS<5:0>						205
MD3CLPPS	—	—	MD3CLPPS<5:0>						205
MD3CHPPS	—	—	MD3CHPPS<5:0>						205
MD3MODPPS	—	—	MD3MODPPS<5:0>						205
MD4CLPPS(1)	—	—	MD4CLPPS<5:0>						205
MD4CHPPS(1)	—	—	MD4CHPPS<5:0>						205
MD4MODPPS(1)	—	—	MD4MODPPS<5:0>						205
PRG1RPPS	—	—	PRG1RPPS<5:0>						205
PRG1FPPS	—	—	PRG1FPPS<5:0>						205
PRG2RPPS	—	—	PRG2RPPS<5:0>						205
PRG2FPPS	—	—	PRG2FPPS<5:0>						205
PRG3RPPS	—	—	PRG3RPPS<5:0>						205
PRG3FPPS	—	—	PRG3FPPS<5:0>						205
PRG4RPPS	—	—	PRG4RPPS<5:0>						205
PRG4FPPS	—	—	PRG4FPPS<5:0>						205
CLC1IN0PPS	—	—	CLCIN0PPS<5:0>						205

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.

TABLE 23-1: TIMER2 OPERATING MODES

Mode	MODE<4:0>		Output Operation	Operation	Timer Control		
	<4:3>	<2:0>			Start	Reset	Stop
Free Running Period	00	000	Period Pulse	Software gate (Figure 23-4)	ON = 1	—	ON = 0
		001		Hardware gate, active-high (Figure 23-5)	ON = 1 and TMRx_ers = 1	—	ON = 0 or TMRx_ers = 0
		010		Hardware gate, active-low	ON = 1 and TMRx_ers = 0	—	ON = 0 or TMRx_ers = 1
		011	Period Pulse with Hardware Reset	Rising or falling edge Reset	ON = 1	TMRx_ers ↕	ON = 0
		100		Rising edge Reset (Figure 23-6)		TMRx_ers ↑	
		101		Falling edge Reset		TMRx_ers ↓	
		110		Low level Reset		TMRx_ers = 0	ON = 0 or TMRx_ers = 0
		111		High level Reset (Figure 23-7)		TMRx_ers = 1	ON = 0 or TMRx_ers = 1
One-shot	01	000	One-shot	Software start (Figure 23-8)	ON = 1	—	ON = 0 or Next clock after TMRx = PRx (**Note 2**)
		001	Edge triggered start (**Note 1**)	Rising edge start (Figure 23-9)	ON = 1 and TMRx_ers ↑	—	
		010		Falling edge start	ON = 1 and TMRx_ers ↓	—	
		011		Any edge start	ON = 1 and TMRx_ers ↕	—	
		100	Edge triggered start and hardware Reset (**Note 1**)	Rising edge start and Rising edge Reset (Figure 23-10)	ON = 1 and TMRx_ers ↑	TMRx_ers ↑	
		101		Falling edge start and Falling edge Reset	ON = 1 and TMRx_ers ↓	TMRx_ers ↓	
		110		Rising edge start and Low level Reset (Figure 23-11)	ON = 1 and TMRx_ers ↑	TMRx_ers = 0	
		111		Falling edge start and High level Reset	ON = 1 and TMRx_ers ↓	TMRx_ers = 1	
Mono-stable	10	000	Reserved				
		001	Edge triggered start (**Note 1**)	Rising edge start (Figure 23-12)	ON = 1 and TMRx_ers ↑	—	ON = 0 or Next clock after TMRx = PRx (**Note 3**)
		010		Falling edge start	ON = 1 and TMRx_ers ↓	—	
		011		Any edge start	ON = 1 and TMRx_ers ↕	—	
Reserved		100	Reserved				
Reserved		101	Reserved				
One-shot		110	Level triggered start and hardware Reset	High level start and Low level Reset (Figure 23-13)	ON = 1 and TMRx_ers = 1	TMRx_ers = 0	ON = 0 or Held in Reset (**Note 2**)
		111		Low level start & High level Reset	ON = 1 and TMRx_ers = 0	TMRx_ers = 1	
Reserved	11	xxx	Reserved				

Note 1: If ON = 0 then an edge is required to restart the timer after ON = 1.

2: When TMRx = PRx then the next clock clears ON and stops TMRx at 00h.

3: When TMRx = PRx then the next clock stops TMRx at 00h but does not clear ON.

26.0 16-BIT PULSE-WIDTH MODULATION (PWM) MODULE

The Pulse-Width Modulation (PWM) module generates a pulse-width modulated signal determined by the phase, duty cycle, period, and offset event counts that are contained in the following registers:

- PWMxPH register
- PWMxDC register
- PWMxPR register
- PWMxOF register

Figure 26-1 shows a simplified block diagram of the PWM operation. Each PWM module has four modes of operation:

- Standard
- Set On Match
- Toggle On Match
- Center Aligned

For a more detailed description of each PWM mode, refer to **Section 26.2 "PWM Modes"**.

Each PWM module has four offset modes:

- Independent Run
- Slave Run with Synchronous Start
- One-Shot Slave with Synchronous Start
- Continuous Run Slave with Synchronous Start and Timer Reset

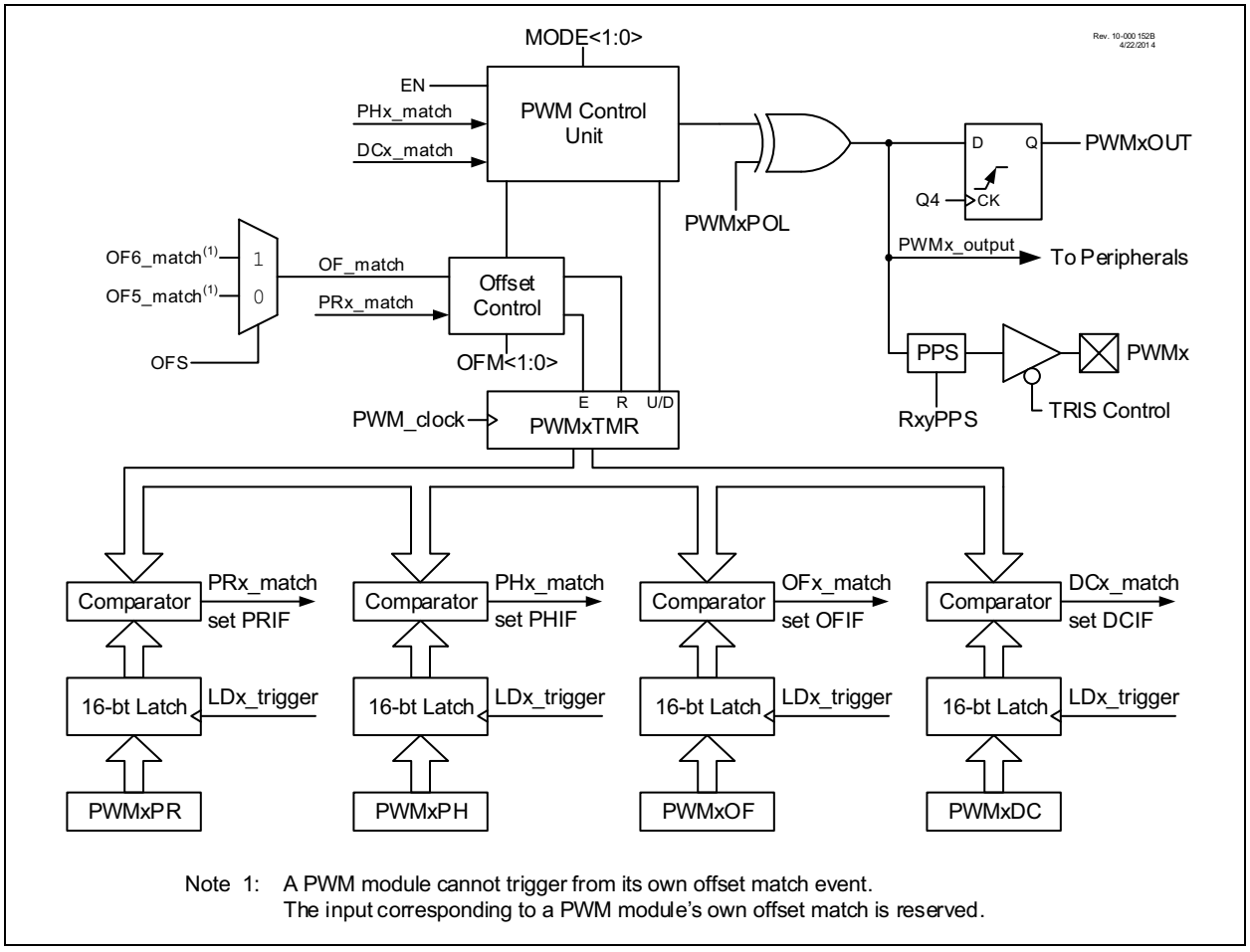
Using the offset modes, each PWM module can offset its waveform relative to any other PWM module in the same device. For a more detailed description of the offset modes refer to **Section 26.3 "Offset Modes"**.

Every PWM module has a configurable reload operation to ensure all event count buffers change at the end of a period, thereby avoiding signal glitches. Figure 26-2 shows a simplified block diagram of the reload operation. For a more detailed description of the reload operation, refer to Section **Section 26.4 "Reload Operation"**.

TABLE 26-1: AVAILABLE 16-BIT PWM MODULES

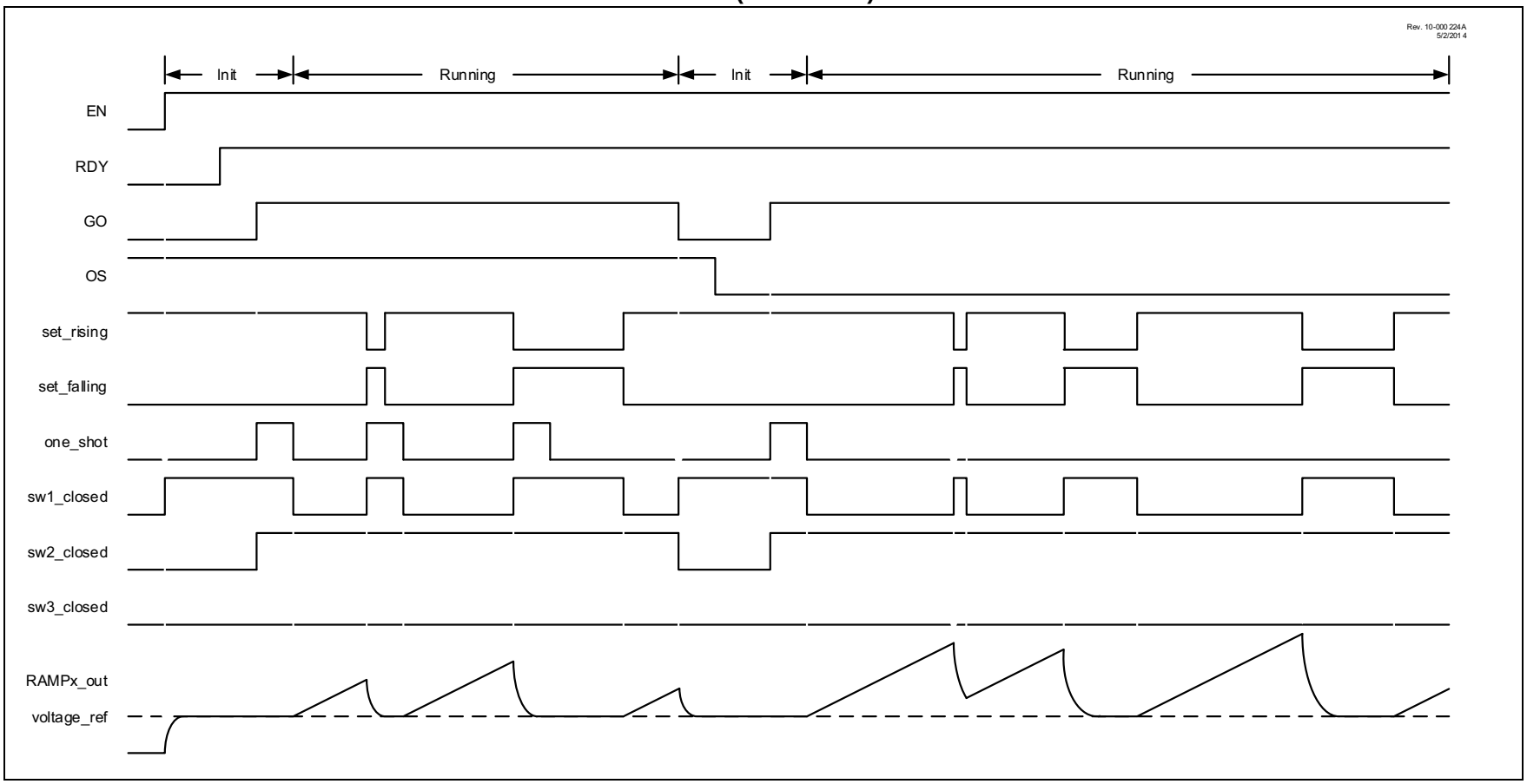
Device	PWM5	PWM6	PWM11	PWM12
PIC16(L)F1778	•	•	•	
PIC16(L)F1777/9	•	•	•	•

FIGURE 26-1: 16-BIT PWM BLOCK DIAGRAM



Note 1: A PWM module cannot trigger from its own offset match event. The input corresponding to a PWM module's own offset match is reserved.

FIGURE 30-5: RISING RAMP GENERATION TIMING DIAGRAM (MODE = 10)



33.1.1.5 TSR Status

The TRMT bit of the TXxSTA register indicates the status of the TSR register. This is a read-only bit. The TRMT bit is set when the TSR register is empty and is cleared when a character is transferred to the TSR register from the TXxREG. The TRMT bit remains clear until all bits have been shifted out of the TSR register. No interrupt logic is tied to this bit, so the user has to poll this bit to determine the TSR status.

Note:	The TSR register is not mapped in data memory, so it is not available to the user.

33.1.1.6 Transmitting 9-Bit Characters

The EUSART supports 9-bit character transmissions. When the TX9 bit of the TXxSTA register is set, the EUSART will shift nine bits out for each character transmitted. The TX9D bit of the TXxSTA register is the ninth, and Most Significant data bit. When transmitting 9-bit data, the TX9D data bit must be written before writing the eight Least Significant bits into the TXxREG. All nine bits of data will be transferred to the TSR shift register immediately after the TXxREG is written.

A special 9-bit Address mode is available for use with multiple receivers. See **Section 33.1.2.7 "Address Detection"** for more information on the Address mode.

33.1.1.7 Asynchronous Transmission Set-up:

1. Initialize the SPxBRGH:SPxBRGL register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see **Section 33.4 "EUSART Baud Rate Generator (BRG)"**).
2. Enable the asynchronous serial port by clearing the SYNC bit and setting the SPEN bit.
3. If 9-bit transmission is desired, set the TX9 control bit. A set ninth data bit will indicate that the eight Least Significant data bits are an address when the receiver is set for address detection.
4. Set the SCKP bit if inverted transmit is desired.
5. Enable the transmission by setting the TXEN control bit. This will cause the TXIF interrupt bit to be set.
6. If interrupts are desired, set the TXIE interrupt enable bit of the PIE1 register. An interrupt will occur immediately provided that the GIE and PEIE bits of the INTCON register are also set.
7. If 9-bit transmission is selected, the ninth bit should be loaded into the TX9D data bit.
8. Load 8-bit data into the TXxREG register. This will start the transmission.

FIGURE 33-3: ASYNCHRONOUS TRANSMISSION

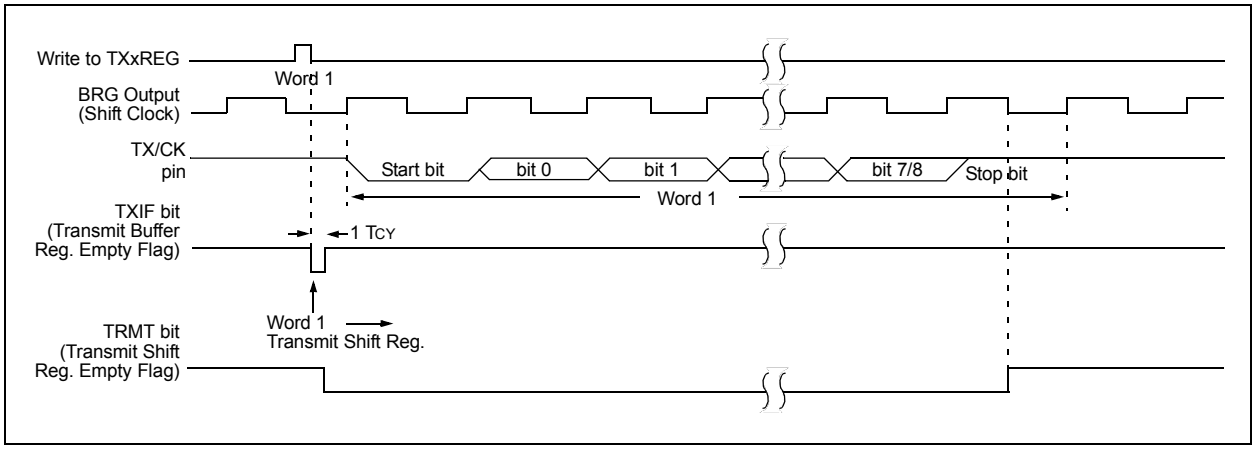


TABLE 33-3: BAUD RATE FORMULAS

Configuration Bits			BRG/EUSART Mode	Baud Rate Formula
SYNC	BRG16	BRGH		
0	0	0	8-bit/Asynchronous	FOSC/[64 (n+1)]
0	0	1	8-bit/Asynchronous	FOSC/[16 (n+1)]
0	1	0	16-bit/Asynchronous	
0	1	1	16-bit/Asynchronous	FOSC/[4 (n+1)]
1	0	x	8-bit/Synchronous	
1	1	x	16-bit/Synchronous	

Legend: x = Don't care, n = value of SPxBRGH:SPxBRGL register pair.

TABLE 33-4: SUMMARY OF REGISTERS ASSOCIATED WITH THE BAUD RATE GENERATOR

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
BAUD1CON	ABDOVF	RCIDL	—	SCKP	BRG16	—	WUE	ABDEN	505
RC1STA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	504
SP1BRGL	SP1BRG<7:0>								506
SP1BRGH	SP1BRG<15:8>								506
TX1STA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	503

Legend: — = unimplemented location, read as '0'. Shaded cells are not used for the Baud Rate Generator.

* Page provides register information.

TABLE 36-3: POWER-DOWN CURRENTS (I_{PD})(1,2)

PIC16LF1777/8/9		Operating Conditions: (unless otherwise stated) Low-Power Sleep Mode						
PIC16F1777/8/9		Low-Power Sleep Mode, VREGPM = 1						
Param No.	Device Characteristics	Min.	Typ†	Max. +85°C	Max. +125°C	Units	Conditions	
							V_{DD}	Note
D023	Base I_{PD}	—	0.05	1.0	8.0	µA	1.8	WDT, BOR, FVR, and SOSC disabled, all Peripherals Inactive
		—	0.08	2.0	9.0	µA	3.0	
D023	Base I_{PD}	—	0.3	2.4	10	µA	2.3	WDT, BOR, FVR, and SOSC disabled, all Peripherals Inactive, Low-Power Sleep mode
		—	0.4	4	12	µA	3.0	
		—	0.5	6	15	µA	5.0	
D023A	Base I_{PD}	—	9.8	17	28	µA	2.3	WDT, BOR, FVR and SOSC disabled, all Peripherals inactive, Normal Power Sleep mode VREGPM = 0
		—	10.3	20	40	µA	3.0	
		—	11.5	22	44	µA	5.0	
D024		—	0.5	6	14	µA	1.8	WDT Current
		—	0.8	7	17	µA	3.0	
D024		—	0.8	6	15	µA	2.3	WDT Current
		—	0.9	7	20	µA	3.0	
		—	1.0	8	22	µA	5.0	
D025		—	15	28	30	µA	1.8	FVR Current (ADC)
		—	24	35	38	µA	3.0	
D025		—	18	33	35	µA	2.3	FVR Current (ADC)
		—	24	35	40	µA	3.0	
		—	26	37	44	µA	5.0	
D025A		—	25	50	55	µA	1.8	FVR Current (DAC)
		—	30	65	70	µA	3.0	
D025A		—	30	55	66	µA	2.3	FVR Current (DAC)
		—	32	68	82	µA	3.0	
		—	35	77	90	µA	5.0	
D026		—	7.5	25	28	µA	3.0	BOR Current
D026		—	10	25	28	µA	3.0	BOR Current
		—	12	28	31	µA	5.0	
D027		—	0.5	4	10	µA	3.0	LPBOR Current
D027		—	0.8	6	15	µA	3.0	LPBOR Current
		—	1	8	17	µA	5.0	
D028		—	0.5	5	9	µA	1.8	SOSC Current
		—	0.8	8.5	12	µA	3.0	
D028		—	1.1	6	10	µA	2.3	SOSC Current
		—	1.3	8.5	20	µA	3.0	
		—	1.4	10	25	µA	5.0	

* These parameters are characterized but not tested.

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

Note 1: The peripheral current is the sum of the base I_{PD} and the additional current consumed when this peripheral is enabled. The peripheral Δ current can be determined by subtracting the base I_{DD} or I_{PD} current from this limit. Max values should be used when calculating total current consumption.

2: The power-down current in Sleep mode does not depend on the oscillator type. Power-down current is measured with the part in Sleep mode, with all I/O pins in high-impedance state and tied to V_{SS}.

3: ADC clock source is FRC.

TABLE 36-4: I/O PORTS (CONTINUED) (CONTINUED)

Standard Operating Conditions (unless otherwise stated)							
Param No.	**Sym.**	**Characteristic**	**Min.**	**Typ†**	**Max.**	**Units**	**Conditions**
		Capacitive Loading Specs on Output Pins					
D101*	COSC2	OSC2 pin	—	—	15	pF	In XT, HS and LP modes when external clock is used to drive OSC1
D101A*	CIO	All I/O pins	—	—	50	pF	

* These parameters are characterized but not tested.

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

Note 1: In EXTRC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended to use an external clock in EXTRC mode.

2: Negative current is defined as current sourced by the pin.

3: The leakage current on the $\overline{MCLR}$ pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.

4: Including OSC2 in CLKOUT mode.

TABLE 36-11: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER, POWER-UP TIMER AND BROWN-OUT RESET PARAMETERS

Standard Operating Conditions (unless otherwise stated)							
Param No.	**Sym.**	**Characteristic**	**Min.**	**Typ†**	**Max.**	**Units**	**Conditions**
30	TMCL	$\overline{MCLR}$ Pulse Width (low)	2	—	—	μs	
31	TWDTLP	Low-Power Watchdog Timer Time-out Period	10	16	27	ms	VDD = 3.3V-5V 1:512 Prescaler used
32	TOST	Oscillator Start-up Timer Period(1)	—	1024	—	Tosc	
33*	TPWRT	Power-up Timer Period, $\overline{PWRTE}$ = 0	40	65	140	ms	
34*	TIOZ	I/O high-impedance from $\overline{MCLR}$ Low or Watchdog Timer Reset	—	—	2.0	μs	
35	VBOR	Brown-out Reset Voltage(2)	2.55 2.30 1.80	2.70 2.45 1.90	2.85 2.60 2.10	V V V	BORV = 0 BORV = 1 (PIC16F1777/8/9) BORV = 1 (PIC16LF1777/8/9)
35A	VLPBOR	Low-Power Brown-out	1.8	2.1	2.5	V	LPBOR = 1
36*	VHYST	Brown-out Reset Hysteresis	0	25	75	mV	-40°C ≤ TA ≤ +85°C
37*	TBORDC	Brown-out Reset DC Response Time	1	3	35	μs	VDD ≤ VBOR

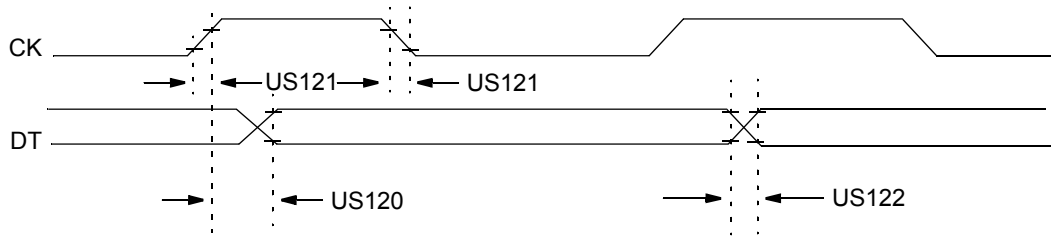
* These parameters are characterized but not tested.

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

Note 1: By design, the Oscillator Start-up Timer (OST) counts the first 1024 cycles, independent of frequency.

2: To ensure these voltage tolerances, VDD and VSS must be capacitively decoupled as close to the device as possible. 0.1 μF and 0.01 μF values in parallel are recommended.

FIGURE 36-15: EUSART SYNCHRONOUS TRANSMISSION (MASTER/SLAVE) TIMING

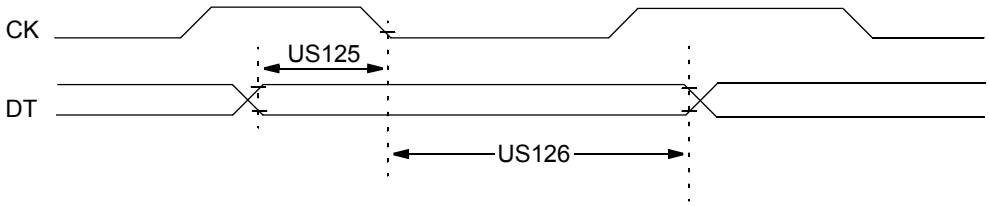


Note: Refer to Figure 36-4 for load conditions.

TABLE 36-23: EUSART SYNCHRONOUS TRANSMISSION REQUIREMENTS

Standard Operating Conditions (unless otherwise stated)						
Param. No.	**Symbol**	**Characteristic**	**Min.**	**Max.**	**Units**	**Conditions**
US120	TCKH2DTV	SYNC XMIT (Master and Slave) Clock high to data-out valid	—	80	ns	3.0V ≤ VDD ≤ 5.5V
			—	100	ns	1.8V ≤ VDD ≤ 5.5V
US121	TCKRF	Clock out rise time and fall time (Master mode)	—	45	ns	3.0V ≤ VDD ≤ 5.5V
			—	50	ns	1.8V ≤ VDD ≤ 5.5V
US122	TDTRF	Data-out rise time and fall time	—	45	ns	3.0V ≤ VDD ≤ 5.5V
			—	50	ns	1.8V ≤ VDD ≤ 5.5V

FIGURE 36-16: EUSART SYNCHRONOUS RECEIVE (MASTER/SLAVE) TIMING



Note: Refer to Figure 36-4 for load conditions.

TABLE 36-24: EUSART SYNCHRONOUS RECEIVE REQUIREMENTS

Standard Operating Conditions (unless otherwise stated)						
Param. No.	**Symbol**	**Characteristic**	**Min.**	**Max.**	**Units**	**Conditions**
US125	TDTV2CKL	SYNC RCV (Master and Slave) Data-setup before CK ↓ (DT hold time)	10	—	ns	
US126	TCKL2DTL	Data-hold after CK ↓ (DT hold time)	15	—	ns	

FIGURE 36-19: SPI SLAVE MODE TIMING (CKE = 0)

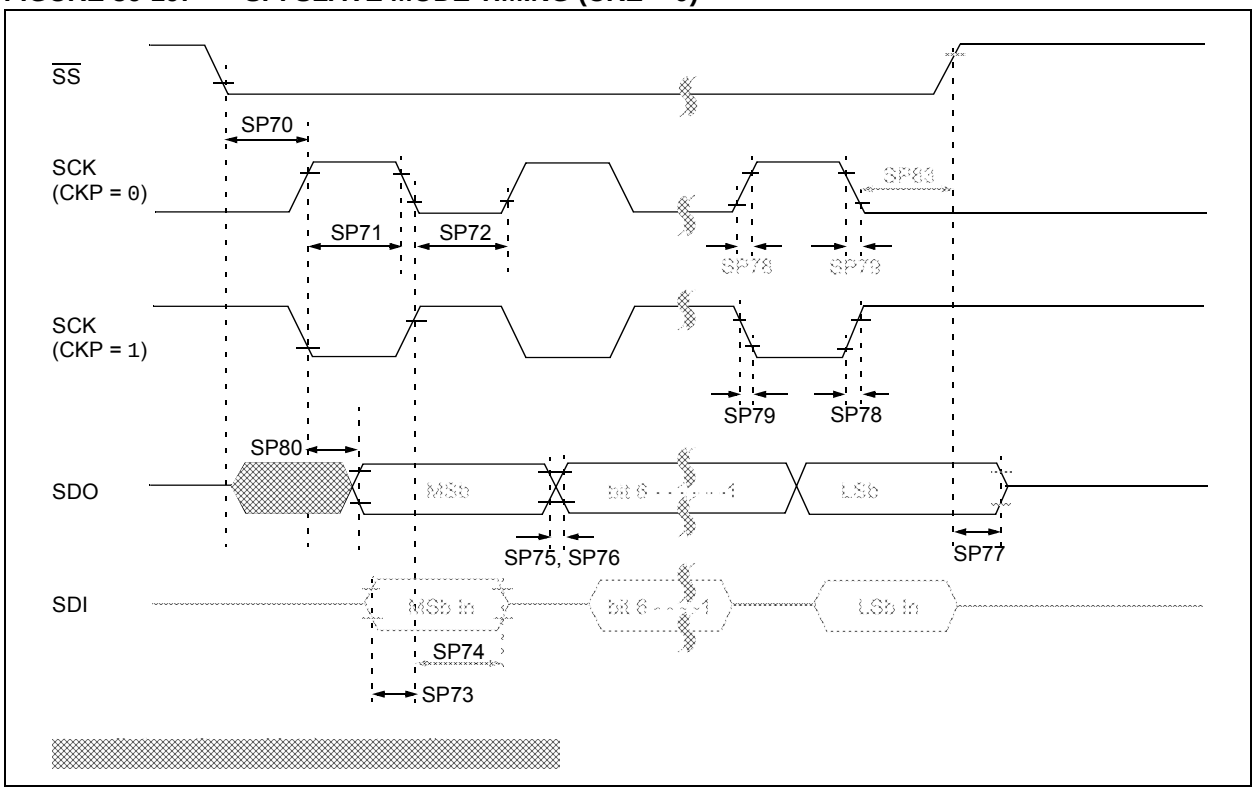
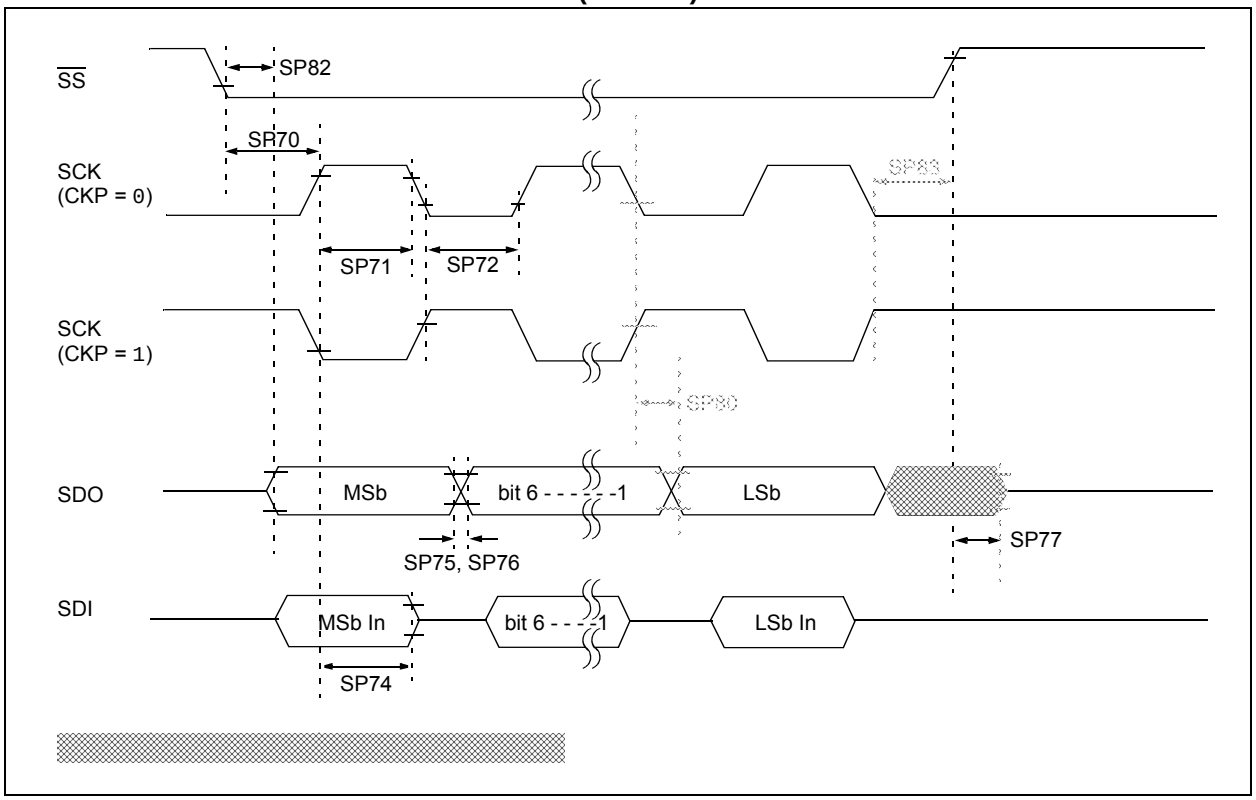


FIGURE 36-20: SPI SLAVE MODE TIMING (CKE = 1)



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

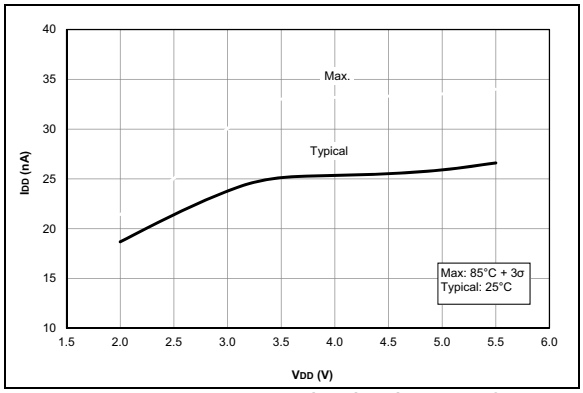


FIGURE 37-37: *IPD, Fixed Voltage Reference (FVR), ADC, PIC16F1777/8/9 Only.*

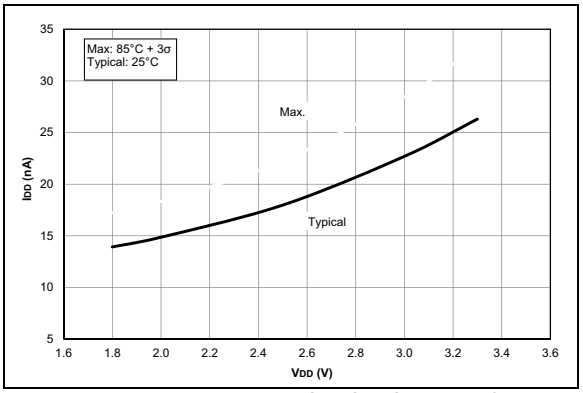


FIGURE 37-38: *IPD, Fixed Voltage Reference (FVR), DAC/Comparator, PIC16LF1777/8/9 Only.*

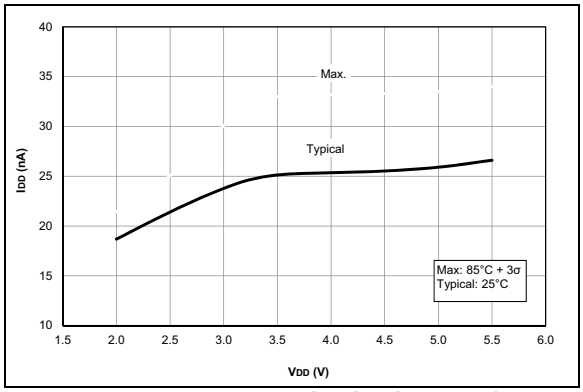


FIGURE 37-39: *IPD, Fixed Voltage Reference (FVR), DAC/Comparator, PIC16F1777/8/9 Only.*

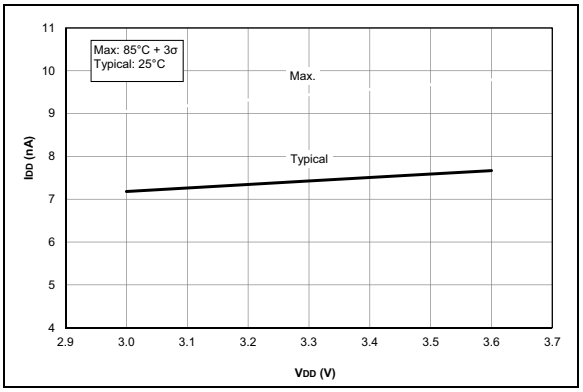


FIGURE 37-40: *IPD, Brown-Out Reset (BOR), BORV = 1, PIC16LF1777/8/9 Only.*

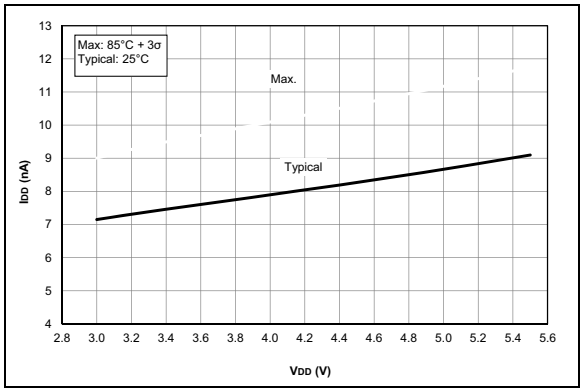


FIGURE 37-41: *IPD, Brown-Out Reset (BOR), BORV = 1, PIC16F1777/8/9 Only.*

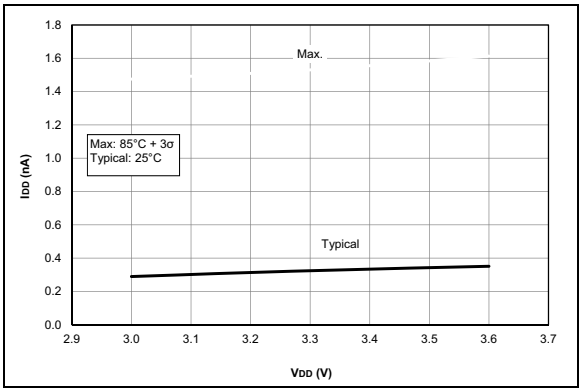


FIGURE 37-42: *IPD, LP Brown-Out Reset (LPBOR = 0), PIC16LF1777/8/9 Only.*