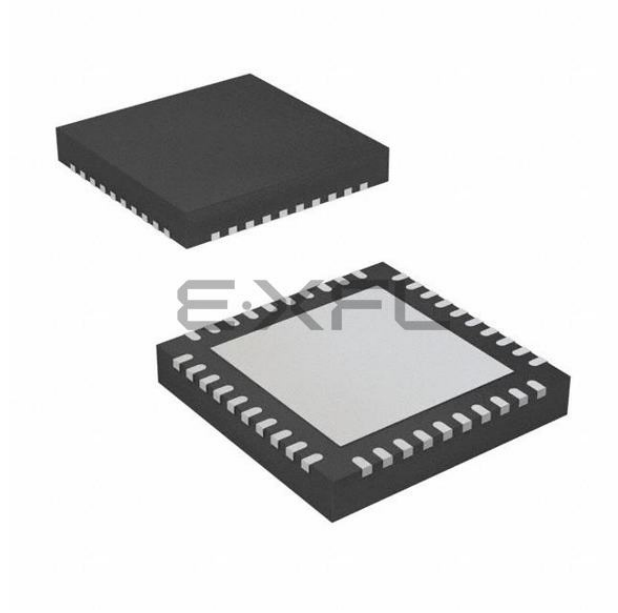


Welcome to E-XFL.COM

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, PSMC, PWM, WDT
Number of I/O	35
Program Memory Size	28KB (16K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	2.3V ~ 5.5V
Data Converters	A/D 14x12b; D/A 1x8b, 3x5b
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/pic16f1789-e-mv

TABLE 1-2: PIC16(L)F1788 PINOUT DESCRIPTION (CONTINUED)

Name	Function	Input Type	Output Type	Description
RA6/C2OUT(1)/OSC2/ CLKOUT/VCAP	RA6	TTL/ST	CMOS	General purpose I/O.
	C2OUT	—	CMOS	Comparator C2 output.
	OSC2	—	XTAL	Crystal/Resonator (LP, XT, HS modes).
	CLKOUT	—	CMOS	FOSC/4 output.
	VCAP	Power	Power	Filter capacitor for Voltage Regulator.
RA7/PSMC1CLK/PSMC2CLK/ PSMC3CLK/PSMC4- CLK/OSC1/CLKIN	RA7	TTL/ST	CMOS	General purpose I/O.
	PSMC1CLK	ST	—	PSMC1 clock input.
	PSMC2CLK	ST	—	PSMC2 clock input.
	PSMC3CLK	ST	—	PSMC3 clock input.
	PSMC4CLK	ST	—	PSMC4 clock input.
	OSC1	—	XTAL	Crystal/Resonator (LP, XT, HS modes).
	CLKIN	ST	—	External clock input (EC mode).
RB0/AN12/C2IN1+/PSMC1IN/ PSMC2IN/PSMC3IN/PSMC4IN/ CCP1(1)/INT	RB0	TTL/ST	CMOS	General purpose I/O.
	AN12	AN	—	ADC Channel 12 input.
	C2IN1+	AN	—	Comparator C2 positive input.
	PSMC1IN	ST	—	PSMC1 Event Trigger input.
	PSMC2IN	ST	—	PSMC2 Event Trigger input.
	PSMC3IN	ST	—	PSMC3 Event Trigger input.
	PSMC4IN	ST	—	PSMC4 Event Trigger input.
	CCP1	ST	CMOS	Capture/Compare/PWM1.
	INT	ST	—	External interrupt.
RB1/AN10/C1IN3-/C2IN3-/ C3IN3-/C4IN3-/OPA2OUT	RB1	TTL/ST	CMOS	General purpose I/O.
	AN10	AN	—	ADC Channel 10 input.
	C1IN3-	AN	—	Comparator C1 negative input.
	C2IN3-	AN	—	Comparator C2 negative input.
	C3IN3-	AN	—	Comparator C3 negative input.
	C4IN3-	AN	—	Comparator C4 negative input.
	OPA2OUT	—	AN	Operational Amplifier 2 output.
RB2/AN8/OPA2IN-/CLKR/ DAC3OUT1	RB2	TTL/ST	CMOS	General purpose I/O.
	AN8	AN	—	ADC Channel 8 input.
	OPA2IN-	AN	—	Operational Amplifier 2 inverting input.
	CLKR	—	CMOS	Clock output.
	DAC3OUT1	—	AN	Digital-to-Analog Converter output.
RB3/AN9/C1IN2-/C2IN2-/ C3IN2-/OPA2IN+/CCP2(1)	RB3	TTL/ST	CMOS	General purpose I/O.
	AN9	AN	—	ADC Channel 9 input.
	C1IN2-	AN	—	Comparator C1 negative input.
	C2IN2-	AN	—	Comparator C2 negative input.
	C3IN2-	AN	—	Comparator C3 negative input.
	OPA2IN+	AN	—	Operational Amplifier 2 non-inverting input.
	CCP2	ST	CMOS	Capture/Compare/PWM2.

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
HV = High Voltage XTAL = Crystal levels

Note 1: Pin functions can be assigned to one of two locations via software. See **Register 13-1**.

2: All pins have interrupt-on-change functionality.

TABLE 3-12: 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 29 (Continued)											
ED1h	PSMC3CON	PSMC3EN	PSMC3LD	P3DBFE	P3DBRE	P3MODE<3:0>				0000 0000	0000 0000
ED2h	PSMC3MDL	P3MDLEN	P3MDLPOL	P3MDLBIT	—	P3MSRC<3:0>				000- 0000	000- 0000
ED3h	PSMC3SYNC	P3POFST	P3PRPOL	P3DCPOL	—	—	P3SYNC<2:0>			000- -000	000- -000
ED4h	PSMC3CLK	—	—	P3CPRE<1:0>		—	—	P3CSRC<1:0>		--00 --00	--00 --00
ED5h	PSMC3OEN	—	—	—	—	—	—	P3OEB	P3OEA	---- --00	---- --00
ED6h	PSMC3POL	—	P3INPOL	—	—	—	—	P3POLB	P3POLA	-0-- --00	-0-- --00
ED7h	PSMC3BLNK	—	—	P3FEBM<1:0>		—	—	P3REBM<1:0>		--00 --00	--00 --00
ED8h	PSMC3REBS	P3REBSIN	—	—	P3REBSC4	P3REBSC3	P3REBSC2	P3REBSC1	—	0--0 000-	0--0 000-
ED9h	PSMC3FEBS	P3FEBSIN	—	—	P3FEBSC4	P3FEBSC3	P3FEBSC2	P3FEBSC1	—	0--0 000-	0--0 000-
EDAh	PSMC3PHS	P3PHSIN	—	—	P3PHSC4	P3PHSC3	P3PHSC2	P3PHSC1	P3PHST	0--0 0000	0--0 0000
EDBh	PSMC3DCS	P3DCSIN	—	—	P3DCSC4	P3DCSC3	P3DCSC2	P3DCSC1	P3DCST	0--0 0000	0--0 0000
EDCh	PSMC3PRS	P3PRSIN	—	—	P3PRSC4	P3PRSC3	P3PRSC2	P3PRSC1	P3PRST	0--0 0000	0--0 0000
EDDh	PSMC3ASDC	P3ASE	P3ASDEN	P3ARSEN	—	—	—	—	P3ASDOV	000- ---0	000- ---0
EDEh	PSMC3ASDL	—	—	—	—	—	—	P3ASDLB	P3ASDLA	---- --00	---- --00
EDFh	PSMC3ASDS	P3ASDSIN	—	—	P3ASDSC4	P3ASDSC3	P3ASDSC2	P3ASDSC1	—	0--0 000-	0--0 000-
EE0h	PSMC3INT	P3TOVIE	P3TPHIE	P3TDCIE	P3TPRIE	P3TOVIF	P3TPHIF	P3TDCIF	P3TPRIF	0000 0000	0000 0000
EE1h	PSMC3PHL	Phase Low Count								0000 0000	0000 0000
EE2h	PSMC3PHH	Phase High Count								0000 0000	0000 0000
EE3h	PSMC3DCL	Duty Cycle Low Count								0000 0000	0000 0000
EE4h	PSMC3DCH	Duty Cycle High Count								0000 0000	0000 0000
EE5h	PSMC3PRL	Period Low Count								0000 0000	0000 0000
EE6h	PSMC3PRH	Period High Count								0000 0000	0000 0000
EE7h	PSMC3TMRL	Time base Low Counter								0000 0001	0000 0001
EE8h	PSMC3TMRH	Time base High Counter								0000 0000	0000 0000
EE9h	PSMC3DBR	Rising Edge Dead-band Counter								0000 0000	0000 0000
EEAh	PSMC3DBF	Falling Edge Dead-band Counter								0000 0000	0000 0000
EEBh	PSMC3BLKR	Rising Edge Blanking Counter								0000 0000	0000 0000
EECh	PSMC3BLKF	Falling Edge Blanking Counter								0000 0000	0000 0000
EEDh	PSMC3FFA	—	—	—	—	Fractional Frequency Adjust Register				---- 0000	---- 0000
EEEh	PSMC3STR0	—	—	—	—	—	—	P3STRB	P3STRA	---- --01	---- --01
EEFh	PSMC3STR1	P3SSYNC	—	—	—	—	—	P3LSMEN	P3HSMEN	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: These registers can be addressed from any bank.
2: Unimplemented, read as '1'.
3: PIC16(L)F1789 only.
4: PIC16F1788/9 only.

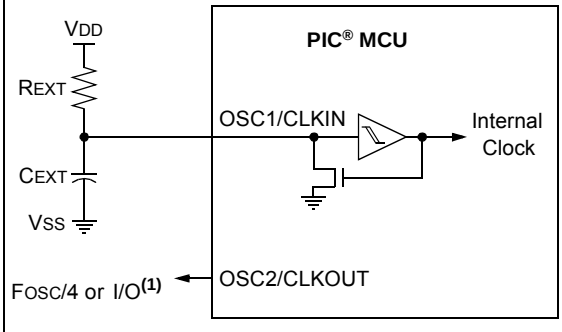
6.2.1.6 External RC Mode

The external Resistor-Capacitor (RC) modes support the use of an external RC circuit. This allows the designer maximum flexibility in frequency choice while keeping costs to a minimum when clock accuracy is not required.

The RC circuit connects to OSC1. OSC2/CLKOUT is available for general purpose I/O or CLKOUT. The function of the OSC2/CLKOUT pin is determined by the $\overline{\text{CLKOUTEN}}$ bit in Configuration Words.

Figure 6-6 shows the external RC mode connections.

FIGURE 6-6: EXTERNAL RC MODES



Recommended values: 10 kΩ ≤ REXT ≤ 100 kΩ, <3V
3 kΩ ≤ REXT ≤ 100 kΩ, 3-5V
CEXT > 20 pF, 2-5V

Note 1: Output depends upon $\overline{\text{CLKOUTEN}}$ bit of the Configuration Words.

The RC oscillator frequency is a function of the supply voltage, the resistor (REXT) and capacitor (CEXT) values and the operating temperature. Other factors affecting the oscillator frequency are:

- threshold voltage variation
- component tolerances
- packaging variations in capacitance

The user also needs to take into account variation due to tolerance of external RC components used.

6.2.2 INTERNAL CLOCK SOURCES

The device may be configured to use the internal oscillator block as the system clock by performing one of the following actions:

- Program the FOSC<2:0> bits in Configuration Words to select the INTOSC clock source, which will be used as the default system clock upon a device Reset.
- Write the SCS<1:0> bits in the OSCCON register to switch the system clock source to the internal oscillator during run-time. See **Section 6.3 "Clock Switching"**for more information.

In **INTOSC** mode, OSC1/CLKIN is available for general purpose I/O. OSC2/CLKOUT is available for general purpose I/O or CLKOUT.

The function of the OSC2/CLKOUT pin is determined by the $\overline{\text{CLKOUTEN}}$ bit in Configuration Words.

The internal oscillator block has two independent oscillators and a dedicated Phase-Lock Loop, HFPLL that can produce one of three internal system clock sources.

1. The **HFINTOSC** (High-Frequency Internal Oscillator) is factory calibrated and operates at 16 MHz. The HFINTOSC source is generated from the 500 kHz MFINTOSC source and the dedicated Phase-Lock Loop, HFPLL. The frequency of the HFINTOSC can be user-adjusted via software using the OSCTUNE register (Register 6-3).
2. The **MFINTOSC** (Medium-Frequency Internal Oscillator) is factory calibrated and operates at 500 kHz. The frequency of the MFINTOSC can be user-adjusted via software using the OSCTUNE register (Register 6-3).
3. The **LFINTOSC** (Low-Frequency Internal Oscillator) is uncalibrated and operates at 31 kHz.

12.3.2 ERASING FLASH PROGRAM MEMORY

While executing code, program memory can only be erased by rows. To erase a row:

1. Load the EEADRH:EEADRL register pair with the address of new row to be erased.
2. Clear the CFGS bit of the EECON1 register.
3. Set the EEPGD, FREE, and WREN bits of the EECON1 register.
4. Write 55h, then AAh, to EECON2 (Flash programming unlock sequence).
5. Set control bit WR of the EECON1 register to begin the erase operation.
6. Poll the FREE bit in the EECON1 register to determine when the row erase has completed.

See Example 12-4.

After the "BSF EECON1,WR" instruction, the processor requires two cycles to set up the erase operation. The user must place two NOP instructions after the WR bit is set. The processor will halt internal operations for the typical 2 ms erase time. This is not Sleep mode as the clocks and peripherals will continue to run. After the erase cycle, the processor will resume operation with the third instruction after the EECON1 write instruction.

12.3.3 WRITING TO FLASH PROGRAM MEMORY

Program memory is programmed using the following steps:

1. Load the starting address of the word(s) to be programmed.
2. Load the write latches 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 12-2 (block writes to program memory with 32 write latches) for more details. The write latches are aligned to the address boundary defined by EEADRL as shown in Table 12-1. Write operations do not cross these boundaries. At the completion of a program memory write operation, the write latches are reset to contain 0x3FFF.

The following steps should be completed to load the write latches and program a block of program memory. These steps are divided into two parts. First, all write latches are loaded with data except for the last program memory location. Then, the last write latch is loaded and the programming sequence is initiated. A special unlock sequence is required to load a write latch with data or initiate a Flash programming operation. This unlock sequence should not be interrupted.

1. Set the EEPGD and WREN bits of the EECON1 register.
2. Clear the CFGS bit of the EECON1 register.
3. Set the LWLO bit of the EECON1 register. When the LWLO bit of the EECON1 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 EEADRH:EEADRL register pair with the address of the location to be written.
5. Load the EEDATH:EEDATL register pair with the program memory data to be written.
6. Write 55h, then AAh, to EECON2, then set the WR bit of the EECON1 register (Flash programming unlock sequence). The write latch is now loaded.
7. Increment the EEADRH:EEADRL 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 EECON1 register. When the LWLO bit of the EECON1 register is '0', the write sequence will initiate the write to Flash program memory.
10. Load the EEDATH:EEDATL register pair with the program memory data to be written.
11. Write 55h, then AAh, to EECON2, then set the WR bit of the EECON1 register (Flash programming unlock sequence). The entire latch block is now written to Flash program memory.

It is not necessary to load the entire write latch block with user program data. However, the entire write latch block will be written to program memory.

An example of the complete write sequence for eight words is shown in Example 12-5. The initial address is loaded into the EEADRH:EEADRL register pair; the eight words of data are loaded using indirect addressing.

After the "BSF EECON1,WR" instruction, the processor requires two cycles to set up the write operation. The user must place two NOP instructions after the WR bit is set. The processor will halt internal operations for the typical 2 ms, only during the cycle in which the write takes place (i.e., the last word of the block write). This is not Sleep mode as the clocks and peripherals will continue to run. The processor does not stall when LWLO = 1, loading the write latches. After the write cycle, the processor will resume operation with the third instruction after the EECON1 WRITE instruction.

REGISTER 12-6: EECON2: EEPROM CONTROL 2 REGISTER

W-0/0	W-0/0	W-0/0	W-0/0	W-0/0	W-0/0	W-0/0	W-0/0
EEPROM Control Register 2							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
S = Bit can only be set	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 **Data EEPROM Unlock Pattern bits**

To unlock writes, a 55h must be written first, followed by an AAh, before setting the WR bit of the EECON1 register. The value written to this register is used to unlock the writes. There are specific timing requirements on these writes. Refer to **Section 12.2.2 "Writing to the Data EEPROM Memory"** for more information.

TABLE 12-3: SUMMARY OF REGISTERS ASSOCIATED WITH DATA EEPROM

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
EECON1	EEPGD	CFGS	LWLO	FREE	WRERR	WREN	WR	RD	128
EECON2	EEPROM Control Register 2 (not a physical register)								129*
EEADRL	EEADRL<7:0>								127
EEADRH	—(1)	EEADRH<6:0>							127
EEDATL	EEDATL<7:0>								127
EEDATH	—	—	EEDATH<5:0>						127
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	97
PIE2	OSFIE	C2IE	C1IE	EEIE	BCL1IE	C4IE	C3IE	CCP2IE	99
PIR2	OSFIF	C2IF	C1IF	EEIF	BCL1IF	C4IF	C3IF	CCP2IF	103

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

* Page provides register information.

2: Unimplemented, read as '1'.

13.9 PORTD Registers (PIC16(L)F1789 only)

13.9.1 DATA REGISTER

PORTD is an 8-bit wide bidirectional port. The corresponding data direction register is TRISD (Register 13-27). Setting a TRISD bit (= 1) will make the corresponding PORTD pin an input (i.e., put the corresponding output driver in a High-Impedance mode). Clearing a TRISD bit (= 0) will make the corresponding PORTD pin an output (i.e., enable the output driver and put the contents of the output latch on the selected pin). Example 13-1 shows how to initialize an I/O port.

Reading the PORTD register (Register 13-26) 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 (LATD).

13.9.2 DIRECTION CONTROL

The TRISD register (Register 13-27) controls the PORTD pin output drivers, even when they are being used as analog inputs. The user should ensure the bits in the TRISD register are maintained set when using them as analog inputs. I/O pins configured as analog inputs always read '0'.

13.9.3 OPEN-DRAIN CONTROL

The ODCOND register (Register 13-31) controls the open-drain feature of the port. Open-drain operation is independently selected for each pin. When an ODCOND bit is set, the corresponding port output becomes an open-drain driver capable of sinking current only. When an ODCOND bit is cleared, the corresponding port output pin is the standard push-pull drive capable of sourcing and sinking current.

13.9.4 SLEW RATE CONTROL

The SLRCOND register (Register 13-32) controls the slew rate option for each port pin. Slew rate control is independently selectable for each port pin. When an SLRCOND bit is set, the corresponding port pin drive is slew rate limited. When an SLRCOND bit is cleared, The corresponding port pin drive slews at the maximum rate possible.

13.9.5 INPUT THRESHOLD CONTROL

The INLVLD register (Register 13-33) controls the input voltage threshold for each of the available PORTD 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 PORTD register and also the level at which an interrupt-on-change occurs, if that feature is enabled. See **Section 31.3 "DC Characteristics"** 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.

13.9.6 PORTD FUNCTIONS AND OUTPUT PRIORITIES

Each PORTD pin is multiplexed with other functions. The pins, their combined functions and their output priorities are shown in Table 13-9.

When multiple outputs are enabled, the actual pin control goes to the peripheral with the highest priority.

Analog input and some digital input functions are not included in the list below. These input functions can remain active when the pin is configured as an output. Certain digital input functions override other port functions and are included in the priority list.

TABLE 13-9: PORTD OUTPUT PRIORITY

Pin Name	Function Priority(1)
RD0	RD0
RD1	OPA3OUT RD1
RD2	RD2
RD3	PSMC4A RD3
RD4	PSMC3F RD4
RD5	PSMC3E RD5
RD6	PSMC3D C3OUT RD6
RD7	PSMC3C C4OUT RD7

Note 1: Priority listed from highest to lowest.

13.12 Register Definitions: PORTE

REGISTER 13-34: PORTE: PORTE REGISTER

U-0	U-0	U-0	U-0	R-x/u	R/W-x/u	R/W-x/u	R/W-x/u
—	—	—	—	RE3	RE2(1)	RE1(1)	RE0(1)
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 **RE<3:0>**: PORTE Input Pin bit(1)
1 = Port pin is > V_{IH}
0 = Port pin is < V_{IL}

Note 1: RE<2:0> are available on PIC16(L)F1789 only.

REGISTER 13-35: TRISE: PORTE TRI-STATE REGISTER

U-0	U-0	U-0	U-0	U-1(1)	R/W-1/1	R/W-1/1	R/W-1/1
—	—	—	—	—	TRISE2(2)	TRISE1(2)	TRISE0(2)
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 **Unimplemented**: Read as '1'

bit 2-0 **TRISA<2:0>:** PORTA Tri-State Control bit(2)
1 = PORTA pin configured as an input (tri-stated)
0 = PORTA pin configured as an output

Note 1: Unimplemented, read as '1'.
2: TRISE<2:0> are available on PIC16(L)F1789 only.

17.2.6 A/D CONVERSION PROCEDURE

This is an example procedure for using the ADC to perform an Analog-to-Digital conversion:

1. Configure Port:
   - Disable pin output driver (Refer to the TRIS register)
   - Configure pin as analog (Refer to the ANSEL register)
   - Disable weak pull-ups either globally (Refer to the OPTION_REG register) or individually (Refer to the appropriate WPUx register)
2. Configure the ADC module:
   - Select ADC conversion clock
   - Configure voltage reference
   - Select ADC input channel
   - Turn on ADC module
3. Configure ADC interrupt (optional):
   - Clear ADC interrupt flag
   - Enable ADC interrupt
   - Enable peripheral interrupt
   - Enable global interrupt(1)
4. Wait the required acquisition time(2).
5. Start conversion by setting the GO/$\overline{\text{DONE}}$ bit.
6. Wait for ADC conversion to complete by one of the following:
   - Polling the GO/$\overline{\text{DONE}}$ bit
   - Waiting for the ADC interrupt (interrupts enabled)
7. Read ADC Result.
8. Clear the ADC interrupt flag (required if interrupt is enabled).

Note 1: The global interrupt can be disabled if the user is attempting to wake-up from Sleep and resume in-line code execution.

2: Refer to **Section 17.4 "ADC Acquisition Requirements"**.

EXAMPLE 17-1: A/D CONVERSION

```
;This code block configures the ADC
;for polling, Vdd and Vss references, Frc
;clock
MOVWF     ADCON1       ;Vdd and Vss Vref
MOVLW     B'00001111'  ;set negative input
MOVWF     ADCON2       ;to negative
                       ;reference
BANKSEL   TRISA        ;
BSF       TRISA,0      ;Set RA0 to input
BANKSEL   ANSEL        ;
BSF       ANSEL,0      ;Set RA0 to analog
BANKSEL   WPUA         ;
BCF       WPUA,0       ;Disable weak
          pull-up on RA0
BANKSEL   ADCON0       ;
MOVLW     B'00000001'  ;Select channel AN0
MOVWF     ADCON0       ;Turn ADC On
CALL      SampleTime   ;Acquisiton delay
BSF       ADCON0,ADGO  ;Start conversion
BTFSC     ADCON0,ADGO  ;Is conversion done?
GOTO      $-1          ;No, test again
BANKSEL   ADRESH       ;
MOVF      ADRESH,W     ;Read upper 2 bits
MOVWF     RESULTHI     ;store in GPR space
```

TABLE 17-3: SUMMARY OF REGISTERS ASSOCIATED WITH ADC

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ADCON0	ADRMD	CHS<4:0>					GO/$\overline{\text{DONE}}$	ADON	177
ADCON1	ADFM	ADCS<2:0>			—	ADNREF	ADPREF<1:0>		178
ADCON2	TRIGSEL<3:0>				CHSN<3:0>				179
ADRESH	A/D Result Register High								180, 181
ADRESL	A/D Result Register Low								180, 181
ANSELA	ANSA7	—	ANSA5	ANSA4	ANSA3	ANSA2	ANSA1	ANSA0	137
ANSELB	—	ANSB6	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1	ANSB0	143
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	97
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	98
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	102
TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	136
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	142
FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFVR<1:0>		ADFVR<1:0>		167

Legend: x = unknown, u = unchanged, – = unimplemented read as '0', q = value depends on condition. Shaded cells are not used for the ADC module.

TABLE 23-5: SUMMARY OF REGISTERS ASSOCIATED WITH TIMER1

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ANSELB	—	ANSB6	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1	ANSB0	143
CCP1CON	—	—	DC1B<1:0>		CCP1M<3:0>				231
CCP2CON	—	—	DC2B<1:0>		CCP2M<3:0>				231
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	97
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	98
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	102
TMR1H	Holding Register for the Most Significant Byte of the 16-bit TMR1 Register								209*
TMR1L	Holding Register for the Least Significant Byte of the 16-bit TMR1 Register								209*
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	142
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	147
T1CON	TMR1CS<1:0>		T1CKPS<1:0>		T1OSCEN	$\overline{\text{T1SYNC}}$	—	TMR1ON	217
T1GCON	TMR1GE	T1GPOL	T1GTM	T1GSPM	T1GGO/$\overline{\text{DONE}}$	T1GVAL	T1GSS<1:0>		218

Legend: — = unimplemented location, read as '0'. Shaded cells are not used by the Timer1 module.

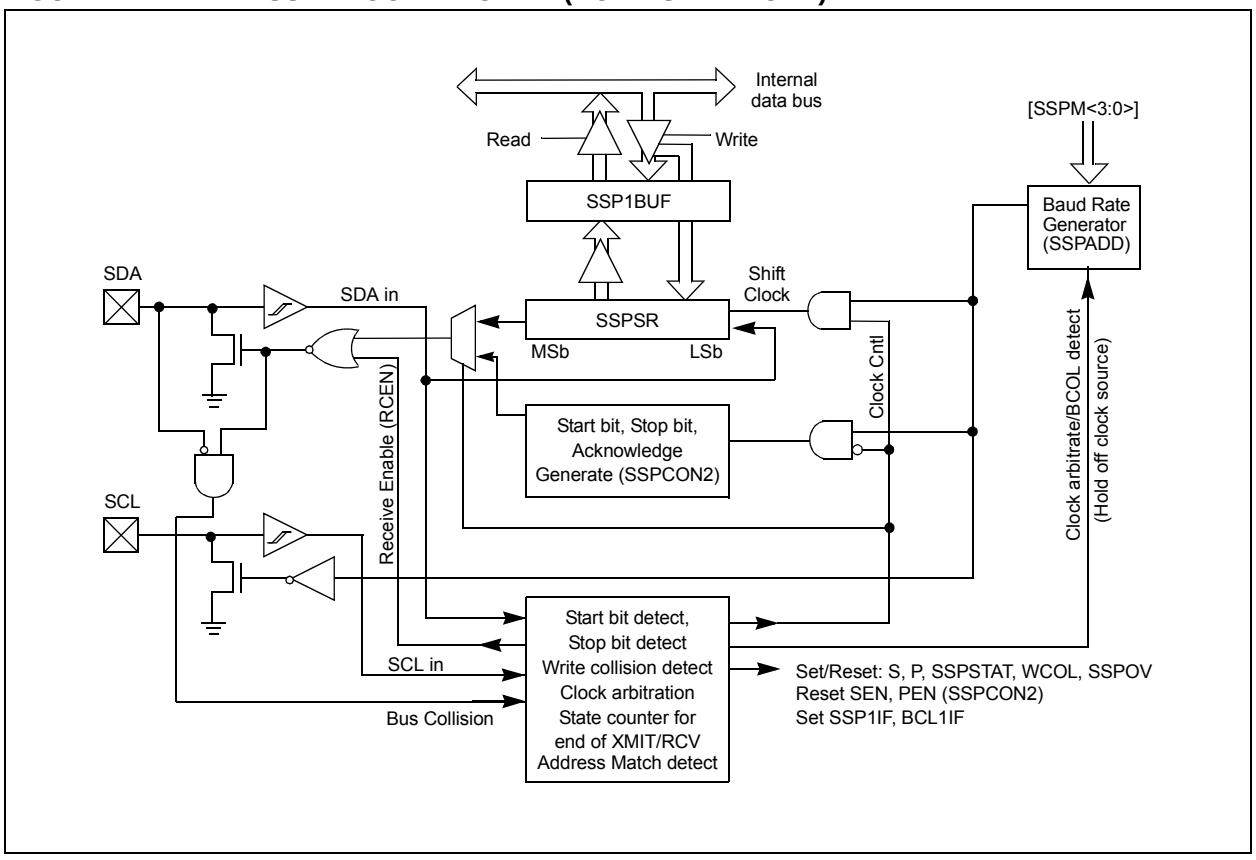
* Page provides register information.

The I^2C interface supports the following modes and features:

- Master mode
- Slave mode
- Byte NACKing (Slave mode)
- Limited multi-master support
- 7-bit and 10-bit addressing
- Start and Stop interrupts
- Interrupt masking
- Clock stretching
- Bus collision detection
- General call address matching
- Address masking
- Address Hold and Data Hold modes
- Selectable SDA hold times

Figure 27-2 is a block diagram of the I^2C interface module in Master mode. Figure 27-3 is a diagram of the I^2C interface module in Slave mode.

FIGURE 27-2: MSSP BLOCK DIAGRAM (I^2C MASTER MODE)



27.4.9 ACKNOWLEDGE SEQUENCE

The 9th SCL pulse for any transferred byte in I^2C is dedicated as an Acknowledge. It allows receiving devices to respond back to the transmitter by pulling the SDA line low. The transmitter must release control of the line during this time to shift in the response. The Acknowledge ($\overline{ACK}$) is an active-low signal, pulling the SDA line low indicated to the transmitter that the device has received the transmitted data and is ready to receive more.

The result of an $\overline{ACK}$ is placed in the ACKSTAT bit of the SSPCON2 register.

Slave software, when the AHEN and DHEN bits are set, allow the user to set the $\overline{ACK}$ value sent back to the transmitter. The ACKDT bit of the SSPCON2 register is set/cleared to determine the response.

Slave hardware will generate an $\overline{ACK}$ response if the AHEN and DHEN bits of the SSPCON3 register are clear.

There are certain conditions where an $\overline{ACK}$ will not be sent by the slave. If the BF bit of the SSPSTAT register or the SSPOV bit of the SSPCON1 register are set when a byte is received.

When the module is addressed, after the 8th falling edge of SCL on the bus, the ACKTIM bit of the SSPCON3 register is set. The ACKTIM bit indicates the acknowledge time of the active bus. The ACKTIM Status bit is only active when the AHEN bit or DHEN bit is enabled.

27.5 I^2C SLAVE MODE OPERATION

The MSSP Slave mode operates in one of four modes selected in the SSPM bits of SSPCON1 register. The modes can be divided into 7-bit and 10-bit Addressing mode. 10-bit Addressing modes operate the same as 7-bit with some additional overhead for handling the larger addresses.

Modes with Start and Stop bit interrupts operated the same as the other modes with SSP1IF additionally getting set upon detection of a Start, Restart, or Stop condition.

27.5.1 SLAVE MODE ADDRESSES

The SSPADD register (Register 27-6) contains the Slave mode address. The first byte received after a Start or Restart condition is compared against the value stored in this register. If the byte matches, the value is loaded into the SSPBUF register and an interrupt is generated. If the value does not match, the module goes idle and no indication is given to the software that anything happened.

The SSP Mask register (Register 27-5) affects the address matching process. See **Section 27.5.9 "SSP Mask Register"** for more information.

27.5.1.1 I^2C Slave 7-bit Addressing Mode

In 7-bit Addressing mode, the LSb of the received data byte is ignored when determining if there is an address match.

27.5.1.2 I^2C Slave 10-bit Addressing Mode

In 10-bit Addressing mode, the first received byte is compared to the binary value of '1 1 1 1 0 A9 A8 0'. A9 and A8 are the two MSb of the 10-bit address and stored in bits 2 and 1 of the SSPADD register.

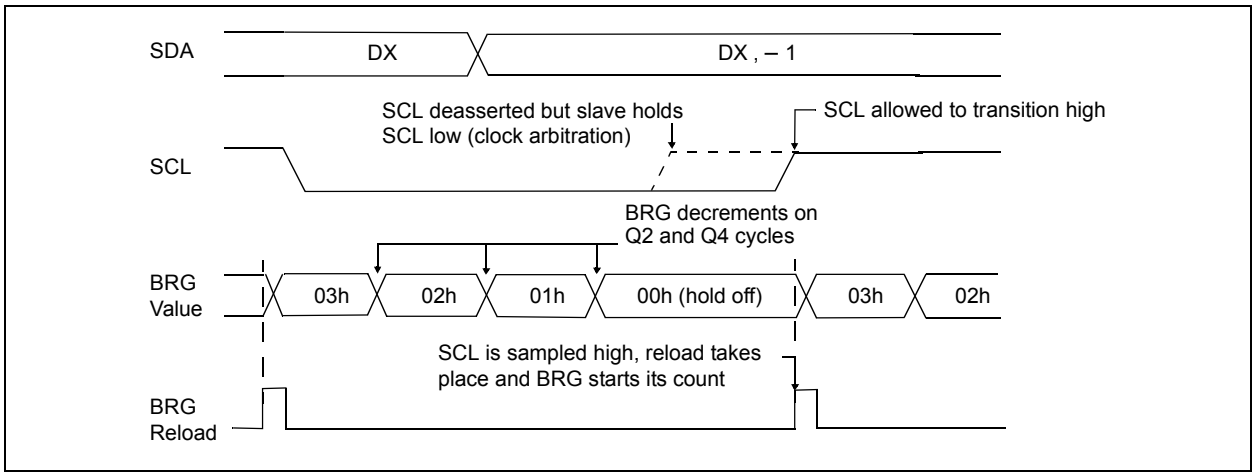
After the acknowledge of the high byte the UA bit is set and SCL is held low until the user updates SSPADD with the low address. The low address byte is clocked in and all eight bits are compared to the low address value in SSPADD. Even if there is not an address match; SSP1IF and UA are set, and SCL is held low until SSPADD is updated to receive a high byte again. When SSPADD is updated the UA bit is cleared. This ensures the module is ready to receive the high address byte on the next communication.

A high and low address match as a write request is required at the start of all 10-bit addressing communication. A transmission can be initiated by issuing a Restart once the slave is addressed, and clocking in the high address with the $R/\overline{W}$ bit set. The slave hardware will then acknowledge the read request and prepare to clock out data. This is only valid for a slave after it has received a complete high and low address byte match.

27.6.2 CLOCK ARBITRATION

Clock arbitration occurs when the master, during any receive, transmit or Repeated Start/Stop condition, releases the SCL pin (SCL allowed to float high). When the SCL pin is allowed to float high, the Baud Rate Generator (BRG) is suspended from counting until the SCL pin is actually sampled high. When the SCL pin is sampled high, the Baud Rate Generator is reloaded with the contents of SSPADD<7:0> and begins counting. This ensures that the SCL high time will always be at least one BRG rollover count in the event that the clock is held low by an external device (Figure 27-25).

FIGURE 27-25: BAUD RATE GENERATOR TIMING WITH CLOCK ARBITRATION



27.6.3 WCOL STATUS FLAG

If the user writes the SSPBUF when a Start, Restart, Stop, Receive or Transmit sequence is in progress, the WCOL is set and the contents of the buffer are unchanged (the write does not occur). Any time the WCOL bit is set it indicates that an action on SSPBUF was attempted while the module was not idle.

Note:	Because queuing of events is not allowed, writing to the lower five bits of SSPCON2 is disabled until the Start condition is complete.

TABLE 28-3: 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 SPBRGH, SPBRGL register pair

TABLE 28-4: 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
BAUDCON	ABDOVF	RCIDL	—	SCKP	BRG16	—	WUE	ABDEN	356
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	355
SPBRGL	BRG<7:0>								357
SPBRGH	BRG<15:8>								357
TXSTA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	354

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

* Page provides register information.

FIGURE 28-7: AUTO-WAKE-UP BIT (WUE) TIMING DURING NORMAL OPERATION

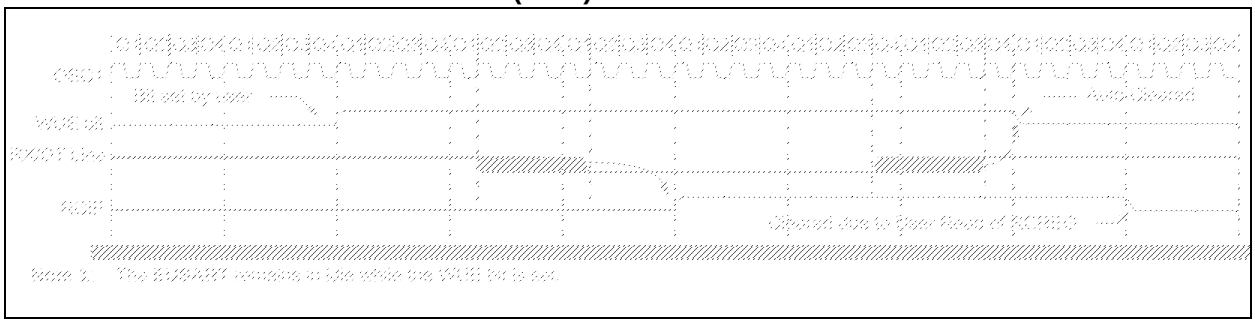
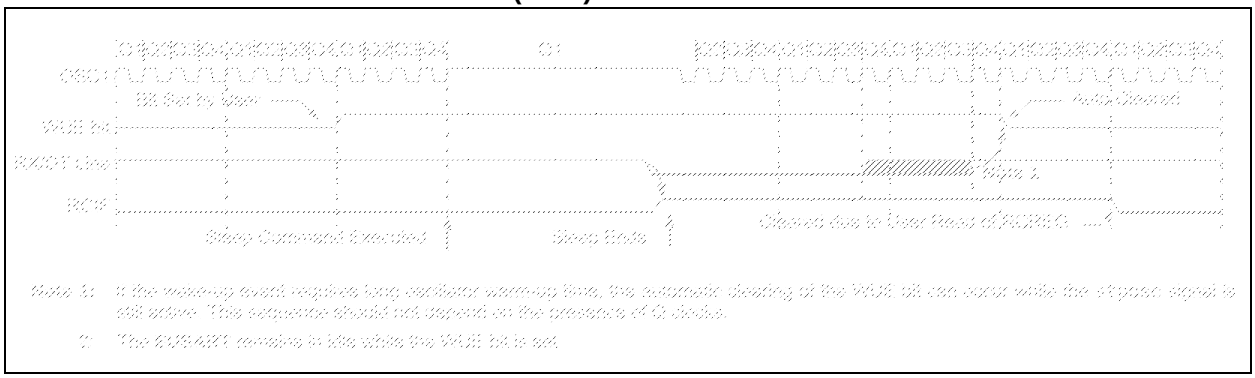


FIGURE 28-8: AUTO-WAKE-UP BIT (WUE) TIMINGS DURING SLEEP



28.5.1.5 Synchronous Master Reception

Data is received at the RX/DT pin. The RX/DT pin output driver is automatically disabled when the EUSART is configured for synchronous master receive operation.

In Synchronous mode, reception is enabled by setting either the Single Receive Enable bit (SREN of the RCSTA register) or the Continuous Receive Enable bit (CREN of the RCSTA register).

When SREN is set and CREN is clear, only as many clock cycles are generated as there are data bits in a single character. The SREN bit is automatically cleared at the completion of one character. When CREN is set, clocks are continuously generated until CREN is cleared. If CREN is cleared in the middle of a character the CK clock stops immediately and the partial character is discarded. If SREN and CREN are both set, then SREN is cleared at the completion of the first character and CREN takes precedence.

To initiate reception, set either SREN or CREN. Data is sampled at the RX/DT pin on the trailing edge of the TX/CK clock pin and is shifted into the Receive Shift Register (RSR). When a complete character is received into the RSR, the RCIF bit is set and the character is automatically transferred to the two character receive FIFO. The Least Significant eight bits of the top character in the receive FIFO are available in RCREG. The RCIF bit remains set as long as there are unread characters in the receive FIFO.

Note:	If the RX/DT function is on an analog pin, the corresponding ANSEL bit must be cleared for the receiver to function.

28.5.1.6 Slave Clock

Synchronous data transfers use a separate clock line, which is synchronous with the data. A device configured as a slave receives the clock on the TX/CK line. The TX/CK pin output driver is automatically disabled when the device is configured for synchronous slave transmit or receive operation. Serial data bits change on the leading edge to ensure they are valid at the trailing edge of each clock. One data bit is transferred for each clock cycle. Only as many clock cycles should be received as there are data bits.

Note:	If the device is configured as a slave and the TX/CK function is on an analog pin, the corresponding ANSEL bit must be cleared.

28.5.1.7 Receive Overrun Error

The receive FIFO buffer can hold two characters. An overrun error will be generated if a third character, in its entirety, is received before RCREG is read to access the FIFO. When this happens the OERR bit of the RCSTA register is set. Previous data in the FIFO will not be overwritten. The two characters in the FIFO buffer can be read, however, no additional characters will be received until the error is cleared. The OERR bit can only be cleared by clearing the overrun condition. If the overrun error occurred when the SREN bit is set and CREN is clear then the error is cleared by reading RCREG. If the overrun occurred when the CREN bit is set then the error condition is cleared by either clearing the CREN bit of the RCSTA register or by clearing the SPEN bit which resets the EUSART.

28.5.1.8 Receiving 9-bit Characters

The EUSART supports 9-bit character reception. When the RX9 bit of the RCSTA register is set, the EUSART will shift nine bits into the RSR for each character received. The RX9D bit of the RCSTA register is the ninth, and Most Significant, data bit of the top unread character in the receive FIFO. When reading 9-bit data from the receive FIFO buffer, the RX9D data bit must be read before reading the eight Least Significant bits from the RCREG.

28.5.1.9 Synchronous Master Reception Set-up:

1. Initialize the SPBRGH, SPBRGL register pair for the appropriate baud rate. Set or clear the BRGH and BRG16 bits, as required, to achieve the desired baud rate.
2. Clear the ANSEL bit for the RX pin (if applicable).
3. Enable the synchronous master serial port by setting bits SYNC, SPEN and CSRC.
4. Ensure bits CREN and SREN are clear.
5. If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
6. If 9-bit reception is desired, set bit RX9.
7. Start reception by setting the SREN bit or for continuous reception, set the CREN bit.
8. Interrupt flag bit RCIF will be set when reception of a character is complete. An interrupt will be generated if the enable bit RCIE was set.
9. Read the RCSTA register to get the ninth bit (if enabled) and determine if any error occurred during reception.
10. Read the 8-bit received data by reading the RCREG register.
11. If an overrun error occurs, clear the error by either clearing the CREN bit of the RCSTA register or by clearing the SPEN bit which resets the EUSART.

30.0 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 30-3 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)

TABLE 30-1: 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 30-2: ABBREVIATION DESCRIPTIONS

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

- 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 4 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.

30.1 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.

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

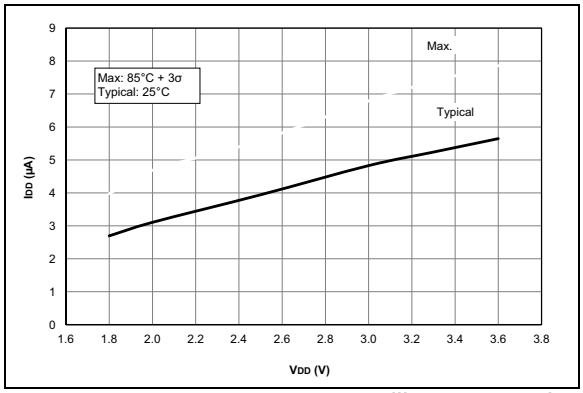


FIGURE 32-7: *IDD, EC Oscillator LP Mode, Fosc = 32 kHz, PIC16LF1788/9 Only.*

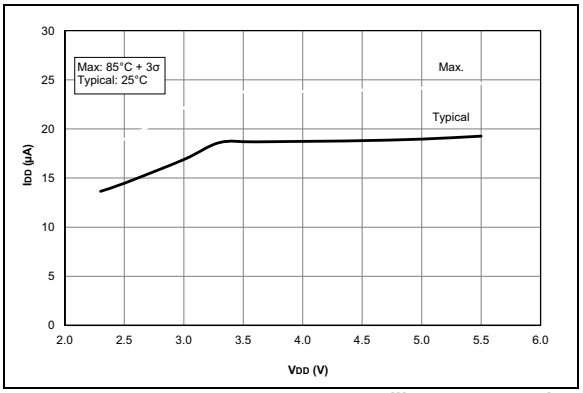


FIGURE 32-8: *IDD, EC Oscillator LP Mode, Fosc = 32 kHz, PIC16F1788/9 Only.*

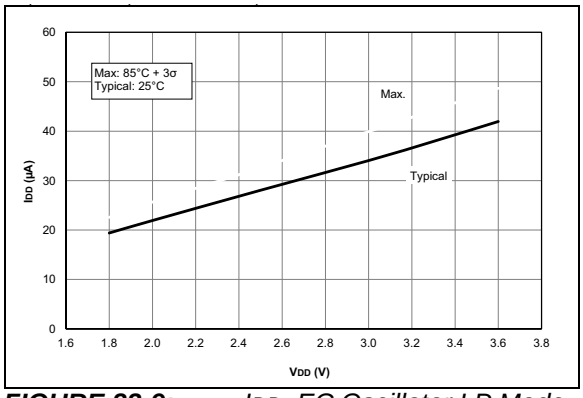


FIGURE 32-9: *IDD, EC Oscillator LP Mode, Fosc = 500 kHz, PIC16LF1788/9 Only.*

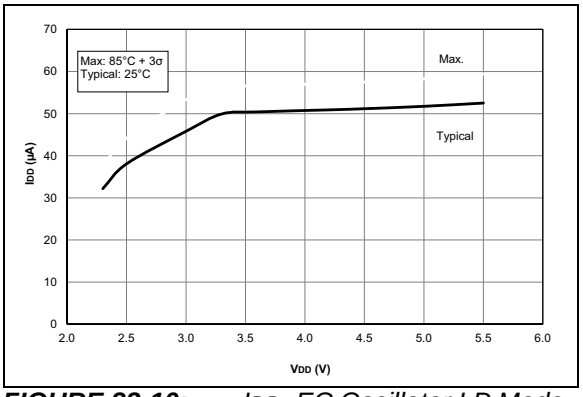


FIGURE 32-10: *IDD, EC Oscillator LP Mode, Fosc = 500 kHz, PIC16F1788/9 Only.*

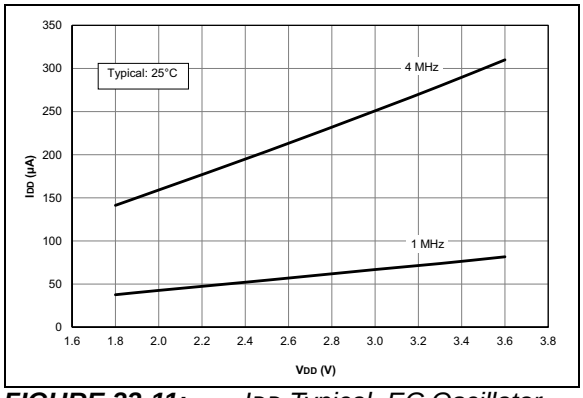


FIGURE 32-11: *IDD Typical, EC Oscillator MP Mode, PIC16LF1788/9 Only.*

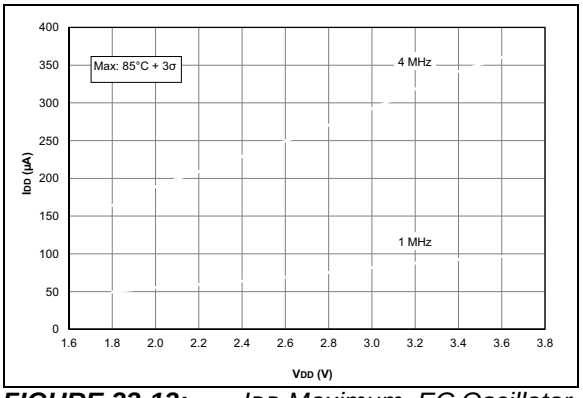


FIGURE 32-12: *IDD Maximum, EC Oscillator MP Mode, PIC16LF1788/9 Only.*

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

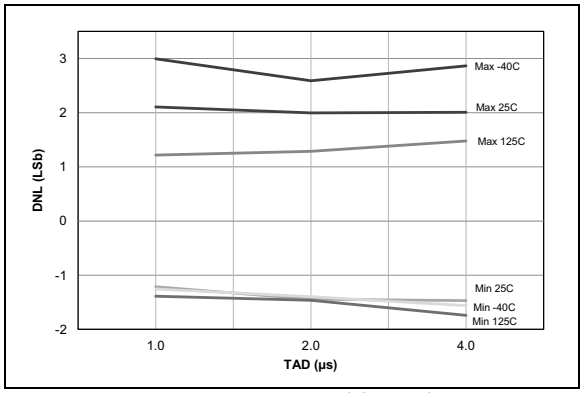


FIGURE 32-97: *ADC 12-bit Mode, Single-Ended DNL, VDD = 5.5V, VREF = 5.5V.*

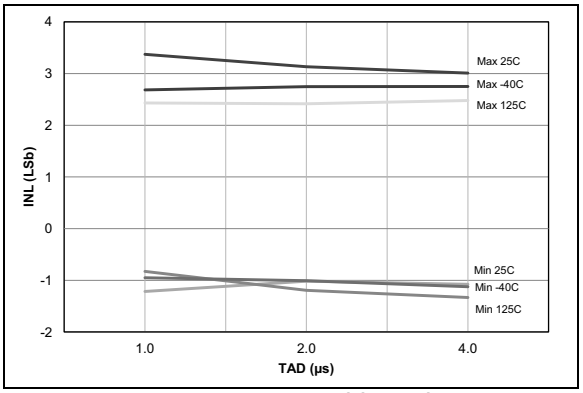


FIGURE 32-98: *ADC 12-bit Mode, Single-Ended INL, VDD = 5.5V, VREF = 5.5V.*

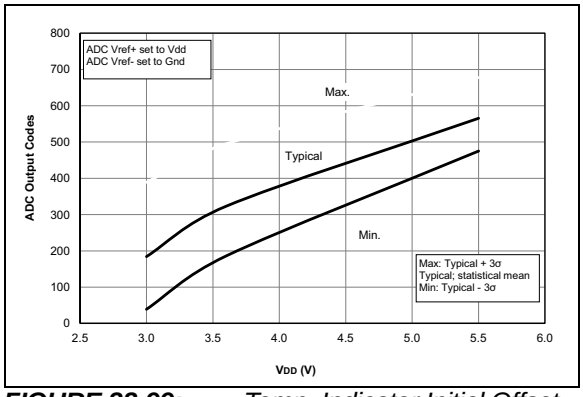


FIGURE 32-99: *Temp. Indicator Initial Offset, High Range, Temp. = 20°C, PIC16F1788/9 Only.*

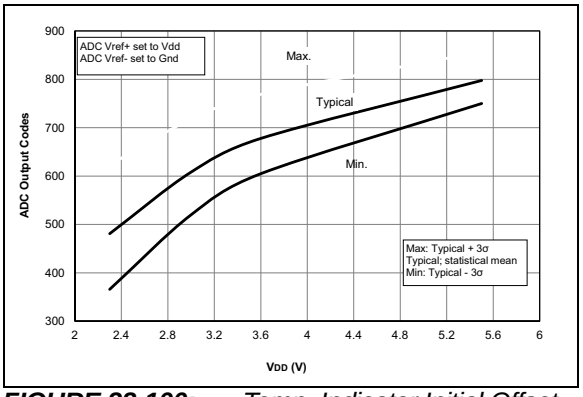


FIGURE 32-100: *Temp. Indicator Initial Offset, Low Range, Temp. = 20°C, PIC16F1788/9 Only.*

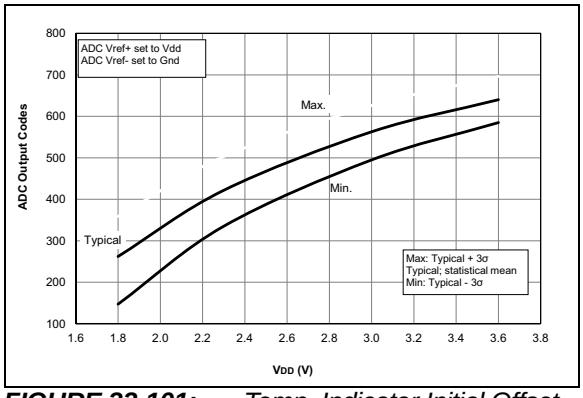


FIGURE 32-101: *Temp. Indicator Initial Offset, Low Range, Temp. = 20°C, PIC16LF1788/9 Only.*

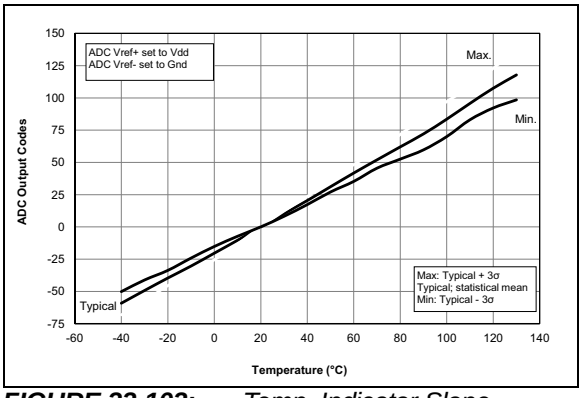
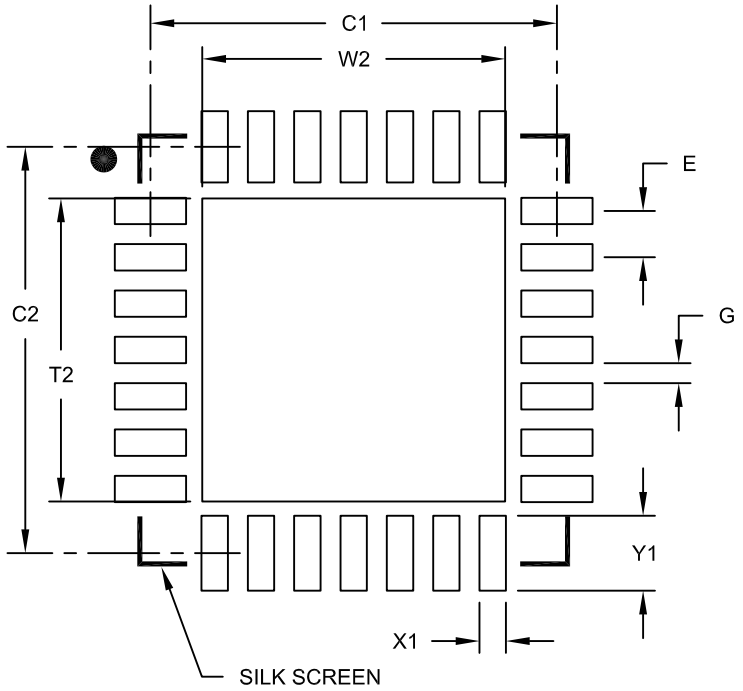


FIGURE 32-102: *Temp. Indicator Slope Normalized to 20°C, High Range, VDD = 5.5V, PIC16F1788/9 Only.*

28-Lead Plastic Quad Flat, No Lead Package (ML) – 6x6 mm Body [QFN] with 0.55 mm Contact Length

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		0.65 BSC	
Optional Center Pad Width	W2			4.25
Optional Center Pad Length	T2			4.25
Contact Pad Spacing	C1		5.70	
Contact Pad Spacing	C2		5.70	
Contact Pad Width (X28)	X1			0.37
Contact Pad Length (X28)	Y1			1.00
Distance Between Pads	G	0.20		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2105A