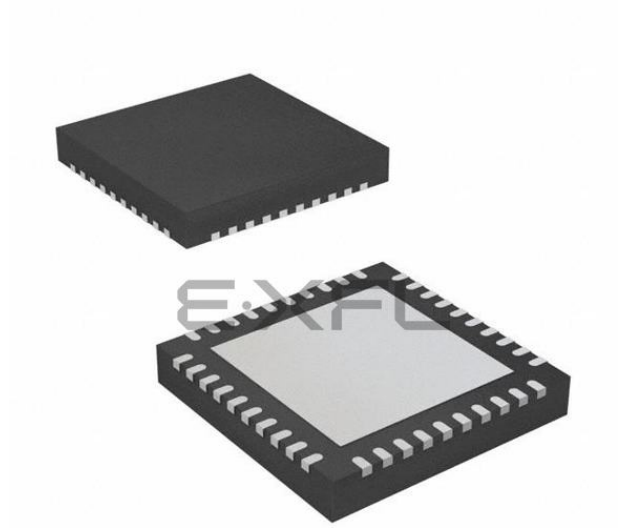


Welcome to E-XFL.COM

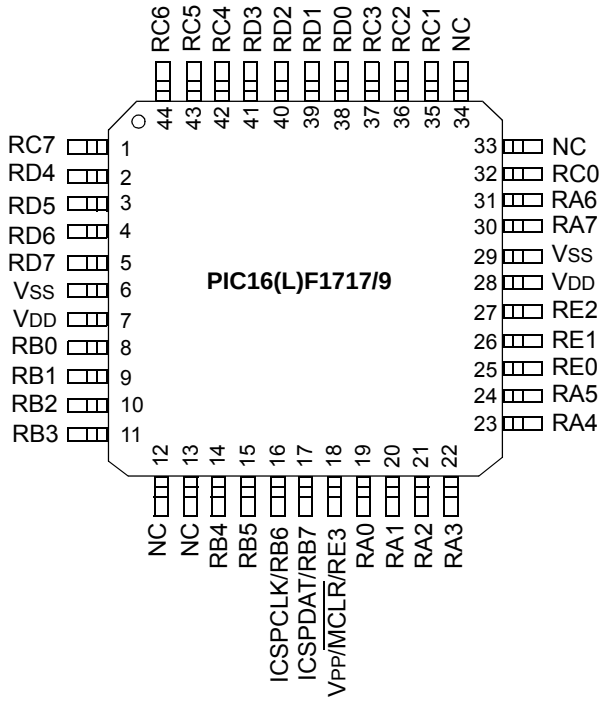
What is "Embedded - Microcontrollers"?

"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	35
Program Memory Size	28KB (16K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 28x10b; D/A 1x5b, 1x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°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/pic16lf1719t-i-mv

FIGURE 5: 44-PIN TQFP (10X10)



Note: See Table 2 for the pin allocation table.

Table of Contents

© 2014-2015 Microchip Technology Inc.

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 4											
20Ch	WPUA	WPUA7	WPUA6	WPUA5	WPUA4	WPUA3	WPUA2	WPUA1	WPUA0	1111 1111	1111 1111
20Dh	WPUB	WPUB7	WPUB6	WPUB5	WPUB4	WPUB3	WPUB2	WPUB1	WPUB0	1111 1111	1111 1111
20Eh	WPUC	WPUC7	WPUC6	WPUC5	WPUC4	WPUC3	WPUC2	WPUC1	WPUC0	1111 1111	1111 1111
20Fh	WPUD(1)	WPUD7	WPUD6	WPUD5	WPUD4	WPUD3	WPUD2	WPUD1	WPUD0	1111 1111	1111 1111
210h	WPUE	—	—	—	—	WPUE3	WPUE2(1)	WPUE1(1)	WPUE0(1)	---- 1111	---- 1111
211h	SSP1BUF	Synchronous Serial Port Receive Buffer/Transmit Register								XXXX XXXX	uuuu uuuu
212h	SSP1ADD	ADD<7:0>								XXXX XXXX	0000 0000
213h	SSP1MSK	MSK<7:0>								XXXX XXXX	1111 1111
214h	SSP1STAT	SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF	0000 0000	0000 0000
215h	SSP1CON1	WCOL	SSPOV	SSPEN	CKP	SSPM<3:0>				0000 0000	0000 0000
216h	SSP1CON2	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	0000 0000	0000 0000
217h	SSP1CON3	ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	0000 0000	0000 0000
218h — 21Fh	—	Unimplemented								—	—
Bank 5											
28Ch	ODCONA	ODA7	ODA6	ODA5	ODA4	ODA3	ODA2	ODA1	ODA0	0000 0000	0000 0000
28Dh	ODCONB	ODB7	ODB6	ODB5	ODB4	ODB3	ODB2	ODB1	ODB0	0000 0000	0000 0000
28Eh	ODCONC	ODC7	ODC6	ODC5	ODC4	ODC3	ODC2	ODC1	ODC0	0000 0000	0000 0000
28Fh	ODCOND(1)	ODD7	ODD6	ODD5	ODD4	ODD3	ODD2	ODD1	ODD0	0000 0000	0000 0000
290h	ODCONE(1)	—	—	—	—	—	ODE2	ODE1	ODE0	---- -000	---- -000
291h	CCPR1L	Capture/Compare/PWM Register 1 (LSB)								xxxx xxxx	uuuu uuuu
292h	CCPR1H	Capture/Compare/PWM Register 1 (MSB)								xxxx xxxx	uuuu uuuu
293h	CCP1CON	—	—	DC1B<1:0>		CCP1M<3:0>				--00 0000	--00 0000
294h — 297h	—	Unimplemented								—	—
298h	CCPR2L	Capture/Compare/PWM Register 2 (LSB)								xxxx xxxx	uuuu uuuu
299h	CCPR2H	Capture/Compare/PWM Register 2 (MSB)								xxxx xxxx	uuuu uuuu
29Ah	CCP2CON	—	—	DC2B<1:0>		CCP2M<3:0>				--00 0000	--00 0000
29Bh — 29Dh	—	Unimplemented								—	—
29Eh	CCPTMRS	P4TSEL<1:0>		P3TSEL<1:0>		C2TSEL<1:0>		C1TSEL<1:0>		0000 0000	0000 0000
29Fh	—	Unimplemented								—	—
Bank 6											
30Ch	SLRCONA	SLRA7	SLRA6	SLRA5	SLRA4	SLRA3	SLRA2	SLRA1	SLRA0	1111 1111	0000 0000
30Dh	SLRCONB	SLRB7	SLRB6	SLRB5	SLRB4	SLRB3	SLRB2	SLRB1	SLRB0	1111 1111	0000 0000
30Eh	SLRCONC	SLRC7	SLRC6	SLRC5	SLRC4	SLRC3	SLRC2	SLRC1	SLRC0	1111 1111	0000 0000
30Fh	SLRCOND(1)	SLRD7	SLRD6	SLRD5	SLRD4	SLRD3	SLRD2	SLRD1	SLRD0	1111 1111	0000 0000
310h	SLRCONE(1)	—	—	—	—	—	SLRE2	SLRE1	SLRE0	---- -111	---- -000
311h — 31Fh	—	Unimplemented								—	—

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 on PIC16(L)F1718.

2: Unimplemented on PIC16LF1717/8/9

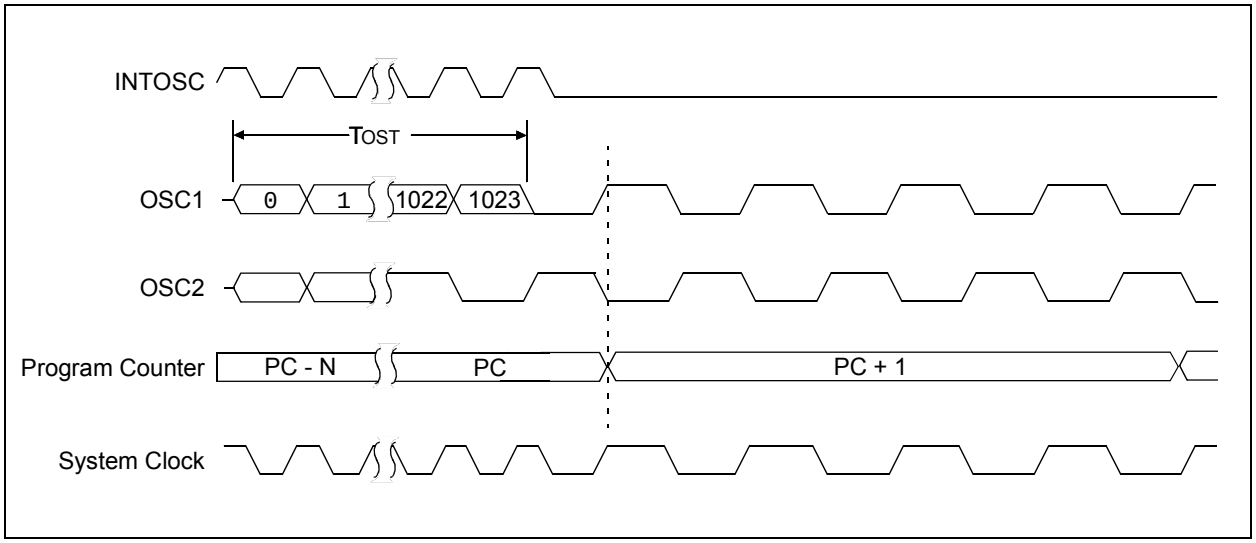
6.4.2 TWO-SPEED START-UP SEQUENCE

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.

6.4.3 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 6-8: TWO-SPEED START-UP



© 2014-2015 Microchip Technology Inc.

TABLE 7-1: SUMMARY OF REGISTERS ASSOCIATED WITH INTERRUPTS

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	90
OPTION_REG	$\overline{\text{WPUEN}}$	INTEDG	TMR0CS	TMR0SE	PSA	PS<2:0>			270
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	91
PIE2	OSFIE	C2IE	C1IE	—	BCL1IE	TMR6IE	TMR4IE	CCP2IE	92
PIE3	—	NCOIE	COGIE	ZCDIE	CLC4IE	CLC3IE	CLC2IE	CLC1IE	93
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	94
PIR2	OSFIF	C2IF	C1IF	—	BCL1IF	TMR6IF	TMR4IF	CCP2IF	95
PIR3	—	NCOIF	COGIF	ZCDIF	CLC4IF	CLC3IF	CLC2IF	CLC1IF	96

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

10.6 Register Definitions: Flash Program Memory Control

REGISTER 10-1: PMDATL: PROGRAM MEMORY DATA LOW BYTE 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
PMDAT<7:0>							
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 7-0 **PMDAT<7:0>**: Read/write value for Least Significant bits of program memory

REGISTER 10-2: PMDATH: PROGRAM MEMORY DATA HIGH BYTE REGISTER

U-0	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
—	—	PMDAT<13:8>					
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 **PMDAT<13:8>**: Read/write value for Most Significant bits of program memory

REGISTER 10-3: PMADRL: PROGRAM MEMORY ADDRESS LOW BYTE REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
PMADR<7:0>							
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 7-0 **PMADR<7:0>**: Specifies the Least Significant bits for program memory address

TABLE 11-7: SUMMARY OF REGISTERS ASSOCIATED WITH PORTE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ANSELE(1)	—	—	—	—	—	ANSE2	ANSE1	ANSE0	146
INLVLE	—	—	—	—	INLVLE3	INLVLE2(1)	INLVLE1(1)	INLVLE0(1)	148
LATE(1)	—	—	—	—	—	LATE2	LATE1	LATE0	146
ODCONE(1)	—	—	—	—	—	ODE2	ODE1	ODE0	147
PORTE	—	—	—	—	RE3	RE2(1)	RE1(1)	RE0(1)	145
SLRCONE(1)	—	—	—	—	—	SLRE2	SLRE1	SLRE0	148
TRISE	—	—	—	—	TRISE3	TRISE2(1)	TRISE1(1)	TRISE0(1)	145
WPUE	—	—	—	—	WPUE3	WPUE2(1)	WPUE1(1)	WPUE0(1)	147

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by PORTE.

Note 1: PIC16(L)F1717/9 only.

16.3 Comparator Hysteresis

A selectable amount of separation voltage can be added to the input pins of each comparator to provide a hysteresis function to the overall operation. Hysteresis is enabled by setting the CxHYS bit of the CMxCON0 register.

See Comparator Specifications in Table 34-18: Comparator Specifications for more information.

16.4 Timer1 Gate Operation

The output resulting from a comparator operation can be used as a source for gate control of Timer1. See **Section 27.6 "Timer1 Gate"** for more information. This feature is useful for timing the duration or interval of an analog event.

It is recommended that the comparator output be synchronized to Timer1. This ensures that Timer1 does not increment while a change in the comparator is occurring.

16.4.1 COMPARATOR OUTPUT SYNCHRONIZATION

The output from a comparator can be synchronized with Timer1 by setting the CxSYNC bit of the CMxCON0 register.

Once enabled, the comparator output is latched on the falling edge of the Timer1 source clock. If a prescaler is used with Timer1, the comparator output is latched after the prescaling function. To prevent a race condition, the comparator output is latched on the falling edge of the Timer1 clock source and Timer1 increments on the rising edge of its clock source. See the Comparator Block Diagram (Figure 16-2) and the Timer1 Block Diagram (Figure 27-1: Timer1 Block Diagram) for more information.

16.5 Comparator Interrupt

An interrupt can be generated upon a change in the output value of the comparator for each comparator, a rising edge detector and a falling edge detector are present.

When either edge detector is triggered and its associated enable bit is set (CxINTP and/or CxINTN bits of the CMxCON1 register), the Corresponding Interrupt Flag bit (CxIF bit of the PIR2 register) will be set.

To enable the interrupt, you must set the following bits:

- CxON, CxPOL and CxSP bits of the CMxCON0 register
- CxIE bit of the PIE2 register
- CxINTP bit of the CMxCON1 register (for a rising edge detection)
- CxINTN bit of the CMxCON1 register (for a falling edge detection)
- PEIE and GIE bits of the INTCON register

The associated interrupt flag bit, CxIF bit of the PIR2 register, must be cleared in software. If another edge is detected while this flag is being cleared, the flag will still be set at the end of the sequence.

Note:	Although a comparator is disabled, an interrupt can be generated by changing the output polarity with the CxPOL bit of the CMxCON0 register, or by switching the comparator on or off with the CxON bit of the CMxCON0 register.

28.6 CCP/PWM Clock Selection

The PIC16(L)F1717/8/9 allows each individual CCP and PWM module to select the timer source that controls the module. Each module has an independent selection.

As there are up to three 8-bit timers with auto-reload (Timer2, Timer4, and Timer6), PWM mode on the CCP and PWM modules can use any of these timers.

The CCPTMRS register is used to select which timer is used.

28.7 Register Definitions: CCP/PWM Timers Control

REGISTER 28-2: CCPTMRS: PWM TIMER SELECTION CONTROL REGISTER 0

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
P4TSEL<1:0>		P3TSEL<1:0>		C2TSEL<1:0>		C1TSEL<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 7-6 **P4TSEL<1:0>:** PWM4 Timer Selection
- 11 = Reserved
- 10 = PWM4 is based off Timer6
- 01 = PWM4 is based off Timer4
- 00 = PWM4 is based off Timer2

bit 5-4 **P3TSEL<1:0>:** PWM3 Timer Selection
- 11 = Reserved
- 10 = PWM3 is based off Timer6
- 01 = PWM3 is based off Timer4
- 00 = PWM3 is based off Timer2

bit 3-2 **C2TSEL<1:0>:** CCP2 (PWM2) Timer Selection
- 11 = Reserved
- 10 = CCP2 is based off Timer 6 in PWM mode
- 01 = CCP2 is based off Timer 4 in PWM mode
- 00 = CCP2 is based off Timer 2 in PWM mode

bit 1-0 **C1TSEL<1:0>:** CCP1 (PWM1) Timer Selection
- 11 = Reserved
- 10 = CCP1 is based off Timer6 in PWM mode
- 01 = CCP1 is based off Timer4 in PWM mode
- 00 = CCP1 is based off Timer2 in PWM mode

© 2014-2015 Microchip Technology Inc.

30.4 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.

30.4.1 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.

30.4.2 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. Table 30-2 was adapted from the Philips I²C specification.

30.4.3 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.

30.4.4 SDA HOLD TIME

The hold time of the SDA pin is selected by the SDAHT bit of the SSP1CON3 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 30-2: 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 SSP1ADD.
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.

30.4.9 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 indicates 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 SSP1CON2 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 SSP1CON2 register is set/cleared to determine the response.

Slave hardware will generate an $\overline{ACK}$ response if the AHEN and DHEN bits of the SSP1CON3 register are clear.

There are certain conditions where an $\overline{ACK}$ will not be sent by the slave. If the BF bit of the SSP1STAT register or the SSPOV bit of the SSP1CON1 register are set when a byte is received.

When the module is addressed, after the eighth falling edge of SCL on the bus, the ACKTIM bit of the SSP1CON3 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.

30.5 I^2C SLAVE MODE OPERATION

The MSSP Slave mode operates in one of four modes selected by the SSPM bits of SSP1CON1 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 operate the same as the other modes with SSP1IF additionally getting set upon detection of a Start, Restart, or Stop condition.

30.5.1 SLAVE MODE ADDRESSES

The SSP1ADD register (Register 30-6) 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 SSP1BUF 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 30-5) affects the address matching process. See **Section 30.5.9 "SSP Mask Register"** for more information.

30.5.1.1 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.

30.5.1.2 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's of the 10-bit address and stored in bits 2 and 1 of the SSP1ADD register.

After the acknowledge of the high byte the UA bit is set and SCL is held low until the user updates SSP1ADD with the low address. The low address byte is clocked in and all eight bits are compared to the low address value in SSP1ADD. Even if there is not an address match; SSP1IF and UA are set, and SCL is held low until SSP1ADD is updated to receive a high byte again. When SSP1ADD 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.

30.5.4 SLAVE MODE 10-BIT ADDRESS RECEPTION

This section describes a standard sequence of events for the MSSP module configured as an I^2C slave in 10-bit Addressing mode.

Figure 30-20 is used as a visual reference for this description.

This is a step by step process of what must be done by slave software to accomplish I^2C communication.

1. Bus starts Idle.
2. Master sends Start condition; S bit of SSP1STAT is set; SSP1IF is set if interrupt on Start detect is enabled.
3. Master sends matching high address with R/$\overline{W}$ bit clear; UA bit of the SSP1STAT register is set.
4. Slave sends $\overline{ACK}$ and SSP1IF is set.
5. Software clears the SSP1IF bit.
6. Software reads received address from SSP1BUF clearing the BF flag.
7. Slave loads low address into SSP1ADD, releasing SCL.
8. Master sends matching low address byte to the slave; UA bit is set.

Note: Updates to the SSP1ADD register are not allowed until after the $\overline{ACK}$ sequence.

9. Slave sends $\overline{ACK}$ and SSP1IF is set.

Note: If the low address does not match, SSP1IF and UA are still set so that the slave software can set SSP1ADD back to the high address. BF is not set because there is no match. CKP is unaffected.

10. Slave clears SSP1IF.
11. Slave reads the received matching address from SSP1BUF clearing BF.
12. Slave loads high address into SSP1ADD.
13. Master clocks a data byte to the slave and clocks out the slaves $\overline{ACK}$ on the 9th SCL pulse; SSP1IF is set.
14. If SEN bit of SSP1CON2 is set, CKP is cleared by hardware and the clock is stretched.
15. Slave clears SSP1IF.
16. Slave reads the received byte from SSP1BUF clearing BF.
17. If SEN is set the slave sets CKP to release the SCL.
18. Steps 13-17 repeat for each received byte.
19. Master sends Stop to end the transmission.

30.5.5 10-BIT ADDRESSING WITH ADDRESS OR DATA HOLD

Reception using 10-bit addressing with AHEN or DHEN set is the same as with 7-bit modes. The only difference is the need to update the SSP1ADD register using the UA bit. All functionality, specifically when the CKP bit is cleared and SCL line is held low are the same. Figure 30-21 can be used as a reference of a slave in 10-bit addressing with AHEN set.

Figure 30-22 shows a standard waveform for a slave transmitter in 10-bit Addressing mode.

© 2014-2015 Microchip Technology Inc.

30.7 BAUD RATE GENERATOR

The MSSP module has a Baud Rate Generator available for clock generation in both I^2C and SPI Master modes. The Baud Rate Generator (BRG) reload value is placed in the SSP1ADD register (Register 30-6). When a write occurs to SSP1BUF, the Baud Rate Generator will automatically begin counting down.

Once the given operation is complete, the internal clock will automatically stop counting and the clock pin will remain in its last state.

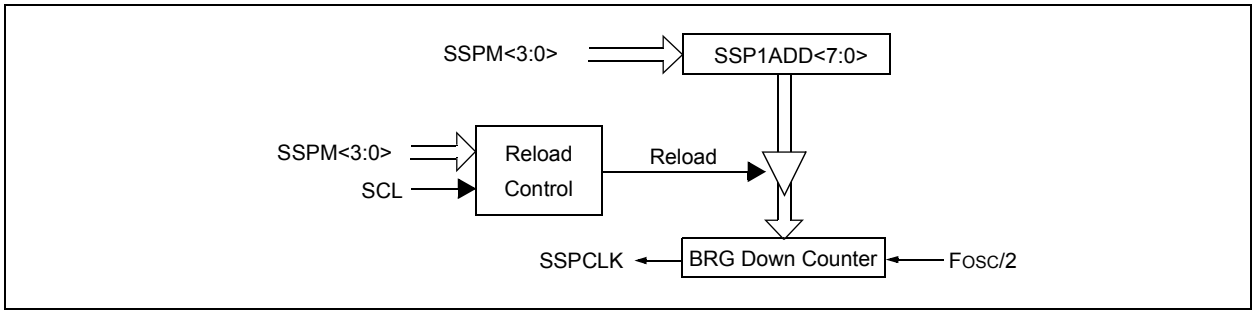
An internal signal “Reload” in Figure 30-40 triggers the value from SSP1ADD to be loaded into the BRG counter. This occurs twice for each oscillation of the module clock line. The logic dictating when the reload signal is asserted depends on the mode the MSSP is being operated in.

Table 30-4 demonstrates clock rates based on instruction cycles and the BRG value loaded into SSP1ADD.

EQUATION 30-1:

$$F_{CLOCK} = \frac{F_{OSC}}{(SSPxADD + 1)(4)}$$

FIGURE 30-40: BAUD RATE GENERATOR BLOCK DIAGRAM



Note: Values of 0x00, 0x01 and 0x02 are not valid for SSP1ADD when used as a Baud Rate Generator for I^2C. This is an implementation limitation.

TABLE 30-4: MSSP CLOCK RATE W/BRG

FOSC	FCY	BRG Value	FCLOCK (2 Rollovers of BRG)
32 MHz	8 MHz	13h	400 kHz
32 MHz	8 MHz	19h	308 kHz
32 MHz	8 MHz	4Fh	100 kHz
16 MHz	4 MHz	09h	400 kHz
16 MHz	4 MHz	0Ch	308 kHz
16 MHz	4 MHz	27h	100 kHz
4 MHz	1 MHz	09h	100 kHz

Note: Refer to the I/O port electrical and timing specifications in Table 34-10 and Figure 34-7 to ensure the system is designed to support IOL requirements.

© 2014-2015 Microchip Technology Inc.

REGISTER 30-5: SSP1MSK: SSP MASK REGISTER

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
MSK<7:0>							
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 7-1 **MSK<7:1>:** Mask bits
1 = The received address bit n is compared to SSP1ADD<n> to detect I^2C address match
0 = The received address bit n is not used to detect I^2C address match

bit 0 **MSK<0>:** Mask bit for I^2C Slave mode, 10-bit Address
I^2C Slave mode, 10-bit address (SSPM<3:0> = 0111 or 1111):
1 = The received address bit 0 is compared to SSP1ADD<0> to detect I^2C address match
0 = The received address bit 0 is not used to detect I^2C address match
I^2C Slave mode, 7-bit address, the bit is ignored

REGISTER 30-6: SSP1ADD: MSSP ADDRESS AND BAUD RATE REGISTER (I^2C MODE)

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
ADD<7:0>							
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	

Master mode:

bit 7-0 **ADD<7:0>:** Baud Rate Clock Divider bits
SCL pin clock period = ((ADD<7:0> + 1) *4)/FOSC

10-Bit Slave mode – Most Significant Address Byte:

bit 7-3 **Not used:** Unused for Most Significant Address Byte. Bit state of this register is a "don't care". Bit pattern sent by master is fixed by I^2C specification and must be equal to '11110'. However, those bits are compared by hardware and are not affected by the value in this register.

bit 2-1 **ADD<2:1>:** Two Most Significant bits of 10-bit address

bit 0 **Not used:** Unused in this mode. Bit state is a "don't care".

10-Bit Slave mode – Least Significant Address Byte:

bit 7-0 **ADD<7:0>:** Eight Least Significant bits of 10-bit address

7-Bit Slave mode:

bit 7-1 **ADD<7:1>:** 7-bit address

bit 0 **Not used:** Unused in this mode. Bit state is a "don't care".

© 2014-2015 Microchip Technology Inc.

31.4.1 AUTO-BAUD DETECT

The EUSART module supports automatic detection and calibration of the baud rate.

In the Auto-Baud Detect (ABD) mode, the clock to the BRG is reversed. Rather than the BRG clocking the incoming RX signal, the RX signal is timing the BRG. The Baud Rate Generator is used to time the period of a received 55h (ASCII "U") which is the Sync character for the LIN bus. The unique feature of this character is that it has five rising edges including the Stop bit edge.

Setting the ABDEN bit of the BAUD1CON register starts the auto-baud calibration sequence. While the ABD sequence takes place, the EUSART state machine is held in Idle. On the first rising edge of the receive line, after the Start bit, the SPBRG begins counting up using the BRG counter clock as shown in Figure 31-6. The fifth rising edge will occur on the RX pin at the end of the eighth bit period. At that time, an accumulated value totaling the proper BRG period is left in the SP1BRGH, SP1BRGL register pair, the ABDEN bit is automatically cleared and the RCIF interrupt flag is set. The value in the RC1REG needs to be read to clear the RCIF interrupt. RC1REG content should be discarded. When calibrating for modes that do not use the SP1BRGH register the user can verify that the SP1BRGL register did not overflow by checking for 00h in the SP1BRGH register.

The BRG auto-baud clock is determined by the BRG16 and BRGH bits as shown in Table 31-6. During ABD, both the SP1BRGH and SP1BRGL registers are used as a 16-bit counter, independent of the BRG16 bit setting. While calibrating the baud rate period, the SP1BRGH and SP1BRGL registers are clocked at 1/8th the BRG base clock rate. The resulting byte measurement is the average bit time when clocked at full speed.

Note 1: If the WUE bit is set with the ABDEN bit, auto-baud detection will occur on the byte following the Break character (see **Section 31.4.3 "Auto-Wake-up on Break"**).

2: It is up to the user to determine that the incoming character baud rate is within the range of the selected BRG clock source. Some combinations of oscillator frequency and EUSART baud rates are not possible.

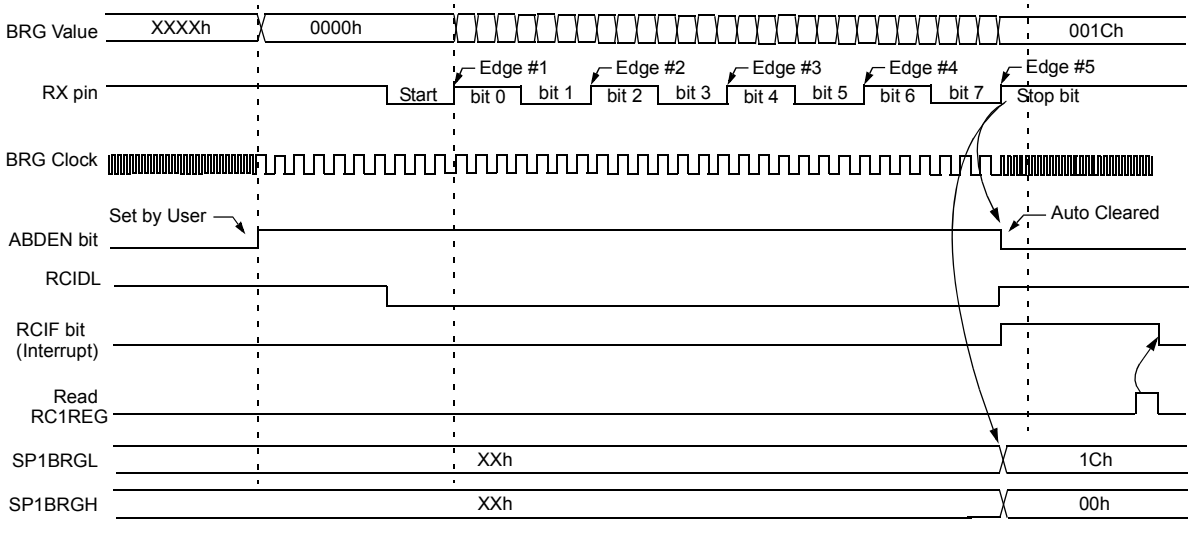
3: During the auto-baud process, the auto-baud counter starts counting at one. Upon completion of the auto-baud sequence, to achieve maximum accuracy, subtract 1 from the SP1BRGH:SP1BRGL register pair.

TABLE 31-6: BRG COUNTER CLOCK RATES

BRG16	BRGH	BRG Base Clock	BRG ABD Clock
0	0	FOSC/64	FOSC/512
0	1	FOSC/16	FOSC/128
1	0	FOSC/16	FOSC/128
1	1	FOSC/4	FOSC/32

Note: During the ABD sequence, SP1BRGL and SP1BRGH registers are both used as a 16-bit counter, independent of the BRG16 setting.

FIGURE 31-6: AUTOMATIC BAUD RATE CALIBRATION



Note 1: The ABD sequence requires the **EUSART** module to be configured in Asynchronous mode.

© 2014-2015 Microchip Technology Inc.

TABLE 34-2: SUPPLY CURRENT (I_{DD})(1,2) (CONTINUED)

PIC16LF1717/8/9		Standard Operating Conditions (unless otherwise stated)					
PIC16F1717/8/9							
Param. No.	Device Characteristics	Min.	Typ.†	Max.	Units	Conditions	
						V_{DD}	Note
D017		—	130	—	μA	1.8	F_{OSC} = 500 kHz, MFINTOSC mode
		—	150	—	μA	3.0	
D017		—	150	—	μA	2.3	F_{OSC} = 500 kHz, MFINTOSC mode (**Note 5**)
		—	170	—	μA	3.0	
		—	220	—	μA	5.0	
D019		—	0.8	—	mA	1.8	F_{OSC} = 16 MHz, HFINTOSC mode
		—	1.2	—	mA	3.0	
D019		—	1.0	—	mA	2.3	F_{OSC} = 16 MHz, HFINTOSC mode (**Note 5**)
		—	1.3	—	mA	3.0	
		—	1.4	—	mA	5.0	
D020		—	2.1	—	mA	3.0	F_{OSC} = 32 MHz, HFINTOSC mode
		—	2.5	—	mA	3.6	
D020		—	2.1	—	mA	3.0	F_{OSC} = 32 MHz, HFINTOSC mode
		—	2.2	—	mA	5.0	
D022		—	2.1	—	mA	3.0	F_{OSC} = 32 MHz, HS Oscillator mode (**Note 6**)
		—	2.5	—	mA	3.6	
D022		—	2.1	—	mA	3.0	F_{OSC} = 32 MHz HS Oscillator mode (**Note 5, Note 6**)
		—	2.2	—	mA	5.0	

† 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 test conditions for all I_{DD} measurements in active operation mode are: OSC1 = external square wave, from rail-to-rail; all I/O pins tri-stated, pulled to V_{DD}; $\overline{MCLR}$ = V_{DD}; WDT disabled.

2: The supply current is mainly a function of the operating voltage and frequency. Other factors, such as I/O pin loading and switching rate, oscillator type, internal code execution pattern and temperature, also have an impact on the current consumption.

3: For RC oscillator configurations, current through R_{EXT} is not included. The current through the resistor can be extended by the formula IR = $V_{DD}/2R_{EXT}$ (mA) with R_{EXT} in kΩ

4: FVR and BOR are disabled.

5: 0.1 μF capacitor on V_{CAP}.

6: 8 MHz clock with 4x PLL enabled.

TABLE 34-17: OPERATIONAL AMPLIFIER (OPA)

Standard Operating Conditions (unless otherwise stated) V_{DD} = 3.0V, T_A = 25°C, OPAxSP = 1 (High GBWP mode)							
Param. No.	**Symbol**	**Characteristics**	**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
OPA01*	GBWP	Gain Bandwidth Product	—	2	—	MHz	
OPA02*	TON	Turn on Time	—	10	—	μs	
OPA03*	PM	Phase Margin	—	40	—	degrees	
OPA04*	SR	Slew Rate	—	3	—	V/μs	
OPA05	OFF	Offset	—	±3	±9	mV	
OPA06	CMRR	Common Mode Rejection Ratio	55	70	—	dB	
OPA07*	AOL	Open Loop Gain	—	90	—	dB	
OPA08	VICM	Input Common Mode Voltage	0	—	VDD	V	VDD > 2.5V
OPA09*	PSRR	Power Supply Rejection Ratio	—	80	—	dB	

* These parameters are characterized but not tested.

TABLE 34-18: COMPARATOR SPECIFICATIONS

Standard Operating Conditions (unless otherwise stated) V_{DD} = 3.0V, T_A = 25°C See **Section 35.0 "DC and AC Characteristics Graphs and Charts"** for operating characterization.							
Param. No.	**Sym.**	**Characteristics**	**Min.**	**Typ.**	**Max.**	**Units**	**Comments**
CM01	VIOFF	Input Offset Voltage	—	±2.5	±5	mV	CxSP = 1, VICM = VDD/2
CM02	VICM	Input Common Mode Voltage	0	—	VDD	V	
CM03	CMRR	Common Mode Rejection Ratio	40	50	—	dB	
CM04A	TRESP(1)	Response Time Rising Edge	—	60	85	ns	CxSP = 1
CM04B		Response Time Falling Edge	—	60	90	ns	CxSP = 1
CM04C		Response Time Rising Edge	—	85	—	ns	CxSP = 0
CM04D		Response Time Falling Edge	—	85	—	ns	CxSP = 0
CM05*	TMC2OV	Comparator Mode Change to Output Valid*	—	—	10	μs	
CM06	CHYSTER	Comparator Hysteresis	20	45	75	mV	CxHYS = 1, CxSP = 1

* These parameters are characterized but not tested.

Note 1: Response time measured with one comparator input at VDD/2, while the other input transitions from VSS to VDD.

Note: Unless otherwise noted, VIN = 5V, FOSC = 500 kHz, CIN = 0.1 µF, TA = 25°C.

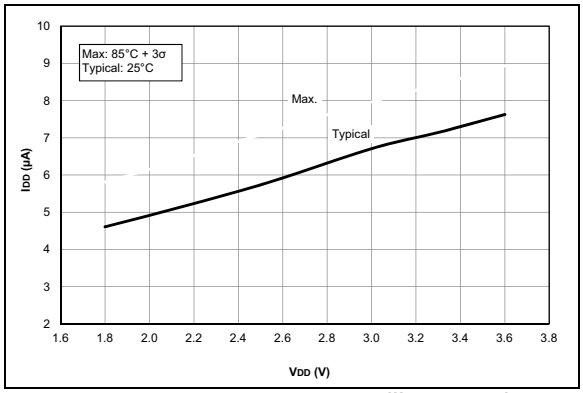


FIGURE 35-1: *IDD, LP Oscillator Mode, Fosc = 32 kHz, PIC16LF1717/8/9 Only.*

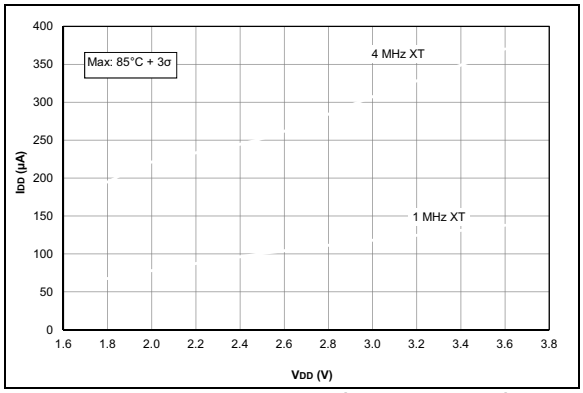


FIGURE 35-4: *IDD Maximum, XT and EXTRC Oscillator, PIC16LF1717/8/9 Only.*

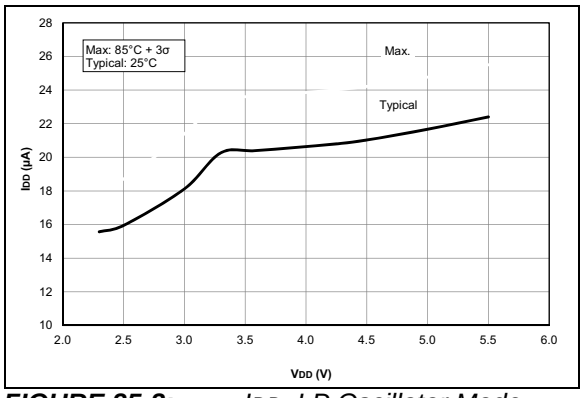


FIGURE 35-2: *IDD, LP Oscillator Mode, Fosc = 32 kHz, PIC16F1717/8/9 Only.*

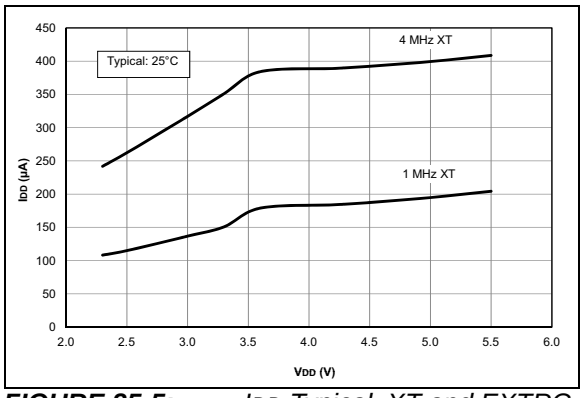


FIGURE 35-5: *IDD Typical, XT and EXTRC Oscillator, PIC16F1717/8/9 Only.*

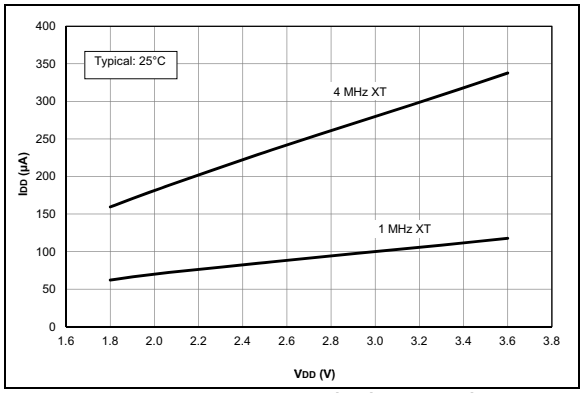


FIGURE 35-3: *IDD Typical, XT and EXTRC Oscillator, PIC16LF1717/8/9 Only.*

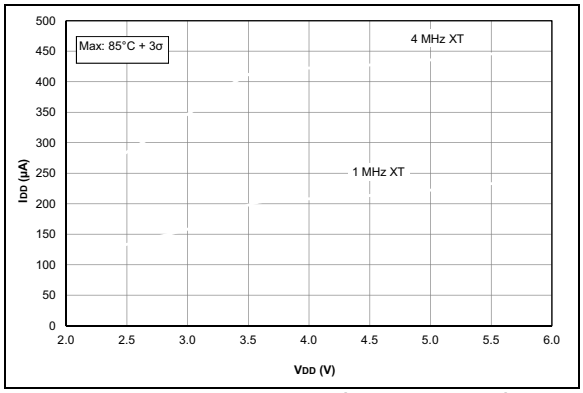


FIGURE 35-6: *IDD Maximum, XT and EXTRC Oscillator, PIC16F1717/8/9 Only.*

36.0 DEVELOPMENT SUPPORT

The PIC microcontrollers (MCU) and dsPIC® digital signal controllers (DSC) are supported with a full range of software and hardware development tools:

- Integrated Development Environment
  - MPLAB® X IDE Software
- Compilers/Assemblers/Linkers
  - MPLAB XC Compiler
  - MPASM™ Assembler
  - MPLINK™ Object Linker/ MPLIB™ Object Librarian
  - MPLAB Assembler/Linker/Librarian for Various Device Families
- Simulators
  - MPLAB X SIM Software Simulator
- Emulators
  - MPLAB REAL ICE™ In-Circuit Emulator
- In-Circuit Debuggers/Programmers
  - MPLAB ICD 3
  - PICkit 3
- Device Programmers
  - MPLAB PM3 Device Programmer
- Low-Cost Demonstration/Development Boards, Evaluation Kits and Starter Kits
- Third-party development tools

36.1 MPLAB X Integrated Development Environment Software

The MPLAB X IDE is a single, unified graphical user interface for Microchip and third-party software, and hardware development tool that runs on Windows®, Linux and Mac OS® X. Based on the NetBeans IDE, MPLAB X IDE is an entirely new IDE with a host of free software components and plug-ins for high-performance application development and debugging. Moving between tools and upgrading from software simulators to hardware debugging and programming tools is simple with the seamless user interface.

With complete project management, visual call graphs, a configurable watch window and a feature-rich editor that includes code completion and context menus, MPLAB X IDE is flexible and friendly enough for new users. With the ability to support multiple tools on multiple projects with simultaneous debugging, MPLAB X IDE is also suitable for the needs of experienced users.

Feature-Rich Editor:

- Color syntax highlighting
- Smart code completion makes suggestions and provides hints as you type
- Automatic code formatting based on user-defined rules
- Live parsing

User-Friendly, Customizable Interface:

- Fully customizable interface: toolbars, toolbar buttons, windows, window placement, etc.
- Call graph window

Project-Based Workspaces:

- Multiple projects
- Multiple tools
- Multiple configurations
- Simultaneous debugging sessions

File History and Bug Tracking:

- Local file history feature
- Built-in support for Bugzilla issue tracker

© 2014-2015 Microchip Technology Inc.

THE MICROCHIP WEBSITE

Microchip provides online support via our website at www.microchip.com. This website is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the website contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip website at www.microchip.com. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support

Customers should contact their distributor, representative or Field Application Engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the website at: http://www.microchip.com/support

© 2014-2015 Microchip Technology Inc.