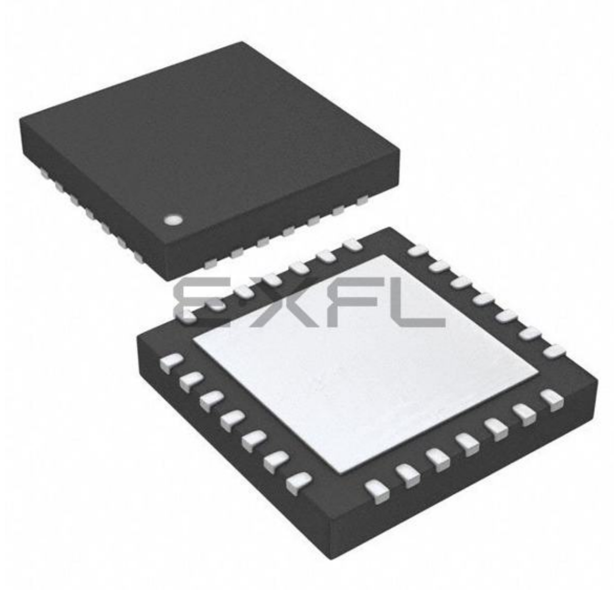


Welcome to E-XFL.COM

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	Obsolete
Core Processor	PIC
Core Size	8-Bit
Speed	10MHz
Connectivity	I²C, SPI
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	16
Program Memory Size	3.5KB (2K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.5V
Data Converters	A/D 5x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-VQFN Exposed Pad
Supplier Device Package	28-QFN (6x6)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf819t-i-mltsl

TABLE 2-1: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Details on page:
Bank 2											
100h(1)	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	23
101h	TMR0	Timer0 Module Register								xxxx xxxx	53
102h(1	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	23
103h(1)	STATUS	IRP	RP1	RP0	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C	0001 1xxx	16
104h(1)	FSR	Indirect Data Memory Address Pointer								xxxx xxxx	23
105h	—	Unimplemented								—	—
106h	PORTB	PORTB Data Latch when written; PORTB pins when read								xxxx xxxx	43
107h	—	Unimplemented								—	—
108h	—	Unimplemented								—	—
109h	—	Unimplemented								—	—
10Ah(1,2)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	23
10Bh(1)	INTCON	GIE	PEIE	TMR0IE	INTE	RBIE	TMR0IF	INTF	RBIF	0000 000x	18
10Ch	EEDATA	EEPROM/Flash Data Register Low Byte								xxxx xxxx	25
10Dh	EEADR	EEPROM/Flash Address Register Low Byte								xxxx xxxx	25
10Eh	EEDATH	—	—	EEPROM/Flash Data Register High Byte						--xx xxxx	25
10Fh	EEADRH	—	—	—	—	—	EEPROM/Flash Address Register High Byte			---- -xxx	25
Bank 3											
180h(1)	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	23
181h	OPTION_REG	$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	17, 54
182h(1)	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	23
183h(1)	STATUS	IRP	RP1	RP0	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C	0001 1xxx	16
184h(1)	FSR	Indirect Data Memory Address Pointer								xxxx xxxx	23
185h	—	Unimplemented								—	—
186h	TRISB	PORTB Data Direction Register								1111 1111	43
187h	—	Unimplemented								—	—
188h	—	Unimplemented								—	—
189h	—	Unimplemented								—	—
18Ah(1,2)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	23
18Bh(1)	INTCON	GIE	PEIE	TMR0IE	INTE	RBIE	TMR0IF	INTF	RBIF	0000 000x	18
18Ch	EECON1	EEPGD	—	—	FREE	WRERR	WREN	WR	RD	x--x x000	26
18Dh	EECON2	EEPROM Control Register 2 (not a physical register)								---- ----	25
18Eh	—	Reserved; maintain clear								0000 0000	—
18Fh	—	Reserved; maintain clear								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: These registers can be addressed from any bank.

2: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<12:8>, whose contents are transferred to the upper byte of the program counter.

3: Pin 5 is an input only; the state of the TRISA5 bit has no effect and will always read '1'.

2.2.2.3 INTCON Register

The INTCON register is a readable and writable register that contains various enable and flag bits for the TMR0 register overflow, RB port change and external RB0/INT pin interrupts.

Note: Interrupt flag bits get set when an interrupt condition occurs regardless of the state of its corresponding enable bit or the Global Interrupt Enable bit, GIE (INTCON<7>). User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 2-3: INTCON: INTERRUPT CONTROL REGISTER (ADDRESS 0Bh, 8Bh, 10Bh, 18Bh)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-x
GIE	PEIE	TMR0IE	INTE	RBIE	TMR0IF	INTF	RBIF
bit 7							bit 0

bit 7 **GIE:** Global Interrupt Enable bit
1 = Enables all unmasked interrupts
0 = Disables all interrupts

bit 6 **PEIE:** Peripheral Interrupt Enable bit
1 = Enables all unmasked peripheral interrupts
0 = Disables all peripheral interrupts

bit 5 **TMR0IE:** TMR0 Overflow Interrupt Enable bit
1 = Enables the TMR0 interrupt
0 = Disables the TMR0 interrupt

bit 4 **INTE:** RB0/INT External Interrupt Enable bit
1 = Enables the RB0/INT external interrupt
0 = Disables the RB0/INT external interrupt

bit 3 **RBIE:** RB Port Change Interrupt Enable bit
1 = Enables the RB port change interrupt
0 = Disables the RB port change interrupt

bit 2 **TMR0IF:** TMR0 Overflow Interrupt Flag bit
1 = TMR0 register has overflowed (must be cleared in software)
0 = TMR0 register did not overflow

bit 1 **INTF:** RB0/INT External Interrupt Flag bit
1 = The RB0/INT external interrupt occurred (must be cleared in software)
0 = The RB0/INT external interrupt did not occur

bit 0 **RBIF:** RB Port Change Interrupt Flag bit
A mismatch condition will continue to set flag bit RBIF. Reading PORTB will end the mismatch condition and allow flag bit RBIF to be cleared.
1 = At least one of the RB7:RB4 pins changed state (must be cleared in software)
0 = None of the RB7:RB4 pins have changed state

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

REGISTER 3-1: EECON1: EEPROM ACCESS CONTROL REGISTER 1 (ADDRESS 18Ch)

R/W-x	U-0	U-0	R/W-x	R/W-x	R/W-0	R/S-0	R/S-0
EEPGD	—	—	FREE	WRERR	WREN	WR	RD
bit 7							bit 0

bit 7 **EEPGD**: Program/Data EEPROM Select bit
1 = Accesses program memory
0 = Accesses data memory
Reads '0' after a POR; this bit cannot be changed while a write operation is in progress.

bit 6-5 **Unimplemented:** Read as '0'

bit 4 **FREE:** EEPROM Forced Row Erase bit
1 = Erase the program memory row addressed by EEADRH:EEADR on the next WR command
0 = Perform write-only

bit 3 **WRERR:** EEPROM Error Flag bit
1 = A write operation is prematurely terminated (any $\overline{\text{MCLR}}$ or any WDT Reset during normal operation)
0 = The write operation completed

bit 2 **WREN:** EEPROM Write Enable bit
1 = Allows write cycles
0 = Inhibits write to the EEPROM

bit 1 **WR:** Write Control bit
1 = Initiates a write cycle. The bit is cleared by hardware once write is complete. The WR bit can only be set (not cleared) in software.
0 = Write cycle to the EEPROM is complete

bit 0 **RD:** Read Control bit
1 = Initiates an EEPROM read, RD is cleared in hardware. The RD bit can only be set (not cleared) in software.
0 = Does not initiate an EEPROM read

Legend:			
R = Readable bit	W = Writable bit	S = Set only	U = Unimplemented bit, read as '0'
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

4.0 OSCILLATOR CONFIGURATIONS

4.1 Oscillator Types

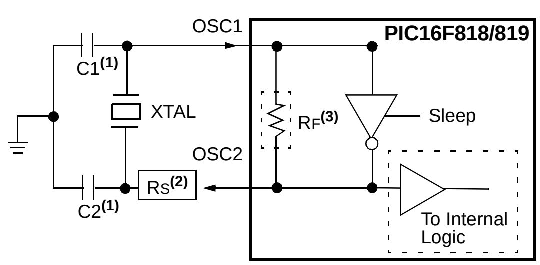
The PIC16F818/819 can be operated in eight different oscillator modes. The user can program three configuration bits (FOSC2:FOSC0) to select one of these eight modes (modes 5-8 are new PIC16 oscillator configurations):

1. LP Low-Power Crystal
2. XT Crystal/Resonator
3. HS High-Speed Crystal/Resonator
4. RC External Resistor/Capacitor with FOSC/4 output on RA6
5. RCIO External Resistor/Capacitor with I/O on RA6
6. INTIO1 Internal Oscillator with FOSC/4 output on RA6 and I/O on RA7
7. INTIO2 Internal Oscillator with I/O on RA6 and RA7
8. ECIO External Clock with I/O on RA6

4.2 Crystal Oscillator/Ceramic Resonators

In XT, LP or HS modes, a crystal or ceramic resonator is connected to the OSC1/CLKI and OSC2/CLKO pins to establish oscillation (see Figure 4-1 and Figure 4-2). The PIC16F818/819 oscillator design requires the use of a parallel cut crystal. Use of a series cut crystal may give a frequency out of the crystal manufacturer's specifications.

FIGURE 4-1: CRYSTAL OPERATION (HS, XT OR LP OSC CONFIGURATION)



Note 1: See Table 4-1 for typical values of C1 and C2.
2: A series resistor (RS) may be required for AT strip cut crystals.
3: RF varies with the crystal chosen (typically between 2 MΩ to 10 MΩ).

TABLE 4-1: CAPACITOR SELECTION FOR CRYSTAL OSCILLATOR (FOR DESIGN GUIDANCE ONLY)

Osc Type	Crystal Freq	Typical Capacitor Values Tested:	
		C1	C2
LP	32 kHz	33 pF	33 pF
	200 kHz	15 pF	15 pF
XT	200 kHz	56 pF	56 pF
	1 MHz	15 pF	15 pF
	4 MHz	15 pF	15 pF
HS	4 MHz	15 pF	15 pF
	8 MHz	15 pF	15 pF
	20 MHz	15 pF	15 pF

Capacitor values are for design guidance only.

These capacitors were tested with the crystals listed below for basic start-up and operation. These values were not optimized.

Different capacitor values may be required to produce acceptable oscillator operation. The user should test the performance of the oscillator over the expected VDD and temperature range for the application.

See the notes following this table for additional information.

Note 1: Higher capacitance increases the stability of the oscillator but also increases the start-up time.

2: Since each crystal has its own characteristics, the user should consult the crystal manufacturer for appropriate values of external components.

3: RS may be required in HS mode, as well as XT mode, to avoid overdriving crystals with low drive level specification.

4: Always verify oscillator performance over the VDD and temperature range that is expected for the application.

4.5.3 OSCILLATOR CONTROL REGISTER

The OSCCON register (Register 4-2) controls several aspects of the system clock's operation.

The Internal Oscillator Select bits, IRCF2:IRCF0, select the frequency output of the internal oscillator block that is used to drive the system clock. The choices are the INTRC source (31.25 kHz), the INTOSC source (8 MHz) or one of the six frequencies derived from the INTOSC postscaler (125 kHz to 4 MHz). Changing the configuration of these bits has an immediate change on the multiplexor's frequency output.

4.5.4 MODIFYING THE IRCF BITS

The IRCF bits can be modified at any time regardless of which clock source is currently being used as the system clock. The internal oscillator allows users to change the frequency during run time. This is achieved by modifying the IRCF bits in the OSCCON register. The sequence of events that occur after the IRCF bits are modified is dependent upon the initial value of the IRCF bits before they are modified. If the INTRC (31.25 kHz, IRCF<2:0> = `000`) is running and the IRCF bits are modified to any other value than '`000`', a 4 ms (approx.) clock switch delay is turned on. Code execution continues at a higher than expected frequency while the new frequency stabilizes. Time sensitive code should wait for the IOFS bit in the OSCCON register to become set before continuing. This bit can be monitored to ensure that the frequency is stable before using the system clock in time critical applications.

If the IRCF bits are modified while the internal oscillator is running at any other frequency than INTRC (31.25 kHz, IRCF<2:0> ≠ `000`), there is no need for a 4 ms (approx.) clock switch delay. The new INTOSC frequency will be stable immediately after the **eight** falling edges. The IOFS bit will remain set after clock switching occurs.

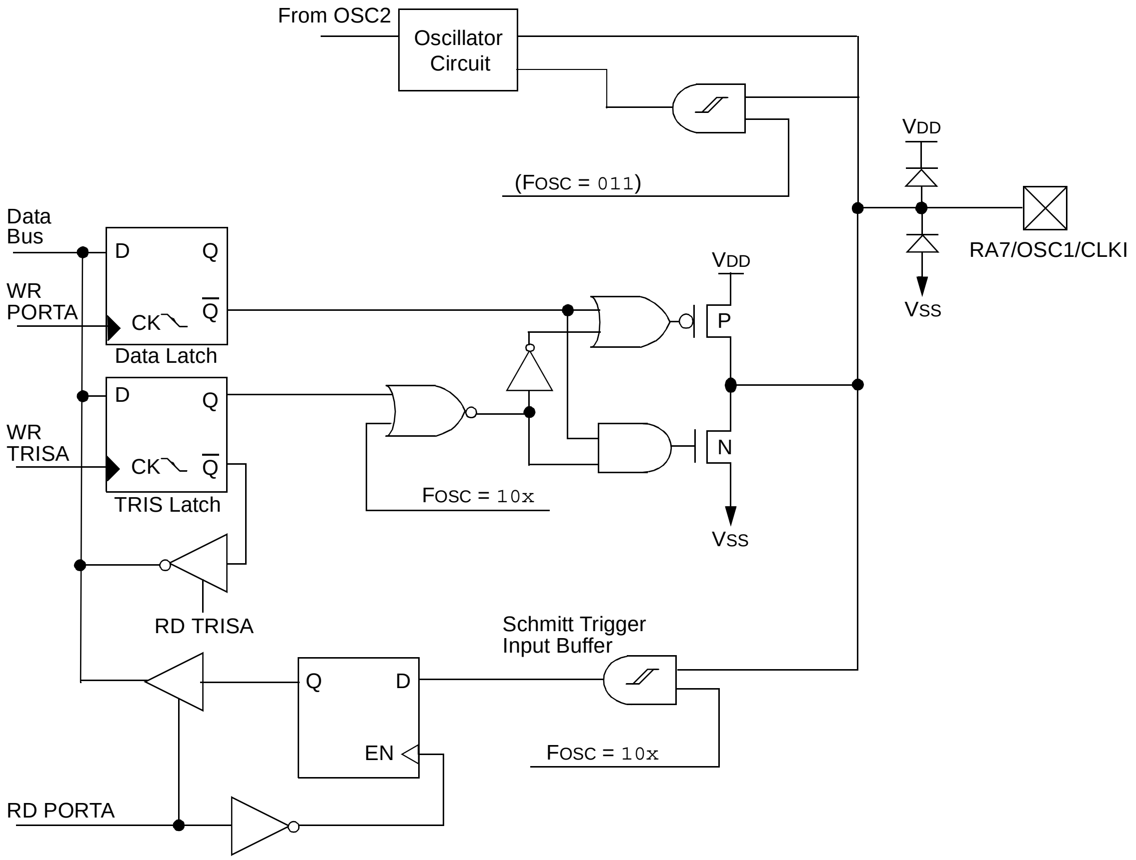
Note:	Caution must be taken when modifying the IRCF bits using `BCF` or `BSF` instructions. It is possible to modify the IRCF bits to a frequency that may be out of the V_{DD} specification range; for example, V_{DD} = 2.0V and IRCF = `111` (8 MHz).

4.5.5 CLOCK TRANSITION SEQUENCE WHEN THE IRCF BITS ARE MODIFIED

Following are three different sequences for switching the internal RC oscillator frequency.

- Clock before switch: 31.25 kHz (IRCF<2:0> = `000`)
  1. IRCF bits are modified to an INTOSC/INTOSC postscaler frequency.
  2. The clock switching circuitry waits for a falling edge of the current clock, at which point CLKO is held low.
  3. The clock switching circuitry then waits for eight falling edges of requested clock, after which it switches CLKO to this new clock source.
  4. The IOFS bit is clear to indicate that the clock is unstable and a 4 ms (approx.) delay is started. Time dependent code should wait for IOFS to become set.
  5. Switchover is complete.
- Clock before switch: One of INTOSC/INTOSC postscaler (IRCF<2:0> ≠ `000`)
  1. IRCF bits are modified to INTRC (IRCF<2:0> = `000`).
  2. The clock switching circuitry waits for a falling edge of the current clock, at which point CLKO is held low.
  3. The clock switching circuitry then waits for eight falling edges of requested clock, after which it switches CLKO to this new clock source.
  4. Oscillator switchover is complete.
- Clock before switch: One of INTOSC/INTOSC postscaler (IRCF<2:0> ≠ `000`)
  1. IRCF bits are modified to a different INTOSC/INTOSC postscaler frequency.
  2. The clock switching circuitry waits for a falling edge of the current clock, at which point CLKO is held low.
  3. The clock switching circuitry then waits for eight falling edges of requested clock, after which it switches CLKO to this new clock source.
  4. The IOFS bit is set.
  5. Oscillator switchover is complete.

FIGURE 5-7: BLOCK DIAGRAM OF RA7/OSC1/CLKI PIN



Note 1: I/O pins have protection diodes to VDD and VSS.

The maximum PWM resolution (bits) for a given PWM frequency is given by the following formula.

EQUATION 9-3:

$$\text{Resolution} = \frac{\log\left(\frac{F_{OSC}}{F_{PWM}}\right)}{\log(2)} \text{ bits}$$

Note:	If the PWM duty cycle value is longer than the PWM period, the CCP1 pin will not be cleared.

9.3.3 SETUP FOR PWM OPERATION

The following steps should be taken when configuring the CCP module for PWM operation:

1. Set the PWM period by writing to the PR2 register.
2. Set the PWM duty cycle by writing to the CCPR1L register and CCP1CON<5:4> bits.
3. Make the CCP1 pin an output by clearing the TRISB<x> bit.
4. Set the TMR2 prescale value and enable Timer2 by writing to T2CON.
5. Configure the CCP1 module for PWM operation.

Note:	The TRISB bit (2 or 3) is dependant upon the setting of configuration bit 12 (CCPMX).

TABLE 9-3: EXAMPLE PWM FREQUENCIES AND RESOLUTIONS AT 20 MHz

PWM Frequency	1.22 kHz	4.88 kHz	19.53 kHz	78.12 kHz	156.3 kHz	208.3 kHz
Timer Prescaler (1, 4, 16)	16	4	1	1	1	1
PR2 Value	0xFF	0xFF	0xFF	0x3F	0x1F	0x17
Maximum Resolution (bits)	10	10	10	8	7	5.5

TABLE 9-4: REGISTERS ASSOCIATED WITH PWM AND TIMER2

Address	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
0Bh,8Bh 10Bh,18Bh	INTCON	GIE	PEIE	TMR0IE	INTE	RBIE	TMR0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF	—	—	SSPIF	CCP1IF	TMR2IF	TMR1IF	-0-- 0000	-0-- 0000
8Ch	PIE1	—	ADIE	—	—	SSPIE	CCP1IE	TMR2IE	TMR1IE	-0-- 0000	-0-- 0000
86h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
11h	TMR2	Timer2 Module Register								0000 0000	0000 0000
92h	PR2	Timer2 Module Period Register								1111 1111	1111 1111
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000
15h	CCPR1L	Capture/Compare/PWM Register 1 (LSB)								xxxx xxxx	uuuu uuuu
16h	CCPR1H	Capture/Compare/PWM Register 1 (MSB)								xxxx xxxx	uuuu uuuu
17h	CCP1CON	—	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	--00 0000

Legend: x = unknown, u = unchanged, - = unimplemented, read as '0'. Shaded cells are not used by PWM and Timer2.

REGISTER 10-2: SSPCON: SYNCHRONOUS SERIAL PORT CONTROL REGISTER 1 (ADDRESS 14h)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0
bit 7							bit 0

bit 7 **WCOL:** Write Collision Detect bit

1 = An attempt to write the SSPBUF register failed because the SSP module is busy (must be cleared in software)
0 = No collision

bit 6 **SSPOV:** Receive Overflow Indicator bit

In SPI mode:
1 = A new byte is received while the SSPBUF register is still holding the previous data. In case of overflow, the data in SSPSR is lost. Overflow can only occur in Slave mode. The user must read the SSPBUF, even if only transmitting data, to avoid setting overflow. In Master mode, the overflow bit is not set since each new reception (and transmission) is initiated by writing to the SSPBUF register.
0 = No overflow

In I^2C mode:
1 = A byte is received while the SSPBUF register is still holding the previous byte. SSPOV is a "don't care" in Transmit mode. SSPOV must be cleared in software in either mode.
0 = No overflow

bit 5 **SSPEN:** Synchronous Serial Port Enable bit[(1)]

In SPI mode:
1 = Enables serial port and configures SCK, SDO and SDI as serial port pins
0 = Disables serial port and configures these pins as I/O port pins

In I^2C mode:
1 = Enables the serial port and configures the SDA and SCL pins as serial port pins
0 = Disables serial port and configures these pins as I/O port pins

Note 1: In both modes, when enabled, these pins must be properly configured as input or output.

bit 4 **CKP:** Clock Polarity Select bit

In SPI mode:
1 = Transmit happens on falling edge, receive on rising edge. Idle state for clock is a high level.
0 = Transmit happens on rising edge, receive on falling edge. Idle state for clock is a low level.

In I^2C Slave mode:
SCK release control.
1 = Enable clock
0 = Holds clock low (clock stretch). (Used to ensure data setup time.)

bit 3-0 **SSPM<3:0>:** Synchronous Serial Port Mode Select bits

0000 = SPI Master mode, clock = OSC/4
0001 = SPI Master mode, clock = OSC/16
0010 = SPI Master mode, clock = OSC/64
0011 = SPI Master mode, clock = TMR2 output/2
0100 = SPI Slave mode, clock = SCK pin. $\overline{SS}$ pin control enabled.
0101 = SPI Slave mode, clock = SCK pin. $\overline{SS}$ pin control disabled. $\overline{SS}$ can be used as I/O pin.
0110 = I^2C Slave mode, 7-bit address
0111 = I^2C Slave mode, 10-bit address
1011 = I^2C Firmware Controlled Master mode (Slave Idle)
1110 = I^2C Slave mode, 7-bit address with Start and Stop bit interrupts enabled
1111 = I^2C Slave mode, 10-bit address with Start and Stop bit interrupts enabled
1000, 1001, 1010, 1100, 1101 = **Reserved**

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

10.3.1 SLAVE MODE

In Slave mode, the SCL and SDA pins must be configured as inputs (TRISB<4,1> set). The SSP module will override the input state with the output data when required (slave-transmitter).

When an address is matched, or the data transfer after an address match is received, the hardware automatically will generate the Acknowledge ($\overline{\text{ACK}}$) pulse and then load the SSPBUF register with the received value currently in the SSPSR register.

Either or both of the following conditions will cause the SSP module not to give this $\overline{\text{ACK}}$ pulse:

a) The Buffer Full bit, BF (SSPSTAT<0>), was set before the transfer was received.
b) The overflow bit, SSPOV (SSPCON<6>), was set before the transfer was received.

In this case, the SSPSR register value is not loaded into the SSPBUF but bit, SSPIF (PIR1<3>), is set. Table 10-2 shows what happens when a data transfer byte is received, given the status of bits BF and SSPOV. The shaded cells show the condition where user software did not properly clear the overflow condition. Flag bit BF is cleared by reading the SSPBUF register while bit SSPOV is cleared through software.

The SCL clock input must have a minimum high and low for proper operation. The high and low times of the I^{2}C specification, as well as the requirement of the SSP module, are shown in timing parameter #100 and parameter #101.

10.3.1.1 Addressing

Once the SSP module has been enabled, it waits for a Start condition to occur. Following the Start condition, the eight bits are shifted into the SSPSR register. All incoming bits are sampled with the rising edge of the clock (SCL) line. The value of register SSPSR<7:1> is compared to the value of the SSPADD register. The address is compared on the falling edge of the eighth clock (SCL) pulse. If the addresses match and the BF and SSPOV bits are clear, the following events occur:

a) The SSPSR register value is loaded into the SSPBUF register.
b) The Buffer Full bit, BF, is set.
c) An $\overline{\text{ACK}}$ pulse is generated.
d) SSP Interrupt Flag bit, SSPIF (PIR1<3>), is set (interrupt is generated if enabled) – on the falling edge of the ninth SCL pulse.

In 10-bit Address mode, two address bytes need to be received by the slave device. The five Most Significant bits (MSbs) of the first address byte specify if this is a 10-bit address. Bit R/$\overline{\text{W}}$ (SSPSTAT<2>) must specify a write so the slave device will receive the second address byte. For a 10-bit address, the first byte would equal '`1111 0 A9 A8 0`', where `A9` and `A8` are the two MSbs of the address.

The sequence of events for 10-bit address is as follows, with steps 7-9 for slave-transmitter:

1. Receive first (high) byte of address (bits SSPIF, BF and bit UA (SSPSTAT<1>) are set).
2. Update the SSPADD register with second (low) byte of address (clears bit UA and releases the SCL line).
3. Read the SSPBUF register (clears bit BF) and clear flag bit, SSPIF.
4. Receive second (low) byte of address (bits SSPIF, BF and UA are set).
5. Update the SSPADD register with the first (high) byte of address; if match releases SCL line, this will clear bit UA.
6. Read the SSPBUF register (clears bit BF) and clear flag bit, SSPIF.
7. Receive Repeated Start condition.
8. Receive first (high) byte of address (bits SSPIF and BF are set).
9. Read the SSPBUF register (clears bit BF) and clear flag bit, SSPIF.

10.3.1.2 Reception

When the R/$\overline{\text{W}}$ bit of the address byte is clear and an address match occurs, the R/$\overline{\text{W}}$ bit of the SSPSTAT register is cleared. The received address is loaded into the SSPBUF register.

When the address byte overflow condition exists, then a no Acknowledge ($\overline{\text{ACK}}$) pulse is given. An overflow condition is indicated if either bit, BF (SSPSTAT<0>), is set or bit, SSPOV (SSPCON<6>), is set.

An SSP interrupt is generated for each data transfer byte. Flag bit, SSPIF (PIR1<3>), must be cleared in software. The SSPSTAT register is used to determine the status of the byte.

10.3.1.3 Transmission

When the R/$\overline{\text{W}}$ bit of the incoming address byte is set and an address match occurs, the R/$\overline{\text{W}}$ bit of the SSPSTAT register is set. The received address is loaded into the SSPBUF register. The $\overline{\text{ACK}}$ pulse will be sent on the ninth bit and pin RB4/SCK/SCL is held low. The transmit data must be loaded into the SSPBUF register which also loads the SSPSR register. Then pin RB4/SCK/SCL should be enabled by setting bit, CKP (SSPCON<4>). The master device must monitor the SCL pin prior to asserting another clock pulse. The slave devices may be holding off the master device by stretching the clock. The eight data bits are shifted out on the falling edge of the SCL input. This ensures that the SDA signal is valid during the SCL high time (Figure 10-7).

NOTES:

11.0 ANALOG-TO-DIGITAL CONVERTER (A/D) MODULE

The Analog-to-Digital (A/D) converter module has five inputs for 18/20 pin devices.

The conversion of an analog input signal results in a corresponding 10-bit digital number. The A/D module has a high and low-voltage reference input that is software selectable to some combination of V_{DD}, V_{SS}, RA2 or RA3.

The A/D converter has a unique feature of being able to operate while the device is in Sleep mode. To operate in Sleep, the A/D conversion clock must be derived from the A/D's internal RC oscillator.

The A/D module has four registers:

- A/D Result High Register (ADRESH)
- A/D Result Low Register (ADRESL)
- A/D Control Register 0 (ADCON0)
- A/D Control Register 1 (ADCON1)

The ADCON0 register, shown in Register 11-1, controls the operation of the A/D module. The ADCON1 register, shown in Register 11-2, configures the functions of the port pins. The port pins can be configured as analog inputs (RA3 can also be a voltage reference) or as digital I/Os.

Additional information on using the A/D module can be found in the *"PIC® Mid-Range MCU Family Reference Manual"* (DS33023).

REGISTER 11-1: ADCON0: A/D CONTROL REGISTER 0 (ADDRESS 1Fh)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0
ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/$\overline{\text{DONE}}$	—	ADON
bit 7							bit 0

bit 7-6 **ADCS1:ADCS0:** A/D Conversion Clock Select bits

If ADCS2 = `0`:
`00` = FOSC/2
`01` = FOSC/8
`10` = FOSC/32
`11` = FRC (clock derived from the internal A/D module RC oscillator)

If ADCS2 = `1`:
`00` = FOSC/4
`01` = FOSC/16
`10` = FOSC/64
`11` = FRC (clock derived from the internal A/D module RC oscillator)

bit 5-3 **CHS2:CHS0:** Analog Channel Select bits
`000` = Channel 0 (RA0/AN0)
`001` = Channel 1 (RA1/AN1)
`010` = Channel 2 (RA2/AN2)
`011` = Channel 3 (RA3/AN3)
`100` = Channel 4 (RA4/AN4)

bit 2 **GO/$\overline{\text{DONE}}$:** A/D Conversion Status bit

If ADON = `1`:
`1` = A/D conversion in progress (setting this bit starts the A/D conversion)
`0` = A/D conversion not in progress (this bit is automatically cleared by hardware when the A/D conversion is complete)

bit 1 **Unimplemented:** Read as '`0`'

bit 0 **ADON:** A/D On bit
`1` = A/D converter module is operating
`0` = A/D converter module is shut-off and consumes no operating current

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

11.5 A/D Operation During Sleep

The A/D module can operate during Sleep mode. This requires that the A/D clock source be set to RC (ADCS1:ADCS0 = 11). When the RC clock source is selected, the A/D module waits one instruction cycle before starting the conversion. This allows the SLEEP instruction to be executed which eliminates all digital switching noise from the conversion. When the conversion is completed, the GO/$\overline{\text{DONE}}$ bit will be cleared and the result loaded into the ADRES register. If the A/D interrupt is enabled, the device will wake-up from Sleep. If the A/D interrupt is not enabled, the A/D module will then be turned off, although the ADON bit will remain set.

When the A/D clock source is another clock option (not RC), a SLEEP instruction will cause the present conversion to be aborted and the A/D module to be turned off, though the ADON bit will remain set.

Turning off the A/D places the A/D module in its lowest current consumption state.

Note:	For the A/D module to operate in Sleep, the A/D clock source must be set to RC (ADCS1:ADCS0 = 11). To perform an A/D conversion in Sleep, ensure the SLEEP instruction immediately follows the instruction that sets the GO/$\overline{\text{DONE}}$ bit.

11.6 Effects of a Reset

A device Reset forces all registers to their Reset state. The A/D module is disabled and any conversion in progress is aborted. All A/D input pins are configured as analog inputs.

The value that is in the ADRESH:ADRESL registers is not modified for a Power-on Reset. The ADRESH:ADRESL registers will contain unknown data after a Power-on Reset.

11.7 Use of the CCP Trigger

An A/D conversion can be started by the “special event trigger” of the CCP module. This requires that the CCP1M3:CCP1M0 bits (CCP1CON<3:0>) be programmed as ‘1011’ and that the A/D module is enabled (ADON bit is set). When the trigger occurs, the GO/$\overline{\text{DONE}}$ bit will be set, starting the A/D conversion and the Timer1 counter will be reset to zero. Timer1 is reset to automatically repeat the A/D acquisition period with minimal software overhead (moving the ADRESH:ADRESL to the desired location). The appropriate analog input channel must be selected and the minimum acquisition done before the “special event trigger” sets the GO/$\overline{\text{DONE}}$ bit (starts a conversion).

If the A/D module is not enabled (ADON is cleared), then the “special event trigger” will be ignored by the A/D module but will still reset the Timer1 counter.

TABLE 11-2: REGISTERS/BITS ASSOCIATED WITH A/D

Address	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
0Bh,8Bh 10Bh,18Bh	INTCON	GIE	PEIE	TMR0IE	INTE	RBIE	TMR0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF	—	—	SSPIF	CCP1IF	TMR2IF	TMR1IF	-0-- 0000	-0-- 0000
8Ch	PIE1	—	ADIE	—	—	SSPIE	CCP1IE	TMR2IE	TMR1IE	-0-- 0000	-0-- 0000
1Eh	ADRESH	A/D Result Register High Byte								xxxx xxxx	uuuu uuuu
9Eh	ADRESL	A/D Result Register Low Byte								xxxx xxxx	uuuu uuuu
1Fh	ADCON0	ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/$\overline{\text{DONE}}$	—	ADON	0000 00-0	0000 00-0
9Fh	ADCON1	ADFM	ADCS2	—	—	PCFG3	PCFG2	PCFG1	PCFG0	00-- 0000	00-- 0000
05h	PORTA	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	xxx0 0000	uuu0 0000
85h	TRISA	TRISA7	TRISA6	TRISA5	PORTA Data Direction Register					1111 1111	1111 1111

Legend: x = unknown, u = unchanged, - = unimplemented, read as ‘0’. Shaded cells are not used for A/D conversion.

12.18 Low-Voltage ICSP Programming

The LVP bit of the Configuration Word register enables Low-Voltage ICSP Programming. This mode allows the microcontroller to be programmed via ICSP using a VDD source in the operating voltage range. This only means that VPP does not have to be brought to VIHH but can instead be left at the normal operating voltage. In this mode, the RB3/PGM pin is dedicated to the programming function and ceases to be a general purpose I/O pin.

If Low-Voltage Programming mode is not used, the LVP bit can be programmed to a '0' and RB3/PGM becomes a digital I/O pin. However, the LVP bit may only be programmed when Programming mode is entered with VIHH on $\overline{\text{MCLR}}$. The LVP bit can only be changed when using high voltage on $\overline{\text{MCLR}}$.

It should be noted that once the LVP bit is programmed to '0', only the High-Voltage Programming mode is available and only this mode can be used to program the device.

When using Low-Voltage ICSP, the part must be supplied at 4.5V to 5.5V if a bulk erase will be executed. This includes reprogramming of the code-protect bits from an ON state to an OFF state. For all other cases of Low-Voltage ICSP, the part may be programmed at the normal operating voltage. This means calibration values, unique user IDs or user code can be reprogrammed or added.

The following LVP steps assume the LVP bit is set in the Configuration Word register.

1. Apply VDD to the VDD pin.
2. Drive $\overline{\text{MCLR}}$ low.
3. Apply VDD to the RB3/PGM pin.
4. Apply VDD to the $\overline{\text{MCLR}}$ pin.
5. Follow with the associated programming steps.

Note 1: The High-Voltage Programming mode is always available, regardless of the state of the LVP bit, by applying VIHH to the $\overline{\text{MCLR}}$ pin.

2: While in Low-Voltage ICSP mode (LVP = 1), the RB3 pin can no longer be used as a general purpose I/O pin.

3: When using Low-Voltage ICSP Programming (LVP) and the pull-ups on PORTB are enabled, bit 3 in the TRISB register must be cleared to disable the pull-up on RB3 and ensure the proper operation of the device.

4: RB3 should not be allowed to float if LVP is enabled. An external pull-down device should be used to default the device to normal operating mode. If RB3 floats high, the PIC16F818/819 device will enter Programming mode.

5: LVP mode is enabled by default on all devices shipped from Microchip. It can be disabled by clearing the LVP bit in the Configuration Word register.

6: Disabling LVP will provide maximum compatibility to other PIC16CXXX devices.

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

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

CALL	Call Subroutine
Syntax:	[*label*] CALL k
Operands:	$0 \le k \le 2047$
Operation:	(PC) + 1 → TOS, k → PC<10:0>, (PCLATH<4:3>) → PC<12:11>
Status Affected:	None
Description:	Call subroutine. First, return address (PC + 1) is pushed onto the stack. The eleven-bit immediate address is loaded into PC bits<10:0>. The upper bits of the PC are loaded from PCLATH. `CALL` is a two-cycle instruction.

CLRF	Clear f
Syntax:	[*label*] CLRF f
Operands:	$0 \le f \le 127$
Operation:	00h → (f) 1 → Z
Status Affected:	Z
Description:	The contents of register ‘f’ are cleared and the Z bit is set.

CLRW	Clear W
Syntax:	[*label*] CLRW
Operands:	None
Operation:	00h → (W) 1 → Z
Status Affected:	Z
Description:	W register is cleared. Zero bit (Z) is set.

CLRWDT	Clear Watchdog Timer
Syntax:	[*label*] CLRWDT
Operands:	None
Operation:	00h → WDT 0 → WDT prescaler, 1 → $\overline{TO}$ 1 → $\overline{PD}$
Status Affected:	$\overline{TO}$, $\overline{PD}$
Description:	`CLRWDT` instruction resets the Watchdog Timer. It also resets the prescaler of the WDT. Status bits $\overline{TO}$ and $\overline{PD}$ are set.

FIGURE 15-1: PIC16F818/819 VOLTAGE-FREQUENCY GRAPH (INDUSTRIAL, EXTENDED)

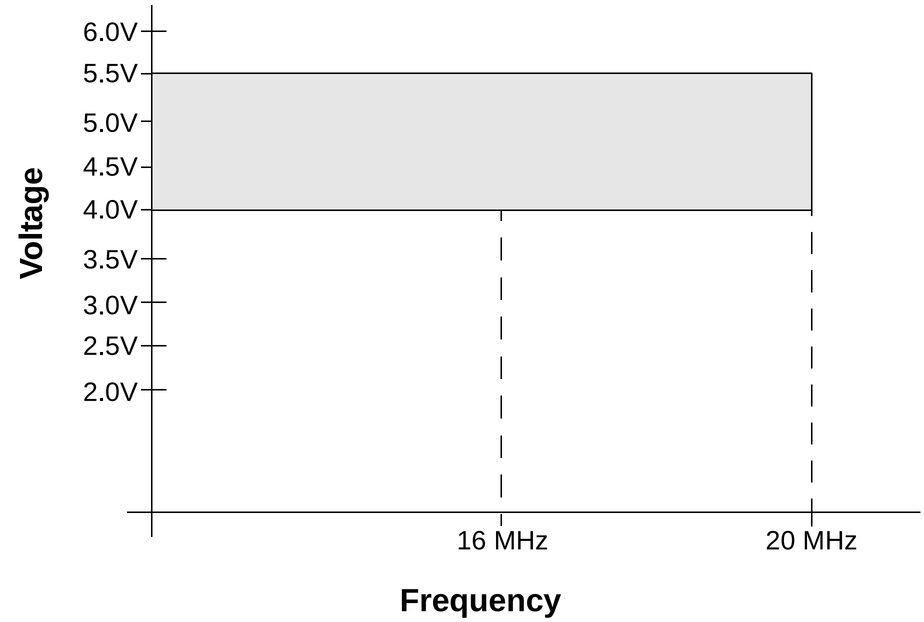
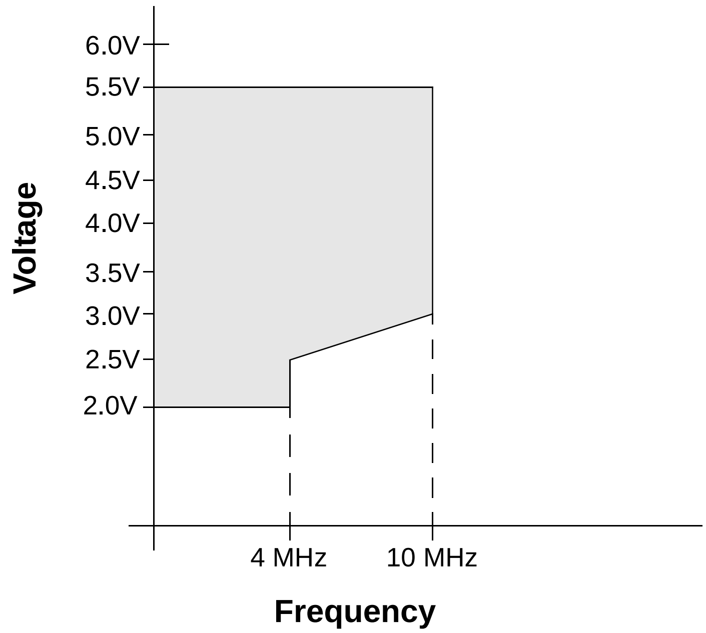


FIGURE 15-2: PIC16LF818/819 VOLTAGE-FREQUENCY GRAPH (INDUSTRIAL)



FMAX = (12 MHz/V) (VDDAPPMIN – 2.5V) + 4 MHz

Note 1: VDDAPPMIN is the minimum voltage of the PIC® device in the application.

2: FMAX has a maximum frequency of 10 MHz.

16.0 DC AND AC CHARACTERISTICS GRAPHS AND TABLES

Note:	The graphs and tables provided following this note are a statistical summary based on a limited number of samples and are provided for informational purposes only. The performance characteristics listed herein are not tested or guaranteed. In some graphs or tables, the data presented may be outside the specified operating range (e.g., outside specified power supply range) and therefore, outside the warranted range.

"Typical" represents the mean of the distribution at 25°C. "Maximum" or "minimum" represents (mean + 3σ) or (mean – 3σ) respectively, where σ is a standard deviation, over the whole temperature range.

FIGURE 16-1: TYPICAL IDD vs. FOSC OVER VDD (HS MODE)

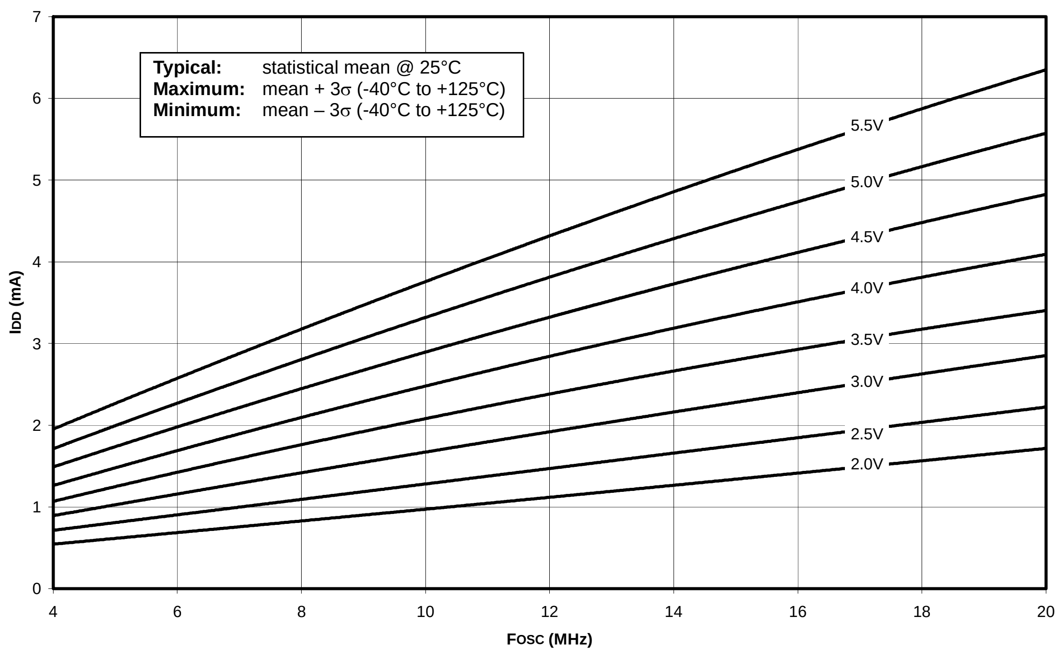


FIGURE 16-2: MAXIMUM IDD vs. FOSC OVER VDD (HS MODE)

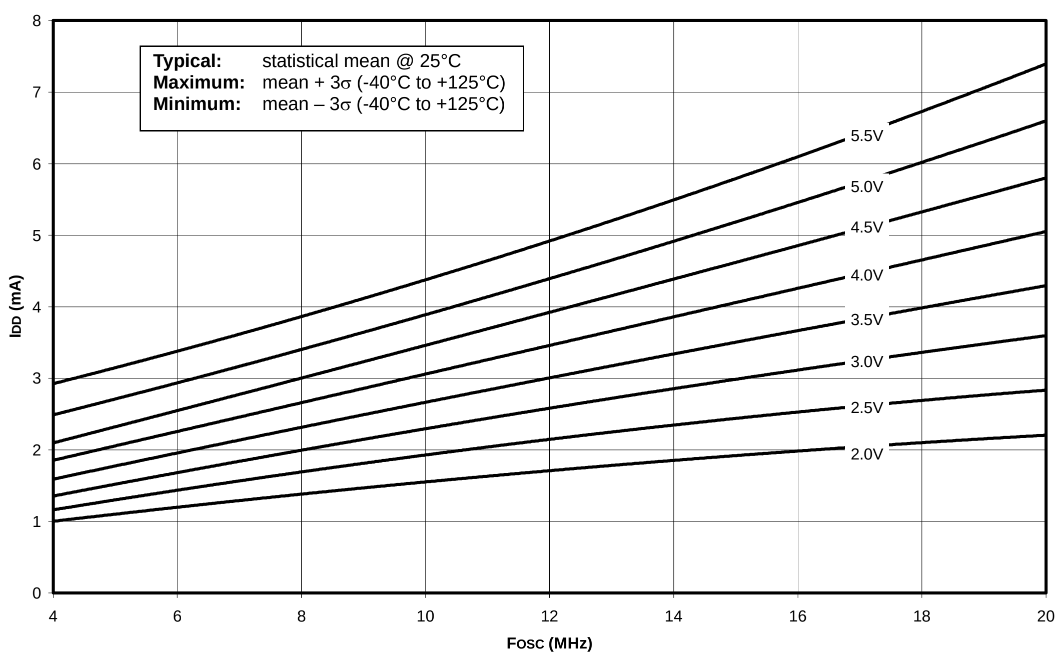


FIGURE 16-19: TYPICAL, MINIMUM AND MAXIMUM V_{OL} vs. I_{OL} (V_{DD} = 5V, -40°C TO +125°C)

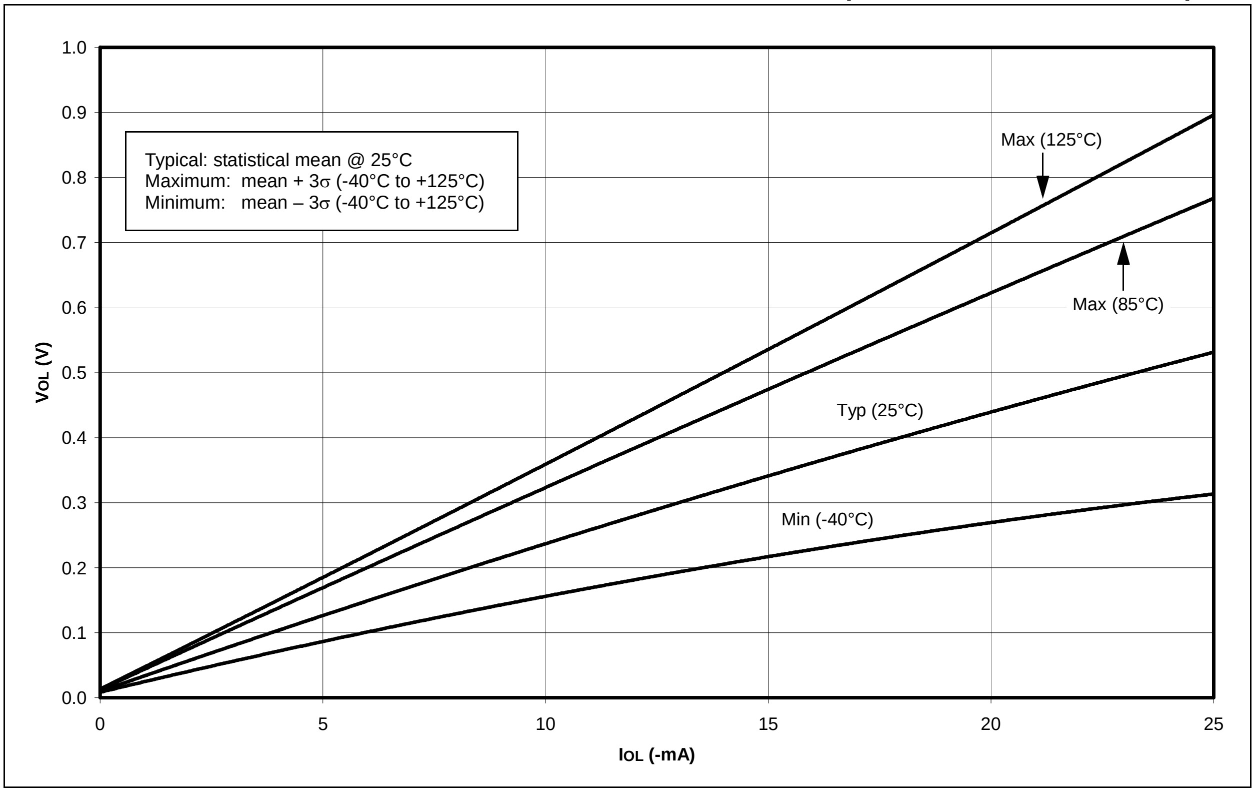
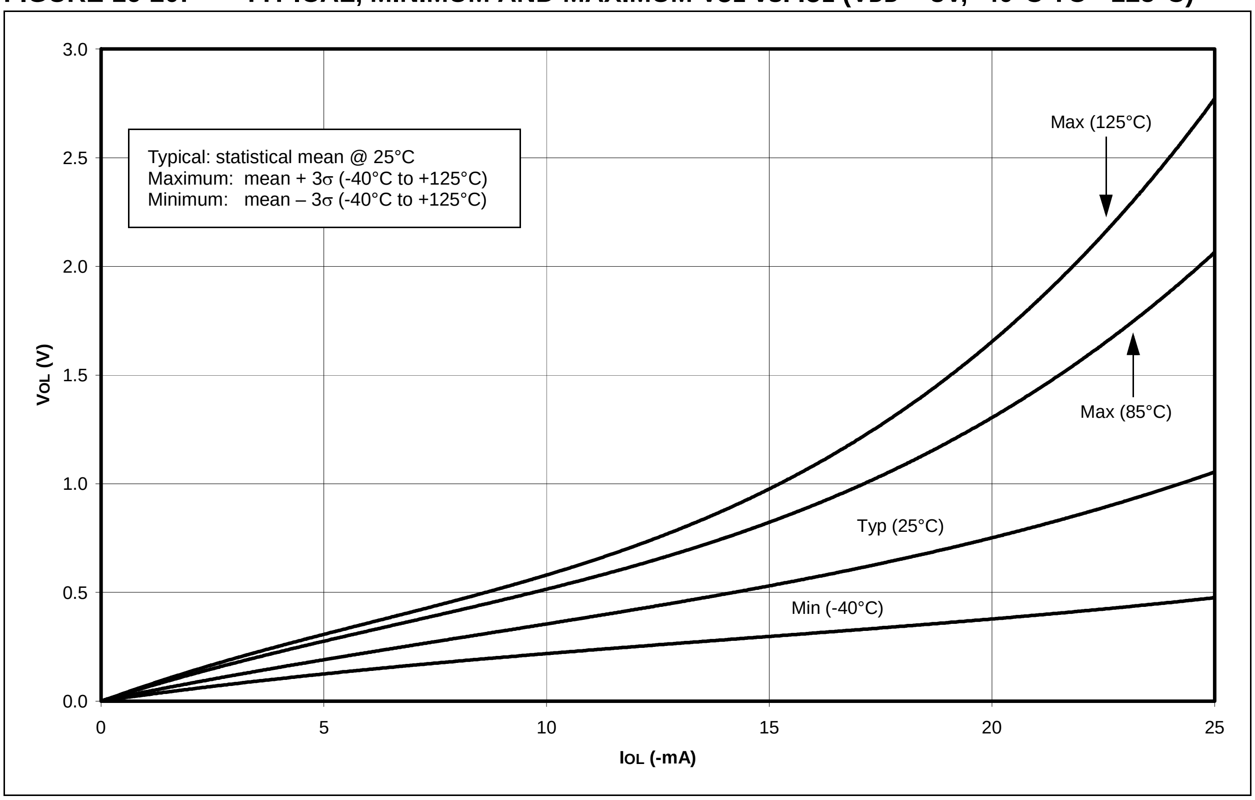
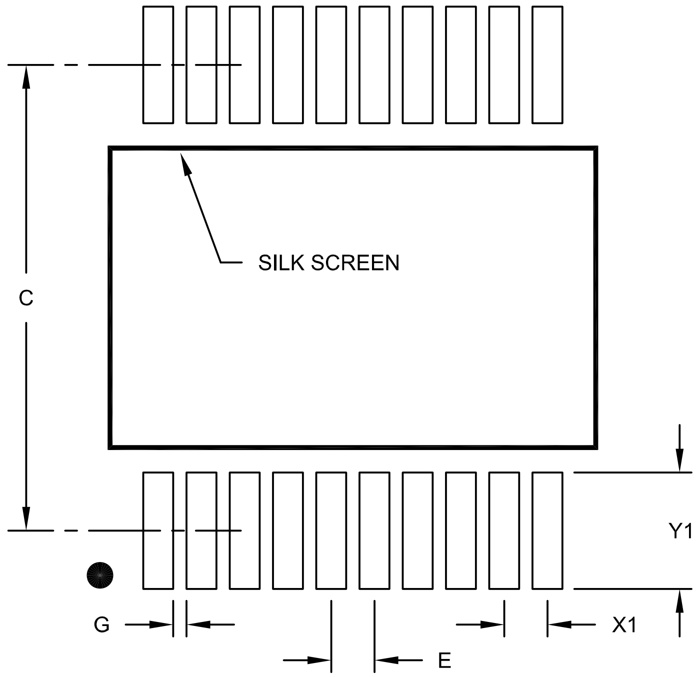


FIGURE 16-20: TYPICAL, MINIMUM AND MAXIMUM V_{OL} vs. I_{OL} (V_{DD} = 3V, -40°C TO +125°C)



20-Lead Plastic Shrink Small Outline (SS) - 5.30 mm Body [SSOP]

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		
Contact Pad Spacing	C		7.20	
Contact Pad Width (X20)	X1			0.45
Contact Pad Length (X20)	Y1			1.75
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. C04-2072A

D

E

F

G

I

V

W