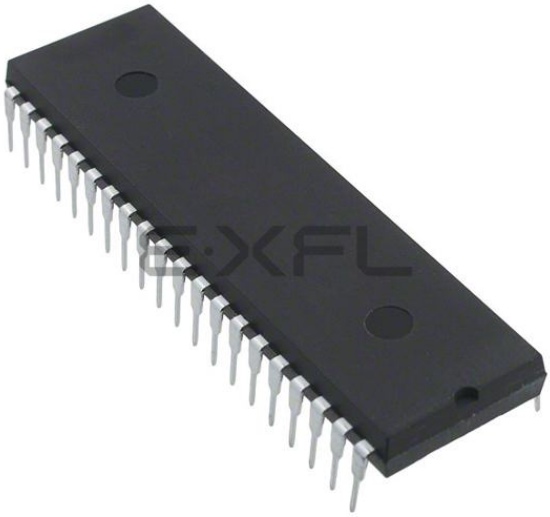


Welcome to E-XFL.COM

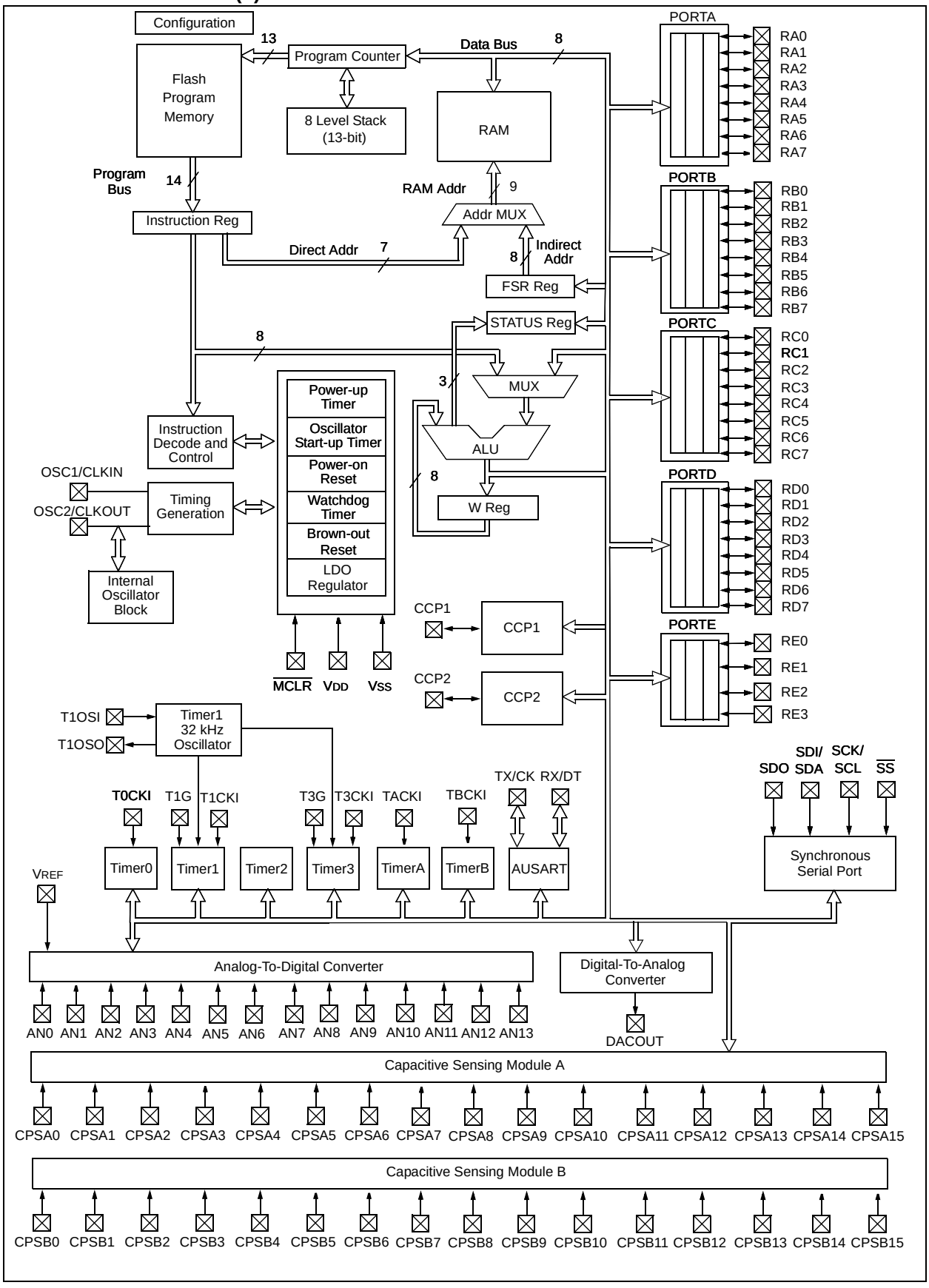
What is "Embedded - Microcontrollers"?

"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	20MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	36
Program Memory Size	14KB (8K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	363 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 5.5V
Data Converters	A/D 14x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Through Hole
Package / Case	40-DIP (0.600", 15.24mm)
Supplier Device Package	40-PDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f707-e-p

FIGURE 1-1: PIC16(L)F707 BLOCK DIAGRAM



4.5.1 INTCON REGISTER

The INTCON register is a readable and writable register, which contains the various enable and flag bits for TMR0 register overflow, PORTB change and external RB0/INT/SEG0 pin interrupts.

Note: Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global enable bit, GIE of the INTCON register. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 4-1: INTCON: INTERRUPT CONTROL REGISTER

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(1)	TMR0IF(2)	INTF	RBIF
bit 7							bit 0

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

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:** Timer0 Overflow Interrupt Enable bit
1 = Enables the Timer0 interrupt
0 = Disables the Timer0 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:** PORTB Change Interrupt Enable bit(1)
1 = Enables the PORTB change interrupt
0 = Disables the PORTB change interrupt

bit 2 **TMR0IF:** Timer0 Overflow Interrupt Flag bit(2)
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:** PORTB Change Interrupt Flag bit
1 = When at least one of the PORTB general purpose I/O pins changed state (must be cleared in software)
0 = None of the PORTB general purpose I/O pins have changed state

Note 1: The appropriate bits in the IOCB register must also be set.
2: TMR0IF bit is set when Timer0 rolls over. Timer0 is unchanged on Reset and should be initialized before clearing TMR0IF bit.

© 2010-2015 Microchip Technology Inc.

4.5.4 PIR1 REGISTER

The PIR1 register contains the interrupt flag bits, as shown in Register 4-4.

Note: Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global enable bit, GIE of the INTCON register. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 4-4: PIR1: PERIPHERAL INTERRUPT REQUEST REGISTER 1

R/W-0	R/W-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF
bit 7							bit 0

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

bit 7 **TMR1GIF:** Timer1 Gate Interrupt Flag bit
1 = Timer1 gate is inactive
0 = Timer1 gate is active

bit 6 **ADIF:** A/D Converter Interrupt Flag bit
1 = A/D conversion complete (must be cleared in software)
0 = A/D conversion has not completed or has not been started

bit 5 **RCIF:** USART Receive Interrupt Flag bit
1 = The USART receive buffer is full (cleared by reading RCREG)
0 = The USART receive buffer is not full

bit 4 **TXIF:** USART Transmit Interrupt Flag bit
1 = The USART transmit buffer is empty (cleared by writing to TXREG)
0 = The USART transmit buffer is full

bit 3 **SSPIF:** Synchronous Serial Port (SSP) Interrupt Flag bit
1 = The Transmission/Reception is complete (must be cleared in software)
0 = Waiting to Transmit/Receive

bit 2 **CCP1IF:** CCP1 Interrupt Flag bit
Capture mode:
1 = A Timer1 register capture occurred (must be cleared in software)
0 = No Timer1 register capture occurred
Compare mode:
1 = A Timer1 register compare match occurred (must be cleared in software)
0 = No Timer1 register compare match occurred
PWM mode:
Unused in this mode

bit 1 **TMR2IF:** Timer2 to PR2 Interrupt Flag bit
1 = A Timer2 to PR2 match occurred (must be cleared in software)
0 = No Timer2 to PR2 match occurred

bit 0 **TMR1IF:** Timer1 Overflow Interrupt Flag bit
1 = The Timer1 register overflowed (must be cleared in software)
0 = The Timer1 register did not overflow

9.3 A/D Acquisition Requirements

For the ADC to meet its specified accuracy, the charge holding capacitor (CHOLD) must be allowed to fully charge to the input channel voltage level. The analog input model is shown in Figure 9-3. The source impedance (RS) and the internal sampling switch (RSS) impedance directly affect the time required to charge the capacitor CHOLD. The sampling switch (RSS) impedance varies over the device voltage (VDD), refer to Figure 9-3. **The maximum recommended impedance for analog sources is 10 kΩ.** As the source impedance is decreased, the acquisition time may be decreased. After the analog input channel is selected (or changed), an A/D acquisition must be done before the conversion can be started. To calculate the minimum acquisition time, Equation 9-1 may be used. This equation assumes that 1/2 LSb error is used (256 steps for the ADC). The 1/2 LSb error is the maximum error allowed for the ADC to meet its specified resolution.

EQUATION 9-1: ACQUISITION TIME EXAMPLE

Assumptions: $Temperature = 50°C$ *and external impedance of* $10k\Omega$ *5.0V* V_{DD}

$$T_{ACQ} = \text{Amplifier Settling Time} + \text{Hold Capacitor Charging Time} + \text{Temperature Coefficient}$$
$$= T_{AMP} + T_C + T_{COFF}$$
$$= 2\mu s + T_C + [(Temperature - 25°C)(0.05\mu s/°C)]$$

The value for T_C *can be approximated with the following equations:*

$$V_{APPLIED}\left(1 - \frac{1}{(2^{n+1}) - 1}\right) = V_{CHOLD} \quad ;[1]\ V_{CHOLD}\ \text{charged to within 1/2 lsb}$$

$$V_{APPLIED}\left(1 - e^{\frac{-T_C}{RC}}\right) = V_{CHOLD} \quad ;[2]\ V_{CHOLD}\ \text{charge response to}\ V_{APPLIED}$$

$$V_{APPLIED}\left(1 - e^{\frac{-T_C}{RC}}\right) = V_{APPLIED}\left(1 - \frac{1}{(2^{n+1}) - 1}\right) \quad ;\text{combining [1] and [2]}$$

Note: *Where n = number of bits of the ADC.*

Solving for T_C*:*

$$T_C = -C_{HOLD}(R_{IC} + R_{SS} + R_S)\ ln(1/511)$$
$$= -10pF(1k\Omega + 7k\Omega + 10k\Omega)\ ln(0.001957)$$
$$= 1.12\mu s$$

Therefore:

$$T_{ACQ} = 2\mu s + 1.12\mu s + [(50°C - 25°C)(0.05\mu s/°C)]$$
$$= 4.42\mu s$$

Note 1: The reference voltage (VREF) has no effect on the equation, since it cancels itself out.

2: The charge holding capacitor (CHOLD) is not discharged after each conversion.

3: The maximum recommended impedance for analog sources is 10 kΩ. This is required to meet the pin leakage specification.

© 2010-2015 Microchip Technology Inc.

TABLE 11-1: REGISTERS ASSOCIATED WITH THE DIGITAL-TO-ANALOG CONVERTER

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
FVRCON	FVRRDY	FVREN	Reserved	Reserved	CDAFVR1	CDAFVR0	ADFVR1	ADFVR0	q000 0000	q000 0000
DACCON0	DACEN	DACLPS	DACOE	—	DACPSS1	DACPSS0	—	—	000- 00--	000- 00--
DACCON1	—	—	—	DACR4	DACR3	DACR2	DACR1	DACR0	---0 0000	---0 0000

Legend: — = Unimplemented locations, read as '0'. Shaded cells are not used by the DAC module.

© 2010-2015 Microchip Technology Inc.

13.3 Timer1/3 Prescaler

Timer1 and Timer3 have four prescaler options allowing 1, 2, 4 or 8 divisions of the clock input. The TxCKPS bits of the TxCON register control the prescale counter. The prescale counter is not directly readable or writable; however, the prescaler counter is cleared upon a write to TMRxH or TMRxL.

13.4 Timer1/3 Oscillator

A dedicated low-power 32.768 kHz oscillator circuit is built-in between pins T1OSI (input) and T1OSO (amplifier output). This internal circuit is to be used in conjunction with an external 32.768 kHz crystal.

The oscillator circuit is enabled by setting the T1OSCEN bit of the T1CON register. The oscillator can provide a clock source to Timer1 and/or Timer3. The oscillator will continue to run during Sleep.

Note:	The oscillator requires a start-up and stabilization time before use. Thus, T1OSCEN should be set and a suitable delay observed prior to enabling Timer1/3.

13.5 Timer1/3 Operation in Asynchronous Counter Mode

If control bit $\overline{\text{TxSYNC}}$ of the TxCON register is set, the external clock input is not synchronized. The timer increments asynchronously to the internal phase clocks. If external clock source is selected, then the timer will continue to run during Sleep and can generate an interrupt on overflow, which will wake-up the processor. However, special precautions in software are needed to read/write the timer (see **Section 13.5.1 "Reading and Writing Timer1/3 in Asynchronous Counter Mode"**).

13.5.1 READING AND WRITING TIMER1/3 IN ASYNCHRONOUS COUNTER MODE

Reading TMRxH or TMRxL while the timer is running from an external asynchronous clock will ensure a valid read (taken care of in hardware). However, the user should keep in mind that reading the 16-bit timer in two 8-bit values itself poses certain problems, since the timer may overflow between the reads.

For writes, it is recommended that the user simply stop the timer and write the desired values. A write contention may occur by writing to the timer registers, while the register is incrementing. This may produce an unpredictable value in the TMRxH:TMRxL register pair.

13.6 Timer1/3 Gate

Timer1/3 can be configured to count freely or the count can be enabled and disabled using Timer1/3 gate circuitry. This is also referred to as Timer1/3 gate count enable.

Timer1/3 gate can also be driven by multiple selectable sources.

13.6.1 TIMER1/3 GATE COUNT ENABLE

The Timer1/3 gate is enabled by setting the TMRxGE bit of the TxGCON register. The polarity of the Timer1/3 gate is configured using the TxGPOL bit of the TxGCON register.

When Timer1/3 gate (TxG) input is active, Timer1/3 will increment on the rising edge of the Timer1/3 clock source. When Timer1/3 gate input is inactive, no incrementing will occur and Timer1/3 will hold the current count. See Figure 13-3 for timing details.

TABLE 13-4: TIMER1/3 GATE ENABLE SELECTIONS

TxCLK	TxGPOL	TxG	Timer1/3 Operation
↑	0	0	Counts
↑	0	1	Holds Count
↑	1	0	Holds Count
↑	1	1	Counts

13.6.2 TIMER1/3 GATE SOURCE SELECTION

The Timer1/3 gate source can be selected from one of four different sources. Source selection is controlled by the TxGSS bits of the TxGCON register. The polarity for each available source is also selectable. Polarity selection is controlled by the TxGPOL bit of the TxGCON register.

14.1 TimerA/B Operation

The TimerA/B modules can be used as either 8-bit timers or 8-bit counters. Additionally, the modules can also be used to set Timer1's/Timer3's period of measurement for the capacitive sensing modules via Timer1's or Timer3's gate feature.

TABLE 14-1: CPSOSC/TIMER ASSOCIATION

Cap Sense Oscillator	Divider Timer	Period Measurement
CPS A	TimerA	Timer1
CPS B	TimerB	Timer3

14.1.1 8-BIT TIMER MODE

The TimerA/B modules will increment every instruction cycle, if used without a prescaler. 8-bit Timer mode is selected by clearing the TMRxCS bit of the TxCON registers.

When TMRx is written, the increment is inhibited for two instruction cycles immediately following the write.

Note:	The value written to the TMRx register can be adjusted, in order to account for the two instruction cycle delay when TMRx is written.

14.1.2 8-BIT COUNTER MODE

In 8-bit Counter mode, the TimerA/B modules will increment on every rising or falling edge of the TxCKI pin or the Capacitive Sensing Oscillator (CPSxOSC) signal. 8-bit Counter mode using the TxCKI pin is selected by setting the TMRxCS bit of the TxCON register to '1' and resetting the TxXCS bit in the CPSxCON0 register to '0'. 8-bit Counter mode using the Capacitive Sensing Oscillator (CPSxOSC) signal is selected by setting the TMRxCS bit in the TxCON register to '1' and setting the TxXCS bit in the CPSxCON0 register to '1'.

The rising or falling transition of the incrementing edge for either input source is determined by the TMRxSE bit in the TxCON register.

14.1.3 SOFTWARE PROGRAMMABLE PRESCALER

For TimerA/B modules, the software programmable prescaler is exclusive to the Timer. The prescaler is enabled by clearing the TMRxPSA bit of the TxCON register.

There are eight prescaler options for TimerA/B modules ranging from 1:2 to 1:256. The prescale values are selectable via the TMRxPS<2:0> bits of the TxCON register for TimerA/B. In order to have a 1:1 prescaler value for the TimerA/B modules, the prescaler must be disabled.

The prescaler is not readable or writable. When the prescaler is enabled or assigned to the Timer module, all instructions writing to the TMRx register will clear the prescaler. Enabling the TimerA/B modules also clears the prescaler.

14.1.4 TIMERA/B INTERRUPT

TimerA/B will generate an interrupt when the corresponding TMR register overflows from FFh to 00h. The TMRxIF interrupt flag bit of the PIR2 register is set every time the TMRx register overflows. These interrupt flag bits are set regardless of whether or not the relative Timer interrupt is enabled. The interrupt flag bits can only be cleared in software. The TimerA/B interrupt enable bits are the TMRxIE in the PIE2 register.

Note:	TimerA/B interrupts cannot wake the processor from Sleep since the timer is frozen during Sleep.

14.1.5 USING TIMERA/B WITH AN EXTERNAL CLOCK

When TimerA/B is in Counter mode, the synchronization of the TxCKI input and the TMRx register is accomplished by sampling the prescaler output on the Q2 and Q4 cycles of the internal phase clocks. Therefore, the high and low periods of the external clock source must meet the timing requirements as shown in **Section 25.0 "Electrical Specifications"**.

14.1.6 TIMER ENABLE

Operation of TimerA/B is enabled by setting the TMRxON bit of the TxCON register. When the module is disabled, the value in the TMRx register is maintained. Enabling the TMRx module will reset the prescaler used by the counter.

14.1.7 OPERATION DURING SLEEP

TimerA and TimerB cannot operate while the processor is in Sleep mode. The contents of the TMRx registers will remain unchanged while the processor is in Sleep mode.

REGISTER 16-2: CPSxCON1: CAPACITIVE SENSING CONTROL REGISTER 1

U-0	U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	—	CPSxCH3	CPSxCH2	CPSxCH1	CPSxCH0
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 **CPSxCH<3:0>:** Capacitive Sensing Channel Select bits

If CPSxON = 0:

These bits are ignored. No channel is selected.

If CPSxON = 1:

0000 = channel 0, (CPSx0)
0001 = channel 1, (CPSx1)
0010 = channel 2, (CPSx2)
0011 = channel 3, (CPSx3)
0100 = channel 4, (CPSx4)
0101 = channel 5, (CPSx5)
0110 = channel 6, (CPSx6)
0111 = channel 7, (CPSx7)
1000 = channel 8, (CPSx8)
1001 = channel 9, (CPSx9)
1010 = channel 10, (CPSx10)
1011 = channel 11, (CPSx11)
1100 = channel 12, (CPSx12)
1101 = channel 13, (CPSx13)
1110 = channel 14, (CPSx14)
1111 = channel 15, (CPSx15)

19.1.2 SLAVE MODE

For any SPI device acting as a slave, the data is transmitted and received as external clock pulses appear on the SCK pin. This external clock must meet the minimum high and low times as specified in the electrical specifications.

19.1.2.1 Slave Mode Operation

The SSP consists of a transmit/receive shift register (SSPSR) and a buffer register (SSPBUF). The SSPSR shifts the data in and out of the device, MSb first. The SSPBUF holds the data that was written to the SSPSR until the received data is ready.

The slave has no control as to when data will be clocked in or out of the device. All data that is to be transmitted, to a master or another slave, must be loaded into the SSPBUF register before the first clock pulse is received.

Once eight bits of data have been received:

- Received byte is moved to the SSPBUF register
- BF bit of the SSPSTAT register is set
- SSPIF bit of the PIR1 register is set

Any write to the SSPBUF register during transmission/ reception of data will be ignored and the Write Collision Detect bit, WCOL of the SSPCON register, will be set. User software must clear the WCOL bit so that it can be determined if the following write(s) to the SSPBUF register completed successfully.

The user's firmware must read SSPBUF, clearing the BF flag, or the SSPOV bit of the SSPCON register will be set with the reception of the next byte and communication will be disabled.

A SPI module transmits and receives at the same time, occasionally causing dummy data to be transmitted/ received. It is up to the user to determine which data is to be used and what can be discarded.

19.1.2.2 Enabling Slave I/O

To enable the serial port, the SSPEN bit of the SSPCON register must be set. If a Slave mode of operation is selected in the SSPM bits of the SSPCON register, the SDI, SDO and SCK pins will be assigned as serial port pins.

For these pins to function as serial port pins, they must have their corresponding data direction bits set or cleared in the associated TRIS register as follows:

- SDI configured as input
- SDO configured as output
- SCK configured as input

Optionally, a fourth pin, Slave Select ($\overline{SS}$) may be used in Slave mode. Slave Select may be configured to operate on one of the following pins via the SSSEL bit in the APFCON register.

- RA5/AN4/$\overline{SS}$
- RA0/AN0/$\overline{SS}$

Upon selection of a Slave Select pin, the appropriate bits must be set in the ANSELA and TRISA registers. Slave Select must be set as an input by setting the corresponding bit in TRISA, and digital I/O must be enabled on the $\overline{SS}$ pin by clearing the corresponding bit of the ANSELA register.

19.1.2.3 Slave Mode Setup

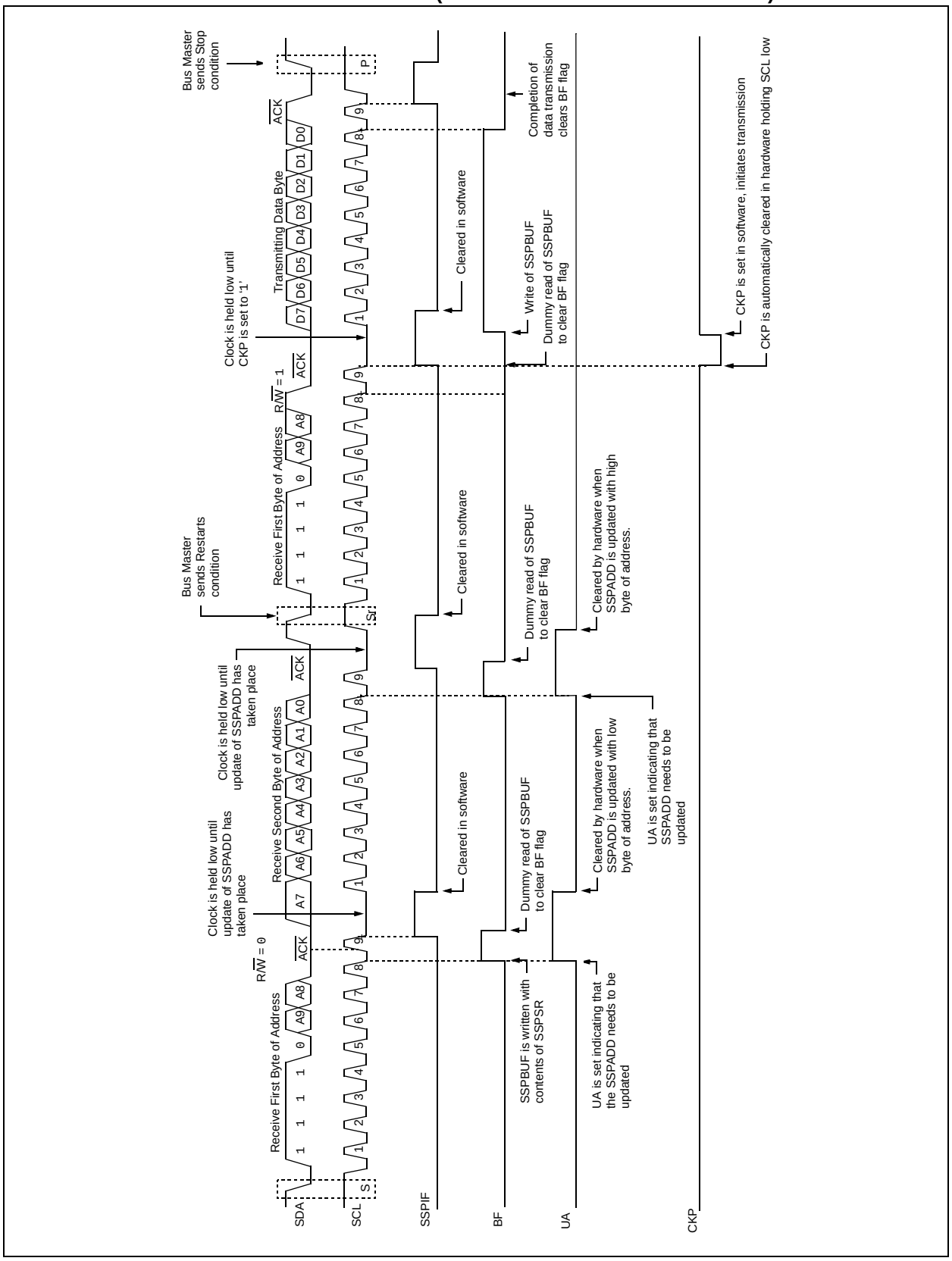
When initializing the SSP module to SPI Slave mode, compatibility must be ensured with the master device. This is done by programming the appropriate control bits of the SSPCON and SSPSTAT registers. These control bits allow the following to be specified:

- SCK as clock input
- Idle state of SCK (CKP bit)
- Data input sample phase (SMP bit)
- Output data on rising/falling edge of SCK (CKE bit)

Figure 19-4 and Figure 19-5 show example waveforms of Slave mode operation.

© 2010-2015 Microchip Technology Inc.

FIGURE 19-13: I²C SLAVE MODE TIMING (TRANSMISSION 10-BIT ADDRESS)



© 2010-2015 Microchip Technology Inc.

SUBWF **Subtract W from f**

Syntax: [*label*] SUBWF f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: (f) - (W) → (destination)

Status Affected: C, DC, Z

Description: Subtract (2's complement method) W register from register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f.

C = 0	W > f
C = 1	W ≤ f
DC = 0	W<3:0> > f<3:0>
DC = 1	W<3:0> ≤ f<3:0>

SWAPF **Swap Nibbles in f**

Syntax: [*label*] SWAPF f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: (f<3:0>) → (destination<7:4>),
(f<7:4>) → (destination<3:0>)

Status Affected: None

Description: The upper and lower nibbles of register 'f' are exchanged. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed in register 'f'.

XORLW **Exclusive OR literal with W**

Syntax: [*label*] XORLW k

Operands: $0 \le k \le 255$

Operation: (W) .XOR. k → (W)

Status Affected: Z

Description: The contents of the W register are XOR'ed with the 8-bit literal 'k'. The result is placed in the W register.

XORWF **Exclusive OR W with f**

Syntax: [*label*] XORWF f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: (W) .XOR. (f) → (destination)

Status Affected: Z

Description: Exclusive OR the contents of the W register with register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f'.

25.5 Thermal Considerations

Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +125°C					
Param No.	**Sym**	**Characteristic**	**Typ**	**Units**	**Conditions**
TH01	θJA	Thermal Resistance Junction to Ambient	47.2	°C/W	40-pin PDIP package
			46	°C/W	44-pin TQFP package
			24.4	°C/W	44-pin QFN 8x8mm package
			TBD	°C/W	40-pin UQFN 5x5mm package
TH02	θJC	Thermal Resistance Junction to Case	24.7	°C/W	40-pin PDIP package
			14.5	°C/W	44-pin TQFP package
			20	°C/W	44-pin QFN 8x8mm package
			TBD	°C/W	40-pin UQFN 5x5mm package
TH03	TJMAX	Maximum Junction Temperature	150	°C	
TH04	PD	Power Dissipation	—	W	PD = PINTERNAL + PI/O
TH05	PINTERNAL	Internal Power Dissipation	—	W	PINTERNAL = IDD x VDD(1)
TH06	PI/O	I/O Power Dissipation	—	W	PI/O = Σ (IOL * VOL) + Σ (IOH * (VDD - VOH))
TH07	PDER	Derated Power	—	W	PDER = PDMAX (TJ - TA)/θJA(2, 3)

Note 1: IDD is current to run the chip alone without driving any load on the output pins.
2: TA = Ambient Temperature
3: TJ = Junction Temperature

© 2010-2015 Microchip Technology Inc.

FIGURE 26-2: PIC16LF707 MAXIMUM IDD vs. FOSC OVER VDD, EC MODE

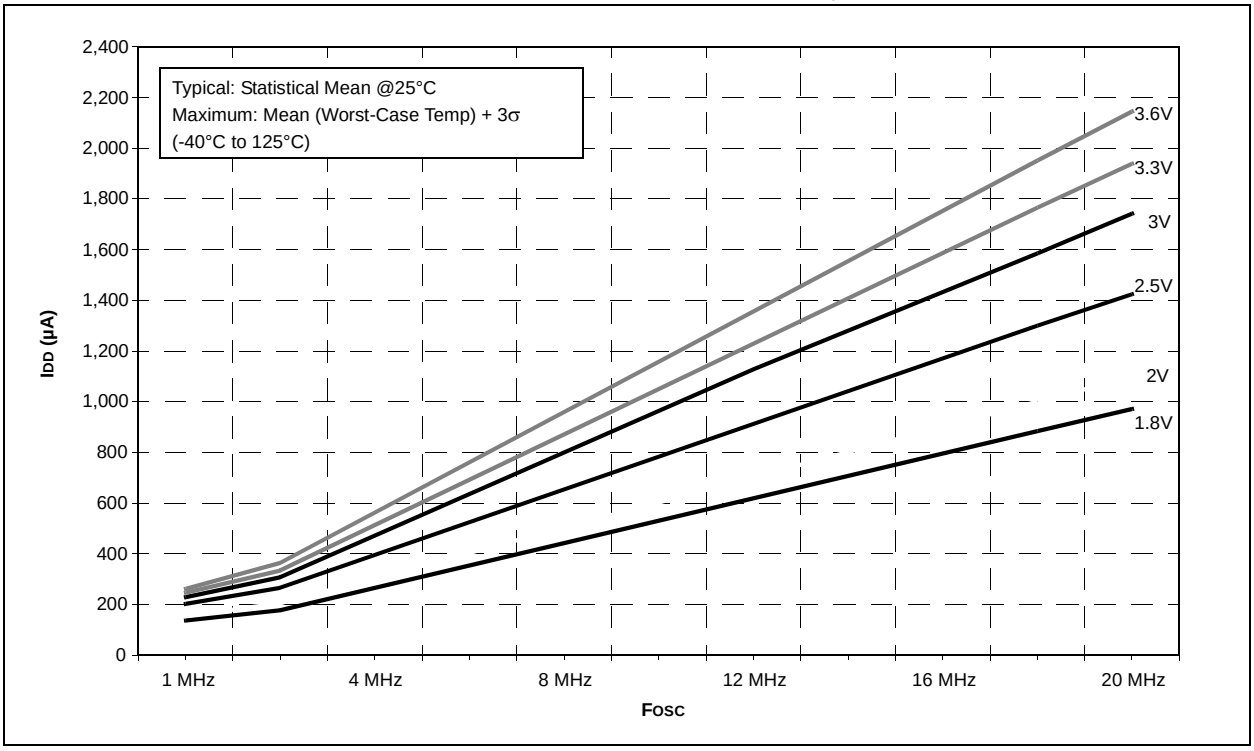


FIGURE 26-3: PIC16F707 TYPICAL IDD vs. FOSC OVER VDD, EC MODE, VCAP = 0.1µF

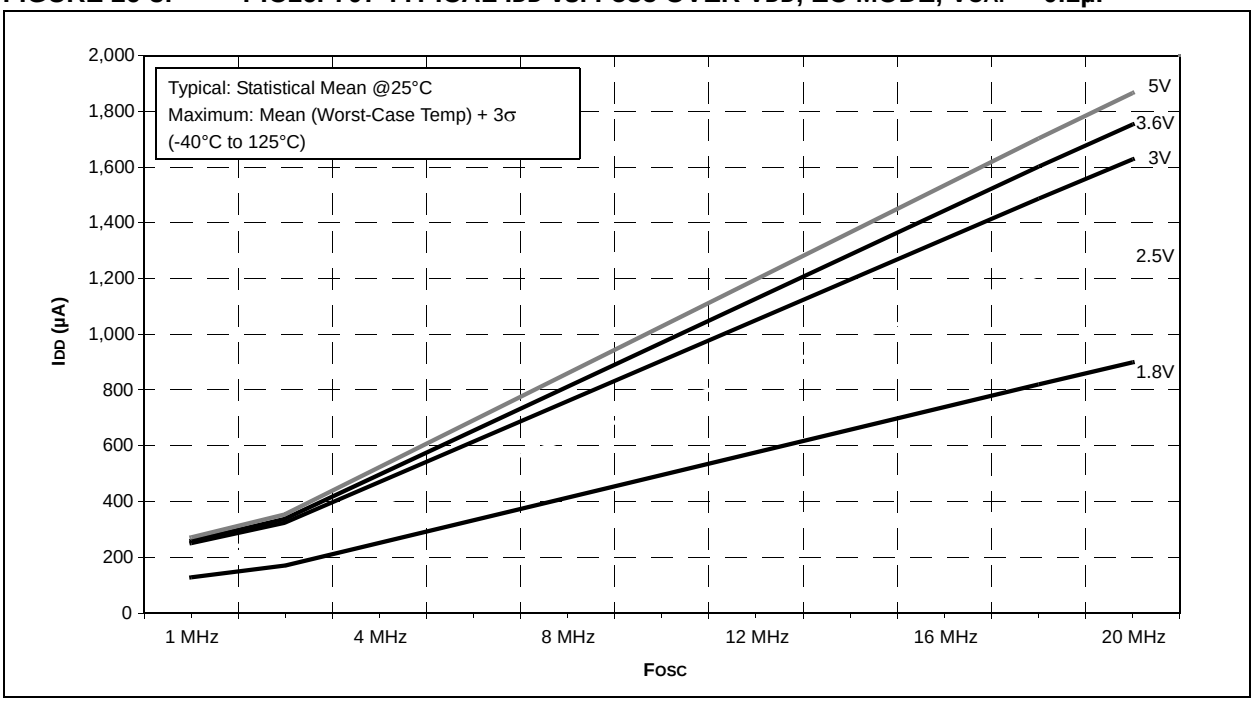


FIGURE 26-14: PIC16LF707 MAXIMUM IDD vs. VDD OVER FOSC, XT MODE

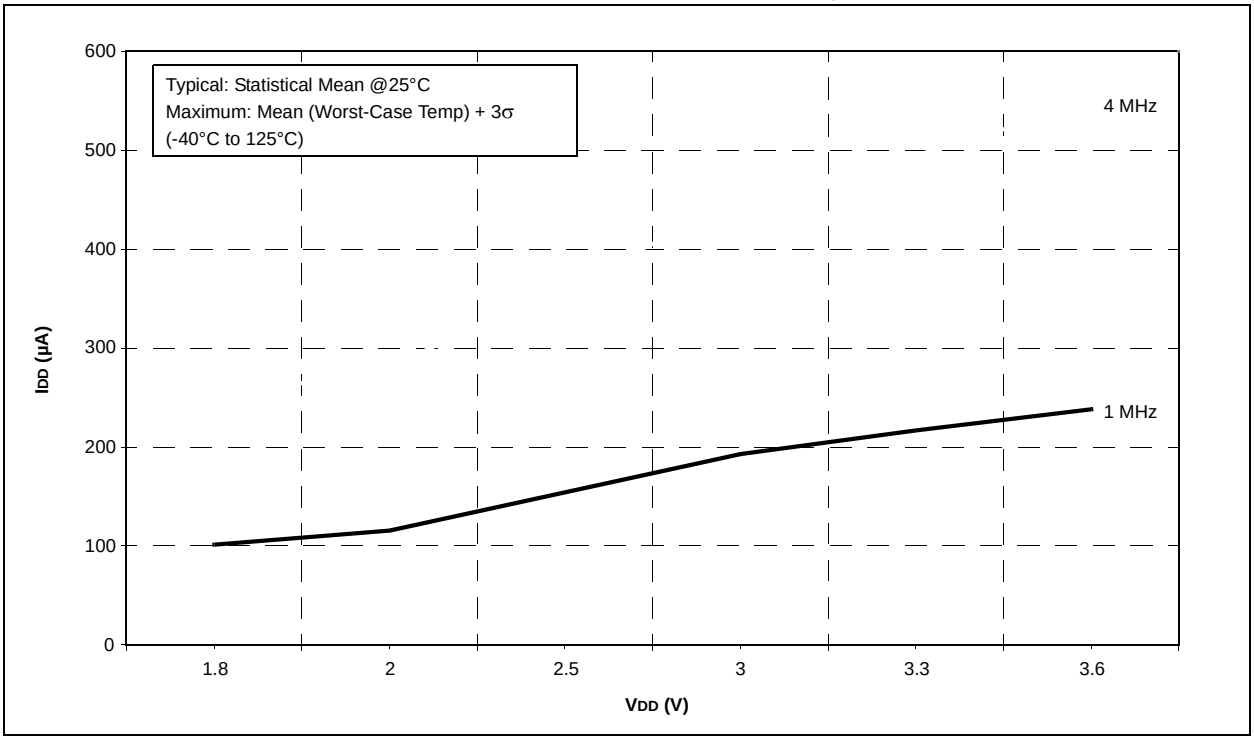


FIGURE 26-15: PIC16F707 TYPICAL IDD vs. VDD OVER FOSC, XT MODE, VCAP = 0.1µF

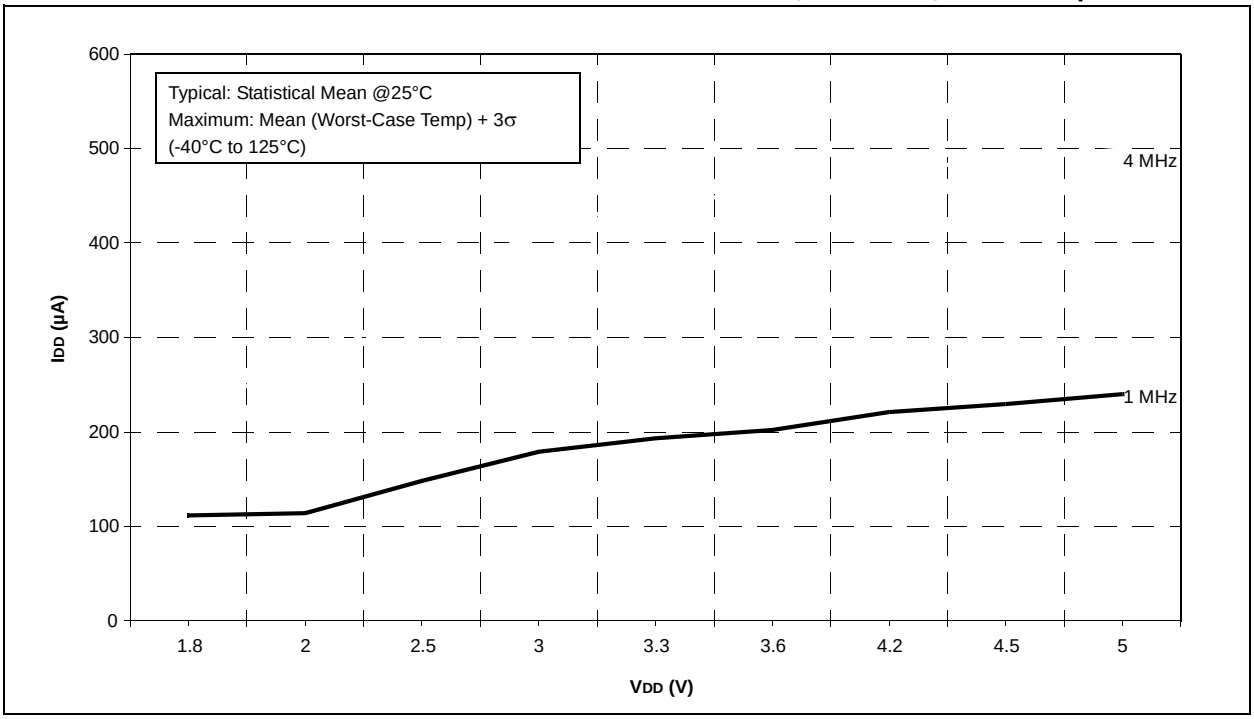


FIGURE 26-20: PIC16LF707 MAXIMUM IDD vs. FOSC OVER VDD, INTOSC MODE

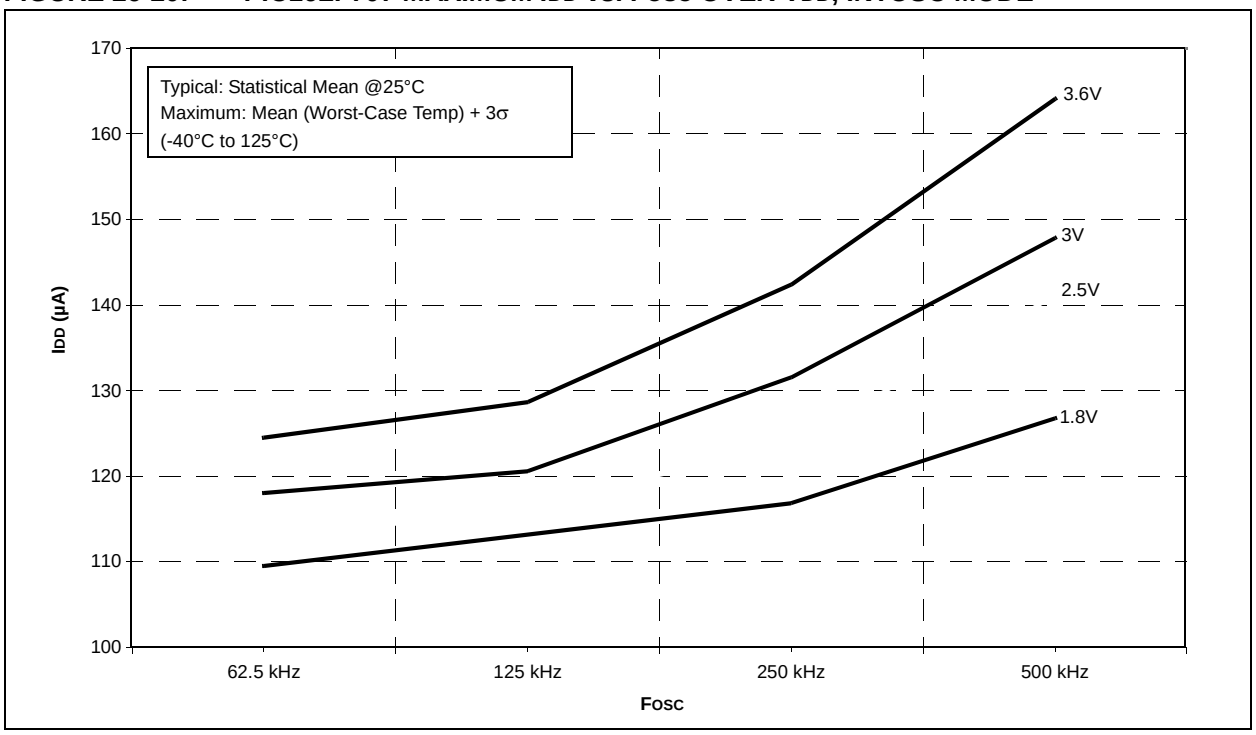
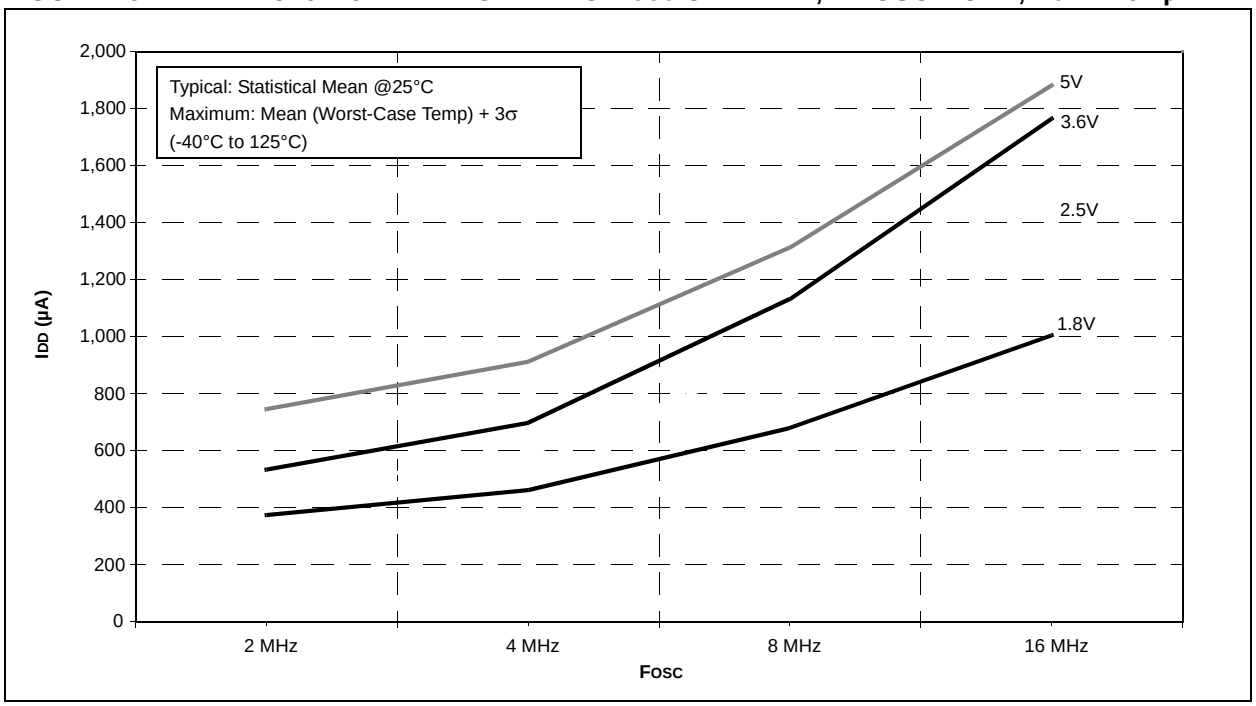


FIGURE 26-21: PIC16F707 MAXIMUM IDD vs. FOSC OVER VDD, INTOSC MODE, VCAP = 0.1µF



© 2010-2015 Microchip Technology Inc.

FIGURE 26-28: PIC16LF707 MAXIMUM BASE I_{PD} vs. V_{DD}

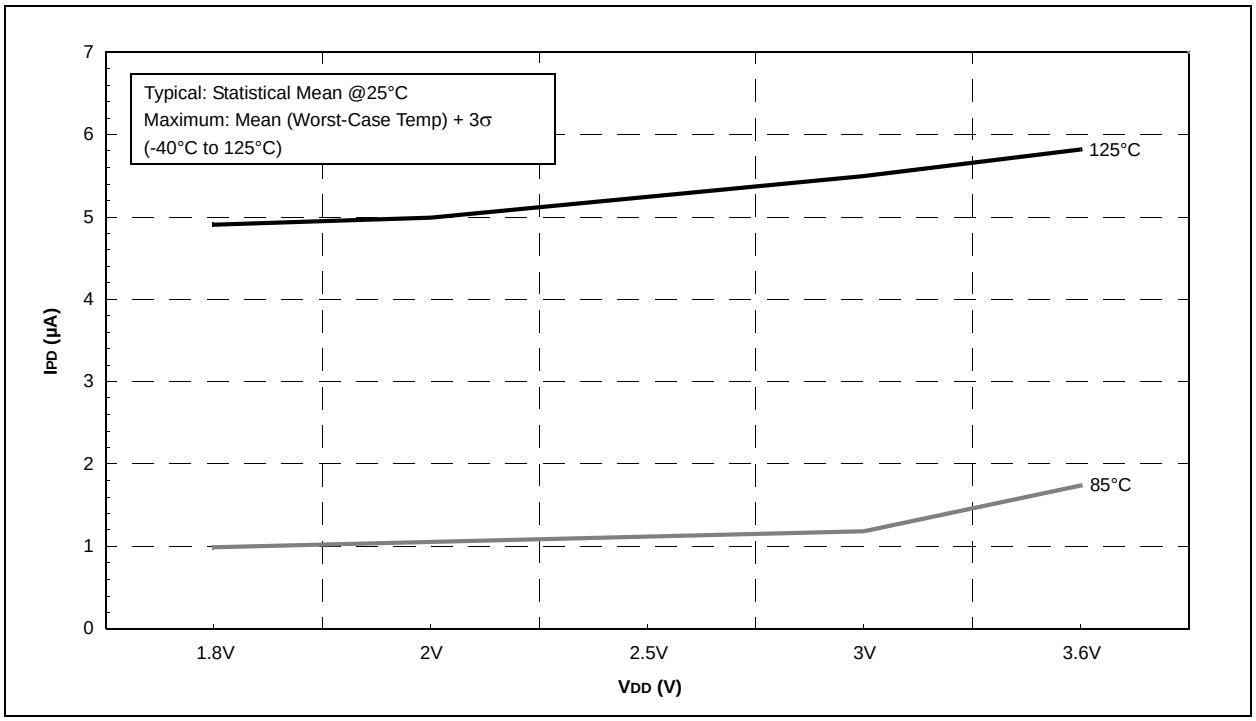
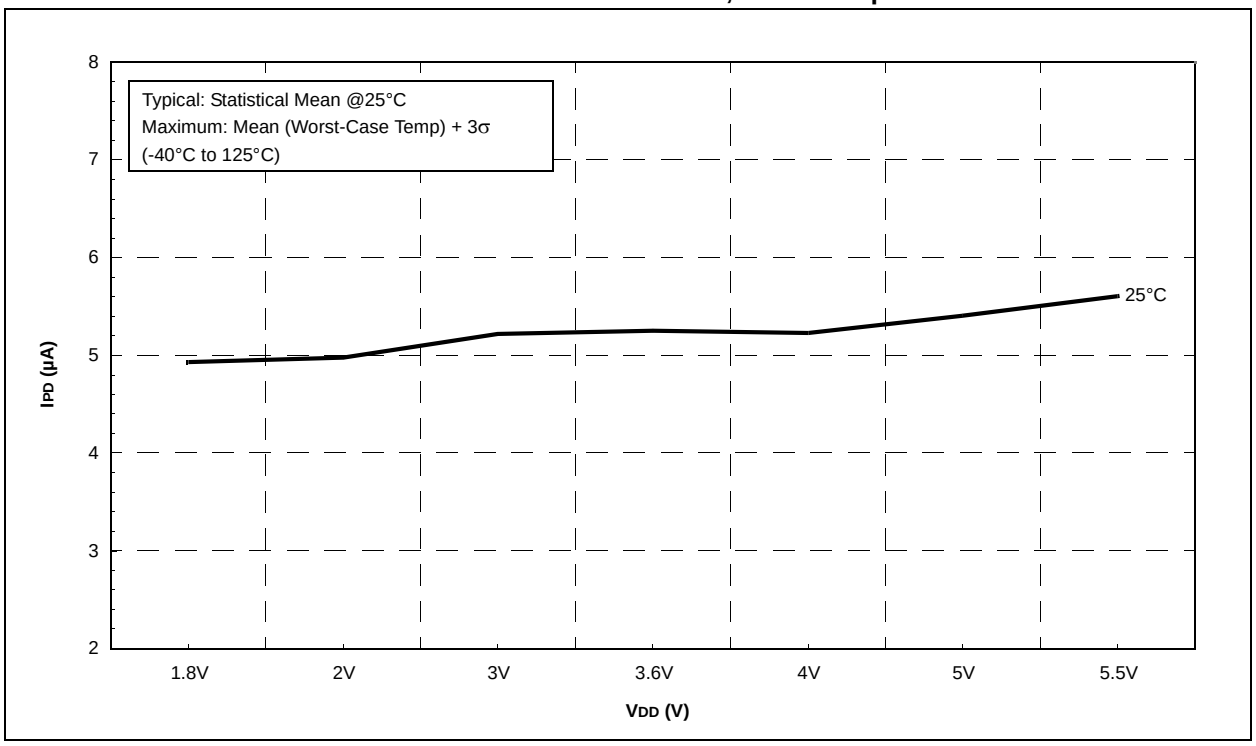


FIGURE 26-29: PIC16F707 TYPICAL BASE I_{PD} vs. V_{DD}, V_{CAP} = 0.1µF



© 2010-2015 Microchip Technology Inc.

FIGURE 26-44: PIC16LF707 TYPICAL ADC IPD vs. VDD

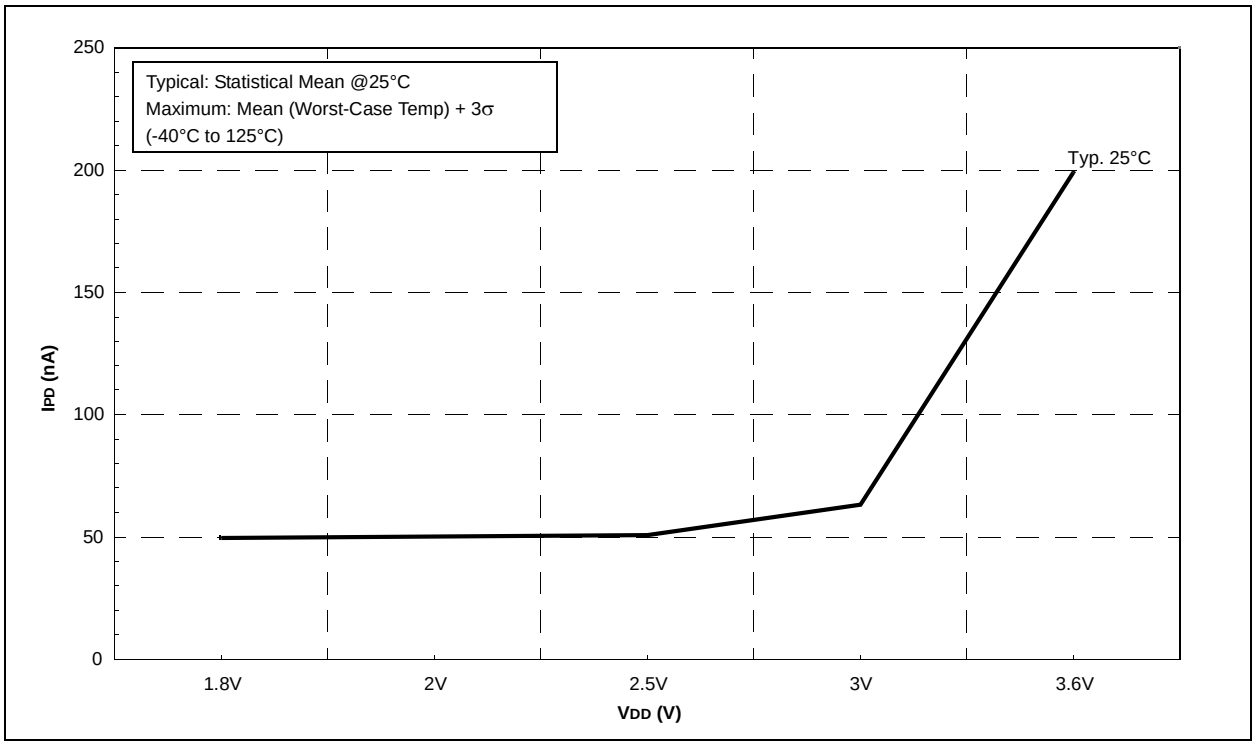
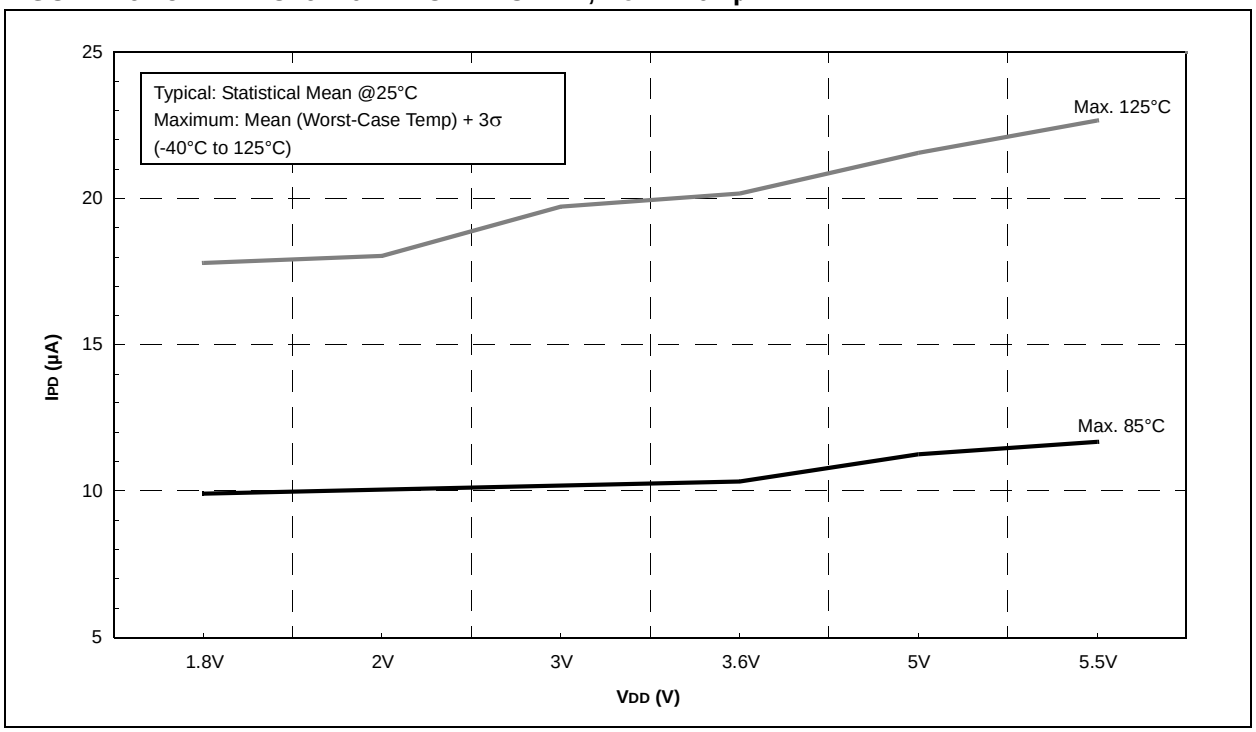


FIGURE 26-45: PIC16F707 ADC IPD vs. VDD, VCAP = 0.1µF



© 2010-2015 Microchip Technology Inc.

FIGURE 26-52: VOH vs. IOH OVER TEMPERATURE, VDD = 5.5V

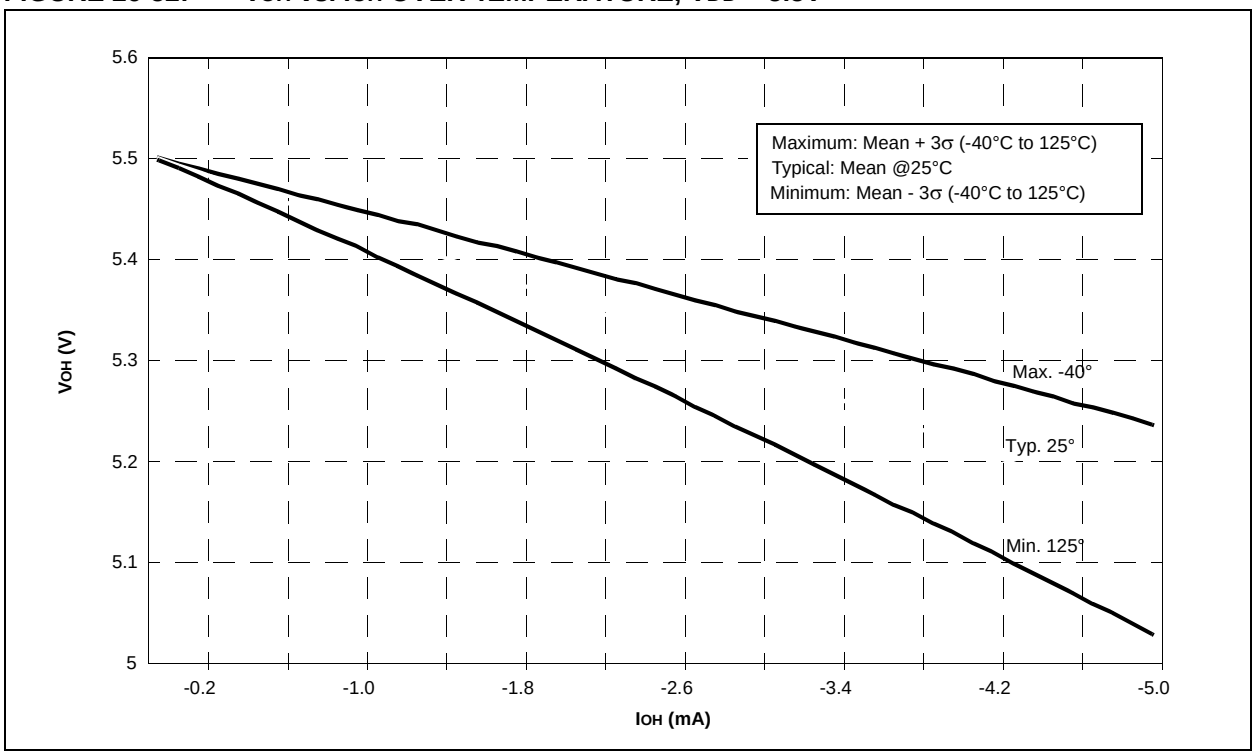
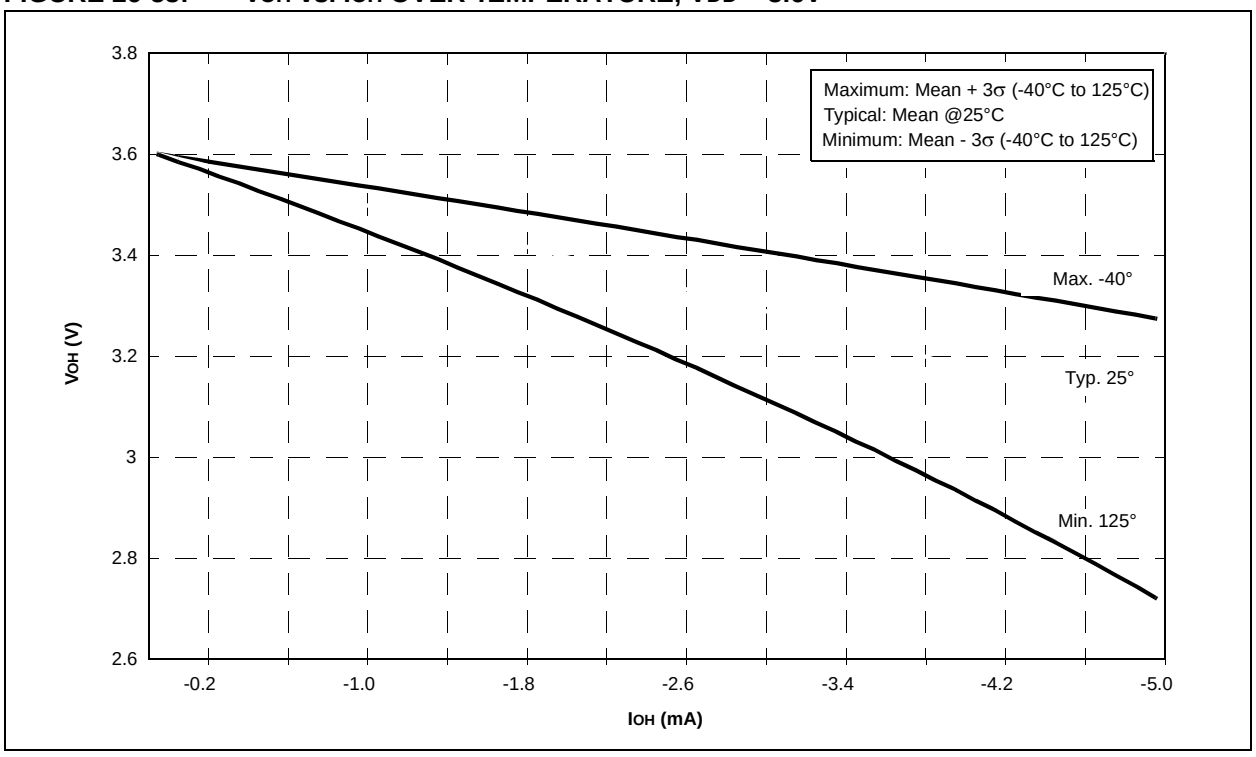


FIGURE 26-53: VOH vs. IOH OVER TEMPERATURE, VDD = 3.6V



© 2010-2015 Microchip Technology Inc.

FIGURE 26-60: PIC16F707 HFINTOSC WAKE-UP FROM SLEEP START-UP TIME

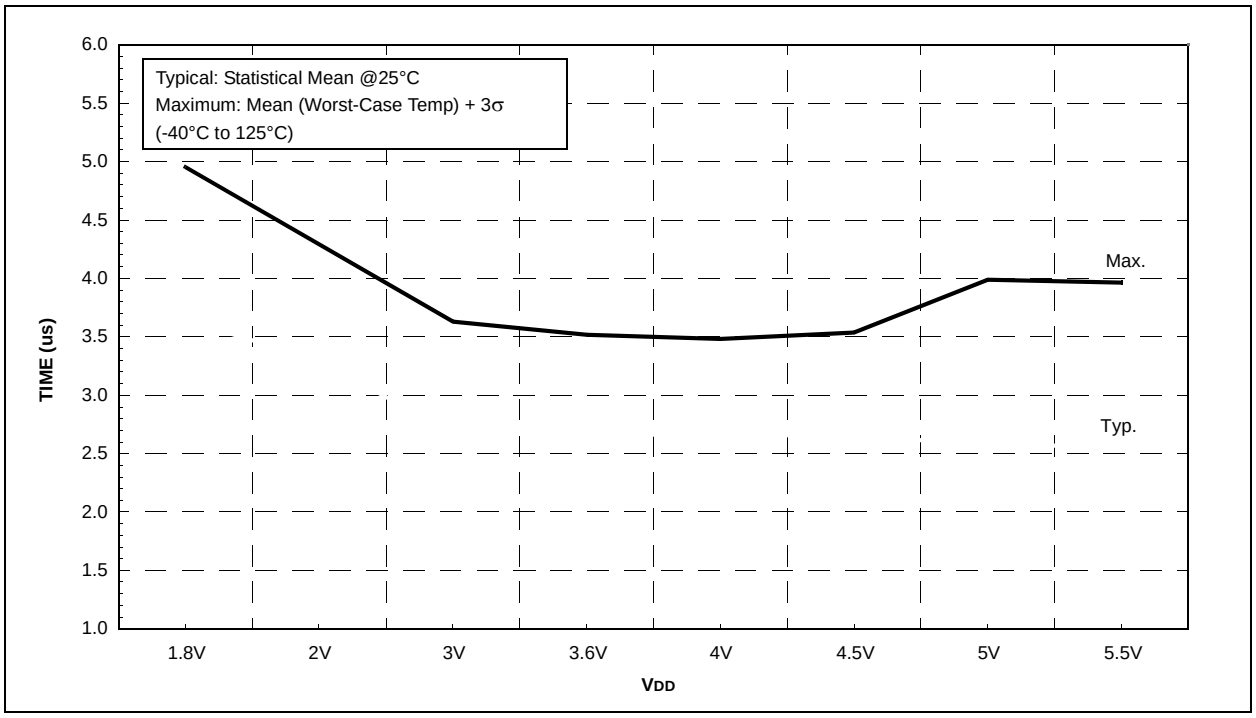
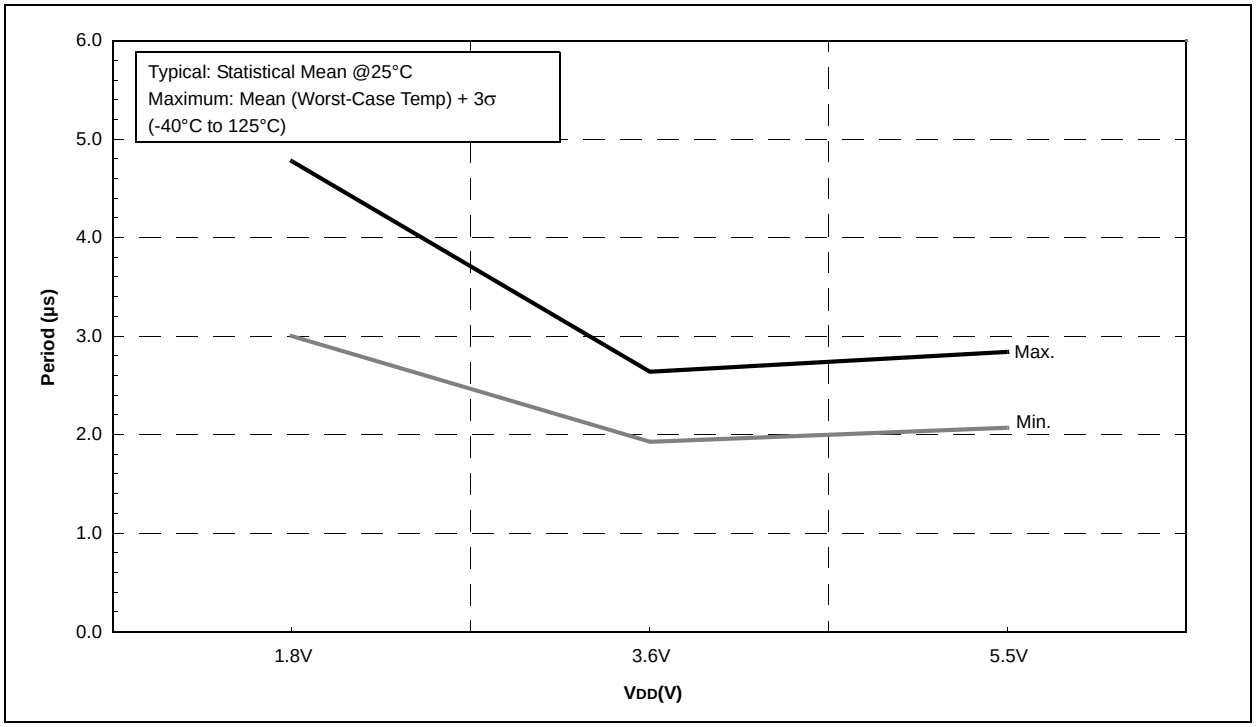


FIGURE 26-61: PIC16F707 A/D INTERNAL RC OSCILLATOR PERIOD



© 2010-2015 Microchip Technology Inc.

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

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

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

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE**.** Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Trademarks

The Microchip name and logo, the Microchip logo, dsPIC, FlashFlex, flexPWR, JukeBlox, KEELOQ, KEELOQ logo, Kleer, LANCheck, MediaLB, MOST, MOST logo, MPLAB, OptoLyzer, PIC, PICSTART, PIC^{32} logo, RightTouch, SpyNIC, SST, SST Logo, SuperFlash and UNI/O are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

The Embedded Control Solutions Company and mTouch are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Analog-for-the-Digital Age, BodyCom, chipKIT, chipKIT logo, CodeGuard, dsPICDEM, dsPICDEM.net, ECAN, In-Circuit Serial Programming, ICSP, Inter-Chip Connectivity, KleerNet, KleerNet logo, MiWi, motorBench, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, MultiTRAK, NetDetach, Omniscient Code Generation, PICDEM, PICDEM.net, PICkit, PICtail, RightTouch logo, REAL ICE, SQI, Serial Quad I/O, Total Endurance, TSHARC, USBCheck, VariSense, ViewSpan, WiperLock, Wireless DNA, and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

Silicon Storage Technology is a registered trademark of Microchip Technology Inc. in other countries.

GestIC is a registered trademark of Microchip Technology Germany II GmbH & Co. KG, a subsidiary of Microchip Technology Inc., in other countries.

All other trademarks mentioned herein are property of their respective companies.

© 2010-2015, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

ISBN: 978-1-5224-0043-1

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

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