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	16MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	17
Program Memory Size	3.5KB (2K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	128 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 12x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Through Hole
Package / Case	20-DIP (0.300", 7.62mm)
Supplier Device Package	20-PDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf720-e-p

TO OUR VALUED CUSTOMERS

It is our intention to provide our valued customers with the best documentation possible to ensure successful use of your Microchip products. To this end, we will continue to improve our publications to better suit your needs. Our publications will be refined and enhanced as new volumes and updates are introduced.

If you have any questions or comments regarding this publication, please contact the Marketing Communications Department via E-mail at **docerrors@microchip.com**. We welcome your feedback.

Most Current Data Sheet

To obtain the most up-to-date version of this data sheet, please register at our Worldwide Website at:

http://www.microchip.com

You can determine the version of a data sheet by examining its literature number found on the bottom outside corner of any page. The last character of the literature number is the version number, (e.g., DS30000000A is version A of document DS30000000).

Errata

An errata sheet, describing minor operational differences from the data sheet and recommended workarounds, may exist for current devices. As device/documentation issues become known to us, we will publish an errata sheet. The errata will specify the revision of silicon and revision of document to which it applies.

To determine if an errata sheet exists for a particular device, please check with one of the following:

- Microchip's Worldwide Website; **http://www.microchip.com**
- Your local Microchip sales office (see last page)

When contacting a sales office, please specify which device, revision of silicon and data sheet (include literature number) you are using.

Customer Notification System

Register on our website at **www.microchip.com** to receive the most current information on all of our products.

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 RA2/INT 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 Interrupt 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	RABIE(1)	TMR0IF(2)	INTF	RABIF
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:** INT External Interrupt Enable bit
1 = Enables the INT external interrupt
0 = Disables the INT external interrupt

bit 3 **RABIE:** PORTA or PORTB Change Interrupt Enable bit(1)
1 = Enables the PORTA or PORTB change interrupt
0 = Disables the PORTA or 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:** INT External Interrupt Flag bit
1 = The INT external interrupt occurred (must be cleared in software)
0 = The INT external interrupt did not occur

bit 0 **RABIF:** PORTA or PORTB Change Interrupt Flag bit
1 = When at least one of the PORTA or PORTB general purpose I/O pins changed state (must be cleared in software)
0 = None of the PORTA or 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.

12.1.3 SOFTWARE PROGRAMMABLE PRESCALER

A single software programmable prescaler is available for use with either Timer0 or the Watchdog Timer (WDT), but not both simultaneously. The prescaler assignment is controlled by the PSA bit of the OPTION_REG register. To assign the prescaler to Timer0, the PSA bit must be cleared to a '0'.

There are eight prescaler options for the Timer0 module ranging from 1:2 to 1:256. The prescale values are selectable via the PS<2:0> bits of the OPTION_REG register. In order to have a 1:1 prescaler value for the Timer0 module, the prescaler must be assigned to the WDT module.

The prescaler is not readable or writable. When assigned to the Timer0 module, all instructions writing to the TMR0 register will clear the prescaler.

Note:	When the prescaler is assigned to WDT, a CLRWDT instruction will clear the prescaler along with the WDT.

12.1.4 TIMER0 INTERRUPT

Timer0 will generate an interrupt when the TMR0 register overflows from FFh to 00h. The TMR0IF interrupt flag bit of the INTCON register is set every time the TMR0 register overflows, regardless of whether or not the Timer0 interrupt is enabled. The TMR0IF bit can only be cleared in software. The Timer0 interrupt enable is the TMR0IE bit of the INTCON register.

Note:	The Timer0 interrupt cannot wake the processor from Sleep since the timer is frozen during Sleep.

12.1.5 8-BIT COUNTER MODE SYNCHRONIZATION

When in 8-Bit Counter mode, the incrementing edge on the T0CKI pin must be synchronized to the instruction clock. Synchronization can be accomplished by sampling the prescaler output on the Q2 and Q4 cycles of the instruction clock. The high and low periods of the external clocking source must meet the timing requirements as shown in **Section 23.0 "Electrical Specifications"**.

14.2 Timer2 Control Register

REGISTER 14-1: T2CON: TIMER2 CONTROL REGISTER

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0
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 **Unimplemented:** Read as ‘0’

bit 6-3 **TOUTPS<3:0>:** Timer2 Output Postscaler Select bits

0000 = 1:1 Postscaler
0001 = 1:2 Postscaler
0010 = 1:3 Postscaler
0011 = 1:4 Postscaler
0100 = 1:5 Postscaler
0101 = 1:6 Postscaler
0110 = 1:7 Postscaler
0111 = 1:8 Postscaler
1000 = 1:9 Postscaler
1001 = 1:10 Postscaler
1010 = 1:11 Postscaler
1011 = 1:12 Postscaler
1100 = 1:13 Postscaler
1101 = 1:14 Postscaler
1110 = 1:15 Postscaler
1111 = 1:16 Postscaler

bit 2 **TMR2ON:** Timer2 On bit

1 = Timer2 is On
0 = Timer2 is Off

bit 1-0 **T2CKPS<1:0>:** Timer2 Clock Prescale Select bits

00 = Prescaler is 1
01 = Prescaler is 4
1x = Prescaler is 16

TABLE 14-1: SUMMARY OF REGISTERS ASSOCIATED WITH TIMER2

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	RABIE	TMR0IF	INTF	RABIF	37
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	38
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	39
PR2	Timer2 module Period Register								98
TMR2	Timer2 module Register								98
T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	99

Legend: x = unknown, u = unchanged, - = unimplemented read as ‘0’. Shaded cells are not used for Timer2 module.

15.0 CAPTURE/COMPARE/PWM (CCP) MODULE

The Capture/Compare/PWM module is a peripheral which allows the user to time and control different events. In Capture mode, the peripheral allows the timing of the duration of an event. The Compare mode allows the user to trigger an external event when a predetermined amount of time has expired. The PWM mode can generate a Pulse-Width Modulated signal of varying frequency and duty cycle.

The timer resources used by the module are shown in Table 15-1.

Additional information on CCP modules is available in the Application Note AN594, *"Using the CCP Modules"* (DS00594).

TABLE 15-1: CCP MODE – TIMER RESOURCES REQUIRED

CCP Mode	Timer Resource
Capture	Timer1
Compare	Timer1
PWM	Timer2

REGISTER 15-1: CCP1CON: CCP1 CONTROL REGISTER

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	DC1	B1	CCP1M3	CCP1M2	CCP1M1	CCP1M0
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 7-6 **Unimplemented:** Read as '0'

bit 5-4 **DC1:B1**: PWM Duty Cycle Least Significant bits
Capture mode:
Unused
Compare mode:
Unused
PWM mode:
These bits are the two LSbs of the PWM duty cycle. The eight MSbs are found in CCPR1L.

bit 3-0 **CCP1M<3:0>:** CCP mode Select bits
0000 = Capture/Compare/PWM off (resets CCP module)
0001 = Unused (reserved)
0010 = Compare mode, toggle output on match (CCP1IF bit of the PIRx register is set)
0011 = Unused (reserved)
0100 = Capture mode, every falling edge
0101 = Capture mode, every rising edge
0110 = Capture mode, every 4th rising edge
0111 = Capture mode, every 16th rising edge
1000 = Compare mode, set output on match (CCP1IF bit of the PIR1 register is set)
1001 = Compare mode, clear output on match (CCP1IF bit of the PIR1 register is set)
1010 = Compare mode, generate software interrupt on match (CCP1IF bit is set of the PIRx register, CCP1 pin is unaffected)
1011 = Compare mode, trigger special event (CCP1IF bit of the PIR1register is set, TMR1 is reset and A/D conversion is started if the ADC module is enabled. CCP1 pin is unaffected.)
11xx = PWM mode.

15.1 Capture Mode

In Capture mode, CCPR1H:CCPR1L captures the 16-bit value of the TMR1 register when an event occurs on pin CCP1. An event is defined as one of the following and is configured by the CCP1M<3:0> bits of the CCP1CON register:

- Every falling edge
- Every rising edge
- Every 4th rising edge
- Every 16th rising edge

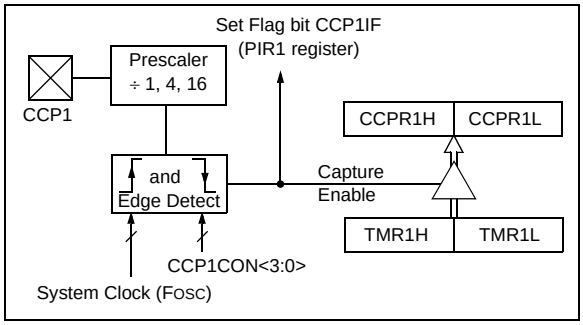
When a capture is made, the Interrupt Request Flag bit CCP1IF of the PIR1 register is set. The interrupt flag must be cleared in software. If another capture occurs before the value in the CCPR1H, CCPR1L register pair is read, the old captured value is overwritten by the new captured value (refer to Figure 15-1).

15.1.1 CCP1 PIN CONFIGURATION

In Capture mode, the CCP1 pin should be configured as an input by setting the associated TRIS control bit.

Note:	If the CCP1 pin is configured as an output, a write to the port can cause a CAPTURE condition.

FIGURE 15-1: CAPTURE MODE OPERATION BLOCK DIAGRAM



15.1.2 TIMER1 MODE SELECTION

Timer1 must be running in Timer mode or Synchronized Counter mode for the CCP module to use the capture feature. In Asynchronous Counter mode or when Timer1 is clocked at FOSC, the capture operation may not work.

Note:	Clocking Timer1 from the system clock (FOSC) should not be used in Capture mode. In order for Capture mode to recognize the trigger event on the CCP1 pin, Timer1 must be clocked from the Instruction Clock (FOSC/4) or from an external clock source.

15.1.3 SOFTWARE INTERRUPT

When the Capture mode is changed, a false capture interrupt may be generated. The user should keep the CCP1IE interrupt enable bit of the PIE1 register clear to avoid false interrupts. Additionally, the user should clear the CCP1IF interrupt flag bit of the PIR1 register following any change in Operating mode.

15.1.4 CCP PRESCALER

There are four prescaler settings specified by the CCP1M<3:0> bits of the CCP1CON register. Whenever the CCP module is turned off, or the CCP module is not in Capture mode, the prescaler counter is cleared. Any Reset will clear the prescaler counter.

Switching from one capture prescaler to another does not clear the prescaler and may generate a false interrupt. To avoid this unexpected operation, turn the module off by clearing the CCP1CON register before changing the prescaler (refer to Example 15-1).

EXAMPLE 15-1: CHANGING BETWEEN CAPTURE PRESCALERS

```
BANKSEL CCP1CON     ;Set Bank bits to point
                    ;to CCP1CON
CLRF    CCP1CON     ;Turn CCP module off
MOVLW   NEW_CAPT_PS ;Load the W reg with
                    ; the new prescaler
                    ; move value and CCP ON
MOVWF   CCP1CON     ;Load CCP1CON with this
                    ; value
```

15.1.5 CAPTURE DURING SLEEP

Capture mode depends upon the Timer1 module for proper operation. There are two options for driving the Timer1 module in Capture mode. It can be driven by the instruction clock (FOSC/4), or by an external clock source.

If Timer1 is clocked by FOSC/4, then Timer1 will not increment during Sleep. When the device wakes from Sleep, Timer1 will continue from its previous state.

If Timer1 is clocked by an external clock source, then Capture mode will operate as defined in **Section 15.1 “Capture Mode”**.

16.1.2.2 Receiving Data

The receiver data recovery circuit initiates character reception on the falling edge of the first bit. The first bit, also known as the Start bit, is always a zero. The data recovery circuit counts one-half bit time to the center of the Start bit and verifies that the bit is still a zero. If it is not a zero, then the data recovery circuit aborts character reception, without generating an error, and resumes looking for the falling edge of the Start bit. If the Start bit zero verification succeeds then the data recovery circuit counts a full-bit time to the center of the next bit. The bit is then sampled by a majority detect circuit and the resulting '0' or '1' is shifted into the RSR. This repeats until all data bits have been sampled and shifted into the RSR. One final bit time is measured and the level sampled. This is the Stop bit, which is always '1'. If the data recovery circuit samples a '0' in the Stop bit position, then a framing error is set for this character, otherwise the framing error is cleared for this character. Refer to **Section 16.1.2.4 "Receive Framing Error"** for more information on framing errors.

Immediately after all data bits and the Stop bit have been received, the character in the RSR is transferred to the AUSART receive FIFO and the RCIF interrupt flag bit of the PIR1 register is set. The top character in the FIFO is transferred out of the FIFO by reading the RCREG register.

Note:	If the receive FIFO is overrun, no additional characters will be received until the overrun condition is cleared. Refer to **Section 16.1.2.5 "Receive Overrun Error"** for more information on overrun errors.

16.1.2.3 Receive Interrupts

The RCIF interrupt flag bit of the PIR1 register is set whenever the AUSART receiver is enabled and there is an unread character in the receive FIFO. The RCIF interrupt flag bit is read-only, it cannot be set or cleared by software.

RCIF interrupts are enabled by setting all of the following bits:

- RCIE interrupt enable bit of the PIE1 register
- PEIE, Peripheral Interrupt Enable bit of the INTCON register
- GIE, Global Interrupt Enable bit of the INTCON register

The RCIF interrupt flag bit of the PIR1 register will be set when there is an unread character in the FIFO, regardless of the state of interrupt enable bits.

16.1.2.4 Receive Framing Error

Each character in the receive FIFO buffer has a corresponding framing error Status bit. A framing error indicates that a Stop bit was not seen at the expected time. The framing error status is accessed via the FERR bit of the RCSTA register. The FERR bit represents the status of the top unread character in the receive FIFO. Therefore, the FERR bit must be read before reading the RCREG.

The FERR bit is read-only and only applies to the top unread character in the receive FIFO. A framing error (FERR = 1) does not preclude reception of additional characters. It is not necessary to clear the FERR bit. Reading the next character from the FIFO buffer will advance the FIFO to the next character and the next corresponding framing error.

The FERR bit can be forced clear by clearing the SPEN bit of the RCSTA register which resets the AUSART. Clearing the CREN bit of the RCSTA register does not affect the FERR bit. A framing error by itself does not generate an interrupt.

Note:	If all receive characters in the receive FIFO have framing errors, repeated reads of the RCREG will not clear the FERR bit.

16.1.2.5 Receive Overrun Error

The receive FIFO buffer can hold two characters. An overrun error will be generated if a third character, in its entirety, is received before the FIFO is accessed. When this happens the OERR bit of the RCSTA register is set. The characters already in the FIFO buffer can be read but no additional characters will be received until the error is cleared. The error must be cleared by either clearing the CREN bit of the RCSTA register or by setting the AUSART by clearing the SPEN bit of the RCSTA register.

16.1.2.6 Receiving 9-bit Characters

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

REGISTER 16-1: TXSTA: TRANSMIT STATUS AND CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R-1	R/W-0
CSRC	TX9	TXEN(1)	SYNC	—	BRGH	TRMT	TX9D
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 **CSRC:** Clock Source Select bit
Asynchronous mode:
Don't care
Synchronous mode:
1 = Master mode (clock generated internally from BRG)
0 = Slave mode (clock from external source)

bit 6 **TX9:** 9-bit Transmit Enable bit
1 = Selects 9-bit transmission
0 = Selects 8-bit transmission

bit 5 **TXEN:** Transmit Enable bit(1)
1 = Transmit enabled
0 = Transmit disabled

bit 4 **SYNC:** AUSART mode Select bit
1 = Synchronous mode
0 = Asynchronous mode

bit 3 **Unimplemented:** Read as '0'

bit 2 **BRGH:** High Baud Rate Select bit
Asynchronous mode:
1 = High speed
0 = Low speed
Synchronous mode:
Unused in this mode

bit 1 **TRMT:** Transmit Shift Register Status bit
1 = TSR empty
0 = TSR full

bit 0 **TX9D:** Ninth bit of Transmit Data
Can be address/data bit or a parity bit.

Note 1: SREN/CREN overrides TXEN in Synchronous mode.

TABLE 16-5: BAUD RATES FOR ASYNCHRONOUS MODES

BAUD RATE	SYNC = 0, BRGH = 0											
	FOSC = 16.0000 MHz			FOSC = 11.0592 MHz			FOSC = 8.000 MHz			FOSC = 4.000 MHz		
	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)
300	—	—	—	—	—	—	—	—	—	300	0.16	207
1200	1201	0.08	207	1200	0.00	143	1202	0.16	103	1202	0.16	51
2400	2403	0.16	103	2400	0.00	71	2404	0.16	51	2404	0.16	25
9600	9615	0.16	25	9600	0.00	17	9615	0.16	12	—	—	—
10417	10416	-0.01	23	10165	-2.42	16	10417	0.00	11	10417	0.00	5
19.2k	19.23k	0.16	12	19.20k	0.00	8	—	—	—	—	—	—
57.6k	—	—	—	57.60k	0.00	2	—	—	—	—	—	—
115.2k	—	—	—	—	—	—	—	—	—	—	—	—

BAUD RATE	SYNC = 0, BRGH = 0					
	FOSC = 3.6864 MHz			FOSC = 1.000 MHz		
	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)
300	300	0.00	191	300	0.16	51
1200	1200	0.00	47	1202	0.16	12
2400	2400	0.00	23	—	—	—
9600	9600	0.00	5	—	—	—
10417	—	—	—	—	—	—
19.2k	19.20k	0.00	2	—	—	—
57.6k	57.60k	0.00	0	—	—	—
115.2k	—	—	—	—	—	—

FIGURE 17-4: SPI MODE WAVEFORM (SLAVE MODE WITH CKE = 0)

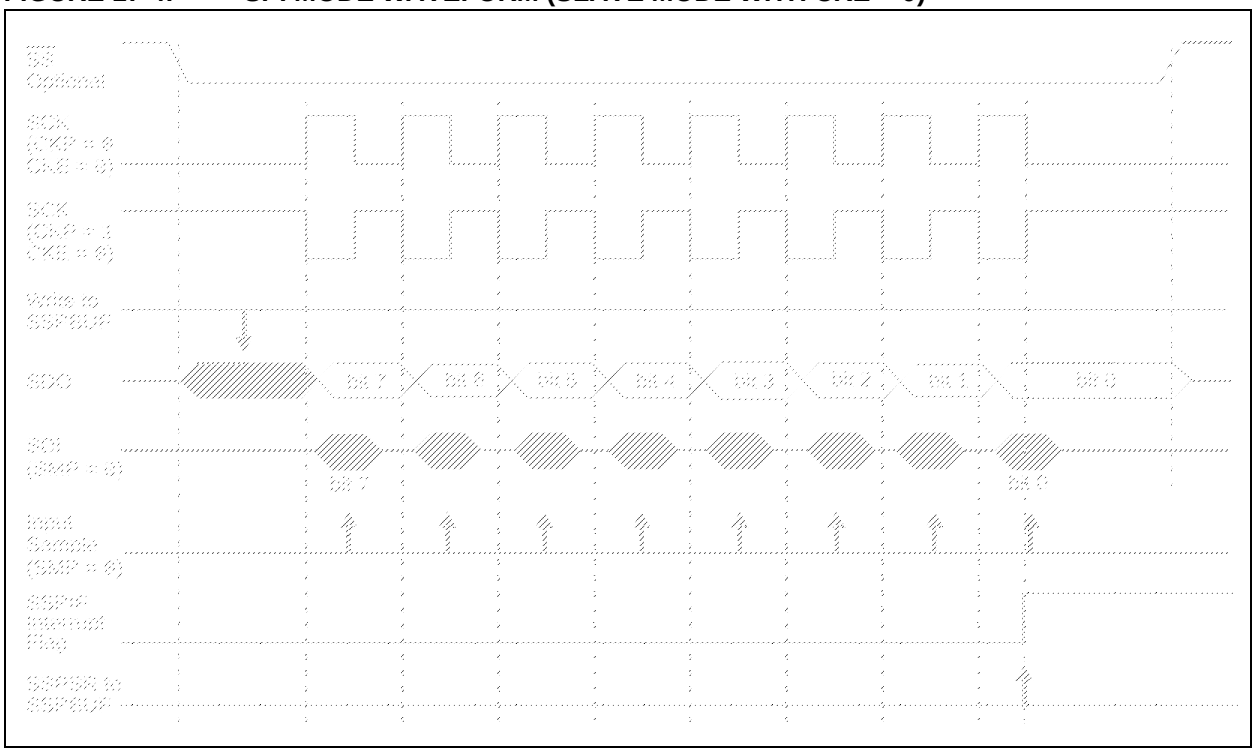
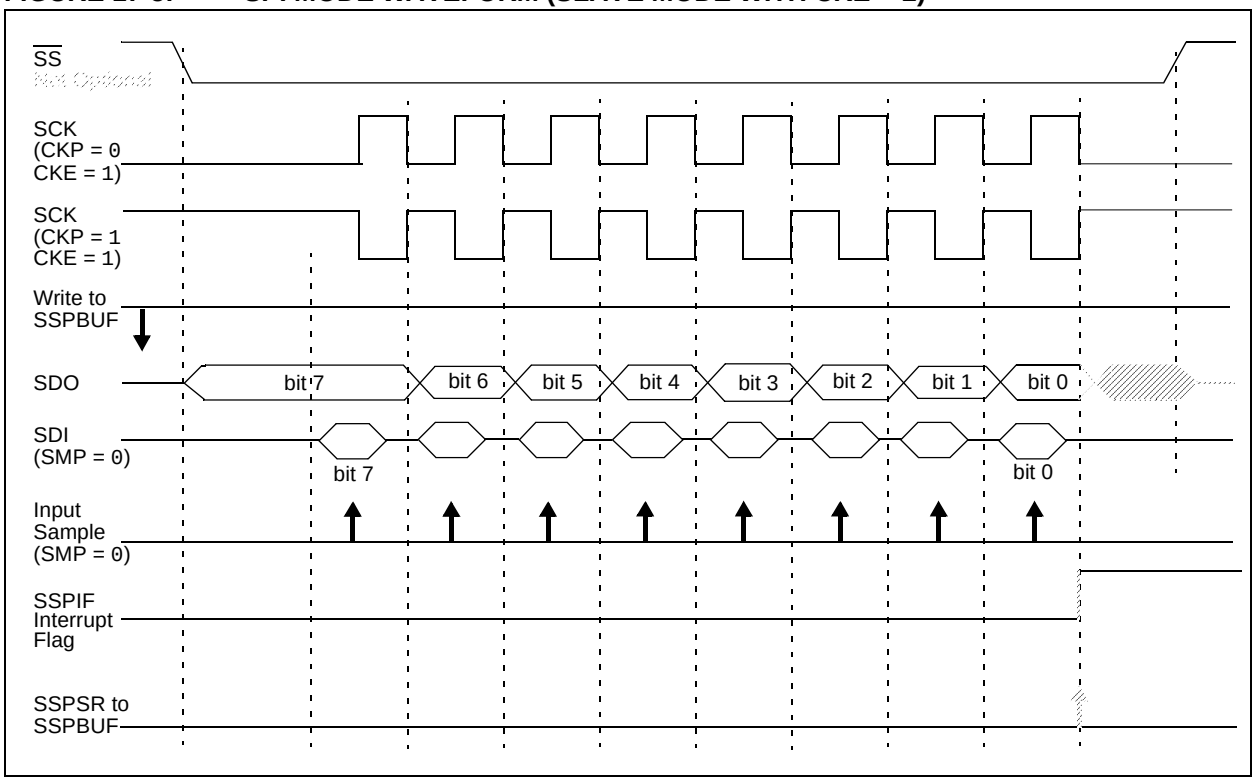


FIGURE 17-5: SPI MODE WAVEFORM (SLAVE MODE WITH CKE = 1)



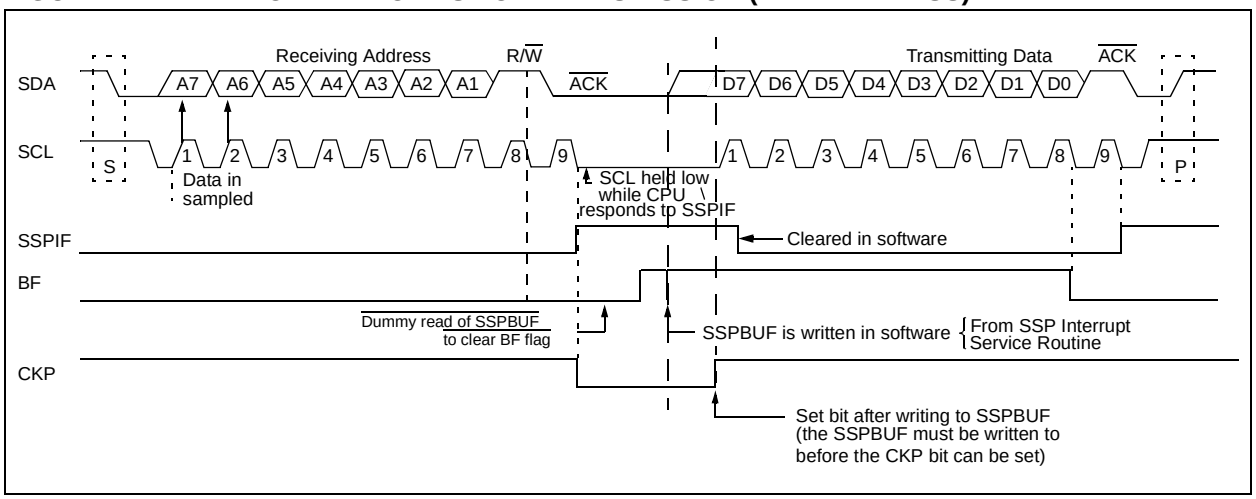
17.2.6 TRANSMISSION

When the R/$\overline{W}$ bit of the received address byte is set and an address match occurs, the R/$\overline{W}$ bit of the SSPSTAT register is set and the slave will respond to the master by reading out data. After the address match, an $\overline{ACK}$ pulse is generated by the slave hardware and the SCL pin is held low (clock is automatically stretched) until the slave is ready to respond. See **Section 17.2.7 "Clock Stretching"**. The data the slave will transmit must be loaded into the SSPBUF register, which sets the BF bit. The SCL line is released by setting the CKP bit of the SSPCON register.

An SSP interrupt is generated for each transferred data byte. The SSPIF flag bit of the PIR1 register initiates an SSP interrupt, and must be cleared by software before the next byte is transmitted. The BF bit of the SSPSTAT register is cleared on the falling edge of the eighth received clock pulse. The SSPIF flag bit is set on the falling edge of the ninth clock pulse.

Following the eighth falling clock edge, control of the SDA line is released back to the master so that the master can acknowledge or not acknowledge the response. If the master sends a not acknowledge, the slave's transmission is complete and the slave must monitor for the next Start condition. If the master acknowledges, control of the bus is returned to the slave to transmit another byte of data. Just as with the previous byte, the clock is stretched by the slave, data must be loaded into the SSPBUF and CKP must be set to release the clock line (SCL).

FIGURE 17-12: I^{2}C WAVEFORMS FOR TRANSMISSION (7-BIT ADDRESS)



REGISTER 18-2: PMDATH: PROGRAM MEMORY DATA HIGH REGISTER

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	PMD13	PMD12	PMD11	PMD10	PMD9	PMD8
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 7-6 **Unimplemented:** Read as ‘0’

bit 5-0 **PMD<13:8>:** The value of the program memory word pointed to by PMADRH and PMADRL after a program memory read command.

REGISTER 18-3: PMDATL: PROGRAM MEMORY DATA LOW REGISTER

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
PMD7	PMD6	PMD5	PMD4	PMD3	PMD2	PMD1	PMD0
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 7-0 **PMD<7:0>:** The value of the program memory word pointed to by PMADRH and PMADRL after a program memory read command.

REGISTER 18-4: PMADRH: PROGRAM MEMORY ADDRESS HIGH REGISTER

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	PMA12	PMA11	PMA10	PMA9	PMA8
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 7-5 **Unimplemented:** Read as ‘0’

bit 4-0 **PMA<12:8>:** Program Memory Read Address bits

22.11 Demonstration/Development Boards, Evaluation Kits, and Starter Kits

A wide variety of demonstration, development and evaluation boards for various PIC MCUs and dsPIC DSCs allows quick application development on fully functional systems. Most boards include prototyping areas for adding custom circuitry and provide application firmware and source code for examination and modification.

The boards support a variety of features, including LEDs, temperature sensors, switches, speakers, RS-232 interfaces, LCD displays, potentiometers and additional EEPROM memory.

The demonstration and development boards can be used in teaching environments, for prototyping custom circuits and for learning about various microcontroller applications.

In addition to the PICDEM™ and dsPICDEM™ demonstration/development board series of circuits, Microchip has a line of evaluation kits and demonstration software for analog filter design, KEELOQ® security ICs, CAN, IrDA®, PowerSmart battery management, SEEVAL® evaluation system, Sigma-Delta ADC, flow rate sensing, plus many more.

Also available are starter kits that contain everything needed to experience the specified device. This usually includes a single application and debug capability, all on one board.

Check the Microchip web page (www.microchip.com) for the complete list of demonstration, development and evaluation kits.

22.12 Third-Party Development Tools

Microchip also offers a great collection of tools from third-party vendors. These tools are carefully selected to offer good value and unique functionality.

- Device Programmers and Gang Programmers from companies, such as SoftLog and CCS
- Software Tools from companies, such as Gimpel and Trace Systems
- Protocol Analyzers from companies, such as Saleae and Total Phase
- Demonstration Boards from companies, such as MikroElektronika, Digilent® and Olimex
- Embedded Ethernet Solutions from companies, such as EZ Web Lynx, WIZnet and IPLogika®

23.1 DC Characteristics: PIC16(L)F720/721-I/E (Industrial, Extended)

PIC16LF720/721			Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended				
PIC16F720/721			**Standard Operating Conditions (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended				
Param. No.	**Sym.**	**Characteristic**	**Min.**	**Typ†**	**Max.**	**Units**	**Conditions**
D001	VDD	**Supply Voltage**					
		PIC16LF720/721	1.8	—	3.6	V	FOSC ≤ 16 MHz: HFINTOSC, EC
D001		**PIC16F720/721**	1.8	—	5.5	V	FOSC ≤ 16 MHz: HFINTOSC, EC
D002*	VDR	**RAM Data Retention Voltage(1)**					
		PIC16LF720/721	1.5	—	—	V	Device in Sleep mode
D002*		**PIC16F720/721**	1.7	—	—	V	Device in Sleep mode
	VPOR*	**Power-on Reset Release Voltage**	—	1.6	—	V	
	VPORR*	**Power-on Reset Rearm Voltage**					
		PIC16LF720/721	—	0.9	—	V	
		PIC16F720/721	—	1.5	—	V	
D003	VFVR	**Fixed Voltage Reference Voltage, Initial Accuracy**	-8	—	6	%	VFVR = 1.024V, VDD ≥ 2.5V VFVR = 2.048V, VDD ≥ 2.5V VFVR = 4.096V, VDD ≥ 4.75V;
D004*	SVDD	**VDD Rise Rate** to ensure internal Power-on Reset signal	0.05	—	—	V/ms	See **Section 3.2 "Power-on Reset (POR)"** for details.

* These parameters are characterized but not tested.

† Data in "Typ" column is at 3.3V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: This is the limit to which VDD can be lowered in Sleep mode without losing RAM data.

23.4 DC Characteristics: PIC16(L)F720/721-I/E (Continued)

DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended				
Param. No.	**Sym.**	**Characteristic**	**Min.**	**Typ†**	**Max.**	**Units**	**Conditions**
D133	TPEW	Erase/Write cycle time	—		2.8	ms	Temperature during programming: 10°C ≤ TA ≤ 40°C
D134*	TRETD	Characteristic Retention	—	40	—	Year	Provided no other specifications are violated
D135	EHEFC	High-Endurance Flash Cell	100K	—	—	E/W	0°C to +60°C Lower byte, Last 128 Addresses in Flash memory

* These parameters are characterized but not tested.

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

Note 1: Negative current is defined as current sourced by the pin.

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

24.0 DC AND AC CHARACTERISTICS GRAPHS AND CHARTS

FIGURE 24-1: PIC16F720/721 MAX IDD vs. FOSC OVER VDD, EC MODE

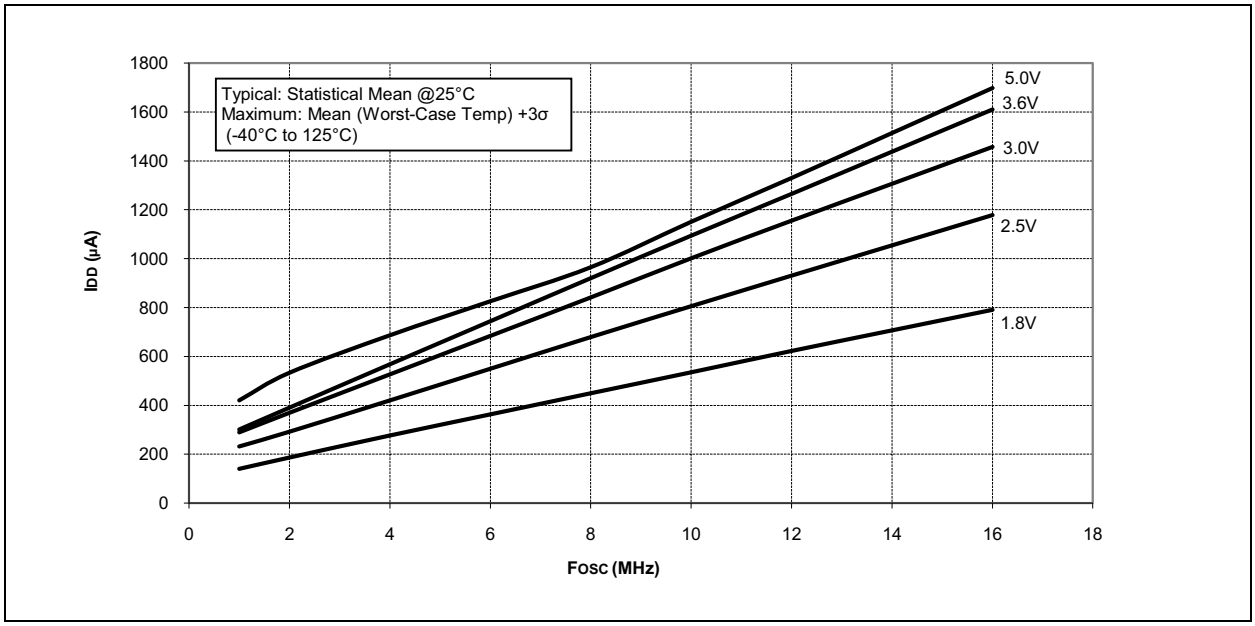


FIGURE 24-2: PIC16F720/721 TYPICAL IDD vs. FOSC OVER VDD, EC MODE

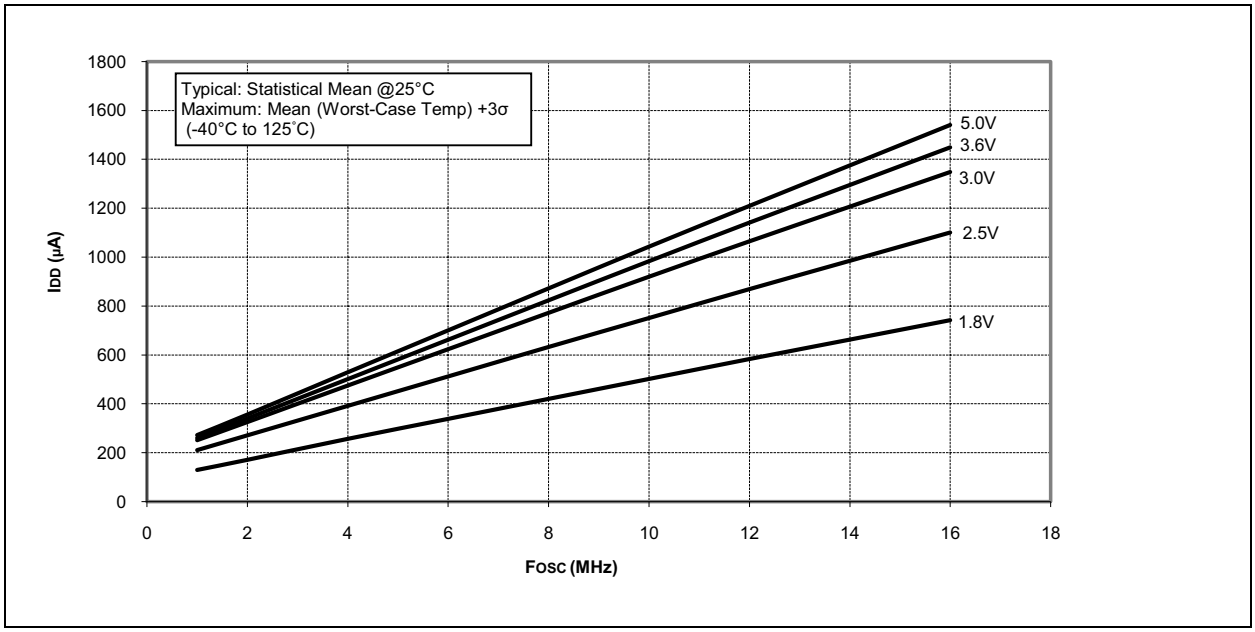


FIGURE 24-7: PIC16LF720/721 MAX. IDD vs. FOSC OVER VDD, MFINTOSC

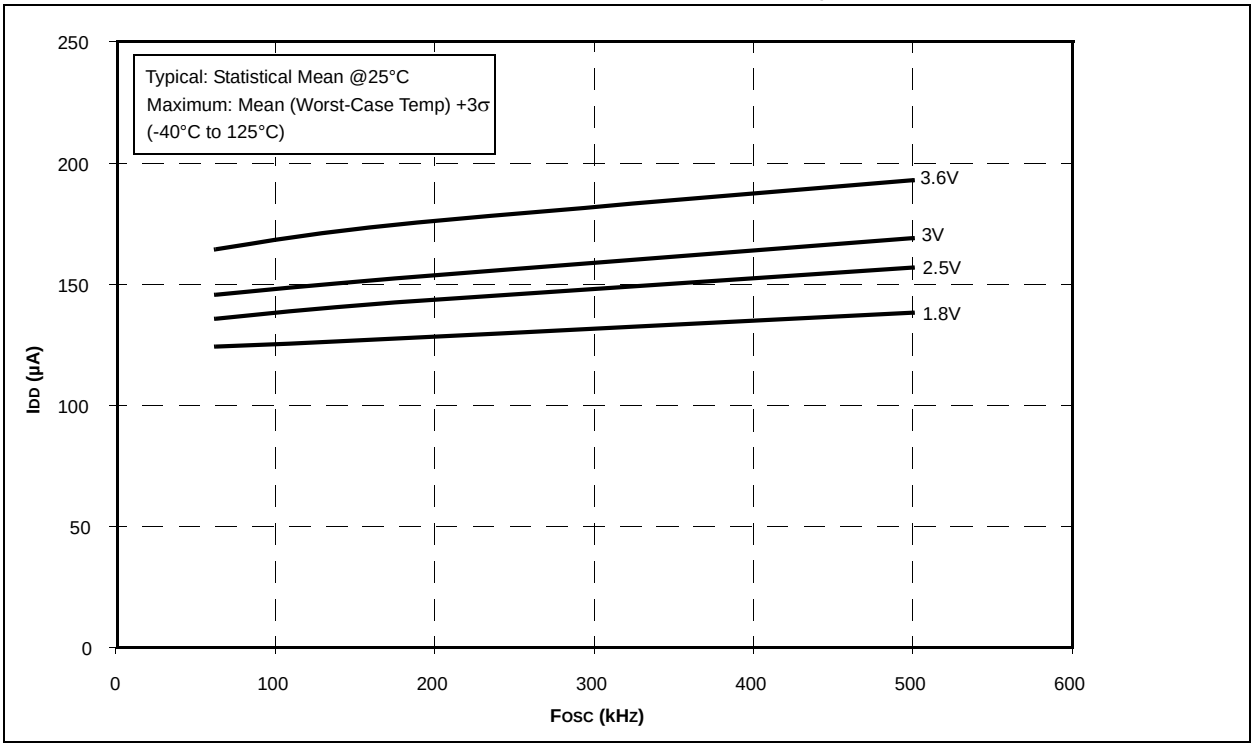


FIGURE 24-8: PIC16LF720/721 TYPICAL IDD vs. FOSC OVER VDD, MFINTOSC

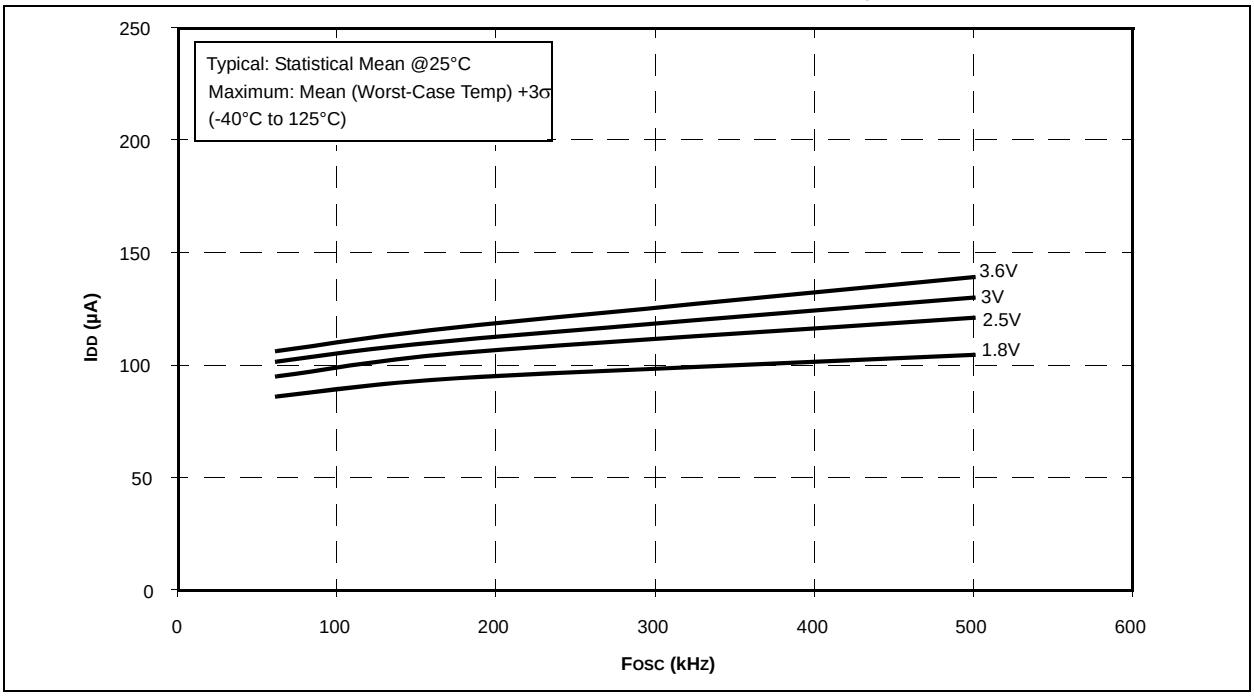


FIGURE 24-11: PIC16LF720/721 MAX. IDD vs. FOSC OVER VDD, HFINTOSC

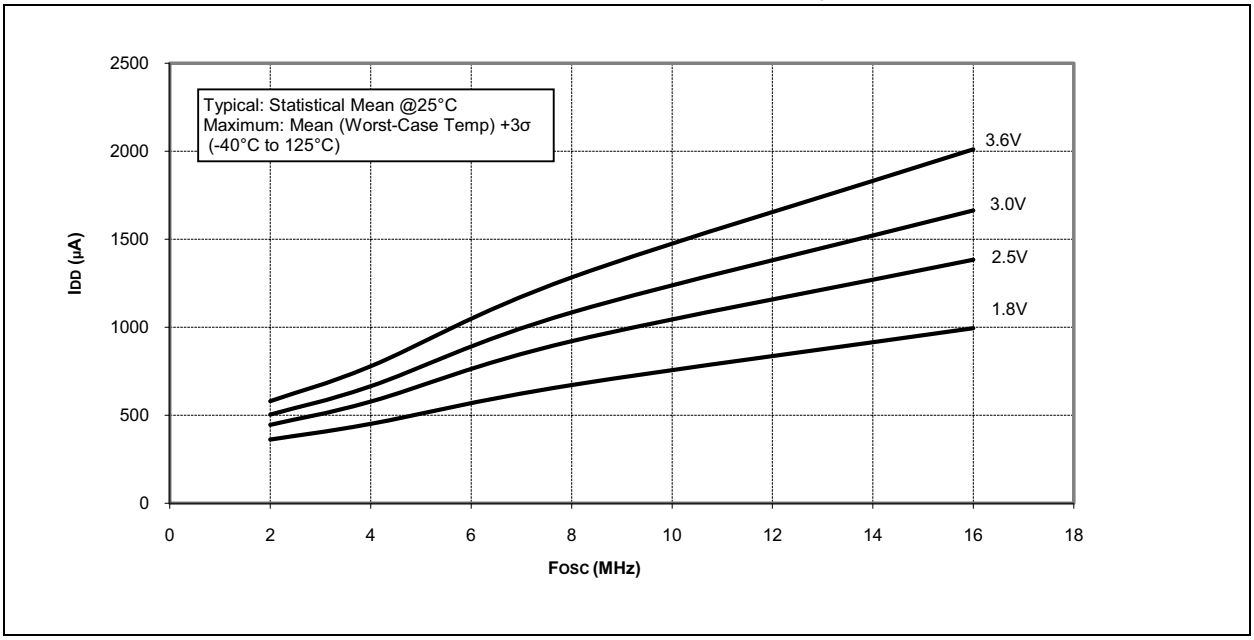


FIGURE 24-12: PIC16LF720/721 TYPICAL IDD vs. FOSC OVER VDD, HFINTOSC

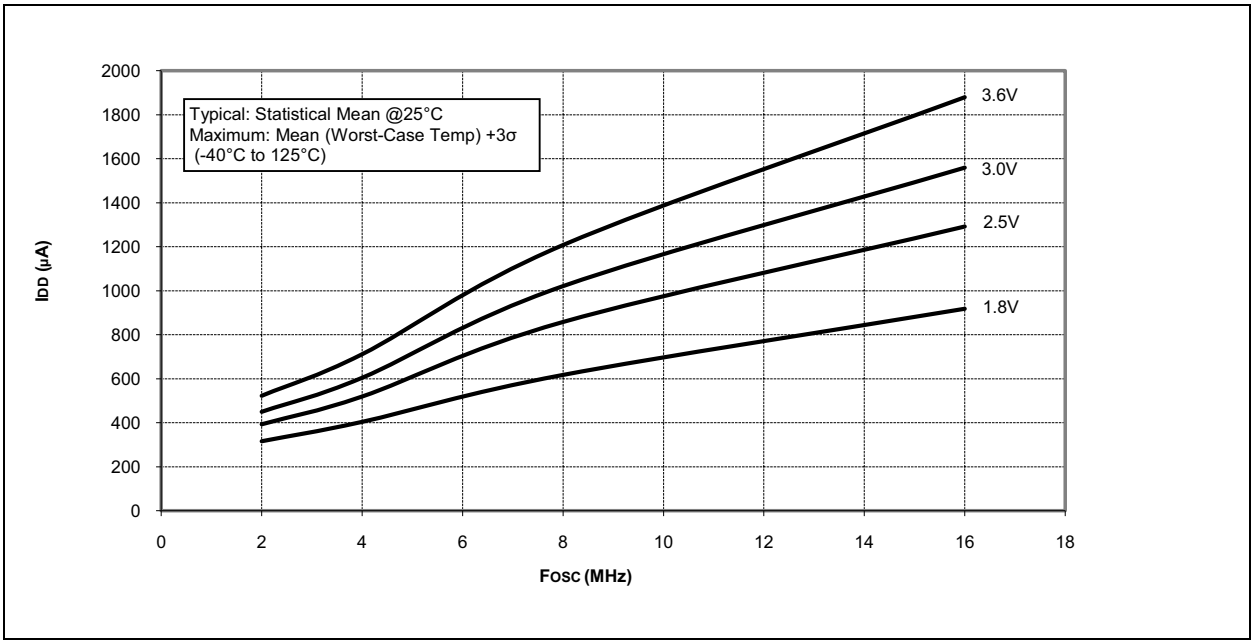
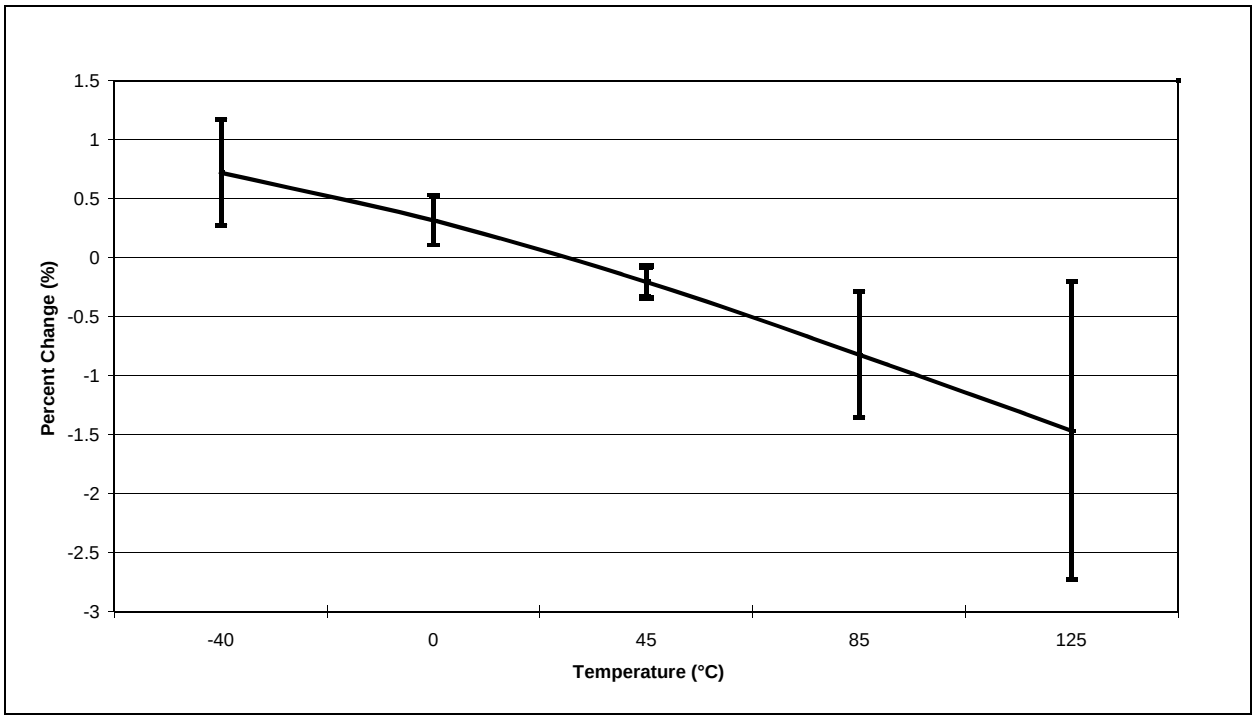


FIGURE 24-36: TYPICAL FVR CHANGE VS. TEMPERATURE NORMALIZED AT 25°C



NOTES: