



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	25MHz
Connectivity	UART/USART
Peripherals	POR, PWM, WDT
Number of I/O	33
Program Memory Size	8KB (4K x 16)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	454 x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 6V
Data Converters	-
Oscillator Type	External
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	44-QFP
Supplier Device Package	44-MQFP (10x10)
Purchase URL	<a href="https://www.e-xfl.com/product-detail/microchip-technology/pic17c43-25e-pq">https://www.e-xfl.com/product-detail/microchip-technology/pic17c43-25e-pq</a>

Table of Contents

1.0 Overview ..... 5

2.0 PIC17C4X Device Varieties ..... 7

3.0 Architectural Overview ..... 9

4.0 Reset ..... 15

5.0 Interrupts ..... 21

6.0 Memory Organization ..... 29

7.0 Table Reads and Table Writes ..... 43

8.0 Hardware Multiplier ..... 49

9.0 I/O Ports ..... 53

10.0 Overview of Timer Resources ..... 65

11.0 Timer0 ..... 67

12.0 Timer1, Timer2, Timer3, PWMs and Captures ..... 71

13.0 Universal Synchronous Asynchronous Receiver Transmitter (USART) Module ..... 83

14.0 Special Features of the CPU ..... 99

15.0 Instruction Set Summary ..... 107

16.0 Development Support ..... 143

17.0 PIC17C42 Electrical Characteristics ..... 147

18.0 PIC17C42 DC and AC Characteristics ..... 163

19.0 PIC17CR42/42A/43/R43/44 Electrical Characteristics ..... 175

20.0 PIC17CR42/42A/43/R43/44 DC and AC Characteristics ..... 193

21.0 Packaging Information ..... 205

Appendix A: Modifications ..... 211

Appendix B: Compatibility ..... 211

Appendix C: What's New ..... 212

Appendix D: What's Changed ..... 212

Appendix E: PIC16/17 Microcontrollers ..... 213

Appendix F: Errata for PIC17C42 Silicon ..... 223

Index ..... 226

PIC17C4X Product Identification System ..... 237

For register and module descriptions in this data sheet, device legends show which devices apply to those sections. For example, the legend below shows that some features of only the PIC17C43, PIC17CR43, PIC17C44 are described in this section.

Applicable Devices				
42	R42	42A	43	R43 44

To Our Valued Customers

We constantly strive to improve the quality of all our products and documentation. We have spent an exceptional amount of time to ensure that these documents are correct. However, we realize that we may have missed a few things. If you find any information that is missing or appears in error from the previous version of the PIC17C4X Data Sheet (Literature Number DS30412B), please use the reader response form in the back of this data sheet to inform us. We appreciate your assistance in making this a better document.

To assist you in the use of this document, Appendix C contains a list of new information in this data sheet, while Appendix D contains information that has changed

**TABLE 4-4: INITIALIZATION CONDITIONS FOR SPECIAL FUNCTION REGISTERS**

Register	Address	Power-on Reset	MCLR Reset WDT Reset	Wake-up from SLEEP through interrupt
<b>Unbanked</b>				
INDF0	00h	0000 0000	0000 0000	0000 0000
FSR0	01h	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCL	02h	0000h	0000h	PC + 1 <sup>(2)</sup>
PCLATH	03h	0000 0000	0000 0000	uuuu uuuu
ALUSTA	04h	1111 xxxx	1111 uuuu	1111 uuuu
T0STA	05h	0000 000-	0000 000-	0000 000-
CPUSTA <sup>(3)</sup>	06h	--11 11--	--11 qq--	--uu qq--
INTSTA	07h	0000 0000	0000 0000	uuuu uuuu <sup>(1)</sup>
INDF1	08h	0000 0000	0000 0000	uuuu uuuu
FSR1	09h	xxxx xxxx	uuuu uuuu	uuuu uuuu
WREG	0Ah	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR0L	0Bh	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR0H	0Ch	xxxx xxxx	uuuu uuuu	uuuu uuuu
TBLPTRL <sup>(4)</sup>	0Dh	xxxx xxxx	uuuu uuuu	uuuu uuuu
TBLPTRH <sup>(4)</sup>	0Eh	xxxx xxxx	uuuu uuuu	uuuu uuuu
TBLPTRL <sup>(5)</sup>	0Dh	0000 0000	0000 0000	uuuu uuuu
TBLPTRH <sup>(5)</sup>	0Eh	0000 0000	0000 0000	uuuu uuuu
BSR	0Fh	0000 0000	0000 0000	uuuu uuuu
<b>Bank 0</b>				
PORTA	10h	0-xx xxxx	0-uu uuuu	uuuu uuuu
DDRB	11h	1111 1111	1111 1111	uuuu uuuu
PORTB	12h	xxxx xxxx	uuuu uuuu	uuuu uuuu
RCSTA	13h	0000 -00x	0000 -00u	uuuu -uuu
RCREG	14h	xxxx xxxx	uuuu uuuu	uuuu uuuu
TXSTA	15h	0000 --1x	0000 --1u	uuuu --uu
TXREG	16h	xxxx xxxx	uuuu uuuu	uuuu uuuu
SPBRG	17h	xxxx xxxx	uuuu uuuu	uuuu uuuu
<b>Bank 1</b>				
DDRC	10h	1111 1111	1111 1111	uuuu uuuu
PORTC	11h	xxxx xxxx	uuuu uuuu	uuuu uuuu
DDRD	12h	1111 1111	1111 1111	uuuu uuuu
PORTD	13h	xxxx xxxx	uuuu uuuu	uuuu uuuu
DDRE	14h	---- -111	---- -111	---- -uuu
PORTE	15h	---- -xxx	---- -uuu	---- -uuu
PIR	16h	0000 0010	0000 0010	uuuu uuuu <sup>(1)</sup>
PIE	17h	0000 0000	0000 0000	uuuu uuuu

Legend: u = unchanged, x = unknown, - = unimplemented read as '0', q = value depends on condition.

Note 1: One or more bits in INTSTA, PIR will be affected (to cause wake-up).

2: When the wake-up is due to an interrupt and the GLINTD bit is cleared, the PC is loaded with the interrupt vector.

3: See Table 4-3 for reset value of specific condition.

4: Only applies to the PIC17C42.

5: Does not apply to the PIC17C42.

## 5.2 Peripheral Interrupt Enable Register (PIE)

This register contains the individual flag bits for the Peripheral interrupts.

**FIGURE 5-3: PIE REGISTER (ADDRESS: 17h, BANK 1)**

R/W - 0	R/W - 0	R/W - 0	R/W - 0	R/W - 0	R/W - 0	R/W - 0	R/W - 0
RBIE	TMR3IE	TMR2IE	TMR1IE	CA2IE	CA1IE	TXIE	RCIE
bit7							bit0
<p>bit 7: <b>RBIE:</b> PORTB Interrupt on Change Enable bit  1 = Enable PORTB interrupt on change  0 = Disable PORTB interrupt on change</p> <p>bit 6: <b>TMR3IE:</b> Timer3 Interrupt Enable bit  1 = Enable Timer3 interrupt  0 = Disable Timer3 interrupt</p> <p>bit 5: <b>TMR2IE:</b> Timer2 Interrupt Enable bit  1 = Enable Timer2 interrupt  0 = Disable Timer2 interrupt</p> <p>bit 4: <b>TMR1IE:</b> Timer1 Interrupt Enable bit  1 = Enable Timer1 interrupt  0 = Disable Timer1 interrupt</p> <p>bit 3: <b>CA2IE:</b> Capture2 Interrupt Enable bit  1 = Enable Capture interrupt on RB1/CAP2 pin  0 = Disable Capture interrupt on RB1/CAP2 pin</p> <p>bit 2: <b>CA1IE:</b> Capture1 Interrupt Enable bit  1 = Enable Capture interrupt on RB2/CAP1 pin  0 = Disable Capture interrupt on RB2/CAP1 pin</p> <p>bit 1: <b>TXIE:</b> USART Transmit Interrupt Enable bit  1 = Enable Transmit buffer empty interrupt  0 = Disable Transmit buffer empty interrupt</p> <p>bit 0: <b>RCIE:</b> USART Receive Interrupt Enable bit  1 = Enable Receive buffer full interrupt  0 = Disable Receive buffer full interrupt</p>							

R = Readable bit  
W = Writable bit  
-n = Value at POR reset

## 5.3 Peripheral Interrupt Request Register (PIR)

This register contains the individual flag bits for the peripheral interrupts.

**Note:** These bits will be set by the specified condition, even if the corresponding interrupt enable bit is cleared (interrupt disabled), or the GLINTD bit is set (all interrupts disabled). Before enabling an interrupt, the user may wish to clear the interrupt flag to ensure that the program does not immediately branch to the peripheral interrupt service routine.

**FIGURE 5-4: PIR REGISTER (ADDRESS: 16h, BANK 1)**

R/W - 0	R/W - 0	R/W - 0	R/W - 0	R/W - 0	R/W - 0	R - 1	R - 0
RBIF	TMR3IF	TMR2IF	TMR1IF	CA2IF	CA1IF	TXIF	RCIF
bit7							bit0
<p><b>bit 7: RBIF:</b> PORTB Interrupt on Change Flag bit  1 = One of the PORTB inputs changed (Software must end the mismatch condition)  0 = None of the PORTB inputs have changed</p> <p><b>bit 6: TMR3IF:</b> Timer3 Interrupt Flag bit  If Capture1 is enabled (<math>CA1/\overline{PR3} = 1</math>)  1 = Timer3 overflowed  0 = Timer3 did not overflow  If Capture1 is disabled (<math>CA1/\overline{PR3} = 0</math>)  1 = Timer3 value has rolled over to 0000h from equalling the period register (PR3H:PR3L) value  0 = Timer3 value has not rolled over to 0000h from equalling the period register (PR3H:PR3L) value</p> <p><b>bit 5: TMR2IF:</b> Timer2 Interrupt Flag bit  1 = Timer2 value has rolled over to 0000h from equalling the period register (PR2) value  0 = Timer2 value has not rolled over to 0000h from equalling the period register (PR2) value</p> <p><b>bit 4: TMR1IF:</b> Timer1 Interrupt Flag bit  If Timer1 is in 8-bit mode (<math>T16 = 0</math>)  1 = Timer1 value has rolled over to 0000h from equalling the period register (PR) value  0 = Timer1 value has not rolled over to 0000h from equalling the period register (PR2) value  If Timer1 is in 16-bit mode (<math>T16 = 1</math>)  1 = TMR1:TMR2 value has rolled over to 0000h from equalling the period register (PR1:PR2) value  0 = TMR1:TMR2 value has not rolled over to 0000h from equalling the period register (PR1:PR2) value</p> <p><b>bit 3: CA2IF:</b> Capture2 Interrupt Flag bit  1 = Capture event occurred on RB1/CAP2 pin  0 = Capture event did not occur on RB1/CAP2 pin</p> <p><b>bit 2: CA1IF:</b> Capture1 Interrupt Flag bit  1 = Capture event occurred on RB0/CAP1 pin  0 = Capture event did not occur on RB0/CAP1 pin</p> <p><b>bit 1: TXIF:</b> USART Transmit Interrupt Flag bit  1 = Transmit buffer is empty  0 = Transmit buffer is full</p> <p><b>bit 0: RCIF:</b> USART Receive Interrupt Flag bit  1 = Receive buffer is full  0 = Receive buffer is empty</p>							
<p>R = Readable bit  W = Writable bit  -n = Value at POR reset</p>							

## 6.2.2.2 CPU STATUS REGISTER (CPUSTA)

The CPUSTA register contains the status and control bits for the CPU. This register is used to globally enable/disable interrupts. If only a specific interrupt is desired to be enabled/disabled, please refer to the INTerrupt Status (INTSTA) register and the Peripheral Interrupt Enable (PIE) register. This register also indicates if the stack is available and contains the Power-down ( $\overline{PD}$ ) and Time-out ( $\overline{TO}$ ) bits. The  $\overline{TO}$ ,  $\overline{PD}$ , and STKAV bits are not writable. These bits are set and cleared according to device logic. Therefore, the result of an instruction with the CPUSTA register as destination may be different than intended.

**FIGURE 6-8: CPUSTA REGISTER (ADDRESS: 06h, UNBANKED)**

U - 0	U - 0	R - 1	R/W - 1	R - 1	R - 1	U - 0	U - 0
—	—	STKAV	GLINTD	$\overline{TO}$	$\overline{PD}$	—	—
bit7							bit0

R = Readable bit  
W = Writable bit  
U = Unimplemented bit, Read as '0'  
- n = Value at POR reset

bit 7-6: **Unimplemented:** Read as '0'

bit 5: **STKAV:** Stack Available bit  
This bit indicates that the 4-bit stack pointer value is Fh, or has rolled over from Fh → 0h (stack overflow).  
1 = Stack is available  
0 = Stack is full, or a stack overflow may have occurred (Once this bit has been cleared by a stack overflow, only a device reset will set this bit)

bit 4: **GLINTD:** Global Interrupt Disable bit  
This bit disables all interrupts. When enabling interrupts, only the sources with their enable bits set can cause an interrupt.  
1 = Disable all interrupts  
0 = Enables all un-masked interrupts

bit 3:  **$\overline{TO}$ :** WDT Time-out Status bit  
1 = After power-up or by a CLRWDT instruction  
0 = A Watchdog Timer time-out occurred

bit 2:  **$\overline{PD}$ :** Power-down Status bit  
1 = After power-up or by the CLRWDT instruction  
0 = By execution of the SLEEP instruction

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

## 6.4.1 INDIRECT ADDRESSING REGISTERS

The PIC17C4X has four registers for indirect addressing. These registers are:

- INDF0 and FSR0
- INDF1 and FSR1

Registers INDF0 and INDF1 are not physically implemented. Reading or writing to these registers activates indirect addressing, with the value in the corresponding FSR register being the address of the data. The FSR is an 8-bit register and allows addressing anywhere in the 256-byte data memory address range. For banked memory, the bank of memory accessed is specified by the value in the BSR.

If file INDF0 (or INDF1) itself is read indirectly via an FSR, all '0's are read (Zero bit is set). Similarly, if INDF0 (or INDF1) is written to indirectly, the operation will be equivalent to a NOP, and the status bits are not affected.

## 6.4.2 INDIRECT ADDRESSING OPERATION

The indirect addressing capability has been enhanced over that of the PIC16CXX family. There are two control bits associated with each FSR register. These two bits configure the FSR register to:

- Auto-decrement the value (address) in the FSR after an indirect access
- Auto-increment the value (address) in the FSR after an indirect access
- No change to the value (address) in the FSR after an indirect access

These control bits are located in the ALUSTA register. The FSR1 register is controlled by the FS3:FS2 bits and FSR0 is controlled by the FS1:FS0 bits.

When using the auto-increment or auto-decrement features, the effect on the FSR is not reflected in the ALUSTA register. For example, if the indirect address causes the FSR to equal '0', the Z bit will not be set.

If the FSR register contains a value of 0h, an indirect read will read 0h (Zero bit is set) while an indirect write will be equivalent to a NOP (status bits are not affected).

Indirect addressing allows single cycle data transfers within the entire data space. This is possible with the use of the MOVPPF and MOVFP instructions, where either 'p' or 'f' is specified as INDF0 (or INDF1).

If the source or destination of the indirect address is in banked memory, the location accessed will be determined by the value in the BSR.

A simple program to clear RAM from 20h - FFh is shown in Example 6-1.

### EXAMPLE 6-1: INDIRECT ADDRESSING

```
MOVLW    0x20      ;
MOVWF     FSR0      ; FSR0 = 20h
BCF       ALUSTA, FS1 ; Increment FSR
BSF       ALUSTA, FS0 ; after access
BCF       ALUSTA, C   ; C = 0
MOVLW     END_RAM + 1 ;
LP CLRf     INDF0      ; Addr(FSR) = 0
CPFSEQ    FSR0        ; FSR0 = END_RAM+1?
GOTO      LP          ; NO, clear next
:          ; YES, All RAM is
:          ; cleared
```

## 6.5 Table Pointer (TBLPTRL and TBLPTRH)

File registers TBLPTRL and TBLPTRH form a 16-bit pointer to address the 64K program memory space. The table pointer is used by instructions TABLWT and TABLRD.

The TABLRD and the TABLWT instructions allow transfer of data between program and data space. The table pointer serves as the 16-bit address of the data word within the program memory. For a more complete description of these registers and the operation of Table Reads and Table Writes, see Section 7.0.

## 6.6 Table Latch (TBLATH, TBLATL)

The table latch (TBLAT) is a 16-bit register, with TBLATH and TBLATL referring to the high and low bytes of the register. It is not mapped into data or program memory. The table latch is used as a temporary holding latch during data transfer between program and data memory (see descriptions of instructions TABLRD, TABLWT, TLRD and TLWT). For a more complete description of these registers and the operation of Table Reads and Table Writes, see Section 7.0.

## 9.0 I/O PORTS

The PIC17C4X devices have five I/O ports, PORTA through PORTE. PORTB through PORTE have a corresponding Data Direction Register (DDR), which is used to configure the port pins as inputs or outputs. These five ports are made up of 33 I/O pins. Some of these ports pins are multiplexed with alternate functions.

PORTC, PORTD, and PORTE are multiplexed with the system bus. These pins are configured as the system bus when the device's configuration bits are selected to Microprocessor or Extended Microcontroller modes. In the two other microcontroller modes, these pins are general purpose I/O.

PORTA and PORTB are multiplexed with the peripheral features of the device. These peripheral features are:

- Timer modules
- Capture module
- PWM module
- USART/SCI module
- External Interrupt pin

When some of these peripheral modules are turned on, the port pin will automatically configure to the alternate function. The modules that do this are:

- PWM module
- USART/SCI module

When a pin is automatically configured as an output by a peripheral module, the pins data direction (DDR) bit is unknown. After disabling the peripheral module, the user should re-initialize the DDR bit to the desired configuration.

The other peripheral modules (which require an input) must have their data direction bit configured appropriately.

**Note:** A pin that is a peripheral input, can be configured as an output (DDRx<y> is cleared). The peripheral events will be determined by the action output on the port pin.

## 9.1 PORTA Register

PORTA is a 6-bit wide latch. PORTA does not have a corresponding Data Direction Register (DDR).

Reading PORTA reads the status of the pins.

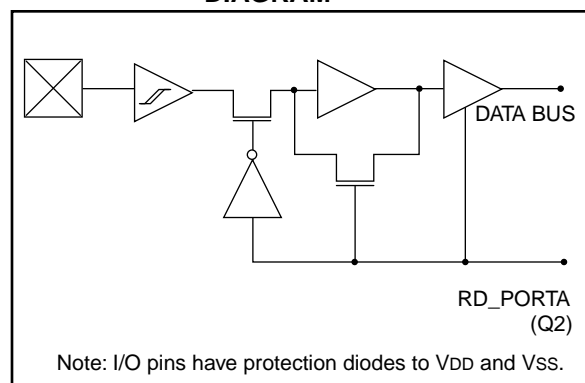
The RA1 pin is multiplexed with TMR0 clock input, and RA4 and RA5 are multiplexed with the USART functions. The control of RA4 and RA5 as outputs is automatically configured by the USART module.

### 9.1.1 USING RA2, RA3 AS OUTPUTS

The RA2 and RA3 pins are open drain outputs. To use the RA2 or the RA3 pin(s) as output(s), simply write to the PORTA register the desired value. A '0' will cause the pin to drive low, while a '1' will cause the pin to float (hi-impedance). An external pull-up resistor should be used to pull the pin high. Writes to PORTA will not affect the other pins.

**Note:** When using the RA2 or RA3 pin(s) as output(s), read-modify-write instructions (such as BCF, BSF, BTG) on PORTA are not recommended. Such operations read the port pins, do the desired operation, and then write this value to the data latch. This may inadvertently cause the RA2 or RA3 pins to switch from input to output (or vice-versa). It is recommended to use a shadow register for PORTA. Do the bit operations on this shadow register and then move it to PORTA.

**FIGURE 9-1: RA0 AND RA1 BLOCK DIAGRAM**





**TABLE 9-5: PORTC FUNCTIONS**

Name	Bit	Buffer Type	Function
RC0/AD0	bit0	TTL	Input/Output or system bus address/data pin.
RC1/AD1	bit1	TTL	Input/Output or system bus address/data pin.
RC2/AD2	bit2	TTL	Input/Output or system bus address/data pin.
RC3/AD3	bit3	TTL	Input/Output or system bus address/data pin.
RC4/AD4	bit4	TTL	Input/Output or system bus address/data pin.
RC5/AD5	bit5	TTL	Input/Output or system bus address/data pin.
RC6/AD6	bit6	TTL	Input/Output or system bus address/data pin.
RC7/AD7	bit7	TTL	Input/Output or system bus address/data pin.

Legend: TTL = TTL input.

**TABLE 9-6: REGISTERS/BITS ASSOCIATED WITH PORTC**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on Power-on Reset	Value on all other resets (Note1)
11h, Bank 1	PORTC	RC7/ AD7	RC6/ AD6	RC5/ AD5	RC4/ AD4	RC3/ AD3	RC2/ AD2	RC1/ AD1	RC0/ AD0	xxxx xxxx	uuuu uuuu
10h, Bank 1	DDRC	Data direction register for PORTC								1111 1111	1111 1111

Legend: x = unknown, u = unchanged.

Note 1: Other (non power-up) resets include: external reset through  $\overline{\text{MCLR}}$  and the Watchdog Timer Reset.

## 9.4 PORTD and DDRD Registers

PORTD is an 8-bit bi-directional port. The corresponding data direction register is DDRD. A '1' in DDRD configures the corresponding port pin as an input. A '0' in the DDRC register configures the corresponding port pin as an output. Reading PORTD reads the status of the pins, whereas writing to it will write to the port latch. PORTD is multiplexed with the system bus. When operating as the system bus, PORTD is the high order byte of the address/data bus (AD15:AD8). The timing for the system bus is shown in the Electrical Characteristics section.

**Note:** This port is configured as the system bus when the device's configuration bits are selected to Microprocessor or Extended Microcontroller modes. In the two other microcontroller modes, this port is a general purpose I/O.

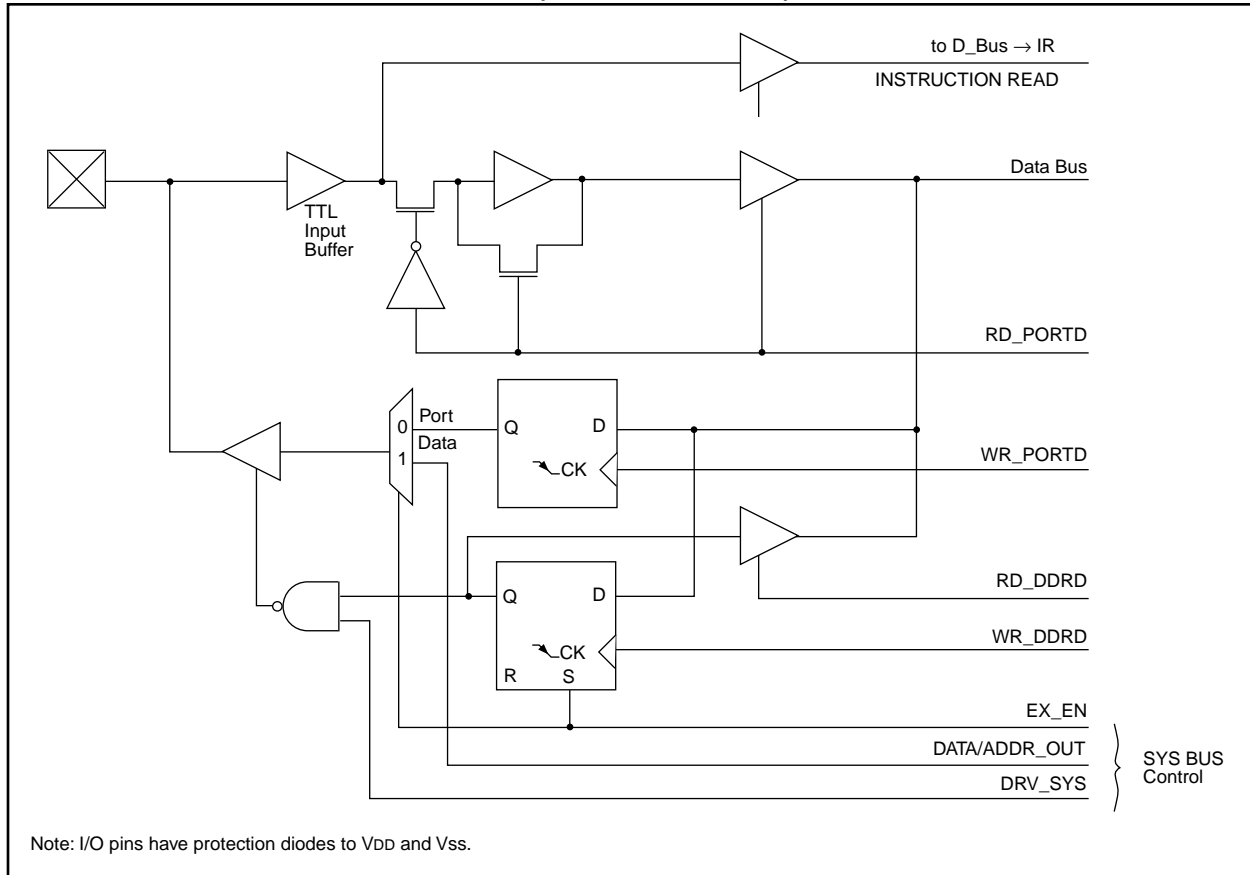
Example 9-3 shows the instruction sequence to initialize PORTD. The Bank Select Register (BSR) must be selected to Bank 1 for the port to be initialized.

### EXAMPLE 9-3: INITIALIZING PORTD

```

MOVLB 1           ; Select Bank 1
CLRF  PORTD       ; Initialize PORTD data
                  ; latches before setting
                  ; the data direction
                  ; register
MOVLW 0xCF        ; Value used to initialize
                  ; data direction
MOVWF DDRD        ; Set RD<3:0> as inputs
                  ; RD<5:4> as outputs
                  ; RD<7:6> as inputs
    
```

**FIGURE 9-7: PORTD BLOCK DIAGRAM (IN I/O PORT MODE)**



## 10.0 OVERVIEW OF TIMER RESOURCES

The PIC17C4X has four timer modules. Each module can generate an interrupt to indicate that an event has occurred. These timers are called:

- Timer0 - 16-bit timer with programmable 8-bit prescaler
- Timer1 - 8-bit timer
- Timer2 - 8-bit timer
- Timer3 - 16-bit timer

For enhanced time-base functionality, two input Captures and two Pulse Width Modulation (PWM) outputs are possible. The PWMs use the TMR1 and TMR2 resources and the input Captures use the TMR3 resource.

### 10.1 Timer0 Overview

The Timer0 module is a simple 16-bit overflow counter. The clock source can be either the internal system clock ( $F_{osc}/4$ ) or an external clock.

The Timer0 module also has a programmable prescaler option. The PS3:PS0 bits (T0STA<4:1>) determine the prescaler value. TMR0 can increment at the following rates: 1:1, 1:2, 1:4, 1:8, 1:16, 1:32, 1:64, 1:128, 1:256.

When Timer0's clock source is an external clock, the Timer0 module can be selected to increment on either the rising or falling edge.

Synchronization of the external clock occurs after the prescaler. When the prescaler is used, the external clock frequency may be higher than the device's frequency. The maximum frequency is 50 MHz, given the high and low time requirements of the clock.

### 10.2 Timer1 Overview

The Timer1 module is an 8-bit timer/counter with an 8-bit period register (PR1). When the TMR1 value rolls over from the period match value to 0h, the TMR1IF flag is set, and an interrupt will be generated when enabled. In counter mode, the clock comes from the RB4/TCLK12 pin, which can also be selected to be the clock for the Timer2 module.

TMR1 can be concatenated to TMR2 to form a 16-bit timer. The TMR1 register is the LSB and TMR2 is the MSB. When in the 16-bit timer mode, there is a corresponding 16-bit period register (PR2:PR1). When the TMR2:TMR1 value rolls over from the period match value to 0h, the TMR1IF flag is set, and an interrupt will be generated when enabled.

### 10.3 Timer2 Overview

The TMR2 module is an 8-bit timer/counter with an 8-bit period register (PR2). When the TMR2 value rolls over from the period match value to 0h, the TMR2IF flag is set, and an interrupt will be generated when enabled. In counter mode, the clock comes from the RB4/TCLK12 pin, which can also be selected to be the clock for the TMR1 module.

TMR1 can be concatenated to TMR2 to form a 16-bit timer. The TMR2 register is the MSB and TMR1 is the LSB. When in the 16-bit timer mode, there is a corresponding 16-bit period register (PR2:PR1). When the TMR2:TMR1 value rolls over from the period match value to 0h, the TMR1IF flag is set, and an interrupt will be generated when enabled.

### 10.4 Timer3 Overview

The Timer3 module is a 16-bit timer/counter with a 16-bit period register. When the TMR3H:TMR3L value rolls over to 0h, the TMR3IF bit is set and an interrupt will be generated when enabled. In counter mode, the clock comes from the RB5/TCLK3 pin.

When operating in the dual capture mode, the period registers become the second 16-bit capture register.

### 10.5 Role of the Timer/Counters

The timer modules are general purpose, but have dedicated resources associated with them. Timer1 and Timer2 are the time-bases for the two Pulse Width Modulation (PWM) outputs, while Timer3 is the time-base for the two input captures.

# PIC17C4X

---

NOTES:

## 14.0 SPECIAL FEATURES OF THE CPU

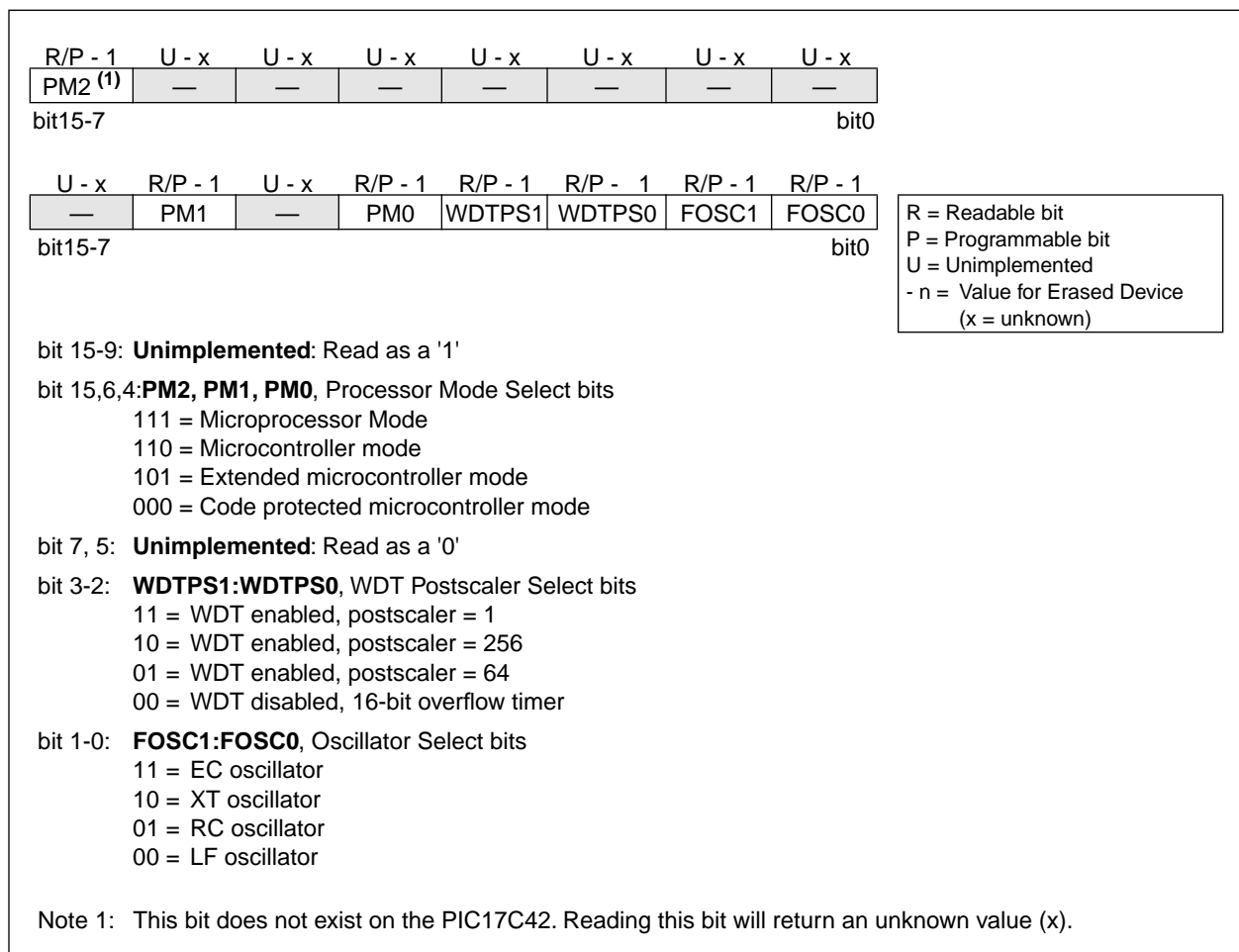
What sets a microcontroller apart from other processors are special circuits to deal with the needs of real time applications. The PIC17CXX family has a host of such features intended to maximize system reliability, minimize cost through elimination of external components, provide power saving operating modes and offer code protection. These are:

- OSC selection
- Reset
  - Power-on Reset (POR)
  - Power-up Timer (PWRT)
  - Oscillator Start-up Timer (OST)
- Interrupts
- Watchdog Timer (WDT)
- SLEEP
- Code protection

The PIC17CXX has a Watchdog Timer which can be shut off only through EPROM bits. It runs off its own RC oscillator for added reliability. There are two timers that offer necessary delays on power-up. One is the Oscillator Start-up Timer (OST), intended to keep the chip in RESET until the crystal oscillator is stable. The other is the Power-up Timer (PWRT), which provides a fixed delay of 96 ms (nominal) on power-up only, designed to keep the part in RESET while the power supply stabilizes. With these two timers on-chip, most applications need no external reset circuitry.

The SLEEP mode is designed to offer a very low current power-down mode. The user can wake from SLEEP through external reset, Watchdog Timer Reset or through an interrupt. Several oscillator options are also made available to allow the part to fit the application. The RC oscillator option saves system cost while the LF crystal option saves power. Configuration bits are used to select various options. This configuration word has the format shown in Figure 14-1.

**FIGURE 14-1: CONFIGURATION WORD**



XORLW		Exclusive OR Literal with WREG							
Syntax:	[ <i>label</i> ] XORLW k								
Operands:	0 ≤ k ≤ 255								
Operation:	(WREG) .XOR. k → (WREG)								
Status Affected:	Z								
Encoding:	<table><tr><td>1011</td><td>0100</td><td>kkkk</td><td>kkkk</td></tr></table>					1011	0100	kkkk	kkkk
1011	0100	kkkk	kkkk						
Description:	The contents of WREG are XOR'ed with the 8-bit literal 'k'. The result is placed in WREG.								
Words:	1								
Cycles:	1								
Q Cycle Activity:									
	Q1	Q2	Q3	Q4					
	Decode	Read literal 'k'	Execute	Write to WREG					

**Example:** XORLW 0xAF

Before Instruction  
WREG = 0xB5

After Instruction  
WREG = 0x1A

XORWF		Exclusive OR WREG with f						
Syntax:	[ <i>label</i> ] XORWF f,d							
Operands:	0 ≤ f ≤ 255 d ∈ [0,1]							
Operation:	(WREG) .XOR. (f) → (dest)							
Status Affected:	Z							
Encoding:	<table><tr><td>0000</td><td>110d</td><td>ffff</td><td>ffff</td></tr></table>				0000	110d	ffff	ffff
0000	110d	ffff	ffff					
Description:	Exclusive OR the contents of WREG with register 'f'. If 'd' is 0 the result is stored in WREG. If 'd' is 1 the result is stored back in the register 'f'.							
Words:	1							
Cycles:	1							
Q Cycle Activity:								
	Q1	Q2	Q3	Q4				
	Decode	Read register 'f'	Execute	Write to destination				

**Example:** XORWF REG, 1

Before Instruction  
REG = 0xAF  
WREG = 0xB5

After Instruction  
REG = 0x1A  
WREG = 0xB5

## 17.0 PIC17C42 ELECTRICAL CHARACTERISTICS

### Absolute Maximum Ratings †

Ambient temperature under bias .....	-55 to +125°C
Storage temperature .....	-65°C to +150°C
Voltage on VDD with respect to VSS .....	0 to +7.5V
Voltage on $\overline{\text{MCLR}}$ with respect to VSS (Note 2) .....	-0.6V to +14V
Voltage on RA2 and RA3 with respect to VSS.....	-0.6V to +12V
Voltage on all other pins with respect to VSS .....	-0.6V to VDD + 0.6V
Total power dissipation (Note 1).....	1.0W
Maximum current out of VSS pin(s) - Total .....	250 mA
Maximum current into VDD pin(s) - Total .....	200 mA
Input clamp current, I <sub>IK</sub> (V <sub>I</sub> < 0 or V <sub>I</sub> > VDD) .....	±20 mA
Output clamp current, I <sub>OK</sub> (V <sub>O</sub> < 0 or V <sub>O</sub> > VDD).....	±20 mA
Maximum output current sunk by any I/O pin (except RA2 and RA3).....	35 mA
Maximum output current sunk by RA2 or RA3 pins .....	60 mA
Maximum output current sourced by any I/O pin .....	20 mA
Maximum current sunk by PORTA and PORTB (combined).....	150 mA
Maximum current sourced by PORTA and PORTB (combined).....	100 mA
Maximum current sunk by PORTC, PORTD and PORTE (combined).....	150 mA
Maximum current sourced by PORTC, PORTD and PORTE (combined).....	100 mA

**Note 1:** Power dissipation is calculated as follows:  $P_{dis} = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD} - V_{OH}) \times I_{OH}\} + \sum (V_{OL} \times I_{OL})$

**Note 2:** Voltage spikes below VSS at the  $\overline{\text{MCLR}}$  pin, inducing currents greater than 80 mA, may cause latch-up. Thus, a series resistor of 50-100Ω should be used when applying a "low" level to the  $\overline{\text{MCLR}}$  pin rather than pulling this pin directly to VSS.

† NOTICE: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

# PIC17C4X

Applicable Devices 42 R42 42A 43 R43 44

FIGURE 18-2: TYPICAL RC OSCILLATOR FREQUENCY vs. VDD

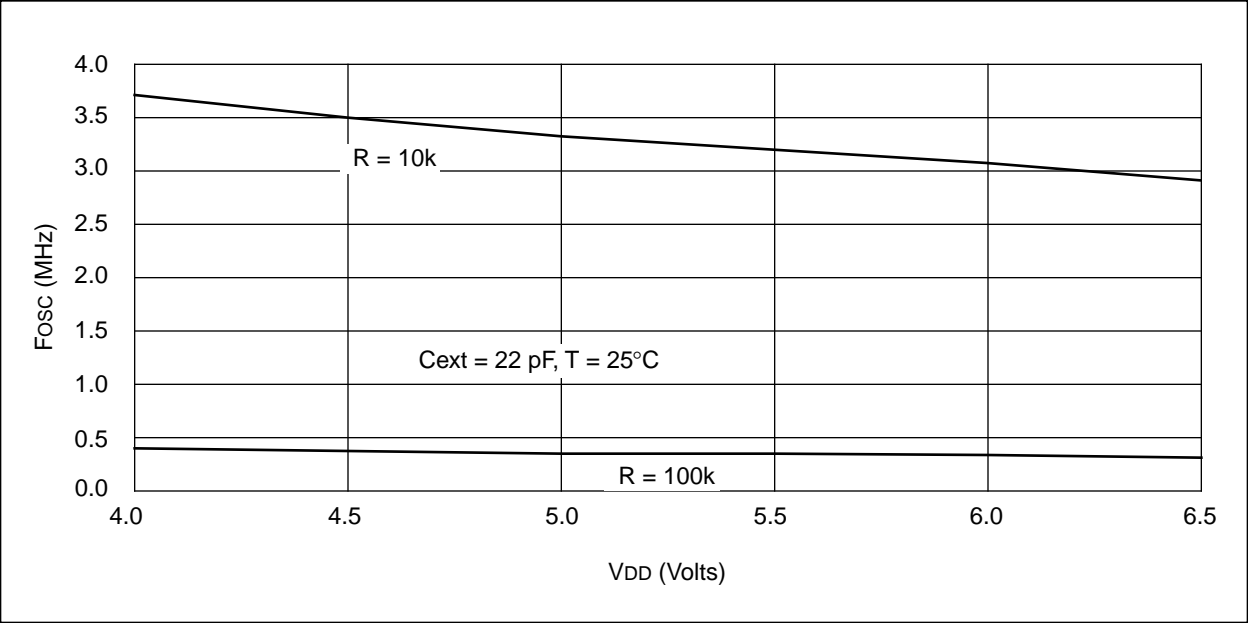
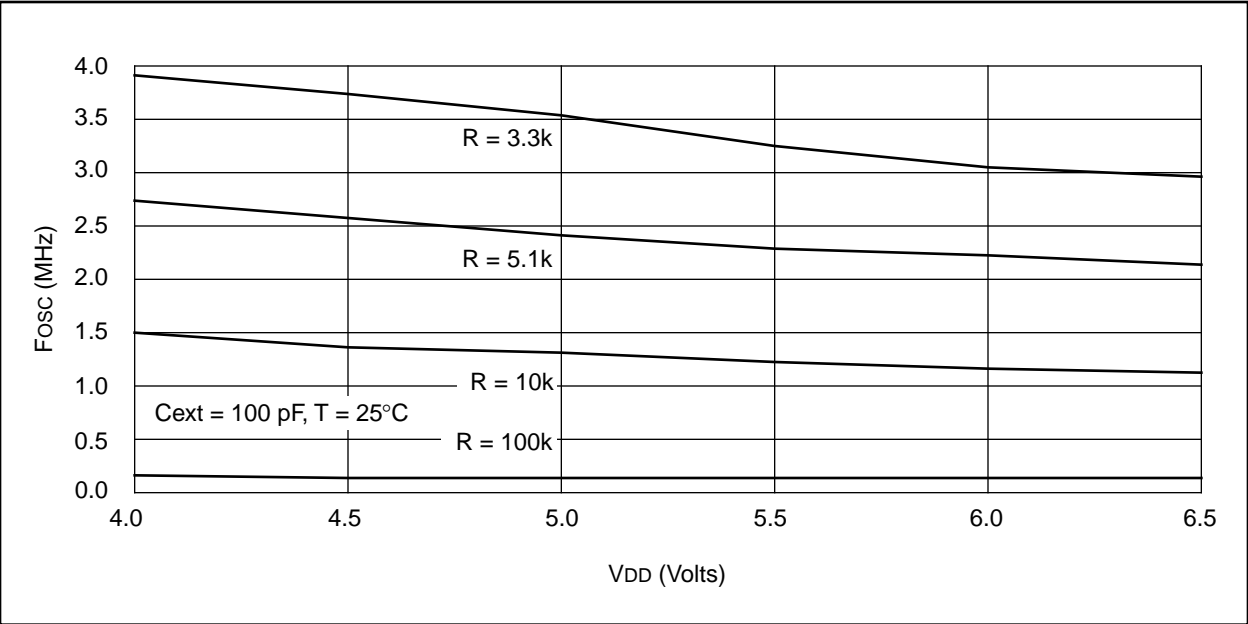
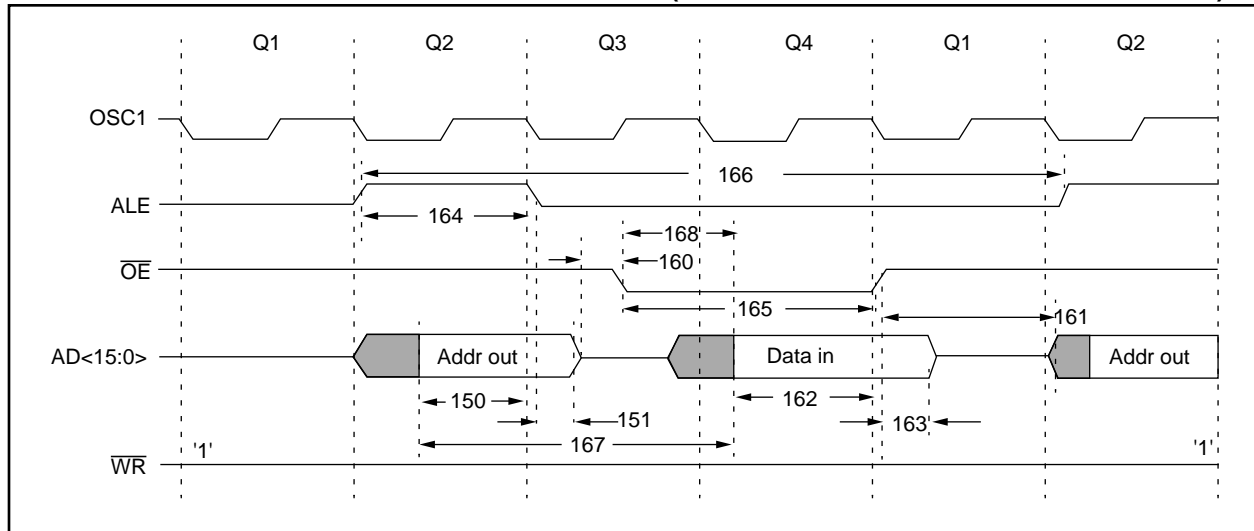


FIGURE 18-3: TYPICAL RC OSCILLATOR FREQUENCY vs. VDD





**FIGURE 19-12: MEMORY INTERFACE READ TIMING (NOT SUPPORTED IN PIC17LC4X DEVICES)**



**TABLE 19-12: MEMORY INTERFACE READ REQUIREMENTS (NOT SUPPORTED IN PIC17LC4X DEVICES)**

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
150	TadV2alL	AD15:AD0 (address) valid to ALE↓ (address setup time)	0.25Tcy - 10	—	—	ns	
151	TalL2adI	ALE↓ to address out invalid (address hold time)	5*	—	—	ns	
160	TadZ2oeL	AD15:AD0 hi-impedance to $\overline{OE}$ ↓	0*	—	—	ns	
161	ToeH2adD	$\overline{OE}$ ↑ to AD15:AD0 driven	0.25Tcy - 15	—	—	ns	
162	TadV2oeH	Data in valid before $\overline{OE}$ ↑ (data setup time)	35	—	—	ns	
163	ToeH2adI	$\overline{OE}$ ↑ to data in invalid (data hold time)	0	—	—	ns	
164	TalH	ALE pulse width	—	0.25Tcy §	—	ns	
165	ToeL	$\overline{OE}$ pulse width	0.5Tcy - 35 §	—	—	ns	
166	TalH2alH	ALE↑ to ALE↑ (cycle time)	—	Tcy §	—	ns	
167	Tacc	Address access time	—	—	0.75Tcy - 30	ns	
168	Toe	Output enable access time ( $\overline{OE}$ low to Data Valid)	—	—	0.5Tcy - 45	ns	

\* These parameters are characterized but not tested.

† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

§ This specification ensured by design.

## 20.0 PIC17CR42/42A/43/R43/44 DC AND AC CHARACTERISTICS

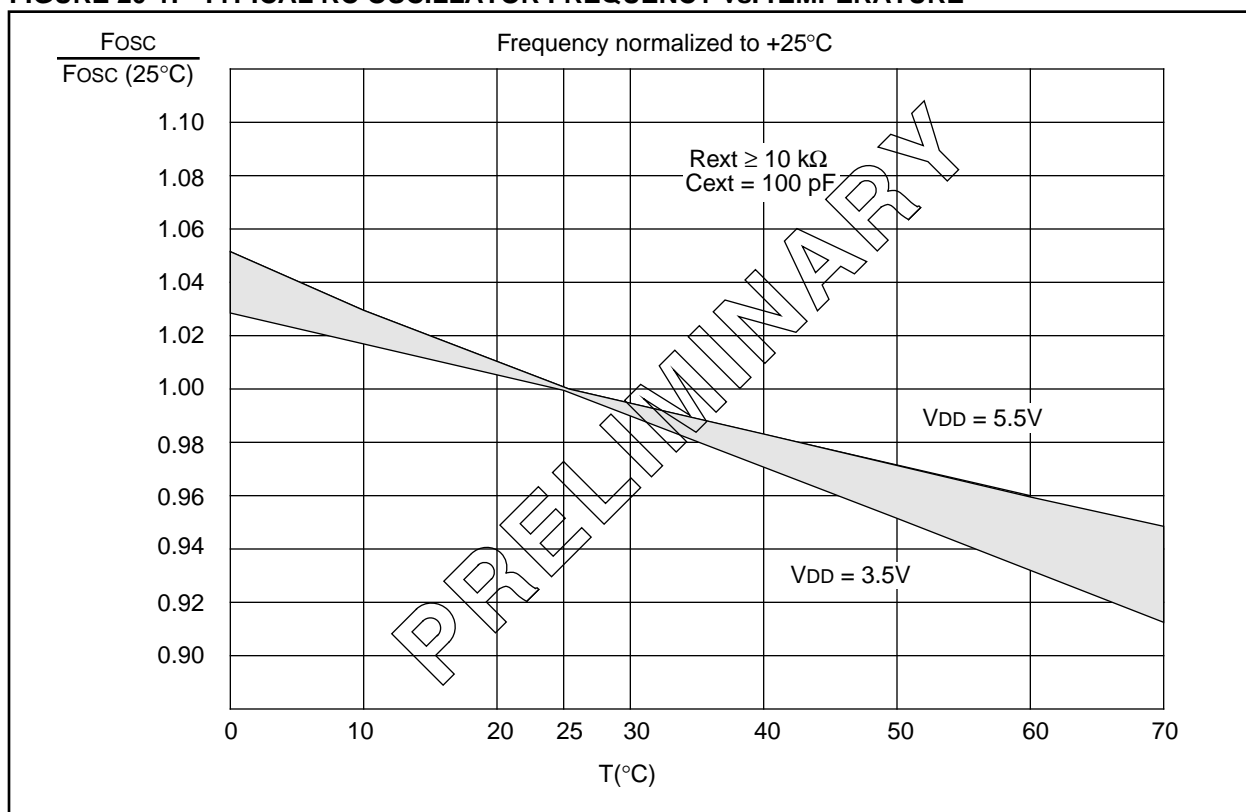
The graphs and tables provided in this section are for design guidance and are not tested nor guaranteed. In some graphs or tables the data presented is outside specified operating range (e.g. outside specified  $V_{DD}$  range). This is for information only and devices are ensured to operate properly only within the specified range.

The data presented in this section is a statistical summary of data collected on units from different lots over a period of time. "Typical" represents the mean of the distribution while "max" or "min" represents  $(\text{mean} + 3\sigma)$  and  $(\text{mean} - 3\sigma)$  respectively where  $\sigma$  is standard deviation.

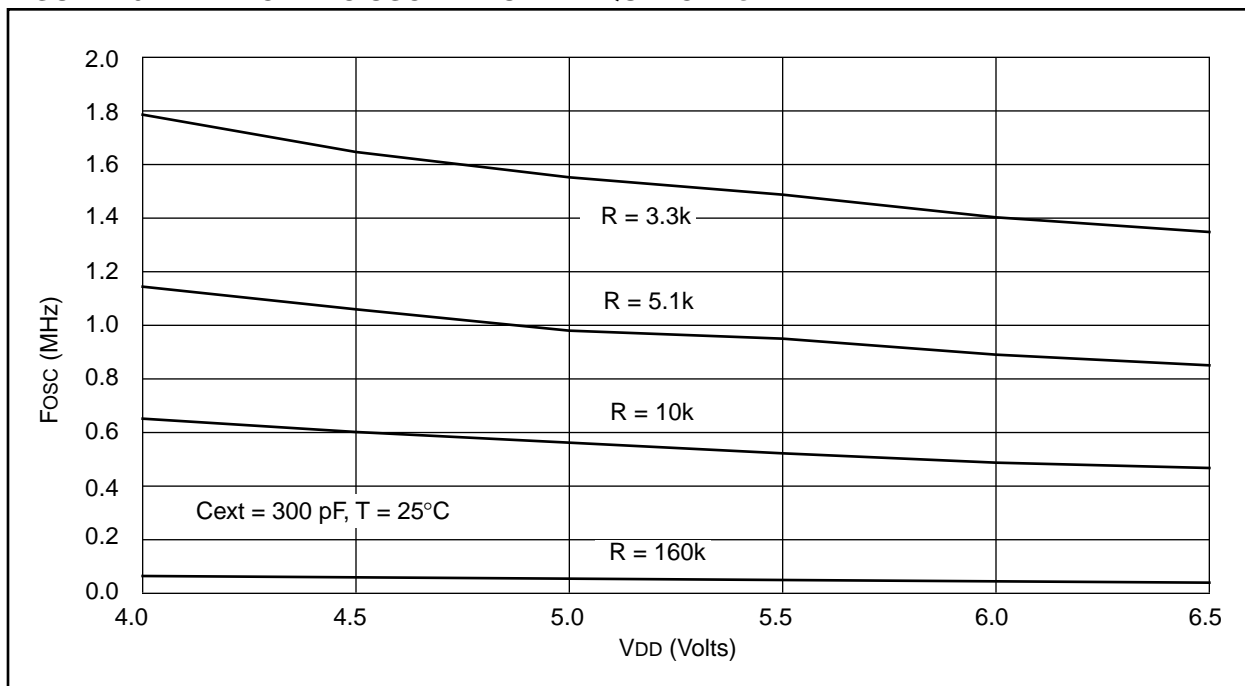
**TABLE 20-1: PIN CAPACITANCE PER PACKAGE TYPE**

Pin Name	Typical Capacitance (pF)			
	40-pin DIP	44-pin PLCC	44-pin MQFP	44-pin TQFP
All pins, except $\overline{\text{MCLR}}$ , $V_{DD}$ , and $V_{SS}$	10	10	10	10
$\overline{\text{MCLR}}$ pin	20	20	20	20

**FIGURE 20-1: TYPICAL RC OSCILLATOR FREQUENCY vs. TEMPERATURE**



**FIGURE 20-4: TYPICAL RC OSCILLATOR FREQUENCY vs. V<sub>DD</sub>**



**TABLE 20-2: RC OSCILLATOR FREQUENCIES**

Cext	Rext	Average	
		Fosc @ 5V, 25°C	
22 pF	10k	3.33 MHz	± 12%
	100k	353 kHz	± 13%
100 pF	3.3k	3.54 MHz	± 10%
	5.1k	2.43 MHz	± 14%
	10k	1.30 MHz	± 17%
	100k	129 kHz	± 10%
300 pF	3.3k	1.54 MHz	± 14%
	5.1k	980 kHz	± 12%
	10k	564 kHz	± 16%
	160k	35 kHz	± 18%

# PIC17C4X

## E.8 PIC17CXX Family of Devices

	Clock			Memory			Peripherals				Features		
	Maximum Frequency of Operation (MHz)	ROM	Program Memory (Words)	RAM Data Memory (bytes)	Timer Module(s)	Captures/PWMs	Serial Port(s) (USART)	Hardware Multiply	External Interrupts	Interrupt Sources	I/O Pins	Voltage Range (Volts)	Packages
PIC17C42	25	2K	—	232	TMR0, TMR1, TMR2, TMR3	2 2	Yes	—	Yes	11	33	4.5-5.5	40-pin DIP; 44-pin PLCC, MQFP
PIC17C42A	25	2K	—	232	TMR0, TMR1, TMR2, TMR3	2 2	Yes	Yes	Yes	11	33	2.5-6.0	40-pin DIP; 44-pin PLCC, TQFP, MQFP
PIC17CR42	25	—	2K	232	TMR0, TMR1, TMR2, TMR3	2 2	Yes	Yes	Yes	11	33	2.5-6.0	40-pin DIP; 44-pin PLCC, TQFP, MQFP
PIC17C43	25	4K	—	454	TMR0, TMR1, TMR2, TMR3	2 2	Yes	Yes	Yes	11	33	2.5-6.0	40-pin DIP; 44-pin PLCC, TQFP, MQFP
PIC17CR43	25	—	4K	454	TMR0, TMR1, TMR2, TMR3	2 2	Yes	Yes	Yes	11	33	2.5-6.0	40-pin DIP; 44-pin PLCC, TQFP, MQFP
PIC17C44	25	8K	—	454	TMR0, TMR1, TMR2, TMR3	2 2	Yes	Yes	Yes	11	33	2.5-6.0	40-pin DIP; 44-pin PLCC, TQFP, MQFP

All PIC16/17 Family devices have Power-on Reset, selectable Watchdog Timer, selectable code protect and high I/O current capability.

# PIC17C4X

---

NOTES: