



Welcome to [E-XFL.COM](https://www.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	4MHz
Connectivity	-
Peripherals	Brown-out Detect/Reset, PWM, WDT
Number of I/O	13
Program Memory Size	1.75KB (1K x 14)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	68 x 8
Voltage - Supply (Vcc/Vdd)	2.5V ~ 6V
Data Converters	A/D 4x8b
Oscillator Type	External
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Through Hole
Package / Case	18-DIP (0.300", 7.62mm)
Supplier Device Package	18-PDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lc711-04e-p

PIC16C71X

TABLE 1-1: PIC16C71X FAMILY OF DEVICES

		PIC16C710	PIC16C71	PIC16C711	PIC16C715	PIC16C72	PIC16CR72 ⁽¹⁾
Clock	Maximum Frequency of Operation (MHz)	20	20	20	20	20	20
	EPROM Program Memory (x14 words)	512	1K	1K	2K	2K	—
Memory	ROM Program Memory (14K words)	—	—	—	—	—	2K
	Data Memory (bytes)	36	36	68	128	128	128
Peripherals	Timer Module(s)	TMR0	TMR0	TMR0	TMR0	TMR0, TMR1, TMR2	TMR0, TMR1, TMR2
	Capture/Compare/PWM Module(s)	—	—	—	—	1	1
	Serial Port(s) (SPI/I ² C, USART)	—	—	—	—	SPI/I ² C	SPI/I ² C
	Parallel Slave Port	—	—	—	—	—	—
	A/D Converter (8-bit) Channels	4	4	4	4	5	5
Features	Interrupt Sources	4	4	4	4	8	8
	I/O Pins	13	13	13	13	22	22
	Voltage Range (Volts)	2.5-6.0	3.0-6.0	2.5-6.0	2.5-5.5	2.5-6.0	3.0-5.5
	In-Circuit Serial Programming	Yes	Yes	Yes	Yes	Yes	Yes
	Brown-out Reset	Yes	—	Yes	Yes	Yes	Yes
	Packages	18-pin DIP, SOIC, 20-pin SSOP	18-pin DIP, SOIC	18-pin DIP, SOIC, 20-pin SSOP	18-pin DIP, SOIC, 20-pin SSOP	28-pin SDIP, SOIC, SSOP	28-pin SDIP, SOIC, SSOP

		PIC16C73A	PIC16C74A	PIC16C76	PIC16C77
Clock	Maximum Frequency of Operation (MHz)	20	20	20	20
	EPROM Program Memory (x14 words)	4K	4K	8K	8K
Memory	Data Memory (bytes)	192	192	376	376
	Timer Module(s)	TMR0, TMR1, TMR2	TMR0, TMR1, TMR2	TMR0, TMR1, TMR2	TMR0, TMR1, TMR2
Peripherals	Capture/Compare/PWM Module(s)	2	2	2	2
	Serial Port(s) (SPI/I ² C, USART)	SPI/I ² C, USART	SPI/I ² C, USART	SPI/I ² C, USART	SPI/I ² C, USART
	Parallel Slave Port	—	Yes	—	Yes
	A/D Converter (8-bit) Channels	5	8	5	8
	Interrupt Sources	11	12	11	12
Features	I/O Pins	22	33	22	33
	Voltage Range (Volts)	2.5-6.0	2.5-6.0	2.5-6.0	2.5-6.0
	In-Circuit Serial Programming	Yes	Yes	Yes	Yes
	Brown-out Reset	Yes	Yes	Yes	Yes
	Packages	28-pin SDIP, SOIC	40-pin DIP; 44-pin PLCC, MQFP, TQFP	28-pin SDIP, SOIC	40-pin DIP; 44-pin PLCC, MQFP, TQFP

All PIC16/17 Family devices have Power-on Reset, selectable Watchdog Timer, selectable code protect and high I/O current capability. All PIC16C7XX Family devices use serial programming with clock pin RB6 and data pin RB7.

Note 1: Please contact your local Microchip sales office for availability of these devices.

3.0 ARCHITECTURAL OVERVIEW

The high performance of the PIC16CXX family can be attributed to a number of architectural features commonly found in RISC microprocessors. To begin with, the PIC16CXX uses a Harvard architecture, in which, program and data are accessed from separate memories using separate buses. This improves bandwidth over traditional von Neumann architecture in which program and data are fetched from the same memory using the same bus. Separating program and data buses further allows instructions to be sized differently than the 8-bit wide data word. Instruction opcodes are 14-bits wide making it possible to have all single word instructions. A 14-bit wide program memory access bus fetches a 14-bit instruction in a single cycle. A two-stage pipeline overlaps fetch and execution of instructions (Example 3-1). Consequently, all instructions (35) execute in a single cycle (200 ns @ 20 MHz) except for program branches.

The table below lists program memory (EPROM) and data memory (RAM) for each PIC16C71X device.

Device	Program Memory	Data Memory
PIC16C710	512 x 14	36 x 8
PIC16C71	1K x 14	36 x 8
PIC16C711	1K x 14	68 x 8
PIC16C715	2K x 14	128 x 8

The PIC16CXX can directly or indirectly address its register files or data memory. All special function registers, including the program counter, are mapped in the data memory. The PIC16CXX has an orthogonal (symmetrical) instruction set that makes it possible to carry out any operation on any register using any addressing mode. This symmetrical nature and lack of 'special optimal situations' make programming with the PIC16CXX simple yet efficient. In addition, the learning curve is reduced significantly.

PIC16CXX devices contain an 8-bit ALU and working register. The ALU is a general purpose arithmetic unit. It performs arithmetic and Boolean functions between the data in the working register and any register file.

The ALU is 8-bits wide and capable of addition, subtraction, shift and logical operations. Unless otherwise mentioned, arithmetic operations are two's complement in nature. In two-operand instructions, typically one operand is the working register (W register). The other operand is a file register or an immediate constant. In single operand instructions, the operand is either the W register or a file register.

The W register is an 8-bit working register used for ALU operations. It is not an addressable register.

Depending on the instruction executed, the ALU may affect the values of the Carry (C), Digit Carry (DC), and Zero (Z) bits in the STATUS register. The C and DC bits operate as a borrow bit and a digit borrow out bit, respectively, in subtraction. See the `SUBLW` and `SUBWF` instructions for examples.

TABLE 3-1: PIC16C710/71/711/715 PINOUT DESCRIPTION

Pin Name	DIP Pin#	SSOP Pin# ⁽⁴⁾	SOIC Pin#	I/O/P Type	Buffer Type	Description
OSC1/CLKIN	16	18	16	I	ST/CMOS ⁽³⁾	Oscillator crystal input/external clock source input.
OSC2/CLKOUT	15	17	15	O	—	Oscillator crystal output. Connects to crystal or resonator in crystal oscillator mode. In RC mode, OSC2 pin outputs CLKOUT which has 1/4 the frequency of OSC1, and denotes the instruction cycle rate.
MCLR/VPP	4	4	4	I/P	ST	Master clear (reset) input or programming voltage input. This pin is an active low reset to the device.
RA0/AN0	17	19	17	I/O	TTL	PORTA is a bi-directional I/O port. RA0 can also be analog input0 RA1 can also be analog input1 RA2 can also be analog input2 RA3 can also be analog input3 or analog reference voltage RA4 can also be the clock input to the Timer0 module. Output is open drain type.
RA1/AN1	18	20	18	I/O	TTL	
RA2/AN2	1	1	1	I/O	TTL	
RA3/AN3/VREF	2	2	2	I/O	TTL	
RA4/T0CKI	3	3	3	I/O	ST	
RB0/INT	6	7	6	I/O	TTL/ST ⁽¹⁾	PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-up on all inputs. RB0 can also be the external interrupt pin. Interrupt on change pin. Interrupt on change pin. Interrupt on change pin. Serial programming clock. Interrupt on change pin. Serial programming data.
RB1	7	8	7	I/O	TTL	
RB2	8	9	8	I/O	TTL	
RB3	9	10	9	I/O	TTL	
RB4	10	11	10	I/O	TTL	
RB5	11	12	11	I/O	TTL	
RB6	12	13	12	I/O	TTL/ST ⁽²⁾	
RB7	13	14	13	I/O	TTL/ST ⁽²⁾	
VSS	5	4, 6	5	P	—	Ground reference for logic and I/O pins.
VDD	14	15, 16	14	P	—	Positive supply for logic and I/O pins.

Legend: I = input O = output I/O = input/output P = power
 — = Not used TTL = TTL input ST = Schmitt Trigger input

- Note 1: This buffer is a Schmitt Trigger input when configured as the external interrupt.
 2: This buffer is a Schmitt Trigger input when used in serial programming mode.
 3: This buffer is a Schmitt Trigger input when configured in RC oscillator mode and a CMOS input otherwise.
 4: The PIC16C71 is not available in SSOP package.

PIC16C71X

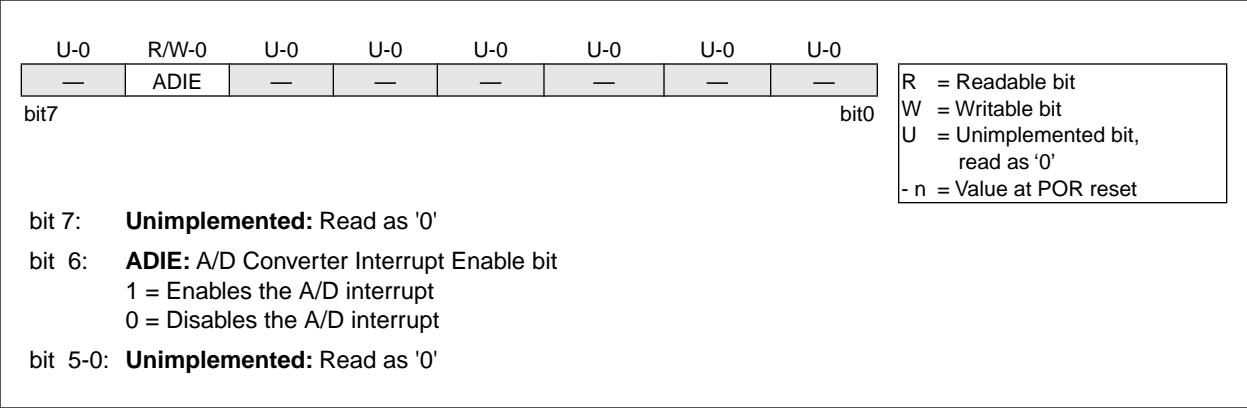
4.2.2.4 PIE1 REGISTER

Applicable Devices	710	71	711	715
---------------------------	-----	----	-----	-----

Note: Bit PEIE (INTCON<6>) must be set to enable any peripheral interrupt.

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

FIGURE 4-10: PIE1 REGISTER (ADDRESS 8Ch)



PIC16C71X

4.2.2.6 PCON REGISTER

Applicable Devices 710 71 711 715

The Power Control (PCON) register contains a flag bit to allow differentiation between a Power-on Reset (POR) to an external MCLR Reset or WDT Reset. Those devices with brown-out detection circuitry contain an additional bit to differentiate a Brown-out Reset (BOR) condition from a Power-on Reset condition. For the PIC16C715 the PCON register also contains status bits MPEEN and PER. MPEEN reflects the value of the MPEEN bit in the configuration word. PER indicates a parity error reset has occurred.

Note: $\overline{\text{BOR}}$ is unknown on Power-on Reset. It must then be set by the user and checked on subsequent resets to see if BOR is clear, indicating a brown-out has occurred. The $\overline{\text{BOR}}$ status bit is a don't care and is not necessarily predictable if the brown-out circuit is disabled (by clearing the BODEN bit in the Configuration word).

FIGURE 4-12: PCON REGISTER (ADDRESS 8Eh), PIC16C710/711

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-q
—	—	—	—	—	—	POR	$\overline{\text{BOR}}$
bit7						bit0	

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

bit 7-2: **Unimplemented:** Read as '0'

bit 1: **$\overline{\text{POR}}$:** Power-on Reset Status bit
1 = No Power-on Reset occurred
0 = A Power-on Reset occurred (must be set in software after a Power-on Reset occurs)

bit 0: **$\overline{\text{BOR}}$:** Brown-out Reset Status bit
1 = No Brown-out Reset occurred
0 = A Brown-out Reset occurred (must be set in software after a Brown-out Reset occurs)

FIGURE 4-13: PCON REGISTER (ADDRESS 8Eh), PIC16C715

R-U	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-q
MPEEN	—	—	—	—	PER	POR	$\overline{\text{BOR}}^{(1)}$
bit7					bit0		

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

bit 7: **MPEEN:** Memory Parity Error Circuitry Status bit
Reflects the value of configuration word bit, MPEEN

bit 6-3: **Unimplemented:** Read as '0'

bit 2: **$\overline{\text{PER}}$:** Memory Parity Error Reset Status bit
1 = No Error occurred
0 = Program Memory Fetch Parity Error occurred (must be set in software after a Parity Error Reset)

bit 1: **$\overline{\text{POR}}$:** Power-on Reset Status bit
1 = No Power-on Reset occurred
0 = A Power-on Reset occurred (must be set in software after a Power-on Reset occurs)

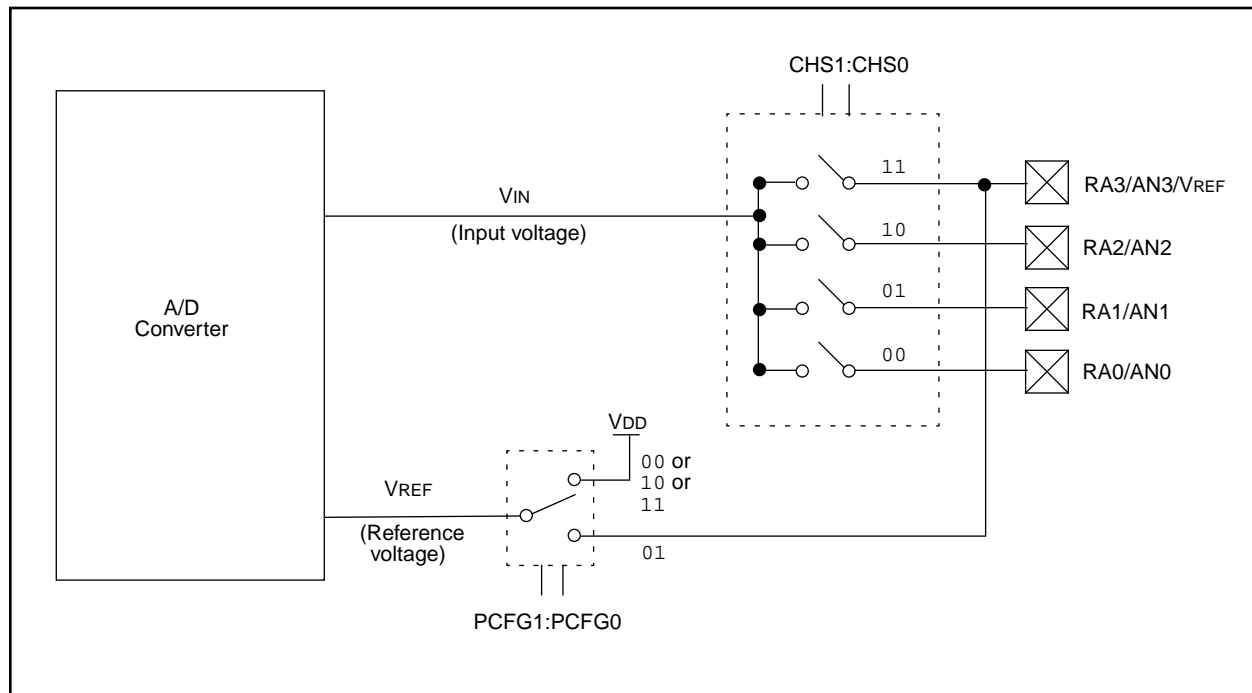
bit 0: **$\overline{\text{BOR}}$:** Brown-out Reset Status bit
1 = No Brown-out Reset occurred
0 = A Brown-out Reset occurred (must be set in software after a Brown-out Reset occurs)

The ADRES register contains the result of the A/D conversion. When the A/D conversion is complete, the result is loaded into the ADRES register, the GO/DONE bit (ADCON0<2>) is cleared, and A/D interrupt flag bit ADIF is set. The block diagram of the A/D module is shown in Figure 7-4.

After the A/D module has been configured as desired, the selected channel must be acquired before the conversion is started. The analog input channels must have their corresponding TRIS bits selected as an input. To determine acquisition time, see Section 7.1. After this acquisition time has elapsed the A/D conversion can be started. The following steps should be followed for doing an A/D conversion:

1. Configure the A/D module:
 - Configure analog pins / voltage reference / and digital I/O (ADCON1)
 - Select A/D input channel (ADCON0)
 - Select A/D conversion clock (ADCON0)
 - Turn on A/D module (ADCON0)
2. Configure A/D interrupt (if desired):
 - Clear ADIF bit
 - Set ADIE bit
 - Set GIE bit
3. Wait the required acquisition time.
4. Start conversion:
 - Set GO/DONE bit (ADCON0)
5. Wait for A/D conversion to complete, by either:
 - Polling for the GO/DONE bit to be cleared
 OR
 - Waiting for the A/D interrupt
6. Read A/D Result register (ADRES), clear bit ADIF if required.
7. For next conversion, go to step 1 or step 2 as required. The A/D conversion time per bit is defined as TAD. A minimum wait of 2TAD is required before next acquisition starts.

FIGURE 7-4: A/D BLOCK DIAGRAM



PIC16C71X

7.4 A/D Conversions

Example 7-2 shows how to perform an A/D conversion. The RA pins are configured as analog inputs. The analog reference (VREF) is the device VDD. The A/D interrupt is enabled, and the A/D conversion clock is FRC. The conversion is performed on the RA0 pin (channel 0).

Note: The GO/DONE bit should **NOT** be set in the same instruction that turns on the A/D.

Clearing the GO/DONE bit during a conversion will abort the current conversion. The ADRES register will NOT be updated with the partially completed A/D conversion sample. That is, the ADRES register will continue to contain the value of the last completed conversion (or the last value written to the ADRES register). After the A/D conversion is aborted, a 2TAD wait is required before the next acquisition is started. After this 2TAD wait, an acquisition is automatically started on the selected channel.

EXAMPLE 7-2: A/D CONVERSION

```
BSF    STATUS, RP0           ; Select Bank 1
CLRF   ADCON1                ; Configure A/D inputs
BCF    STATUS, RP0           ; Select Bank 0
MOVLW  0xC1                  ; RC Clock, A/D is on, Channel 0 is selected
MOVWF  ADCON0                ;
BSF    INTCON, ADIE           ; Enable A/D Interrupt
BSF    INTCON, GIE            ; Enable all interrupts
;
; Ensure that the required sampling time for the selected input channel has elapsed.
; Then the conversion may be started.
;
BSF    ADCON0, GO             ; Start A/D Conversion
:      ; The ADIF bit will be set and the GO/DONE bit
:      ; is cleared upon completion of the A/D Conversion.
```


8.5 Interrupts

Applicable Devices	710	71	711	715
---------------------------	-----	----	-----	-----

The PIC16C71X family has 4 sources of interrupt.

Interrupt Sources
External interrupt RB0/INT
TMR0 overflow interrupt
PORTB change interrupts (pins RB7:RB4)
A/D Interrupt

The interrupt control register (INTCON) records individual interrupt requests in flag bits. It also has individual and global interrupt enable bits.

Note: Individual interrupt flag bits are set regardless of the status of their corresponding mask bit or the GIE bit.

A global interrupt enable bit, GIE (INTCON<7>) enables (if set) all un-masked interrupts or disables (if cleared) all interrupts. When bit GIE is enabled, and an interrupt's flag bit and mask bit are set, the interrupt will vector immediately. Individual interrupts can be disabled through their corresponding enable bits in various registers. Individual interrupt bits are set regardless of the status of the GIE bit. The GIE bit is cleared on reset.

The "return from interrupt" instruction, RETFIE, exits the interrupt routine as well as sets the GIE bit, which re-enables interrupts.

The RB0/INT pin interrupt, the RB port change interrupt and the TMR0 overflow interrupt flags are contained in the INTCON register.

The peripheral interrupt flags are contained in the special function registers PIR1 and PIR2. The corresponding interrupt enable bits are contained in special function registers PIE1 and PIE2, and the peripheral interrupt enable bit is contained in special function register INTCON.

When an interrupt is responded to, the GIE bit is cleared to disable any further interrupt, the return address is pushed onto the stack and the PC is loaded with 0004h. Once in the interrupt service routine the source(s) of the interrupt can be determined by polling the interrupt flag bits. The interrupt flag bit(s) must be cleared in software before re-enabling interrupts to avoid recursive interrupts.

For external interrupt events, such as the INT pin or PORTB change interrupt, the interrupt latency will be three or four instruction cycles. The exact latency depends when the interrupt event occurs (Figure 8-19). The latency is the same for one or two cycle instructions. Individual interrupt flag bits are set regardless of the status of their corresponding mask bit or the GIE bit.

Note: For the PIC16C71

If an interrupt occurs while the Global Interrupt Enable (GIE) bit is being cleared, the GIE bit may unintentionally be re-enabled by the user's Interrupt Service Routine (the RETFIE instruction). The events that would cause this to occur are:

1. An instruction clears the GIE bit while an interrupt is acknowledged.
2. The program branches to the Interrupt vector and executes the Interrupt Service Routine.
3. The Interrupt Service Routine completes with the execution of the RETFIE instruction. This causes the GIE bit to be set (enables interrupts), and the program returns to the instruction after the one which was meant to disable interrupts.

Perform the following to ensure that interrupts are globally disabled:

```

LOOP BCF    INTCON, GIE    ; Disable global
                               ; interrupt bit
      BTFSC INTCON, GIE    ; Global interrupt
                               ; disabled?
      GOTO  LOOP           ; NO, try again
      :                   ; Yes, continue
                               ; with program
                               ; flow

```

PIC16C71X

TABLE 9-2: PIC16CXX INSTRUCTION SET

Mnemonic, Operands	Description	Cycles	14-Bit Opcode				Status Affected	Notes	
			MSb		LSb				
BYTE-ORIENTED FILE REGISTER OPERATIONS									
ADDWF	f, d	Add W and f	1	00	0111	dfff	ffff	C,DC,Z	1,2
ANDWF	f, d	AND W with f	1	00	0101	dfff	ffff	Z	1,2
CLRF	f	Clear f	1	00	0001	1fff	ffff	Z	2
CLRW	-	Clear W	1	00	0001	0xxx	xxxx	Z	
COMF	f, d	Complement f	1	00	1001	dfff	ffff	Z	1,2
DECF	f, d	Decrement f	1	00	0011	dfff	ffff	Z	1,2
DECFSZ	f, d	Decrement f, Skip if 0	1(2)	00	1011	dfff	ffff		1,2,3
INCF	f, d	Increment f	1	00	1010	dfff	ffff	Z	1,2
INCFSZ	f, d	Increment f, Skip if 0	1(2)	00	1111	dfff	ffff		1,2,3
IORWF	f, d	Inclusive OR W with f	1	00	0100	dfff	ffff	Z	1,2
MOVF	f, d	Move f	1	00	1000	dfff	ffff	Z	1,2
MOVWF	f	Move W to f	1	00	0000	1fff	ffff		
NOP	-	No Operation	1	00	0000	0xx0	0000		
RLF	f, d	Rotate Left f through Carry	1	00	1101	dfff	ffff	C	1,2
RRF	f, d	Rotate Right f through Carry	1	00	1100	dfff	ffff	C	1,2
SUBWF	f, d	Subtract W from f	1	00	0010	dfff	ffff	C,DC,Z	1,2
SWAPF	f, d	Swap nibbles in f	1	00	1110	dfff	ffff		1,2
XORWF	f, d	Exclusive OR W with f	1	00	0110	dfff	ffff	Z	1,2
BIT-ORIENTED FILE REGISTER OPERATIONS									
BCF	f, b	Bit Clear f	1	01	00bb	bfff	ffff		1,2
BSF	f, b	Bit Set f	1	01	01bb	bfff	ffff		1,2
BTFSC	f, b	Bit Test f, Skip if Clear	1 (2)	01	10bb	bfff	ffff		3
BTFSS	f, b	Bit Test f, Skip if Set	1 (2)	01	11bb	bfff	ffff		3
LITERAL AND CONTROL OPERATIONS									
ADDLW	k	Add literal and W	1	11	111x	kkkk	kkkk	C,DC,Z	
ANDLW	k	AND literal with W	1	11	1001	kkkk	kkkk	Z	
CALL	k	Call subroutine	2	10	0kkk	kkkk	kkkk		
CLRWDT	-	Clear Watchdog Timer	1	00	0000	0110	0100	TO,PD	
GOTO	k	Go to address	2	10	1kkk	kkkk	kkkk		
IORLW	k	Inclusive OR literal with W	1	11	1000	kkkk	kkkk	Z	
MOVLW	k	Move literal to W	1	11	00xx	kkkk	kkkk		
RETFIE	-	Return from interrupt	2	00	0000	0000	1001		
RETLW	k	Return with literal in W	2	11	01xx	kkkk	kkkk		
RETURN	-	Return from Subroutine	2	00	0000	0000	1000		
SLEEP	-	Go into standby mode	1	00	0000	0110	0011	TO,PD	
SUBLW	k	Subtract W from literal	1	11	110x	kkkk	kkkk	C,DC,Z	
XORLW	k	Exclusive OR literal with W	1	11	1010	kkkk	kkkk	Z	

Note 1: When an I/O register is modified as a function of itself (e.g., `MOVF PORTB, 1`), the value used will be that value present on the pins themselves. For example, if the data latch is '1' for a pin configured as input and is driven low by an external device, the data will be written back with a '0'.

2: If this instruction is executed on the TMR0 register (and, where applicable, d = 1), the prescaler will be cleared if assigned to the Timer0 Module.

3: If Program Counter (PC) is modified or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a NOP.

PIC16C71X

BCF Bit Clear f

Syntax: `[label] BCF f,b`

Operands: $0 \leq f \leq 127$
 $0 \leq b \leq 7$

Operation: $0 \rightarrow (f)$

Status Affected: None

Encoding:

01	00bb	bfff	ffff
----	------	------	------

Description: Bit 'b' in register 'f' is cleared.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process data	Write register 'f'

Example

```
BCF    FLAG_REG, 7

Before Instruction
FLAG_REG = 0xC7
After Instruction
FLAG_REG = 0x47
```

BSF Bit Set f

Syntax: `[label] BSF f,b`

Operands: $0 \leq f \leq 127$
 $0 \leq b \leq 7$

Operation: $1 \rightarrow (f)$

Status Affected: None

Encoding:

01	01bb	bfff	ffff
----	------	------	------

Description: Bit 'b' in register 'f' is set.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process data	Write register 'f'

Example

```
BSF    FLAG_REG, 7

Before Instruction
FLAG_REG = 0x0A
After Instruction
FLAG_REG = 0x8A
```

BTFSC Bit Test, Skip if Clear

Syntax: `[label] BTFSC f,b`

Operands: $0 \leq f \leq 127$
 $0 \leq b \leq 7$

Operation: skip if $(f) = 0$

Status Affected: None

Encoding:

01	10bb	bfff	ffff
----	------	------	------

Description: If bit 'b' in register 'f' is '1' then the next instruction is executed.
 If bit 'b', in register 'f', is '0' then the next instruction is discarded, and a NOP is executed instead, making this a 2Tcy instruction.

Words: 1

Cycles: 1(2)

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process data	NOP

If Skip: (2nd Cycle)

Q1	Q2	Q3	Q4
NOP	NOP	NOP	NOP

Example

```
HERE    BTFSC FLAG, 1
FALSE   GOTO  PROCESS_CODE
TRUE    :
```

Before Instruction
 PC = address HERE
 After Instruction
 if $FLAG<1> = 0$,
 PC = address TRUE
 if $FLAG<1> = 1$,
 PC = address FALSE

NOP		No Operation			
Syntax:	[<i>label</i>] NOP				
Operands:	None				
Operation:	No operation				
Status Affected:	None				
Encoding:	00	0000	0xx0	0000	
Description:	No operation.				
Words:	1				
Cycles:	1				
Q Cycle Activity:	Q1	Q2	Q3	Q4	
	Decode	NOP	NOP	NOP	
Example	NOP				

RETFIE		Return from Interrupt			
Syntax:	[<i>label</i>] RETFIE				
Operands:	None				
Operation:	TOS → PC, 1 → GIE				
Status Affected:	None				
Encoding:	00	0000	0000	1001	
Description:	Return from Interrupt. Stack is POPed and Top of Stack (TOS) is loaded in the PC. Interrupts are enabled by setting Global Interrupt Enable bit, GIE (INTCON<7>). This is a two cycle instruction.				
Words:	1				
Cycles:	2				
Q Cycle Activity:	Q1	Q2	Q3	Q4	
1st Cycle	Decode	NOP	Set the GIE bit	Pop from the Stack	
2nd Cycle	NOP	NOP	NOP	NOP	

Example

```

RETFIE
After Interrupt
    PC = TOS
    GIE = 1

```

OPTION	Load Option Register			
Syntax:	[<i>label</i>] OPTION			
Operands:	None			
Operation:	(W) → OPTION			
Status Affected:	None			
Encoding:	00	0000	0110	0010
Description:	The contents of the W register are loaded in the OPTION register. This instruction is supported for code compatibility with PIC16C5X products. Since OPTION is a readable/writable register, the user can directly address it.			
Words:	1			
Cycles:	1			
Example	<div>To maintain upward compatibility with future PIC16CXX products, do not use this instruction.</div>			

PIC16C71X

SLEEP

Syntax: [*label*] SLEEP

Operands: None

Operation: 00h → WDT,
0 → WDT prescaler,
1 → \overline{TO} ,
0 → \overline{PD}

Status Affected: \overline{TO} , \overline{PD}

Encoding:

00	0000	0110	0011
----	------	------	------

Description: The power-down status bit, \overline{PD} is cleared. Time-out status bit, \overline{TO} is set. Watchdog Timer and its prescaler are cleared. The processor is put into SLEEP mode with the oscillator stopped. See Section 8.8 for more details.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	NOP	NOP	Go to Sleep

Example: SLEEP

SUBLW

Subtract W from Literal

Syntax: [*label*] SUBLW k

Operands: $0 \leq k \leq 255$

Operation: $k - (W) \rightarrow (W)$

Status Affected: C, DC, Z

Encoding:

11	110x	kkkk	kkkk
----	------	------	------

Description: The W register is subtracted (2's complement method) from the eight bit literal 'k'. The result is placed in the W register.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read literal 'k'	Process data	Write to W

Example 1: SUBLW 0x02

Before Instruction

W = 1
C = ?
Z = ?

After Instruction

W = 1
C = 1; result is positive
Z = 0

Example 2: Before Instruction

W = 2
C = ?
Z = ?

After Instruction

W = 0
C = 1; result is zero
Z = 1

Example 3: Before Instruction

W = 3
C = ?
Z = ?

After Instruction

W = 0xFF
C = 0; result is negative
Z = 0

FIGURE 12-18: TYPICAL I_{DD} vs. CAPACITANCE @ 500 kHz (RC MODE)

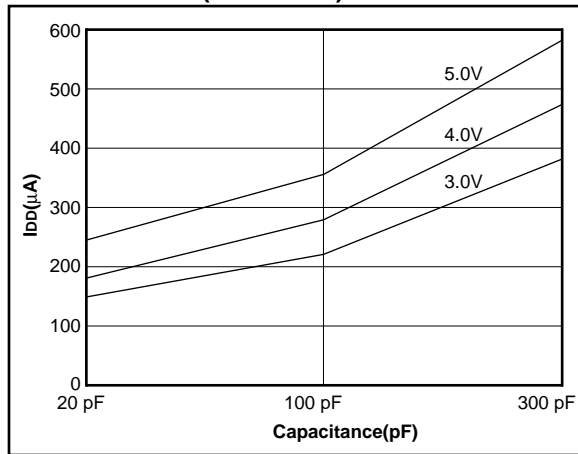


TABLE 12-1: RC OSCILLATOR FREQUENCIES

Cext	Rext	Average	
		Fosc @ 5V, 25°C	
22 pF	5k	4.12 MHz	± 1.4%
	10k	2.35 MHz	± 1.4%
	100k	268 kHz	± 1.1%
100 pF	3.3k	1.80 MHz	± 1.0%
	5k	1.27 MHz	± 1.0%
	10k	688 kHz	± 1.2%
	100k	77.2 kHz	± 1.0%
300 pF	3.3k	707 kHz	± 1.4%
	5k	501 kHz	± 1.2%
	10k	269 kHz	± 1.6%
	100k	28.3 kHz	± 1.1%

The percentage variation indicated here is part to part variation due to normal process distribution. The variation indicated is ± 3 standard deviation from average value for $V_{DD} = 5V$.

FIGURE 12-19: TRANSCONDUCTANCE(gm) OF HS OSCILLATOR vs. V_{DD}

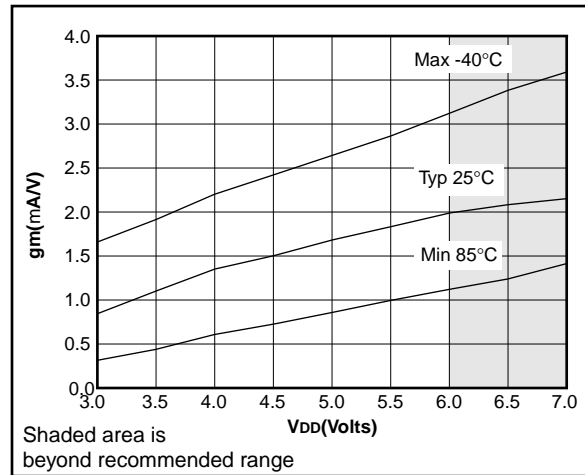


FIGURE 12-20: TRANSCONDUCTANCE(gm) OF LP OSCILLATOR vs. V_{DD}

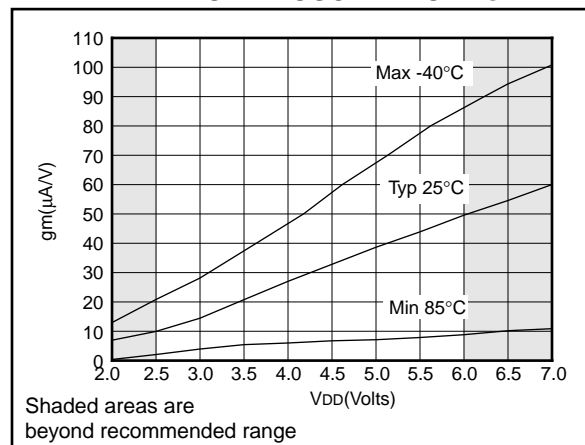
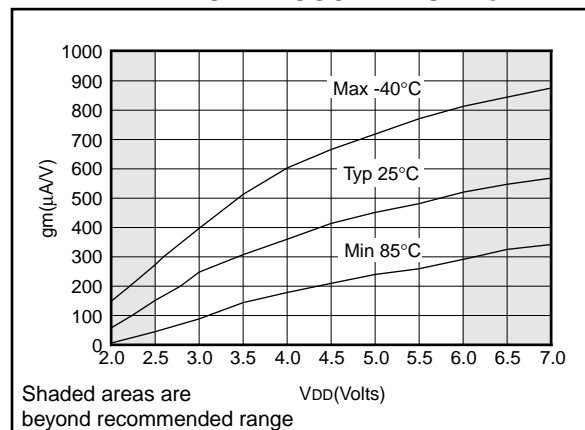


FIGURE 12-21: TRANSCONDUCTANCE(gm) OF XT OSCILLATOR vs. V_{DD}



**TABLE 13-7: A/D CONVERTER CHARACTERISTICS:
PIC16LC715-04 (COMMERCIAL, INDUSTRIAL)**

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
	NR	Resolution	—	—	8-bits	—	$V_{REF} = V_{DD}$, $V_{SS} \leq A_{IN} \leq V_{REF}$
	NINT	Integral error	—	—	less than ± 1 LSb	—	$V_{REF} = V_{DD}$, $V_{SS} \leq A_{IN} \leq V_{REF}$
	NDIF	Differential error	—	—	less than ± 1 LSb	—	$V_{REF} = V_{DD}$, $V_{SS} \leq A_{IN} \leq V_{REF}$
	NFS	Full scale error	—	—	less than ± 1 LSb	—	$V_{REF} = V_{DD}$, $V_{SS} \leq A_{IN} \leq V_{REF}$
	NOFF	Offset error	—	—	less than ± 1 LSb	—	$V_{REF} = V_{DD}$, $V_{SS} \leq A_{IN} \leq V_{REF}$
	—	Monotonicity	—	guaranteed	—	—	$V_{SS} \leq A_{IN} \leq V_{REF}$
	VREF	Reference voltage	2.5V	—	$V_{DD} + 0.3$	V	
	VAIN	Analog input voltage	$V_{SS} - 0.3$	—	$V_{REF} + 0.3$	V	
	ZAIN	Recommended impedance of analog voltage source	—	—	10.0	k Ω	
	IAD	A/D conversion current (V_{DD})	—	90	—	μ A	Average current consumption when A/D is on. (Note 1)
	IREF	VREF input current (Note 2)	—	—	1 10	mA μ A	During sampling All other times

* 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.

Note 1: When A/D is off, it will not consume any current other than minor leakage current. The power-down current spec includes any such leakage from the A/D module.

2: VREF current is from RA3 pin or VDD pin, whichever is selected as reference input.

FIGURE 14-8: TYPICAL I_{PD} vs. V_{DD} BROWN-OUT DETECT ENABLED (RC MODE)

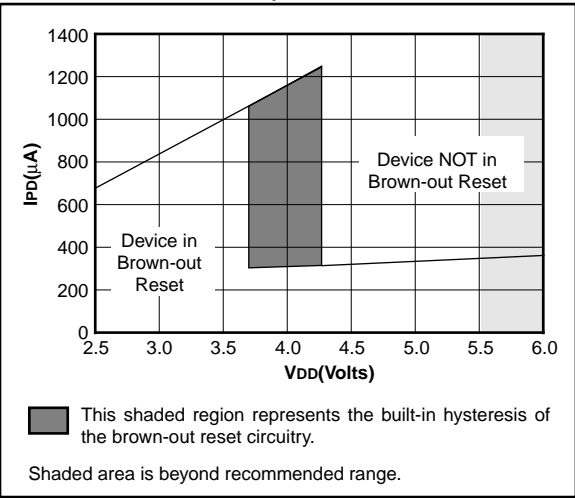


FIGURE 14-9: MAXIMUM I_{PD} vs. V_{DD} BROWN-OUT DETECT ENABLED (85°C TO -40°C, RC MODE)

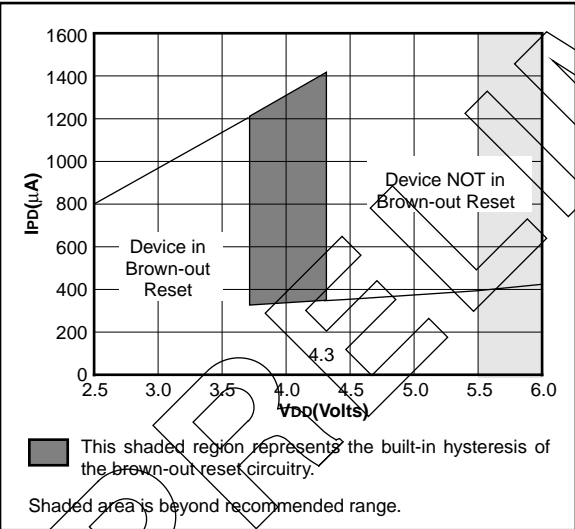


FIGURE 14-10: TYPICAL I_{PD} vs. TIMER1 ENABLED (32 kHz, RC0/RC1 = 33 pF/33 pF, RC MODE)

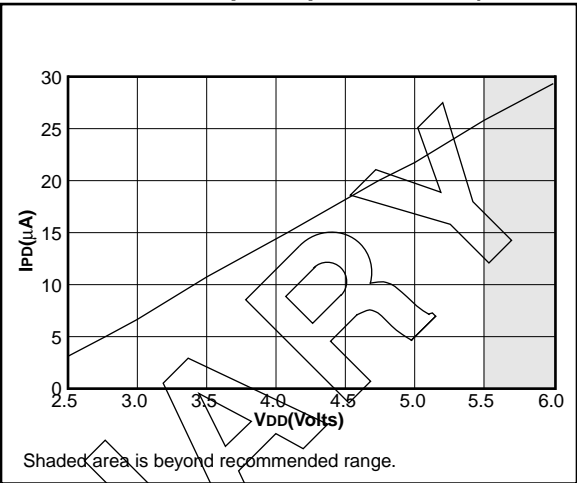


FIGURE 14-11: MAXIMUM I_{PD} vs. TIMER1 ENABLED (32 kHz, RC0/RC1 = 33 pF/33 pF, 85°C TO -40°C, RC MODE)

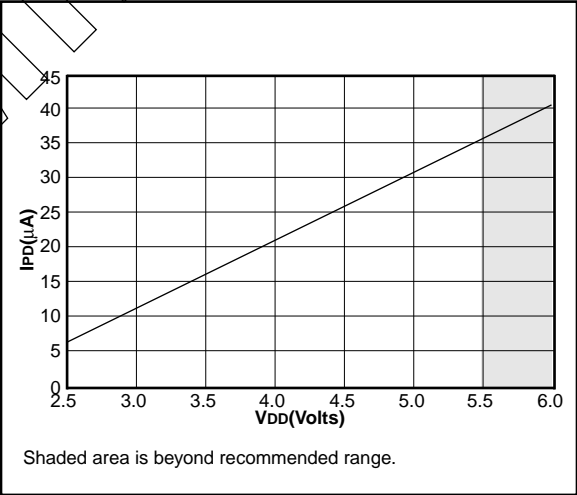


FIGURE 14-14: TYPICAL I_{DD} vs. FREQUENCY (RC MODE @ 100 pF, 25°C)

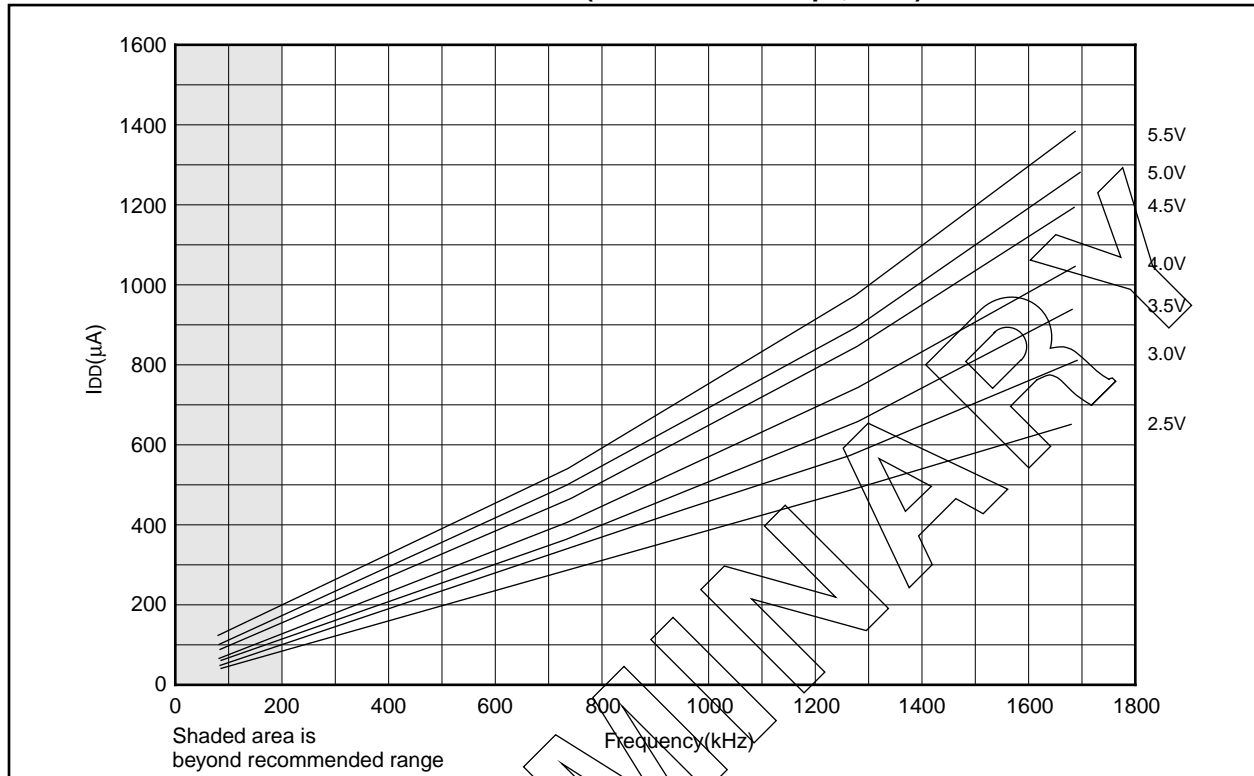


FIGURE 14-15: MAXIMUM I_{DD} vs. FREQUENCY (RC MODE @ 100 pF, -40°C TO 85°C)

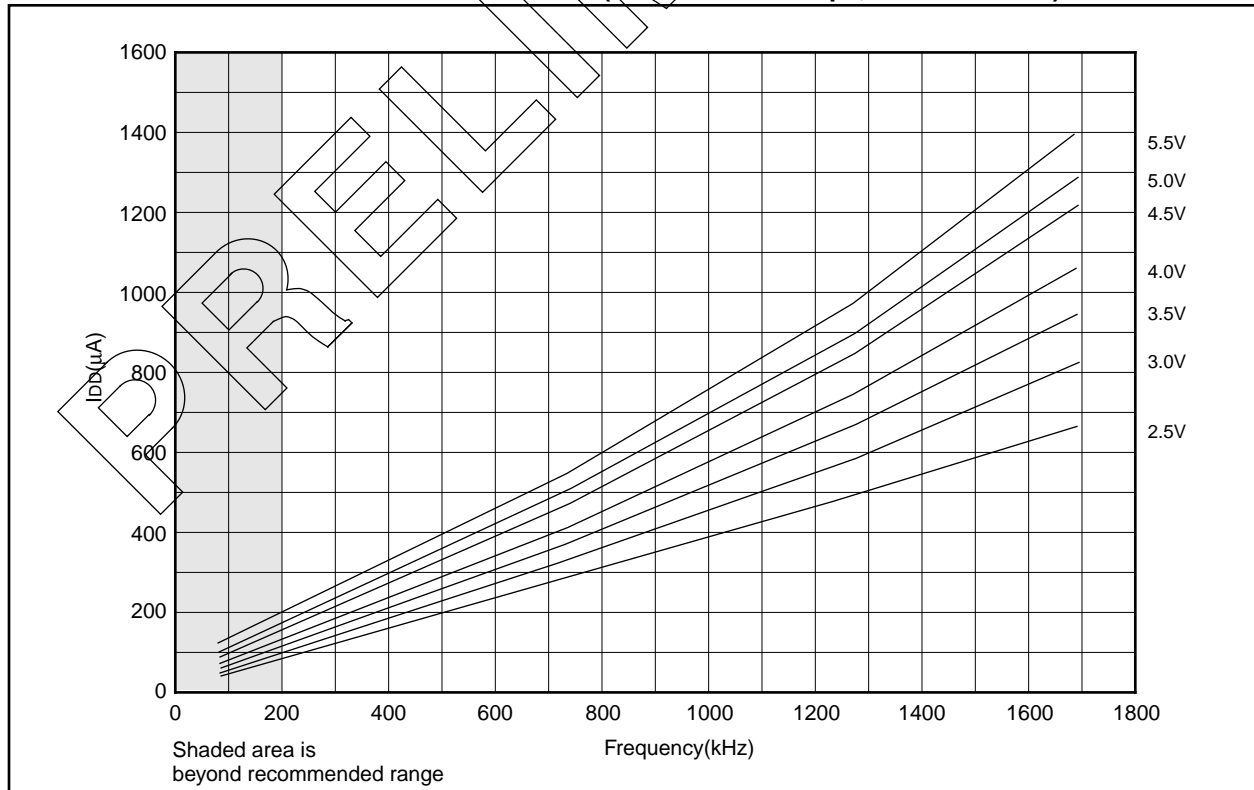


FIGURE 14-18: TYPICAL I_{DD} vs. CAPACITANCE @ 500 kHz (RC MODE)

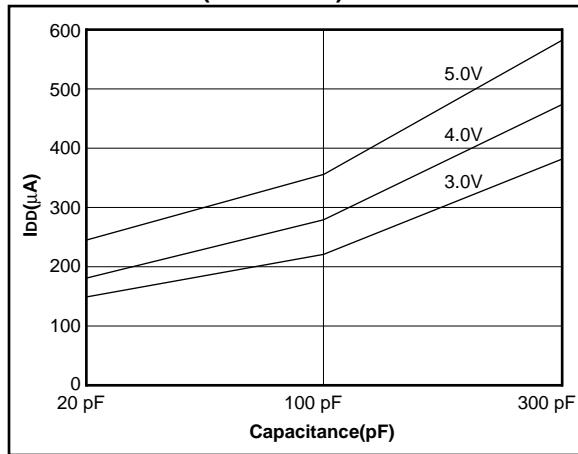


TABLE 14-1: RC OSCILLATOR FREQUENCIES

Cext	Rext	Average	
		Fosc @ 5V, 25°C	
22 pF	5k	4.12 MHz	± 1.4%
	10k	2.35 MHz	± 1.4%
	100k	268 kHz	± 1.1%
100 pF	3.3k	1.80 MHz	± 1.0%
	5k	1.27 MHz	± 1.0%
	10k	688 kHz	± 1.2%
	100k	77.2 kHz	± 1.0%
300 pF	3.3k	707 kHz	± 1.4%
	5k	501 kHz	± 1.2%
	10k	269 kHz	± 1.6%
	100k	28.3 kHz	± 1.1%

The percentage variation indicated here is part to part variation due to normal process distribution. The variation indicated is ± 3 standard deviation from average value for $V_{DD} = 5V$.

FIGURE 14-19: TRANSCONDUCTANCE(gm) OF HS OSCILLATOR vs. V_{DD}

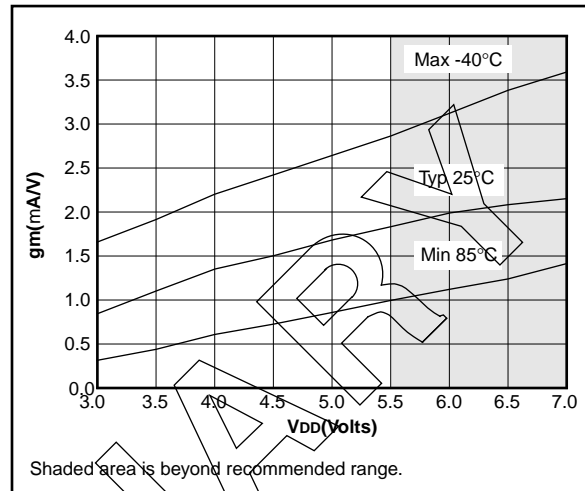


FIGURE 14-20: TRANSCONDUCTANCE(gm) OF LP OSCILLATOR vs. V_{DD}

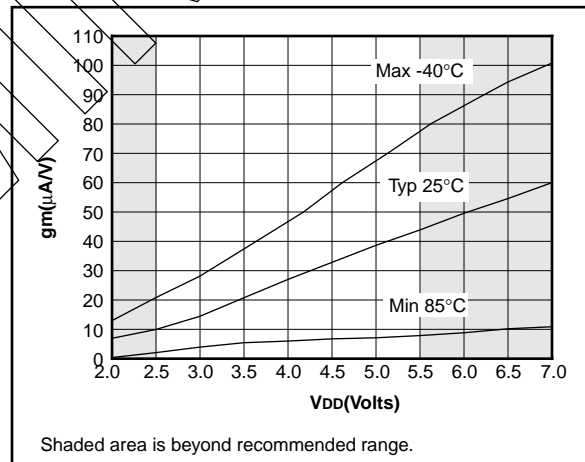
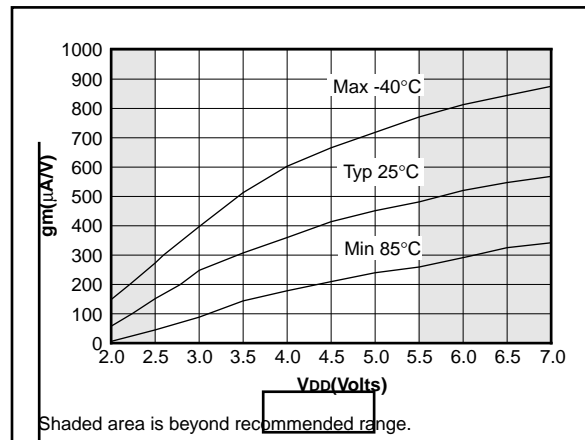


FIGURE 14-21: TRANSCONDUCTANCE(gm) OF XT OSCILLATOR vs. V_{DD}



PIC16C71X

Applicable Devices	710	71	711	715
--------------------	-----	----	-----	-----

15.4 Timing Parameter Symbolology

The timing parameter symbols have been created following one of the following formats:

- 1. TppS2ppS
- 2. TppS

T			
F	Frequency	T	Time

Lowercase letters (pp) and their meanings:

pp			
cc	CCP1	osc	OSC1
ck	CLKOUT	rd	RD
cs	CS	rw	RD or WR
di	SDI	sc	SCK
do	SDO	ss	SS
dt	Data in	t0	T0CKI
io	I/O port	t1	T1CKI
mc	MCLR	wr	WR

Uppercase letters and their meanings:

S			
F	Fall	P	Period
H	High	R	Rise
I	Invalid (Hi-impedance)	V	Valid
L	Low	Z	Hi-impedance

FIGURE 15-1: LOAD CONDITIONS

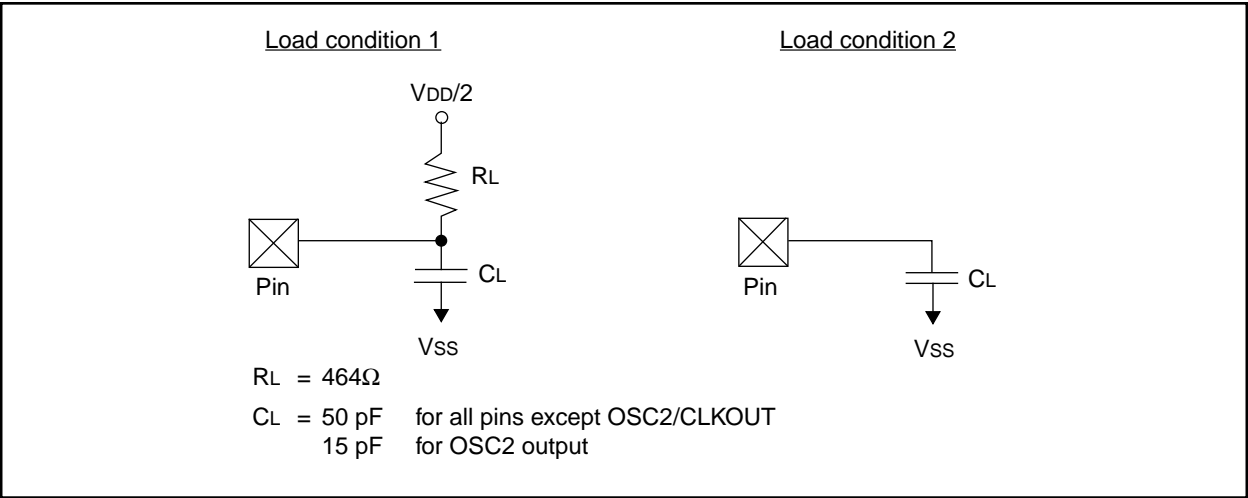


FIGURE 15-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING

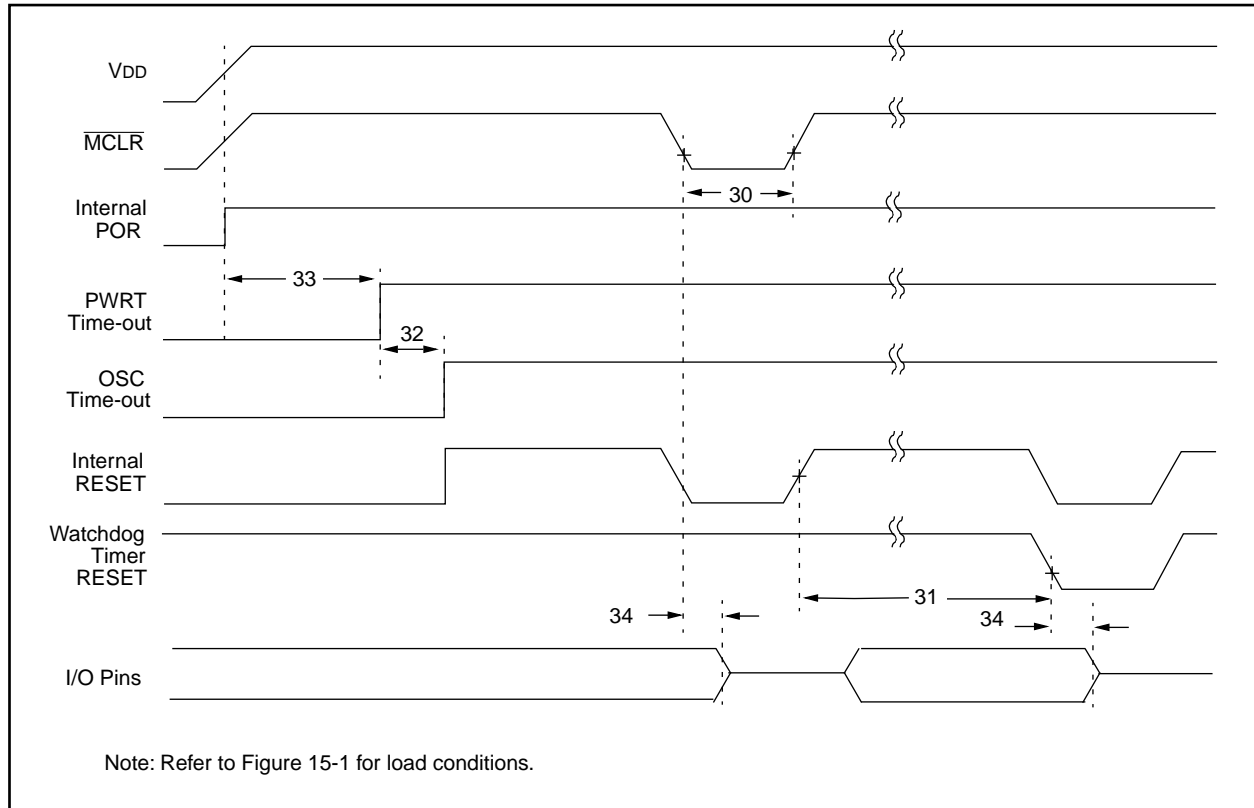


TABLE 15-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER REQUIREMENTS

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
30	TmCL	MCLR Pulse Width (low)	200	—	—	ns	VDD = 5V, -40°C to +85°C
31	Twdt	Watchdog Timer Time-out Period (No Prescaler)	7*	18	33*	ms	VDD = 5V, -40°C to +85°C
32	Tost	Oscillation Start-up Timer Period	—	1024 TOSC	—	—	TOSC = OSC1 period
33	Tpwrt	Power-up Timer Period	28*	72	132*	ms	VDD = 5V, -40°C to +85°C
34	Tioz	I/O High Impedance from MCLR Low	—	—	100	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.

PIC16C71X

Applicable Devices 710 71 711 715

FIGURE 16-14: MAXIMUM I_{DD} vs. FREQ WITH A/D OFF (EXT CLOCK, -55° TO +125°C)

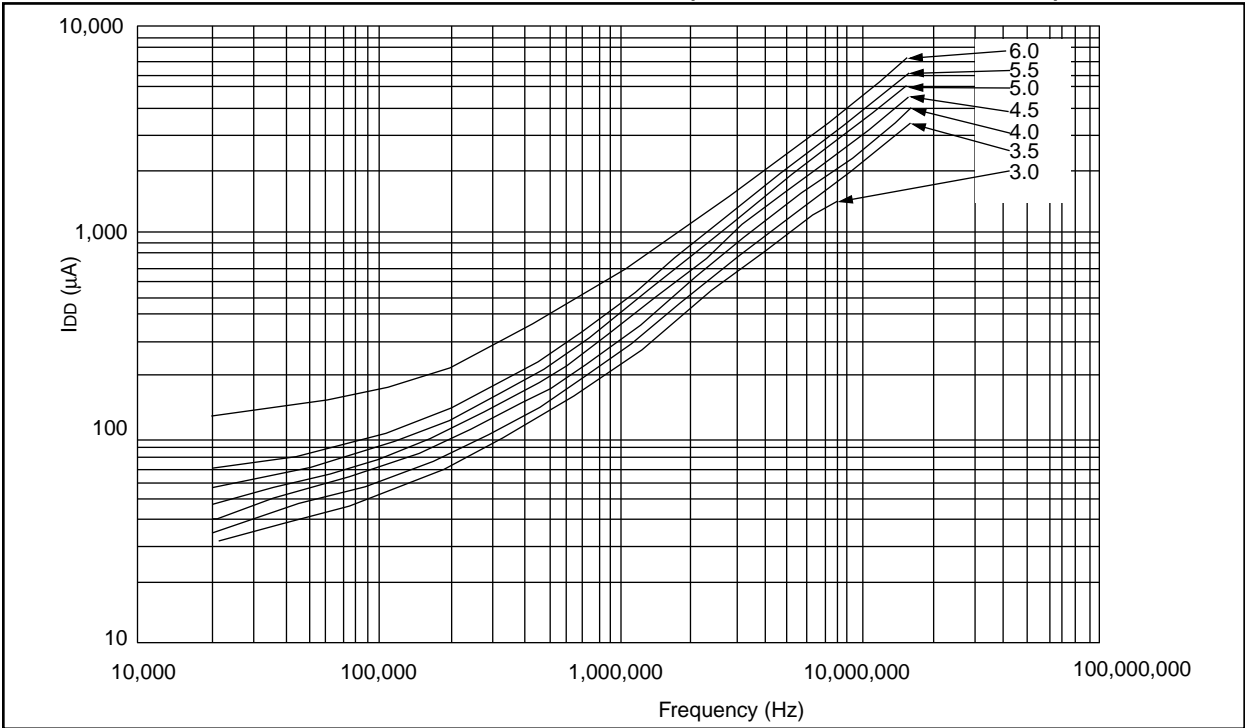


FIGURE 16-15: WDT TIMER TIME-OUT PERIOD vs. V_{DD}

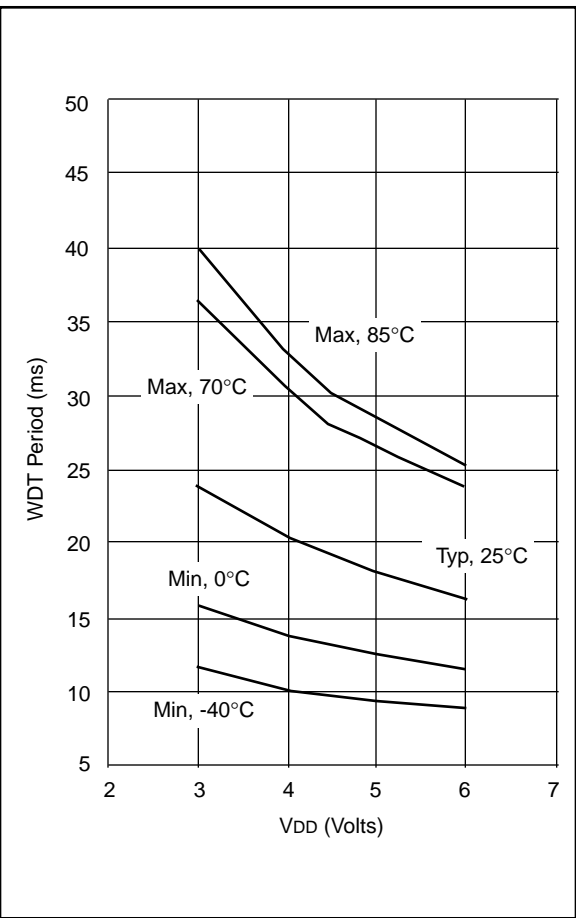
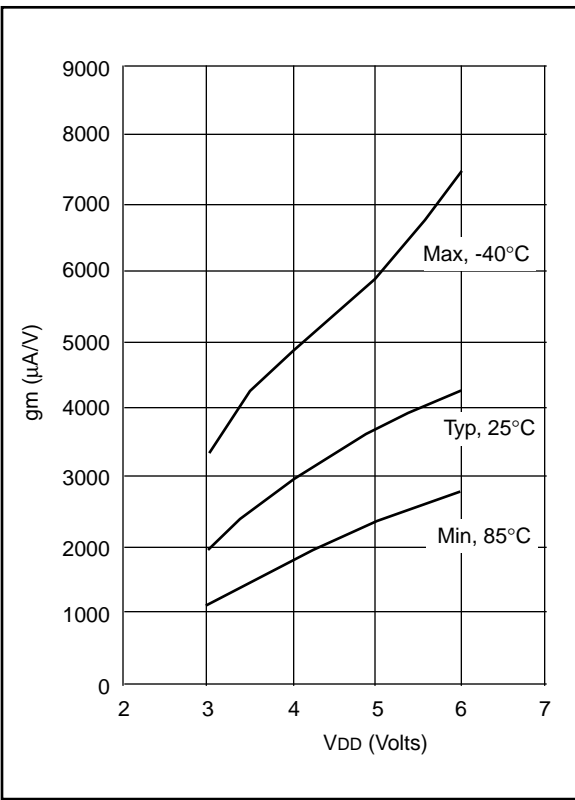


FIGURE 16-16: TRANSCONDUCTANCE (gm) OF HS OSCILLATOR vs. V_{DD}



Data based on matrix samples. See first page of this section for details.