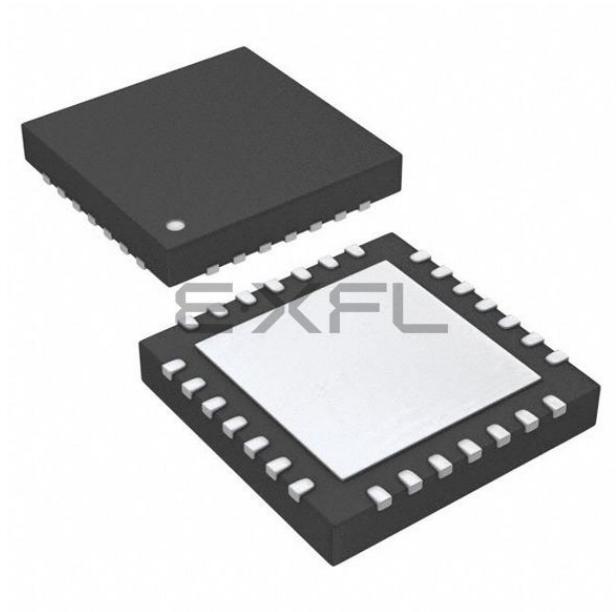


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

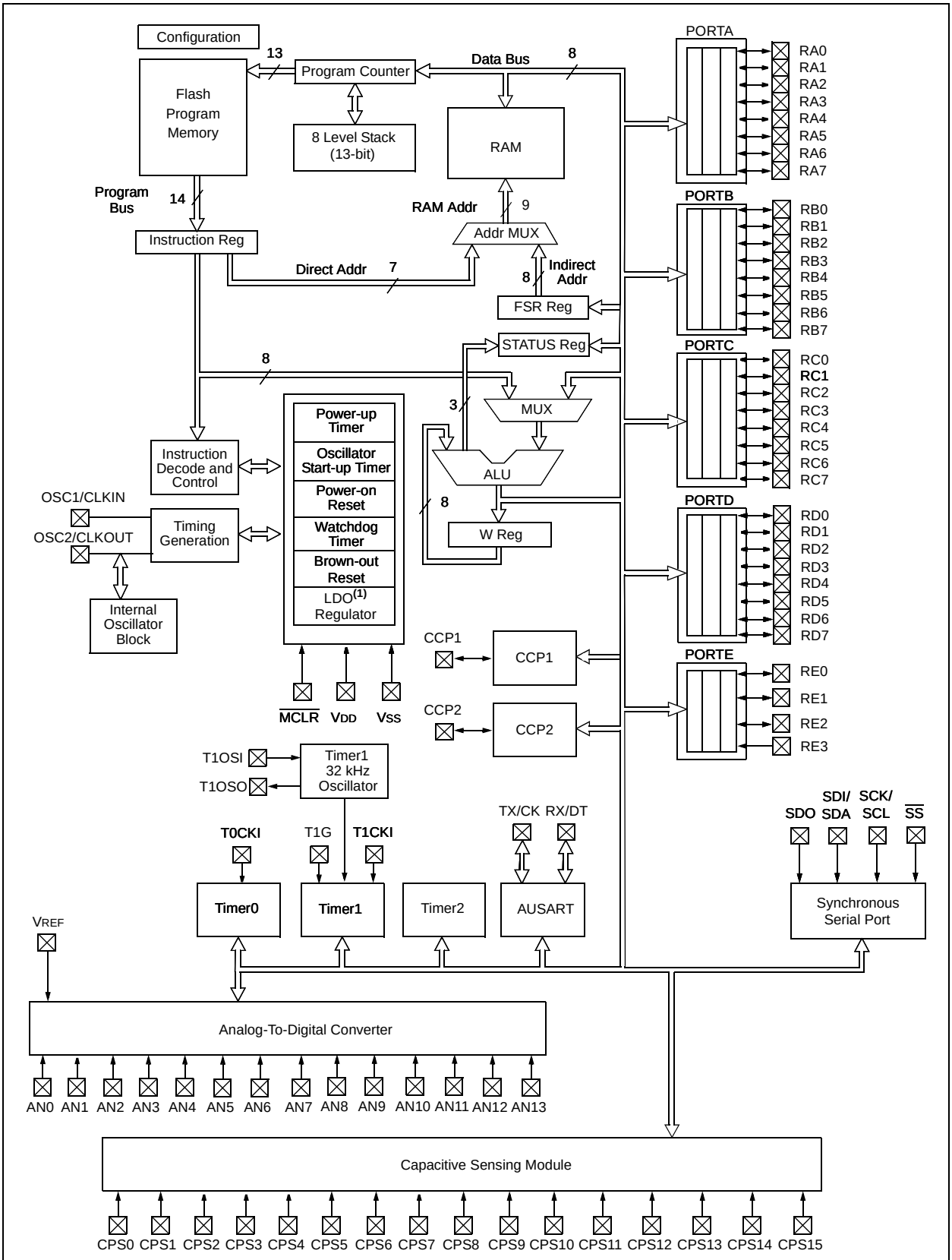
"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	20MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	25
Program Memory Size	14KB (8K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	368 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 11x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-VQFN Exposed Pad
Supplier Device Package	28-QFN (6x6)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf726t-i-ml

Table of Contents

FIGURE 1-2: PIC16F724/727/PIC16LF724/727 BLOCK DIAGRAM

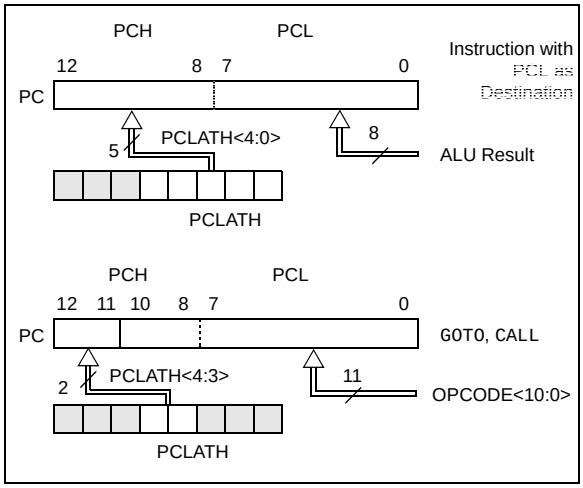


Note 1: PIC16F724/727 only.

2.3 PCL and PCLATH

The Program Counter (PC) is 13 bits wide. The low byte comes from the PCL register, which is a readable and writable register. The high byte (PC<12:8>) is not directly readable or writable and comes from PCLATH. On any Reset, the PC is cleared. Figure 2-7 shows the two situations for the loading of the PC. The upper example in Figure 2-7 shows how the PC is loaded on a write to PCL (PCLATH<4:0> → PCH). The lower example in Figure 2-7 shows how the PC is loaded during a CALL or GOTO instruction (PCLATH<4:3> → PCH).

FIGURE 2-7: LOADING OF PC IN DIFFERENT SITUATIONS



2.3.1 COMPUTED GOTO

A computed GOTO is accomplished by adding an offset to the program counter (ADDWF PCL). When performing a table read using a computed GOTO method, care should be exercised if the table location crosses a PCL memory boundary (each 256-byte block). Refer to the Application Note *AN556, Implementing a Table Read* (DS00556).

2.3.2 STACK

All devices have an 8-level x 13-bit wide hardware stack (refer to Figures 2-1 and 2-3). The stack space is not part of either program or data space and the Stack Pointer is not readable or writable. The PC is PUSHed onto the stack when a CALL instruction is executed or an interrupt causes a branch. The stack is POPed in the event of a RETURN, RETLW or a RETFIE instruction execution. PCLATH is not affected by a PUSH or POP operation.

The stack operates as a circular buffer. This means that after the stack has been PUSHed eight times, the ninth PUSH overwrites the value that was stored from the first PUSH. The tenth PUSH overwrites the second PUSH (and so on).

Note 1:	There are no Status bits to indicate Stack Overflow or Stack Underflow conditions.
2:	There are no instructions/mnemonics called PUSH or POP. These are actions that occur from the execution of the CALL, RETURN, RETLW and RETFIE instructions or the vectoring to an interrupt address.

2.4 Program Memory Paging

All devices are capable of addressing a continuous 8K word block of program memory. The CALL and GOTO instructions provide only 11 bits of address to allow branching within any 2K program memory page. When doing a CALL or GOTO instruction, the upper two bits of the address are provided by PCLATH<4:3>. When doing a CALL or GOTO instruction, the user must ensure that the page select bits are programmed so that the desired program memory page is addressed. If a return from a CALL instruction (or interrupt) is executed, the entire 13-bit PC is POPed off the stack. Therefore, manipulation of the PCLATH<4:3> bits is not required for the RETURN instructions (which POPs the address from the stack).

Note:	The contents of the PCLATH register are unchanged after a RETURN or RETFIE instruction is executed. The user must rewrite the contents of the PCLATH register for any subsequent subroutine calls or GOTO instructions.

Example 2-1 shows the calling of a subroutine in page 1 of the program memory. This example assumes that PCLATH is saved and restored by the Interrupt Service Routine (if interrupts are used).

EXAMPLE 2-1: CALL OF A SUBROUTINE IN PAGE 1 FROM PAGE 0

```
        ORG 500h
        PAGESEL SUB_P1  ;Select page 1
                        ;(800h-FFFh)
        CALL    SUB1_P1 ;Call subroutine in
        :               ;page 1 (800h-FFFh)
        :
        ORG     900h    ;page 1 (800h-FFFh)
SUB1_P1
        :               ;called subroutine
                        ;page 1 (800h-FFFh)
        :
        RETURN          ;return to
                        ;Call subroutine
                        ;in page 0
                        ;(000h-7FFh)
```

2.5 Indirect Addressing, INDF and FSR Registers

The INDF register is not a physical register. Addressing the INDF register will cause indirect addressing.

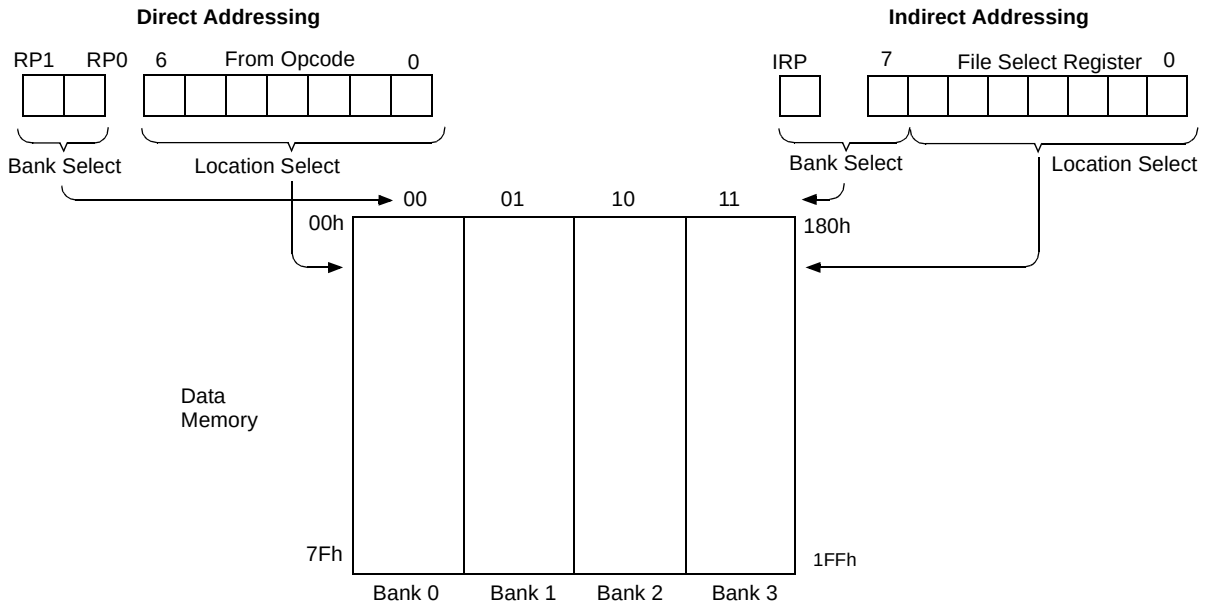
Indirect addressing is possible by using the INDF register. Any instruction using the INDF register actually accesses data pointed to by the File Select Register (FSR). Reading INDF itself indirectly will produce 00h. Writing to the INDF register indirectly results in a no operation (although Status bits may be affected). An effective 9-bit address is obtained by concatenating the 8-bit FSR register and the IRP bit of the STATUS register, as shown in Figure 2-8.

A simple program to clear RAM location 020h-02Fh using indirect addressing is shown in Example 2-2.

EXAMPLE 2-2: INDIRECT ADDRESSING

```
     MOVLW     020h   ;initialize pointer
     MOVWF     FSR    ;to RAM
     BANKISEL  020h
NEXT CLRF      INDF   ;clear INDF register
     INCF      FSR    ;inc pointer
     BTFSS     FSR,4  ;all done?
     GOTO      NEXT   ;no clear next
CONTINUE              ;yes continue
```

FIGURE 2-8: DIRECT/INDIRECT ADDRESSING



Note: For memory map detail, refer to Figures 2-4 and 2-5.

3.6 Time-out Sequence

On power-up, the time-out sequence is as follows: first, PWRT time out is invoked after POR has expired, then OST is activated after the PWRT time out has expired. The total time out will vary based on oscillator configuration and $\overline{\text{PWRTE}}$ bit status. For example, in EC mode with $\overline{\text{PWRTE}}$ bit = 1 (PWRT disabled), there will be no time out at all. Figure 3-4, Figure 3-5 and Figure 3-6 depict time-out sequences.

Since the time outs occur from the POR pulse, if $\overline{\text{MCLR}}$ is kept low long enough, the time outs will expire. Then, bringing $\overline{\text{MCLR}}$ high will begin execution immediately (see Figure 3-5). This is useful for testing purposes or to synchronize more than one PIC16(L)F722/3/4/6/7 device operating in parallel.

Table 3-3 shows the Reset conditions for some special registers.

3.7 Power Control (PCON) Register

The Power Control (PCON) register has two Status bits to indicate what type of Reset that last occurred.

Bit 0 is $\overline{\text{BOR}}$ (Brown-out Reset). $\overline{\text{BOR}}$ is unknown on Power-on Reset. It must then be set by the user and checked on subsequent Resets to see if $\overline{\text{BOR}}$ = 0, indicating that 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 (BOREN<1:0> = 00 in the Configuration Word register).

Bit 1 is $\overline{\text{POR}}$ (Power-on Reset). It is a ‘0’ on Power-on Reset and unaffected otherwise. The user must write a ‘1’ to this bit following a Power-on Reset. On a subsequent Reset, if $\overline{\text{POR}}$ is ‘0’, it will indicate that a Power-on Reset has occurred (i.e., V_{DD} may have gone too low).

For more information, see **Section 3.5 “Brown-Out Reset (BOR)”**.

TABLE 3-2: TIME OUT IN VARIOUS SITUATIONS

Oscillator Configuration	Power-up		Brown-out Reset		Wake-up from Sleep
	$\overline{\text{PWRTE}}$ = 0	$\overline{\text{PWRTE}}$ = 1	$\overline{\text{PWRTE}}$ = 0	$\overline{\text{PWRTE}}$ = 1	
XT, HS, LP(1)	T_{PWRT} + 1024 • T_{OSC}	1024 • T_{OSC}	T_{PWRT} + 1024 • T_{OSC}	1024 • T_{OSC}	1024 • T_{OSC}
RC, EC, INTOSC	T_{PWRT}	—	T_{PWRT}	—	—

Note 1: LP mode with T1OSC disabled.

TABLE 3-3: RESET BITS AND THEIR SIGNIFICANCE

$\overline{\text{POR}}$	$\overline{\text{BOR}}$	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Condition
0	u	1	1	Power-on Reset
1	0	1	1	Brown-out Reset
u	u	0	u	WDT Reset
u	u	0	0	WDT Wake-up
u	u	u	u	$\overline{\text{MCLR}}$ Reset during normal operation
u	u	1	0	$\overline{\text{MCLR}}$ Reset during Sleep

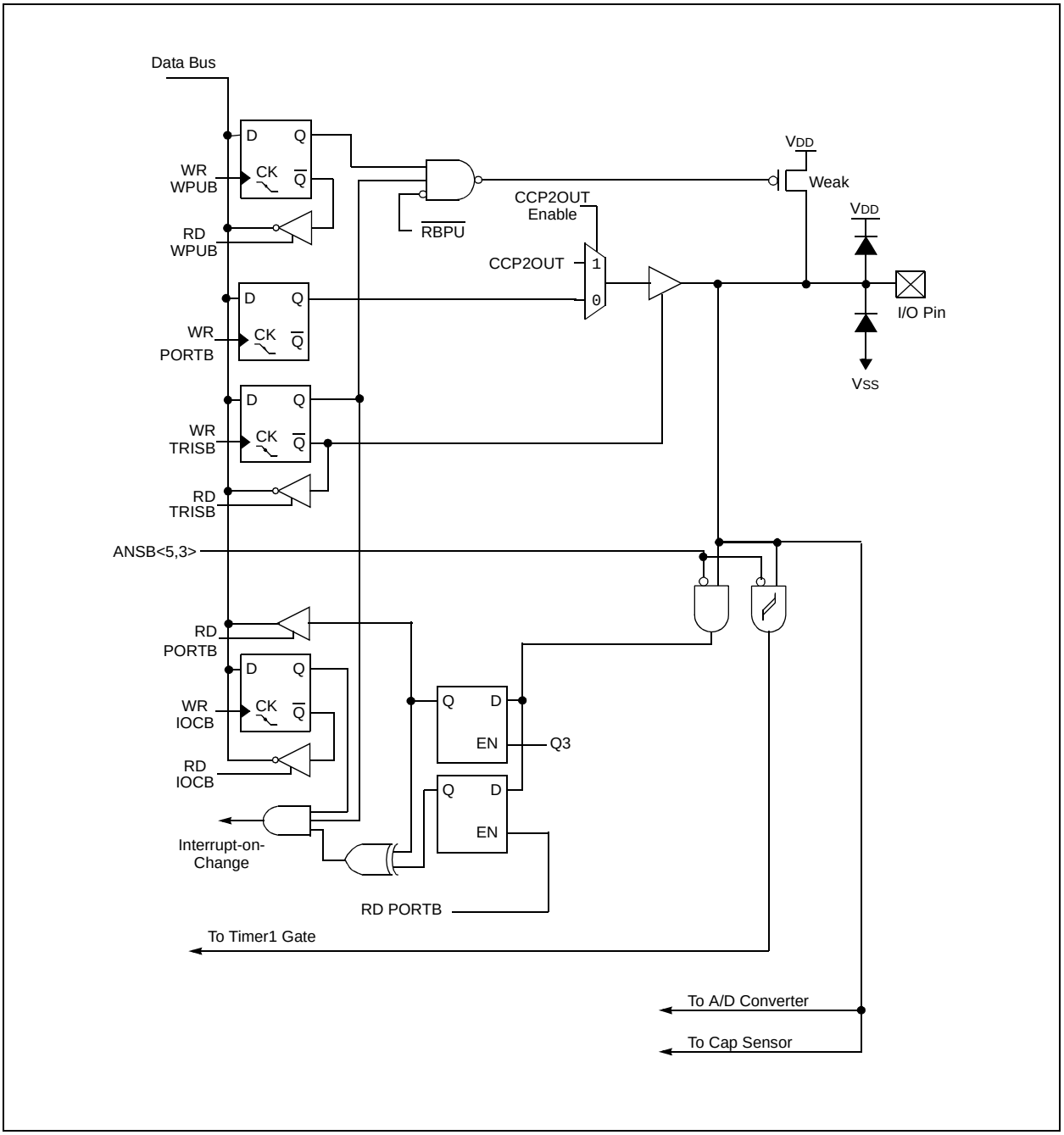
Legend: u = unchanged, x = unknown

TABLE 6-1: SUMMARY OF REGISTERS ASSOCIATED WITH PORTA

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
ADCON0	—	—	CHS3	CHS2	CHS1	CHS0	GO/$\overline{\text{DONE}}$	ADON	0000 0000	0000 0000
ADCON1	—	ADCS2	ADCS1	ADCS0	—	—	ADREF1	ADREF0	-000 --00	-000 --00
ANSELA	—	—	ANSA5	ANSA4	ANSA3	ANSA2	ANSA1	ANSA0	--11 1111	--11 1111
APFCON	—	—	—	—	—	—	SSSEL	CCP2SEL	---- --00	---- --00
CPSCON0	CPSON	—	—	—	CPSRNG1	CPSRNG0	CPSOUT	T0XCS	0--- 0000	0--- 0000
CPSCON1	—	—	—	—	CPSCH3	CPSCH2	CPSCH1	CPSCH0	---- 0000	---- 0000
CONFIG2(1)	—	—	VCAPEN1	VCAPEN0	—	—	—	—	—	—
OPTION_REG	$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
PORTA	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	xxxx xxxx	xxxx xxxx
SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	1111 1111	1111 1111

Legend: x = unknown, u = unchanged, – = unimplemented locations read as '0'. Shaded cells are not used by PORTA.
Note 1: PIC16F72X only.

FIGURE 6-10: BLOCK DIAGRAM OF RB5



9.1 ADC Configuration

When configuring and using the ADC, the following functions must be considered:

- Port configuration
- Channel selection
- ADC voltage reference selection
- ADC conversion clock source
- Interrupt control
- Results formatting

9.1.1 PORT CONFIGURATION

The ADC can be used to convert both analog and digital signals. When converting analog signals, the I/O pin should be configured for analog by setting the associated TRIS and ANSEL bits. Refer to **Section 6.0 "I/O Ports"** for more information.

Note:	Analog voltages on any pin that is defined as a digital input may cause the input buffer to conduct excess current.

9.1.2 CHANNEL SELECTION

The CHS bits of the ADCON0 register determine which channel is connected to the sample and hold circuit.

When changing channels, a delay is required before starting the next conversion. Refer to **Section 9.2 "ADC Operation"** for more information.

9.1.3 ADC VOLTAGE REFERENCE

The ADREF bits of the ADCON1 register provides control of the positive voltage reference. The positive voltage reference can be either V_{DD}, an external voltage source or the internal Fixed Voltage Reference. The negative voltage reference is always connected to the ground reference. See **Section 10.0 "Fixed Voltage Reference"** for more details on the Fixed Voltage Reference.

9.1.4 CONVERSION CLOCK

The source of the conversion clock is software selectable via the ADCS bits of the ADCON1 register. There are seven possible clock options:

- $F_{OSC}/2$
- $F_{OSC}/4$
- $F_{OSC}/8$
- $F_{OSC}/16$
- $F_{OSC}/32$
- $F_{OSC}/64$
- F_{RC} (dedicated internal oscillator)

The time to complete one bit conversion is defined as T_{AD}. One full 8-bit conversion requires 10 T_{AD} periods as shown in Figure 9-2.

For correct conversion, the appropriate T_{AD} specification must be met. Refer to the A/D conversion requirements in **Section 23.0 "Electrical Specifications"** for more information. Table 9-1 gives examples of appropriate ADC clock selections.

Note:	Unless using the F_{RC}, any changes in the system clock frequency will change the ADC clock frequency, which may adversely affect the ADC result.

TABLE 9-1: ADC CLOCK PERIOD (T_{AD}) VS. DEVICE OPERATING FREQUENCIES

ADC Clock Period (T_{AD})		Device Frequency (F_{OSC})				
ADC Clock Source	**ADCS<2:0>**	**20 MHz**	**16 MHz**	**8 MHz**	**4 MHz**	**1 MHz**
Fosc/2	000	100 ns(2)	125 ns(2)	250 ns(2)	500 ns(2)	2.0 μs
Fosc/4	100	200 ns(2)	250 ns(2)	500 ns(2)	1.0 μs	4.0 μs
Fosc/8	001	400 ns(2)	0.5 μs(2)	1.0 μs	2.0 μs	8.0 μs(3)
Fosc/16	101	800 ns	1.0 μs	2.0 μs	4.0 μs	16.0 μs(3)
Fosc/32	010	1.6 μs	2.0 μs	4.0 μs	8.0 μs(3)	32.0 μs(3)
Fosc/64	110	3.2 μs	4.0 μs	8.0 μs(3)	16.0 μs(3)	64.0 μs(3)
FRC	x11	1.0-6.0 μs(1,4)	1.0-6.0 μs(1,4)	1.0-6.0 μs(1,4)	1.0-6.0 μs(1,4)	1.0-6.0 μs(1,4)

Legend: Shaded cells are outside of recommended range.

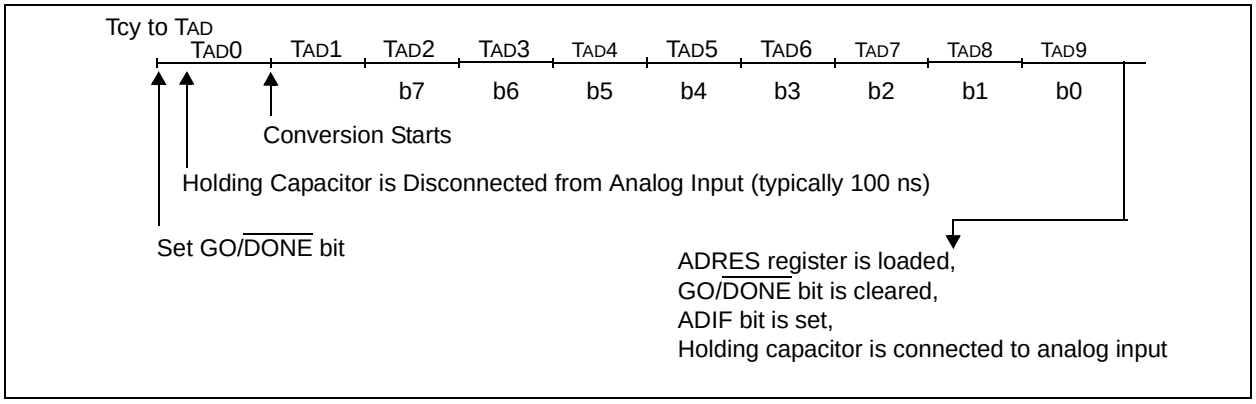
Note 1: The FRC source has a typical T_{AD} time of 1.6 μs for V_{DD}.

2: These values violate the minimum required T_{AD} time.

3: For faster conversion times, the selection of another clock source is recommended.

4: When the device frequency is greater than 1 MHz, the FRC clock source is only recommended if the conversion will be performed during Sleep.

FIGURE 9-2: ANALOG-TO-DIGITAL CONVERSION T_{AD} CYCLES



9.3 A/D Acquisition Requirements

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

EQUATION 9-1: ACQUISITION TIME EXAMPLE

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

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

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

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

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

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

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

Solving for T_C*:*

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

Therefore:

$$\begin{aligned} T_{ACQ} &= 2MS + 1.12MS + [(50°C - 25°C)(0.05MS/°C)] \\ &= 4.42MS \end{aligned}$$

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

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

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

TABLE 12-6: SUMMARY OF REGISTERS ASSOCIATED WITH TIMER1

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
ANSELB	—	—	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1	ANSB0	--11 1111	--11 1111
CCP1CON	—	—	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	--00 0000
CCP2CON	—	—	DC2B1	DC2B0	CCP2M3	CCP2M2	CCP2M1	CCP2M0	--00 0000	--00 0000
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx	xxxx xxxx
TMR1H	Holding Register for the Most Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	uuuu uuuu
TMR1L	Holding Register for the Least Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	uuuu uuuu
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	1111 1111
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111
T1CON	TMR1CS1	TMR1CS0	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{\text{T1SYNC}}$	—	TMR1ON	0000 00-0	uuuu uu-u
T1GCON	TMR1GE	T1GPOL	T1GTM	T1GSPM	T1$\overline{\text{GGO}}$/$\overline{\text{DONE}}$	T1GVAL	T1GSS1	T1GSS0	0000 0x00	uuuu uxuu

Legend: x = unknown, u = unchanged, – = unimplemented, read as '0'. Shaded cells are not used by the Timer1 module.

20.0 IN-CIRCUIT SERIAL PROGRAMMING™ (ICSP™)

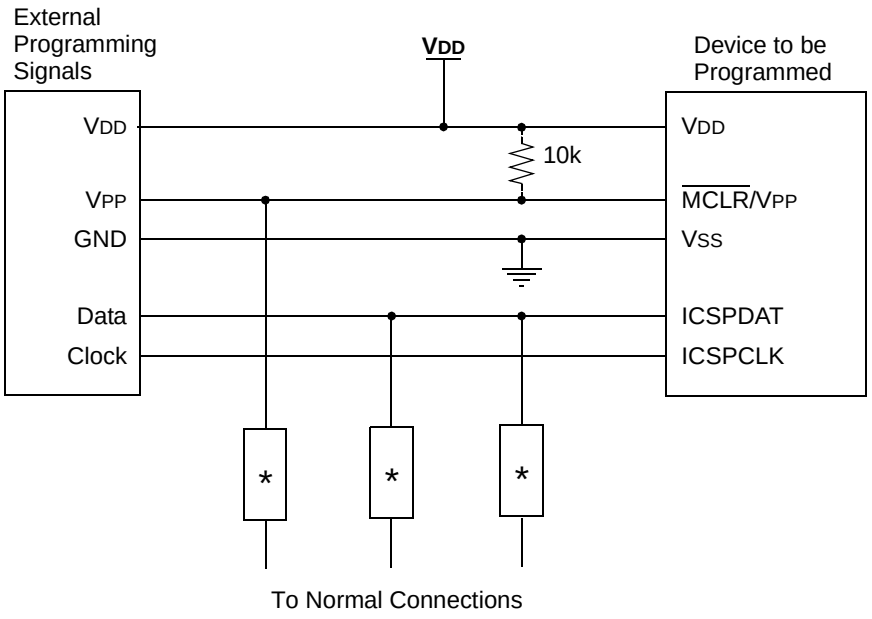
ICSP™ programming allows customers to manufacture circuit boards with unprogrammed devices. Programming can be done after the assembly process allowing the device to be programmed with the most recent firmware or a custom firmware. Five pins are needed for ICSP™ programming:

- ICSPCLK
- ICSPDAT
- MCLR/VPP
- VDD
- VSS

The device is placed into Program/Verify mode by holding the ICSPCLK and ICSPDAT pins low then raising the voltage on MCLR/VPP from 0v to VPP. In Program/Verify mode the program memory, User IDs and the Configuration Words are programmed through serial communications. The ICSPDAT pin is a bidirectional I/O used for transferring the serial data and the ISCPCLK pin is the clock input. For more information on ICSP, refer to the *"PIC16(L)F72x Memory Programming Specification"* (DS41332).

Note:	The ICD 2 produces a VPP voltage greater than the maximum VPP specification of the PIC16(L)F722/3/4/6/7. When using this programmer, an external circuit, such as the AC164112 MPLAB ICD 2 VPP voltage limiter, is required to keep the VPP voltage within the device specifications.

FIGURE 20-1: TYPICAL CONNECTION FOR ICSP™ PROGRAMMING



* Isolation devices (as required).

RLF **Rotate Left f through Carry**

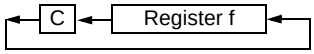
Syntax: [*label*] RLF f,d

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

Operation: See description below

Status Affected: C

Description: The contents of register 'f' are rotated one bit to the left through the Carry flag. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is stored back in register 'f'.



Words: 1

Cycles: 1

Example:

```
RLF     REG1,0
```

Before Instruction

```
REG1    =   1110 0110
C       =   0
```

After Instruction

```
REG1    =   1110 0110
W       =   1100 1100
C       =   1
```

RRF **Rotate Right f through Carry**

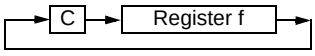
Syntax: [*label*] RRF f,d

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

Operation: See description below

Status Affected: C

Description: The contents of register 'f' are rotated one bit to the right through the Carry flag. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.



SLEEP **Enter Sleep mode**

Syntax: [*label*] SLEEP

Operands: None

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

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

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.

SUBLW **Subtract W from literal**

Syntax: [*label*] SUBLW k

Operands: $0 \le k \le 255$

Operation: k - (W) → (W)

Status Affected: C, DC, Z

Description: The W register is subtracted (2's complement method) from the 8-bit literal 'k'. The result is placed in the W register.

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

SUBWF **Subtract W from f**

Syntax: [*label*] SUBWF f,d

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

Operation: (f) - (W) → (destination)

Status Affected: C, DC, Z

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

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

SWAPF **Swap Nibbles in f**

Syntax: [*label*] SWAPF f,d

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

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

Status Affected: None

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

XORLW **Exclusive OR literal with W**

Syntax: [*label*] XORLW k

Operands: $0 \le k \le 255$

Operation: (W) .XOR. k → (W)

Status Affected: Z

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

XORWF **Exclusive OR W with f**

Syntax: [*label*] XORWF f,d

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

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

Status Affected: Z

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

22.0 DEVELOPMENT SUPPORT

The PIC® microcontrollers (MCU) and dsPIC® digital signal controllers (DSC) are supported with a full range of software and hardware development tools:

- Integrated Development Environment
  - MPLAB® X IDE Software
- Compilers/Assemblers/Linkers
  - MPLAB XC Compiler
  - MPASM™ Assembler
  - MPLINK™ Object Linker/ MPLIB™ Object Librarian
  - MPLAB Assembler/Linker/Librarian for Various Device Families
- Simulators
  - MPLAB X SIM Software Simulator
- Emulators
  - MPLAB REAL ICE™ In-Circuit Emulator
- In-Circuit Debuggers/Programmers
  - MPLAB ICD 3
  - PICkit™ 3
- Device Programmers
  - MPLAB PM3 Device Programmer
- Low-Cost Demonstration/Development Boards, Evaluation Kits and Starter Kits
- Third-party development tools

22.1 MPLAB X Integrated Development Environment Software

The MPLAB X IDE is a single, unified graphical user interface for Microchip and third-party software, and hardware development tool that runs on Windows®, Linux and Mac OS® X. Based on the NetBeans IDE, MPLAB X IDE is an entirely new IDE with a host of free software components and plug-ins for high-performance application development and debugging. Moving between tools and upgrading from software simulators to hardware debugging and programming tools is simple with the seamless user interface.

With complete project management, visual call graphs, a configurable watch window and a feature-rich editor that includes code completion and context menus, MPLAB X IDE is flexible and friendly enough for new users. With the ability to support multiple tools on multiple projects with simultaneous debugging, MPLAB X IDE is also suitable for the needs of experienced users.

Feature-Rich Editor:

- Color syntax highlighting
- Smart code completion makes suggestions and provides hints as you type
- Automatic code formatting based on user-defined rules
- Live parsing

User-Friendly, Customizable Interface:

- Fully customizable interface: toolbars, toolbar buttons, windows, window placement, etc.
- Call graph window

Project-Based Workspaces:

- Multiple projects
- Multiple tools
- Multiple configurations
- Simultaneous debugging sessions

File History and Bug Tracking:

- Local file history feature
- Built-in support for Bugzilla issue tracker

23.3 DC Characteristics: PIC16(L)F722/3/4/6/7-I/E (Power-Down)

PIC16LF722/3/4/6/7			Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended					
PIC16F722/3/4/6/7			**Standard Operating Conditions (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended					
Param No.	**Device Characteristics**	**Min.**	**Typ†**	**Max. +85°C**	**Max. +125°C**	**Units**	**Conditions**	
							VDD	**Note**
	Power-down Base Current (IPD)(2)							
D020		—	0.02	0.7	3.9	µA	1.8	WDT, BOR, FVR, and T1OSC disabled, all Peripherals Inactive
		—	0.08	1.0	4.3	µA	3.0	
D020		—	4.3	10.2	17	µA	1.8	WDT, BOR, FVR, and T1OSC disabled, all Peripherals Inactive
		—	5	10.5	18	µA	3.0	
		—	5.5	11.8	21	µA	5.0	
D021		—	0.5	1.7	4.1	µA	1.8	LPWDT Current **(Note 1)**
		—	0.8	2.5	4.8	µA	3.0	
D021		—	6	13.5	18	µA	1.8	LPWDT Current **(Note 1)**
		—	6.5	14.5	19	µA	3.0	
		—	7.5	16	22	µA	5.0	
D021A		—	8.5	14	19	µA	1.8	FVR current **(Note 1. Note 3)**
		—	8.5	14	20	µA	3.0	
D021A		—	23	44	48	µA	1.8	FVR current **(Note 1, Note 3, Note 5)**
		—	25	45	55	µA	3.0	
		—	26	60	70	µA	5.0	
D022		—	—	—	—	µA	1.8	BOR Current **(Note 1, Note 3)**
		—	7.5	12	22	µA	3.0	
D022		—	—	—	—	µA	1.8	BOR Current **(Note 1, Note 3, Note 5)**
		—	23	42	49	µA	3.0	
		—	25	46	50	µA	5.0	
D026		—	0.6	2	—	µA	1.8	T1OSC Current **(Note 1)**
		—	1.8	3.0	—	µA	3.0	
D026		—	4.5	11.1	—	µA	1.8	T1OSC Current **(Note 1)**
		—	6	12.5	—	µA	3.0	
		—	7	13.5	—	µA	5.0	

† 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: The peripheral current is the sum of the base IDD or IPD and the additional current consumed when this peripheral is enabled. The peripheral Δ current can be determined by subtracting the base IDD or IPD current from this limit. Max values should be used when calculating total current consumption.

2: The power-down current in Sleep mode does not depend on the oscillator type. Power-down current is measured with the part in Sleep mode, with all I/O pins in high-impedance state and tied to VDD.

3: Fixed Voltage Reference is automatically enabled whenever the BOR is enabled

4: A/D oscillator source is FRC

5: 0.1 µF capacitor on VCAP (RA0).

23.5 Thermal Considerations

Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +125°C					
Param No.	**Sym.**	**Characteristic**	**Typ.**	**Units**	**Conditions**
TH01	θJA	Thermal Resistance Junction to Ambient	60	°C/W	28-pin SPDIP package
			80	°C/W	28-pin SOIC package
			90	°C/W	28-pin SSOP package
			27.5	°C/W	28-pin UQFN 4x4mm package
			27.5	°C/W	28-pin QFN 6x6mm package
			47.2	°C/W	40-pin PDIP package
			46	°C/W	44-pin TQFP package
			24.4	°C/W	44-pin QFN 8x8mm package
TH02	θJC	Thermal Resistance Junction to Case	31.4	°C/W	28-pin SPDIP package
			24	°C/W	28-pin SOIC package
			24	°C/W	28-pin SSOP package
			24	°C/W	28-pin UQFN 4x4mm package
			24	°C/W	28-pin QFN 6x6mm package
			24.7	°C/W	40-pin PDIP package
			14.5	°C/W	44-pin TQFP package
			20	°C/W	44-pin QFN 8x8mm package
TH03	TJMAX	Maximum Junction Temperature	150	°C	
TH04	PD	Power Dissipation	—	W	PD = PINTERNAL + PI/O
TH05	PINTERNAL	Internal Power Dissipation	—	W	PINTERNAL = IDD x VDD(1)
TH06	PI/O	I/O Power Dissipation	—	W	PI/O = Σ (IOL * VOL) + Σ (IOH * (VDD - VOH))
TH07	PDER	Derated Power	—	W	PDER = PDMAX (TJ - TA)/θJA(2)

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

FIGURE 24-49: TTL INPUT THRESHOLD VIN vs. VDD OVER TEMPERATURE

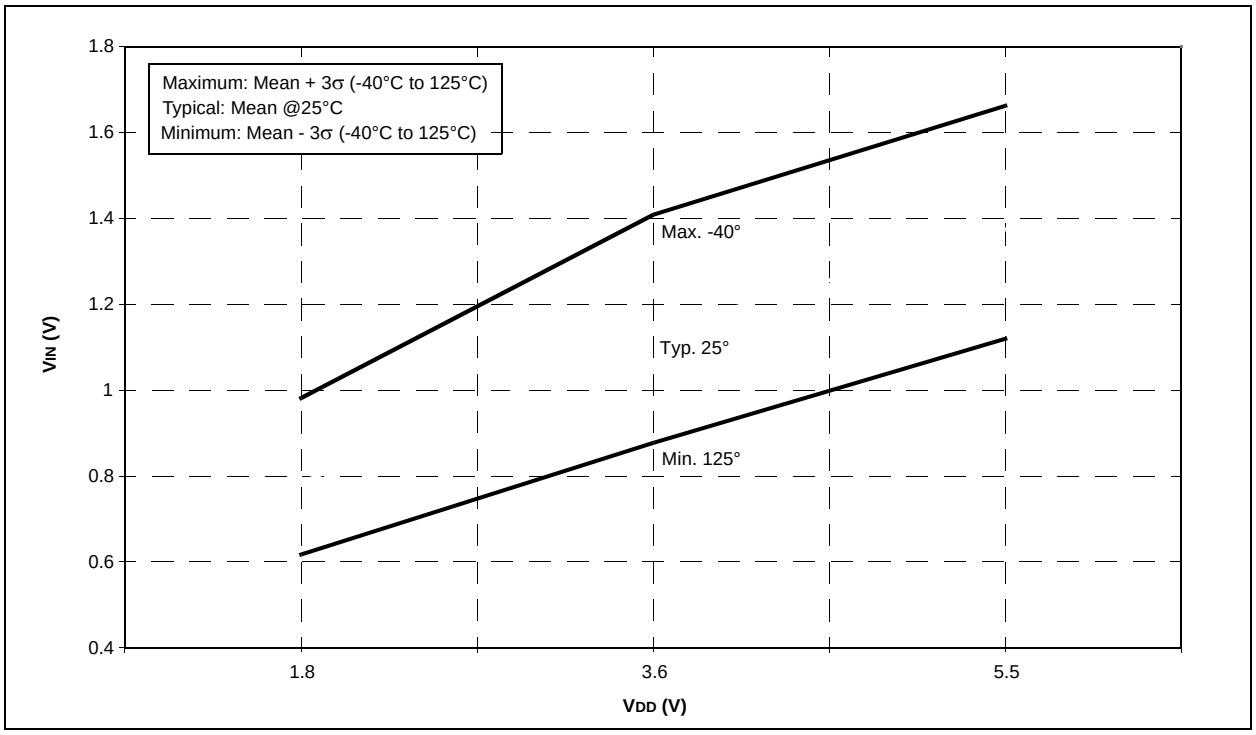


FIGURE 24-50: SCHMITT TRIGGER INPUT THRESHOLD VIN vs. VDD OVER TEMPERATURE

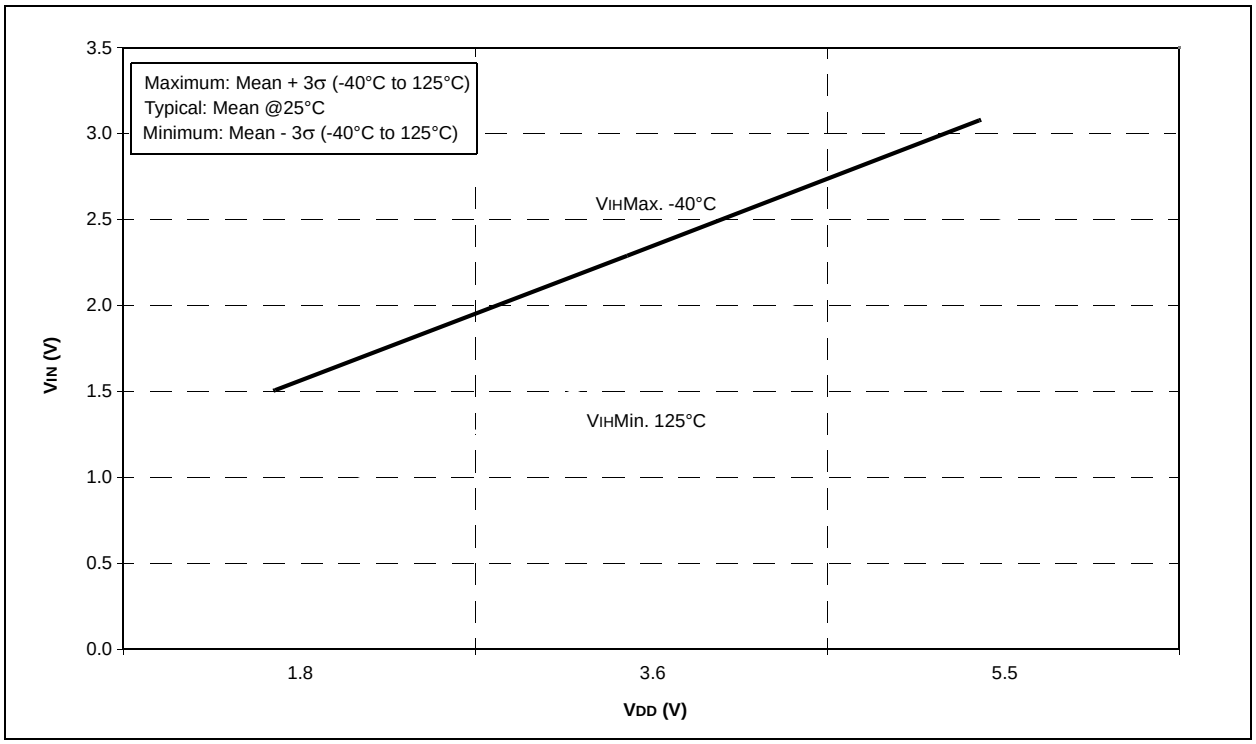


FIGURE 24-61: PIC16F722/3/4/6/7 A/D INTERNAL RC OSCILLATOR PERIOD

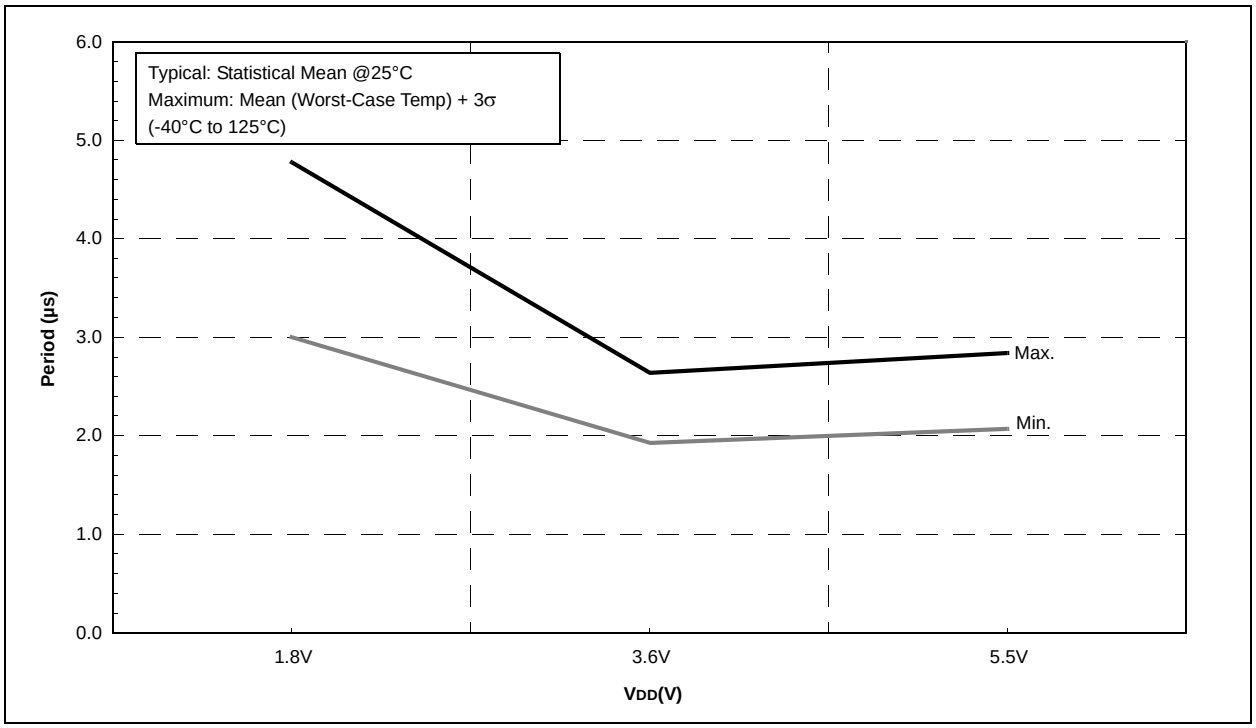
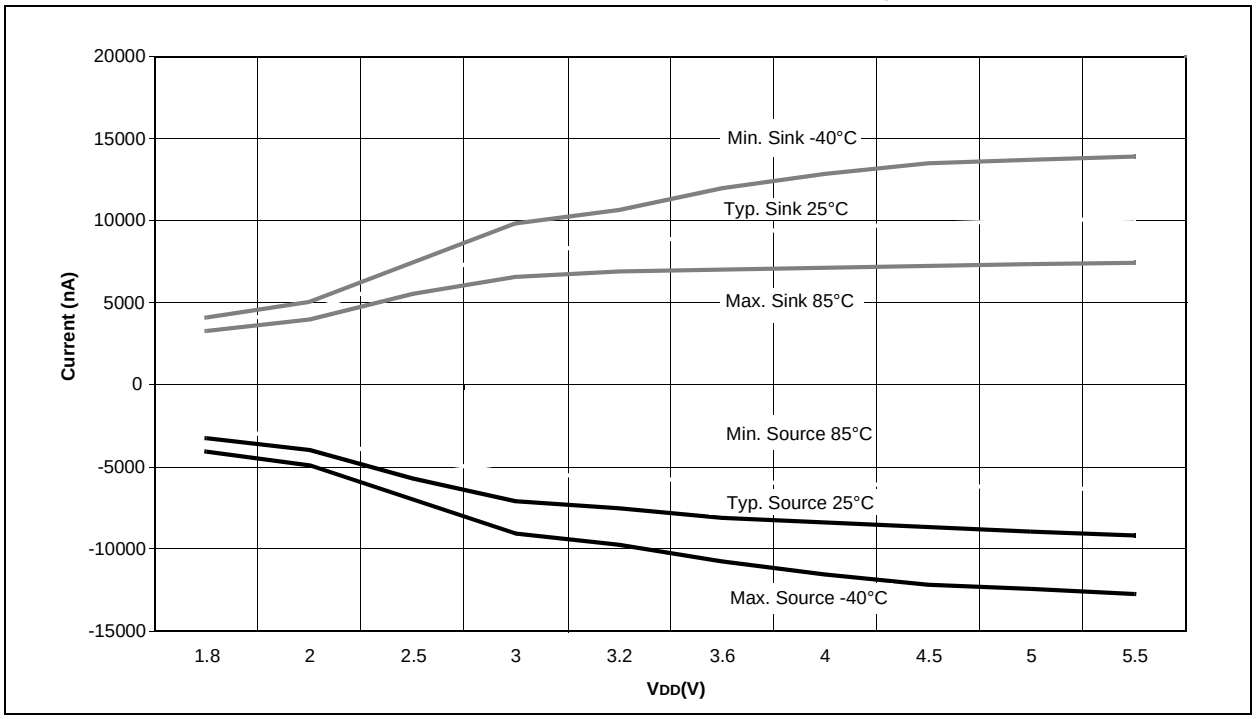


FIGURE 24-62: PIC16F722/3/4/6/7 CAP SENSE OUTPUT CURRENT, POWER MODE = HIGH



APPENDIX A: DATA SHEET REVISION HISTORY

Revision A (12/2007)

Original release.

Revision B (08/2008)

Electrical Specification updates; Package Drawings; miscellaneous updates.

Revision C (04/2009)

Revised data sheet title; Revised Low-Power Features section; Revised Section 6.2.2.4 RA3/AN3/VREF; Revised Figure 16-8 Synchronous Reception.

Revision D (07/2009)

Removed the Preliminary Label; Updated the “Electrical Characteristics” section; Added charts in the “Char. Data” section; Deleted “Based 8-Bit CMOS” from title; Updated the “Special Microcontroller Features” section and the “Peripheral Features” section; Changed the title of the “Low Power Features” section into “Extreme Low-Power Management PIC16LF72X with nanoWatt XLP” and updated this section; Inserted new section – “Analog Features” (page 1); Changed the title of the “Peripheral Features” section into “Peripheral Highlights” and updated the section.

Revision E (10/2009)

Added paragraph to section 5.0 (LDO Voltage Regulator); Updated the Electrical Specifications section (Added another absolute Maximum Rating; Updated section 23.1 and Table 23-4); Updated the Pin Diagrams with the UQFN package; Updated Table 1, adding UQFN; Updated section 23.5 (Thermal Considerations); Updated the Packaging Information section adding the UQFN Package; Updated the Product Identification System section.

Revision F (12/2015)

Updated Table 2; Updated 23.1, 23.3 and 9.2.4 Sections; Updated Figure 23-9; Other minor corrections.

APPENDIX B: MIGRATING FROM OTHER PIC® DEVICES

This discusses some of the issues in migrating from other PIC® devices to the PIC16F72X family of devices.

B.1 PIC16F77 to PIC16F72X

TABLE B-1: FEATURE COMPARISON

Feature	PIC16F77	PIC16F727
Max. Operating Speed	20 MHz	20 MHz
Max. Program Memory (Words)	8K	8K
Max. SRAM (Bytes)	368	368
A/D Resolution	8-bit	8-bit
Timers (8/16-bit)	2/1	2/1
Oscillator Modes	4	8
Brown-out Reset	Y	Y
Internal Pull-ups	RB<7:0>	RB<7:0>
Interrupt-on-change	RB<7:4>	RB<7:0>
Comparator	0	0
USART	Y	Y
Extended WDT	N	N
Software Control Option of WDT/BOR	N	N
INTOSC Frequencies	None	500 kHz - 16 MHz
Clock Switching	N	N