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	8MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	66
Program Memory Size	16KB (8K x 16)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	678 x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	A/D 16x10b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	84-LCC (J-Lead)
Supplier Device Package	84-PLCC (29.31x29.31)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic17lc762-08i-l

Table of Contents

TO OUR VALUED CUSTOMERS

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

If you have any questions or comments regarding this publication, please contact the Marketing Communications Department via E-mail at **docerrors@mail.microchip.com** or fax the **Reader Response Form** in the back of this data sheet to (480) 792-4150. We welcome your feedback.

Most Current Data Sheet

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

http://www.microchip.com

You can determine the version of a data sheet by examining its literature number found on the bottom outside corner of any page. The last character of the literature number is the version number, (e.g., DS30000A is version A of document DS30000).

Errata

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

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

- Microchip's Worldwide Web site; http://www.microchip.com
- Your local Microchip sales office (see last page)
- The Microchip Corporate Literature Center; U.S. FAX: (480) 792-7277

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

Customer Notification System

Register on our web site at **www.microchip.com/cn** to receive the most current information on all of our products.

NOTES:

5.1.4 TIME-OUT SEQUENCE

On power-up, the time-out sequence is as follows: First, the internal POR signal goes high when the POR trip point is reached. If MCLR is high, then both the OST and PWRT timers start. In general, the PWRT time-out is longer, except with low frequency crystals/resonators. The total time-out also varies based on oscillator configuration. Table 5-1 shows the times that are associated with the oscillator configuration. Figure 5-5 and Figure 5-6 display these time-out sequences.

If the device voltage is not within electrical specification at the end of a time-out, the MCLR/VPP pin must be held low until the voltage is within the device specification. The use of an external RC delay is sufficient for many of these applications.

The time-out sequence begins from the first rising edge of MCLR.

Table 5-3 shows the RESET conditions for some special registers, while Table 5-4 shows the initialization conditions for all the registers.

TABLE 5-1: TIME-OUT IN VARIOUS SITUATIONS

Oscillator Configuration	POR, BOR	Wake-up from SLEEP	MCLR Reset
XT, LF	Greater of: 96 ms or 1024TOSC	1024TOSC	—
EC, RC	Greater of: 96 ms or 1024TOSC	—	—

TABLE 5-2: STATUS BITS AND THEIR SIGNIFICANCE

POR	BOR(1)	TO	PD	Event
0	0	1	1	Power-on Reset
1	1	1	0	MCLR Reset during SLEEP or interrupt wake-up from SLEEP
1	1	0	1	WDT Reset during normal operation
1	1	0	0	WDT Wake-up during SLEEP
1	1	1	1	MCLR Reset during normal operation
1	0	1	1	Brown-out Reset
0	0	0	x	Illegal, TO is set on POR
0	0	x	0	Illegal, PD is set on POR
x	x	1	1	CLRWDT instruction executed

Note 1: When BODEN is enabled, else the BOR status bit is unknown.

TABLE 5-3: RESET CONDITION FOR THE PROGRAM COUNTER AND THE CPUSTA REGISTER

Event		PCH:PCL	CPUSTA(4)	OST Active
Power-on Reset		0000h	--11 1100	Yes
Brown-out Reset		0000h	--11 1110	Yes
MCLR Reset during normal operation		0000h	--11 1111	No
MCLR Reset during SLEEP		0000h	--11 1011	Yes(2)
WDT Reset during normal operation		0000h	--11 0111	No
WDT Reset during SLEEP(3)		0000h	--11 0011	Yes(2)
Interrupt Wake-up from SLEEP	GLINTD is set	PC + 1	--11 1011	Yes(2)
	GLINTD is clear	PC + 1(1)	--10 1011	Yes(2)

Legend: u = unchanged, x = unknown, - = unimplemented, read as '0'

Note 1: On wake-up, this instruction is executed. The instruction at the appropriate interrupt vector is fetched and then executed.

2: The OST is only active (on wake-up) when the oscillator is configured for XT or LF modes.

3: The Program Counter = 0; that is, the device branches to the RESET vector and places SFRs in WDT Reset states. This is different from the mid-range devices.

4: When BODEN is enabled, else the BOR status bit is unknown.

6.4 Interrupt Operation

Global Interrupt Disable bit, GLINTD (CPUSTA<4>), enables all unmasked interrupts (if clear), or disables all interrupts (if set). Individual interrupts can be disabled through their corresponding enable bits in the INTSTA register. Peripheral interrupts need either the global peripheral enable PEIE bit disabled, or the specific peripheral enable bit disabled. Disabling the peripherals via the global peripheral enable bit, disables all peripheral interrupts. GLINTD is set on RESET (interrupts disabled).

The `RETFIE` instruction clears the GLINTD bit while forcing the Program Counter (PC) to the value loaded at the Top-of-Stack.

When an interrupt is responded to, the GLINTD bit is automatically set to disable any further interrupt, the return address is pushed onto the stack and the PC is loaded with the interrupt vector. There are four interrupt vectors which help reduce interrupt latency.

The peripheral interrupt vector has multiple interrupt sources. Once in the peripheral Interrupt Service Routine, the source(s) of the interrupt can be determined by polling the interrupt flag bits. The peripheral interrupt flag bit(s) must be cleared in software before re-enabling interrupts to avoid continuous interrupts.

The PIC17C7XX devices have four interrupt vectors. These vectors and their hardware priority are shown in Table 6-1. If two enabled interrupts occur “at the same time”, the interrupt of the highest priority will be serviced first. This means that the vector address of that interrupt will be loaded into the program counter (PC).

TABLE 6-1: INTERRUPT VECTORS/ PRIORITIES

Address	Vector	Priority
0008h	External Interrupt on RA0/ INT pin (INTF)	1 (Highest)
0010h	TMR0 Overflow Interrupt (T0IF)	2
0018h	External Interrupt on T0CKI (T0CKIF)	3
0020h	Peripherals (PEIF)	4 (Lowest)

Note 1: Individual interrupt flag bits are set, regardless of the status of their corresponding mask bit or the GLINTD bit.

2: Before disabling any of the INTSTA enable bits, the GLINTD bit should be set (disabled).

6.5 RA0/INT Interrupt

The external interrupt on the RA0/INT pin is edge triggered. Either the rising edge if the INTEDG bit (T0STA<7>) is set, or the falling edge if the INTEDG bit is clear. When a valid edge appears on the RA0/INT pin, the INTF bit (INTSTA<4>) is set. This interrupt can be disabled by clearing the INTE control bit (INTSTA<0>). The INT interrupt can wake the processor from SLEEP. See Section 17.4 for details on SLEEP operation.

6.6 T0CKI Interrupt

The external interrupt on the RA1/T0CKI pin is edge triggered. Either the rising edge if the T0SE bit (T0STA<6>) is set, or the falling edge if the T0SE bit is clear. When a valid edge appears on the RA1/T0CKI pin, the T0CKIF bit (INTSTA<6>) is set. This interrupt can be disabled by clearing the T0CKIE control bit (INTSTA<2>). The T0CKI interrupt can wake up the processor from SLEEP. See Section 17.4 for details on SLEEP operation.

6.7 Peripheral Interrupt

The peripheral interrupt flag indicates that at least one of the peripheral interrupts occurred (PEIF is set). The PEIF bit is a read only bit and is a bit wise OR of all the flag bits in the PIR registers AND’d with the corresponding enable bits in the PIE registers. Some of the peripheral interrupts can wake the processor from SLEEP. See Section 17.4 for details on SLEEP operation.

6.8 Context Saving During Interrupts

During an interrupt, only the returned PC value is saved on the stack. Typically, users may wish to save key registers during an interrupt; e.g. WREG, ALUSTA and the BSR registers. This requires implementation in software.

Example 6-2 shows the saving and restoring of information for an Interrupt Service Routine. This is for a simple interrupt scheme, where only one interrupt may occur at a time (no interrupt nesting). The SFRs are stored in the non-banked GPR area.

Example 6-2 shows the saving and restoring of information for a more complex Interrupt Service Routine. This is useful where nesting of interrupts is required. A maximum of 6 levels can be done by this example. The BSR is stored in the non-banked GPR area, while the other registers would be stored in a particular bank. Therefore, 6 saves may be done with this routine (since there are 6 non-banked GPR registers). These routines require a dedicated indirect addressing register, FSR0, to be selected for this.

The PUSH and POP code segments could either be in each Interrupt Service Routine, or could be subroutines that were called. Depending on the application, other registers may also need to be saved.

10.0 I/O PORTS

PIC17C75X devices have seven I/O ports, PORTA through PORTG. PIC17C76X devices have nine I/O ports, PORTA through PORTJ. PORTB through PORTJ have a corresponding Data Direction Register (DDR), which is used to configure the port pins as inputs or outputs. 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, PORTB, PORTE<3>, PORTF, PORTG and the upper four bits of PORTH are multiplexed with the peripheral features of the device. These peripheral features are:

- Timer Modules
- Capture Modules
- PWM Modules
- USART/SCI Modules
- SSP Module
- A/D 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
- SSP 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 bits 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.

When the device enters the "RESET state", the Data Direction registers (DDR) are forced set, which will make the I/O hi-impedance inputs. The RESET state of some peripheral modules may force the I/O to other operations, such as analog inputs or the system bus.

10.4 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 DDRD register configures the corresponding port pin as an output. Reading PORTD reads the status of the pins, whereas writing to PORTD 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 Specifications section.

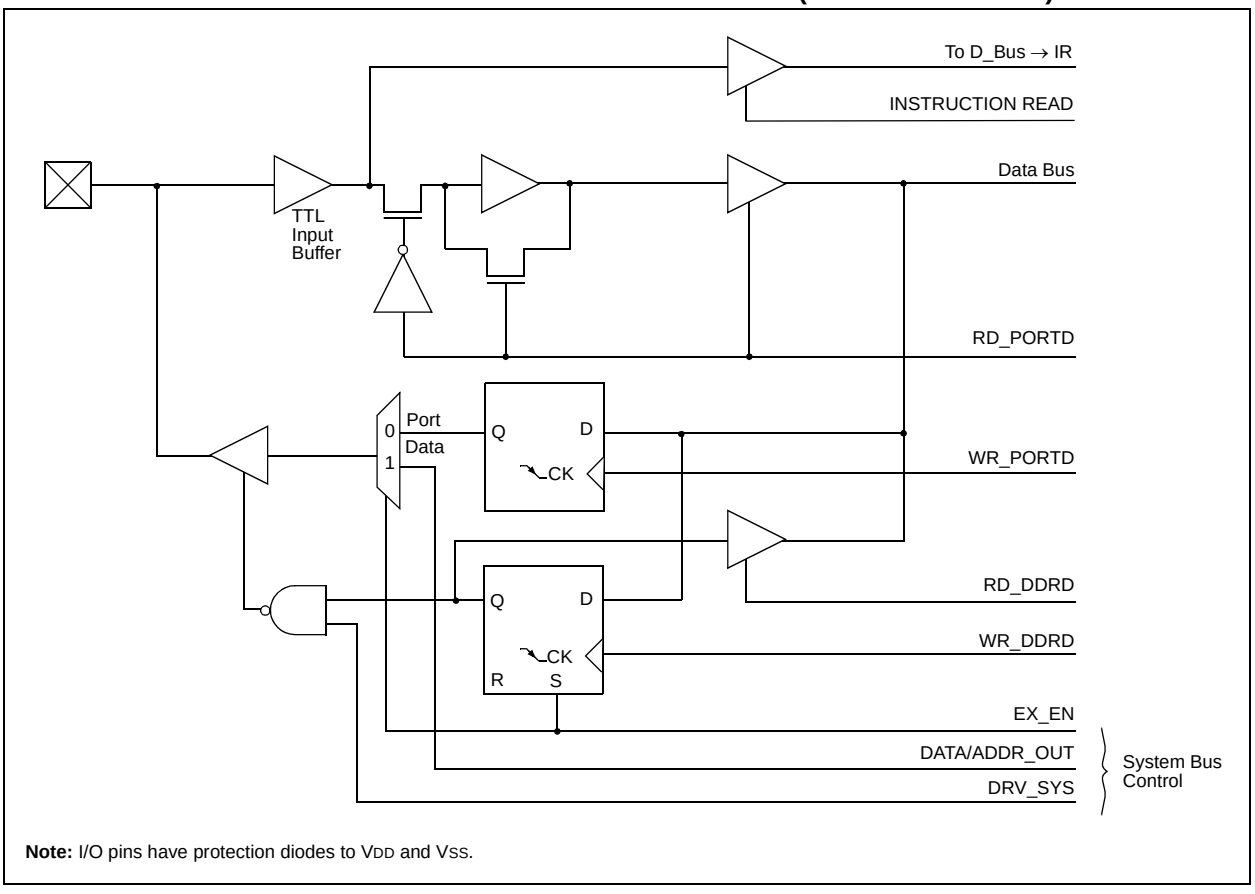
Note:	This port is configured as the system bus when the device's configuration bits are selected to Microprocessor or Extended Microcontroller modes. In the two other microcontroller modes, this port is a general purpose I/O.

Example 10-4 shows an instruction sequence to initialize PORTD. The Bank Select Register (BSR) must be selected to Bank 1 for the port to be initialized. The following example uses the `MOVLB` instruction to load the BSR register for bank selection.

EXAMPLE 10-4: INITIALIZING PORTD

```
MOVLB   1          ; Select Bank 1
CLRF    PORTD, F   ; Initialize PORTD data
                   ; latches before setting
                   ; the data direction reg
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 10-10: BLOCK DIAGRAM OF RD7:RD0 PORT PINS (IN I/O PORT MODE)



Note: I/O pins have protection diodes to VDD and VSS.

TABLE 10-13: PORTG FUNCTIONS

Name	Bit	Buffer Type	Function
RG0/AN3	bit0	ST	Input/output or analog input 3.
RG1/AN2	bit1	ST	Input/output or analog input 2.
RG2/AN1/V_{REF-}	bit2	ST	Input/output or analog input 1 or the ground reference voltage.
RG3/AN0/V_{REF+}	bit3	ST	Input/output or analog input 0 or the positive reference voltage.
RG4/CAP3	bit4	ST	Input/output or the Capture3 input pin.
RG5/PWM3	bit5	ST	Input/output or the PWM3 output pin.
RG6/RX2/DT2	bit6	ST	Input/output or the USART2 (SCI) Asynchronous Receive or USART2 (SCI) Synchronous Data.
RG7/TX2/CK2	bit7	ST	Input/output or the USART2 (SCI) Asynchronous Transmit or USART2 (SCI) Synchronous Clock.

Legend: ST = Schmitt Trigger input

TABLE 10-14: REGISTERS/BITS ASSOCIATED WITH PORTG

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	$\overline{MCLR}$, WDT
12h, Bank 5	DDRG	Data Direction Register for PORTG								1111 1111	1111 1111
13h, Bank 5	PORTG	RG7/ TX2/CK2	RG6/ RX2/DT2	RG5/ PWM3	RG4/ CAP3	RG3/ AN0	RG2/ AN1	RG1/ AN2	RG0/ AN3	xxxx 0000	uuuu 0000
15h, Bank 5	ADCON1	ADCS1	ADCS0	ADFM	—	PCFG3	PCFG2	PCFG1	PCFG0	000- 0000	000- 0000

Legend: x = unknown, u = unchanged, - = unimplemented, read as '0'. Shaded cells are not used by PORTG.

15.2.1.1 Addressing

Once the MSSP module has been enabled, it waits for a START condition to occur. Following the START condition, the 8-bits are shifted into the SSPSR register. All incoming bits are sampled with the rising edge of the clock (SCL) line. The value of register SSPSR<7:1> is compared to the value of the SSPADD register. The address is compared on the falling edge of the eighth clock (SCL) pulse. If the addresses match and the BF and SSPOV bits are clear, the following events occur:

a) The SSPSR register value is loaded into the SSPBUF register on the falling edge of the 8th SCL pulse.
b) The buffer full bit, BF, is set on the falling edge of the 8th SCL pulse.
c) An $\overline{ACK}$ pulse is generated.
d) SSP interrupt flag bit, SSPIF (PIR2<7>), is set (interrupt is generated if enabled) - on the falling edge of the 9th SCL pulse.

In 10-bit address mode, two address bytes need to be received by the slave. The five Most Significant bits (MSbs) of the first address byte specify if this is a 10-bit address. Bit R/$\overline{W}$ (SSPSTAT<2>) must specify a write so the slave device will receive the second address byte. For a 10-bit address, the first byte would equal '1111 0 A9 A8 0', where A9 and A8 are the two MSbs of the address. The sequence of events for a 10-bit address is as follows, with steps 7- 9 for slave-transmitter:

1. Receive first (high) byte of Address (bits SSPIF, BF and bit UA (SSPSTAT<1>) are set).
2. Update the SSPADD register with second (low) byte of Address (clears bit UA and releases the SCL line).
3. Read the SSPBUF register (clears bit BF) and clear flag bit SSPIF.
4. Receive second (low) byte of Address (bits SSPIF, BF and UA are set).
5. Update the SSPADD register with the first (high) byte of Address. This will clear bit UA and release the SCL line.
6. Read the SSPBUF register (clears bit BF) and clear flag bit SSPIF.
7. Receive Repeated Start condition.
8. Receive first (high) byte of Address (bits SSPIF and BF are set).
9. Read the SSPBUF register (clears bit BF) and clear flag bit SSPIF.

Note:	Following the Repeated Start condition (step 7) in 10-bit mode, the user only needs to match the first 7-bit address. The user does not update the SSPADD for the second half of the address.

15.2.1.2 Slave Reception

When the R/$\overline{W}$ bit of the address byte is clear and an address match occurs, the R/$\overline{W}$ bit of the SSPSTAT register is cleared. The received address is loaded into the SSPBUF register.

When the address byte overflow condition exists, then no acknowledge ($\overline{ACK}$) pulse is given. An overflow condition is defined as either bit BF (SSPSTAT<0>) is set, or bit SSPOV (SSPCON1<6>) is set.

An SSP interrupt is generated for each data transfer byte. Flag bit SSPIF (PIR2<7>) must be cleared in software. The SSPSTAT register is used to determine the status of the received byte.

Note:	The SSPBUF will be loaded if the SSPOV bit is set and the BF flag is cleared. If a read of the SSPBUF was performed, but the user did not clear the state of the SSPOV bit before the next receive occurred, the $\overline{ACK}$ is not sent and the SSPBUF is updated.

TABLE 15-2: DATA TRANSFER RECEIVED BYTE ACTIONS

Status Bits as Data Transfer is Received		SSPSR → SSPBUF	Generate $\overline{ACK}$ Pulse	Set bit SSPIF (SSP Interrupt occurs if enabled)
BF	SSPOV			
0	0	Yes	Yes	Yes
1	0	No	No	Yes
1	1	No	No	Yes
0	1	Yes	No	Yes

Note 1: Shaded cells show the conditions where the user software did not properly clear the overflow condition.

15.2.10 I²C MASTER MODE REPEATED START CONDITION TIMING

A Repeated Start condition occurs when the RSEN bit (SSPCON2<1>) is programmed high and the I²C module is in the idle state. When the RSEN bit is set, the SCL pin is asserted low. When the SCL pin is sampled low, the baud rate generator is loaded with the contents of SSPADD<6:0> and begins counting. The SDA pin is released (brought high) for one baud rate generator count (TBRG). When the baud rate generator times out, if SDA is sampled high, the SCL pin will be de-asserted (brought high). When SCL is sampled high the baud rate generator is reloaded with the contents of SSPADD<6:0> and begins counting. SDA and SCL must be sampled high for one TBRG. This action is then followed by assertion of the SDA pin (SDA is low) for one TBRG while SCL is high. Following this, the RSEN bit in the SSPCON2 register will be automatically cleared and the baud rate generator is not reloaded, leaving the SDA pin held low. As soon as a START condition is detected on the SDA and SCL pins, the S bit (SSPSTAT<3>) will be set. The SSPIF bit will not be set until the baud rate generator has timed out.

Note 1: If the RSEN is programmed while any other event is in progress, it will not take effect.

2: A bus collision during the Repeated Start condition occurs if:

- SDA is sampled low when SCL goes from low to high.
- SCL goes low before SDA is asserted low. This may indicate that another master is attempting to transmit a data "1".

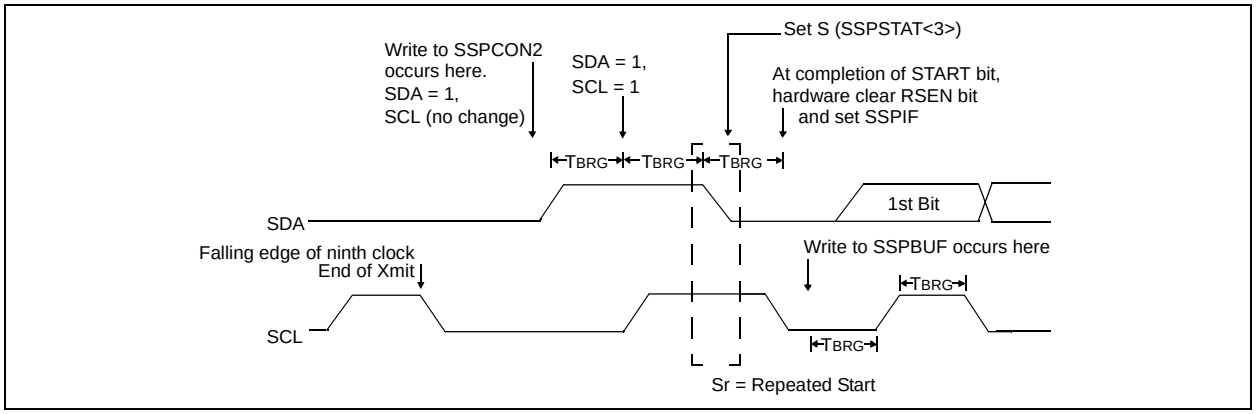
Immediately following the SSPIF bit getting set, the user may write the SSPBUF with the 7-bit address in 7-bit mode, or the default first address in 10-bit mode. After the first eight bits are transmitted and an ACK is received, the user may then transmit an additional eight bits of address (10-bit mode), or eight bits of data (7-bit mode).

15.2.10.1 WCOL status flag

If the user writes the SSPBUF when a Repeated Start sequence is in progress, then WCOL is set and the contents of the buffer are unchanged (the write doesn't occur).

Note: Because queueing of events is not allowed, writing of the lower 5 bits of SSPCON2 is disabled until the Repeated Start condition is complete.

FIGURE 15-22: REPEAT START CONDITION WAVEFORM



CLRWDT **Clear Watchdog Timer**

Syntax: [*label*] CLRWDT

Operands: None

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

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

Encoding:

0000	0000	0000	0100

Description: CLRWDT instruction resets the Watchdog Timer. It also resets the postscaler of the WDT. Status bits $\overline{\text{TO}}$ and $\overline{\text{PD}}$ are set.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	No operation	Process Data	No operation

Example: CLRWDT

Before Instruction
WDT counter = ?

After Instruction
WDT counter = 0x00
$\overline{\text{WDT}}$ Postscaler = 0
$\overline{\text{TO}}$ = 1
$\overline{\text{PD}}$ = 1

COMF **Complement f**

Syntax: [*label*] COMF f,d

Operands: $0 \le f \le 255$
$d \in [0,1]$

Operation: $(\bar{f}) \rightarrow$ (dest)

Status Affected: Z

Encoding:

0001	001d	ffff	ffff

Description: The contents of register 'f' are complemented. If 'd' is 0 the result is stored in WREG. If 'd' is 1 the result is stored back in register 'f'.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	Write to destination

Example: COMF REG1,0

Before Instruction
REG1 = 0x13

After Instruction
REG1 = 0x13
WREG = 0xEC

TABLRD **Table Read**

Example1: `TABLRD 1, 1, REG ;`

Before Instruction

REG	=	0x53
TBLATH	=	0xAA
TBLATL	=	0x55
TBLPTR	=	0xA356
MEMORY(TBLPTR)	=	0x1234

After Instruction (table write completion)

REG	=	0xAA
TBLATH	=	0x12
TBLATL	=	0x34
TBLPTR	=	0xA357
MEMORY(TBLPTR)	=	0x5678

Example2: `TABLRD 0, 0, REG ;`

Before Instruction

REG	=	0x53
TBLATH	=	0xAA
TBLATL	=	0x55
TBLPTR	=	0xA356
MEMORY(TBLPTR)	=	0x1234

After Instruction (table write completion)

REG	=	0x55
TBLATH	=	0x12
TBLATL	=	0x34
TBLPTR	=	0xA356
MEMORY(TBLPTR)	=	0x1234

TABLWT **Table Write**

Syntax: [*label*] TABLWT t,i,f

Operands: $0 \le f \le 255$
$i \in [0,1]$
$t \in [0,1]$

Operation: If t = 0,
f → TBLATL;
If t = 1,
f → TBLATH;
TBLAT → Prog Mem (TBLPTR);
If i = 1,
TBLPTR + 1 → TBLPTR
If i = 0,
TBLPTR is unchanged

Status Affected: None

Encoding:

1010	11ti	ffff	ffff

Description:

1. Load value in 'f' into 16-bit table latch (TBLAT)
   If t = 1: load into high byte;
   If t = 0: load into low byte
2. The contents of TBLAT are written to the program memory location pointed to by TBLPTR.
   If TBLPTR points to external program memory location, then the instruction takes two-cycle.
   If TBLPTR points to an internal EPROM location, then the instruction is terminated when an interrupt is received.

Note: The $\overline{MCLR}/V_{PP}$ pin must be at the programming voltage for successful programming of internal memory.
If $\overline{MCLR}/V_{PP} = V_{DD}$
the programming sequence of internal memory will be interrupted. A short write will occur (2 T_{CY}). The internal memory location will not be affected.

3. The TBLPTR can be automatically incremented
   If i = 1; TBLPTR is not incremented
   If i = 0; TBLPTR is incremented

Words: 1

Cycles: 2 (many if write is to on-chip EPROM program memory)

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	Write register TBLATH or TBLATL
No operation	No operation (Table Pointer on Address bus)	No operation	No operation (Table Latch on Address bus, $\overline{WR}$ goes low)

TLWT **Table Latch Write**

Syntax: [*label*] TLWT t,f

Operands: $0 \le f \le 255$
$t \in [0,1]$

Operation: If t = 0,
f → TBLATL;
If t = 1,
f → TBLATH

Status Affected: None

Encoding:

1010	01tx	ffff	ffff

Description: Data from file register 'f' is written into the 16-bit table latch (TBLAT).
If t = 1; high byte is written
If t = 0; low byte is written
This instruction is used in conjunction with TABLWT to transfer data from data memory to program memory.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	Write register TBLATH or TBLATL

Example: TLWT t, RAM

Before Instruction
t = 0
RAM = 0xB7
TBLAT = 0x0000 (TBLATH = 0x00) (TBLATL = 0x00)

After Instruction
RAM = 0xB7
TBLAT = 0x00B7 (TBLATH = 0x00) (TBLATL = 0xB7)

Before Instruction
t = 1
RAM = 0xB7
TBLAT = 0x0000 (TBLATH = 0x00) (TBLATL = 0x00)

After Instruction
RAM = 0xB7
TBLAT = 0xB700 (TBLATH = 0xB7) (TBLATL = 0x00)

TSTFSZ **Test f, skip if 0**

Syntax: [*label*] TSTFSZ f

Operands: $0 \le f \le 255$

Operation: skip if f = 0

Status Affected: None

Encoding:

0011	0011	ffff	ffff

Description: If 'f' = 0, the next instruction, fetched during the current instruction execution, is discarded and a NOP is executed, making this a two-cycle instruction.

Words: 1

Cycles: 1 (2)

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	No operation

If skip:

Q1	Q2	Q3	Q4
No operation	No operation	No operation	No operation

Example:
```
HERE    TSTFSZ  CNT
NZERO     :
ZERO      :
```

Before Instruction
PC = Address (HERE)

After Instruction
If CNT = 0x00,
PC = Address (ZERO)
If CNT ¼ 0x00,
PC = Address (NZERO)

19.8 MPLAB ICD In-Circuit Debugger

Microchip's In-Circuit Debugger, MPLAB ICD, is a powerful, low cost, run-time development tool. This tool is based on the FLASH PIC16F87X and can be used to develop for this and other PIC microcontrollers from the PIC16CXXX family. The MPLAB ICD utilizes the in-circuit debugging capability built into the PIC16F87X. This feature, along with Microchip's In-Circuit Serial Programming™ protocol, offers cost-effective in-circuit FLASH debugging from the graphical user interface of the MPLAB Integrated Development Environment. This enables a designer to develop and debug source code by watching variables, single-stepping and setting break points. Running at full speed enables testing hardware in real-time.

19.9 PRO MATE II Universal Device Programmer

The PRO MATE II universal device programmer is a full-featured programmer, capable of operating in stand-alone mode, as well as PC-hosted mode. The PRO MATE II device programmer is CE compliant.

The PRO MATE II device programmer has programmable V_{DD} and V_{PP} supplies, which allow it to verify programmed memory at V_{DD} min and V_{DD} max for maximum reliability. It has an LCD display for instructions and error messages, keys to enter commands and a modular detachable socket assembly to support various package types. In stand-alone mode, the PRO MATE II device programmer can read, verify, or program PIC MCU devices. It can also set code protection in this mode.

19.10 PICSTART Plus Entry Level Development Programmer

The PICSTART Plus development programmer is an easy-to-use, low cost, prototype programmer. It connects to the PC via a COM (RS-232) port. MPLAB Integrated Development Environment software makes using the programmer simple and efficient.

The PICSTART Plus development programmer supports all PIC devices with up to 40 pins. Larger pin count devices, such as the PIC16C92X and PIC17C76X, may be supported with an adapter socket. The PICSTART Plus development programmer is CE compliant.

19.11 PICDEM 1 Low Cost PIC MCU Demonstration Board

The PICDEM 1 demonstration board is a simple board which demonstrates the capabilities of several of Microchip's microcontrollers. The microcontrollers supported are: PIC16C5X (PIC16C54 to PIC16C58A), PIC16C61, PIC16C62X, PIC16C71, PIC16C8X, PIC17C42, PIC17C43 and PIC17C44. All necessary hardware and software is included to run basic demo programs. The user can program the sample microcontrollers provided with the PICDEM 1 demonstration board on a PRO MATE II device programmer, or a PICSTART Plus development programmer, and easily test firmware. The user can also connect the PICDEM 1 demonstration board to the MPLAB ICE in-circuit emulator and download the firmware to the emulator for testing. A prototype area is available for the user to build some additional hardware and connect it to the microcontroller socket(s). Some of the features include an RS-232 interface, a potentiometer for simulated analog input, push button switches and eight LEDs connected to PORTB.

19.12 PICDEM 2 Low Cost PIC16CXX Demonstration Board

The PICDEM 2 demonstration board is a simple demonstration board that supports the PIC16C62, PIC16C64, PIC16C65, PIC16C73 and PIC16C74 microcontrollers. All the necessary hardware and software is included to run the basic demonstration programs. The user can program the sample microcontrollers provided with the PICDEM 2 demonstration board on a PRO MATE II device programmer, or a PICSTART Plus development programmer, and easily test firmware. The MPLAB ICE in-circuit emulator may also be used with the PICDEM 2 demonstration board to test firmware. A prototype area has been provided to the user for adding additional hardware and connecting it to the microcontroller socket(s). Some of the features include a RS-232 interface, push button switches, a potentiometer for simulated analog input, a serial EEPROM to demonstrate usage of the I^2C™ bus and separate headers for connection to an LCD module and a keypad.

PIC17LC7XX-08 (Commercial, Industrial)			**Standard Operating Conditions (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +85°C for industrial and 0°C ≤ TA ≤ +70°C for commercial				
PIC17C7XX-16 (Commercial, Industrial, Extended) **PIC17C7XX-33** (Commercial, Industrial, Extended)			**Standard Operating Conditions (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +125°C for extended -40°C ≤ TA ≤ +85°C for industrial 0°C ≤ TA ≤ +70°C for commercial				
Param. No.	**Sym**	**Characteristic**	**Min**	**Typ†**	**Max**	**Units**	**Conditions**
D010	IDD	**Supply Current (Note 2)**					
		PIC17LC7XX	—	3	6	mA	FOSC = 4 MHz **(Note 4)**
D010		PIC17C7XX	—	3	6	mA	FOSC = 4 MHz **(Note 4)**
D011		PIC17LC7XX	—	5	10	mA	FOSC = 8 MHz
D011		PIC17C7XX	—	5	10	mA	FOSC = 8 MHz
D012			—	9	18	mA	FOSC = 16 MHz
D014		PIC17LC7XX	—	85	150	µA	FOSC = 32 kHz, (EC osc configuration)
D015		PIC17C7XX	—	15	30	mA	FOSC = 33 MHz
D021	IPD	**Power-down Current (Note 3)**					
		PIC17LC7XX	—	<1	5	µA	VDD = 3.0V, WDT disabled
D021 (commercial, industrial)		PIC17C7XX	—	<1	20	µA	VDD = 5.5V, WDT disabled
D021A (extended)			—	2	20	µA	VDD = 5.5V, WDT disabled
		Module Differential Current					
D023	ΔIBOR	BOR circuitry	–	75	150	µA	VDD = 4.5V, BODEN enabled
D024	ΔIWDT	Watchdog Timer	–	10	35	µA	VDD = 5.5V
D026	ΔIAD	A/D converter	–	1	–	µA	VDD = 5.5V, A/D not converting

† Data in "Typ" column is at 5V, 25°C unless otherwise stated.

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

2: The supply current is mainly a function of the operating voltage and frequency. Other factors such as I/O pin loading and switching rate, oscillator type, internal code execution pattern and temperature also have an impact on the current consumption.

The test conditions for all IDD measurements in active operation mode are:

OSC1 = external square wave, from rail to rail; all I/O pins tri-stated, pulled to VDD or VSS, T0CKI = VDD, MCLR = VDD; WDT disabled.

Current consumed from the oscillator and I/O's driving external capacitive or resistive loads needs to be considered.

For the RC oscillator, the current through the external pull-up resistor (R) can be estimated as: VDD/(2 • R).

For capacitive loads, the current can be estimated (for an individual I/O pin) as (CL • VDD) • f

CL = Total capacitive load on the I/O pin; f = average frequency the I/O pin switches.

The capacitive currents are most significant when the device is configured for external execution (includes Extended Microcontroller mode).

3: 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 hi-impedance state and tied to VDD or VSS.

4: For RC osc configuration, current through REXT is not included. The current through the resistor can be estimated by the formula IR = VDD/2REXT (mA) with REXT in kOhm.

5: This is the voltage where the device enters the Brown-out Reset. When BOR is enabled, the device (-16) will operate correctly to this trip point.

FIGURE 20-20: USART SYNCHRONOUS RECEIVE (MASTER/SLAVE) TIMING

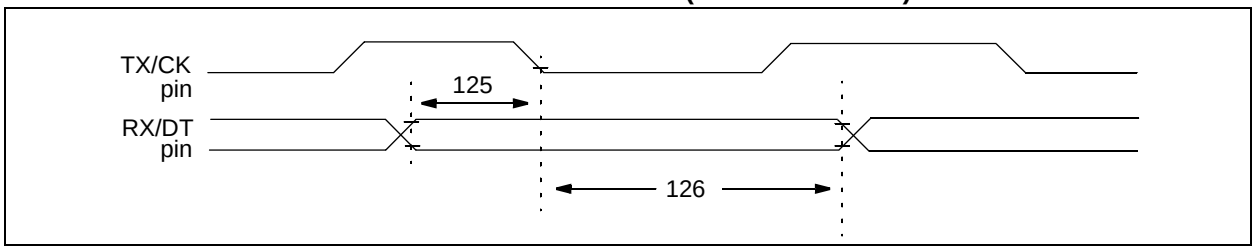


TABLE 20-15: USART SYNCHRONOUS RECEIVE REQUIREMENTS

Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
125	TdtV2ckL	SYNC RCV (MASTER & SLAVE) Data setup before CK↓ (DT setup time)	15	—	—	ns	
126	TckL2dtl	Data hold after CK↓ (DT hold time)	15	—	—	ns	

† Data in “Typ” column is at 5V, 25°C unless otherwise stated.

FIGURE 21-19: TYPICAL, MINIMUM AND MAXIMUM VOH vs. IOH (VDD = 3V, -40°C TO +125°C)

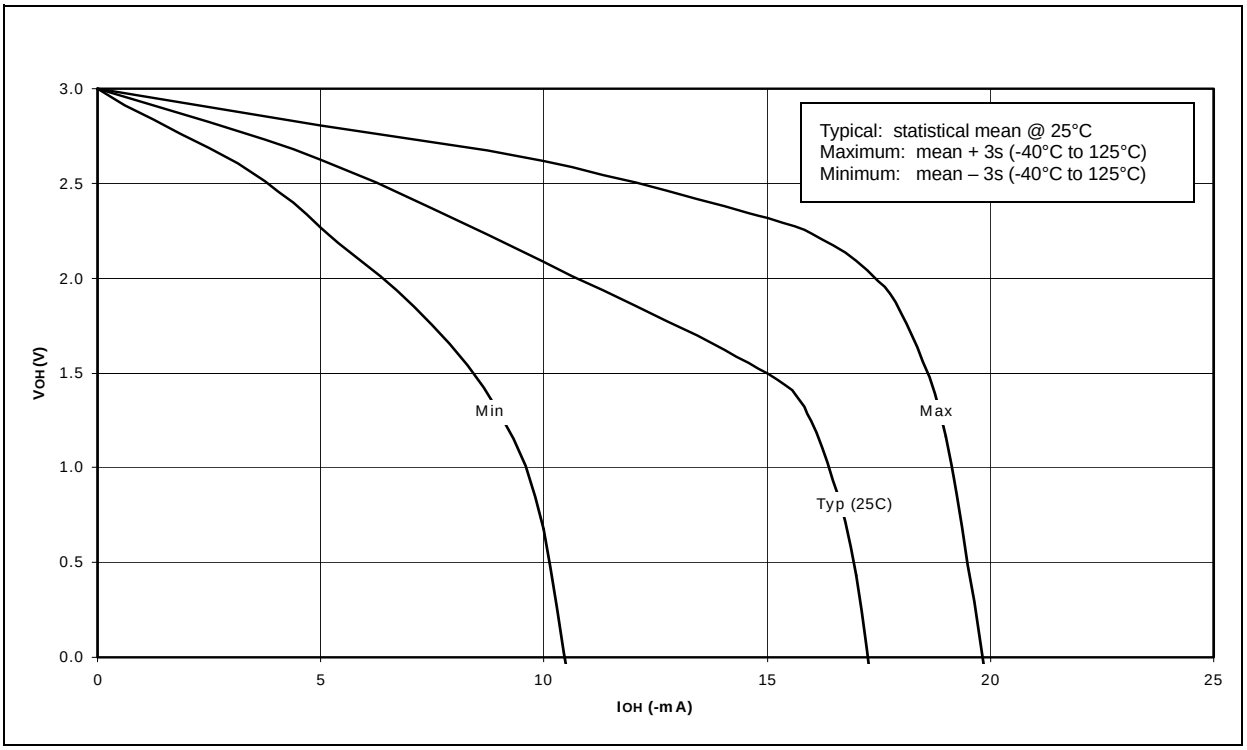
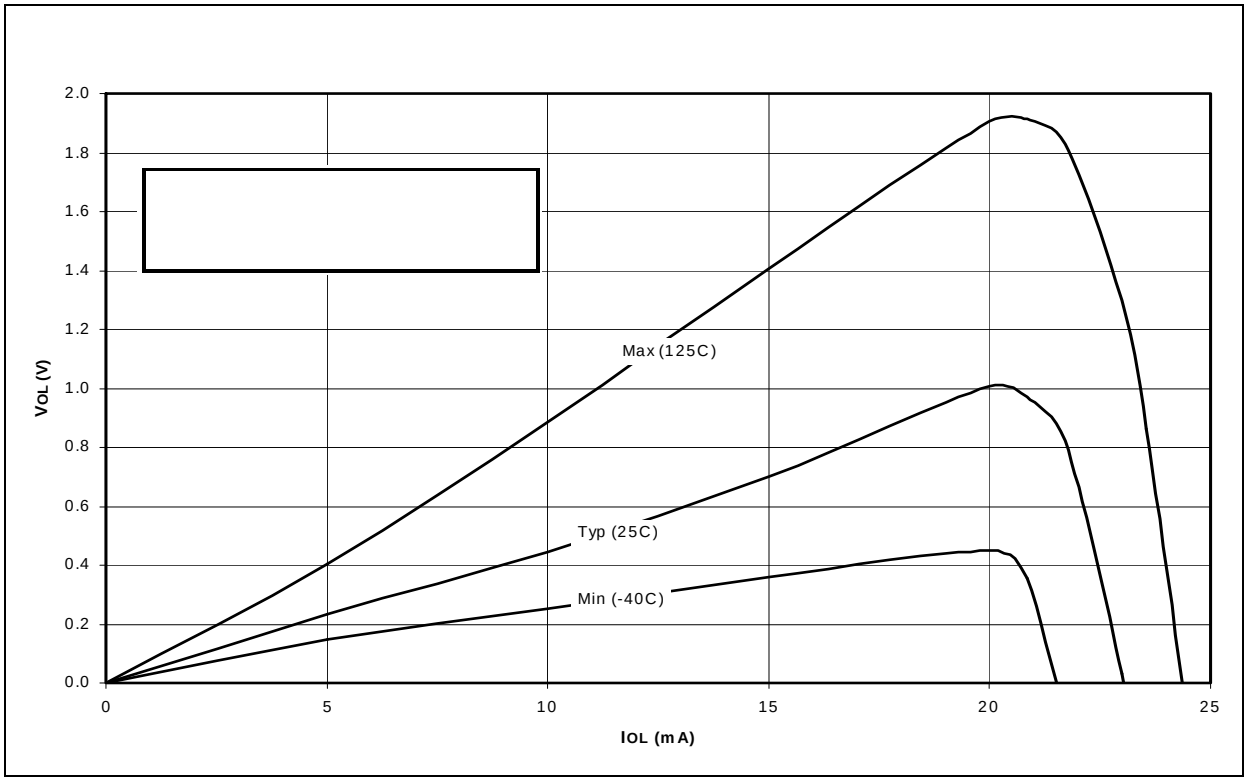
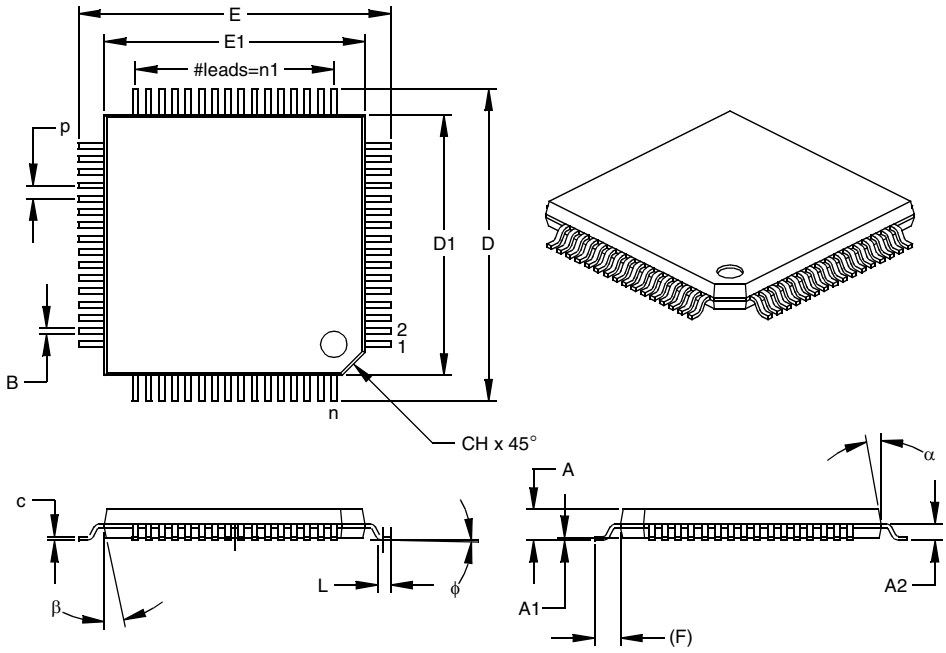


FIGURE 21-20: TYPICAL, MINIMUM AND MAXIMUM VOL vs. IOL (VDD = 3V, -40°C TO +125°C)



64-Lead Plastic Thin Quad Flatpack (PT) 10x10x1 mm Body, 1.0/0.10 mm Lead Form (TQFP)

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



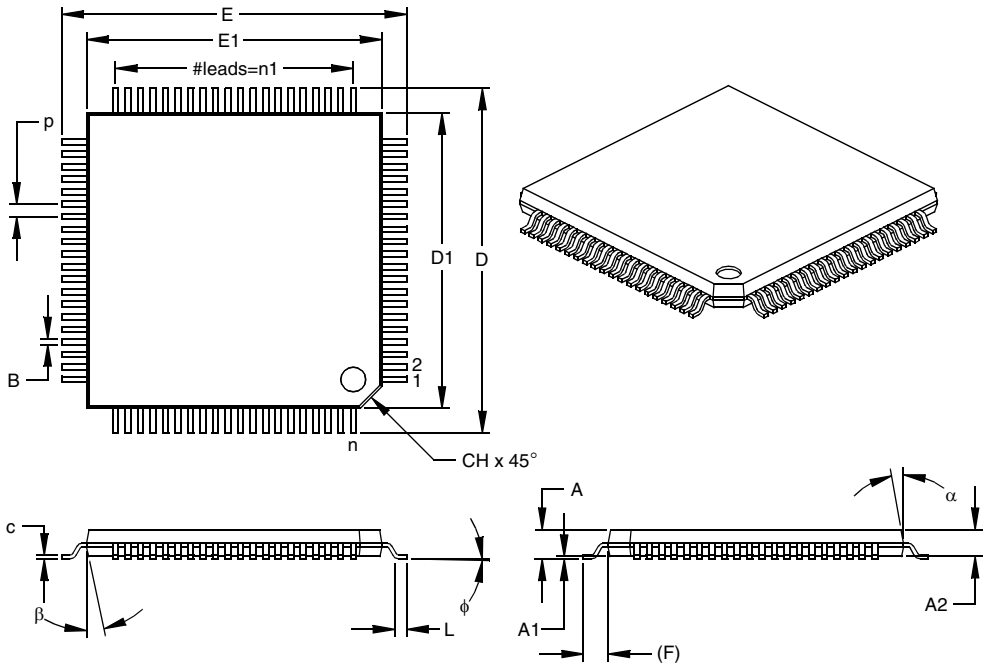
Units		INCHES			MILLIMETERS*		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		64			64	
Pitch	p		.020			0.50	
Pins per Side	n1		16			16	
Overall Height	A	.039	.043	.047	1.00	1.10	1.20
Molded Package Thickness	A2	.037	.039	.041	0.95	1.00	1.05
Standoff §	A1	.002	.006	.010	0.05	0.15	0.25
Foot Length	L	.018	.024	.030	0.45	0.60	0.75
Footprint (Reference)	(F)		.039			1.00	
Foot Angle	ϕ	0	3.5	7	0	3.5	7
Overall Width	E	.463	.472	.482	11.75	12.00	12.25
Overall Length	D	.463	.472	.482	11.75	12.00	12.25
Molded Package Width	E1	.390	.394	.398	9.90	10.00	10.10
Molded Package Length	D1	.390	.394	.398	9.90	10.00	10.10
Lead Thickness	c	.005	.007	.009	0.13	0.18	0.23
Lead Width	B	.007	.009	.011	0.17	0.22	0.27
Pin 1 Corner Chamfer	CH	.025	.035	.045	0.64	0.89	1.14
Mold Draft Angle Top	α	5	10	15	5	10	15
Mold Draft Angle Bottom	β	5	10	15	5	10	15

* Controlling Parameter
§ Significant Characteristic

Notes:
Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010” (0.254mm) per side.
JEDEC Equivalent: MS-026
Drawing No. C04-085

80-Lead Plastic Thin Quad Flatpack (PT) 12x12x1 mm Body, 1.0/0.10 mm Lead Form (TQFP)

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Units		INCHES			MILLIMETERS*		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		80			80	
Pitch	p		.020			0.50	
Pins per Side	n1		20			20	
Overall Height	A	.039	.043	.047	1.00	1.10	1.20
Molded Package Thickness	A2	.037	.039	.041	0.95	1.00	1.05
Standoff §	A1	.002	.004	.006	0.05	0.10	0.15
Foot Length	L	.018	.024	.030	0.45	0.60	0.75
Footprint (Reference)	(F)		.039			1.00	
Foot Angle	ϕ	0	3.5	7	0	3.5	7
Overall Width	E	.541	.551	.561	13.75	14.00	14.25
Overall Length	D	.541	.551	.561	13.75	14.00	14.25
Molded Package Width	E1	.463	.472	.482	11.75	12.00	12.25
Molded Package Length	D1	.463	.472	.482	11.75	12.00	12.25
Lead Thickness	c	.004	.006	.008	0.09	0.15	0.20
Lead Width	B	.007	.009	.011	0.17	0.22	0.27
Pin 1 Corner Chamfer	CH	.025	.035	.045	0.64	0.89	1.14
Mold Draft Angle Top	α	5	10	15	5	10	15
Mold Draft Angle Bottom	β	5	10	15	5	10	15

* Controlling Parameter
§ Significant Characteristic

Notes:
Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010” (0.254mm) per side.
JEDEC Equivalent: MS-026
Drawing No. C04-092

APPENDIX C: WHAT'S NEW

This is a new Data Sheet for the Following Devices:

- PIC17C752
- PIC17C756A
- PIC17C762
- PIC17C766

This Data Sheet is based on the PIC17C75X Data Sheet (DS30246A).

APPENDIX D: WHAT'S CHANGED

Clarified the TAD vs. device maximum operating frequency tables in Section 16.2.

Added device characteristic graphs and charts in Section 21.

Removed the "Preliminary" status from the entire document.

Revision C (January 2013)

Added a note to each package outline drawing.