



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 "<u>Embedded -</u> <u>Microcontrollers</u>"

Details

Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	4MHz
Connectivity	-
Peripherals	POR, WDT
Number of I/O	13
Program Memory Size	896B (512 x 14)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	80 x 8
Voltage - Supply (Vcc/Vdd)	2.5V ~ 5.5V
Data Converters	-
Oscillator Type	External
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SSOP (0.209", 5.30mm Width)
Supplier Device Package	20-SSOP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lc554t-04-ss

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong

TABLE 1-1: PIC16C55X FAMILY OF DEVICES

		PIC16C554	PIC16C557	PIC16C558	
Clock	Maximum Frequency of Operation (MHz)	20	20	20	
Memory	EPROM Program Memory (x14 words)	512	2K	2K	
	Data Memory (bytes)	80	128	128	
Peripherals	Timer Module(s)	TMR0	TMR0	TMR0	
	Interrupt Sources	3	3	3	
	I/O Pins	13	22	13	
Features	Voltage Range (Volts)	2.5-5.5	2.5-5.5	2.5-5.5	
	Brown-out Reset	—	—	—	
	Packages	18-pin DIP, SOIC; 20-pin SSOP	28-pin DIP, SOIC; 28-pin SSOP	18-pin DIP, SOIC, SSOP	
All PIC [®] Family devices have Power-on Reset, selectable Watchdog Timer, selectable code protect and high					

I/O current capability. All PIC16C55X Family devices use serial programming with clock pin RB6 and data pin RB7.

FIGURE 4-5: DATA MEMORY MAP FOR THE PIC16C558

File Address	3		File Address		
00h	INDF ⁽¹⁾	INDF ⁽¹⁾	80h		
01h	TMR0	OPTION	81h		
02h	PCL	PCL	82h		
03h	STATUS	STATUS	83h		
04h	FSR	FSR	84h		
05h	PORTA	TRISA	85h		
06h	PORTB	TRISB	86h		
07h			87h		
08h			88h		
09h			89h		
0Ah	PCLATH	PCLATH	8Ah		
0Bh	INTCON	INTCON	8Bh		
0Ch			8Ch		
0Dh			8Dh		
0Eh		PCON	8Eh		
0Fh			8Fh		
10h			90h		
11h			91h		
12h			92h		
13h			93h		
14h			94h		
15h			95h		
16h			96h		
17h		_	97h		
18h			98h		
19h			99h		
1Ah			9Ah		
1Bh			9Bh		
1Ch			9Ch		
1Dh			9Dh		
1Eh		_	9Eh		
1Fh			9Fh		
20h			A0h		
	General	General	71011		
	Purpose Register	Purpose Register			
	regiotor	rtogiotor	BFh		
			C0h		
			\neg		
756			FFb		
7 - 11 -	Bank 0	Bank 1			
Unimplemented data memory locations, read as '0'. Note 1: Not a physical register.					

4.2.2 SPECIAL FUNCTION REGISTERS

The Special Function Registers are registers used by the CPU and peripheral functions for controlling the desired operation of the device (Table 4-1). These registers are static RAM.

The Special Function Registers can be classified into two sets (core and peripheral). The special function registers associated with the "core" functions are described in this section. Those related to the operation of the peripheral features are described in the section of that peripheral feature.

4.2.2.4 PCON Register

The PCON register contains a flag bit to differentiate between a Power-on Reset, an external MCLR Reset or WDT Reset. See Section 6.3 and Section 6.4 for detailed RESET operation.

REGISTER 4-4: PCON REGISTER (ADDRESS 8Eh) U-0 U-0 U-0 R/W-0 U-0 U-0 U-0 U-0 POR bit7 bit 7-2 Unimplemented: Read as '0' bit 1 POR: Power-on Reset status bit 1 = No Power-on Reset occurred 0 = Power-on Reset occurred bit 0 Unimplemented: Read as '0' Legend: R = Readable bit W = Writable bit U = Unimplemented bit, read as '0' - n = Value at POR reset '1' = Bit is set '0' = Bit is cleared x = Bit is unknown

bit0

5.0 I/O PORTS

The PIC16C554 and PIC16C558 have two ports, PORTA and PORTB. The PIC16C557 has three ports, PORTA, PORTB and PORTC.

5.1 PORTA and TRISA Registers

PORTA is a 5-bit wide latch. RA4 is a Schmitt Trigger input and an open-drain output. Port RA4 is multiplexed with the T0CKI clock input. All other RA port pins have Schmitt Trigger input levels and full CMOS output drivers. All pins have data direction bits (TRIS registers) which can configure these pins as input or output.

A '1' in the TRISA register puts the corresponding output driver in a Hi-impedance mode. A '0' in the TRISA register puts the contents of the output latch on the selected pin(s).

Reading the PORTA register reads the status of the pins, whereas writing to it will write to the port latch. All write operations are read-modify-write operations. So a write to a port implies that the port pins are first read, then this value is modified and written to the port data latch.

Note 1: On RESET, the TRISA register is set to all inputs.

FIGURE 5-1: BLOCK DIAGRAM OF PORT PINS RA<3:0>



FIGURE 5-2: BL

BLOCK DIAGRAM OF RA4



5.2 PORTB and TRISB Registers

PORTB is an 8-bit wide bi-directional port. The corresponding data direction register is TRISB. A '1' in the TRISB register puts the corresponding output driver in a Hi-impedance mode. A '0' in the TRISB register puts the contents of the output latch on the selected pin(s).

Reading PORTB register reads the status of the pins whereas writing to it will write to the port latch. All write operations are read-modify-write operations. So a write to a port implies that the port pins are first read, then this value is modified and written to the port data latch.

Each of the PORTB pins has a weak internal pull-up ($\approx 200 \ \mu A$ typical). A single control bit can turn on all the pull-ups. This is done by clearing the RBPU (OPTION<7>) bit. The weak pull-up is automatically turned off when the port pin is configured as an output. The pull-ups are disabled on Power-on Reset.

Four of PORTB's pins, RB7:RB4, have an interrupt-onchange feature. Only pins configured as inputs can cause this interrupt to occur (i.e., any RB7:RB4 pin configured as an output is excluded from the interrupton-change comparison). The input pins (of RB7:RB4) are compared with the old value latched on the last read of PORTB. The "mismatch" outputs of RB7:RB4 are OR'ed together to generate the RBIF interrupt (flag latched in INTCON<0>). This interrupt can wake the device from SLEEP. The user, in the interrupt service routine, can clear the interrupt in the following manner:

- Any read or write of PORTB (this will end the mismatch condition)
- Clear flag bit RBIF

A mismatch condition will continue to set flag bit RBIF. Reading PORTB will end the mismatch condition, and allow flag bit RBIF to be cleared.

The interrupt on mismatch feature, together with software configurable pull-ups on these four pins, allows easy interface to a key pad and make it possible for wake-up on key-depression. (See AN552 in the Microchip *Embedded Control Handbook*.)

Note 1: If a change on the I/O pin should occur when the read operation is being executed (start of the Q2 cycle), then the RBIF interrupt flag may not get set.

The interrupt-on-change feature is recommended for wake-up on key depression operation and operations where PORTB is only used for the interrupt-on-change feature. Polling of PORTB is not recommended while using the interrupt-on-change feature.



FIGURE 5-3: BLOCK DIAGRAM OF RB7:RB4 PINS

5.3 PORTC and TRISC Registers⁽¹⁾

PORTC is a 8-bit wide latch. All pins have data direction bits (TRIS registers) which can configure these pins as input or output.

A '1' in the TRISC register puts the corresponding output driver in a Hi-impedance mode. A '0' in the TRISC register puts the contents of the output latch on the selected pin(s).

Reading the PORTC register reads the status of the pins, whereas writing to it will write to the port latch. All write operations are read-modify-write operations. So a write to a port implies that the port pins are first read, then this value is modified and written to the port data latch

FIGURE 5-5: BLOCK DIAGRAM OF



Name	Bit #	Buffer Type	Function
RC0	Bit 0	TTL	Bi-directional I/O port.
RC1	Bit 1	TTL	Bi-directional I/O port.
RC2	Bit 2	TTL	Bi-directional I/O port.
RC3	Bit 3	TTL	Bi-directional I/O port.
RC4	Bit 4	TTL	Bi-directional I/O port.
RC5	Bit 5	TTL	Bi-directional I/O port.
RC6	Bit 6	TTL	Bi-directional I/O port.
RC7	Bit 7	TTL	Bi-directional I/O port.

Legend: ST = Schmitt Trigger, TTL = TTL input

TABLE 5-6: SUMMARY OF REGISTERS ASSOCIATED WITH PORTC AND TRISC

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR	Value on All Other RESETS
07h	PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	uuuu uuuu
87h	TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111

Legend: x = unknown, u = unchanged Note 1: PIC16C557 ONLY.

5.4 I/O Programming Considerations

5.4.1 BI-DIRECTIONAL I/O PORTS

Any instruction which writes, operates internally as a read followed by a write operation. The BCF and BSF instructions, for example, read the register into the CPU, execute the bit operation and write the result back to the register. Caution must be used when these instructions are applied to a port with both inputs and outputs defined. For example, a BSF operation on bit5 of PORTB will cause all eight bits of PORTB to be read into the CPU. Then the BSF operation takes place on bit5 and PORTB is written to the output latches. If another bit of PORTB is used as a bi-directional I/O pin (e.g., bit 0) and it is defined as an input at this time, the input signal present on the pin itself would be read into the CPU and re-written to the data latch of this particular pin, overwriting the previous content. As long as the pin stays in the Input mode, no problem occurs. However, if bit 0 is switched into Output mode later on, the content of the data latch may now be unknown.

Reading the port register, reads the values of the port pins. Writing to the port register writes the value to the port latch. When using read-modify-write instructions (ex. BCF, BSF, etc.) on a port, the value of the port pins is read, the desired operation is done to this value, and this value is then written to the port latch.

Example 5-1 shows the effect of two sequential read-modify-write instructions (ex., ${\tt BCF}$, ${\tt BSF}$, etc.) on an I/O port.

A pin actively outputting a low or high should not be driven from external devices at the same time in order to change the level on this pin ("wired-or", "wired-and"). The resulting high output currents may damage the chip.

8.1 Instruction Descriptions

ADDLW	Add Literal and W				
Syntax:	[<i>label</i>] ADDLW k				
Operands:	$0 \le k \le 255$				
Operation:	$(W) + k \to (W)$				
Status Affected:	C, DC, Z				
Encoding:	11 111x kkkk kkkk				
Description:	The contents of the W register are added to the eight bit literal 'k' and the result is placed in the W register.				
Words:	1				
Cycles:	1				
Example	ADDLW 0x15				
	Before Instruction				
	W = 0x10				
	After Instruction				
	W = 0x25				

ADDWF	Add W and f				
Syntax:	[label] ADDWF f,d				
Operands:	$0 \le f \le 127$				
	$d \in [0,1]$				
Operation:	$(W) + (f) \to (dest)$				
Status Affected:	C, DC, Z				
Encoding:	00 0111 dfff ffff				
Description:	Add 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'.				
Words:	1				
Cycles:	1				
Example	ADDWF FSR, 0				
	Before Instruction				
	W = 0x17				
	FSR = 0xC2				
	After Instruction				
	W = 0xD9				
	FSR = 0xC2				

ANDLW	AND Li	teral wit	h W		
Syntax:	[label]	ANDLW	/ k		
Operands:	$0 \le k \le 2$	255			
Operation:	(W) .AN	ID. (k) →	→ (W)		
Status Affected:	Z				
Encoding:	11	1001	kkkk	kkkk	
Description:	The contents of W register are AND'ed with the eight bit literal 'k'. The result is placed in the W register.				
Words:	1				
Cycles:	1				
Example	ANDLW	0x5F			
	Before Instruction				
	W	=	0xA3		
	After Instruction				
	W	=	0x03		

ANDWF	AND W with f						
Syntax:	[label] ANDWF f,d						
Operands:	$\begin{array}{l} 0 \leq f \leq 127 \\ d \in [0,1] \end{array}$						
Operation:	(W) .AND. (f) \rightarrow (dest)						
Status Affected:	Z						
Encoding:	00 0101 dfff ffff						
Description:	AND 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'.						
Words:	1						
Cycles:	1						
Example	ANDWF FSR, 1						
	Before Instruction						
	W = 0x17						
	FSR = 0xC2						
	After Instruction						
	W = 0x17						
	FSR = 0x02						

-

PIC16C55X

INCFSZ	Increment f, Skip if 0	IORWF	Inclusive OR W with f
Syntax:	[<i>label</i>] INCFSZ f,d	Syntax:	[<i>label</i>] IORWF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$	Operands:	$0 \le f \le 127$ d \in [0,1]
Operation:	(f) + 1 \rightarrow (dest), skip if result = 0	Operation:	(W) .OR. (f) \rightarrow (dest)
Status Affected:	None	Status Affected:	Z
Encoding:	00 1111 dfff ffff	Encoding:	00 0100 dfff ffff
Description:	The contents of register 'f' are incremented. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'. If the result is 0, the next instruction, which is already fetched, is discarded. A NOP is executed instead making it a two-cycle instruction.	Description: Words: Cycles:	Inclusive OR the W register with register 'f'. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'. 1
Words:	1	Example	IORWF RESULT, 0
Cycles: Example	1(2) HERE INCESS CNT, 1		Before Instruction RESULT = 0x13 W = 0x91
	CONTINUE • • •		After Instruction RESULT = 0x13 W = 0x93
	Before Instruction		Z = 1
	PC = address HERE After Instruction CNT = CNT + 1 if CNT = 0, PC = address CONTINUE if CNT \neq 0, PC = address HERE +1		

IORLW	Inclusiv	ve OR L	iteral wit.	h W	
Syntax:	[label]	IORLW	' k		
Operands:	$0 \le k \le 2$	55			
Operation:	(W) .OR.	$k \rightarrow (W$	')		
Status Affected:	Z				
Encoding:	11	1000	kkkk	kkkk	
Description:	The conte OR'ed with result is pl	nts of the h the eigh aced in t	W register nt bit literal he W regist	r is 'k'. The ter.	
Words:	1				
Cycles:	1				
Example	IORLW	0x35			
	Before In	structio	า		
	W	= 0)x9A		
	After Instruction				
	W	= ()xBF		
	Z	= 1	l		

MOVLW	Move Literal to W				
Syntax:	[<i>label</i>] MOVLW k				
Operands:	$0 \le k \le 2k$	55			
Operation:	$k \rightarrow (W)$				
Status Affected:	None				
Encoding:	11	00xx	kkkk	kkkk	
Description:	The eight bit literal 'k' is loaded into W register. The don't cares will assemble as 0's.				
Words:	1				
Cycles:	1				
Example	MOVLW	0x5A			
	After Instruction				
	W	= 0	x5A		

PIC16C55X

MOVF	Move f					
Syntax:	[label]	MOVF	f,d			
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]					
Operation:	(f) \rightarrow (des	st)				
Status Affected:	Z					
Encoding:	00	1000	dfff	ffff		
	moved to a destination dependant upon the status of d. If $d = 0$, des- tination is W register. If $d = 1$, the destination is file register f itself. d = 1 is useful to test a file register since status flag Z is affected.					
Words:	1					
Cycles:	1					
Example	MOVF	FSR,	0			
	After Inst W Z	ruction = value = 1	e in FSR I	register		

NOP	No Operation				
Syntax:	[label]	NOP			
Operands:	None				
Operation:	No opera	ation			
Status Affected:	None				
Encoding:	00	0000	0xx0	0000	
Description:	No operat	tion.	•		
Words:	1				
Cycles:	1				
Example	NOP				

MOVWF	Move W to f					
Syntax:	[label] MO	/WF	f			
Operands:	$0 \leq f \leq 127$					
Operation:	$(W) \to (f)$					
Status Affected:	None					
Encoding:	00 000	0	lfff	ffff		
Description:	Move data from 'f'.	W r	egister to	register		
Words:	1					
Cycles:	1					
Example	MOVWF OP1	ION	ſ			
	Before Instruc	tion				
	OPTION	=	0xFF			
	W	=	0x4F			
	After Instruction	n				
	OPTION	=	0x4F			
	W	=	0x4F			

OPTION	Load Option Register			
Syntax:	[label]	OPTION	٧	
Operands:	None			
Operation:	$(W) \rightarrow O$	PTION		
Status Affected:	None			
Encoding:	00	0000	0110	0010
Description:	The content loaded in the instruction compatibility Since OPT register, the address it.	nts of the he OPTIC is support ity with PI TION is a he user ca	W register DN register rted for coo C16C5X p readable/v n directly	r are r. This de roducts. vritable
Words:	1			
Cycles:	1			
Example				
	To maintain upward compatibility with future PIC MCU products, do not use this instruction.			

RRF	Rotate Right f through Carry					
Syntax:	[label]	RRF 1	,d			
Operands:	$0 \le f \le 127$ $d \in [0,1]$,				
Operation:	See descr	iption b	below			
Status Affected:	С					
Encoding:	00	1100	dff	f	ffff	
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'.					
	C	 →	Regist	er f	┠╼	
Words:	1					
Cycles:	1					
Example	RRF		REG	1,0		
	Before Ins	tructio	า			
	REG1	= 1	L110	011	.0	
	С	= ()			
	After Instru	uction				
	REG1	= 1	L110	011	.0	
	W	= (0111	001	.1	
	С	= ()			

SLEEP

Syntax:	[label]	SLEEP			
Operands:	None				
Operation:	$\begin{array}{l} 00h \rightarrow WDT, \\ 0 \rightarrow WDT \ prescaler, \\ 1 \rightarrow \overline{TO}, \\ 0 \rightarrow \overline{PD} \end{array}$				
Status Affected:	$\overline{\text{TO}}, \overline{\text{PD}}$				
Encoding:	00	0000	0110	0011	
Description:	The power-down status bit, \overline{PD} is cleared. Timeout 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 6.8 for more details				
Words:	1				
Cycles:	1				
Example:	SLEEP				

SUBLW	Subtract W from Literal						
Syntax:	[label]	SUBLW	/ k				
Operands:	$0 \le k \le 25$	55					
Operation:	k - (W) \rightarrow	• (W)					
Status Affected:	C, DC, Z						
Encoding:	11	110x	kkkk kkkk				
Description:	The W reg plement m 'k'. The res	The W register is subtracted (2's com- plement method) from the eight bit literal 'k'. The result is placed in the W register.					
Words:	1						
Cycles:	1						
Example 1:	SUBLW	0x02					
	Before In	structio	n				
	W	=	1				
	С	=	?				
	After Inst	ruction					
	W	=	1				
	С	=	1; result is positive				
Example 2:	Before In	structio	n				
	W	=	2				
	С	=	?				
	After Inst	ruction					
	W	=	0				
	С	=	1; result is zero				
Example 3:	Before In	structio	n				
	W	=	3				
	С	=	?				
	After Inst	ruction					
	W	=	0xFF				
	С	=	u; result is nega-				

tive

 \odot 1996-2013 Microchip Technology Inc.

9.8 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 PIC MCUs and can be used to develop for this and other PIC microcontrollers. The MPLAB ICD utilizes the in-circuit debugging capability built into the FLASH devices. This feature, along with Microchip's In-Circuit Serial ProgrammingTM 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.

9.9 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 VDD and VPP supplies, which allow it to verify programmed memory at VDD min and VDD 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 devices. It can also set code protection in this mode.

9.10 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.

9.11 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 incircuit 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.

9.12 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^{TM} bus and separate headers for connection to an LCD module and a keypad.







^{© 1996-2013} Microchip Technology Inc.

PIC16C55X



28-Lead Plastic Small Outline (SO) - Wide, 300 mil (SOIC)

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







	Units	INCHES*		MILLIMETERS		6	
Dimension	Limits	MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		28			28	
Pitch	р		.050			1.27	
Overall Height	Α	.093	.099	.104	2.36	2.50	2.64
Molded Package Thickness	A2	.088	.091	.094	2.24	2.31	2.39
Standoff §	A1	.004	.008	.012	0.10	0.20	0.30
Overall Width	E	.394	.407	.420	10.01	10.34	10.67
Molded Package Width	E1	.288	.295	.299	7.32	7.49	7.59
Overall Length	D	.695	.704	.712	17.65	17.87	18.08
Chamfer Distance	h	.010	.020	.029	0.25	0.50	0.74
Foot Length	L	.016	.033	.050	0.41	0.84	1.27
Foot Angle Top	¢	0	4	8	0	4	8
Lead Thickness	С	.009	.011	.013	0.23	0.28	0.33
Lead Width	В	.014	.017	.020	0.36	0.42	0.51
Mold Draft Angle Top	α	0	12	15	0	12	15
Mold Draft Angle Bottom	β	0	12	15	0	12	15

* Controlling Parameter § Significant Characteristic

Notes:

Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" (0.254mm) per side.

JEDEC Equivalent: MS-013

Drawing No. C04-052

18-Lead Ceramic Dual In-line with Window (JW) - 300 mil (CERDIP)

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



	Units	INCHES*		MILLIMETERS		8	
Dimension	Limits	MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		18			18	
Pitch	р		.100			2.54	
Top to Seating Plane	Α	.170	.183	.195	4.32	4.64	4.95
Ceramic Package Height	A2	.155	.160	.165	3.94	4.06	4.19
Standoff	A1	.015	.023	.030	0.38	0.57	0.76
Shoulder to Shoulder Width	Е	.300	.313	.325	7.62	7.94	8.26
Ceramic Pkg. Width	E1	.285	.290	.295	7.24	7.37	7.49
Overall Length	D	.880	.900	.920	22.35	22.86	23.37
Tip to Seating Plane	L	.125	.138	.150	3.18	3.49	3.81
Lead Thickness	С	.008	.010	.012	0.20	0.25	0.30
Upper Lead Width	B1	.050	.055	.060	1.27	1.40	1.52
Lower Lead Width	В	.016	.019	.021	0.41	0.47	0.53
Overall Row Spacing §	eB	.345	.385	.425	8.76	9.78	10.80
Window Width	W1	.130	.140	.150	3.30	3.56	3.81
Window Length	W2	.190	.200	.210	4.83	5.08	5.33

Significant Characteristic JEDEC Equivalent: MO-036 Drawing No. C04-010

28-Lead Plastic Shrink Small Outline (SS) - 209 mil, 5.30 mm (SSOP)

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		28			28	
Pitch	р		.026			0.65	
Overall Height	Α	.068	.073	.078	1.73	1.85	1.98
Molded Package Thickness	A2	.064	.068	.072	1.63	1.73	1.83
Standoff §	A1	.002	.006	.010	0.05	0.15	0.25
Overall Width	Е	.299	.309	.319	7.59	7.85	8.10
Molded Package Width	E1	.201	.207	.212	5.11	5.25	5.38
Overall Length	D	.396	.402	.407	10.06	10.20	10.34
Foot Length	L	.022	.030	.037	0.56	0.75	0.94
Lead Thickness	С	.004	.007	.010	0.10	0.18	0.25
Foot Angle	¢	0	4	8	0.00	101.60	203.20
Lead Width	В	.010	.013	.015	0.25	0.32	0.38
Mold Draft Angle Top	α	0	5	10	0	5	10
Mold Draft Angle Bottom	β	0	5	10	0	5	10

Controlling Parameter § Significant Characteristic

Notes:

Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" (0.254mm) per side. JEDEC Equivalent: MS-150

Drawing No. C04-073

A2

NOTES:

INDEX

A

ADDLW Instruction	
ADDWF Instruction	
ANDLW Instruction	
ANDWF Instruction	
Architectural Overview	9
Assembler	
MPASM Assembler	67

В

BCF Instruction	
Block Diagram	
TIMER0	47
TMR0/WDT PRESCALER	50
BSF Instruction	
BTFSC Instruction	
BTFSS Instruction	57

С

CALL Instruction	57
Clocking Scheme/Instruction Cycle	
CLRF Instruction	
CLRW Instruction	
CLRWDT Instruction	
Code Protection	
COMF Instruction	
Configuration Bits	31

D

Data Memory Organization1	13
DECF Instruction	58
DECFSZ Instruction	59
Development Support6	37

Ε

rrata	3
xternal Crystal Oscillator Circuit	4

G

General purpose Register File	13
GOTO Instruction	59

I

	~~
I/O Ports	23
I/O Programming Considerations	28
ICEPIC In-Circuit Emulator	68
ID Locations	46
INCF Instruction	59
INCFSZ Instruction	60
In-Circuit Serial Programming	46
Indirect Addressing, INDF and FSR Registers	21
Instruction Flow/Pipelining	12
Instruction Set	
ADDLW	55
ADDWF	55
ANDLW	55
ANDWF	55
BCF	56
BSF	56
BTFSC	56
BTFSS	57
CALL	57
CLRF	57

CLRW	58
CLRWDT	58
COMF	58
DECF	58
DECFSZ	59
GOTO	59
INCF	59
INCFSZ	60
IORLW	60
IORWF	60
MOVF	61
MOVLW	60
MOVWF	61
NOP	61
OPTION	61
RETFIE	62
RETLW	62
RETURN	62
RLF	62
RRF	63
SLEEP	63
SUBLW	63
SUBWF	64
SWAPF	64
TRIS	64
XORLW	65
XORWF	65
Instruction Set Summary	53
INT Interrupt	42
INTCON Register	19
Interrupts	41
IORLW Instruction	60
IORWF Instruction	60

Κ

Μ

MOVF Instruction
MOVLW Instruction
MOVWF Instruction
MPLAB C17 and MPLAB C18 C Compilers 67
MPLAB ICD In-Circuit Debugger 69
MPLAB ICE High Performance Universal In-Circuit Emulator
with MPLAB IDE
MPLAB Integrated Development Environment Software 67
MPLINK Object Linker/MPLIB Object Librarian 68

Ν

0

One-Time-Programmable (OTP) Devices	7
OPTION Instruction	61
OPTION Register	
Oscillator Configurations	33
Oscillator Start-up Timer (OST)	36

Ρ

PCL and PCLATH	21
PCON Register	20
PICDEM 1 Low Cost PIC MCU Demonstration Board	69
PICDEM 17 Demonstration Board	70
PICDEM 2 Low Cost PIC16CXX Demonstration Board	69
PICDEM 3 Low Cost PIC16CXXX Demonstration Board	70

PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.

PART NO.	<u>x /xx xxx</u>	Examples:
Device	Temperature Package Pattern Range	a) PIC17C756–16L Commercial Temp., PLCC package, 16 MHz, normal VDD limits
Device	PIC17C756: Standard VDD range PIC17C756T: (Tape and Reel) PIC17LC756: Extended VDD range	 b) PIC17LC756–08/PT Commercial Temp., TQFP package, 8MHz, extended VDD limits c) PIC17C756–33I/PT Industrial Temp., TQFP content of the set that the set the set that the set that the set that the
Temperature Range	$\begin{array}{rcl} - & = & 0^{\circ}C \text{ to } +70^{\circ}C \\ I & = & -40^{\circ}C \text{ to } +85^{\circ}C \end{array}$	I QFP package, 33 MHz, normal VDD limits
Package	CL = Windowed LCC PT = TQFP L = PLCC	
Pattern	QTP, SQTP, ROM Code (factory specified) or Special Requirements. Blank for OTP and Windowed devices.	

* JW Devices are UV erasable and can be programmed to any device configuration. JW Devices meet the electrical requirement of each oscillator type.

Sales and Support

Data Sheets

Products supported by a preliminary Data Sheet may have an errata sheet describing minor operational differences and recommended workarounds. To determine if an errata sheet exists for a particular device, please contact one of the following:

1. Your local Microchip sales office

2. The Microchip Worldwide Site (www.microchip.com)