



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	Obsolete
Core Processor	PIC
Core Size	8-Bit
Speed	40MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	33
Program Memory Size	16KB (8K x 16)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	2.5V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Through Hole
Package / Case	40-DIP (0.600", 15.24mm)
Supplier Device Package	40-PDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18lc442-i-p

Email: info@E-XFL.COM

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

#### 3.1 Power-on Reset (POR)

A Power-on Reset pulse is generated on-chip when VDD rise is detected. To take advantage of the POR circuitry, just tie the MCLR pin directly (or through a resistor) to VDD. This will eliminate external RC components usually needed to create a Power-on Reset delay. A minimum rise rate for VDD is specified (parameter D004). For a slow rise time, see Figure 3-2.

When the device starts normal operation (i.e., exits the RESET condition), device operating parameters (voltage, frequency, temperature, etc.) must be met to ensure operation. If these conditions are not met, the device must be held in reset until the operating conditions are met.

FIGURE 3-2: EXTERNAL POWER-ON RESET CIRCUIT (FOR SLOW VDD POWER-UP)



ing into MCLR from external capacitor C in the event of MCLR/VPP pin breakdown, due to Electrostatic Discharge (ESD) or Electrical Overstress (EOS).

#### 3.2 Power-up Timer (PWRT)

The Power-up Timer provides a fixed nominal time-out (parameter #33) only on power-up from the POR. The Power-up Timer operates on an internal RC oscillator. The chip is kept in reset as long as the PWRT is active. The PWRT's time delay allows VDD to rise to an acceptable level. A configuration bit is provided to enable/ disable the PWRT.

The power-up time delay will vary from chip-to-chip due to VDD, temperature and process variation. See DC parameter #33 for details.

### 3.3 Oscillator Start-up Timer (OST)

The Oscillator Start-up Timer (OST) provides a 1024 oscillator cycle (from OSC1 input) delay after the PWRT delay is over (parameter #32). This ensures that the crystal oscillator or resonator has started and stabilized.

The OST time-out is invoked only for XT, LP and HS modes and only on Power-on Reset or wake-up from SLEEP.

#### 3.4 PLL Lock Time-out

With the PLL enabled, the time-out sequence following a Power-on Reset is different from other oscillator modes. A portion of the Power-up Timer is used to provide a fixed time-out that is sufficient for the PLL to lock to the main oscillator frequency. This PLL lock time-out (TPLL) is typically 2 ms and follows the oscillator startup time-out (OST).

#### 3.5 Brown-out Reset (BOR)

A configuration bit, BOREN, can disable (if clear/ programmed), or enable (if set) the Brown-out Reset circuitry. If VDD falls below parameter D005 for greater than parameter #35, the brown-out situation will reset the chip. A RESET may not occur if VDD falls below parameter D005 for less than parameter #35. The chip will remain in Brown-out Reset until VDD rises above BVDD. The Power-up Timer will then be invoked and will keep the chip in RESET an additional time delay (parameter #33). If VDD drops below BVDD while the Power-up Timer is running, the chip will go back into a Brown-out Reset and the Power-up Timer will be initialized. Once VDD rises above BVDD, the Power-up Timer will execute the additional time delay.

#### 3.6 Time-out Sequence

On power-up, the time-out sequence is as follows: First, PWRT time-out is invoked after the POR time delay has expired. Then, OST is activated. The total time-out will vary based on oscillator configuration and the status of the PWRT. For example, in RC mode with the PWRT disabled, there will be no time-out at all. Figure 3-3, Figure 3-4, Figure 3-5, Figure 3-6 and Figure 3-7 depict time-out sequences on power-up.

Since the time-outs occur from the POR pulse, if  $\overline{\text{MCLR}}$  is kept low long enough, the time-outs will expire. Bringing  $\overline{\text{MCLR}}$  high will begin execution immediately (Figure 3-5). This is useful for testing purposes or to synchronize more than one PIC18CXXX device operating in parallel.

Table 3-2 shows the RESET conditions for some Special Function Registers, while Table 3-3 shows the RESET conditions for all the registers.

#### 4.10 Access Bank

The Access Bank is an architectural enhancement, which is very useful for C compiler code optimization. The techniques used by the C compiler may also be useful for programs written in assembly.

This data memory region can be used for:

- Intermediate computational values
- · Local variables of subroutines
- Faster context saving/switching of variables
- Common variables
- Faster evaluation/control of SFRs (no banking)

The Access Bank is comprised of the upper 128 bytes in Bank 15 (SFRs) and the lower 128 bytes in Bank 0. These two sections will be referred to as Access RAM High and Access RAM Low, respectively. Figure 4-6 and Figure 4-7 indicate the Access RAM areas.

A bit in the instruction word specifies if the operation is to occur in the bank specified by the BSR register or in the Access Bank. This bit is denoted by the 'a' bit (for access bit).

When forced in the Access Bank (a = '0'), the last address in Access RAM Low is followed by the first address in Access RAM High. Access RAM High maps the Special Function registers, so that these registers can be accessed without any software overhead. This is useful for testing status flags and modifying control bits.

#### 4.11 Bank Select Register (BSR)

The need for a large general purpose memory space dictates a RAM banking scheme. The data memory is partitioned into sixteen banks. When using direct addressing, the BSR should be configured for the desired bank.

BSR<3:0> holds the upper 4 bits of the 12-bit RAM address. The BSR<7:4> bits will always read '0's, and writes will have no effect.

A MOVLB instruction has been provided in the instruction set to assist in selecting banks.

If the currently selected bank is not implemented, any read will return all '0's and all writes are ignored. The STATUS register bits will be set/cleared as appropriate for the instruction performed.

Each Bank extends up to FFh (256 bytes). All data memory is implemented as static RAM.

A MOVFF instruction ignores the BSR, since the 12-bit addresses are embedded into the instruction word.

Section 4.12 provides a description of indirect addressing, which allows linear addressing of the entire RAM space.



#### © 1999-2013 Microchip Technology Inc.

#### 4.13.1 RCON REGISTER

The Reset Control (RCON) register contains flag bits that allow differentiation between the sources of a device RESET. These flags include the TO, PD, POR, BOR and RI bits. This register is readable and writable.

# Note 1: If the BOREN configuration bit is set (Brown-out Reset enabled), the BOR bit is '1' on a Power-on Reset. After a Brown-out Reset has occurred, the BOR bit will be clear and must be set by firmware to indicate the occurrence of the next Brown-out Reset. If the BOREN configuration bit is clear (Brown-out Reset disabled), BOR is unknown after Power-on Reset and Brown-out Reset conditions. 2: It is recommended that the POR bit be set after a Power-on Reset has been detected, so that subsequent Power-on

Resets may be detected.

#### REGISTER 4-3: RCON REGISTER

R/W-0	R/W-0	U-0	R/W-1	R/W-1	R/W-1	R/W-0	R/W-0
IPEN	LWRT	—	RI	TO	PD	POR	BOR
bit 7							bit 0

bit / IPEN: Interrupt Priority Enable bi
--

- 1 = Enable priority levels on interrupts
- 0 = Disable priority levels on interrupts (16CXXX compatibility mode)
- bit 6 LWRT: Long Write Enable bit
  - 1 = Enable TBLWT to internal program memory
    - Once this bit is set, it can only be cleared by a POR or MCLR Reset.
  - 0 = Disable TBLWT to internal program memory; TBLWT only to external program memory
- bit 5 Unimplemented: Read as '0'
- bit 4 **RI:** RESET Instruction Flag bit
  - 1 = The RESET instruction was not executed
  - The RESET instruction was executed causing a device RESET (must be set in software after a Brown-out Reset occurs)
- bit 3 TO: Watchdog Time-out Flag bit
  - 1 = After power-up, CLRWDT instruction, or SLEEP instruction
  - 0 = A WDT time-out occurred
- bit 2 **PD**: Power-down Detection Flag bit
  - 1 = After power-up or by the CLRWDT instruction
  - 0 = By execution of the SLEEP instruction
- bit 1 **POR:** Power-on Reset Status bit
  - 1 = A Power-on Reset has not occurred
  - 0 = A Power-on Reset occurred
    - (must be set in software after a Power-on Reset occurs)
- bit 0 **BOR:** Brown-out Reset Status bit
  - 1 = A Brown-out Reset has not occurred
  - 0 = A Brown-out Reset occurred
    - (must be set in software after a Brown-out Reset occurs)

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

#### 7.3 PIE Registers

The PIE registers contain the individual enable bits for the peripheral interrupts. Due to the number of peripheral interrupt sources, there are two Peripheral Interrupt Enable Registers (PIE1, PIE2). When IPEN = 0, the PEIE bit must be set to enable any of these peripheral interrupts.

#### REGISTER 7-6: PERIPHERAL INTERRUPT ENABLE REGISTER 1 (PIE1)

	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	PSPIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE
	bit 7							bit 0
bit 7	PSPIE: Pa	rallel Slave	Port Read/	Vrite Interru	pt Enable bit			
	1 = Enable 0 = Disable	es the PSP re es the PSP r	ead/write in ead/write ir	terrupt nterrupt				
bit 6	ADIE: A/D	Converter In	nterrupt Ena	able bit				
	1 = Enable 0 = Disable	es the A/D in es the A/D in	terrupt nterrupt					
bit 5	RCIE: USA	ART Receive	Interrupt E	nable bit				
	1 = Enable	s the USAR	T receive in	iterrupt				
	0 = Disable	es the USAF		nterrupt				
bit 4	IXIE: USA		t Interrupt E	nable bit				
	1 = Enable 0 = Disable	es the USAR	T transmit i RT transmit	nterrupt interrupt				
bit 3	SSPIE: Ma	aster Synchr	onous Seria	al Port Interr	upt Enable bit			
	1 = Enable	s the MSSP	interrupt					
	0 = Disable	es the MSSF	p interrupt	_				
bit 2	CCP1IE: C	CP1 Interru	pt Enable b	it				
	1 = Enable 0 = Disable	es the CCP1 es the CCP1	interrupt interrupt					
bit 1	TMR2IE: T	MR2 to PR2	2 Match Inte	errupt Enabl	e bit			
	1 = Enable 0 = Disable	es the TMR2 es the TMR2	to PR2 ma to PR2 ma	tch interrupt atch interrup	: t			
bit 0	TMR1IE: T	MR1 Overflo	ow Interrupt	Enable bit				
	1 = Enable 0 = Disable	es the TMR1 es the TMR1	overflow in overflow ir	terrupt nterrupt				
	Legend:							
	R = Reada	ble bit	VV = V	Vritable bit	U = Unimple	emented bi	t, read as '(	)'
	- n = Value	at POR	'1' = E	Bit is set	'0' = Bit is c	leared	x = Bit is ur	nknown





#### 10.1 Timer1 Operation

Timer1 can operate in one of these modes:

- As a timer
- As a synchronous counter
- As an asynchronous counter

The operating mode is determined by the clock select bit, TMR1CS (T1CON<1>).

When TMR1CS = 0, Timer1 increments every instruction cycle. When TMR1CS = 1, Timer1 increments on every rising edge of the external clock input or the Timer1 oscillator, if enabled.

When the Timer1 oscillator is enabled (T1OSCEN is set), the RC1/T1OSI and RC0/T1OSO/T1CKI pins become inputs. That is, the TRISC<1:0> value is ignored.

Timer1 also has an internal "RESET input". This RESET can be generated by the CCP module (Section 13.0).



#### FIGURE 10-2: TIMER1 BLOCK DIAGRAM: 16-BIT READ/WRITE MODE



## FIGURE 10-1: TIMER1 BLOCK DIAGRAM

## 14.4 MSSP I<sup>2</sup>C Operation

The MSSP module in I<sup>2</sup>C mode, fully implements all master and slave functions (including general call support) and provides interrupts on START and STOP bits in hardware to determine a free bus (multi-master function). The MSSP module implements the standard mode specifications, as well as 7-bit and 10-bit addressing.

Two pins are used for data transfer. These are the RC3/ SCK/SCL pin, which is the clock (SCL), and the RC4/ SDI/SDA pin, which is the data (SDA). The user must configure these pins as inputs or outputs through the TRISC<4:3> bits.

The MSSP module functions are enabled by setting MSSP enable bit SSPEN (SSPCON<5>).





The MSSP module has six registers for  $I^2C$  operation. These are the:

- MSSP Control Register1 (SSPCON1)
- MSSP Control Register2 (SSPCON2)
- MSSP Status Register (SSPSTAT)
- Serial Receive/Transmit Buffer (SSPBUF)
- MSSP Shift Register (SSPSR) Not directly accessible
- MSSP Address Register (SSPADD)

The SSPCON1 register allows control of the  $I^2C$  operation. Four mode selection bits (SSPCON<3:0>) allow one of the following  $I^2C$  modes to be selected:

- I<sup>2</sup>C Master mode, clock = OSC/4 (SSPADD +1)
- I<sup>2</sup>C Slave mode (7-bit address)
- I<sup>2</sup>C Slave mode (10-bit address)
- I<sup>2</sup>C Slave mode (7-bit address), with START and STOP bit interrupts enabled
- I<sup>2</sup>C Slave mode (10-bit address), with START and STOP bit interrupts enabled
- I<sup>2</sup>C Firmware controlled master operation, slave is idle

Selection of any I<sup>2</sup>C mode with the SSPEN bit set, forces the SCL and SDA pins to be open drain, provided these pins are programmed to be inputs by setting the appropriate TRISC bits.

#### 14.4.1 SLAVE MODE

In Slave mode, the SCL and SDA pins must be configured as inputs (TRISC<4:3> set). The MSSP module will override the input state with the output data when required (slave-transmitter).

When an address is matched or the data transfer after an address match is received, the hardware automatically will generate the Acknowledge ( $\overline{ACK}$ ) pulse and load the SSPBUF register with the received value currently in the SSPSR register.

There are certain conditions that will cause the MSSP module not to give this ACK pulse. These are if either (or both):

- a) The buffer full bit BF (SSPSTAT<0>) was set before the transfer was received.
- b) The overflow bit SSPOV (SSPCON<6>) was set before the transfer was received.

In this case, the SSPSR register value is not loaded into the SSPBUF, but bit SSPIF (PIR1<3>) is set. The BF bit is cleared by reading the SSPBUF register, while bit SSPOV is cleared through software.

The SCL clock input must have a minimum high and low for proper operation. The high and low times of the  $I^2C$  specification, as well as the requirement of the MSSP module, are shown in timing parameter #100 and parameter #101.

#### 14.4.1.1 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.
- b) The buffer full bit BF is set.
- c) An ACK pulse is generated.
- MSSP interrupt flag bit SSPIF (PIR1<3>) is set (interrupt is generated if enabled) on the falling edge of the ninth 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/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 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).
- 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. If match releases SCL line, this will clear bit UA.
- 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.

#### 14.4.1.2 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 <u>byte</u> overflow condition exists, then no Acknowledge (ACK) pulse is given. An overflow condition is defined as either bit BF (SSPSTAT<0>) is set, or bit SSPOV (SSPCON<6>) is set.

An MSSP interrupt is generated for each data transfer byte. Flag bit SSPIF (PIR1<3>) must be cleared in software. The SSPSTAT register is used to determine the status of the byte.

#### 14.4.1.3 Transmission

When the  $R/\overline{W}$  bit of the incoming address byte is set and an address match occurs, the  $R/\overline{W}$  bit of the SSPSTAT register is set. The received address is loaded into the SSPBUF register. The ACK pulse will be sent on the ninth bit and pin RC3/SCK/SCL is held low. The transmit data must be loaded into the SSPBUF register, which also loads the SSPSR register. Then pin RC3/SCK/SCL should be enabled by setting bit CKP (SSPCON<4>). The master must monitor the SCL pin prior to asserting another clock pulse. The slave devices may be holding off the master by stretching the clock. The eight data bits are shifted out on the falling edge of the SCL input. This ensures that the SDA signal is valid during the SCL high time (Figure 14-9).

An MSSP interrupt is generated for each data transfer byte. The SSPIF bit must be cleared in software and the SSPSTAT register is used to determine the status of the byte. The SSPIF bit is set on the falling edge of the ninth clock pulse.

As a slave-transmitter, the  $\overline{ACK}$  pulse from the master-receiver is latched on the rising edge of the ninth SCL input pulse. If the SDA line is high (not  $\overline{ACK}$ ), then the data transfer is complete. When the  $\overline{ACK}$  is latched by the slave, the slave logic is reset (resets SSPSTAT register) and the slave monitors for another occurrence of the START bit. If the SDA line was low ( $\overline{ACK}$ ), the transmit data must be loaded into the SSPBUF register, which also loads the SSPSR register. Pin RC3/SCK/SCL should be enabled by setting bit CKP.



#### 14.4.5 BAUD RATE GENERATOR

In I<sup>2</sup>C Master mode, the reload value for the BRG is located in the lower 7 bits of the SSPADD register (Figure 14-14). When the BRG is loaded with this value, the BRG counts down to 0 and stops until another reload has taken place. The BRG count is dec-

remented twice per instruction cycle (Tcr) on the Q2 and Q4 clocks. In  $I^2C$  Master mode, the BRG is reloaded automatically. If Clock Arbitration is taking place, for instance, the BRG will be reloaded when the SCL pin is sampled high (Figure 14-15).













#### FIGURE 15-3: ASYNCHRONOUS TRANSMISSION (BACK TO BACK)



#### TABLE 15-6: REGISTERS ASSOCIATED WITH ASYNCHRONOUS TRANSMISSION

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
INTCON	GIE/GIEH	PEIE/ GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INTOIF	RBIF	0000 000x	0000 000u
PIR1	PSPIF <sup>(1)</sup>	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
PIE1	PSPIE <sup>(1)</sup>	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
IPR1	PSPIP <sup>(1)</sup>	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP	0000 0000	0000 0000
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 -00x	0000 -00x
TXREG	USART Transmit Register								0000 0000	0000 0000
TXSTA	CSRC	TX9	TXEN	SYNC		BRGH	TRMT	TX9D	0000 -010	0000 -010
SPBRG	G Baud Rate Generator Register								0000 0000	0000 0000

Legend: x = unknown, - = unimplemented locations read as '0'.

Shaded cells are not used for Asynchronous Transmission.

**Note 1:** The PSPIF, PSPIE and PSPIP bits are reserved on the PIC18C2X2 devices. Always maintain these bits clear.

#### FIGURE 17-2: LOW VOLTAGE DETECT (LVD) BLOCK DIAGRAM



The LVD module has an additional feature that allows the user to supply the trip voltage to the module from an external source. This mode is enabled when bits LVDL3:LVDL0 are set to 1111. In this state, the comparator input is multiplexed from the external input pin LVDIN (Figure 17-3). This gives flexibility, because it allows a user to configure the Low Voltage Detect interrupt to occur at any voltage in the valid operating range.



FIGURE 17-3: LOW VOLTAGE DETECT (LVD) WITH EXTERNAL INPUT BLOCK DIAGRAM

#### 17.2.1 REFERENCE VOLTAGE SET POINT

The Internal Reference Voltage of the LVD module may be used by other internal circuitry (the Programmable Brown-out Reset). If these circuits are disabled (lower current consumption), the reference voltage circuit requires a time to become stable before a low voltage condition can be reliably detected. This time is invariant of system clock speed. This start-up time is specified in electrical specification parameter #36. The low voltage interrupt flag will not be enabled until a stable reference voltage is reached. Refer to the waveform in Figure 17-4.

#### 17.2.2 CURRENT CONSUMPTION

When the module is enabled, the LVD comparator and voltage divider are enabled and will consume static current. The voltage divider can be tapped from multiple places in the resistor array. Total current consumption, when enabled, is specified in electrical specification parameter #D022B.

#### 17.3 Operation During SLEEP

When enabled, the LVD circuitry continues to operate during SLEEP. If the device voltage crosses the trip point, the LVDIF bit will be set and the device will wakeup from SLEEP. Device execution will continue from the interrupt vector address, if interrupts have been globally enabled.

#### 17.4 Effects of a RESET

A device RESET forces all registers to their RESET state. This forces the LVD module to be turned off.

MULLW Multiply Literal with WREG									
Synt	ax:	[ label ]	MULLW	k					
Ope	rands:	$0 \le k \le 25$	55						
Ope	ration:	(WREG)	$x k \to PF$	RODH:PI	RODL				
Statu	us Affected:	None							
Enco	oding:	0000	1101	kkkk	kkkk				
Des	cription:	An unsigned multiplication is car- ried out between the contents of WREG and the 8-bit literal 'k'. The 16-bit result is placed in PRODH:PRODL register pair. PRODH contains the high byte. WREG is unchanged. None of the status flags are affected. Note that neither overflow, nor carry is possible in this opera- tion. A zero result is possible but not detected							
Wor	ds:	1	1						
Cvcl	es:	1							
QC	vcle Activity:								
	Q1	Q2	Q3		Q4				
	Decode	Read literal 'k'	Proce Data	ss a re Pl P	Write gisters RODH: RODL				
<u>Exar</u>	<u>mple</u> :	MULLW	0xC4						
Before Instruction									
	WREG PRODH PRODL	= 0: = ? = ?	xE2						
	After Instruct	ion							
	WREG PRODH PRODL	= 0: = 0: = 0:	xE2 xAD x08						

MULWF	Multiply \	WREG with	f			
Syntax:	[ label ]	MULWF f	[,a]			
Operands:	$0 \le f \le 255$ a $\in [0,1]$	5				
Operation:	(WREG) >	$x(f) \rightarrow PROE$	DH:PRODL			
Status Affected:	None					
Encoding:	0000	001a ffi	f ffff			
Description:	An unsigned multiplication is car- ried out between the contents of WREG and the register file loca- tion 'f'. The 16-bit result is stored in the PRODH:PRODL register pair. PRODH contains the high byte. Both WREG and 'f' are unchanged. None of the status flags are affected. Note that neither overflow, nor carry is possible in this opera- tion. A zero result is possible but not detected. If 'a' is 0, the Access Bank will be selected, overriding the BSR value. If 'a'= 1, then the bank will be selected as per the BSR value (default).					
Words:	1					
Cycles:	1					
Q Cycle Activity:	0.0	0.0	<b>.</b>			
Q1	Q2	Q3	Q4			
Decode	Read register 'f'	Process Data	Write registers PRODH: PRODL			
Example:	MULWF	REG, 1				
Before Instru	iction					
WREG REG PRODH PRODL	= 0x = 0x = ? = ?	:C4 :B5				
After Instruct	ion					
WREG	= 0 x	:C4				

ner instruction		
WREG	=	0xC4
REG	=	0xB5
PRODH	=	0x8A
PRODL	=	0x94

## 20.0 DEVELOPMENT SUPPORT

The PIC<sup>®</sup> microcontrollers are supported with a full range of hardware and software development tools:

- Integrated Development Environment
  - MPLAB<sup>®</sup> IDE Software
- Assemblers/Compilers/Linkers
  - MPASM<sup>™</sup> Assembler
  - MPLAB C17 and MPLAB C18 C Compilers
  - MPLINK™ Object Linker/
  - MPLIB<sup>™</sup> Object Librarian
- Simulators
  - MPLAB SIM Software Simulator
- Emulators
  - MPLAB ICE 2000 In-Circuit Emulator
- ICEPIC<sup>™</sup> In-Circuit Emulator
- In-Circuit Debugger
  - MPLAB ICD for PIC16F87X
- Device Programmers
  - PRO MATE<sup>®</sup> II Universal Device Programmer
- PICSTART<sup>®</sup> Plus Entry-Level Development Programmer
- · Low Cost Demonstration Boards
  - PICDEM<sup>™</sup>1 Demonstration Board
  - PICDEM 2 Demonstration Board
  - PICDEM 3 Demonstration Board
  - PICDEM 17 Demonstration Board
  - KEELOQ<sup>®</sup> Demonstration Board

#### 20.1 MPLAB Integrated Development Environment Software

The MPLAB IDE software brings an ease of software development previously unseen in the 8-bit microcontroller market. The MPLAB IDE is a Windows<sup>®</sup>-based application that contains:

- · An interface to debugging tools
  - simulator
  - programmer (sold separately)
  - emulator (sold separately)
  - in-circuit debugger (sold separately)
- A full-featured editor
- · A project manager
- Customizable toolbar and key mapping
- · A status bar
- On-line help

The MPLAB IDE allows you to:

- Edit your source files (either assembly or 'C')
- One touch assemble (or compile) and download to PIC MCU emulator and simulator tools (automatically updates all project information)
- Debug using:
  - source files
  - absolute listing file
  - machine code

The ability to use MPLAB IDE with multiple debugging tools allows users to easily switch from the costeffective simulator to a full-featured emulator with minimal retraining.

#### 20.2 MPASM Assembler

The MPASM assembler is a full-featured universal macro assembler for all PIC MCUs.

The MPASM assembler has a command line interface and a Windows shell. It can be used as a stand-alone application on a Windows 3.x or greater system, or it can be used through MPLAB IDE. The MPASM assembler generates relocatable object files for the MPLINK object linker, Intel<sup>®</sup> standard HEX files, MAP files to detail memory usage and symbol reference, an absolute LST file that contains source lines and generated machine code, and a COD file for debugging.

The MPASM assembler features include:

- Integration into MPLAB IDE projects.
- User-defined macros to streamline assembly code.
- Conditional assembly for multi-purpose source files.
- Directives that allow complete control over the assembly process.

#### 20.3 MPLAB C17 and MPLAB C18 C Compilers

The MPLAB C17 and MPLAB C18 Code Development Systems are complete ANSI 'C' compilers for Microchip's PIC17CXXX and PIC18CXXX family of microcontrollers, respectively. These compilers provide powerful integration capabilities and ease of use not found with other compilers.

For easier source level debugging, the compilers provide symbol information that is compatible with the MPLAB IDE memory display.

Note:



## TABLE 21-19: USART SYNCHRONOUS TRANSMISSION REQUIREMENTS

Refer to Figure 21-4 for load conditions.

Param. No.	Symbol	Characteristic		Min	Max	Units	Conditions
120	TckH2dtV	SYNC XMIT (MASTER & SLAVE)					
		Clock high to data out valid	PIC18 <b>C</b> XXX		40	ns	
			PIC18LCXXX	_	100	ns	
121	Tckrf	Clock out rise time and fall time	PIC18CXXX	_	25	ns	
	(Master mode)	PIC18LCXXX	_	50	ns		
122	Tdtrf	Data out rise time and fall time	PIC18CXXX		25	ns	
			PIC18LCXXX		50	ns	

#### FIGURE 21-21: USART SYNCHRONOUS RECEIVE (MASTER/SLAVE) TIMING



#### TABLE 21-20: USART SYNCHRONOUS RECEIVE REQUIREMENTS

Param. No.	Symbol	Characteristic	Min	Max	Units	Conditions
125	TdtV2ckl	SYNC RCV (MASTER & SLAVE) Data hold before CK $\downarrow$ (DT hold time)	10		ns	
126	TckL2dtl	Data hold after CK $\downarrow$ (DT hold time)	15		ns	



FIGURE 22-17: TYPICAL, MINIMUM AND MAXIMUM WDT PERIOD vs. VDD (-40°C TO +125°C)







FIGURE 22-25: MINIMUM AND MAXIMUM VIN vs. VDD, (TTL INPUT, -40°C TO +125°C)





© 1999-2013 Microchip Technology Inc.