



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	25MHz
Connectivity	EBI/EMI, I ² C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	69
Program Memory Size	48KB (24K x 16)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	3.75K x 8
Voltage - Supply (Vcc/Vdd)	4.2V ~ 5.5V
Data Converters	A/D 16x10b
Oscillator Type	External
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	80-TQFP
Supplier Device Package	80-TQFP (12x12)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f8525-e-pt

Email: info@E-XFL.COM

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

TABLE 2-2:CAPACITOR SELECTION FOR
CRYSTAL OSCILLATOR

Ranges Tested:							
Mode	Mode Freq C1						
LP	32.0 kHz	33 pF	33 pF				
XT	200 kHz	47-68 pF	47-68 pF 15 pF 15 pF				
	1.0 MHz	15 pF					
	4.0 MHz	15 pF					
HS	4.0 MHz	15 pF	15 pF				
	8.0 MHz	15-33 pF	15-33 pF				
	20.0 MHz	15-33 pF	15-33 pF				
	25.0 MHz	15-33 pF	15-33 pF				

These values are for design guidance only. See notes following this table.

Crystals Used

-	
32 kHz	4 MHz
200 kHz	8 MHz
1 MHz	20 MHz

- Note 1: Higher capacitance increases the stability of the oscillator but also increases the start-up time.
 - 2: Rs (see Figure 2-1) may be required in HS mode, as well as XT mode, to avoid overdriving crystals with low drive level specification.
 - 3: Since each resonator/crystal has its own characteristics, the user should consult the resonator/crystal manufacturer for appropriate values of external components or verify oscillator performance.

An external clock source may also be connected to the OSC1 pin in the HS, XT and LP modes as shown in Figure 2-2.

FIGURE 2-2: EXTERNAL CLOCK INPUT OPERATION (HS, XT OR LP OSCILLATOR CONFIGURATION)



2.3 RC Oscillator

For timing insensitive applications, the "RC" and "RCIO" device options offer additional cost savings. The RC oscillator frequency is a function of the supply voltage, the resistor (REXT) and capacitor (CEXT) values and the operating temperature. In addition to this, the oscillator frequency will vary from unit to unit due to normal process parameter variation. Furthermore, the difference in lead frame capacitance between package types will also affect the oscillation frequency, especially for low CEXT values. The user also needs to take into account variation due to tolerance of external R and C components used. Figure 2-3 shows how the R/C combination is connected.

In the RC Oscillator mode, the oscillator frequency divided by 4 is available on the OSC2 pin. This signal may be used for test purposes or to synchronize other logic.



The RCIO Oscillator mode functions like the RC mode except that the OSC2 pin becomes an additional general purpose I/O pin. The I/O pin becomes bit 6 of PORTA (RA6).

4.0 MEMORY ORGANIZATION

There are three memory blocks in PIC18F6525/6621/ 8525/8621 devices. They are:

- Program Memory
- Data RAM
- Data EEPROM

Data and program memory use separate busses which allow for concurrent access of these blocks. Additional detailed information for Flash program memory and data EEPROM is provided in Section 5.0 "Flash Program Memory" and Section 7.0 "Data EEPROM Memory", respectively.

In addition to on-chip Flash, the PIC18F8525/8621 devices are also capable of accessing external program memory through an external memory bus. Depending on the selected operating mode (discussed in Section 4.1.1 "PIC18F6525/6621/8525/8621 Program Memory Modes"), the controllers may access either internal or external program memory exclusively, or both internal and external memory in selected blocks. Additional information on the external memory interface is provided in Section 6.0 "External Memory Interface".

4.1 Program Memory Organization

A 21-bit program counter is capable of addressing the 2-Mbyte program memory space. Accessing a location between the physically implemented memory and the 2-Mbyte address will cause a read of all '0's (a NOP instruction).

The PIC18F6525 and PIC18F8525 each have 48 Kbytes of on-chip Flash memory, while the PIC18F6621 and PIC18F8621 have 64 Kbytes of Flash. This means that PIC18FX525 devices can store internally up to 24,576 single-word instructions and PIC18FX621 devices can store up to 32,768 single-word instructions.

The Reset vector address is at 0000h and the interrupt vector addresses are at 0008h and 0018h.

Figure 4-1 shows the program memory map for PIC18FX525 devices, while Figure 4-2 shows the program memory map for PIC18FX621 devices.

4.1.1 PIC18F6525/6621/8525/8621 PROGRAM MEMORY MODES

PIC18F8525/8621 devices differ significantly from their PIC18 predecessors in their utilization of program memory. In addition to available on-chip Flash program memory, these controllers can also address up to 2 Mbytes of external program memory through the external memory interface. There are four distinct operating modes available to the controllers:

- Microprocessor (MP)
- Microprocessor with Boot Block (MPBB)
- Extended Microcontroller (EMC)
- Microcontroller (MC)

The Program Memory mode is determined by setting the two Least Significant bits of the CONFIG3L Configuration Byte register as shown in Register 4-1 (see **Section 24.1 "Configuration Bits**" for additional details on the device configuration bits).

The Program Memory modes operate as follows:

- The **Microprocessor Mode** permits access only to external program memory; the contents of the on-chip Flash memory are ignored. The 21-bit program counter permits access to a 2-Mbyte linear program memory space.
- The Microprocessor with Boot Block Mode accesses on-chip Flash memory from addresses 000000h to 0007FFh. Above this, external program memory is accessed all the way up to the 2-Mbyte limit. Program execution automatically switches between the two memories as required.
- The Microcontroller Mode accesses only on-chip Flash memory. Attempts to read above the physical limit of the on-chip Flash (BFFFh for the PIC18FX525, FFFFh for the PIC18FX621) causes a read of all '0's (a NOP instruction). The Microcontroller mode is also the only operating mode available to PIC18F6525/6621 devices.
- The Extended Microcontroller Mode allows access to both internal and external program memories as a single block. The device can access its entire on-chip Flash memory; above this, the device accesses external program memory up to the 2-Mbyte program space limit. As with Boot Block mode, execution automatically switches between the two memories as required.

In all modes, the microcontroller has complete access to data RAM and EEPROM.

Figure 4-3 compares the memory maps of the different program memory modes. The differences between on-chip and external memory access limitations are more fully explained in Table 4-1.

4.7.1 TWO-WORD INSTRUCTIONS

The PIC18F6525/6621/8525/8621 devices have four two-word instructions: MOVFF, CALL, GOTO and LFSR. The second word of these instructions has the 4 MSBs set to '1's and is a special kind of NOP instruction. The lower 12 bits of the second word contain data to be used by the instruction. If the first word of the instruction is executed, the data in the second word is accessed. If the second word of the instruction is executed by itself (first word was skipped), it will execute as a NOP. This action is necessary when the two-word instruction is preceded by a conditional instruction that changes the PC. A program example that demonstrates this concept is shown in Example 4-3. Refer to **Section 25.0** "Instruction Set Summary" for further details of the instruction set.

EXAMPLE 4-3:	TWO-WORD INSTRUCTIONS

CASE 1:							
Object Code	Source Code						
0110 0110 0000 0000	TSTFSZ REG1 ; is RAM location 0?						
1100 0001 0010 0011	MOVFF REG1, REG2 ; No, execute 2-word instruction						
1111 0100 0101 0110	; 2nd operand holds address of REG2						
0010 0100 0000 0000	ADDWF REG3 ; continue code						

CASE 2:

Object Code	Source Code	
0110 0110 0000 0000	TSTFSZ REG1	; is RAM location 0?
1100 0001 0010 0011	MOVFF REG1, REG2	; Yes
1111 0100 0101 0110		; 2nd operand becomes NOP
0010 0100 0000 0000	ADDWF REG3	; continue code

4.8 Look-up Tables

Look-up tables are implemented two ways. These are:

- Computed GOTO
- Table Reads

4.8.1 COMPUTED GOTO

A computed GOTO is accomplished by adding an offset to the program counter (ADDWF PCL).

A look-up table can be formed with an ADDWF PCL instruction and a group of RETLW 0xnn instructions. WREG is loaded with an offset into the table before executing a call to that table. The first instruction of the called

routine is the ADDWF PCL instruction. The next instruction executed will be one of the RETLW 0xnn instructions that returns the value 0xnn to the calling function.

The offset value (value in WREG) specifies the number of bytes that the program counter should advance.

In this method, only one data byte may be stored in each instruction location and room on the return address stack is required.

Note: The ADDWF PCL instruction does not update PCLATH and PCLATU. A read operation on PCL must be performed to update PCLATH and PCLATU.

EXAMPLE 4-4: COMPUTED GOTO USING AN OFFSET VALUE

MAIN:	ORG MOVLW CALL	0x0000 0x00 TABLE	
	ORG	0x8000	
TABLE	MOVF	PCL, F	; A simple read of PCL will update PCLATH, PCLATU
	RLNCF	W, W	; Multiply by 2 to get correct offset in table
	ADDWF	PCL	; Add the modified offset to force jump into table
	RETLW	`A'	
	RETLW	`B′	
	RETLW	`C′	
	RETLW	'D'	
	RETLW	`E′	
	END		

NOTES:

If the device fetches or accesses external memory while EBDIS = 1, the pins will switch to external bus. If the EBDIS bit is set by a program executing from external memory, the action of setting the bit will be delayed until the program branches into the internal memory. At that time, the pins will change from external bus to I/O ports.

When the device is executing out of internal memory (EBDIS = 0) in Microprocessor with Boot Block mode or Extended Microcontroller mode, the control signals will NOT be active. They will go to a state where the AD<15:0> and A<19:16> are tri-state; the \overline{CE} , \overline{OE} , WRH, WRL, UB and LB signals are '1' and ALE and BA0 are '0'.

Name	Port	Bit	Function
RD0/AD0	PORTD	bit 0	Input/Output or System Bus Address bit 0 or Data bit 0
RD1/AD1	PORTD	bit 1	Input/Output or System Bus Address bit 1 or Data bit 1
RD2/AD2	PORTD	bit 2	Input/Output or System Bus Address bit 2 or Data bit 2
RD3/AD3	PORTD	bit 3	Input/Output or System Bus Address bit 3 or Data bit 3
RD4/AD4	PORTD	bit 4	Input/Output or System Bus Address bit 4 or Data bit 4
RD5/AD5	PORTD	bit 5	Input/Output or System Bus Address bit 5 or Data bit 5
RD6/AD6	PORTD	bit 6	Input/Output or System Bus Address bit 6 or Data bit 6
RD7/AD7	PORTD	bit 7	Input/Output or System Bus Address bit 7 or Data bit 7
RE0/AD8	PORTE	bit 0	Input/Output or System Bus Address bit 8 or Data bit 8
RE1/AD9	PORTE	bit 1	Input/Output or System Bus Address bit 9 or Data bit 9
RE2/AD10	PORTE	bit 2	Input/Output or System Bus Address bit 10 or Data bit 10
RE3/AD11	PORTE	bit 3	Input/Output or System Bus Address bit 11 or Data bit 11
RE4/AD12	PORTE	bit 4	Input/Output or System Bus Address bit 12 or Data bit 12
RE5/AD13	PORTE	bit 5	Input/Output or System Bus Address bit 13 or Data bit 13
RE6/AD14	PORTE	bit 6	Input/Output or System Bus Address bit 14 or Data bit 14
RE7/AD15	PORTE	bit 7	Input/Output or System Bus Address bit 15 or Data bit 15
RH0/A16	PORTH	bit 0	Input/Output or System Bus Address bit 16
RH1/A17	PORTH	bit 1	Input/Output or System Bus Address bit 17
RH2/A18	PORTH	bit 2	Input/Output or System Bus Address bit 18
RH3/A19	PORTH	bit 3	Input/Output or System Bus Address bit 19
RJ0/ALE	PORTJ	bit 0	Input/Output or System Bus Address Latch Enable (ALE) Control pin
RJ1/OE	PORTJ	bit 1	Input/Output or System Bus Output Enable (OE) Control pin
RJ2/WRL	PORTJ	bit 2	Input/Output or System Bus Write Low (WRL) Control pin
RJ3/WRH	PORTJ	bit 3	Input/Output or System Bus Write High (WRH) Control pin
RJ4/BA0	PORTJ	bit 4	Input/Output or System Bus Byte Address bit 0
RJ5/CE	PORTJ	bit 5	Input/Output or System Bus Chip Enable (CE) Control pin
RJ6/LB	PORTJ	bit 6	Input/Output or System Bus Lower Byte Enable (IB) Control pin
RJ7/UB	PORTJ	bit 7	Input/Output or System Bus Upper Byte Enable (UB) Control pin

TABLE 6-1: PIC18F8525/8621 EXTERNAL BUS – I/O PORT FUNCTIONS

9.2 PIR Registers

The PIR registers contain the individual flag bits for the peripheral interrupts. Due to the number of peripheral interrupt sources, there are three Peripheral Interrupt Request Flag registers (PIR1, PIR2 and PIR3).

- Note 1: Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the Global Interrupt Enable bit, GIE (INTCON<7>).
 - 2: User software should ensure the appropriate interrupt flag bits are cleared prior to enabling an interrupt and after servicing that interrupt.

REGISTER 9-4: PIR1: PERIPHERAL INTERRUPT REQUEST (FLAG) REGISTER 1

R/W-0	R/W-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
PSPIF ⁽¹⁾	ADIF	RC1IF	TX1IF	SSPIF	CCP1IF	TMR2IF	TMR1IF
bit 7							bit 0

bit 7	PSPIF: Parallel Slave Port Read/Write Interrupt Flag bit ⁽¹⁾
	 1 = A read or a write operation has taken place (must be cleared in software) 0 = No read or write has occurred
	Note 1: Enabled only in Microcontroller mode for PIC18F8525/8621 devices.
bit 6	ADIF: A/D Converter Interrupt Flag bit
	 1 = An A/D conversion completed (must be cleared in software) 0 = The A/D conversion is not complete
bit 5	RC1IF: USART1 Receive Interrupt Flag bit
	 1 = The USART1 receive buffer, RCREGx, is full (cleared when RCREGx is read) 0 = The USART1 receive buffer is empty
bit 4	TX1IF: USART1 Transmit Interrupt Flag bit
	 1 = The USART1 transmit buffer, TXREGx, is empty (cleared when TXREGx is written) 0 = The USART1 transmit buffer is full
bit 3	SSPIF: Master Synchronous Serial Port Interrupt Flag bit
	 1 = The transmission/reception is complete (must be cleared in software) 0 = Waiting to transmit/receive
bit 2	CCP1IF: ECCP1 Interrupt Flag bit
	<u>Capture mode:</u> 1 = A TMR1 register capture occurred (must be cleared in software) 0 = No TMR1 register capture occurred
	<u>Compare mode:</u> 1 = A TMR1 register compare match occurred (must be cleared in software) 0 = No TMR1 register compare match occurred
	<u>PWM mode:</u> Unused in this mode.
bit 1	TMR2IF: TMR2 to PR2 Match Interrupt Flag bit
	1 = TMR2 to PR2 match occurred (must be cleared in software)0 = No TMR2 to PR2 match occurred
bit 0	TMR1IF: TMR1 Overflow Interrupt Flag bit
	1 = TMR1 register overflowed (must be cleared in software)0 = TMR1 register did not overflow
	Lagand

Legena.			
R = Readable bit	W = Writable bit	U = Unimplemented bit,	read as '0'
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

EGISTER 9-6: PIR3: PERIPHERAL INTERRUPT REQUEST (FLAG) REGISTER 3								
	U-0	U-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
		_	RC2IF	TX2IF	TMR4IF	CCP5IF	CCP4IF	CCP3IF
	bit 7							bit 0
bit 7-6	Unimplem	ented: Rea	d as '0'					
bit 5	RC2IF: US	ART2 Rece	ive Interrupt	Flag bit				
	1 = The U 0 = The U	SART2 rece SART2 rece	eive buffer, F eive buffer is	RCREGx, is t empty	full (cleared	when RCRE	EGx is read)	
bit 4	TX2IF: US	ART2 Trans	mit Interrup	t Flag bit				
	1 = The U 0 = The U	SART2 tran SART2 tran	smit buffer, ⁻ smit buffer i	TXREGx, is s full	empty (clea	red when T	KREGx is wi	ritten)
bit 3	TMR4IF: TMR3 Overflow Interrupt Flag bit							
	 1 = TMR4 register overflowed (must be cleared in software) 0 = TMR4 register did not overflow 							
bit 2-0	CCPxIF: CCPx Interrupt Flag bit (ECCP3, CCP4 and CCP5)							
	<u>Capture m</u> 1 = A TMF 0 = No TM	<u>Capture mode:</u> 1 = A TMR1 or TMR3 register capture occurred (must be cleared in software) 0 = No TMR1 or TMR3 register capture occurred						
<u>Compare mode:</u> 1 = A TMR1 or TMR3 register compare match occurred (must be cleared in software) 0 = No TMR1 or TMR3 register compare match occurred						are)		
	<u>PWM mod</u> Unused in	<u>e:</u> this mode.						
	Legend:							

Legena:			
R = Readable bit	W = Writable bit	U = Unimplemented b	oit, read as '0'
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

17.1.3 ECCP MODULES AND TIMER RESOURCES

Like the standard CCP modules, the ECCP modules can utilize Timers 1, 2, 3 or 4, depending on the mode selected. Timer1 and Timer3 are available for modules in Capture or Compare modes, while Timer2 and Timer4 are available for modules in PWM mode. Additional details on timer resources are provided in Section 16.1.1 "CCP Modules and Timer Resources".

17.2 Capture and Compare Modes

Except for the operation of the special event trigger discussed below, the Capture and Compare modes of the ECCP module are identical in operation to that of CCP4. These are discussed in detail in Section 16.2 "Capture Mode" and Section 16.3 "Compare Mode".

17.2.1 SPECIAL EVENT TRIGGER

In this mode, an internal hardware trigger is generated in Compare mode, on a match between the CCPR register pair and the selected timer. This can be used in turn to initiate an action.

The special event trigger output of either ECCP1 or ECCP2 resets the TMR1 or TMR3 register pair, depending on which timer resource is currently selected. This allows the CCPRx register to effectively be a 16-bit programmable period register for Timer1 or Timer3. In addition, the ECCP2 special event trigger will also start an A/D conversion if the A/D module is enabled.

The triggers are not implemented for ECCP3, CCP4 or CCP5. Selecting the Special Event mode (CCPxM3:CCPxM0 = 1011) for these modules has the same effect as selecting the Compare with Software Interrupt mode (CCPxM3:CCPxM0 = 1010).

Note: The special event trigger from ECCP2 will not set the Timer1 or Timer3 interrupt flag bits.

17.3 Standard PWM Mode

When configured in Single Output mode, the ECCP module functions identically to the standard CCP module in PWM mode as described in **Section 16.4** "**PWM Mode**". This is also sometimes referred to as "Compatible CCP" mode as in Tables 17-1 through 17-3.

Note:	When setting up single output PWM opera- tions, users are free to use either of the
	processes described in Section 16.4.3
	"Setup for PWM Operation" or
	Section 17.4.9 "Setup for PWM Opera-
	tion". The latter is more generic but will
	work for either single or multi-output PWM.

17.4 Enhanced PWM Mode

The Enhanced PWM mode provides additional PWM output options for a broader range of control applications. The module is a backward compatible version of the standard CCP module and offers up to four outputs, designated PxA through PxD. Users are also able to select the polarity of the signal (either active-high or active-low). The module's output mode and polarity are configured by setting the PxM1:PxM0 and CCPxM3CCPxM0 bits of the CCPxCON register (CCPxCON<7:6> and CCPxCON<3:0>, respectively).

For the sake of clarity, Enhanced PWM mode operation is described generically throughout this section with respect to ECCP1 and TMR2 modules. Control register names are presented in terms of ECCP1. All three Enhanced modules, as well as the two timer resources, can be used interchangeably and function identically. TMR2 or TMR4 can be selected for PWM operation by selecting the proper bits in T3CON.

Figure 17-1 shows a simplified block diagram of PWM operation. All control registers are double-buffered and are loaded at the beginning of a new PWM cycle (the period boundary when Timer2 resets) in order to prevent glitches on any of the outputs. The exception is the PWM Delay register, ECCP1DEL, which is loaded at either the duty cycle boundary or the boundary period (whichever comes first). Because of the buffering, the module waits until the assigned timer resets instead of starting immediately. This means that Enhanced PWM waveforms do not exactly match the standard PWM waveforms, but are instead offset by one full instruction cycle (4 Tosc).

As before, the user must manually configure the appropriate TRIS bits for output.

17.4.1 PWM PERIOD

The PWM period is specified by writing to the PR2 register. The PWM period can be calculated using the equation:

EQUATION 17-1:

 $PWM Period = [(PR2) + 1] \bullet 4 \bullet TOSC \bullet$ (TMR2 Prescale Value)

PWM frequency is defined as 1/[PWM period]. When TMR2 is equal to PR2, the following three events occur on the next increment cycle:

- TMR2 is cleared
- The ECCP1 pin is set (if PWM duty cycle = 0%, the ECCP1 pin will not be set)
- The PWM duty cycle is copied from CCPR1L into CCPR1H
 - Note: The Timer2 postscaler (see Section 13.0 "Timer2 Module") is not used in the determination of the PWM frequency. The postscaler could be used to have a servo update rate at a different frequency than the PWM output.



17.4.2 PWM DUTY CYCLE

The PWM duty cycle is specified by writing to the CCPR1L register and to the CCP1CON<5:4> bits. Up to 10-bit resolution is available. The CCPR1L contains the eight MSbs and the CCP1CON<5:4> contains the two LSbs. This 10-bit value is represented by CCPRxL:CCPxCON<5:4>. The PWM duty cycle is calculated by the equation:

EQUATION 17-2:

PWM Duty Cycle = (CCPR1L:CCP1CON<5:4>) • Tosc • (TMR2 Prescale Value)

CCPR1L and CCP1CON<5:4> can be written to at any time but the duty cycle value is not copied into CCPR1H until a match between PR2 and TMR2 occurs (i.e., the period is complete). In PWM mode, CCPR1H is a read-only register.

The CCPRxH register and a 2-bit internal latch are used to double-buffer the PWM duty cycle. This double-buffering is essential for glitchless PWM operation. When the CCPR1H and 2-bit latch match TMR2, concatenated with an internal 2-bit Q clock or two bits of the TMR2 prescaler, the ECCP1 pin is cleared. The maximum PWM resolution (bits) for a given PWM frequency is given by the equation:

EQUATION 17-3:



Note: If the PWM duty cycle value is longer than the PWM period, the ECCP1 pin will not be cleared.

TABLE 17-4: EXA	IPLE PWM FREQUENCIES AND RESOLUTIONS AT 40 MHz
-----------------	--

PWM Frequency	2.44 kHz	9.77 kHz	39.06 kHz	156.25 kHz	312.50 kHz	416.67 kHz
Timer Prescaler (1, 4, 16)	16	4	1	1	1	1
PR2 Value	FFh	FFh	FFh	3Fh	1Fh	17h
Maximum Resolution (bits)	10	10	10	8	7	6.58





18.4.14 SLEEP OPERATION

While in Sleep mode, the I²C module can receive addresses or data and when an address match or complete byte transfer occurs, wake the processor from Sleep (if the MSSP interrupt is enabled).

18.4.15 EFFECT OF A RESET

A Reset disables the MSSP module and terminates the current transfer.

18.4.16 MULTI-MASTER MODE

In Multi-Master mode, the interrupt generation on the detection of the Start and Stop conditions allows the determination of when the bus is free. The Stop (P) and Start (S) bits are cleared from a Reset or when the MSSP module is disabled. Control of the I²C bus may be taken when the P bit (SSPSTAT<4>) is set, or the bus is Idle with both the S and P bits clear. When the bus is busy, enabling the MSSP interrupt will generate the interrupt when the Stop condition occurs.

In multi-master operation, the SDA line must be monitored for arbitration to see if the signal level is the expected output level. This check is performed in hardware with the result placed in the BCLIF bit.

The states where arbitration can be lost are:

- · Address Transfer
- Data Transfer
- A Start Condition
- A Repeated Start Condition
- An Acknowledge Condition

18.4.17 MULTI-MASTER COMMUNICATION, BUS COLLISION AND BUS ARBITRATION

Multi-Master mode support is achieved by bus arbitration. When the master outputs address/data bits onto the SDA pin, arbitration takes place when the master outputs a '1' on SDA, by letting SDA float high and another master asserts a '0'. When the SCL pin floats high, data should be stable. If the expected data on SDA is a '1' and the data sampled on the SDA pin = 0, then a bus collision has taken place. The master will set the Bus Collision Interrupt Flag, BCLIF and reset the I^2C port to its Idle state (Figure 18-25).

If a transmit was in progress when the bus collision occurred, the transmission is halted, the BF flag is cleared, the SDA and SCL lines are deasserted and the SSPBUF can be written to. When the user services the bus collision Interrupt Service Routine and if the I^2C bus is free, the user can resume communication by asserting a Start condition.

If a Start, Repeated Start, Stop or Acknowledge condition was in progress when the bus collision occurred, the condition is aborted, the SDA and SCL lines are deasserted and the respective control bits in the SSPCON2 register are cleared. When the user services the bus collision Interrupt Service Routine and if the I^2C bus is free, the user can resume communication by asserting a Start condition.

The master will continue to monitor the SDA and SCL pins. If a Stop condition occurs, the SSPIF bit will be set.

A write to the SSPBUF will start the transmission of data at the first data bit regardless of where the transmitter left off when the bus collision occurred.

In Multi-Master mode, the interrupt generation on the detection of Start and Stop conditions allows the determination of when the bus is free. Control of the I^2C bus can be taken when the P bit is set in the SSPSTAT register, or the bus is Idle and the S and P bits are cleared.

FIGURE 18-25: BUS COLLISION TIMING FOR TRANSMIT AND ACKNOWLEDGE



REGISTER 24-1:	CONFIG1	H: CONFIC	GURATION	REGISTE	R 1 HIGH (BYTE AD	DRESS 30	0001h)
	U-0	U-0	R/P-1	U-0	R/P-1	R/P-1	R/P-1	R/P-1
		_	OSCSEN	—	FOSC3	FOSC2	FOSC1	FOSC0
	bit 7							bit 0
bit 7 C	Unimplom	ented. Doo						
Dit 7-6		ented: Real		Owitch Ench				
DIT 5	USCSEN:	Uscillator Sy	Stem Clock	Switch Enac			,	
	1 = Oscilla0 = Timer1	oscillator s	vstem clock	option is disa switch optioi	abled (main n is enabled	oscillator is (oscillator s	source) witching is e	enabled)
bit 4	Unimplem	ented: Read	d as '0'				C C	
bit 3-0	FOSC3:FO	SC0: Oscill	ator Selection	n bits				
	1111 = RC 1110 = HS 1101 = EC 1000 = EC 1011 = Re 1000 = Re 1000 = Re 0111 = RC 0110 = HS 0101 = EC 0010 = HS 0001 = XT 0000 = LP	oscillator w oscillator w oscillator w oscillator w oscillator w served; do r served; do r served; do r served; do r oscillator w oscillator w oscillator w oscillator w oscillator oscillator oscillator	ith OSC2 co ith SW enab ith OSC2 con ith OSC2 con not use not use not use rith OSC2 co ith HW enab ith OSC2 co ith OSC2 co ith OSC2 co	nfigured as I led 4x PLL nfigured as I nfigured as I led 4x PLL nfigured as I nfigured as o nfigured as o	RA6 and SW RA6 and HV RA6 RA6 divide by 4 c divide by 4 c	V enabled 4: V enabled 4 Clock output	x PLL x PLL	
	Legend:							
	R = Reada	ble bit	P = Progra	ammable bit	U = Unim	nplemented	bit, read as	'0'

-n – Value when device is unprogrammed	u – Unchanged from programmed state
	u – Onchangeu nom programmeu state

Dh)			
, U-0			
bit 0			
ks			
u = Unchanged from programmed state			
ES			
_			
R			
EV0			
bit 0			

bit 4-0 **REV4:REV0:** Revision ID bits

These bits are used to indicate the device revision.

Legend:		
R = Readable bit	P = Programmable bit	U = Unimplemented bit, read as '0'
-n = Value when device	e is unprogrammed	u = Unchanged from programmed state

REGISTER 24-14: DEVID2: DEVICE ID REGISTER 2 FOR PIC18F6525/6621/8525/8621 DEVICES (ADDRESS 3FFFFh)

R-0	R-0	R-0	R-0	R-1	R-0	R-1	R-0
DEV10	DEV9	DEV8	DEV7	DEV6	DEV5	DEV4	DEV3
bit 7							bit 0

bit 7-0 DEV10:DEV3: Device ID bits

These bits are used with the DEV2:DEV0 bits in the Device ID Register 1 to identify the part number.

0000 1010 = PIC18F6525/6621/8525/8621

Legend:		
R = Readable bit	P = Programmable bit	U = Unimplemented bit, read as '0'
-n = Value when device	is unprogrammed	u = Unchanged from programmed state

MUL	LW.	iteral with	w				
Synta	ax:	[label] N	[<i>label</i>] MULLW k				
Oper	ands:	$0 \le k \le 255$					
Oper	ation:	(W) x k \rightarrow F	PRODH:PRO	DL			
Statu	s Affected:	None					
Enco	ding:	0000	1101 kł	kk kkkk			
Desc	ription:	An unsigned multiplication is carried out between the contents of W and the 8-bit literal 'k'. The 16-bit result is placed in PRODH:PRODL register pair. PRODH contains the high byte. W is unchanged. None of the Status flags are affected. Note that neither overflow nor carry is possible in this operation. A zero result is possible but not detected.					
Word	ls:	1	1				
Cycle	es:	1					
QC	ycle Activity:						
	Q1	Q2	Q3	Q4			
	Decode	Read literal 'k'	Process Data	Write registers PRODH: PRODL			
<u>Exan</u>	nple:	MULLW	0xC4				
	Before Instruct	tion					
	W PRODH PRODL	= 0xl = ? = ?	E2				
	PRODH PRODL	= 0x = 0x = 0x	E2 AD 08				

Synta		Multiply	W with	f				
	ax:	[label] N	IULWF	f [,a]				
Oper	ands:	0 ≤ f ≤ 255 a ∈ [0,1]	0 ≤ f ≤ 255 a ∈ [0,1]					
Oper	ation:	(W) x (f) \rightarrow	PRODH	I:PROD	L			
Statu	is Affected:	None						
Enco	oding:	0000	001a	ffff	ffff			
		register file is stored in register pa byte. Both W an None of the Note that n possible in is possible the Access overriding 'a' = 1, the as per the	location the PRO ir. PROD d 'f' are u e Status either ov this ope but not o Bank wi the BSR n the bar BSR valu	't'. The DDH:PR H conta unchang flags are verflow r ration. A detected ill be se value. I hk will b ue (defa	16-bit result ODL ins the high led. e affected. nor carry is a zero result I. If 'a' is '0', lected, f e selected ult).			
Word	ls:	1		,	,			
0	es:	1						
Cycle								
	ycle Activity:							
Q C	ycle Activity: Q1	Q2	Q	3	Q4			

0xC4

0xB5

0x8A

0x94

? ? =

=

=

= = =

PRODH

PRODL

After Instruction W

REG

PRODH

PRODL

26.20 PICDEM 17 Demonstration Board

The PICDEM 17 demonstration board is an evaluation board that demonstrates the capabilities of several Microchip microcontrollers, including PIC17C752, PIC17C756A, PIC17C762 and PIC17C766. A programmed sample is included. The PRO MATE II device programmer, or the PICSTART Plus development programmer, can be used to reprogram the device for user tailored application development. The PICDEM 17 demonstration board supports program download and execution from external on-board Flash memory. A generous prototype area is available for user hardware expansion.

26.21 PICDEM 18R PIC18C601/801 Demonstration Board

The PICDEM 18R demonstration board serves to assist development of the PIC18C601/801 family of Microchip microcontrollers. It provides hardware implementation of both 8-bit Multiplexed/Demultiplexed and 16-bit Memory modes. The board includes 2 Mb external Flash memory and 128 Kb SRAM memory, as well as serial EEPROM, allowing access to the wide range of memory types supported by the PIC18C601/801.

26.22 PICDEM LIN PIC16C43X Demonstration Board

The powerful LIN hardware and software kit includes a series of boards and three PIC microcontrollers. The small footprint PIC16C432 and PIC16C433 are used as slaves in the LIN communication and feature onboard LIN transceivers. A PIC16F874 Flash microcontroller serves as the master. All three microcontrollers are programmed with firmware to provide LIN bus communication.

26.23 PICkit[™] 1 Flash Starter Kit

A complete "development system in a box", the PICkit[™] Flash Starter Kit includes a convenient multi-section board for programming, evaluation and development of 8/14-pin Flash PIC[®] microcontrollers. Powered via USB, the board operates under a simple Windows GUI. The PICkit 1 Starter Kit includes the User's Guide (on CD ROM), PICkit 1 tutorial software and code for various applications. Also included are MPLAB[®] IDE (Integrated Development Environment) software, software and hardware "Tips 'n Tricks for 8-pin Flash PIC[®] Microcontrollers" Handbook and a USB interface cable. Supports all current 8/14-pin Flash PIC microcontrollers, as well as many future planned devices.

26.24 PICDEM USB PIC16C7X5 Demonstration Board

The PICDEM USB Demonstration Board shows off the capabilities of the PIC16C745 and PIC16C765 USB microcontrollers. This board provides the basis for future USB products.

26.25 Evaluation and Programming Tools

In addition to the PICDEM series of circuits, Microchip has a line of evaluation kits and demonstration software for these products.

- KEELOQ evaluation and programming tools for Microchip's HCS Secure Data Products
- CAN developers kit for automotive network applications
- Analog design boards and filter design software
- PowerSmart battery charging evaluation/ calibration kits
- IrDA[®] development kit
- microID development and rfLab[™] development software
- SEEVAL[®] designer kit for memory evaluation and endurance calculations
- PICDEM MSC demo boards for Switching mode power supply, high-power IR driver, delta sigma ADC and flow rate sensor

Check the Microchip web page and the latest Product Selector Guide for the complete list of demonstration and evaluation kits.

DC Characteristics			$\begin{array}{l} \mbox{Standard Operating Conditions (unless otherwise stated)} \\ \mbox{Operating temperature } -40^\circ C \leq TA \leq +85^\circ C \mbox{ for industrial} \\ -40^\circ C \leq TA \leq +125^\circ C \mbox{ for extended} \end{array}$					
Param No.	Sym	Characteristic	Min	Тур†	Max	Units	Conditions	
		Internal Program Memory Programming Specifications						
D110	Vpp	Voltage on MCLR/VPP pin	9.00	—	13.25	V	(Note 2)	
D112	IPP	Current into MCLR/VPP pin	—	—	300	μΑ		
D113	IDDP	Supply Current during Programming	—	—	1.0	mA		
		Data EEPROM Memory						
D120	ED	Byte Endurance	100K 10K	1M 100K	—	E/W E/W	-40°C to +85°C -40°C to +125°C	
D121	Vdrw	VDD for Read/Write	Vmin	_	5.5	V	Using EECON to read/write VMIN = Minimum operating voltage	
D122	TDEW	Erase/Write Cycle Time	—	4	—	ms		
D123	TRETD	Characteristic Retention	40	—	—	Year	Provided no other specifications are violated	
D124	Tref	Number of Total Erase/Write Cycles before Refresh ⁽¹⁾	1M 100K	10M 1M	—	E/W E/W	-40°C to +85°C -40°C to +125°C	
		Program Flash Memory						
D130	Eр	Cell Endurance	10K 1K	100K 10K	—	E/W E/W	-40°C to +85°C -40°C to +125°C	
D131	Vpr	VDD for Read	VMIN	—	5.5	V	VMIN = Minimum operating voltage	
D132	VIE	VDD for Block Erase	4.5	—	5.5	V	Using ICSP™ port	
D132A	Viw	VDD for Externally Timed Erase or Write	4.5	—	5.5	V	Using ICSP port	
D132B	Vpew	VDD for Self-Timed Write and Row Erase	VMIN	—	5.5	V	VMIN = Minimum operating voltage	
D133	TIE	ICSP Block Erase Cycle Time		4	_	ms	VDD > 4.5V	
D133A	Tiw	ICSP Erase or Write Cycle Time (externally timed)	1	—	—	ms	VDD > 4.5V	
D133A	Tiw	Self-Timed Write Cycle Time	—	2	—	ms		
D134	TRETD	Characteristic Retention	40	—	—	Year	Provided no other specifications are violated	

TABLE 27-4: MEMORY PROGRAMMING REQUIREMENTS

† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: Refer to Section 7.8 "Using the Data EEPROM" for a more detailed discussion on data EEPROM endurance.

2: Required only if Low-Voltage Programming is disabled.







FIGURE 28-27: MINIMUM AND MAXIMUM VIN vs. VDD (I²C INPUT, -40°C TO +125°C)



FIGURE 28-29: A/D NONLINEARITY vs. VREFH (VDD = 5V, -40°C TO +125°C)

Slave Synchronization
Slow Rise Time (MCLR Tied to VDD
via 1 k Ω Resistor)
SPI Mode (Master Mode)
SPI Mode (Slave Mode with CKE = 0)
SPI Mode (Slave Mode with CKE = 1)
Stop Condition Receive or Transmit Mode
Synchronous Reception
(Master Mode, SREN)
Synchronous Transmission
Synchronous Transmission (Through TXEN)
Time-out Sequence on POR w/PLL Enabled
$(\overline{MCLR}$ Tied to VDD via 1 k Ω Resistor)
Time-out Sequence on Power-up (MCLR
Not Tied to VDD): Case 1
Time-out Sequence on Power-up (MCLR
Not Tied to VDD): Case 2
Time-out Sequence on Power-up (MCLR
Tied to VDD via 1 k Ω Resistor)
Timer0 and Timer1 External Clock
Timing for Transition Between Timer1 and
OSC1 (EC with PLL Active, SCS1 = 1)27
Timing for Transition Between Timer1 and
OSC1 (HS with PLL Active, SCS1 = 1)27
Transition Between Timer1 and
OSC1 (HS, XT, LP)26
Transition Between Timer1 and
OSC1 (RC, EC)28
Transition from OSC1 to Timer1 Oscillator26
Wake-up from Sleep via Interrupt270
Timing Specifications
A/D Conversion Requirements
Capture/Compare/PWM Requirements
CLKO and I/O Requirements
EUSART Synchronous Receive
Requirements353
EUSART Synchronous Transmission
Requirements353
Example SPI Mode Requirements
(Master Mode, CKE = 0)
Example SPI Mode Requirements
(Master Mode, CKE = 1)
Example SPI Mode Requirements
(Slave Mode, CKE = 0)
Example SPI Slave Mode
Requirements (CKE = 1)

External Clock Requirements					
I ² C Bus Data Requirements (Slave Mode)					
I ² C Bus Start/Stop Bits Requirements					
(Slave Mode) 349					
Master SSP I ² C Bus Data Requirements					
Master SSP I ² C Bus Start/Stop Bits					
Requirements					
Parallel Slave Port Requirements					
PLL Clock					
Program Memory Read Requirements					
Program Memory Write Requirements					
Reset, Watchdog Timer, Oscillator					
Start-up Timer, Power-up Timer					
and Brown-out Reset Requirements					
Timer() and Timer1 External					
Clock Requirements 342					
F 3F MODE DIL 111, 120					
Type Word Instructions					
Example Coope					
Example Cases 40					
V					
Voltago Poforonco Specificatione 222					
W					
Wake-up from Sleep 259, 269					
Using Interrupts 269					
Watchdog Timer (WDT) 259, 267					
Accorded Pogisters					
Control Dogistor					
Control Register					
F USISUAIEI					
Programming Considerations					
KU Uscillator					
Lime-out Period					
WCOL					
WOUL Status Flag					
WWW, On-Line Support 5					

Х

XORLW	5
XORWF	6