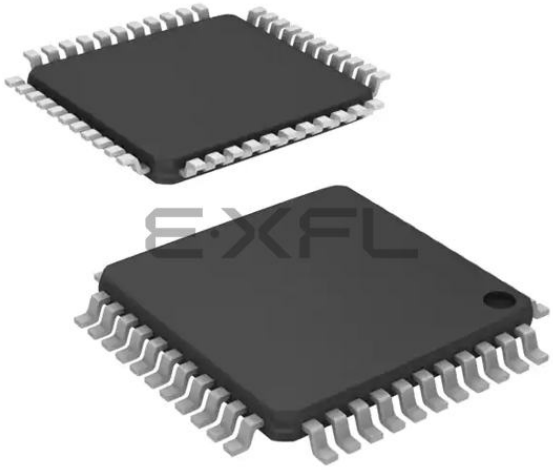


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	48MHz
Connectivity	I²C, LINbus, SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	34
Program Memory Size	128KB (64K x 16)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	3.8K x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 2.75V
Data Converters	A/D 13x10b/12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-TQFP
Supplier Device Package	44-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18lf47j53-i-pt

1.0 DEVICE OVERVIEW

This document contains device-specific information for the following devices:

- PIC18F26J53
- PIC18F27J53
- PIC18F46J53
- PIC18F47J53
- PIC18LF26J53
- PIC18LF27J53
- PIC18LF46J53
- PIC18LF47J53

This family introduces a new line of low-voltage Universal Serial Bus (USB) microcontrollers with the main traditional advantage of all PIC18 microcontrollers, namely, high computational performance and a rich feature set at an extremely competitive price point. These features make the PIC18F47J53 family a logical choice for many high-performance applications, where cost is a primary consideration.

1.1 Core Features

1.1.1 XLP TECHNOLOGY

All of the devices in the PIC18F47J53 family incorporate a range of features that can significantly reduce power consumption during operation. Key features are:

- **Alternate Run Modes:** By clocking the controller from the Timer1 source or the internal RC oscillator, power consumption during code execution can be reduced by as much as 90%.
- **Multiple Idle Modes:** The controller can also run with its CPU core disabled but the peripherals still active. In these states, power consumption can be reduced even further, to as little as 4% of normal operational requirements.
- **On-the-Fly Mode Switching:** The power-managed modes are invoked by user code during operation, allowing the users to incorporate power-saving ideas into their application's software design.
- **Deep Sleep:** The 2.5V internal core voltage regulator on F parts can be shutdown to cut power consumption to as low as 15 nA (typical). Certain features can remain operating during Deep Sleep, such as the Real-Time Clock Calendar.
- **Ultra Low Power Wake-Up:** Waking from Sleep or Deep Sleep modes after a period of time can be done without an oscillator/clock source, saving power for applications requiring periodic activity.

1.1.2 UNIVERSAL SERIAL BUS (USB)

Devices in the PIC18F47J53 family incorporate a fully-featured USB communications module with a built-in transceiver that is compliant with the USB Specification Revision 2.0. The module supports both low-speed and full-speed communication for all supported data transfer types.

1.1.3 OSCILLATOR OPTIONS AND FEATURES

All of the devices in the PIC18F47J53 family offer five different oscillator options, allowing users a range of choices in developing application hardware. These include:

- Two Crystal modes, using crystals or ceramic resonators.
- Two External Clock modes, offering the option of a divide-by-4 clock output.
- An internal oscillator block, which provides an 8 MHz clock and an INTRC source (approximately 31 kHz, stable over temperature and V_{DD}), as well as a range of six user-selectable clock frequencies, between 125 kHz to 4 MHz, for a total of eight clock frequencies. This option frees an oscillator pin for use as an additional general purpose I/O.
- A Phase Lock Loop (PLL) frequency multiplier available to the high-speed crystal, and external and internal oscillators, providing a clock speed up to 48 MHz.
- Dual clock operation, allowing the USB module to run from a high-frequency oscillator while the rest of the microcontroller is clocked at a different frequency.

The internal oscillator block provides a stable reference source that gives the PIC18F47J53 family additional features for robust operation:

- **Fail-Safe Clock Monitor:** This option constantly monitors the main clock source against a reference signal provided by the internal oscillator. If a clock failure occurs, the controller is switched to the internal oscillator, allowing for continued low-speed operation or a safe application shutdown.
- **Two-Speed Start-up:** This option allows the internal oscillator to serve as the clock source from Power-on Reset (POR), or wake-up from Sleep mode, until the primary clock source is available.

1.1.4 EXPANDED MEMORY

The PIC18F47J53 family provides ample room for application code, from 64 Kbytes to 128 Kbytes of code space. The Flash cells for program memory are rated to last in excess of 10000 erase/write cycles. Data retention without refresh is conservatively estimated to be greater than 20 years.

The Flash program memory is readable and writable during normal operation. The PIC18F47J53 family also provides plenty of room for dynamic application data with up to 3.8 Kbytes of data RAM.

© 2009-2016 Microchip Technology Inc.

FIGURE 5-3: TIME-OUT SEQUENCE ON POWER-UP (MCLR NOT TIED TO VDD): CASE 1

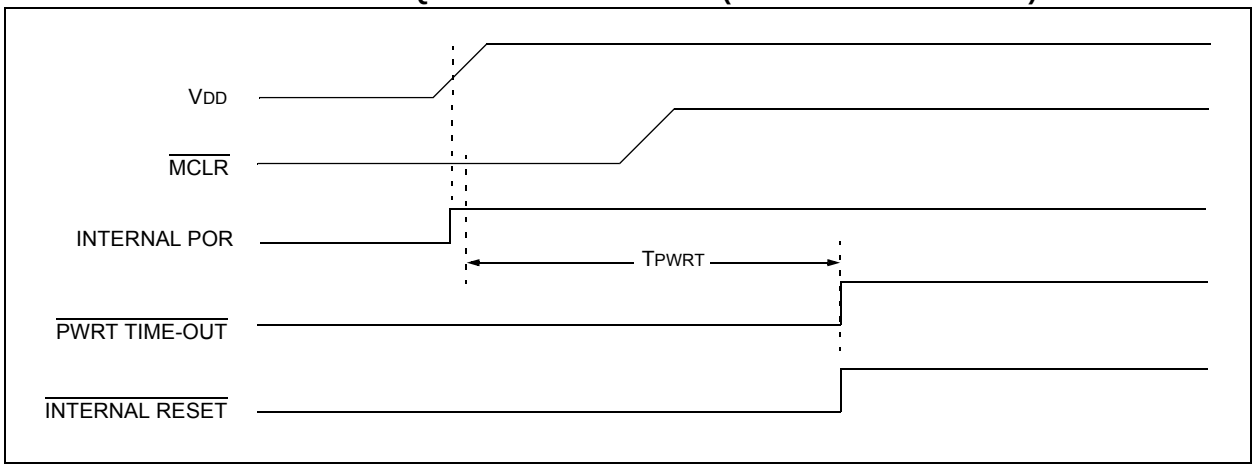


FIGURE 5-4: TIME-OUT SEQUENCE ON POWER-UP (MCLR NOT TIED TO VDD): CASE 2

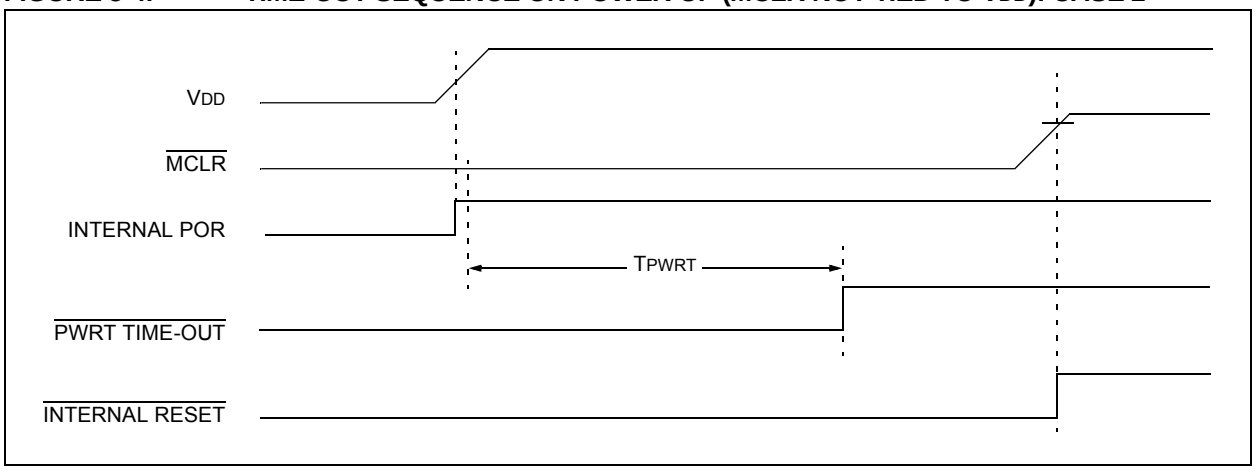
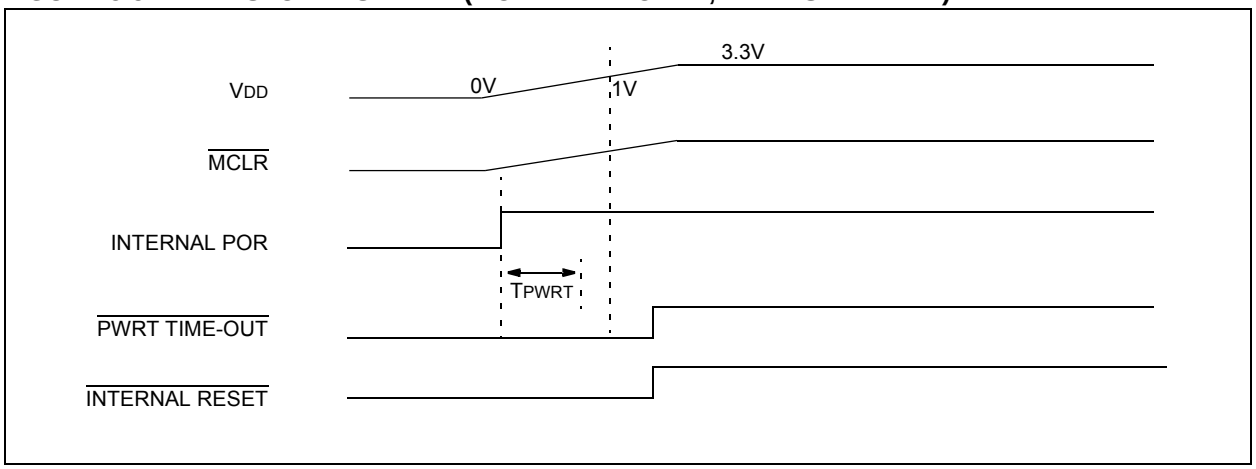


FIGURE 5-5: SLOW RISE TIME (MCLR TIED TO VDD, VDD RISE > TPWRT)



© 2009-2016 Microchip Technology Inc.

TABLE 5-2: INITIALIZATION CONDITIONS FOR ALL REGISTERS (CONTINUED)

Register	Applicable Devices		Power-on Reset, Brown-out Reset, Wake From Deep Sleep	$\overline{\text{MCLR}}$ Resets WDT Reset RESET Instruction Stack Resets CM Resets	Wake-up via WDT or Interrupt
PMCONL	PIC18F2XJ53	PIC18F4XJ53	000- 0000	000- 0000	uuu- uuuu
PMMODEH	PIC18F2XJ53	PIC18F4XJ53	0000 0000	0000 0000	uuuu uuuu
PMMODEL	PIC18F2XJ53	PIC18F4XJ53	0000 0000	0000 0000	uuuu uuuu
PMDOUT2H	PIC18F2XJ53	PIC18F4XJ53	0000 0000	0000 0000	uuuu uuuu
PMDOUT2L	PIC18F2XJ53	PIC18F4XJ53	0000 0000	0000 0000	uuuu uuuu
PMDIN2H	PIC18F2XJ53	PIC18F4XJ53	0000 0000	0000 0000	uuuu uuuu
PMDIN2L	PIC18F2XJ53	PIC18F4XJ53	0000 0000	0000 0000	uuuu uuuu
PMEH	PIC18F2XJ53	PIC18F4XJ53	0000 0000	0000 0000	uuuu uuuu
PMEL	PIC18F2XJ53	PIC18F4XJ53	0000 0000	0000 0000	uuuu uuuu
PMSTATH	PIC18F2XJ53	PIC18F4XJ53	00-- 0000	00-- 0000	uu-- uuuu
PMSTATL	PIC18F2XJ53	PIC18F4XJ53	10-- 1111	10-- 1111	uu-- uuuu
CVRCON	PIC18F2XJ53	PIC18F4XJ53	0000 0000	0000 0000	uuuu uuuu
CCPTMRS0	PIC18F2XJ53	PIC18F4XJ53	0000 0000	0000 0000	uuuu uuuu
CCPTMRS1	PIC18F2XJ53	PIC18F4XJ53	0000 0000	0000 0000	uuuu uuuu
CCPTMRS2	PIC18F2XJ53	PIC18F4XJ53	0000 0000	0000 0000	uuuu uuuu
DSGPR1(6)	PIC18F2XJ53	PIC18F4XJ53	uuuu uuuu	uuuu uuuu	uuuu uuuu
DSGPR0(6)	PIC18F2XJ53	PIC18F4XJ53	uuuu uuuu	uuuu uuuu	uuuu uuuu
DSCONH(6)	PIC18F2XJ53	PIC18F4XJ53	0--- --00	0--- --00	u--- --uu
DSCONL(6)	PIC18F2XJ53	PIC18F4XJ53	---- -000	---- -000	---- -uuu
DSWAKEH(6)	PIC18F2XJ53	PIC18F4XJ53	---- ---q	---- ---0	---- ---u
DSWAKEL(6)	PIC18F2XJ53	PIC18F4XJ53	q-qq qq-q	0-00 00-0	u-uu uu-u
ANCON1	PIC18F2XJ53	PIC18F4XJ53	00-0 0000	uu-u uuuu	uu-u uuuu
ANCON0	PIC18F2XJ53	PIC18F4XJ53	0000 0000	uuuu uuuu	uuuu uuuu
ALRMCFG	PIC18F2XJ53	PIC18F4XJ53	0000 0000	uuuu uuuu	uuuu uuuu
ALRMRPT	PIC18F2XJ53	PIC18F4XJ53	0000 0000	uuuu uuuu	uuuu uuuu
ALRMVALH	PIC18F2XJ53	PIC18F4XJ53	xxxx xxxx	uuuu uuuu	uuuu uuuu
ALRMVALL	PIC18F2XJ53	PIC18F4XJ53	xxxx xxxx	uuuu uuuu	uuuu uuuu
ODCON1	PIC18F2XJ53	PIC18F4XJ53	---- 0000	---- uuuu	---- uuuu
ODCON2	PIC18F2XJ53	PIC18F4XJ53	---- --00	---- --uu	---- --uu
ODCON3	PIC18F2XJ53	PIC18F4XJ53	---- --00	---- --uu	---- --uu
RTCCFG	PIC18F2XJ53	PIC18F4XJ53	0-00 0000	u-uu uuuu	u-uu uuuu
RTCCAL	PIC18F2XJ53	PIC18F4XJ53	0000 0000	uuuu uuuu	uuuu uuuu

Legend: u = unchanged, x = unknown, - = unimplemented bit, read as '0', q = value depends on condition.

Note 1: When the wake-up is due to an interrupt and the GIEL or GIEH bit is set, the TOSU, TOSH and TOSL are updated with the current value of the PC. The STKPTR is modified to point to the next location in the hardware stack.

2: When the wake-up is due to an interrupt and the GIEL or GIEH bit is set, the PC is loaded with the interrupt vector (0008h or 0018h).

3: One or more bits in the INTCONx or PIRx registers will be affected (to cause wake-up).

4: See Table 5-1 for Reset value for specific condition.

5: Not implemented for PIC18F2XJ53 devices.

6: Not implemented for "LF" devices.

TABLE 6-4: REGISTER FILE SUMMARY (PIC18F47J53 FAMILY) (CONTINUED)

Addr.	File Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR
F7Fh	SPBRGH1	EUSART1 Baud Rate Generator High Byte								0000 0000
F7Eh	BAUDCON1	ABDOVF	RCIDL	RXDTP	TXCKP	BRG16	—	WUE	ABDEN	0100 0-00
F7Dh	SPBRGH2	EUSART2 Baud Rate Generator High Byte								0000 0000
F7Ch	BAUDCON2	ABDOVF	RCIDL	RXDTP	TXCKP	BRG16	—	WUE	ABDEN	0100 0-00
F7Bh	TMR3H	Timer3 Register High Byte								xxxx xxxx
F7Ah	TMR3L	Timer3 Register Low Byte								xxxx xxxx
F79h	T3CON	TMR3CS1	TMR3CS0	T3CKPS1	T3CKPS0	T3OSCEN	$\overline{\text{T3SYNC}}$	RD16	TMR3ON	0000 0000
F78h	TMR4	Timer4 Register								0000 0000
F77h	PR4	Timer4 Period Register								1111 1111
F76h	T4CON	—	T4OUTPS3	T4OUTPS2	T4OUTPS1	T4OUTPS0	TMR4ON	T4CKPS1	T4CKPS0	-000 0000
F75h	SSP2BUF	MSSP2 Receive Buffer/Transmit Register								xxxx xxxx
F74h	SSP2ADD	MSSP2 Address Register (I^2C Slave Mode). MSSP2 Baud Rate Reload Register (I^2C Master Mode).								---- ----
F74h	SSP2MSK	MSK7	MSK6	MSK5	MSK4	MSK3	MSK2	MSK1	MSK0	0000 0000
F73h	SSP2STAT	SMP	CKE	D/$\overline{\text{A}}$	P	S	R/$\overline{\text{W}}$	UA	BF	1111 1111
F72h	SSP2CON1	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000
F71h	SSP2CON2	GCEN	ACKSTAT	ACKDT ADMSK5	ACKEN ADMSK4	RCEN ADMSK3	PEN ADMSK2	RSEN ADMSK1	SEN	0000 0000
F70h	CMSTAT	—	—	—	—	—	COUT3	COUT2	COUT1	---- -111
F6Fh	PMADDRH/ PMDOUT1H(1)	—	CS1	Parallel Master Port Address High Byte						-000 0000
		Parallel Port Out Data High Byte (Buffer 1)								0000 0000
F6Eh	PMADDRL/ PMDOUT1L(1)	Parallel Master Port Address Low Byte/ Parallel Port Out Data Low Byte (Buffer 1)								0000 0000
F6Dh	PMDIN1H(1)	Parallel Port In Data High Byte (Buffer 1)								0000 0000
F6Ch	PMDIN1L(1)	Parallel Port In Data Low Byte (Buffer 1)								0000 0000
F6Bh	TXADDRL	SPI DMA Transmit Data Pointer Low Byte								xxxx xxxx
F6Ah	TXADDRH	—	—	—	—	SPI DMA Transmit Data Pointer High Byte				---- xxxx
F69h	RXADDRL	SPI DMA Receive Data Pointer Low Byte								xxxx xxxx
F68h	RXADDRH	—	—	—	—	SPI DMA Receive Data Pointer High Byte				---- xxxx
F67h	DMABCL	SPI DMA Byte Count Low Byte								xxxx xxxx
F66h	DMABCH	—	—	—	—	—	—	SPI DMA Byte Count High Byte		---- --xx
F65h	UCON(1)	—	PPBRST	SE0	PKTDIS	USBEN	RESUME	SUSPND	—	-0x0 000-
F64h	USTAT	—	ENDP3	ENDP2	ENDP1	ENDP0	DIR	PPBI	—	-xxx xxx-
F63h	UEIR	BTSEF	—	—	BTOEF	DFN8EF	CRC16EF	CRC5EF	PIDEF	0--0 0000
F62h	UIR	—	SOFIF	STALLIF	IDLEIF	TRNIF	ACTVIF	UERRIF	URSTIF	-000 0000
F61h	UFRMH	—	—	—	—	—	FRM10	FRM9	FRM8	---- -xxx
F60h	UFRML	FRM7	FRM6	FRM5	FRM4	FRM3	FRM2	FRM1	FRM0	xxxx xxxx
F5Fh	PMCONH(1)	PMPEN	—	PSIDL	ADRMUX1	ADRMUX0	PTBEEN	PTWREN	PTRDEN	0-00 0000
F5Eh	PMCONL(1)	CSF1	CSF0	ALP	—	CS1P	BEP	WRSP	RDSP	000- 0000
F5Dh	PMMODEH(1)	BUSY	IRQM1	IRQM0	INCM1	INCM0	MODE16	MODE1	MODE0	0000 0000
F5Ch	PMMODEL(1)	WAITB1	WAITB0	WAITM3	WAITM2	WAITM1	WAITM0	WAITE1	WAITE0	0000 0000
F5Bh	PMDOUT2H(1)	Parallel Port Out Data High Byte (Buffer 2)								0000 0000
F5Ah	PMDOUT2L(1)	Parallel Port Out Data Low Byte (Buffer 2)								0000 0000
F59h	PMDIN2H(1)	Parallel Port In Data High Byte (Buffer 2)								0000 0000
F58h	PMDIN2L(1)	Parallel Port In Data Low Byte (Buffer 2)								0000 0000
F57h	PMEH(1)	PTEN15	PTEN14	PTEN13	PTEN12	PTEN11	PTEN10	PTEN9	PTEN8	0000 0000
F56h	PMEL(1)	PTEN7	PTEN6	PTEN5	PTEN4	PTEN3	PTEN2	PTEN1	PTEN0	0000 0000
F55h	PMSTATH(1)	IBF	IBOV	—	—	IB3F	IB2F	IB1F	IB0F	00-- 0000
F54h	PMSTATL(1)	OBE	OBUF	—	—	OB3E	OB2E	OB1E	OB0E	10-- 1111

Legend: x = unknown, u = unchanged, - = unimplemented, q = value depends on condition, r = reserved, do not modify

Note 1: Implemented only for 44-pin devices (PIC18F46J53, PIC18F47J53, PIC18LF46J53 and PIC18LF47J53).

2: Implemented only for 28-pin devices (PIC18F26J53, PIC18F27J53, PIC18LF26J53 and PIC18LF27J53).

3: Implemented only for devices with 128 Kbyte of program memory (PIC18F27J53, PIC18F47J53, PIC18LF27J53 and PIC18LF47J53).

© 2009-2016 Microchip Technology Inc.

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, 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 (ACCESS F9Eh)

R/W-0	R/W-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
PMPIF(1)	ADIF	RC1IF	TX1IF	SSP1IF	CCP1IF	TMR2IF	TMR1IF
bit 7							bit 0

Legend:			
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

bit 7 **PMPIF:** Parallel Master 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

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:** EUSART1 Receive Interrupt Flag bit
1 = The EUSART1 receive buffer, RCREG1, is full (cleared when RCREG1 is read)
0 = The EUSART1 receive buffer is empty

bit 4 **TX1IF:** EUSART1 Transmit Interrupt Flag bit
1 = The EUSART1 transmit buffer, TXREG1, is empty (cleared when TXREG1 is written)
0 = The EUSART1 transmit buffer is full

bit 3 **SSP1IF:** Master Synchronous Serial Port 1 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
Capture mode:
1 = A TMR1/TMR3 register capture occurred (must be cleared in software)
0 = No TMR1/TMR3 register capture occurred
Compare mode:
1 = A TMR1/TMR3 register compare match occurred (must be cleared in software)
0 = No TMR1/TMR3 register compare match occurred
PWM mode:
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

Note 1: These bits are unimplemented on 28-pin devices.

10.3 PORTB, TRISB and LATB Registers

PORTB is an 8-bit wide, bidirectional port. The corresponding Data Direction register is TRISB. Setting a TRISB bit (= 1) will make the corresponding PORTB pin an input (i.e., put the corresponding output driver in a High-Impedance mode). Clearing a TRISB bit (= 0) will make the corresponding PORTB pin an output (i.e., put the contents of the output latch on the selected pin).

The Data Latch register (LATB) is also memory mapped. Read-modify-write operations on the LATB register read and write the latched output value for PORTB.

EXAMPLE 10-3: INITIALIZING PORTB

```
CLRF   PORTB        ; Initialize PORTB by
                    ; clearing output
                    ; data latches
CLRF   LATB         ; Alternate method
                    ; to clear output
                    ; data latches
MOVLW  0x3F         ; Configure as digital I/O
MOVFF  WREG ADCON1  ; pins in this example

MOVLW  0CFh         ; Value used to
                    ; initialize data
                    ; direction
MOVWF  TRISB        ; Set RB<3:0> as inputs
                    ; RB<5:4> as outputs
                    ; RB<7:6> as inputs
```

Each of the PORTB pins has a weak internal pull-up. A single control bit can turn on all the pull-ups. This is performed by clearing bit, $\overline{\text{RBPU}}$ (INTCON2<7>). The weak pull-up is automatically turned off when the port pin is configured as an output. The pull-ups are disabled on a POR. The integrated weak pull-ups consist of a semiconductor structure similar to, but somewhat different from, a discrete resistor. On an unloaded I/O pin, the weak pull-ups are intended to provide logic high indication, but will not necessarily pull the pin all the way to V_{DD} levels.

Note:	On a POR, the RB<3:0> bits are configured as analog inputs by default and read as '0'; RB<7:4> bits are configured as digital inputs.

Four of the PORTB pins (RB<7:4>) have an interrupt-on-change feature. Only pins configured as inputs can cause this interrupt to occur (i.e., any RB<7:4> pin configured as an output is excluded from the interrupt-on-change comparison). The input pins (of RB<7:4>) are compared with the old value latched on the last read of PORTB. The "mismatch" outputs of RB<7:4> are ORed together to generate the RB Port Change Interrupt with Flag bit, RBIF (INTCON<0>).

This interrupt can wake the device from Sleep mode or any of the Idle modes. Application software can clear the interrupt flag by following these steps:

1. Any read or write of PORTB (except with the MOVFF (ANY), PORTB instruction).
2. Wait one instruction cycle (such as executing a NOP instruction).
3. Clear flag bit, RBIF.

A mismatch condition continues to set flag bit, RBIF. Reading PORTB will end the mismatch condition and allow flag bit, RBIF, to be cleared after one instruction cycle of delay.

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.

The RB5 pin is multiplexed with the Timer0 module clock input and one of the comparator outputs to become the RB5/CCP5/KBI1/SDI1/SDA1/RP8 pin.

TABLE 10-11: PORTE I/O SUMMARY

Pin	Function	TRIS Setting	I/O	I/O Type	Description
RE0/AN5/ PMRD	RE0	1	I	ST	PORTE<0> data input; disabled when analog input is enabled.
		0	O	DIG	LATE<0> data output; not affected by analog input.
	AN5	1	I	ANA	A/D Input Channel 5; default input configuration on POR.
	PMRD	1	I	ST/TTL	Parallel Master Port (io_rd_in).
		0	O	DIG	Parallel Master Port read strobe.
RE1/AN6/ PMWR	RE1	1	I	ST	PORTE<1> data input; disabled when analog input is enabled.
		0	O	DIG	LATE<1> data output; not affected by analog input.
	AN6	1	I	ANA	A/D Input Channel 6; default input configuration on POR.
	PMWR	1	I	ST/TTL	Parallel Master Port (io_wr_in).
		0	O	DIG	Parallel Master Port write strobe.
RE2/AN7/ PMCS	RE2	1	I	ST	PORTE<2> data input; disabled when analog input is enabled.
		0	O	DIG	LATE<2> data output; not affected by an analog input.
	AN7	1	I	ANA	A/D Input Channel 7; default input configuration on POR.
	PMCS	0	O	DIG	Parallel Master Port byte enable.
VSS1	—	—	P	—	Ground reference for logic and I/O pins.
VSS2					
AVSS1	—	—	P	—	Ground reference for analog modules.
VDD1	—	—	P	—	Positive supply for peripheral digital logic and I/O pins.
VDD2					
VDDCORE/VCAP	VDDCORE	—	P	—	Positive supply for microcontroller core logic (regulator disabled).
	VCAP	—	P	—	External filter capacitor connection (regulator enabled).
AVDD1	—	—	P	—	Positive supply for analog modules.
AVDD2			—		
VUSB	—	—	P	—	USB voltage input pin.

Legend: DIG = Digital level output; TTL = TTL input buffer; ST = Schmitt Trigger input buffer; ANA = Analog level I = Input; O = Output; P = Power

TABLE 10-12: SUMMARY OF REGISTERS ASSOCIATED WITH PORTE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PORTE(1)	—	—	—	—	—	RE2	RE1	RE0
LATE(1)	—	—	—	—	—	LATE2	LATE1	LATE0
TRISE(1)	RDPU	REPU	—	—	—	TRISE2	TRISE1	TRISE0
ANCON0	PCFG7(1)	PCFG6(1)	PCFG5(1)	PCFG4	PCFG3	PCFG2	PCFG1	PCFG0

Legend: — = unimplemented, read as '0'. Shaded cells are not used by PORTE.

Note 1: These registers and/or bits are not available in 28-pin devices (PIC18F26J53, PIC18F27J53, PIC18LF26J53 and PIC18LF26J53).

Note: bit 7 **RDPU:** PORTD Pull-up Enable bit
0 = All PORTD pull-ups are disabled
1 = PORTD pull-ups are enabled for any input pad
bit 6 **REPU:** PORTE Pull-up Enable bit
0 = All PORTE pull-ups are disabled
1 = PORTE pull-ups are enabled for any input pad

© 2009-2016 Microchip Technology Inc.

REGISTER 11-3: PMMODEH: PARALLEL PORT MODE REGISTER HIGH BYTE (BANKED F5Dh)(1)

R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
BUSY	IRQM1	IRQM0	INCM1	INCM0	MODE16	MODE1	MODE0
bit 7							bit 0

Legend:			
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

bit 7 **BUSY:** Busy bit (Master mode only)
1 = Port is busy
0 = Port is not busy

bit 6-5 **IRQM<1:0>:** Interrupt Request Mode bits
11 = Interrupt generated when Read Buffer 3 is read or Write Buffer 3 is written (Buffered PSP mode) or on a read or write operation when PMA<1:0> = 11 (Addressable PSP mode only)
10 = No interrupt generated, processor stall activated
01 = Interrupt generated at the end of the read/write cycle
00 = No interrupt generated

bit 4-3 **INCM<1:0>:** Increment Mode bits
11 = PSP read and write buffers auto-increment (Legacy PSP mode only)
10 = Decrement ADDR<15,13:0> by 1 every read/write cycle
01 = Increment ADDR<15,13:0> by 1 every read/write cycle
00 = No increment or decrement of address

bit 2 **MODE16:** 8/16-Bit Mode bit
1 = 16-bit mode: Data register is 16 bits; a read or write to the Data register invokes two 8-bit transfers
0 = 8-bit mode: Data register is 8 bits; a read or write to the Data register invokes one 8-bit transfer

bit 1-0 **MODE<1:0>:** Parallel Port Mode Select bits
11 = Master Mode 1 (PMCSx, PMRD/PMWR, PMENB, PMBE, PMA<x:0> and PMD<7:0>)
10 = Master Mode 2 (PMCSx, PMRD, PMWR, PMBE, PMA<x:0> and PMD<7:0>)
01 = Enhanced PSP, control signals (PMRD, PMWR, PMCSx, PMD<7:0> and PMA<1:0>)
00 = Legacy Parallel Slave Port, control signals (PMRD, PMWR, PMCSx and PMD<7:0>)

Note 1: This register is only available on 44-pin devices.

FIGURE 11-24: WRITE TIMING, 16-BIT MULTIPLEXED DATA, PARTIALLY MULTIPLEXED ADDRESS

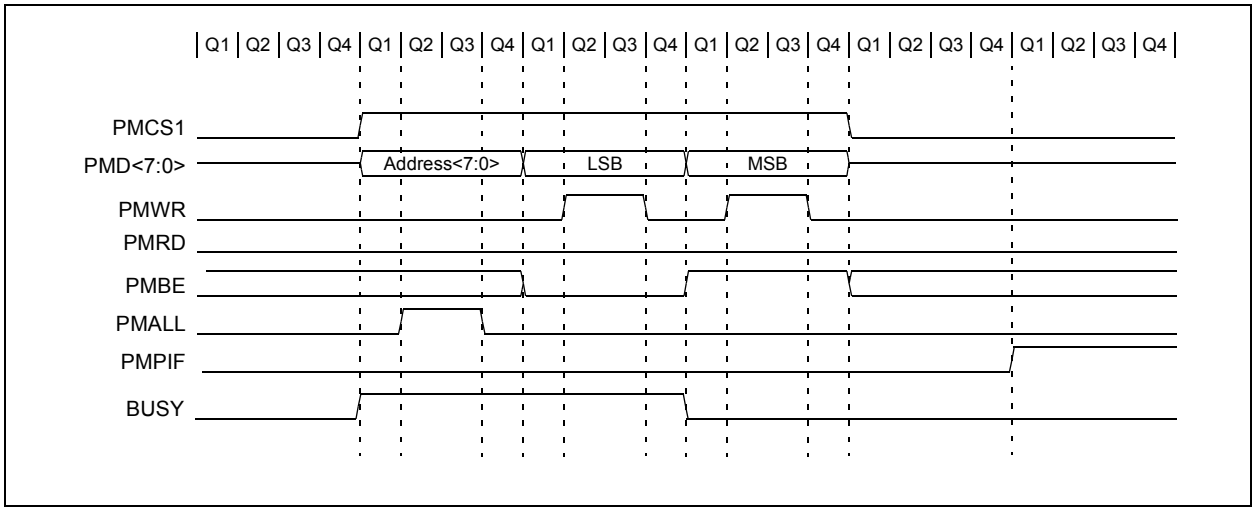


FIGURE 11-25: READ TIMING, 16-BIT MULTIPLEXED DATA, FULLY MULTIPLEXED 16-BIT ADDRESS

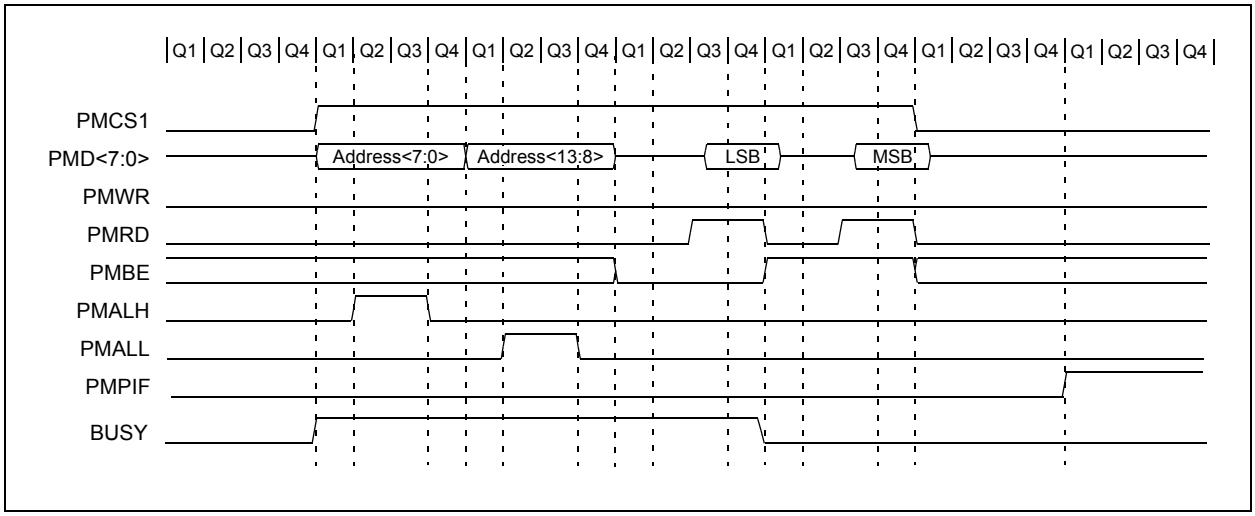


FIGURE 11-26: WRITE TIMING, 16-BIT MULTIPLEXED DATA, FULLY MULTIPLEXED 16-BIT ADDRESS

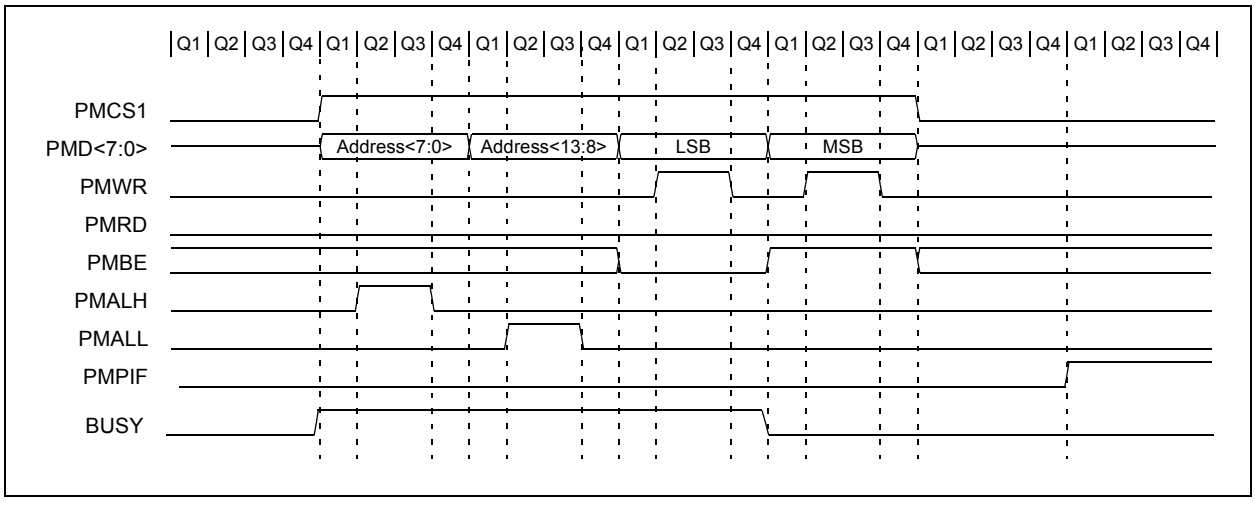


FIGURE 13-7: TIMER1 GATE SINGLE PULSE AND TOGGLE COMBINED MODE

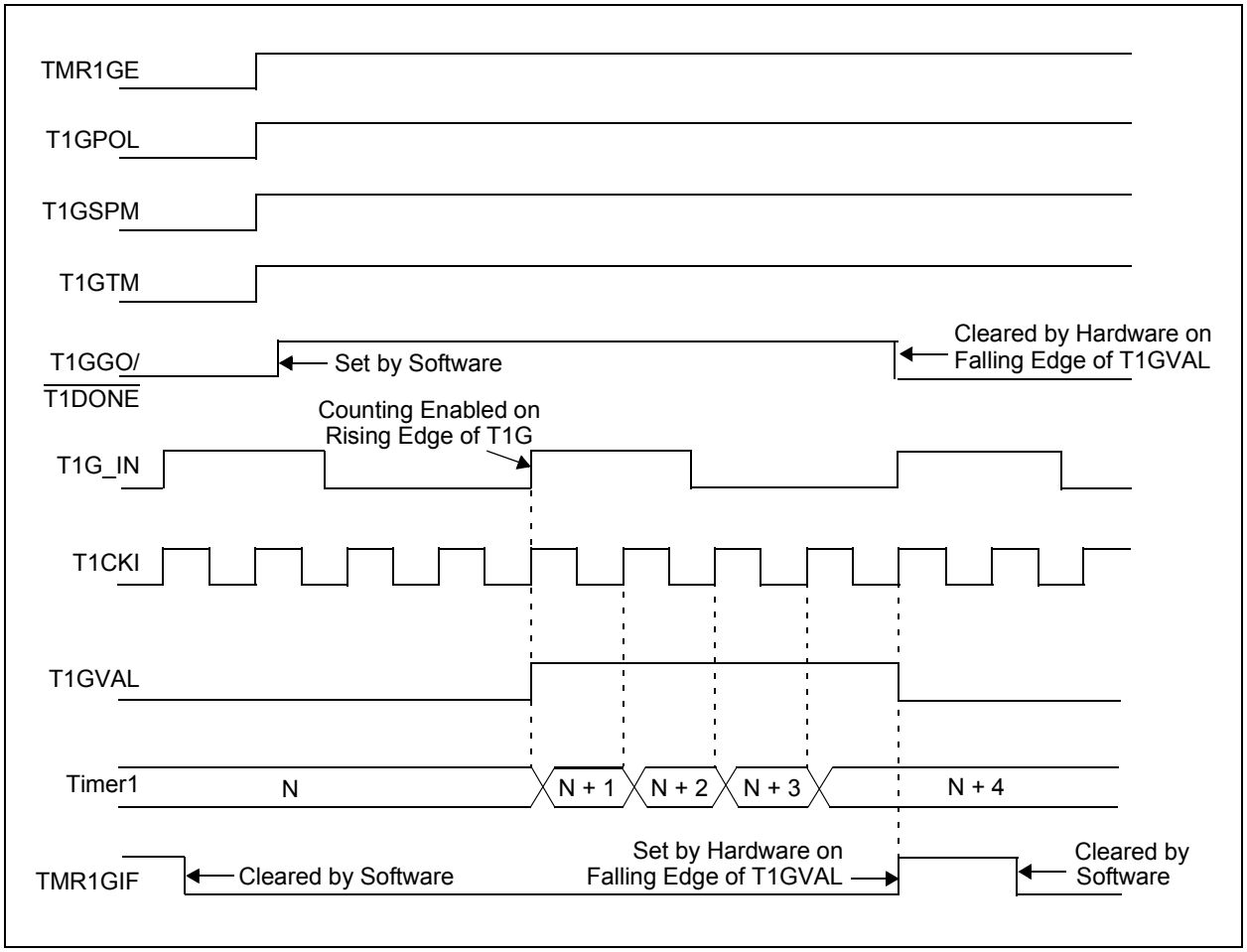


TABLE 13-5: REGISTERS ASSOCIATED WITH TIMER1 AS A TIMER/COUNTER

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF
PIR1	PMPIF(1)	ADIF	RC1IF	TX1IF	SSP1IF	CCP1IF	TMR2IF	TMR1IF
PIE1	PMPIE(1)	ADIE	RC1IE	TX1IE	SSP1IE	CCP1IE	TMR2IE	TMR1IE
IPR1	PMPIP(1)	ADIP	RC1IP	TX1IP	SSP1IP	CCP1IP	TMR2IP	TMR1IP
TMR1L	Timer1 Register Low Byte							
TMR1H	Timer1 Register High Byte							
T1CON	TMR1CS1	TMR1CS0	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{\text{T1SYNC}}$	RD16	TMR1ON
T1GCON	TMR1GE	T1GPOL	T1GTM	T1GSPM	$\overline{\text{T1GGO}}$/$\overline{\text{T1DONE}}$	T1GVAL	T1GSS1	T1GSS0
OSCCON2	—	SOSCRUN	—	SOSCDRV	SOSCGO	PRISD	—	—

Legend: — = unimplemented, read as '0'. Shaded cells are not used by the Timer1 module.

Note 1: These bits are only available in 44-pin devices.

© 2009-2016 Microchip Technology Inc.

19.4 PWM (Enhanced Mode)

The Enhanced PWM mode can generate a PWM signal on up to four different output pins with up to 10 bits of resolution. It can do this through four different PWM Output modes:

- Single PWM mode
- Half-Bridge PWM mode
- Full-Bridge PWM, Forward mode
- Full-Bridge PWM, Reverse mode

To select an Enhanced PWM mode, the PxM bits of the CCPxCON register must be set appropriately.

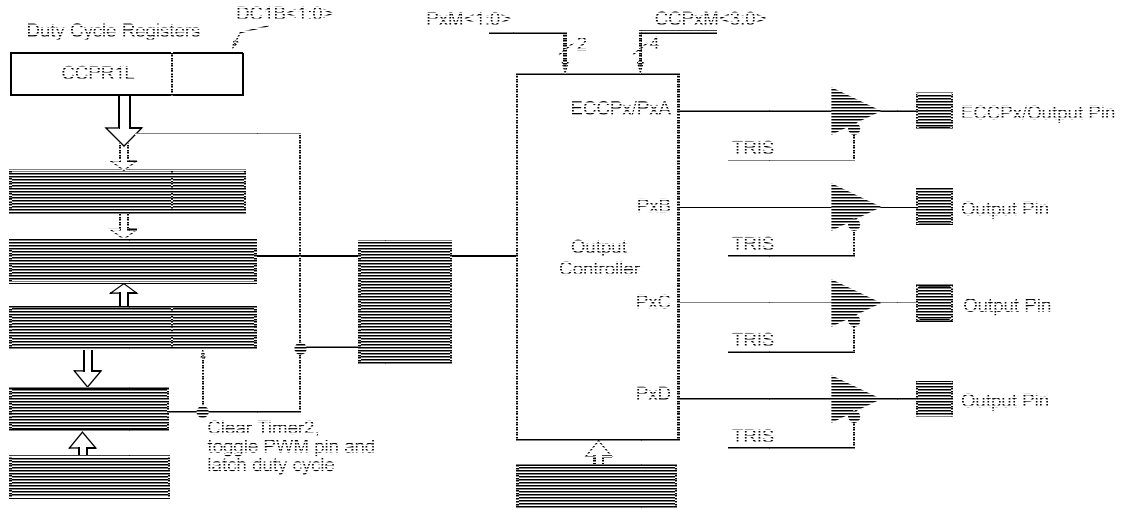
The PWM outputs are multiplexed with I/O pins and are designated: PxA, PxB, PxC and PxD. The polarity of the PWM pins is configurable and is selected by setting the CCPxM bits in the CCPxCON register appropriately.

Table 19-1 provides the pin assignments for each Enhanced PWM mode.

Figure 19-3 provides an example of a simplified block diagram of the Enhanced PWM module.

Note:	To prevent the generation of an incomplete waveform when the PWM is first enabled, the ECCP module waits until the start of a new PWM period before generating a PWM signal.

FIGURE 19-3: SIMPLIFIED BLOCK DIAGRAM OF THE ENHANCED PWM MODE EXAMPLE



Note 1: The 8-bit timer TMR2 register is concatenated with the 2-bit internal Q clock, or 2 bits of the prescaler, to create the 10-bit time base.

Note 1:	The TRIS register value for each PWM output must be configured appropriately.
2:	Any pin not used by an Enhanced PWM mode is available for alternate pin functions.

20.5.14 SLEEP OPERATION

While in Sleep mode, the I^2C 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).

20.5.15 EFFECTS OF A RESET

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

20.5.16 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 Start and Stop bits are cleared from a Reset or when the MSSP module is disabled. Control of the I^2C bus may be taken when the P bit (SSPxSTAT<4>) is set, or the bus is Idle, with both the Start and Stop 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 SDAx 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 BCLxIF bit.

The states where arbitration can be lost are:

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

20.5.17 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 SDAx pin, arbitration takes place when the master outputs a '1' on SDAx, by letting SDAx float high, and another master asserts a '0'. When the SCLx pin floats high, data should be stable. If the expected data on SDAx is a '1' and the data sampled on the SDAx pin = 0, then a bus collision has taken place. The master will set the Bus Collision Interrupt Flag, BCLxIF, and reset the I^2C port to its Idle state (Figure 20-27).

If a transmit was in progress when the bus collision occurred, the transmission is Halted, the BF flag is cleared, the SDAx and SCLx lines are deasserted and the SSPxBUF 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.

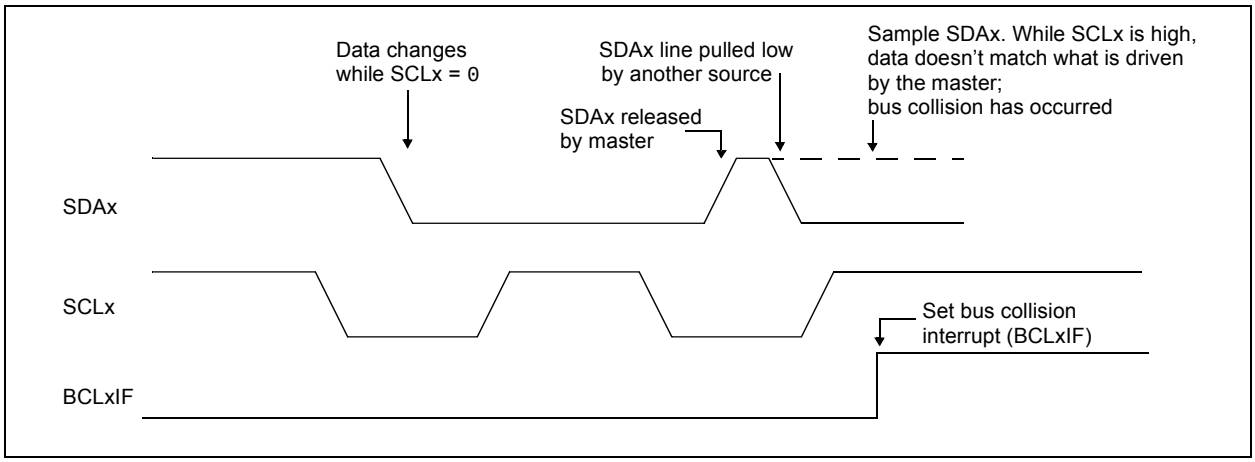
If a Start, Repeated Start, Stop or Acknowledge condition was in progress when the bus collision occurred, the condition is aborted, the SDAx and SCLx lines are deasserted and the respective control bits in the SSPxCON2 register are cleared. When the user services the bus collision Interrupt Service Routine (ISR), 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 SDAx and SCLx pins. If a Stop condition occurs, the SSPxIF bit will be set.

A write to the SSPxBUF 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 Stop bit is set in the SSPxSTAT register, or the bus is Idle and the Start and Stop bits are cleared.

FIGURE 20-27: BUS COLLISION TIMING FOR TRANSMIT AND ACKNOWLEDGE



21.1.3 AUTO-BAUD RATE DETECT

The Enhanced USART module supports the automatic detection and calibration of baud rate. This feature is active only in Asynchronous mode and while the WUE bit is clear.

The automatic baud rate measurement sequence (Figure 21-1) begins whenever a Start bit is received and the ABDEN bit is set. The calculation is self-averaging.

In the Auto-Baud Rate Detect (ABD) mode, the clock to the BRG is reversed. Rather than the BRG clocking the incoming RXx signal, the RXx signal is timing the BRG. In ABD mode, the internal BRG is used as a counter to time the bit period of the incoming serial byte stream.

Once the ABDEN bit is set, the state machine will clear the BRG and look for a Start bit. The ABD must receive a byte with the value, 55h (ASCII “U”, which is also the LIN/J2602 bus Sync character), in order to calculate the proper bit rate. The measurement is taken over both a low and high bit time in order to minimize any effects caused by asymmetry of the incoming signal. After a Start bit, the SPBRGx begins counting up, using the pre-selected clock source on the first rising edge of RXx. After eight bits on the RXx pin, or the fifth rising edge, an accumulated value totalling the proper BRG period is left in the SPBRGHx:SPBRGx register pair. Once the 5th edge is seen (this should correspond to the Stop bit), the ABDEN bit is automatically cleared.

If a rollover of the BRG occurs (an overflow from FFFFh to 0000h), the event is trapped by the ABDOVF status bit (BAUDCONx<7>). It is set in hardware by BRG roll-overs and can be set or cleared by the user in software. ABD mode remains active after rollover events and the ABDEN bit remains set (Figure 21-2).

While calibrating the baud rate period, the BRG registers are clocked at 1/8th the preconfigured clock rate. Note that the BRG clock will be configured by the BRG16 and BRGH bits. Independent of the BRG16 bit setting, both the SPBRGx and SPBRGHx will be used as a 16-bit counter. This allows the user to verify that no carry occurred for 8-bit modes by checking for 00h in the SPBRGHx register.

Refer to Table 21-4 for counter clock rates to the BRG.

While the ABD sequence takes place, the EUSART state machine is held in Idle. The RCxIF interrupt is set once the fifth rising edge on RXx is detected. The value in the RCREGx needs to be read to clear the RCxIF interrupt. The contents of RCREGx should be discarded.

Note 1: If the WUE bit is set with the ABDEN bit, Auto-Baud Rate Detection will occur on the byte *following* the Break character.

2: It is up to the user to determine that the incoming character baud rate is within the range of the selected BRG clock source. Some combinations of oscillator frequency and EUSART baud rates are not possible due to bit error rates. Overall system timing and communication baud rates must be taken into consideration when using the Auto-Baud Rate Detection feature.

TABLE 21-4: BRG COUNTER CLOCK RATES

BRG16	BRGH	BRG Counter Clock
0	0	FOSC/512
0	1	FOSC/128
1	0	FOSC/128
1	1	FOSC/32

Note: During the ABD sequence, SPBRGx and SPBRGHx are both used as a 16-bit counter, independent of the BRG16 setting.

21.1.3.1 ABD and EUSART Transmission

Since the BRG clock is reversed during ABD acquisition, the EUSART transmitter cannot be used during ABD. This means that whenever the ABDEN bit is set, TXREGx cannot be written to. Users should also ensure that ABDEN does not become set during a transmit sequence. Failing to do this may result in unpredictable EUSART operation.

21.2.2 EUSART ASYNCHRONOUS RECEIVER

The receiver block diagram is displayed in Figure 21-6. The data is received on the RXx pin and drives the data recovery block. The data recovery block is actually a high-speed shifter, operating at x16 times the baud rate, whereas the main receive serial shifter operates at the bit rate or at FOSC. This mode would typically be used in RS-232 systems.

To set up an Asynchronous Reception:

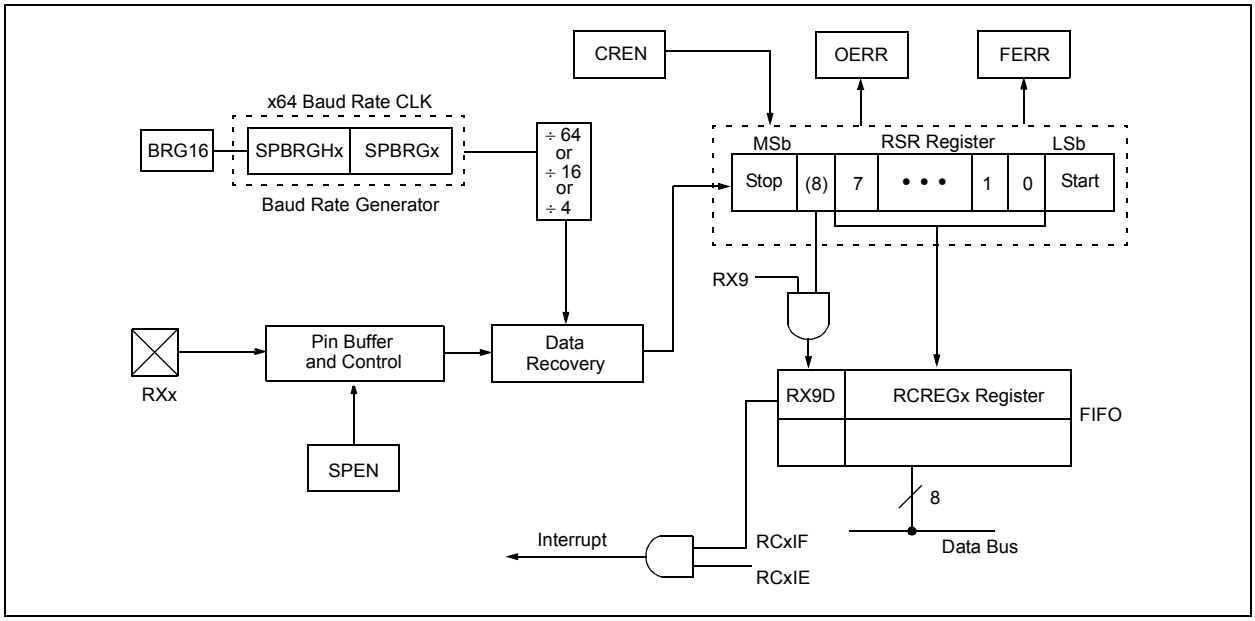
1. Initialize the SPBRGHx:SPBRGx registers for the appropriate baud rate. Set or clear the BRGH and BRG16 bits, as required, to achieve the desired baud rate.
2. Enable the asynchronous serial port by clearing bit, SYNC, and setting bit, SPEN.
3. If interrupts are desired, set enable bit, RCxIE.
4. If 9-bit reception is desired, set bit, RX9.
5. Enable the reception by setting bit, CREN.
6. Flag bit, RCxIF, will be set when reception is complete and an interrupt will be generated if enable bit, RCxIE, was set.
7. Read the RCSTAx register to get the ninth bit (if enabled) and determine if any error occurred during reception.
8. Read the 8-bit received data by reading the RCREGx register.
9. If any error occurred, clear the error by clearing enable bit, CREN.
10. If using interrupts, ensure that the GIE and PEIE bits in the INTCON register (INTCON<7:6>) are set.

21.2.3 SETTING UP 9-BIT MODE WITH ADDRESS DETECT

This mode would typically be used in RS-485 systems. To set up an Asynchronous Reception with Address Detect Enable:

1. Initialize the SPBRGHx:SPBRGx registers for the appropriate baud rate. Set or clear the BRGH and BRG16 bits, as required, to achieve the desired baud rate.
2. Enable the asynchronous serial port by clearing the SYNC bit and setting the SPEN bit.
3. If interrupts are required, set the RCEN bit and select the desired priority level with the RCxIP bit.
4. Set the RX9 bit to enable 9-bit reception.
5. Set the ADDEN bit to enable address detect.
6. Enable reception by setting the CREN bit.
7. The RCxIF bit will be set when reception is complete. The interrupt will be Acknowledged if the RCxIE and GIE bits are set.
8. Read the RCSTAx register to determine if any error occurred during reception, as well as read bit 9 of data (if applicable).
9. Read RCREGx to determine if the device is being addressed.
10. If any error occurred, clear the CREN bit.
11. If the device has been addressed, clear the ADDEN bit to allow all received data into the receive buffer and interrupt the CPU.

FIGURE 21-6: EUSARTx RECEIVE BLOCK DIAGRAM



22.1 A/D Acquisition Requirements

For the A/D Converter to meet its specified accuracy, the charge holding capacitor (CHOLD) must be allowed to fully charge to the input channel voltage level. The analog input model is illustrated in Figure 22-2. The source impedance (RS) and the internal sampling switch (RSS) impedance directly affect the time required to charge the capacitor, CHOLD. The sampling switch (RSS) impedance varies over the device voltage (VDD). The source impedance affects the offset voltage at the analog input (due to pin leakage current). **The maximum recommended impedance for analog sources is 2.5 kΩ for 10-bit conversions and 1 kΩ for 12-bit conversions**. After the analog input channel is selected (changed), the channel must be sampled for at least the minimum acquisition time before starting a conversion.

Note:	When the conversion is started, the holding capacitor is disconnected from the input pin.

To calculate the minimum acquisition time, Equation 22-1 may be used. This equation assumes that 1/2 LSb error is used (1,024 steps for the 10-bit A/D). The 1/2 LSb error is the maximum error allowed for the A/D to meet its specified resolution.

Equation 22-3 provides the calculation of the minimum required acquisition time, TACQ. This calculation is based on the following application system assumptions:

CHOLD	=	25 pF
Rs	=	2.5 kΩ
Conversion Error	≤	1/2 LSb
VDD	=	3V → Rss = 2 kΩ
Temperature	=	85°C (system max.)

EQUATION 22-1: ACQUISITION TIME

$$T_{ACQ} = \text{Amplifier Settling Time} + \text{Holding Capacitor Charging Time} + \text{Temperature Coefficient}$$
$$= T_{AMP} + T_C + T_{COFF}$$

EQUATION 22-2: A/D MINIMUM CHARGING TIME

$$V_{HOLD} = (V_{REF} - (V_{REF}/2048)) \bullet (1 - e^{(-T_C/C_{HOLD}(R_{IC} + R_{SS} + R_S))})$$
or
$$T_C = -(C_{HOLD})(R_{IC} + R_{SS} + R_S)\ln(1/2048)$$

EQUATION 22-3: CALCULATING THE MINIMUM REQUIRED ACQUISITION TIME

$$T_{ACQ} = T_{AMP} + T_C + T_{COFF}$$
$$T_{AMP} = 0.2\ \mu s$$
$$T_{COFF} = (\text{Temp} - 25°C)(0.02\ \mu s/°C)$$
$$(85°C - 25°C)(0.02\ \mu s/°C)$$
$$1.2\ \mu s$$

Temperature coefficient is only required for temperatures > 25°C. Below 25°C, $T_{COFF} = 0\ \mu s$.

$$T_C = -(C_{HOLD})(R_{IC} + R_{SS} + R_S)\ln(1/2048)\ \mu s$$
$$-(25\ pF)(1\ k\Omega + 2\ k\Omega + 2.5\ k\Omega)\ln(0.0004883)\ \mu s$$
$$1.05\ \mu s$$
$$T_{ACQ} = 0.2\ \mu s + 1.05\ \mu s + 1.2\ \mu s$$
$$2.45\ \mu s$$

© 2009-2016 Microchip Technology Inc.

REGISTER 23-2: UCFG: USB CONFIGURATION REGISTER (BANKED F39h)

R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
UTEYE	UOEMON	—	UPUEN(1,2)	UTRDIS(1,3)	FSEN(1)	PPB1	PPB0
bit 7							bit 0

Legend:			
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

bit 7 **UTEYE:** USB Eye Pattern Test Enable bit
1 = Eye pattern test is enabled
0 = Eye pattern test is disabled

bit 6 **UOEMON:** USB OE Monitor Enable bit
1 = $\overline{\text{UOE}}$ signal active, indicating intervals during which the D+/D- lines are driving
0 = $\overline{\text{UOE}}$ signal inactive

bit 5 **Unimplemented:** Read as '0'

bit 4 **UPUEN:** USB On-Chip Pull-up Enable bit(1,2)
1 = On-chip pull-up is enabled (pull-up on D+ with FSEN = 1 or D- with FSEN = 0)
0 = On-chip pull-up is disabled

bit 3 **UTRDIS:** On-Chip Transceiver Disable bit(1,3)
1 = On-chip transceiver is disabled
0 = On-chip transceiver is active

bit 2 **FSEN:** Full-Speed Enable bit(1)
1 = Full-speed device: controls transceiver edge rates; requires input clock at 48 MHz
0 = Low-speed device: controls transceiver edge rates; requires input clock at 6 MHz

bit 1-0 **PPB<1:0>:** Ping-Pong Buffers Configuration bits
11 = Even/Odd ping-pong buffers are enabled for Endpoints 1 to 15
10 = Even/Odd ping-pong buffers are enabled for all endpoints
01 = Even/Odd ping-pong buffer are enabled for OUT Endpoint 0
00 = Even/Odd ping-pong buffers are disabled

Note 1: The UPUEN, UTRDIS and FSEN bits should never be changed while the USB module is enabled. These values must be preconfigured prior to enabling the module.
2: This bit is only valid when the on-chip transceiver is active (UTRDIS = 0); otherwise, it is ignored.
3: If UTRDIS is set, the $\overline{\text{UOE}}$ signal will be active – independent of the UOEMON bit setting.

23.5.2 USB INTERRUPT ENABLE REGISTER (UIE)

The USB Interrupt Enable (UIE) register (Register 23-8) contains the enable bits for the USB status interrupt sources. Setting any of these bits will enable the respective interrupt source in the UIR register.

The values in this register only affect the propagation of an interrupt condition to the microcontroller's interrupt logic. The flag bits are still set by their interrupt conditions, allowing them to be polled and serviced without actually generating an interrupt.

REGISTER 23-8: UIE: USB INTERRUPT ENABLE REGISTER (BANKED F36h)

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	SOFIE	STALLIE	IDLEIE	TRNIE	ACTVIE	UERRIE	URSTIE
bit 7							bit 0

Legend:			
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

bit 7 **Unimplemented:** Read as '0'

bit 6 **SOFIE:** Start-Of-Frame Token Interrupt Enable bit
1 = Start-Of-Frame token interrupt is enabled
0 = Start-Of-Frame token interrupt is disabled

bit 5 **STALLIE:** STALL Handshake Interrupt Enable bit
1 = STALL interrupt is enabled
0 = STALL interrupt is disabled

bit 4 **IDLEIE:** Idle Detect Interrupt Enable bit
1 = Idle detect interrupt is enabled
0 = Idle detect interrupt is disabled

bit 3 **TRNIE:** Transaction Complete Interrupt Enable bit
1 = Transaction interrupt is enabled
0 = Transaction interrupt is disabled

bit 2 **ACTVIE:** Bus Activity Detect Interrupt Enable bit
1 = Bus activity detect interrupt is enabled
0 = Bus activity detect interrupt is disabled

bit 1 **UERRIE:** USB Error Interrupt Enable bit
1 = USB error interrupt is enabled
0 = USB error interrupt is disabled

bit 0 **URSTIE:** USB Reset Interrupt Enable bit
1 = USB Reset interrupt is enabled
0 = USB Reset interrupt is disabled

23.6.4 USB TRANSCEIVER CURRENT CONSUMPTION

The USB transceiver consumes a variable amount of current depending on the characteristic impedance of the USB cable, the length of the cable, the VUSB supply voltage and the actual data patterns moving across the USB cable. Longer cables have larger capacitances and consume more total energy when switching output states.

Data patterns that consist of “IN” traffic consume far more current than “OUT” traffic. IN traffic requires the PIC® MCU to drive the USB cable, whereas OUT traffic requires that the host drive the USB cable.

The data that is sent across the USB cable is NRZI encoded. In the NRZI encoding scheme, ‘0’ bits cause a toggling of the output state of the transceiver (either from a “J” state to a “K” state or vise versa). With the exception of the effects of bit stuffing, NRZI encoded ‘1’ bits do not cause the output state of the transceiver to change. Therefore, IN traffic consisting of data bits of value, ‘0’, cause the most current consumption, as the transceiver must charge/discharge the USB cable in order to change states.

More details about NRZI encoding and bit stuffing can be found in the USB 2.0 Specification’s Section 7.1, although knowledge of such details is not required to make USB applications using the PIC18F47J53 family of microcontrollers. Among other things, the SIE handles bit stuffing/unstuffing, NRZI encoding/decoding and CRC generation/checking in hardware.

The total transceiver current consumption will be application-specific. However, to help estimate how much current actually may be required in full-speed applications, Equation 23-1 can be used.

See Equation 23-2 to know how this equation can be used for a theoretical application.

EQUATION 23-1: ESTIMATING USB TRANSCEIVER CURRENT CONSUMPTION

$$I_{XCVR} = \frac{(40\ mA \bullet V_{USB} \bullet P_{ZERO} \bullet P_{IN} \bullet L_{CABLE})}{(3.3V \bullet 5m)} + I_{PULLUP}$$

Legend: VUSB – Voltage applied to the VUSB pin in volts (should be 3.0V to 3.6V).

PZERO – Percentage (in decimal) of the IN traffic bits sent by the PIC® MCU that are a value of ‘0’.

PIN – Percentage (in decimal) of total bus bandwidth that is used for IN traffic.

LCABLE – Length (in meters) of the USB cable. The USB 2.0 Specification requires that full-speed applications use cables no longer than 5m.

IPULLUP – Current which the nominal, 1.5 kΩ pull-up resistor (when enabled) must supply to the USB cable. On the host or hub end of the USB cable, 15 kΩ nominal resistors (14.25 kΩ to 24.8 kΩ) are present which pull both the D+ and D- lines to ground. During bus Idle conditions (such as between packets or during USB Suspend mode), this results in up to 218 μA of quiescent current drawn at 3.3V.

IPULLUP is also dependant on bus traffic conditions and can be as high as 2.2 mA when the USB bandwidth is fully utilized (either IN or OUT traffic) for data that drives the lines to the “K” state most of the time.

BZ **Branch if Zero**

Syntax: BZ n

Operands: -128 ≤ n ≤ 127

Operation: if Zero bit is '1',
(PC) + 2 + 2n → PC

Status Affected: None

Encoding:

1110	0000	nnnn	nnnn

Description: If the Zero bit is '1', then the program will branch.

The 2's complement number '2n' is added to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be PC + 2 + 2n. This instruction is then a 2-cycle instruction.

Words: 1

Cycles: 1(2)

Q Cycle Activity:

If Jump:

Q1	Q2	Q3	Q4
Decode	Read literal 'n'	Process Data	Write to PC
No operation	No operation	No operation	No operation

If No Jump:

Q1	Q2	Q3	Q4
Decode	Read literal 'n'	Process Data	No operation

Example: HERE BZ Jump

Before Instruction
PC = address (HERE)
After Instruction
If Zero = 1;
PC = address (Jump)
If Zero = 0;
PC = address (HERE + 2)

CALL **Subroutine Call**

Syntax: CALL k {,s}

Operands: 0 ≤ k ≤ 1048575
s ∈ [0,1]

Operation: (PC) + 4 → TOS,
k → PC<20:1>;
if s = 1
(W) → WS,
(STATUS) → STATUSS,
(BSR) → BSRS

Status Affected: None

Encoding:

1st word (k<7:0>)	1110	110s	$k_7$kkk	kkkk$_0$
2nd word(k<19:8>)	1111	k_{19}kkk	kkkk	kkkk$_8$

Description: Subroutine call of entire 2-Mbyte memory range. First, return address (PC+ 4) is pushed onto the return stack. If 's' = 1, the W, STATUS and BSR registers are also pushed into their respective shadow registers, WS, STATUSS and BSRS. If 's' = 0, no update occurs (default). Then, the 20-bit value 'k' is loaded into PC<20:1>. CALL is a 2-cycle instruction.

Words: 2

Cycles: 2

Q Cycle Activity:

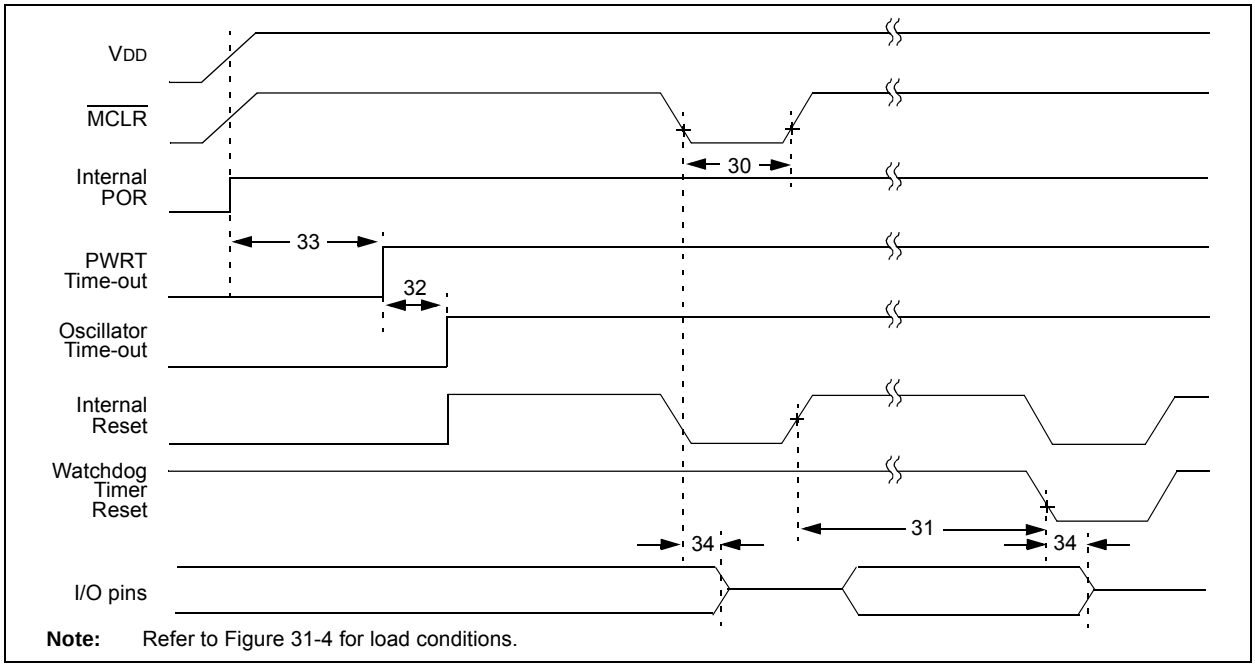
Q1	Q2	Q3	Q4
Decode	Read literal 'k'<7:0>,	Push PC to stack	Read literal 'k'<19:8>, Write to PC
No operation	No operation	No operation	No operation

Example: HERE CALL THERE,1

Before Instruction
PC = address (HERE)
After Instruction
PC = address (THERE)
TOS = address (HERE + 4)
WS = W
BSRS = BSR
STATUSS = STATUS

© 2009-2016 Microchip Technology Inc.

FIGURE 31-7: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING



Note: Refer to Figure 31-4 for load conditions.

TABLE 31-14: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER, POWER-UP TIMER AND BROWN-OUT RESET REQUIREMENTS

Param. No.	Symbol	Characteristic	Min.	Typ.	Max.	Units	Conditions
30	TMCL	$\overline{\text{MCLR}}$ Pulse Width (low)	2	—	—	μs	
31	TWDT	Watchdog Timer Time-out Period (no postscaler)	2.67	4.0	5.53	ms	
32	TOST	Oscillator Start-up Timer Period	1024 TOSC	—	1024 TOSC	—	TOSC = OSC1 period
33	TPWRT	Power-up Timer Period	—	1.0	—	ms	
34	TIOZ	I/O High-Impedance from $\overline{\text{MCLR}}$ Low or Watchdog Timer Reset	—	—	3 TCY + 2	μs	**(Note 1)**
36	TIRVST	Time for Internal Reference Voltage to become Stable	—	20	—	μs	
37	TLVD	High/Low-Voltage Detect Pulse Width	—	200	—	μs	
38	TCSD	CPU Start-up Time	—	200	—	μs	**(Note 2)**

Note 1: The maximum TIOZ is the lesser of (3 TCY + 2 μs) or 700 μs.

2: $\overline{\text{MCLR}}$ rising edge to code execution, assuming TPWRT (and TOST if applicable) has already expired.