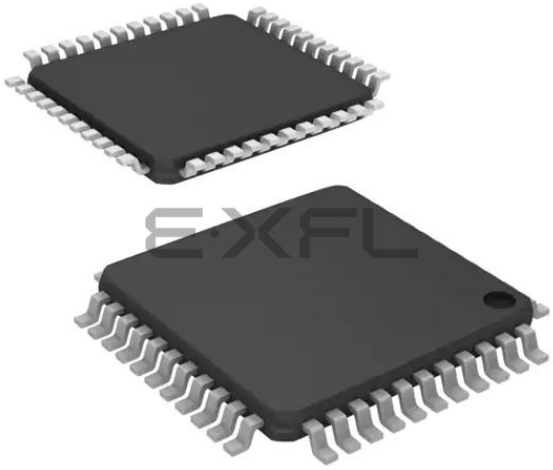


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

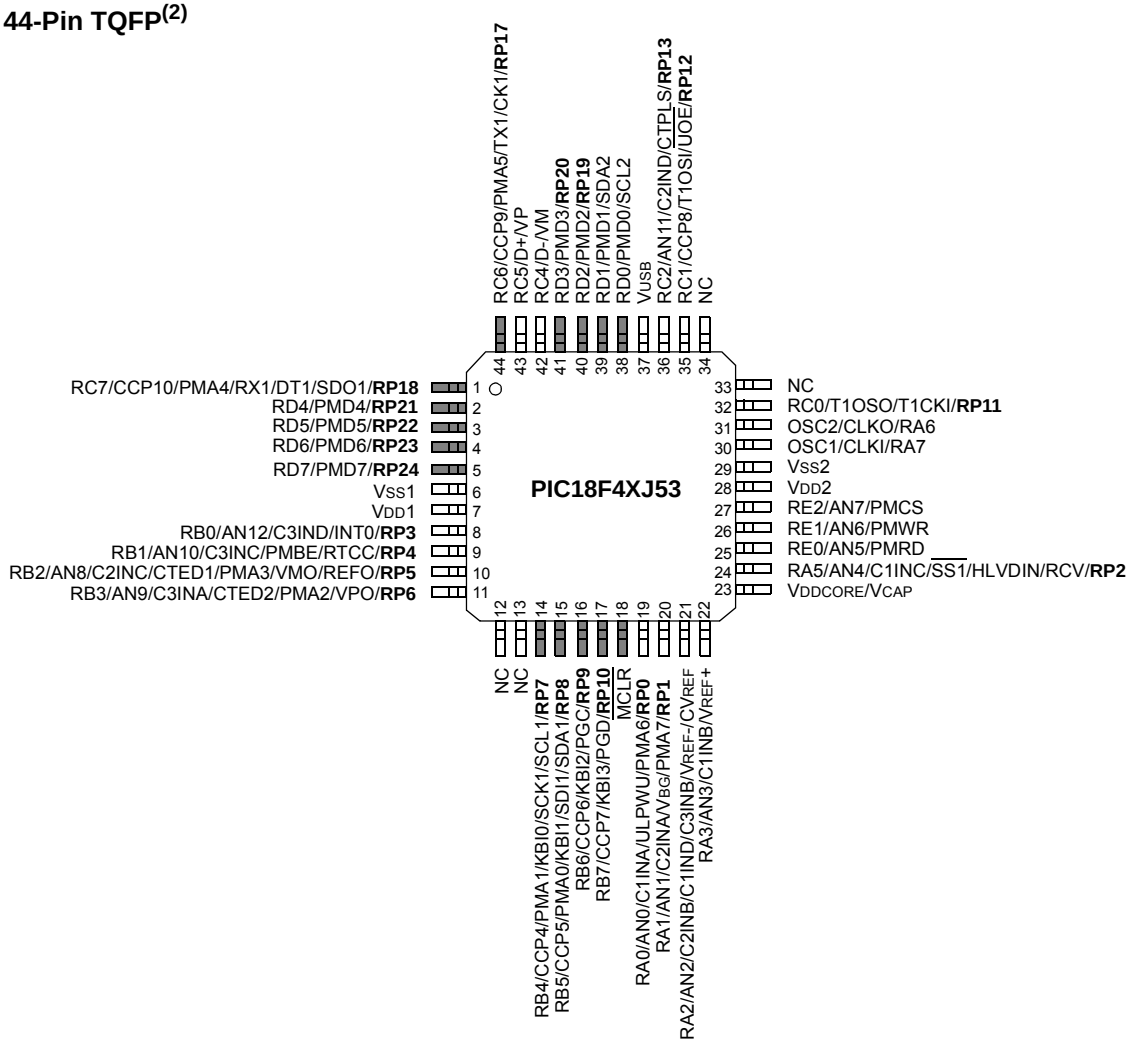
"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" 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)	2.15V ~ 3.6V
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/pic18f47j53t-i-pt

Pin Diagrams (Continued)

44-Pin TQFP(2)



Legend: **RPn** represents remappable pins.
Shaded pins are 5.5V tolerant.

Note: Dedicated AVDD/AVSS pins are available only on the 44-pin QFN package. Other packages internally tie AVDD/AVSS to VDD/VSS.

4.2.3 RC_RUN MODE

In RC_RUN mode, the CPU and peripherals are clocked from the internal oscillator; the primary clock is shut down. This mode provides the best power conservation of all the Run modes while still executing code. It works well for user applications, which are not highly timing-sensitive or do not require high-speed clocks at all times.

This mode is entered by setting the SCS<1:0> bits (OSCCON<1:0>) to '11'. When the clock source is switched to the internal oscillator block (see Figure 4-3), the primary oscillator is shut down and the OSTS bit is cleared.

On transitions from RC_RUN mode to PRI_RUN mode, the device continues to be clocked from the INTOSC block while the primary clock is started. When the primary clock becomes ready, a clock switch to the primary clock occurs (see Figure 4-4). When the clock switch is complete, the OSTS bit is set and the primary clock is providing the device clock. The IDLEN and SCS bits are not affected by the switch. The INTRC clock source will continue to run if either the WDT or the FSCM is enabled.

FIGURE 4-3: TRANSITION TIMING TO RC_RUN MODE

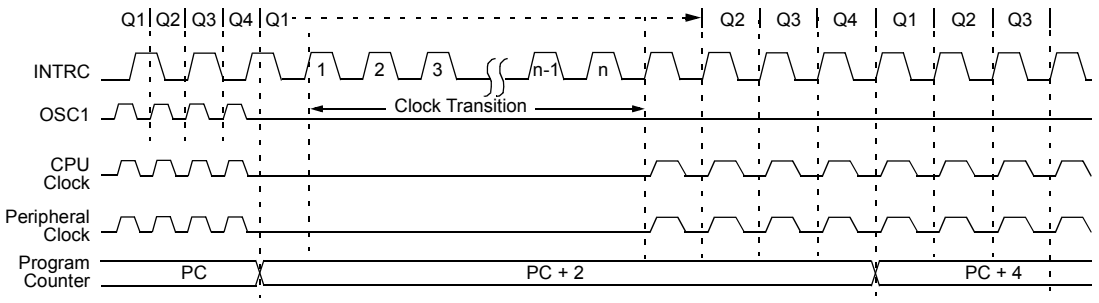
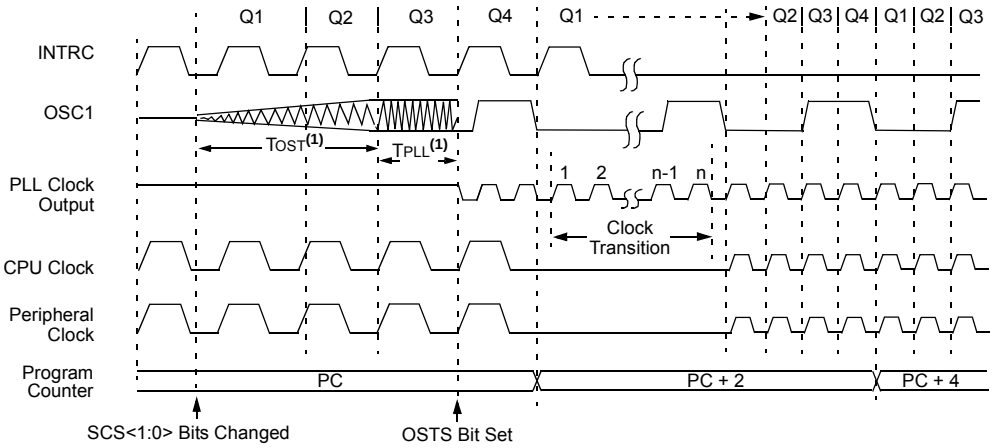


FIGURE 4-4: TRANSITION TIMING FROM RC_RUN MODE TO PRI_RUN MODE



Note 1: TOST = 1024 TOSC; TPLL = 2 ms (approx). These intervals are not shown to scale.

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
TMR8	PIC18F2XJ53	PIC18F4XJ53	0000 0000	–	–
PR8	PIC18F2XJ53	PIC18F4XJ53	1111 1111	–	–
T8CON	PIC18F2XJ53	PIC18F4XJ53	-000 0000	–	–
PSTR3CON	PIC18F2XJ53	PIC18F4XJ53	00-0 0001	–	–
ECCP3AS	PIC18F2XJ53	PIC18F4XJ53	0000 0000	–	–
ECCP3DEL	PIC18F2XJ53	PIC18F4XJ53	0000 0000	–	–
CCPR3H	PIC18F2XJ53	PIC18F4XJ53	xxxx xxxx	–	–
CCPR3L	PIC18F2XJ53	PIC18F4XJ53	xxxx xxxx	–	–
CCP3CON	PIC18F2XJ53	PIC18F4XJ53	0000 0000	–	–
CCPR4H	PIC18F2XJ53	PIC18F4XJ53	xxxx xxxx	–	–
CCPR4L	PIC18F2XJ53	PIC18F4XJ53	xxxx xxxx	–	–
CCP4CON	PIC18F2XJ53	PIC18F4XJ53	--00 0000	–	–
CCPR5H	PIC18F2XJ53	PIC18F4XJ53	xxxx xxxx	–	–
CCPR5L	PIC18F2XJ53	PIC18F4XJ53	xxxx xxxx	–	–
CCP5CON	PIC18F2XJ53	PIC18F4XJ53	--00 0000	–	–
CCPR6H	PIC18F2XJ53	PIC18F4XJ53	xxxx xxxx	–	–
CCPR6L	PIC18F2XJ53	PIC18F4XJ53	xxxx xxxx	–	–
CCP6CON	PIC18F2XJ53	PIC18F4XJ53	--00 0000	–	–
CCPR7H	PIC18F2XJ53	PIC18F4XJ53	xxxx xxxx	–	–
CCPR7L	PIC18F2XJ53	PIC18F4XJ53	xxxx xxxx	–	–
CCP7CON	PIC18F2XJ53	PIC18F4XJ53	--00 0000	–	–
CCPR8H	PIC18F2XJ53	PIC18F4XJ53	xxxx xxxx	–	–
CCPR8L	PIC18F2XJ53	PIC18F4XJ53	xxxx xxxx	–	–
CCP8CON	PIC18F2XJ53	PIC18F4XJ53	--00 0000	–	–
CCPR9H	PIC18F2XJ53	PIC18F4XJ53	xxxx xxxx	–	–
CCPR9L	PIC18F2XJ53	PIC18F4XJ53	xxxx xxxx	–	–
CCP9CON	PIC18F2XJ53	PIC18F4XJ53	--00 0000	–	–
CCPR10H	PIC18F2XJ53	PIC18F4XJ53	xxxx xxxx	–	–
CCPR10L	PIC18F2XJ53	PIC18F4XJ53	xxxx xxxx	–	–
CCP10CON	PIC18F2XJ53	PIC18F4XJ53	--00 0000	–	–
RPINR24	PIC18F2XJ53	PIC18F4XJ53	---1 1111	---1 1111	---u uuuu
RPINR23	PIC18F2XJ53	PIC18F4XJ53	---1 1111	---1 1111	---u 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.

9.5 RCON Register

The RCON register contains bits used to determine the cause of the last Reset or wake-up from Idle or Sleep mode. RCON also contains the bit that enables interrupt priorities (IPEN).

REGISTER 9-19: RCON: RESET CONTROL REGISTER (ACCESS FD0h)

R/W-0	U-0	R/W-1	R/W-1	R-1	R-1	R/W-0	R/W-0
IPEN	—	$\overline{CM}$	$\overline{RI}$	$\overline{TO}$	$\overline{PD}$	$\overline{POR}$	$\overline{BOR}$
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 **IPEN:** Interrupt Priority Enable bit
1 = Enable priority levels on interrupts
0 = Disable priority levels on interrupts (PIC16CXXX Compatibility mode)

bit 6 **Unimplemented:** Read as '0'

bit 5 **$\overline{CM}$:** Configuration Mismatch Flag bit
For details on bit operation, see Register 5-1.

bit 4 **$\overline{RI}$:** RESET Instruction Flag bit
For details on bit operation, see Register 5-1.

bit 3 **$\overline{TO}$:** Watchdog Timer Time-out Flag bit
For details on bit operation, see Register 5-1.

bit 2 **$\overline{PD}$:** Power-Down Detection Flag bit
For details on bit operation, see Register 5-1.

bit 1 **$\overline{POR}$:** Power-on Reset Status bit
For details on bit operation, see Register 5-1.

bit 0 **$\overline{BOR}$:** Brown-out Reset Status bit
For details on bit operation, see Register 5-1.

10.6 PORTE, TRISE and LATE Registers

Note: PORTE is available only in 44-pin devices.

Depending on the particular PIC18F47J53 family device selected, PORTE is implemented in two different ways.

For 44-pin devices, PORTE is a 3-bit wide port. Three pins (RE0/AN5/PMRD, RE1/AN6/PMWR and RE2/AN7/PMCS) are individually configurable as inputs or outputs. These pins have Schmitt Trigger input buffers. When selected as analog inputs, these pins will read as '0's.

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

TRISE controls the direction of the RE pins, even when they are being used as analog inputs. The user must make sure to keep the pins configured as inputs when using them as analog inputs.

Note: On a POR, RE<2:0> are configured as analog inputs.

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

EXAMPLE 10-6: INITIALIZING PORTE

```
CLRF   PORTE    ; Initialize PORTE by
                ; clearing output
                ; data latches
CLRF   LATE     ; Alternate method
                ; to clear output
                ; data latches
MOVLW  0Ah      ; Configure A/D
MOVWF  ADCON1   ; for digital inputs
MOVLW  03h      ; Value used to
                ; initialize data
                ; direction
MOVWF  TRISE    ; Set RE<0> as inputs
                ; RE<1> as outputs
                ; RE<2> as inputs
```

Each of the PORTE pins has a weak internal pull-up. A single control bit can turn on all the pull-ups. This is performed by setting bit, REPU (TRISE<6>). 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 that the pull-ups can be used for any set of features, similar to the pull-ups found on PORTB.

REGISTER 10-45: RPOR24: PERIPHERAL PIN SELECT OUTPUT REGISTER 24 (BANKED ED8h)(1)

U-0	U-0	U-0	R/$\overline{W}$-0	R/$\overline{W}$-0	R/$\overline{W}$-0	R/$\overline{W}$-0	R/$\overline{W}$-0
—	—	—	RP24R4	RP24R3	RP24R2	RP24R1	RP24R0
bit 7							bit 0

Legend:	R/$\overline{W}$ = Readable bit, Writable bit if IOLOCK = 0		
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 7-5 **Unimplemented:** Read as '0'

bit 4-0 **RP24R<4:0>:** Peripheral Output Function is Assigned to RP24 Output Pin bits (see Table 10-14 for peripheral function numbers)

Note 1: RP24 pins are not available on 28-pin devices.

15.5.2 TIMER3/5 GATE SOURCE SELECTION

The Timer3/5 gate source can be selected from one of four different sources. Source selection is controlled by the TxGSS<1:0> bits (TxGCON<1:0>). The polarity for each available source is also selectable and is controlled by the TxGPOL bit (TxGCON<6>).

TABLE 15-2: TIMER3/5 GATE SOURCES

TxGSS<1:0>	Timerx Gate Source
00	TxG timer gate pin
01	TMR4/6 matches PR4/6
10	Comparator 1 output
11	Comparator 2 output

15.5.2.1 TxG Pin Gate Operation

The TxG pin is one source for Timer3/5 gate control. It can be used to supply an external source to the gate circuitry.

15.5.2.2 Timer4/6 Match Gate Operation

The TMR4/6 register will increment until it matches the value in the PR4/6 register. On the very next increment cycle, TMR4/6 will be reset to 00h. When this Reset occurs, a low-to-high pulse will automatically be generated and internally supplied to the Timer3/5 gate circuitry.

15.5.3 TIMER3/5 GATE-TOGGLE MODE

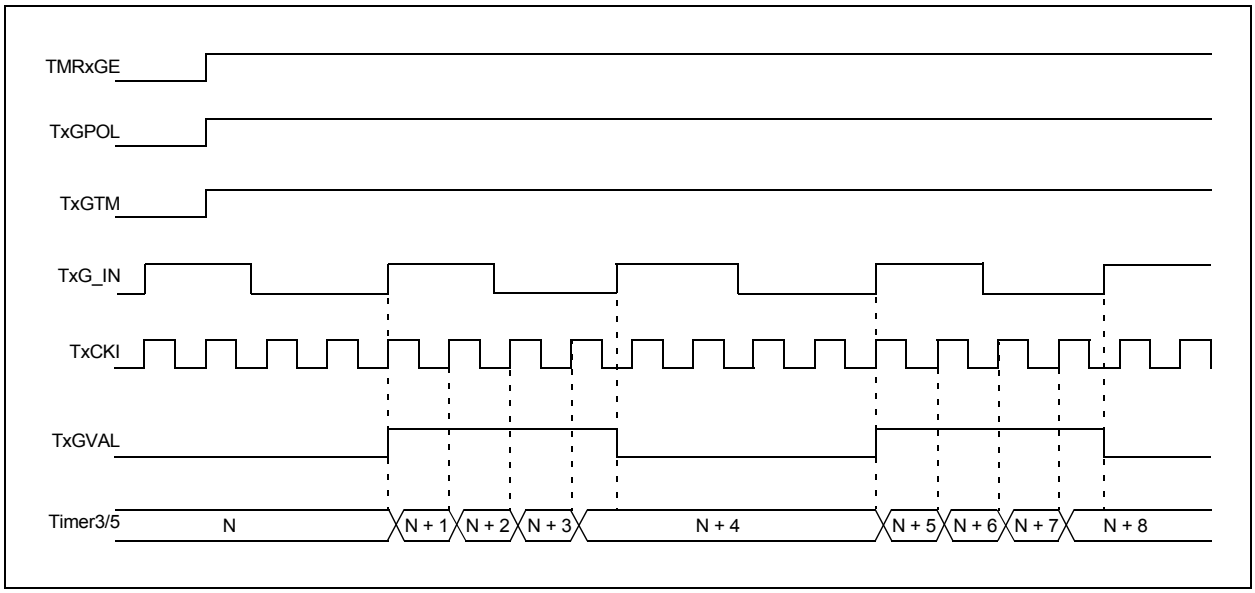
When Timer3/5 Gate Toggle mode is enabled, it is possible to measure the full cycle length of a Timer3/5 gate signal, as opposed to the duration of a single level pulse.

The Timer1 gate source is routed through a flip-flop that changes state on every incrementing edge of the signal. (For timing details, see Figure 15-3.)

The TxGVAL bit will indicate when the Toggled mode is active and the timer is counting.

Timer3/5 Gate Toggle mode is enabled by setting the TxGTM bit (TxGCON<5>). When the TxGTM bit is cleared, the flip-flop is cleared and held clear. This is necessary in order to control which edge is measured.

FIGURE 15-3: TIMER3/5 GATE TOGGLE MODE



NOTES:

REGISTER 17-9: DAY: DAY VALUE REGISTER(1)

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	DAYTEN1	DAYTEN0	DAYONE3	DAYONE2	DAYONE1	DAYONE0
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 7-6 **Unimplemented:** Read as '0'

bit 5-4 **DAYTEN<1:0>:** Binary Coded Decimal value of Day's Tens Digit bits
Contains a value from 0 to 3.

bit 3-0 **DAYONE<3:0>:** Binary Coded Decimal Value of Day's Ones Digit bits
Contains a value from 0 to 9.

Note 1: A write to this register is only allowed when RTCWREN = 1.

REGISTER 17-10: WKDY: WEEKDAY VALUE REGISTER(1)

U-0	U-0	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x
—	—	—	—	—	WDAY2	WDAY1	WDAY0
bit 7							bit 0

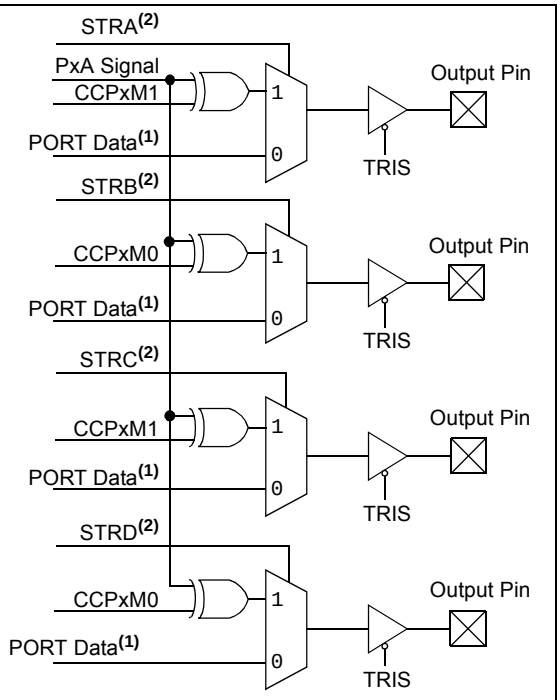
Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-3 **Unimplemented:** Read as '0'

bit 2-0 **WDAY<2:0>:** Binary Coded Decimal Value of Weekday Digit bits
Contains a value from 0 to 6.

Note 1: A write to this register is only allowed when RTCWREN = 1.

FIGURE 19-16: SIMPLIFIED STEERING BLOCK DIAGRAM



Note 1: Port outputs are configured as displayed when the CCPxCON register bits, PxM<1:0> = 00 and CCP1M<3:2> = 11.

2: Single PWM output requires setting at least one of the STR<D:A> bits.

19.4.7.1 Steering Synchronization

The STRSYNC bit of the PSTRxCON register gives the user two choices for when the steering event will happen. When the STRSYNC bit is '0', the steering event will happen at the end of the instruction that writes to the PSTRxCON register. In this case, the output signal at the Px<D:A> pins may be an incomplete PWM waveform. This operation is useful when the user firmware needs to immediately remove a PWM signal from the pin.

When the STRSYNC bit is '1', the effective steering update will happen at the beginning of the next PWM period. In this case, steering on/off the PWM output will always produce a complete PWM waveform.

Figure 19-17 and Figure 19-18 illustrate the timing diagrams of the PWM steering depending on the STRSYNC setting.

FIGURE 19-17: EXAMPLE OF STEERING EVENT AT END OF INSTRUCTION (STRSYNC = 0)

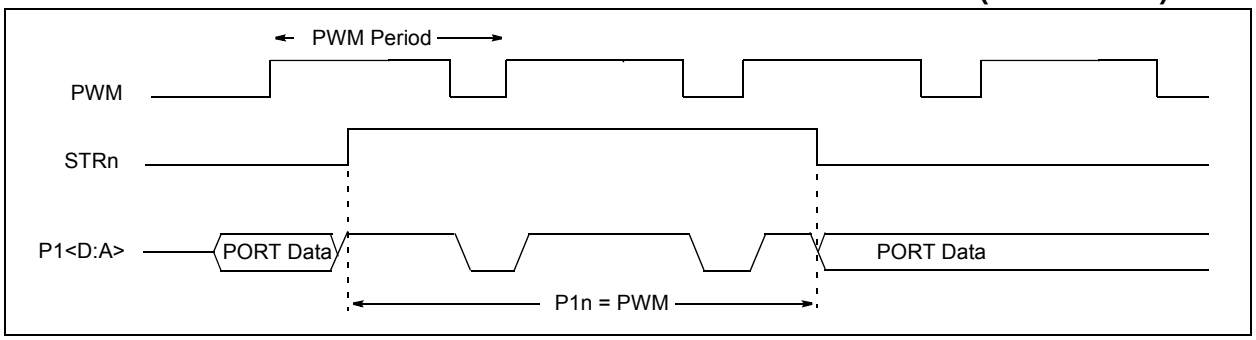
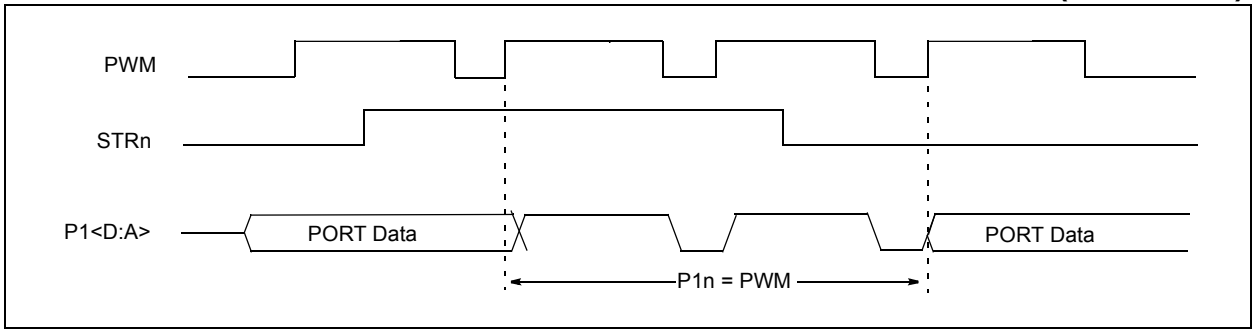


FIGURE 19-18: EXAMPLE OF STEERING EVENT AT BEGINNING OF INSTRUCTION (STRSYNC = 1)



20.3.4 ENABLING SPI I/O

To enable the serial port, MSSP Enable bit, SSPEN (SSPxCON1<5>), must be set. To reset or reconfigure SPI mode, clear the SSPEN bit, re-initialize the SSPxCON1 registers and then set the SSPEN bit. This configures the SDIx, SDOx, SCKx and $\overline{SS}$x pins as serial port pins. For the pins to behave as the serial port function, the appropriate TRISx bits, PCFGx bits and Peripheral Pin Select registers (if using MSSP2) should be correctly initialized prior to setting the SSPEN bit.

A typical SPI serial port initialization process follows:

- Initialize the ODCON3 register (optional open-drain output control)
- Initialize the remappable pin functions (if using MSSP2, see **Section 10.7 "Peripheral Pin Select (PPS)"**)
- Initialize the SCKx/LAT value to the desired Idle SCKx level (if master device)
- Initialize the SCKx/PCFGx bit (if in Slave mode and multiplexed with the ANx function)
- Initialize the SCKx/TRIS bit as output (Master mode) or input (Slave mode)
- Initialize the SDIx/PCFGx bit (if SDIx is multiplexed with the ANx function)
- Initialize the SDIx/TRIS bit
- Initialize the $\overline{SS}$x/PCFG bit (if in Slave mode and multiplexed the with ANx function)
- Initialize the $\overline{SS}$x/TRIS bit (Slave modes)
- Initialize the SDOx/TRIS bit
- Initialize the SSPxSTAT register
- Initialize the SSPxCON1 register
- Set the SSPEN bit to enable the module

Any MSSP1 serial port function that is not desired may be overridden by programming the corresponding Data Direction (TRIS) register to the opposite value. If individual MSSP2 serial port functions will not be used, they may be left unmapped.

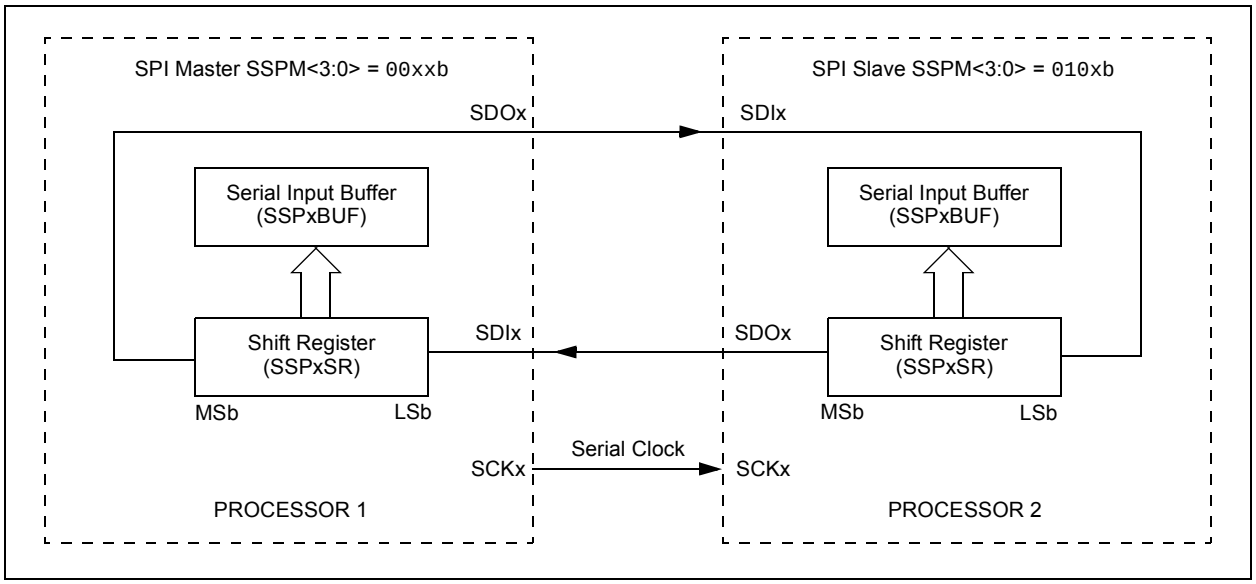
Note:	When MSSP2 is used in SPI Master mode, the SCK2 function must be configured as both an output and an input in the PPS module. SCK2 must be initialized as an output pin (by writing 0x0A to one of the RPORx registers). Additionally, SCK2IN must also be mapped to the same pin by initializing the RPINR22 register. Failure to initialize SCK2/SCK2IN as both output and input will prevent the module from receiving data on the SDI2 pin, as the module uses the SCK2IN signal to latch the received data.

20.3.5 TYPICAL CONNECTION

Figure 20-2 illustrates a typical connection between two microcontrollers. The master controller (Processor 1) initiates the data transfer by sending the SCKx signal. Data is shifted out of both shift registers on their programmed clock edge and latched on the opposite edge of the clock. Both processors should be programmed to the same Clock Polarity (CKP), then both controllers would send and receive data at the same time. Whether the data is meaningful (or dummy data) depends on the application software. This leads to three scenarios for data transmission:

- Master sends valid data – Slave sends dummy data
- Master sends valid data – Slave sends valid data
- Master sends dummy data – Slave sends valid data

FIGURE 20-2: SPI MASTER/SLAVE CONNECTION



REGISTER 21-1: TXSTAx: TRANSMIT STATUS AND CONTROL REGISTER (1, ACCESS FADh; 2, FA8h)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R-1	R/W-0
CSRC	TX9	TXEN(1)	SYNC	SENDB	BRGH	TRMT	TX9D
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 **CSRC:** Clock Source Select bit
Asynchronous mode:
Don't care.
Synchronous mode:
1 = Master mode (clock generated internally from BRG)
0 = Slave mode (clock from external source)

bit 6 **TX9:** 9-Bit Transmit Enable bit
1 = Selects 9-bit transmission
0 = Selects 8-bit transmission

bit 5 **TXEN:** Transmit Enable bit(1)
1 = Transmit is enabled
0 = Transmit is disabled

bit 4 **SYNC:** EUSART Mode Select bit
1 = Synchronous mode
0 = Asynchronous mode

bit 3 **SENDB:** Send Break Character bit
Asynchronous mode:
1 = Send Sync Break on next transmission (cleared by hardware upon completion)
0 = Sync Break transmission completed
Synchronous mode:
Don't care.

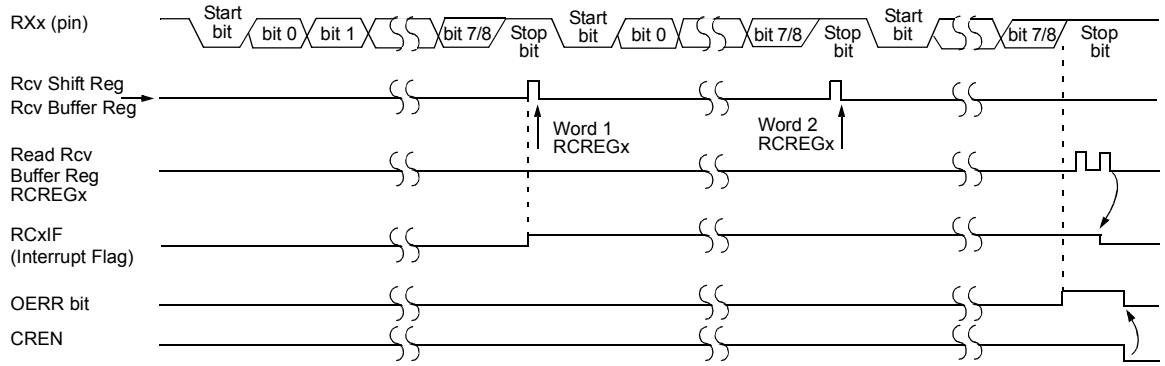
bit 2 **BRGH:** High Baud Rate Select bit
Asynchronous mode:
1 = High speed
0 = Low speed
Synchronous mode:
Unused in this mode.

bit 1 **TRMT:** Transmit Shift Register Status bit
1 = TSR is empty
0 = TSR is full

bit 0 **TX9D:** 9th bit of Transmit Data
Can be address/data bit or a parity bit.

Note 1: SREN/CREN overrides TXEN in Sync mode.

FIGURE 21-7: ASYNCHRONOUS RECEPTION



Note: This timing diagram shows three words appearing on the RXx input. The RCREGx (Receive Buffer) is read after the third word causing the OERR (Overrun) bit to be set.

TABLE 21-6: REGISTERS ASSOCIATED WITH ASYNCHRONOUS RECEPTION

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
PIR3	SSP2IF	BCL2IF	RC2IF	TX2IF	TMR4IF	CTMUIF	TMR3GIF	RTCCIF
PIE3	SSP2IE	BCL2IE	RC2IE	TX2IE	TMR4IE	CTMUIE	TMR3GIE	RTCCIE
IPR3	SSP2IP	BCL2IP	RC2IP	TX2IP	TMR4IP	CTMUIP	TMR3GIP	RTCCIP
RCSTAx	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D
RCREGx	EUSARTx Receive Register							
TXSTAx	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D
BAUDCONx	ABDOVF	RCIDL	RXDTP	TXCKP	BRG16	—	WUE	ABDEN
SPBRGHx	EUSARTx Baud Rate Generator High Byte							
SPBRGx	EUSARTx Baud Rate Generator Low Byte							

Legend: — = unimplemented locations read as '0'. Shaded cells are not used for asynchronous reception.

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

21.2.4 AUTO-WAKE-UP ON SYNC BREAK CHARACTER

During Sleep mode, all clocks to the EUSART are suspended. Because of this, the BRG is inactive and a proper byte reception cannot be performed. The auto-wake-up feature allows the controller to wake-up due to activity on the RXx/DTx line while the EUSART is operating in Asynchronous mode.

The auto-wake-up feature is enabled by setting the WUE bit (BAUDCONx<1>). Once set, the typical receive sequence on RXx/DTx is disabled and the EUSART remains in an Idle state, monitoring for a wake-up event independent of the CPU mode. A wake-up event consists of a high-to-low transition on the RXx/DTx line. (This coincides with the start of a Sync Break or a Wake-up Signal character for the LIN/J2602 protocol.)

Following a wake-up event, the module generates an RCxIF interrupt. The interrupt is generated synchronously to the Q clocks in normal operating modes (Figure 21-8) and asynchronously if the device is in Sleep mode (Figure 21-9). The interrupt condition is cleared by reading the RCREGx register.

REGISTER 22-3: CONFIG3H: CONFIGURATION REGISTER 3 HIGH (BYTE ADDRESS 300005h)

U-1	U-1	U-1	U-1	R/WO-1	U-0	R/WO-1	R/WO-1
—	—	—	—	MSSPMSK	—	ADCSEL	IOL1WAY
bit 7							bit 0

Legend:			
R = Readable bit	WO = Write-Once bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-4 **Unimplemented:** Program the corresponding Flash Configuration bit to '1'

bit 3 **MSSPMSK:** MSSP 7-Bit Address Masking Mode Enable bit
1 = 7-Bit Address Masking mode enabled
0 = 5-Bit Address Masking mode enabled

bit 2 **Unimplemented:** Read as '0'

bit 1 **ADCSEL:** A/D Converter Mode bit
1 = 10-Bit Conversion mode enabled
0 = 12-Bit Conversion mode enabled

bit 0 **IOL1WAY:** IOLOCK One Way Set Enable bit
1 = IOLOCK bit (PPSCON<0>) can be set once, provided the unlock sequence has been completed. Once set, the Peripheral Pin Select registers cannot be written to a second time.
0 = IOLOCK bit (PPSCON<0>) can be set and cleared as needed, provided the unlock sequence has been completed

23.5.1.1 Bus Activity Detect Interrupt Bit (ACTVIF)

The ACTVIF bit cannot be cleared immediately after the USB module wakes up from Suspend mode or while the USB module is suspended. A few clock cycles are required to synchronize the internal hardware state machine before the ACTVIF bit can be cleared by firmware. Clearing the ACTVIF bit before the internal hardware is synchronized may not have an effect on the value of ACTVIF. Additionally, if the USB module uses the clock from the 96 MHz PLL source, then after clearing the SUSPND bit, the USB module may not be immediately operational while waiting for the 96 MHz PLL to lock. The application code should clear the ACTVIF flag as provided in Example 23-1.

Note:	Only one ACTVIF interrupt is generated when resuming from the USB bus Idle condition. If user firmware clears the ACTVIF bit, the bit will not immediately become set again, even when there is continuous bus traffic. Bus traffic must cease long enough to generate another IDLEIF condition before another ACTVIF interrupt can be generated.

EXAMPLE 23-1: CLEARING ACTVIF BIT (UIR<2>)

Assembly:

```
    BCF     UCON, SUSPND
LOOP:
    BTFSS   UIR, ACTVIF
    BRA     DONE
    BCF     UIR, ACTVIF
    BRA     LOOP
DONE:
```

C:

```
UCONbits.SUSPND = 0;
while (UIRbits.ACTVIF) { UIRbits.ACTVIF = 0; }
```

REGISTER 28-2: CONFIG1H: CONFIGURATION REGISTER 1 HIGH (BYTE ADDRESS 300001h)

U-1	U-1	U-1	U-1	U-0	R/WO-1	R/WO-1	R/WO-1
—	—	—	—	—	CP0	CPDIV1	CPDIV0
bit 7							bit 0

Legend:			
R = Readable bit	WO = Write-Once bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 7-4 **Unimplemented:** Program the corresponding Flash Configuration bit to ‘1’

bit 3 **Unimplemented:** Maintain as ‘0’

bit 2 **CP0:** Code Protection bit
- 1 = Program memory is not code-protected
- 0 = Program memory is code-protected

bit 1-0 **CPDIV<1:0>:** CPU System Clock Selection bits
- 11 = No CPU system clock divide
- 10 = CPU system clock is divided by 2
- 01 = CPU system clock is divided by 3
- 00 = CPU system clock is divided by 6

TABLE 29-2: PIC18F47J53 FAMILY INSTRUCTION SET

Mnemonic, Operands		Description	Cycles	16-Bit Instruction Word MSb LSb	Status Affected	Notes
BYTE-ORIENTED OPERATIONS						
ADDWF	f, d, a	Add WREG and f	1	0010 01da ffff ffff	C, DC, Z, OV, N	1, 2
ADDWFC	f, d, a	Add WREG and Carry bit to f	1	0010 00da ffff ffff	C, DC, Z, OV, N	1, 2
ANDWF	f, d, a	AND WREG with f	1	0001 01da ffff ffff	Z, N	1,2
CLRF	f, a	Clear f	1	0110 101a ffff ffff	Z	2
COMF	f, d, a	Complement f	1	0001 11da ffff ffff	Z, N	1, 2
CPFSEQ	f, a	Compare f with WREG, Skip =	1 (2 or 3)	0110 001a ffff ffff	None	4
CPFSGT	f, a	Compare f with WREG, Skip >	1 (2 or 3)	0110 010a ffff ffff	None	4
CPFSLT	f, a	Compare f with WREG, Skip <	1 (2 or 3)	0110 000a ffff ffff	None	1, 2
DECF	f, d, a	Decrement f	1	0000 01da ffff ffff	C, DC, Z, OV, N	1, 2, 3, 4
DECFSZ	f, d, a	Decrement f, Skip if 0	1 (2 or 3)	0010 11da ffff ffff	None	1, 2, 3, 4
DCFSNZ	f, d, a	Decrement f, Skip if Not 0	1 (2 or 3)	0100 11da ffff ffff	None	1, 2
INCF	f, d, a	Increment f	1	0010 10da ffff ffff	C, DC, Z, OV, N	1, 2, 3, 4
INCFSZ	f, d, a	Increment f, Skip if 0	1 (2 or 3)	0011 11da ffff ffff	None	4
INFSNZ	f, d, a	Increment f, Skip if Not 0	1 (2 or 3)	0100 10da ffff ffff	None	1, 2
IORWF	f, d, a	Inclusive OR WREG with f	1	0001 00da ffff ffff	Z, N	1, 2
MOVF	f, d, a	Move f	1	0101 00da ffff ffff	Z, N	1
MOVFF	f_s, f_d	Move f_s (source) to 1st word f_d (destination) 2nd word	2	1100 ffff ffff ffff 1111 ffff ffff ffff	None	
MOVWF	f, a	Move WREG to f	1	0110 111a ffff ffff	None	
MULWF	f, a	Multiply WREG with f	1	0000 001a ffff ffff	None	1, 2
NEGF	f, a	Negate f	1	0110 110a ffff ffff	C, DC, Z, OV, N	
RLCF	f, d, a	Rotate Left f through Carry	1	0011 01da ffff ffff	C, Z, N	1, 2
RLNCF	f, d, a	Rotate Left f (No Carry)	1	0100 01da ffff ffff	Z, N	
RRCF	f, d, a	Rotate Right f through Carry	1	0011 00da ffff ffff	C, Z, N	
RRNCF	f, d, a	Rotate Right f (No Carry)	1	0100 00da ffff ffff	Z, N	
SETF	f, a	Set f	1	0110 100a ffff ffff	None	1, 2
SUBFWB	f, d, a	Subtract f from WREG with Borrow	1	0101 01da ffff ffff	C, DC, Z, OV, N	
SUBWF	f, d, a	Subtract WREG from f	1	0101 11da ffff ffff	C, DC, Z, OV, N	1, 2
SUBWFB	f, d, a	Subtract WREG from f with Borrow	1	0101 10da ffff ffff	C, DC, Z, OV, N	
SWAPF	f, d, a	Swap Nibbles in f	1	0011 10da ffff ffff	None	4
TSTFSZ	f, a	Test f, Skip if 0	1 (2 or 3)	0110 011a ffff ffff	None	1, 2
XORWF	f, d, a	Exclusive OR WREG with f	1	0001 10da ffff ffff	Z, N	

Note 1: When a PORT register is modified as a function of itself (e.g., `MOVF PORTB, 1, 0`), the value used will be that value present on the pins themselves. For example, if the data latch is '1' for a pin configured as input and is driven low by an external device, the data will be written back with a '0'.

2: If this instruction is executed on the TMR0 register (and, where applicable, d = 1), the prescaler will be cleared if assigned.

3: If Program Counter (PC) is modified or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a NOP.

4: Some instructions are two-word instructions. The second word of these instructions will be executed as a NOP unless the first word of the instruction retrieves the information embedded in these 16 bits. This ensures that all program memory locations have a valid instruction.

MOVSS **Move Indexed to Indexed**

Syntax: MOVSS [z_s], [z_d]

Operands: $0 \le z_s \le 127$
$0 \le z_d \le 127$

Operation: ((FSR2) + z_s) → ((FSR2) + z_d)

Status Affected: None

Encoding:

1st word (source)	1110	1011	1zzz	$zzzz_s$
2nd word (dest.)	1111	xxxx	xzzz	$zzzz_d$

Description: The contents of the source register are moved to the destination register. The addresses of the source and destination registers are determined by adding the 7-bit literal offsets, 'z_s' or 'z_d', respectively, to the value of FSR2. Both registers can be located anywhere in the 4096-byte data memory space (000h to FFFh).

The MOVSS instruction cannot use the PCL, TOSU, TOSH or TOSL as the destination register.

If the resultant source address points to an Indirect Addressing register, the value returned will be 00h. If the resultant destination address points to an Indirect Addressing register, the instruction will execute as a NOP.

Words: 2

Cycles: 2

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Determine source addr	Determine source addr	Read source reg
Decode	Determine dest addr	Determine dest addr	Write to dest reg

Example: MOVSS [05h], [06h]

Before Instruction
FSR2 = 80h
Contents of 85h = 33h
Contents of 86h = 11h
After Instruction
FSR2 = 80h
Contents of 85h = 33h
Contents of 86h = 33h

PUSHL **Store Literal at FSR2, Decrement FSR2**

Syntax: PUSHL k

Operands: $0 \le k \le 255$

Operation: k → (FSR2),
FSR2 – 1 → FSR2

Status Affected: None

Encoding:

1111	1010	kkkk	kkkk

Description: The 8-bit literal 'k' is written to the data memory address specified by FSR2. FSR2 is decremented by 1 after the operation.

This instruction allows users to push values onto a software stack.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read 'k'	Process data	Write to destination

Example: PUSHL 08h

Before Instruction
FSR2H:FSR2L = 01ECh
Memory (01ECh) = 00h

After Instruction
FSR2H:FSR2L = 01EBh
Memory (01ECh) = 08h

31.2 DC Characteristics: Power-Down and Supply Current PIC18F47J53 Family (Industrial)

PIC18LF47J53 Family		Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial					
PIC18F47J53 Family		**Standard Operating Conditions (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +85°C for industrial					
Param. No.	**Device**	**Typ.**	**Max.**	**Units**	**Conditions**		
	Power-Down Current (IPD)(1) – Sleep mode						
	PIC18LFXXJ53	0.1	1.6	µA	-40°C	VDD = 2.0V, VDDCORE = 2.0V	**Sleep mode**, REGSLP = 1
		0.2	1.6	µA	+25°C		
		0.8	7.0	µA	+60°C		
		2.1	11.5	µA	+85°C		
	PIC18LFXXJ53	0.2	2.0	µA	-40°C	VDD = 2.5V, VDDCORE = 2.5V	
		0.5	2.0	µA	+25°C		
		1.4	9.0	µA	+60°C		
		3.2	15.0	µA	+85°C		
	PIC18FXXJ53	3.0	6.0	µA	-40°C	VDD = 2.15V Vddcore = 10 µF Capacitor	
		3.8	6.0	µA	+25°C		
		4.7	9.0	µA	+60°C		
		6.4	18.5	µA	+85°C		
	PIC18FXXJ53	3.3	9.0	µA	-40°C	VDD = 3.3V Vddcore = 10 µF Capacitor	
		4.2	9.0	µA	+25°C		
		5.5	12.0	µA	+60°C		
		7.8	22.0	µA	+85°C		
	Power-Down Current (IPD)(1) – Deep Sleep mode						
	PIC18FXXJ53	2	25	nA	-40°C	VDD = 2.15V, VDDCORE = 10 µF Capacitor	**Deep Sleep** mode
		9	100	nA	+25°C		
		72	250	nA	+60°C		
		0.26	1.0	µA	+85°C		
	PIC18FXXJ53	17	50	nA	-40°C	VDD = 3.3V, VDDCORE = 10 µF Capacitor	
		53	150	nA	+25°C		
		186	400	nA	+60°C		
		0.50	2.0	µA	+85°C		

Note 1: The power-down current in Sleep mode does not depend on the oscillator type. Power-down current is measured with the part in Sleep mode, with all I/O pins in high-impedance state and tied to VDD or VSS and all features that add delta current disabled (such as WDT, Timer1 oscillator, BOR, etc.).

2: The supply current is mainly a function of operating voltage, frequency and mode. Other factors, such as I/O pin loading and switching rate, oscillator type and circuit, internal code execution pattern and temperature, also have an impact on the current consumption. All features that add delta current are disabled (USB module, WDT, etc.). The test conditions for all IDD measurements in active operation mode are:
OSC1 = external square wave, from rail-to-rail; all I/O pins tri-stated, pulled to VDD/VSS;
$\overline{MCLR}$ = VDD; WDT disabled unless otherwise specified.

3: Low-power Timer1 with standard, low-cost 32 kHz crystals have an operating temperature range of -10°C to +70°C. Extended temperature crystals are available at a much higher cost.

4: This is the module differential current when the USB module is enabled and clocked at 48 MHz, but with no USB cable attached. When the USB cable is attached or data is being transmitted, the current consumption may be much higher (see **Section 23.6.4 "USB Transceiver Current Consumption"**). During USB Suspend mode (USBEN = 1, SUSPND = 1, bus in Idle state), the USB module current will be dominated by the D+ or D- pull-up resistor. The integrated pull-up resistors use "resistor switching" according to the `resistor_ecn` supplement to the USB 2.0 Specifications, and therefore, may be as low as 900Ω during Idle conditions.

APPENDIX B: MIGRATION FROM PIC18F46J50 TO PIC18F47J53

Code for the devices in the PIC18F46J50 family can be migrated to the PIC18F47J53 without many changes. The differences between the two device families are listed in Table B-1.

TABLE B-1: NOTABLE DIFFERENCES BETWEEN PIC18F47J53 AND PIC18F46J50 FAMILIES

Characteristic	PIC18F47J53 Family	PIC18F46J50 Family
Max Program Memory	128 Kbytes	64 Kbytes
Oscillator options	PLL can be enabled at start-up with Config bit option	Requires firmware to set the PLLEN bit at run time
SOSC Oscillator Options	Low-power oscillator option for SOSC, with run-time switch	Low-power oscillator option for SOSC, only via Configuration bit setting
T1CKI Clock Input	T1CKI can be used as a clock input without enabling the Timer1 oscillator	No
INTOSC	Up to 8 MHz	Up to 8 MHz
Timers	8	5
ECCP	3	2
CCP	7	0
SPI Fosc/8 Master Clock Option	Yes	No
ADC	13 Channel, 10/12-bit conversion modes with Special Event Trigger option.	13 Channel, 10-bit only
Peripheral Module Disable Bits	Yes, allowing further power reduction	No
Band Gap Voltage Reference Output	Yes, enabled on pin RA1 by setting the VBGOE bit (WDTCON<4>)	No
REPU/RDPU Pull-Up Enable Bits	Moved to TRISE register (avoids read, modify, write issues)	Pull-up bits configured in PORTE register
Comparators	Three, each with four input-pin selections	Two, each with two input-pin selections
Increased Output Drive Strength	RA0 through RA5, RDx and REx	No