$ % && & $'

( ) $ % && &
$

- ,	!
$ -	- $
$, #	%.
,	/ % 0#
$	. 1$, - 2! 3 2, ! 3
-	. 4 3 564 - 7 3 - 8 8 4
9 7	' %
- , #	: ' ; . ⊲ %¡ * &' =
-	>5! , 0
- 3 7 , #	
3! , #	(: (? * %
6 , <6 6 =	) 6 @ (' 6
4 $	! 4 &A* &?
7	
7	/ ?B$ @ %AB$ < ! =
	,
- $	%? C>-
, 4 -	%? C>- <&) * &) =
- 235	D * &%% E&'

TABLE 1-3: PIC18F6XJ1X PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name	Pin Number 64-TQFP	Pin Type	Buffer Type	Description
				PORTC is a bidirectional I/O port.
RC0/T1OSO/T13CKI	30			
RC0		I/O	ST	Digital I/O.
T1OSO		O	—	Timer1 oscillator output.
T13CKI		I	ST	Timer1/Timer3 external clock input.
RC1/T1OSI/ECCP2/P2A	29			
RC1		I/O	ST	Digital I/O.
T1OSI		I	CMOS	Timer1 oscillator input.
ECCP2(1)		I/O	ST	Capture 2 input/Compare 2 output/PWM2 output.
P2A(1)		O	—	ECCP2 PWM Output A.
RC2/ECCP1/P1A	33			
RC2		I/O	ST	Digital I/O.
ECCP1		I/O	ST	Capture 1 input/Compare 1 output/PWM1 output.
P1A		O	—	ECCP1 PWM Output A.
RC3/SCK1/SCL1	34			
RC3		I/O	ST	Digital I/O.
SCK1		I/O	ST	Synchronous serial clock input/output for SPI mode.
SCL1		I/O	I^2C	Synchronous serial clock input/output for I^2C mode.
RC4/SDI1/SDA1	35			
RC4		I/O	ST	Digital I/O.
SDI1		I	ST	SPI data in.
SDA1		I/O	I^2C	I^2C data I/O.
RC5/SDO1	36			
RC5		I/O	ST	Digital I/O.
SDO1		O	—	SPI data out.
RC6/TX1/CK1	31			
RC6		I/O	ST	Digital I/O.
TX1		O	—	EUSART1 asynchronous transmit.
CK1		I/O	ST	EUSART1 synchronous clock (see related RX1/DT1).
RC7/RX1/DT1	32			
RC7		I/O	ST	Digital I/O.
RX1		I	ST	EUSART1 asynchronous receive.
DT1		I/O	ST	EUSART1 synchronous data (see related TX1/CK1).

Legend: TTL = TTL compatible input; ST = Schmitt Trigger input with CMOS levels; I = Input; P = Power; I^2C = ST with I^2C™ or SMB levels; CMOS = CMOS compatible input or output; Analog = Analog input; O = Output; OD = Open-Drain (no P diode to V_{DD})

Note 1: Default assignment for ECCP2/P2A when Configuration bit, CCP2MX, is set.

2: Alternate assignment for ECCP2/P2A when Configuration bit, CCP2MX, is cleared.

TABLE 1-4: PIC18F8XJ1X PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name	Pin Number 80-TQFP	Pin Type	Buffer Type	Description
				PORTC is a bidirectional I/O port.
RC0/T1OSO/T13CKI	36			
RC0		I/O	ST	Digital I/O.
T1OSO		O	—	Timer1 oscillator output.
T13CKI		I	ST	Timer1/Timer3 external clock input.
RC1/T1OSI/ECCP2/P2A	35			
RC1		I/O	ST	Digital I/O.
T1OSI		I	CMOS	Timer1 oscillator input.
ECCP2(2)		I/O	ST	Capture 2 input/Compare 2 output/PWM2 output.
P2A(2)		O	—	ECCP2 PWM Output A.
RC2/ECCP1/P1A	43			
RC2		I/O	ST	Digital I/O.
ECCP1		I/O	ST	Capture 1 input/Compare 1 output/PWM1 output.
P1A		O	—	ECCP1 PWM Output A.
RC3/SCK1/SCL1	44			
RC3		I/O	ST	Digital I/O.
SCK1		I/O	ST	Synchronous serial clock input/output for SPI mode.
SCL1		I/O	I^2C	Synchronous serial clock input/output for I^2C mode.
RC4/SDI1/SDA1	45			
RC4		I/O	ST	Digital I/O.
SDI1		I	ST	SPI data in.
SDA1		I/O	I^2C	I^2C data I/O.
RC5/SDO1	46			
RC5		I/O	ST	Digital I/O.
SDO1		O	—	SPI data out.
RC6/TX1/CK1	37			
RC6		I/O	ST	Digital I/O.
TX1		O	—	EUSART1 asynchronous transmit.
CK1		I/O	ST	EUSART1 synchronous clock (see related RX1/DT1).
RC7/RX1/DT1	38			
RC7		I/O	ST	Digital I/O.
RX1		I	ST	EUSART1 asynchronous receive.
DT1		I/O	ST	EUSART1 synchronous data (see related TX1/CK1).

Legend: TTL = TTL compatible input; ST = Schmitt Trigger input with CMOS levels; I = Input; P = Power; I^2C = ST with I^2C™ or SMB levels; CMOS = CMOS compatible input or output; Analog = Analog input; O = Output; OD = Open-Drain (no P diode to V_{DD})

Note 1: Alternate assignment for ECCP2/P2A when Configuration bit, CCP2MX, is cleared (Extended Microcontroller mode).
2: Default assignment for ECCP2/P2A for all devices in all operating modes (CCP2MX is set).
3: Default assignments for P1B/P1C/P3B/P3C (ECCPMX Configuration bit is set).
4: Alternate assignment for ECCP2/P2A when CCP2MX is cleared (Microcontroller mode).
5: Alternate assignments for P1B/P1C/P3B/P3C (ECCPMX Configuration bit is cleared).
6: Default assignment for PMP data and control pins when PMPMX Configuration bit is set.
7: Alternate assignment for PMP data and control pins when PMPMX Configuration bit is cleared (programmed).

2.0 GUIDELINES FOR GETTING STARTED WITH PIC18FJ MICROCONTROLLERS

2.1 Basic Connection Requirements

Getting started with the PIC18F87J11 family family of 8-bit microcontrollers requires attention to a minimal set of device pin connections before proceeding with development.

The following pins must always be connected:

- All VDD and VSS pins
  (see **Section 2.2 "Power Supply Pins"**)
- All AVDD and AVSS pins, regardless of whether or not the analog device features are used
  (see **Section 2.2 "Power Supply Pins"**)
- $\overline{\text{MCLR}}$ pin
  (see **Section 2.3 "Master Clear (MCLR) Pin"**)
- ENVREG (if implemented) and VCAP/VDDCORE pins
  (see **Section 2.4 "Voltage Regulator Pins (ENVREG and VCAP/VDDCORE)"**)

These pins must also be connected if they are being used in the end application:

- PGC/PGD pins used for In-Circuit Serial Programming™ (ICSP™) and debugging purposes
  (see **Section 2.5 "ICSP Pins"**)
- OSCI and OSCO pins when an external oscillator source is used
  (see **Section 2.6 "External Oscillator Pins"**)

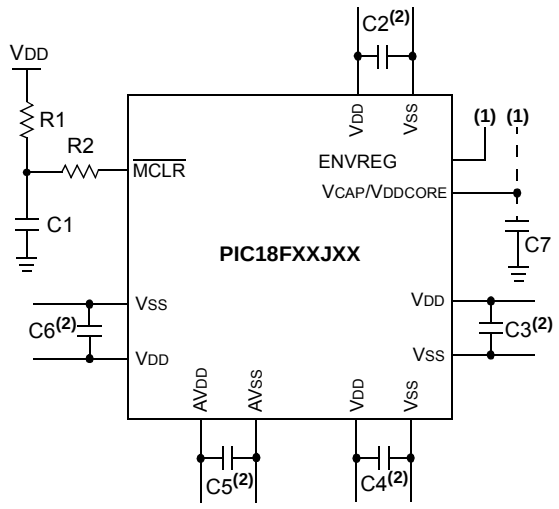
Additionally, the following pins may be required:

- VREF+/VREF- pins are used when external voltage reference for analog modules is implemented

Note:	The AVDD and AVSS pins must always be connected, regardless of whether any of the analog modules are being used.

The minimum mandatory connections are shown in Figure 2-1.

FIGURE 2-1: RECOMMENDED MINIMUM CONNECTIONS



Key (all values are recommendations):

C1 through C6: 0.1 F, 20V ceramic

C7: 10 F, 6.3V or greater, tantalum or ceramic

R1: 10 kq

R2: 100q to 470q

Note 1: See **Section 2.4 "Voltage Regulator Pins (ENVREG and VCAP/VDDCORE)"** for explanation of ENVREG pin connections.

2: The example shown is for a PIC18F device with five VDD/VSS and AVDD/AVSS pairs. Other devices may have more or less pairs; adjust the number of decoupling capacitors appropriately.

TABLE 18-2: REGISTERS ASSOCIATED WITH CAPTURE, COMPARE, TIMER1 AND TIMER3

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset Values on Page:
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	61
RCON	IPEN	—	$\overline{\text{CM}}$	$\overline{\text{RI}}$	$\overline{\text{TO}}$	$\overline{\text{PD}}$	$\overline{\text{POR}}$	$\overline{\text{BOR}}$	62
PIR1	PMPIF	ADIF	RC1IF	TX1IF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	64
PIE1	PMPIE	ADIE	RC1IE	TX1IE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	64
IPR1	PMPIP	ADIP	RC1IP	TX1IP	SSP1IP	CCP1IP	TMR2IP	TMR1IP	64
PIR2	OSCFIF	CM2IF	CM1IF	—	BCL1IF	LVDIF	TMR3IF	CCP2IF	64
PIE2	OSCFIE	CM2IE	CM1IE	—	BCL1IE	LVDIE	TMR3IE	CCP2IE	64
IPR2	OSCFIP	CM2IP	CM1IP	—	BCL1IP	LVDIP	TMR3IP	CCP2IP	64
PIR3	SSP2IF	BCL2IF	RC2IF	TX2IF	TMR4IF	CCP5IF	CCP4IF	CCP3IF	64
PIE3	SSP2IE	BCL2IE	RC2IE	TX2IE	TMR4IE	CCP5IE	CCP4IE	CCP3IE	64
IPR3	SSP2IP	BCL2IP	RC2IP	TX2IP	TMR4IP	CCP5IP	CCP4IP	CCP3IP	64
TRISG	—	—	—	TRISG4	TRISG3	TRISG2	TRISG1	TRISG0	64
TMR1L(1)	Timer1 Register Low Byte								62
TMR1H(1)	Timer1 Register High Byte								62
ODCON1(2)	—	—	—	CCP5OD	CCP4OD	ECCP3OD	ECCP2OD	ECCP1OD	62
T1CON(1)	RD16	T1RUN	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{\text{T1SYNC}}$	TMR1CS	TMR1ON	62
TMR3H	Timer3 Register High Byte								65
TMR3L	Timer3 Register Low Byte								65
T3CON	RD16	T3CCP2	T3CKPS1	T3CKPS0	T3CCP1	$\overline{\text{T3SYNC}}$	TMR3CS	TMR3ON	65
CCPR4L	Capture/Compare/PWM Register 4 Low Byte								65
CCPR4H	Capture/Compare/PWM Register 4 High Byte								65
CCPR5L	Capture/Compare/PWM Register 5 Low Byte								65
CCPR5H	Capture/Compare/PWM Register 5 High Byte								65
CCP4CON	—	—	DC4B1	DC4B0	CCP4M3	CCP4M2	CCP4M1	CCP4M0	65
CCP5CON	—	—	DC5B1	DC5B0	CCP5M3	CCP5M2	CCP5M1	CCP5M0	65

Legend: — = unimplemented, read as ‘0’. Shaded cells are not used by Capture/Compare, Timer1 or Timer3.

Note 1: Default (legacy) SFR at this address, available when WDTCON<4> = 0.

2: Configuration SFR, overlaps with the default SFR at this address; available only when WDTCON<4> = 1.

REGISTER 19-1: CCPxCON: ECCPx CONTROL REGISTER (ECCP1/ECCP2/ECCP3) (CONTINUED)

bit 3-0 **CCPxM<3:0>:** Enhanced CCPx Module Mode Select bits
0000 = Capture/Compare/PWM off (resets ECCPx module)
0001 = Reserved
0010 = Compare mode, toggle output on match
0011 = Capture mode
0100 = Capture mode: Every falling edge
0101 = Capture mode: Every rising edge
0110 = Capture mode: Every 4th rising edge
0111 = Capture mode: Every 16th rising edge
1000 = Compare mode: Initialize ECCPx pin low; set output on compare match (set CCPxIF)
1001 = Compare mode: Initialize ECCPx pin high; clear output on compare match (set CCPxIF)
1010 = Compare mode: Generate software interrupt only; ECCPx pin reverts to I/O state
1011 = Compare mode: Trigger special event (ECCPx resets TMR1 or TMR3, sets CCPxIF bit, ECCPx trigger also starts A/D conversion if A/D module is enabled)(1)
1100 = PWM mode: PxA, PxC active-high; PxB, PxD active-high
1101 = PWM mode: PxA, PxC active-high; PxB, PxD active-low
1110 = PWM mode: PxA, PxC active-low; PxB, PxD active-high
1111 = PWM mode: PxA, PxC active-low; PxB, PxD active-low

Note 1: Implemented only for ECCP1 and ECCP2; the same as '1010' for ECCP3.

19.1 ECCP Outputs and Configuration

Each of the Enhanced CCP modules may have up to four PWM outputs, depending on the selected operating mode. These outputs, designated PxA through PxD, are multiplexed with various I/O pins. Some ECCP pin assignments are constant, while others change based on device configuration. For those pins that do change, the controlling bits are:

- CCP2MX Configuration bit
- ECCPMX Configuration bit (80-pin devices only)
- Program Memory Operating mode, set by the EMBx Configuration bits (80-pin devices only)

The pin assignments for the Enhanced CCP modules are summarized in Table 19-1, Table 19-2 and Table 19-3. To configure the I/O pins as PWM outputs, the proper PWM mode must be selected by setting the PxMx and CCPxMx bits (CCPxCON<7:6> and <3:0>, respectively). The appropriate TRIS direction bits for the corresponding port pins must also be set as outputs.

19.1.1 ECCP1/ECCP3 OUTPUTS AND PROGRAM MEMORY MODE

In 80-pin devices, the use of Extended Microcontroller mode has an indirect effect on the use of ECCP1 and ECCP3 in Enhanced PWM modes. By default, PWM outputs, P1B/P1C and P3B/P3C, are multiplexed to PORTE pins along with the high-order byte of the External Memory Bus. When the bus is active in Extended Microcontroller mode, it overrides the Enhanced CCP outputs and makes them unavailable. Because of this, ECCP1 and ECCP3 can only be used in compatible (single output) PWM modes when the device is in Extended Microcontroller mode and default pin configuration.

An exception to this configuration is when a 12-bit address width is selected for the external bus (EMB<1:0> Configuration bits = 01). In this case, the upper pins of PORTE continue to operate as digital I/O, even when the external bus is active. P1B/P1C and P3B/P3C remain available for use as Enhanced PWM outputs.

If an application requires the use of additional PWM outputs during enhanced microcontroller operation, the P1B/P1C and P3B/P3C outputs can be reassigned to the upper bits of PORTH. This is done by clearing the ECCPMX Configuration bit.

19.1.2 ECCP2 OUTPUTS AND PROGRAM MEMORY MODES

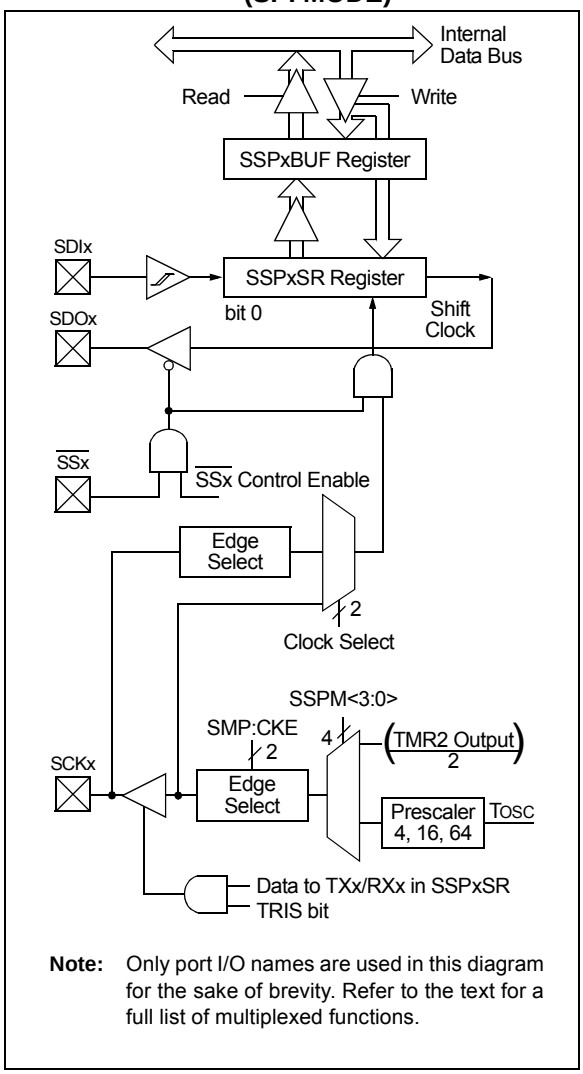
For 80-pin devices, the program memory mode of the device (**Section 6.1.3 "PIC18F8xJ11/8XJ16 Program Memory Modes"**) also impacts pin multiplexing for the module.

The ECCP2 input/output (ECCP2/P2A) can be multiplexed to one of three pins. The default assignment (CCP2MX Configuration bit is set) for all devices is RC1. Clearing CCP2MX reassigns ECCP2/P2A to RE7.

An additional option exists for 80-pin devices. When these devices are operating in Microcontroller mode, the multiplexing options described above still apply. In Extended Microcontroller mode, clearing CCP2MX reassigns ECCP2/P2A to RB3.

Changing the pin assignment of ECCP2 does not automatically change any requirements for configuring the port pin. Users must always verify that the appropriate TRIS register is configured correctly for ECCP2 operation regardless of where it is located.

FIGURE 20-1: MSSPx BLOCK DIAGRAM (SPI MODE)



Note: Only port I/O names are used in this diagram for the sake of brevity. Refer to the text for a full list of multiplexed functions.

20.3.1 REGISTERS

Each MSSP module has four registers for SPI mode operation. These are:

- MSSPx Control Register 1 (SSPxCON1)
- MSSPx Status Register (SSPxSTAT)
- Serial Receive/Transmit Buffer Register (SSPxBUF)
- MSSPx Shift Register (SSPxSR) – Not directly accessible

SSPxCON1 and SSPxSTAT are the control and status registers in SPI mode operation. The SSPxCON1 register is readable and writable. The lower 6 bits of the SSPxSTAT are read-only. The upper two bits of the SSPxSTAT are read/write.

SSPxSR is the shift register used for shifting data in or out. SSPxBUF is the buffer register to which data bytes are written to or read from.

In receive operations, SSPxSR and SSPxBUF together create a double-buffered receiver. When SSPxSR receives a complete byte, it is transferred to SSPxBUF and the SSPxIF interrupt is set.

During transmission, the SSPxBUF is not double-buffered. A write to SSPxBUF will write to both SSPxBUF and SSPxSR.

Note: Because the SSPxBUF register is double-buffered, using read-modify-write instructions, such as BCF, COMF, etc., will not work.

Similarly, when debugging under an in-circuit debugger, performing actions that cause reads of SSPxBUF (mouse hovering, watch, etc.) can consume data that the application code was expecting to receive.

2007-2012 Microchip Technology Inc.

FIGURE 21-8: AUTO-WAKE-UP BIT (WUE) TIMINGS DURING NORMAL OPERATION

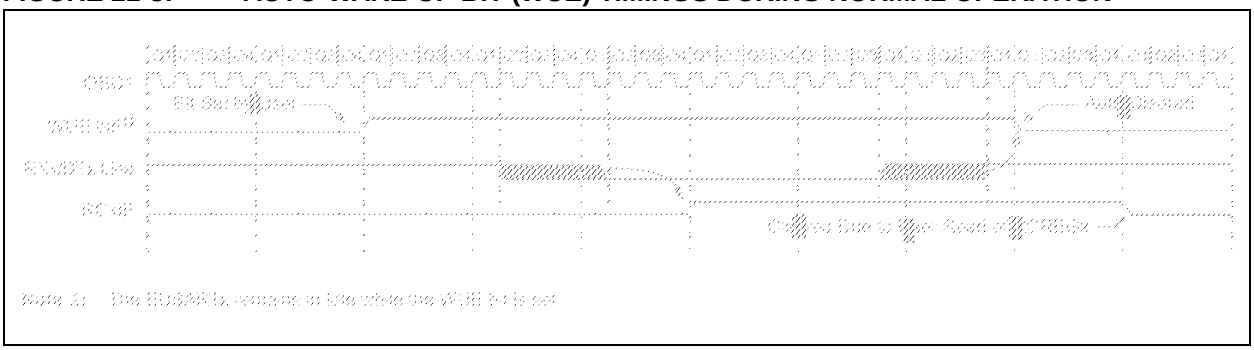


FIGURE 21-9: AUTO-WAKE-UP BIT (WUE) TIMINGS DURING SLEEP

REGISTER 23-1: CMxCON: COMPARATORx CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
CON	COE	CPOL	EVPOL1	EVPOL0	CREF	CCH1	CCH0
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 **CON:** Comparator Enable bit
1 = Comparator is enabled
0 = Comparator is disabled

bit 6 **COE:** Comparator Output Enable bit
1 = Comparator output is present on the CxOUT pin
0 = Comparator output is internal only

bit 5 **CPOL:** Comparator Output Polarity Select bit
1 = Comparator output is inverted
0 = Comparator output is not inverted

bit 4-3 **EVPOL<1:0>:** Interrupt Polarity Select bits
11 = Interrupt generation on any change of the output(1)
10 = Interrupt generation only on high-to-low transition of the output
01 = Interrupt generation only on low-to-high transition of the output
00 = Interrupt generation is disabled

bit 2 **CREF:** Comparator Reference Select bit (non-inverting input)
1 = Non-inverting input connects to internal CVREF voltage
0 = Non-inverting input connects to CxINA pin

bit 1-0 **CCH<1:0>:** Comparator Channel Select bits
11 = Inverting input of comparator connects to VIRV
10 = Inverting input of comparator connects to CxIND pin(2)
01 = Inverting input of comparator connects to CxINC pin(2)
00 = Inverting input of comparator connects to CxINB pin

Note 1: The CMxIF bit is automatically set any time this mode is selected and must be cleared by the application after the initial configuration.

2: Available in 80-pin devices only.

2007-2012 Microchip Technology Inc.

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 two-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>) 2nd word(k<19:8>)	1110 | 110s | $k_7$kkk | kkkk $_0$ 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. Then, the 20-bit value ‘k’ is loaded into PC<20:1>. CALL is a two-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

2007-2012 Microchip Technology Inc.

GOTO	Unconditional Branch
Syntax:	GOTO k
Operands:	0 k 1048575
Operation:	k PC<20:1>
Status Affected:	None

Encoding:

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

Description: GOTO allows an unconditional branch anywhere within entire 2-Mbyte memory range. The 20-bit value 'k' is loaded into PC<20:1>. GOTO is always a two-cycle instruction.

Words: 2

Cycles: 2

Q Cycle Activity:

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

Example: GOTO THERE

After Instruction
PC = Address (THERE)

INCF	Increment f
Syntax:	INCF f {,d {,a}}
Operands:	0 f 255 d [0,1] a [0,1]
Operation:	(f) + 1 dest
Status Affected:	C, DC, N, OV, Z

Encoding:

0010	10da	ffff	ffff

Description: The contents of register 'f' are incremented. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is placed back in register 'f'.

If 'a' is '0', the Access Bank is selected. If 'a' is '1', the BSR is used to select the GPR bank.

If 'a' is '0' and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever f 95 (5Fh). See **Section 26.2.3 "Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode"** for details.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	Write to destination

Example: INCF CNT, 1, 0

Before Instruction
CNT = FFh
Z = 0
C = ?
DC = ?

After Instruction
CNT = 00h
Z = 1
C = 1
DC = 1

2007-2012 Microchip Technology Inc.

MOVLW **Move Literal to W**

Syntax:	MOVLW k
Operands:	0 k 255
Operation:	k W
Status Affected:	None
Encoding:	0000 | 1110 | kkkk | kkkk
Description:	The eight-bit literal ‘k’ is loaded into W.
Words:	1
Cycles:	1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read literal ‘k’	Process Data	Write to W

Example: MOVLW 5Ah

After Instruction
W = 5Ah

MOVWF **Move W to f**

Syntax:	MOVWF f {,a}
Operands:	0 f 255 a [0,1]
Operation:	(W) f
Status Affected:	None
Encoding:	0110 | 111a | ffff | ffff
Description:	Move data from W to register ‘f’. Location ‘f’ can be anywhere in the 256-byte bank. If ‘a’ is ‘0’, the Access Bank is selected. If ‘a’ is ‘1’, the BSR is used to select the GPR bank. If ‘a’ is ‘0’ and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever f 95 (5Fh). See **Section 26.2.3 “Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode”** for details.
Words:	1
Cycles:	1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register ‘f’	Process Data	Write register ‘f’

Example: MOVWF REG, 0

Before Instruction
W = 4Fh
REG = FFh
After Instruction
W = 4Fh
REG = 4Fh