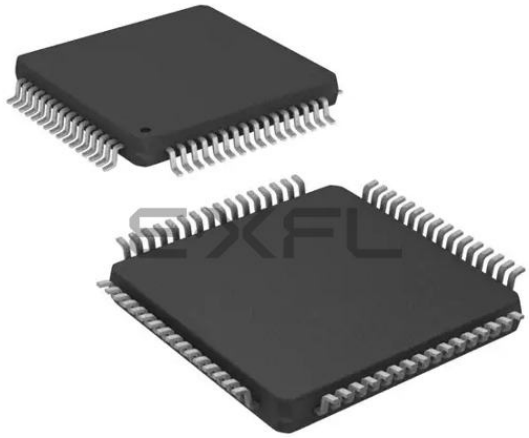


Welcome to E-XFL.COM

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	64MHz
Connectivity	I²C, IrDA, LINbus, SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, HLVD, LCD, POR, PWM, WDT
Number of I/O	52
Program Memory Size	64KB (32K x 16)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 3.6V
Data Converters	A/D 16x10b/12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-TQFP
Supplier Device Package	64-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f66j94t-i-pt

1.1.3 MEMORY OPTIONS

The PIC18F9XJ94 family provides ample room for application code, from 32 Kbytes to 128 Kbytes of code space. The Flash cells for program memory are rated to last up to 20,000 erase/write cycles. Data retention without refresh is conservatively estimated to be greater than 10 years.

The Flash program memory is readable and writable. During normal operation, the PIC18F9XJ94 family also provides plenty of room for dynamic application data with up to 3,578 bytes of data RAM.

1.1.4 UNIVERSAL SERIAL BUS (USB)

Devices in the PIC18F9XJ94 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.

1.1.5 EXTERNAL MEMORY BUS

Should 128 Kbytes of memory be inadequate for an application, the 80-pin and 100-pin members of the PIC18F9XJ94 family have an External Memory Bus (EMB), enabling the controller's internal Program Counter to address a memory space of up to 2 Mbytes. This is a level of data access that few 8-bit devices can claim and enables:

- Using combinations of on-chip and external memory of up to 2 Mbytes
- Using external Flash memory for reprogrammable application code or large data tables
- Using external RAM devices for storing large amounts of variable data

1.1.6 EXTENDED INSTRUCTION SET

The PIC18F9XJ94 family implements the optional extension to the PIC18 instruction set, adding eight new instructions and an Indexed Addressing mode. Enabled as a device configuration option, the extension has been specifically designed to optimize re-entrant application code originally developed in high-level languages, such as 'C'.

1.1.7 EASY MIGRATION

All devices share the same rich set of peripherals. This provides a smooth migration path within the device family as applications evolve and grow.

The consistent pinout scheme, used throughout the entire family, also aids in migrating to the next larger device. This is true when moving between the 64-pin members, between the 80-pin members, between the 100-pin members or even jumping from 64-pin to 80-pin to 100-pin devices.

The PIC18F9XJ94 family is also largely pin compatible with other PIC18 families, such as the PIC18F87J90, PIC18F87J11 and the PIC18F87J50. This allows a new dimension to the evolution of applications, allowing developers to select different price points within Microchip's PIC18 portfolio, while maintaining a similar feature set.

1.2 LCD Controller

The on-chip LCD driver includes many features that make the integration of displays in low-power applications easier. These include an integrated voltage regulator with charge pump and an integrated internal resistor ladder that allows contrast control in software and display operation above device V_{DD}.

TABLE 1-4: PIC18FXXJ94 PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name	Pin Number			Pin Type	Buffer Type	Description
	100	80	64			
SEG56/RK0	46					
SEG56				O	Analog	SEG56 output for LCD.
RK0				I/O	ST/DIG	General purpose I/O pin.
SEG57/RK1	55					
SEG57				O	Analog	SEG57 output for LCD.
RK1				I/O	ST/DIG	General purpose I/O pin.
SEG58/RK2	60					
SEG58				O	Analog	SEG58 output for LCD.
RK2				I/O	ST/DIG	General purpose I/O pin.
SEG59/RK3	63					
SEG59				O	Analog	SEG59 output for LCD.
RK3				I/O	ST/DIG	General purpose I/O pin.
SEG60/RK4	66					
SEG60				O	Analog	SEG60 output for LCD.
RK4				I/O	ST/DIG	General purpose I/O pin.
SEG61/RK5	71					
SEG61				O	Analog	SEG61 output for LCD.
RK5				I/O	ST/DIG	General purpose I/O pin.
SEG62/RK6	80					
SEG62				O	Analog	SEG62 output for LCD.
RK6				I/O	ST/DIG	General purpose I/O pin.
SEG63/RK7	85					
SEG63				O	Analog	SEG63 output for LCD.
RK7				I/O	ST/DIG	General purpose I/O pin.

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

REGISTER 3-2: OSCCON2: OSCILLATOR CONTROL REGISTER 2

R/W-0	R/W-0	R-0	U-0	R/C-0	R/W-0	R/W-0	U-0
CLKLOCK(2)	IOLOCK(1)	LOCK	—	CF	POSCEN	SOSCGO	—
bit 7							bit 0

Legend:	C = Clearable bit		
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 **CLKLOCK:** Clock Lock Enabled bit(2)
- 1 = Clock and PLL selection are locked and may not be modified
- 0 = Clock and PLL selection are not locked, configurations may be modified

bit 6 **IOLOCK:** I/O Lock Enable bit(1)
- 1 = I/O lock is active (If IOL1WAY (CONFIG5H<0> = 1), the bit cannot be cleared, once it is set, except on a device Reset.)
- 0 = I/O lock is not active

bit 5 **LOCK:** PLL Lock Status bit (read-only)
- 1 = Indicates that PLL module is in lock or PLL start-up timer is satisfied
- 0 = Indicates that PLL module is out of lock, PLL start-up timer is in progress or PLL is disabled

bit 4 **Unimplemented:** Read as '0'

bit 3 **CF:** Clock Fail Detect bit (readable/clearable by application)
- 1 = FSCM has detected a clock failure
- 0 = FSCM has not detected A clock failure

bit 2 **POSCEN:** Primary Oscillator (POSC) Enable bit
- 1 = Enables Primary Oscillator in Sleep mode
- 0 = Disables Primary Oscillator in Sleep mode

bit 1 **SOSCGO:** 32 kHz Secondary (LP) Oscillator Enable bit
- 1 = Enables Secondary Oscillator independent of other SOSC enable requests; provides a way to keep the SOSC running even when not actively used by the system
- 0 = Disables Secondary Oscillator; the SOSC will be enabled if directly requested by the system. Reset on POR or BOR only.

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

Note 1: The IOLOCK bit cannot be cleared once it has been set, provided that the IOL1WAY (CONFIG5H<0>) = 1.

2: If the user wants to change the clock source, ensure that the FSCM<1:0> bits (CONFIG3L<5:4>) are set appropriately.

TABLE 4-3: DELAY TIMES FOR EXITING FROM SLEEP MODE

Clock Source		Exit Delay	Oscillator Delay	Notes
EC		TPM	—	**1**
ECPLL		TPM	TLOCK	**1, 3**
MS, HS		TPM	TOST	**1, 2**
MSPLL, HSPLL		TPM	TOST + TLOCK	**1, 2, 3**
SOSC	(Off during Sleep)	TPM	TOST	**1, 2**
	(On during Sleep)	TPM	—	**1**
FRC, FRCDIV		TPM	TFRC	**1, 4**
LPRC	(Off during Sleep)	TPM	TLPRC	**1, 4**
	(On during Sleep)	TPM	—	**1**
FRCPLL		TPM	TLOCK	**1, 3**

Note 1: TPM = Start-up delay for program memory stabilization.

2: TOST = Oscillator Start-up Timer (OST); a delay of 1024 oscillator periods before the oscillator clock is released to the system.

3: TLOCK = PLL lock time.

4: TFRC and TLPRC are RC Oscillator start-up times.

TABLE 4-4: DELAY TIMES FOR EXITING FROM RETENTION SLEEP MODE

Clock Source		Exit Delay	Oscillator Delay	Notes
EC		TRETR + TPM	—	**1, 2**
ECPLL		TRETR + TPM	TLOCK	**1, 2, 4**
MS, HS		TRETR + TPM	TOST	**1, 2, 3**
MSPLL, HSPLL		TRETR + TPM	TOST + TLOCK	**1, 2, 3, 4**
SOSC	(Off during Sleep)	TRETR + TPM	TOST	**1, 2, 3**
	(On during Sleep)	TRETR + TPM	—	**1, 2**
FRC, FRCDIV		TRETR + TPM	TFRC	**1, 2, 5**
LPRC	(Off during Sleep)	TRETR + TPM	TLPRC	**1, 2, 5**
	(On during Sleep)	TRETR + TPM	—	**1, 2**
FRCPLL		TRETR + TPM	TLOCK	**1, 2, 4**

Note 1: TRETR = Retention regulator start-up delay.

2: TPM = Start-up delay for program memory stabilization; applicable only when IPEN (RCON<7>) = 0.

3: TOST = Oscillator Start-up Timer; a delay of 1024 oscillator periods before the oscillator clock is released to the system.

4: TLOCK = PLL lock time.

5: TFRC and TLPRC are RC Oscillator start-up times.

TABLE 5-3: INITIALIZATION CONDITIONS FOR ALL REGISTERS (CONTINUED)

Register	Applicable Devices			Power-on Reset, Brown-out Reset	$\overline{\text{MCLR}}$ Resets, WDT Reset, RESET Instruction, Stack Resets	Wake-up via WDT or Interrupt
ADCBUF22H	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF22L	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF21H	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF21L	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF20H	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF20L	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF19H	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF19L	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF18H	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF18L	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF17H	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF17L	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF16H	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF16L	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF15H	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF15L	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF14H	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF14L	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF13H	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF13L	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF12H	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF12L	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF11H	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF11L	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF10H	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF10L	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF9H	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF9L	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF8H	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF8L	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF7H	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF7L	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu
ADCBUF6H	64-pin	80-pin	100-pin	xxxx xxxx	xxxx xxxx	uuuu uuuu

Legend: u = unchanged; x = unknown; - = unimplemented bit, read as '0'; q = value depends on condition. Shaded cells indicate that conditions do not apply for the designated device.

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 5-2 for Reset value for specific condition.

5: Bits 7,6 are unimplemented on 64 and 80-pin devices.

6: If the V_{BAT} is always powered, the DSGPx register values will remain unchanged after the first POR.

6.2.3 INSTRUCTIONS IN PROGRAM MEMORY

The program memory is addressed in bytes. Instructions are stored as two or four bytes in program memory. The Least Significant Byte of an instruction word is always stored in a program memory location with an even address (LSB = 0). To maintain alignment with instruction boundaries, the PC increments in steps of two and the LSB will always read '0' (see **Section 6.1.3 "Program Counter"**).

Figure 6-5 shows an example of how instruction words are stored in the program memory.

The CALL and GOTO instructions have the absolute program memory address embedded into the instruction. Since instructions are always stored on word boundaries, the data contained in the instruction is a word address. The word address is written to PC<20:1> which accesses the desired byte address in program memory. Instruction #2 in Figure 6-5 shows how the instruction, GOTO 0006h, is encoded in the program memory. Program branch instructions, which encode a relative address offset, operate in the same manner. The offset value stored in a branch instruction represents the number of single-word instructions that the PC will be offset by. For more details on the instruction set, see **Section 29.0 "Instruction Set Summary"**.

FIGURE 6-5: INSTRUCTIONS IN PROGRAM MEMORY

			LSB = 1	LSB = 0	Word Address ↓
Program Memory Byte Locations →					000000h
					000002h
					000004h
					000006h
Instruction 1:	MOVLW	055h	0Fh	55h	000008h
Instruction 2:	GOTO	0006h	EFh	06h	00000Ah
			F0h	00h	00000Ch
Instruction 3:	MOVFF	123h, 456h	C1h	23h	00000Eh
			F4h	56h	000010h
					000012h
					000014h

6.2.4 TWO-WORD INSTRUCTIONS

The standard PIC18 instruction set has four, two-word instructions: CALL, MOVFF, GOTO and LSFR. In all cases, the second word of the instructions always has '1111' as its four Most Significant bits. The other 12 bits are literal data, usually a data memory address.

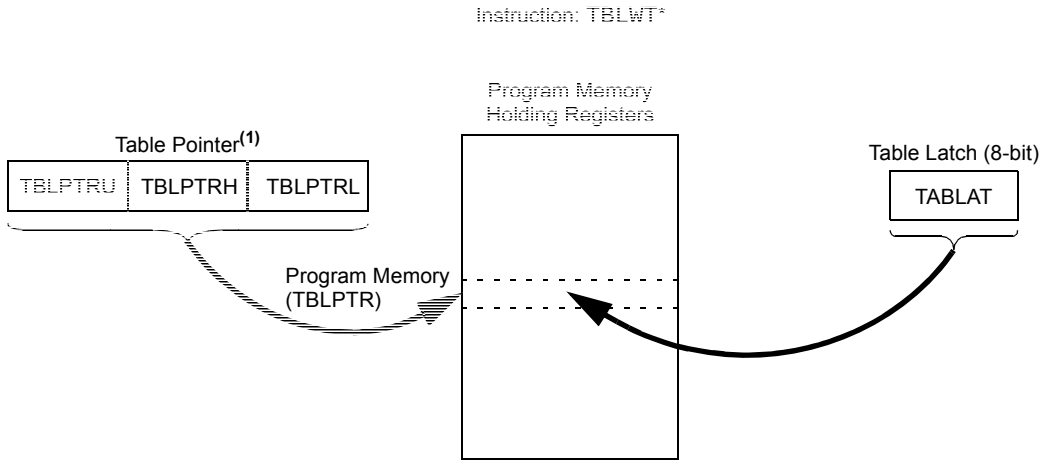
The use of '1111' in the 4 MSbs of an instruction specifies a special form of NOP. If the instruction is executed in proper sequence, immediately after the first word, the data in the second word is accessed and used by the instruction sequence. If the first word is skipped, for some reason, and the second word is executed by itself, a NOP is executed instead. This is necessary for cases when the two-word instruction is preceded by a conditional instruction that changes the PC. Example 6-4 shows how this works.

Note: For information on two-word instructions in the extended instruction set, see **Section 6.5 "Program Memory and the Extended Instruction Set"**.

EXAMPLE 6-4: TWO-WORD INSTRUCTIONS

```
CASE 1:
Object Code          Source Code
0110 0110 0000 0000  TSTFSZ   REG1       ; is RAM location 0?
1100 0001 0010 0011  MOVFF    REG1, REG2 ; No, skip this word
1111 0100 0101 0110                      ; Execute this word as a NOP
0010 0100 0000 0000  ADDWF    REG3       ; continue code
CASE 2:
Object Code          Source Code
0110 0110 0000 0000  TSTFSZ   REG1       ; is RAM location 0?
1100 0001 0010 0011  MOVFF    REG1, REG2 ; Yes, execute this word
1111 0100 0101 0110                      ; 2nd word of instruction
0010 0100 0000 0000  ADDWF    REG3       ; continue code
```

FIGURE 7-2: TABLE WRITE OPERATION



Note 1: The Table Pointer actually points to one of 64 holding registers; the address of which is determined by TBLPTRL<5:0>. The process for physically writing data to the program memory array is discussed in Section 7.5 "Writing to Flash Program Memory".

7.2 Control Registers

Several control registers are used in conjunction with the TBLRD and TBLWT instructions. These include:

- EECON1 register
- EECON2 register
- TABLAT register
- TBLPTR registers

7.2.1 EECON1 AND EECON2 REGISTERS

The EECON1 register (Register 7-1) is the control register for memory accesses. The EECON2 register is not a physical register; it is used exclusively in the memory write and erase sequences. Reading EECON2 will read all '0's.

The WWPROG bit, when set, will allow programming two bytes per word on the execution of the WR command. If this bit is cleared, the WR command will result in programming on a block of 64 bytes.

The FREE bit, when set, will allow a program memory erase operation. When FREE is set, the erase operation is initiated on the next WR command. When FREE is clear, only writes are enabled.

The WREN bit, when set, will allow a write operation. On power-up, the WREN bit is clear. The WRERR bit is set in hardware when the WR bit is set, and cleared when the internal programming timer expires and the write operation is complete.

Note:	During normal operation, the WRERR is read as '1'. This can indicate that a write operation was prematurely terminated by a Reset or a write operation was attempted improperly.

8.7 8-Bit Data Width Mode

In 8-Bit Data Width mode, the External Memory Bus operates only in Multiplexed mode; that is, data shares the 8 Least Significant bits of the address bus.

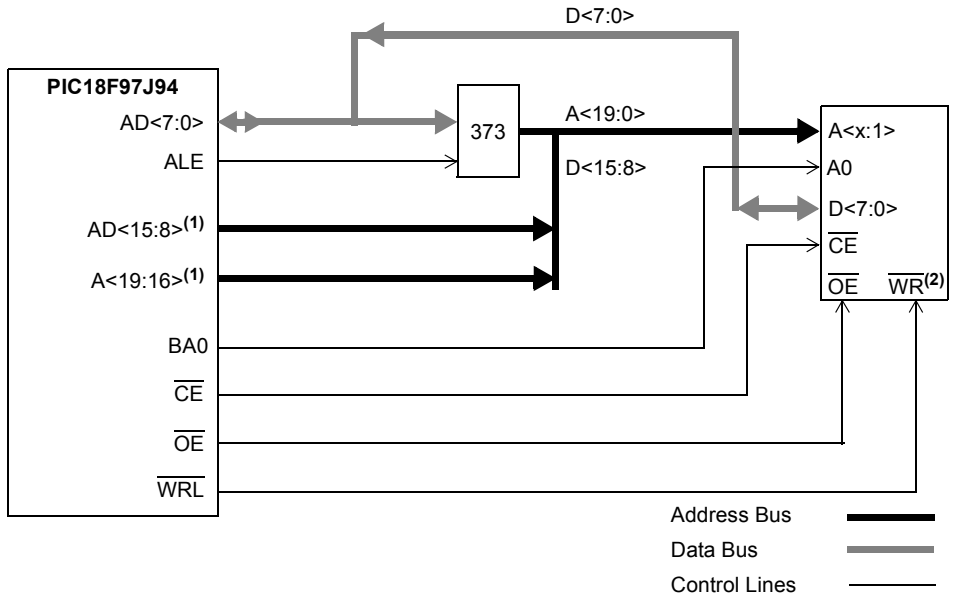
Figure 8-6 shows an example of 8-Bit Multiplexed mode for 100-pin devices. This mode is used for a single, 8-bit memory, connected for 16-bit operation. The instructions will be fetched as two 8-bit bytes on a shared data/address bus. The two bytes are sequentially fetched within one instruction cycle (TCY). Therefore, the designer must choose external memory devices, according to timing calculations based on 1/2 TCY (2 times the instruction rate). For proper memory speed selection, glue logic propagation delay times must be considered, along with setup and hold times.

The Address Latch Enable (ALE) pin indicates that the address bits, AD<15:0>, are available on the External Memory Bus interface. The Output Enable ($\overline{OE}$) signal will enable one byte of program memory for a portion of the instruction cycle, then BA0 will change and the second byte will be enabled to form the 16-bit instruction word. The Least Significant bit of the address, BA0, must be connected to the memory devices in this mode. The Chip Enable ($\overline{CE}$) signal is active at any time that the microcontroller accesses external memory, whether reading or writing. It is inactive (asserted high) whenever the device is in Sleep mode.

This generally includes basic EPROM and Flash devices. It allows table writes to byte-wide external memories.

During a TBLWT instruction cycle, the TABLAT data is presented on the upper and lower bytes of the AD<15:0> bus. The appropriate level of the BA0 control line is strobed on the LSb of the TBLPTR.

FIGURE 8-6: 8-BIT MULTIPLEXED MODE EXAMPLE



Note 1: Upper order address bits are only used for 20-bit address width. The upper AD byte is used for all address widths except 8-bit.

2: This signal only applies to table writes. See **Section 7.1 "Table Reads and Table Writes"**.

11.9 PORTH, LATH and TRISH Registers

Note: PORTH is available only on 80-pin and 100-pin devices.

PORTH is an 8-bit wide, bidirectional I/O port. The corresponding Data Direction and Output Latch registers are TRISH and LATH.

All pins on PORTH are implemented with Schmitt Trigger input buffers. Each pin is individually configurable as an input or output.

EXAMPLE 11-8: INITIALIZING PORTH

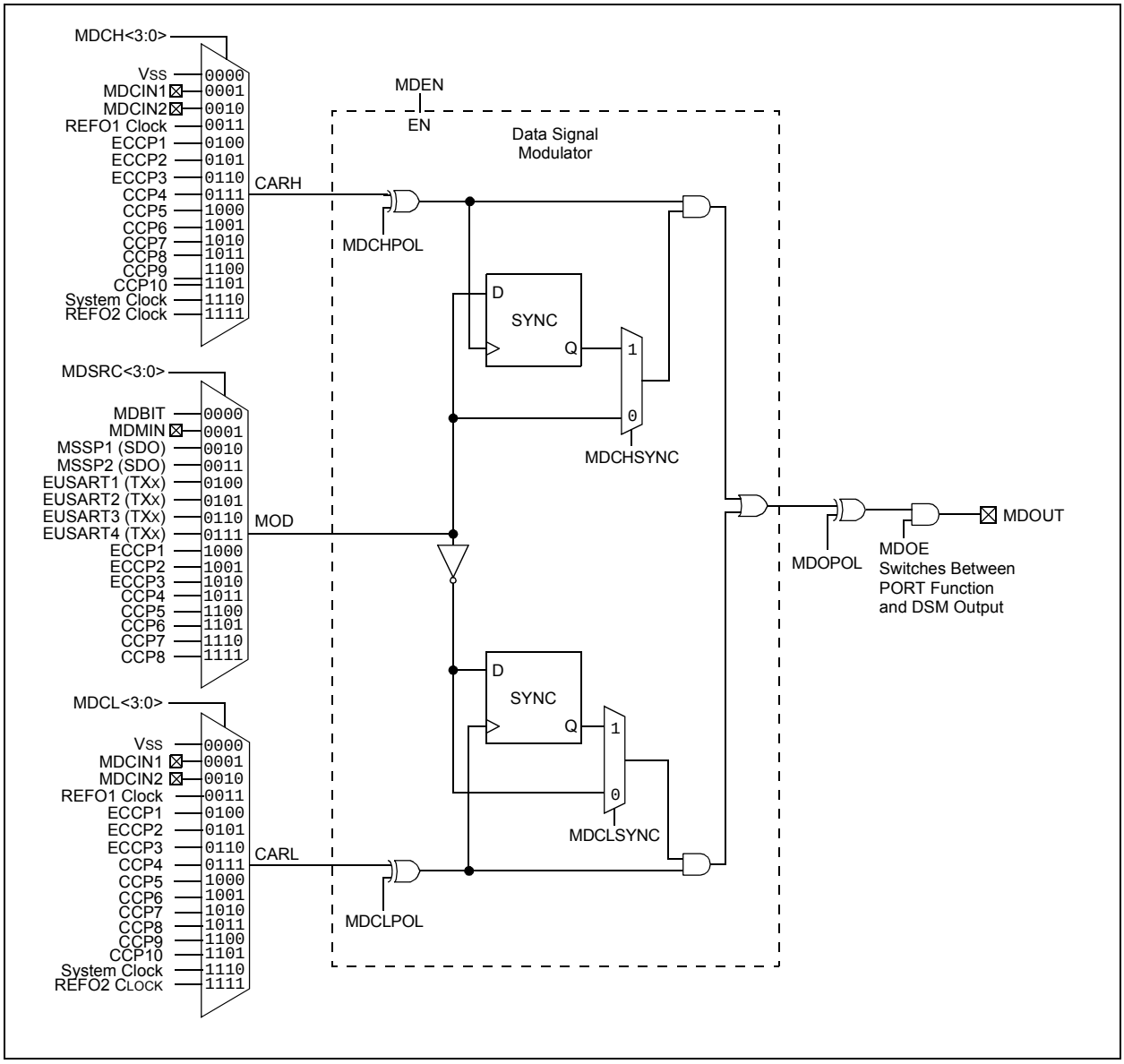
```
CLRF     PORTH    ; Initialize PORTH by
                  ; clearing output
                  ; data latches
CLRF     LATH     ; Alternate method
                  ; to clear output
                  ; data latches
BANKSEL  ANCON2   ; Select bank with ANCON2 register
MOVLW    0Fh      ; Configure PORTH as
MOVWF    ANCON2   ; digital I/O
MOVLW    0Fh      ; Configure PORTH as
MOVWF    ANCON1   ; digital I/O
BANKSEL  TRISH    ; Select bank with TRISH register
MOVLW    0CFh     ; Value used to
                  ; initialize data
                  ; direction
MOVWF    TRISH    ; Set RH3:RH0 as inputs
                  ; RH5:RH4 as outputs
                  ; RH7:RH6 as inputs
```

TABLE 11-8: PORTH FUNCTIONS

Pin Name	Function	TRIS Setting	I/O	I/O Type	Description
RH0/AN23/ SEG47/A16	RH0	0	O	DIG	LATH<0> data output; not affected by analog input.
		1	I	ST	PORTH<0> data input.
	AN23	1	I	ANA	A/D Input Channel 23. Default input configuration on POR; does not affect digital output.
	SEG47	0	O	ANA	LCD Segment 47 output; disables all other pin functions.
	A16	x	O	DIG	External Memory Bus Address<16> output.
RH1/AN22/ SEG46/A17	RH1	0	O	DIG	LATH<1> data output; not affected by analog input.
		1	I	ST	PORTH<1> data input.
	AN22	1	I	ANA	A/D Input Channel 22. Default input configuration on POR; does not affect digital output.
	SEG46	0	O	ANA	LCD Segment 46 output; disables all other pin functions.
	A17	x	O	DIG	External Memory Bus Address<17> output.
RH2/AN21/ SEG45/A18	RH2	0	O	DIG	LATH<2> data output; not affected by analog input.
		1	I	ST	PORTH<2> data input.
	AN21	1	I	ANA	A/D Input Channel 21. Default input configuration on POR; does not affect digital output.
	SEG45	0	O	ANA	LCD Segment 45 output; disables all other pin functions.
	A18	x	O	DIG	External Memory Bus Address<18> output.
RH3/AN20/ SEG44/A19	RH3	0	O	DIG	LATH<3> data output; not affected by analog input.
		1	I	ST	PORTH<3> data input.
	AN20	1	I	ANA	A/D Input Channel 20. Default input configuration on POR; does not affect digital output.
	SEG44	0	O	ANA	LCD Segment 44 output; disables all other pin functions.
	A19	x	O	DIG	External Memory Bus Address<19> output.
RH4/C2INC/ AN12/SEG40	RH4	0	O	DIG	LATH<4> data output; not affected by analog input.
		1	I	ST	PORTH<4> data input; disabled when analog input is enabled.
	C2INC	1	I	ANA	Comparator 2 Input C.
	AN12	1	I	ANA	A/D Input Channel 12. Default input configuration on POR; does not affect digital output.
	SEG40	0	O	ANA	LCD Segment 40 output; disables all other pin functions.

Legend: O = Output, I = Input, ANA = Analog Signal, DIG = Digital Output, ST = Schmitt Trigger Buffer Input, x = Don't care (TRIS bit does not affect port direction or is overridden for this option).

FIGURE 12-1: SIMPLIFIED BLOCK DIAGRAM OF THE DATA SIGNAL MODULATOR



12.1 DSM Operation

The DSM module can be enabled by setting the MDEN bit in the MDCON register. Clearing the MDEN bit in the MDCON register disables the DSM module by automatically switching the Carrier High and Carrier Low signals to the VSS signal source. The Modulator signal source is also switched to the MDBIT in the MDCON register. This not only assures that the DSM module is inactive, but that it is also consuming the least amount of current.

The Modulation Carrier High and Modulation Carrier Low Control registers are not affected when the MDEN bit is cleared, and the DSM module is disabled. The values inside these registers remain unchanged while the DSM is inactive. The sources for the Carrier High, Carrier Low and Modulator signals will once again be selected when the MDEN bit is set, and the DSM module is again enabled and active.

The modulated output signal can be disabled without shutting down the DSM module. The DSM module will remain active and continue to mix signals, but the output value will not be sent to the MDOUT pin. During the time that the output is disabled, the MDOUT pin will remain low. The modulated output can be disabled by clearing the MDOE bit in the MDCON register.

12.2 Modulator Signal Sources

The Modulator signal can be supplied from the following sources:

- ECCP1 Signal
- ECCP2 Signal
- ECCP3 Signal
- CCP2 Signal
- CCP3 Signal
- CCP4 Signal
- CCP5 Signal
- CCP6 Signal
- CCP7 Signal
- CCP8 Signal
- MSSP1 SDO Signal (SPI mode only)
- MSSP2 SDO Signal (SPI mode only)
- EUSART1 TX1 Signal
- EUSART2 TX2 Signal
- EUSART3 TX3 Signal
- EUSART4 TX4 Signal
- External Signal on MDMIN Pin (RF0/MDMIN)
- MDBIT bit in the MDCON Register

The Modulator signal is selected by configuring the MDSRC<3:0> bits in the MDSRC register.

12.3 Carrier Signal Sources

The Carrier High signal and Carrier Low signal can be supplied from the following sources:

- ECCP1 Signal
- ECCP2 Signal
- ECCP3 Signal
- CCP5 Signal
- CCP6 Signal
- CCP7 Signal
- CCP8 Signal
- CCP9 Signal
- CCP10 Signal
- Reference Clock Output Module Signal (REFO1)
- Reference Clock Output Module Signal (REFO2)
- System Clock
- External Signals on the MDCIN1 and MDCIN2 pins are available though PPS. Refer to **Section 11.15 “PPS-Lite”** for setup.
- VSS

The Carrier High signal is selected by configuring the MDCH<3:0> bits in the MDCARH register. The Carrier Low signal is selected by configuring the MDCL<3:0> bits in the MDCARL register.

12.4 Carrier Synchronization

During the time when the DSM switches between Carrier High and Carrier Low signal sources, the carrier data in the modulated output signal can become truncated. To prevent this, the carrier signal can be synchronized to the Modulator signal. When synchronization is enabled, the carrier pulse that is being mixed at the time of the transition is allowed to transition low before the DSM switches over to the next carrier source.

Synchronization is enabled separately for the Carrier High and Carrier Low signal sources. Synchronization for the Carrier High signal can be enabled by setting the MDCHSYNC bit in the MDCARH register. Synchronization for the Carrier Low signal can be enabled by setting the MDCLSYNC bit in the MDCARL register.

Figure 12-1 through Figure 12-6 show timing diagrams using various synchronization methods.

20.5.2 OPERATION

The MSSPx module functions are enabled by setting the MSSPx Enable bit, SSPEN (SSPxCON1<5>).

The SSPxCON1 register allows control of the I^2C operation. Four mode selection bits (SSPxCON1<3:0>) allow one of the following I^2C modes to be selected:

- I^2C Master mode, clock
- I^2C Slave mode (7-bit address)
- I^2C Slave mode (10-bit address)
- I^2C Slave mode (7-bit address) with Start and Stop bit interrupts enabled
- I^2C Slave mode (10-bit address) with Start and Stop bit interrupts enabled
- I^2C Firmware Controlled Master mode, slave is Idle

Selection of any I^2C mode, with the SSPEN bit set, forces the SCLx and SDAx pins to be open-drain, provided these pins are programmed as inputs by setting the appropriate TRISC or TRISD bits. To ensure proper operation of the module, pull-up resistors must be provided externally to the SCLx and SDAx pins.

20.5.3 SLAVE MODE

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

The I^2C Slave mode hardware will always generate an interrupt on an address match. Address masking will allow the hardware to generate an interrupt for more than one address (up to 31 in 7-bit addressing and up to 63 in 10-bit addressing). Through the mode select bits, the user can also choose to interrupt on Start and Stop bits.

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

Any combination of the following conditions will cause the MSSPx module not to give this $\overline{ACK}$ pulse:

- The Buffer Full bit, BF (SSPxSTAT<0>), was set before the transfer was received.
- The overflow bit, SSPOV (SSPxCON1<6>), was set before the transfer was received.

In this case, the SSPxSR register value is not loaded into the SSPxBUF, but bit, SSPxIF, is set. The BF bit is cleared by reading the SSPxBUF register, while bit, SSPOV, is cleared through software.

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

20.5.4 ADDRESSING

Once the MSSPx module has been enabled, it waits for a Start condition to occur. Following the Start condition, the 8 bits are shifted into the SSPxSR register. All incoming bits are sampled with the rising edge of the clock (SCLx) line. The value of register, SSPxSR<7:1>, is compared to the value of the SSPxADD register. The address is compared on the falling edge of the eighth clock (SCLx) pulse. If the addresses match, and the BF and SSPOV bits are clear, the following events occur:

1. The SSPxSR register value is loaded into the SSPxBUF register.
2. The Buffer Full bit, BF, is set.
3. An $\overline{ACK}$ pulse is generated.
4. The MSSPx Interrupt Flag bit, SSPxIF, is set (and interrupt is generated if enabled) on the falling edge of the ninth SCLx pulse.

In 10-Bit Addressing mode, two address bytes need to be received by the slave. The five Most Significant bits (MSbs) of the first address byte specify if this is a 10-bit address. The R/$\overline{W}$ (SSPxSTAT<2>) bit must specify a write so the slave device will receive the second address byte. For a 10-bit address, the first byte would equal '11110 A9 A8 0', where 'A9' and 'A8' are the two MSbs of the address. The sequence of events for 10-bit addressing is as follows, with Steps 7 through 9 for the slave-transmitter:

1. Receive first (high) byte of address (bits, SSPxIF, BF and UA, are set on address match).
2. Update the SSPxADD register with second (low) byte of address (clears bit, UA, and releases the SCLx line).
3. Read the SSPxBUF register (clears bit, BF) and clear flag bit, SSPxIF.
4. Receive second (low) byte of address (bits, SSPxIF, BF and UA, are set).
5. Update the SSPxADD register with the first (high) byte of address. If match releases SCLx line, this will clear bit, UA.
6. Read the SSPxBUF register (clears bit, BF) and clear flag bit SSPxIF.
7. Receive Repeated Start condition.
8. Receive first (high) byte of address (bits, SSPxIF and BF, are set).
9. Read the SSPxBUF register (clears bit, BF) and clear flag bit, SSPxIF.

25.0 HIGH/LOW-VOLTAGE DETECT (HLVD)

The PIC18FXXJ94 of devices has a High/Low-Voltage Detect module (HLVD). This is a programmable circuit that sets both a device voltage trip point and the direction of change from that point. If the device experiences an excursion past the trip point in that direction, an interrupt flag is set. If the interrupt is enabled, the program execution branches to the interrupt vector address and the software responds to the interrupt.

The High/Low-Voltage Detect Control register (Register 25-1) completely controls the operation of the HLVD module. This allows the circuitry to be “turned off” by the user under software control, which minimizes the current consumption for the device.

The module’s block diagram is shown in Figure 25-1.

REGISTER 25-1: HLVDCON: HIGH/LOW-VOLTAGE DETECT CONTROL REGISTER

R/W-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
VDIRMAG	BGVST	IRVST	HLVDEN	HLVDL3(1)	HLVDL2(1)	HLVDL1(1)	HLVDL0(1)
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 **VDIRMAG:** Voltage Direction Magnitude Select bit
1 = Event occurs when voltage equals or exceeds trip point (HLVDL<3:0>)
0 = Event occurs when voltage equals or falls below trip point (HLVDL<3:0>)

bit 6 **BGVST:** Band Gap Reference Voltages Stable Status Flag bit
1 = Internal band gap voltage references are stable
0 = Internal band gap voltage references are not stable

bit 5 **IRVST:** Internal Reference Voltage Stable Flag bit
1 = Indicates that the voltage detect logic will generate the interrupt flag at the specified voltage range
0 = Indicates that the voltage detect logic will not generate the interrupt flag at the specified voltage range and the HLVD interrupt should not be enabled

bit 4 **HLVDEN:** High/Low-Voltage Detect Power Enable bit
1 = HLVD is enabled
0 = HLVD is disabled

bit 3-0 **HLVDL<3:0>:** Voltage Detection Limit bits(1)
1111 = External analog input is used (input comes from the HLVDIN pin)
1110 = Maximum setting
.
.
.
0100 = Minimum setting

Note 1: For the electrical specifications, see Parameter D420.

26.2.5 INTERRUPTS

The CTMU sets its interrupt flag (PIR3<3>) whenever the current source is enabled, then disabled. An interrupt is generated only if the corresponding interrupt enable bit (PIE3<3>) is also set. If edge sequencing is not enabled (i.e., Edge 1 must occur before Edge 2), it is necessary to monitor the Edge Status bits, and determine which edge occurred last and caused the interrupt.

26.3 CTMU Module Initialization

The following sequence is a general guideline used to initialize the CTMU module:

1. Select the current source range using the IRNGx bits (CTMUCON1<1:0>).
2. Adjust the current source trim using the ITRIMx bits (CTMUCON1<7:2>).
3. Configure the edge input sources for Edge 1 and Edge 2 by setting the EDG1SELx and EDG2SELx bits (CTMUCON3<5:2> and CTMUCON2<5:2>, respectively).
4. Configure the input polarities for the edge inputs using the EDG1POL and EDG2POL bits (CTMUCON3<6> and CTMUCON2<6>).

   The default configuration is for negative edge polarity (high-to-low transitions).
5. Enable edge sequencing using the EDGSEQEN bit (CTMUCON<2>).

   By default, edge sequencing is disabled.
6. Select the operating mode (Measurement or Time Delay) with the TGEN bit (CTMUCON<4>).

   The default mode is Time/Capacitance Measurement mode.
7. Configure the module to automatically trigger an A/D conversion when the second edge event has occurred using the CTTRIG bit (CTMUCON<0>).

   The conversion trigger is disabled by default.
8. Discharge the connected circuit by setting the IDISSEN bit (CTMUCON<1>).
9. After waiting a sufficient time for the circuit to discharge, clear the IDISSEN bit.
10. Disable the module by clearing the CTMUEN bit (CTMUCON<7>).
11. Clear the Edge Status bits, EDG2STAT and EDG1STAT (CTMUCON3<1:0>).

    Both bits should be cleared simultaneously, if possible, to avoid re-enabling the CTMU current source.
12. Enable both edge inputs by setting the EDGEN bit (CTMUCON<3>).
13. Enable the module by setting the CTMUEN bit.

Depending on the type of measurement or pulse generation being performed, one or more additional modules may also need to be initialized and configured with the CTMU module:

- Edge Source Generation: In addition to the external edge input pins, CCP1/CCP2 Special Event Triggers can be used as edge sources for the CTMU.
- Capacitance or Time Measurement: The CTMU module uses the A/D Converter to measure the voltage across a capacitor that is connected to one of the analog input channels.
- Pulse Generation: When generating system clock independent, output pulses, the CTMU module uses Comparator 2 and the associated comparator voltage reference.

26.4 Calibrating the CTMU Module

The CTMU requires calibration for precise measurements of capacitance and time, as well as for accurate time delay. If the application only requires measurement of a relative change in capacitance or time, calibration is usually not necessary. An example of a less precise application is a capacitive touch switch, in which the touch circuit has a baseline capacitance and the added capacitance of the human body changes the overall capacitance of a circuit.

If actual capacitance or time measurement is required, two hardware calibrations must take place:

- The current source needs calibration to set it to a precise current.
- The circuit being measured needs calibration to measure or nullify any capacitance other than that to be measured.

26.4.1 CURRENT SOURCE CALIBRATION

The current source on board the CTMU module has a range of ±62% nominal for each of three current ranges. For precise measurements, it is possible to measure and adjust this current source by placing a high-precision resistor, RCAL, onto an unused analog channel. An example circuit is shown in Figure 26-2.

To measure the current source:

1. Initialize the A/D Converter.
2. Initialize the CTMU.
3. Enable the current source by setting EDG1STAT (CTMUCON3<0>).
4. Issue time delay for voltage across RCAL to stabilize and the A/D Sample-and-Hold (S/H) capacitor to charge.
5. Perform the A/D conversion.
6. Calculate the current source current using $I = V/R_{CAL}$, where RCAL is a high-precision resistance and V is measured by performing an A/D conversion.

ADDWFC **ADD W and Carry bit to f**

Syntax: ADDWFC f {,d {,a}}

Operands: $0 \le f \le 255$
$d \in [0,1]$
$a \in [0,1]$

Operation: (W) + (f) + (C) → dest

Status Affected: N,OV, C, DC, Z

Encoding:

0010	00da	ffff	ffff

Description: Add W, the Carry flag and data memory location 'f'. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is placed in data memory location '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 $f \le 95$ (5Fh). See **Section 29.2.3 "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: `ADDWFC REG, 0, 1`

Before Instruction
Carry bit = 1
REG = 02h
W = 4Dh
After Instruction
Carry bit = 0
REG = 02h
W = 50h

ANDLW **AND Literal with W**

Syntax: ANDLW k

Operands: $0 \le k \le 255$

Operation: (W) .AND. k → W

Status Affected: N, Z

Encoding:

0000	1011	kkkk	kkkk

Description: The contents of W are ANDed with the 8-bit literal 'k'. The result is placed in W.

Words: 1

Cycles: 1

Q Cycle Activity:

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

Example: `ANDLW 05Fh`

Before Instruction
W = A3h
After Instruction
W = 03h

29.2 Extended Instruction Set

In addition to the standard 75 instructions of the PIC18 instruction set, the PIC18FXXJ94 of devices also provides an optional extension to the core CPU functionality. The added features include eight additional instructions that augment Indirect and Indexed Addressing operations and the implementation of Indexed Literal Offset Addressing for many of the standard PIC18 instructions.

The additional features of the extended instruction set are enabled by default on unprogrammed devices. Users must properly set or clear the XINST Configuration bit during programming to enable or disable these features.

The instructions in the extended set can all be classified as literal operations, which either manipulate the File Select Registers, or use them for Indexed Addressing. Two of the instructions, ADDFSR and SUBFSR, each have an additional special instantiation for using FSR2. These versions (ADDULNK and SUBULNK) allow for automatic return after execution.

The extended instructions are specifically implemented to optimize re-entrant program code (that is, code that is recursive or that uses a software stack) written in high-level languages, particularly C. Among other things, they allow users working in high-level languages to perform certain operations on data structures more efficiently. These include:

- Dynamic allocation and deallocation of software stack space when entering and leaving subroutines
- Function Pointer invocation
- Software Stack Pointer manipulation
- Manipulation of variables located in a software stack

A summary of the instructions in the extended instruction set is provided in Table 29-3. Detailed descriptions are provided in **Section 29.2.2 "Extended Instruction Set"**. The opcode field descriptions in Table 29-1 (page 566) apply to both the standard and extended PIC18 instruction sets.

Note:	The instruction set extension and the Indexed Literal Offset Addressing mode were designed for optimizing applications written in C; the user may likely never use these instructions directly in assembler. The syntax for these commands is provided as a reference for users who may be reviewing code that has been generated by a compiler.

29.2.1 EXTENDED INSTRUCTION SYNTAX

Most of the extended instructions use indexed arguments, using one of the File Select Registers and some offset to specify a source or destination register. When an argument for an instruction serves as part of Indexed Addressing, it is enclosed in square brackets ("[ ]"). This is done to indicate that the argument is used as an index or offset. The MPASM™ Assembler will flag an error if it determines that an index or offset value is not bracketed.

When the extended instruction set is enabled, brackets are also used to indicate index arguments in byte-oriented and bit-oriented instructions. This is in addition to other changes in their syntax. For more details, see **Section 29.2.3.1 "Extended Instruction Syntax with Standard PIC18 Commands"**.

Note:	In the past, square brackets have been used to denote optional arguments in the PIC18 and earlier instruction sets. In this text and going forward, optional arguments are denoted by braces ("{ }").

TABLE 29-3: EXTENSIONS TO THE PIC18 INSTRUCTION SET

Mnemonic, Operands		Description	Cycles	16-Bit Instruction Word				Status Affected
				MSb			LSb	
ADDFSR	f, k	Add Literal to FSR	1	1110	1000	ffkk	kkkk	None
ADDULNK	k	Add Literal to FSR2 and Return	2	1110	1000	11kk	kkkk	None
CALLW		Call Subroutine using WREG	2	0000	0000	0001	0100	None
MOVSF	z_s, f_d	Move z_s (source) to 1st word f_d (destination) 2nd word	2	1110 1111	1011 ffff	0zzz ffff	zzzz ffff	None
MOVSS	z_s, z_d	Move z_s (source) to 1st word z_d (destination) 2nd word	2	1110 1111	1011 xxxx	1zzz xzzz	zzzz zzzz	None
PUSHL	k	Store Literal at FSR2, Decrement FSR2	1	1110	1010	kkkk	kkkk	None
SUBFSR	f, k	Subtract Literal from FSR	1	1110	1001	ffkk	kkkk	None
SUBULNK	k	Subtract Literal from FSR2 and return	2	1110	1001	11kk	kkkk	None

TABLE 30-2: DC CHARACTERISTICS: POWER-DOWN AND SUPPLY CURRENT PIC18FXXJ94 (INDUSTRIAL)

PIC18FXXJ94 Family (Industrial)				**Standard Operating Conditions: 2V to 3.6V (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +85°C for industrial		
Param No.	**Typ.(1)**	**Max.**	**Units**	**Conditions**		
DC60	3.7	7.0	µA	-40°C	2.0V	Sleep(2)
	3.7	7.0	µA	+25°C		
	5.0	9.0	µA	+60°C		
	9.0	18	µA	+85°C		
	3.7	8.0	µA	-40°C	3.3V	
	3.7	8.0	µA	+25°C		
	5.0	11.0	µA	+60°C		
	10	20	µA	+85°C		
DC61	0.07	0.55	µA	-40°C	2.0V	Retention Sleep or Retention Deep Sleep(3)
	0.09	0.55	µA	+25°C		
	2.0	3.2	µA	+60°C		
	7.0	8.5	µA	+85°C		
	0.10	0.65	µA	-40°C	3.3V	
	0.15	0.65	µA	+25°C		
	2.0	3.5	µA	+60°C		
	7.2	9.0	µA	+85°C		
DC70	0.06	0.5	µA	-40°C	2.0V	Deep Sleep
	0.08	0.5	µA	+25°C		
	0.21	0.8	µA	+60°C		
	0.41	1.5	µA	+85°C		
	0.09	0.6	µA	-40°C	3.3V	
	0.11	0.6	µA	+25°C		
	0.42	1.2	µA	+60°C		
	0.8	4.8	µA	+85°C		
	0.4	3.0	µA	-40°C TO +85°C	0	RTCC with VBAT mode (LPRC or SOSC)(4)

Note 1: Data in the Typical column is at 3.3V, 25°C; typical parameters are for design guidance only and are not tested.

2: Retention regulator is disabled; SRETEN (RCON4<4>= 0), $\overline{\text{RETEN}}$ (CONFIG7L<0>= 1).

3: Retention regulator is enabled; SRETEN (RCON4<4> = 1), $\overline{\text{RETEN}}$ (CONFIG7L<0> = 0).

4: VBAT pin is connected to the battery and RTCC is running with VDD = 0.

TABLE 30-3: DC CHARACTERISTICS: POWER-DOWN AND SUPPLY CURRENT PIC18F97J94 FAMILY (INDUSTRIAL)

Param No.	**Device**	**Typ.**	**Max.**	**Units**	**Conditions**		
	Supply Current (IDD)						
	All Devices	22	55	µA	-40°C to +85°C	VDD = 2.0V	FOSC = 31 kHz, **RC_RUN**
		23	56	µA	-40°C to +85°C	VDD = 3.3V	
		21	54	µA	-40°C to +85°C	VDD = 2.0V	FOSC = 31 kHz, **RC_IDLE**
		22	55	µA	-40°C to +85°C	VDD = 3.3V	

TABLE 30-7: DC CHARACTERISTICS: POWER-DOWN AND SUPPLY CURRENT PIC18F97J94 FAMILY (INDUSTRIAL)

Param No.	Device	Typ.(1)	Max.	Units	Conditions		
Module Differential Currents (ΔIWDT, ΔIBOR, ΔIHLVD, ΔIDSBOR, ΔIDSWDT, ΔIOSCB, ΔIADRC, ΔILCD, ΔIUSB)							
D020 (ΔIWDT)	**Watchdog Timer**	0.4	1	µA	-40°C to +85°C	VDD = 2.0V	
		0.4	1	µA	-40°C to +85°C	VDD = 3.3V	
D021 (ΔIBOR)	**Brown-out Reset**	4	8	µA	-40°C to +85°C	VDD = 2.0V	High-Power BOR
		5	9	µA	-40°C to +85°C	VDD = 3.3V	
D022 (ΔIHLVD)	**High/Low-Voltage Detect**	4	8	µA	-40°C to +85°C	VDD = 2.0V	
		5	9	µA	-40°C to +85°C	VDD = 3.3V	
D023 (ΔIDSBOR)	**Deep Sleep BOR**	135	480	nA	-40°C to +85°C	VDD = 2.0V to 3.3V	ΔDeep Sleep BOR(2)
D024 (ΔIDSWDT)	**Deep Sleep Watchdog Timer**	290	480	nA	-40°C to +85°C	VDD = 2.0V to 3.3V	ΔDeep Sleep WDT(2)
D025 (ΔIOSCB)	**Real-Time Clock/ Calendar with Timer1 Oscillator**	0.38	1	µA	-40°C to +85°C	VDD = 2.0V	Sleep mode 32.768 kHz, T1OSCEN = 1, LPT1OSC = 0
		0.55	1	µA	-40°C to +85°C	VDD = 3.3V	
D027 (ΔILCD)	**LCD Module**	0.6	4	µA	-40°C to +85°C	VDD = 3.3V	ΔLCD External/Internal, 1/8 MUX, 1/3 Bias(2,3)
		6	30	µA	-40°C to +85°C	VDD = 2.0V	ΔLCD Charge Pump, 1/8 MUX, 1/3 Bias(2,4)
		7	40	µA	-40°C to +85°C	VDD = 3.3V	
D028 (ΔIADRC)	**A/D with RC**	330	500	µA	-40°C to +85°C	VDD = 2.0V	
		385	500	µA	-40°C to +85°C	VDD = 3.3V	
D028 (ΔIUSB)	**USB Module**	1	2	mA	-40°C to +85°C	VDD and VUSB3V3 = 3.3V	USB enabled, no cable connected; traffic makes a large difference(5)

Note 1: Data in the Typical column is at 3.3V, 25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

2: Incremental current while the module is enabled and running.

3: LCD is enabled and running, no glass is connected; the resistor ladder current is not included.

4: LCD is enabled and running, no glass is connected.

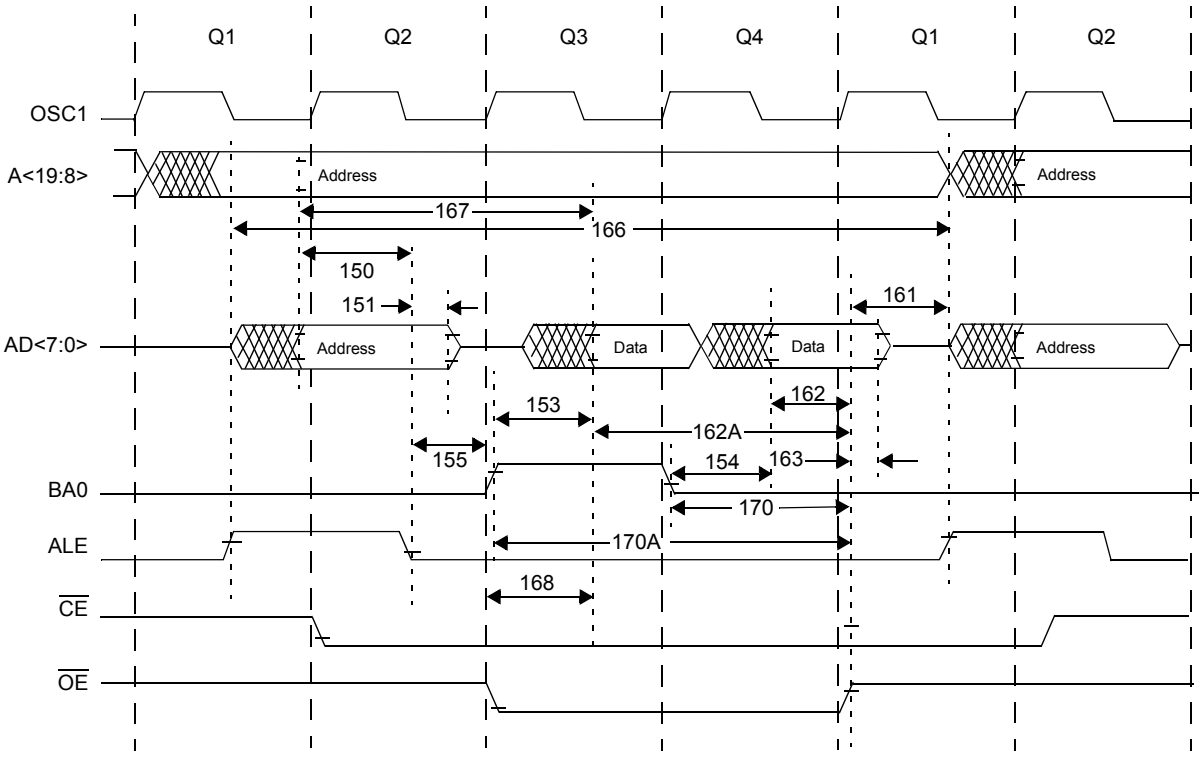
5: This is the module differential current when the USB module is enabled and clocked at 48 MHz, but with no USB cable attached. When the USB cable is attached, or data is being transmitted, the current consumption may be much higher (see **Section 27.6.4 "USB Transceiver Current Consumption"**). During 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 Specification*" and therefore, may be as low as 900Ω during Idle conditions.

TABLE 30-8: DC CHARACTERISTICS: POWER-DOWN AND SUPPLY CURRENT PIC18F97J94 FAMILY (INDUSTRIAL)

DC CHARACTERISTICS			**Standard Operating Conditions: 3.0V < VDD < 3.6V** -40°C ≤ TA ≤ +85°C for Industrial (unless otherwise stated)				
Param No.	**Sym.**	**Characteristic**	**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
	VBT	Operating Voltage	2.0	—	3.6	V	Battery connected to VBAT pin
	VBTADC	VBAT A/D Monitoring Voltage Specification(1)	1.6	—	3.6	V	A/D monitoring the VBAT pin using the internal A/D channel

Note 1: Measure A/D value using the A/D represented by the equation (Measured Voltage = ((VBAT/2)/VDD) * 1024) for 10-bit A/D; Measured Voltage = ((VBAT/2)/VDD) * 4096) for 12-bit A/D.

FIGURE 30-5: PROGRAM MEMORY FETCH TIMING DIAGRAM (8-BIT)

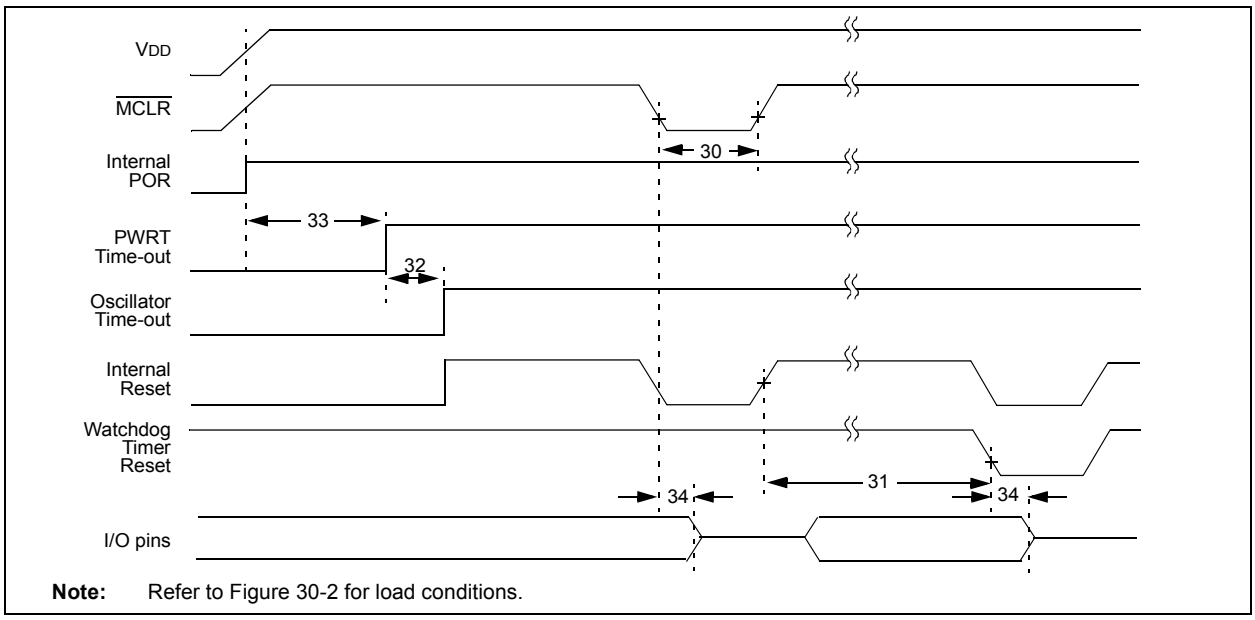


Note: Fmax = 25 MHz in 8-Bit External Memory mode.

TABLE 30-24: PROGRAM MEMORY FETCH TIMING REQUIREMENTS (8-BIT)

Param. No.	Symbol	Characteristics	Min.	Typ.	Max.	Units
150	TadV2alL	Address Out Valid to ALE ↓ (address setup time)	0.25 TCY – 10	—	—	ns
151	TalL2adl	ALE ↓ to Address Out Invalid (address hold time)	5	—	—	ns
153	BA01	BA0 ↑ to Most Significant Data Valid	0.125 TCY	—	—	ns
154	BA02	BA0 ↓ to Least Significant Data Valid	0.125 TCY	—	—	ns
155	TalL2oeL	ALE ↓ to $\overline{OE}$ ↓	0.125 TCY	—	—	ns
161	ToeH2adD	$\overline{OE}$ ↑ to A/D Driven	0.125 TCY – 5	—	—	ns
162	TadV2oeH	Least Significant Data Valid Before $\overline{OE}$ ↑ (data setup time)	20	—	—	ns
162A	TadV2oeH	Most Significant Data Valid Before $\overline{OE}$ ↑ (data setup time)	0.25 TCY + 20	—	—	ns
163	ToeH2adI	$\overline{OE}$ ↑ to Data in Invalid (data hold time)	0	—	—	ns
166	TalH2alH	ALE ↑ to ALE ↑ (cycle time)	—	TCY	—	ns
167	TACC	Address Valid to Data Valid	0.5 TCY – 10	—	—	ns
168	Toe	$\overline{OE}$ ↓ to Data Valid	—	—	0.125 TCY + 5	ns
170	TubH2oeH	BA0 = 0 Valid Before $\overline{OE}$ ↑	0.25 TCY	—	—	ns
170A	TubL2oeH	BA0 = 1 Valid Before $\overline{OE}$ ↑	0.5 TCY	—	—	ns

FIGURE 30-7: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING



Note: Refer to Figure 30-2 for load conditions.

FIGURE 30-8: BROWN-OUT RESET TIMING