Microchip PIC Assembly Language: Table of Instruction Mnemonics & Their Functions

Instruction	Cycles	Form	Function		Status Affected
ADDLW	1	ADDLW **k**	**W** = **W** + **k**		C, DC, Z
ADDWF	1	ADDWF **f**,**d**	**d** = 0 → **W** = **W** + **f**	**d** =1 → **f** = **f** + **W**	C, DC, Z
ANDLW	1	ANDLW **k**	**W** = **W** AND **k**		Z
ANDWF	1	ANDWF **f**,**d**	d = 0 → W = W AND f	**d** =1 → **f** = **f** AND **W**	Z
BCF	1	BCF **f**,**b**	CLEAR bit "**b**" of register "**f**"		none
BSF	1	BSF **f**,**b**	SET bit "**b**" of register "**f**"		none
BTFSC	2	BTFSC **f**,**b**	SKIP next line if bit "**b**" of register "**f**" is clear (= 0)		none
BTFSS	2	BTFSS **f**,**b**	SKIP next line if bit "**b**" of register "**f**" is set (= 1)		none
CALL	2	CALL **k**	CALL subroutine that starts at memory location "**k**" (move to line after CALL when RETURN is reached in subroutine)		none
CLRF	1	CLRF **f**	Clear memory location "**f**" and set "Z" bit (bit 2) of STATUS register		Z
CLRW	1	CLRW	Clear "**W**" register and set "Z" bit (bit 2) of STATUS register		Z
CLRWDT	1	CLRWDT	Reset (CLEAR) Watchdog Timer, reset prescaler of WDT		!TO, !PD (1,1)*
COMF	1	COMF **f**,**d**	**d** = 0 → **W** = 1's compliment of "**f**"	**d** = 1 → **f** = 1's compliment of "**f**"	Z
DECF	1	DECF **f**,**d**	**d** = 0 → **W** = **f** - 1	**d** = 1 → **f** = **f** - 1	Z
DECFSZ	2	DECFSZ **f**,**d**	**d** = 0 → **W** = **f** - 1, Next line skipped if result = 0	**d** = 1 → **f** = **f** - 1, Next line skipped if result = 0	none
GOTO	2	GOTO **k**	Jump (GOTO) to address "**k**"		none
INCF	1	INCF **f**,**d**	**d** = 0 → **W** = **f** + 1	**d** = 1 → **f** = **f** + 1	Z
INCFSZ	2	INCFSZ **f**,**d**	**d** = 0 → **W** = **f** + 1, Next line skipped if result = 0	**d** = 1 → **f** = **f** + 1, Next line skipped if result = 0	none
IORLW	1	IORLW **k**	**W** = **W** OR **k**		Z
IORWF	1	IORWF **f**,**d**	**d** = 0 → **W** = **W** OR **f**	**d** = 1 → **f** = **f** OR **W**	Z
MOVF	1	MOVF **f**,**d**	**d** = 0 → **W** = **f**	**d** = 1 → **f** = **f** (useful to test a file register)	Z
MOVLW	1	MOVLW **k**	**W** = **k**		none
MOVWF	1	MOVWF **f**	**f** = **W**		none
NOP	1	NOP	No Operation (Useful for killing time or for subroutine place holders)		none
RETFIE	2	RETFIE	Return From Interrupt		none
RETLW	2	RETLW **k**	RETURN from subroutine with "**k**" in "**W**" (Put this at the end of a subroutine where you need to return a number back)		none
RETURN	2	RETURN	Jump to the line of code that is immediately after the line that this subroutine was called from		none
RLF	1	RLF **f**,**d**	**d** = 0 → **W** = "**f**" rotated 1 bit to the left through "C" flag	**d** = 1 → **f** = "**f**" rotated 1 bit to the left through "C" flag	C
RRF	1	RRF **f**,**d**	**d** = 0 → **W** = "**f**" rotated 1 bit to the right through "C" flag	**d** = 1 → **f** = "**f**" rotated 1 bit to the right through "C" flag	C
SLEEP	1	SLEEP	Puts PIC into SLEEP mode with the oscillator stopped (00h → WDT, 0 → WDT prescaler)		!TO, !PD (1,0)*
SUBLW	1	SUBLW **k**	**W** = **k** - **W**		C, DC, Z
SUBWF	1	SUBWF **f**,**d**	**d** = 0 → **W** = **f** - **W**	**d** = 1 → **f** = **f** - **W**	C, DC, Z
SWAPF	1	SWAPF **f**,**d**	**d** = 0 → **W** = "**f**" with nibbles swapped (i.e. 4E → E4)	**d** = 1 → **f** = "**f**" with nibbles swapped (i.e. 4E → E4)	none
XORLW	1	XORLW **k**	**W** = **W** XOR **k**		Z
XORWF	1	XORWF **f**,**d**	**d** = 0 → **W** = **W** XOR **f**	**d** = 1 → **f** = **W** XOR **f**	Z

* (?, ?) = ($\overline{TO}$, $\overline{PD}$)

W → The "W" register (or accumulator) is used as temporary storage for many instructions
f → "f" represents a user defined variable or a system memory location
k → "k" represents a literal number or a defined constant
d → The value of "d" (0 or 1) determines where the numerical result of an operation will be stored (if **d** = 0 then result in **W**, if **d** = 1 then result in **f**)
b → "b" represents a particular bit in an 8-bit memory location (7, 6, 5, 4, 3, 2, 1, 0)