

PIC18F INSTRUCTION SET SUMMARY

Mnemonic, operands <i>[] is optional, default d=1 (in f), a=0 (no BSR)</i>	Description	Cycles ($f_{osc}/4$)	Encoding <i>x values are ignored during decode</i>	STATUS bit affected	Notes
BYTE-ORIENTED FILE REGISTER (f) OPERATIONS					
ADDWF f [, d [, a]]	(WREG) + (f)	1	0010 01da ffff ffff	C, DC, Z, OV, N	
ADDWFC f [, d [, a]]	(WREG) + C + (f)	1	0010 00da ffff ffff	C, DC, Z, OV, N	
ANDWF f [, d [, a]]	(WREG) & (f), bitwise AND operation	1	0001 01da ffff ffff	Z, N	
CLRF f [, a]	(f) ← 0, clears f	1	0110 101a ffff ffff	Z	
COMF f [, d [, a]]	(~f), complement f (inverts all bits)	1	0001 11da ffff ffff	Z, N	
CPFSEQ f [, a]	Compare (f) with (WREG), skip if =	1 (2 or 3)	0110 001a ffff ffff	None	
CPFSGT f [, a]	Compare (f) with (WREG), skip if >	1 (2 or 3)	0110 010a ffff ffff	None	
CPFSLT f [, a]	Compare (f) with (WREG), skip if <	1 (2 or 3)	0110 000a ffff ffff	None	
DECF f [, d [, a]]	Decrement (f)	1	0000 01da ffff ffff	C, DC, Z, OV, N	
DECFSZ f [, d [, a]]	Decrement (f), skip next if result == 0	1 (2 or 3)	0010 11da ffff ffff	None	
DCFSNZ f [, d [, a]]	Decrement (f), skip next if result != 0	1 (2 or 3)	0100 11da ffff ffff	None	
INCF f [, d [, a]]	Increment (f)	1	0010 10da ffff ffff	C, DC, Z, OV, N	
INCFSZ f [, d [, a]]	Increment (f), skip next if result == 0	1 (2 or 3)	0011 11da ffff ffff	None	
INFSNZ f [, d [, a]]	Increment (f), skip next if result != 0	1 (2 or 3)	0100 10da ffff ffff	None	
IORWF f [, d [, a]]	(WREG) (f), bitwise OR operation	1	0001 00da ffff ffff	Z, N	
MOVF f [, d [, a]]	Move (f) to (WREG) or to itself	1	0101 00da ffff ffff	Z, N	
MOVFF f_s, f_d	(f_d) ← (f_s), 1 st word encodes f_s 2 nd word encodes f_d	2	1100 ffff ffff ffff 1111 ffff ffff ffff	None	
MOVWF f [, a]	(f) ← (WREG), move WREG to f	1	0110 111a ffff ffff	None	
MULWF f [, a]	Multiply (WREG) with (f)	1	0000 001a ffff ffff	None	
NEGF f [, a]	Negate (f), using 2's complement	1	0110 110a ffff ffff	C, DC, Z, OV, N	
RLCF f [, d [, a]]	Rotate left (f) through carry bit	1	0011 01da ffff ffff	C, Z, N	
RLNCF f [, d [, a]]	Rotate left (f)	1	0100 01da ffff ffff	Z, N	
RRCF f [, d [, a]]	Rotate right (f) through carry bit	1	0011 00da ffff ffff	C, Z, N	
RRNCF f [, d [, a]]	Rotate right (f)	1	0100 00da ffff ffff	Z, N	
SETF f [, a]	(f) ← 0xFF, sets all bits of f	1	0110 100a ffff ffff	None	
SUBFWB f [, d [, a]]	(WREG) - (f), with borrow	1	0101 01da ffff ffff	C, DC, Z, OV, N	
SUBWF f [, d [, a]]	(f) - (WREG)	1	0101 11da ffff ffff	C, DC, Z, OV, N	
SUBWFB f [, d [, a]]	(f) - (WREG), with borrow	1	0101 10da ffff ffff	C, DC, Z, OV, N	
SWAPF f [, d [, a]]	Swap nibbles of (f)	1	0011 10da ffff ffff	None	
TSTFSZ f [, a]	Test (f), skip if == 0	1 (2 or 3)	0110 011a ffff ffff	None	
XORWF f [, d [, a]]	(WREG) ⊕ (f), bitwise XOR operation	1	0001 10da ffff ffff	Z, N	
BIT-ORIENTED FILE REGISTER (f) OPERATIONS					
BCF f, b [, a]	Clears bit b of (f)	1	1001 bbba ffff ffff	None	
BSF f, b [, a]	Sets bit b of (f)	1	1000 bbba ffff ffff	None	
BTFSC f, b [, a]	Bit test b of (f), skip if clear (== 0)	1 (2 or 3)	1011 bbba ffff ffff	None	
BTFSS f, b [, a]	Bit test b of (f), skip if set (== 1)	1 (2 or 3)	1010 bbba ffff ffff	None	

PIC18F INSTRUCTION SET SUMMARY

BTG	f, b [, a]	Toggle bit b of (f)	1	0111 bbba ffff ffff	None	
CONTROL OPERATIONS						
BC	n	Branch to n if C	1 (2)	1110 0010 nnnn nnnn	None	
BN	n	Branch to n if N	1 (2)	1110 0110 nnnn nnnn	None	
BNC	n	Branch to n if C == 0	1 (2)	1110 0011 nnnn nnnn	None	
BNN	n	Branch to n if N == 0	1 (2)	1110 0111 nnnn nnnn	None	
BNOV	n	Branch to n if OV == 0	1 (2)	1110 0101 nnnn nnnn	None	
BNZ	n	Branch to n if Z != 0	1 (2)	1110 0001 nnnn nnnn	None	
BOV	n	Branch to n if OV	1 (2)	1110 0100 nnnn nnnn	None	
BRA	n	Branch to n	2	1101 0nnn nnnn nnnn	None	
BZ	n	Branch to n if Z	1 (2)	1110 0000 nnnn nnnn	None	
CALL	k [, s]	Call subroutine 1 st word 2 nd word	2	1110 110s k7kkk kkkk ₀ 1111 k ₁₉ kkk kkkk kkkk ₈	None	
CLRWDT		Clear watchdog timer	1	0000 0000 0000 0100	TO, PD	
DAW		Decimal adjust (WREG)	1	0000 0000 0000 0111	C	
GOTO	k	Go to address 1 st word 2 nd word	2	1110 1111 k7kkk kkkk ₀ 1111 k ₁₉ kk kkkk kkkk ₈	None	
NOP		No operation	1	0000 0000 0000 0000	None	
NOP		No operation	1	1111 xxxx xxxx xxxx	None	
POP		Pop top of return stack (TOS)	1	0000 0000 0000 0110	None	
PUSH		Push top of return stack (TOS)	1	0000 0000 0000 0101	None	
RCALL	n	Relative call	2	1101 1nnn nnnn nnnn	None	
RESET		Software device reset	1	0000 0000 1111 1111	All	
RETFIE	[s]	Return from interrupt	2	0000 0000 0001 000s	GIE, PEIE	
RETLW	k	Return with literal in WREG	2	0000 1100 kkkk kkkk	None	
RETURN	[s]	Return from subroutine	2	0000 0000 0001 001s	None	
SLEEP		Go into standby mode	1	0000 0000 0000 0011	TO, PD	
LITERAL OPERATIONS						
ADDLW	k	(WREG) + k	1	0000 1111 kkkk kkkk	C, DC, Z, OV, N	
ANDLW	k	(WREG) & k	1	0000 1011 kkkk kkkk	Z, N	
IORLW	k	(WREG) k	1	0000 1001 kkkk kkkk	Z, N	
LFSR	f, k	Move literal to FSRx 1 st word 2 nd word	2	1110 1110 00ff kkkk 1111 0000 kkkk kkkk	None	
MOVLB	k	BSR<3:0> ← k	1	0000 0001 0000 kkkk	None	
MOVLW	k	(WREG) ← k	1	0000 1110 kkkk kkkk	None	
MULLW	k	(WREG) * k, multiply with literal	1	0000 1101 kkkk kkkk		
RETLW	k	Return with literal in (WREG)	2	0000 1100 kkkk kkkk	None	
SUBLW	k	k - (WREG)	1	0000 1000 kkkk kkkk	C, DC, Z, OV, N	
XORLW	k	(WREG) ⊕ k, bitwise XOR	1	0000 1010 kkkk kkkk	Z, N	