MICROCHIP

PICmicro® 18C MCU Family Reference Manual

Table 31-1: PIC18CXXX Instruction Set Summary

Mnemonic,	Operands	Description	Cycles	16-Bit Instruction Word				Status Affected	Notes
				MSb			LSb		
BYTE-ORIENTED FILE REGISTER OPERATIONS									
ADDWF	f, d, a	Add WREG and f	1	0010	01da	ffff	ffff	C, DC, Z, OV, N	1, 2, 3
ADDWFC	f, d, a	Add WREG and Carry bit to f	1	0010	00da	ffff	ffff	C, DC, Z, OV, N	1, 2, 3
ANDWF	f, d, a	AND WREG with f	1	0001	01da	ffff	ffff	Z, N	1,2, 3
CLRF	f, a	Clear f	1	0110	101a	ffff	ffff	Z	2, 3
COMF	f, d, a	Complement f	1	0001	11da	ffff	ffff	Z, N	1, 2, 3
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	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	1, 2, 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, 2
MOVFF	f_s, f_d	Move f_s (source) to 1st word f_d (destination) 2nd word	2	1100 1111	ffff ffff	ffff ffff	ffff ffff	None	2
MOVWF	f, a	Move WREG to f	1	0110	111a	ffff	ffff	None	2
MULWF	f, a	Multiply WREG with f	1	0000	001a	ffff	ffff	None	
NEGF	f, a	Negate f	1	0110	110a	ffff	ffff	C, DC, Z, OV, N	1, 2
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	1, 2
RRCF	f, d, a	Rotate Right f through Carry	1	0011	00da	ffff	ffff	C, Z, N	1, 2
RRNCF	f, d, a	Rotate Right f (No Carry)	1	0100	00da	ffff	ffff	Z, N	1, 2
SETF	f, a	Set f	1	0110	100a	ffff	ffff	None	2
SUBFWB	f, d, a	Subtract f from WREG with borrow	1	0101	01da	ffff	ffff	C, DC, Z, OV, N	1, 2
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	1, 2
SWAPF	f, d, a	Swap nibbles in f	1	0011	10da	ffff	ffff	None	1, 2, 4
TSTFSZ	f, a	Test f, skip if 0	1 (2 or 3)	0110	011a	ffff	ffff	None	2
XORWF	f, d, a	Exclusive OR WREG with f	1	0001	10da	ffff	ffff	Z, N	
BIT-ORIENTED FILE REGISTER OPERATIONS									
BCF	f, b, a	Bit Clear f	1	1001	bbba	ffff	ffff	None	1, 2
BSF	f, b, a	Bit Set f	1	1000	bbba	ffff	ffff	None	1, 2
BTFSC	f, b, a	Bit Test f, Skip if Clear	1 (2 or 3)	1011	bbba	ffff	ffff	None	3, 4
BTFSS	f, b, a	Bit Test f, Skip if Set	1 (2 or 3)	1010	bbba	ffff	ffff	None	3, 4
BTG	f, d, a	Bit Toggle f	1	0111	bbba	ffff	ffff	None	1, 2

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 2 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.

5: If the table write starts the write cycle to internal program memory, the write continues until terminated.

Table 31-1: PIC18CXXX Instruction Set Summary (Continued)

Mnemonic, Operands	Description	Cycles	16-Bit Instruction Word				Status Affected	Notes
			MSb			LSb		
CONTROL OPERATIONS								
BC n	Branch if Carry	1 (2)	1110	0010	nnnn	nnnn	None	
BN n	Branch if Negative	1 (2)	1110	0110	nnnn	nnnn	None	
BNC n	Branch if Not Carry	1 (2)	1110	0011	nnnn	nnnn	None	
BNN n	Branch if Not Negative	1 (2)	1110	0111	nnnn	nnnn	None	
BNOV n	Branch if Not Overflow	1 (2)	1110	0101	nnnn	nnnn	None	
BNZ n	Branch if Not Zero	1 (2)	1110	0001	nnnn	nnnn	None	
BOV n	Branch if Overflow	1 (2)	1110	0100	nnnn	nnnn	None	
BRA n	Branch Unconditionally	2	1101	0nnn	nnnn	nnnn	None	
BZ n	Branch if Zero	1 (2)	1110	0000	nnnn	nnnn	None	
CALL n, s	Call subroutine 1st word 2nd word	2	1110 1111	110s kkkk	kkkk kkkk	kkkk kkkk	None	
CLRWDT —	Clear Watchdog Timer	1	0000	0000	0000	0100	$\overline{\text{TO}}$, $\overline{\text{PD}}$	
DAW —	Decimal Adjust WREG	1	0000	0000	0000	0111	C	
GOTO n	Go to address 1st word 2nd word	2	1110 1111	1111 kkkk	kkkk kkkk	kkkk kkkk	None	
NOP —	No Operation	1	0000	0000	0000	0000	None	
NOP —	No Operation	1	1111	xxxx	xxxx	xxxx	None	4
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 enable	2	0000	0000	0001	000s	GIE/GIEH, PEIE/GIEL	
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	$\overline{\text{TO}}$, $\overline{\text{PD}}$	

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 2 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.

5: If the table write starts the write cycle to internal program memory, the write continues until terminated.

Table 31-1: PIC18CXXX Instruction Set Summary (Continued)

Mnemonic, Operands	Description	Cycles	16-Bit Instruction Word				Status Affected	Notes
			MSb			LSb		
LITERAL OPERATIONS								
ADDLW k	Add literal and WREG	1	0000	1111	kkkk	kkkk	C, DC, Z, OV, N	
ANDLW k	AND literal with WREG	1	0000	1011	kkkk	kkkk	Z, N	
IORLW k	Inclusive OR literal with WREG	1	0000	1001	kkkk	kkkk	Z, N	
LFSR f, k	Move literal (12-bit) 1st word to FSRx 2nd word	2	1110 1111	1110 0000	00ff kkkk	kkkk kkkk	None	
MOVLB k	Move literal to BSR<3:0>	1	0000	0001	0000	kkkk	None	
MOVLW k	Move literal to WREG	1	0000	1110	kkkk	kkkk	None	
MULLW k	Multiply literal with WREG	1	0000	1101	kkkk	kkkk	None	
RETLW k	Return with literal in WREG	2	0000	1100	kkkk	kkkk	None	
SUBLW k	Subtract WREG from literal	1	0000	1000	kkkk	kkkk	C, DC, Z, OV, N	
XORLW k	Exclusive OR literal with WREG	1	0000	1010	kkkk	kkkk	Z, N	
DATA MEMORY ↔ PROGRAM MEMORY OPERATIONS								
TBLRD*	Table Read	2	0000	0000	0000	1000	None	
TBLRD*+	Table Read with post-increment	2	0000	0000	0000	1001	None	
TBLRD*-	Table Read with post-decrement	2	0000	0000	0000	1010	None	
TBLRD+*	Table Read with pre-increment	2	0000	0000	0000	1011	None	
TBLWT*	Table Write	2 (5)	0000	0000	0000	1100	None	
TBLWT*+	Table Write with post-increment	2 (5)	0000	0000	0000	1101	None	
TBLWT*-	Table Write with post-decrement	2 (5)	0000	0000	0000	1110	None	
TBLWT+*	Table Write with pre-increment	2 (5)	0000	0000	0000	1111	None	

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 2 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.

5: If the table write starts the write cycle to internal program memory, the write continues until terminated.