様々なデータの移動は汎用レジスタである Wreg を介して行う。

PIC16アセンブラ　早見表

大文字小文字は区別しない。

◆ 定数の表記法

2進数 B'11000100'

10進数 D'12'

16進数 H'6a' or 6ah or 0x6a

◆ ニーモニックで使われる略称

L : リテラル（定数）

W : Wreg（ワーキングレジスタ）

F : ファイルレジスタ（PICではメモリをこう呼ぶ）

◆ 特殊なニーモニック（マクロ）

banksel addr そのアドレスが存在するバンクに切り替える

アドレスはバンク 0 が 00h～7fh

バンク 1 が 80h～ffh

インクルードファイルを参考にすること

例

banksel TRISA TRISA が存在するバンクに切り替える

◆ デフォルトで定義されている定数

STATUSレジスタ用

Z 2　 ゼロフラグの位置

C 0 　 キャリーフラグの位置

movf などのコマンド用

W 0 　 ワーキングレジスタ (Wreg)

F 1 　 ファイルレジスタ

◆ 定数定義

T1 EQU 0x20　　アドレスを指定して変数名として使う

T2 EQU 0x21　　ことが多い

T3 EQU 0x22

◆ ビット操作命令

bcf addr,b ビット b をクリア（0 にする）

b は 0～7 の値をとる

bsf addr,b ビット b をセット（1 にする）

例

bcf PORTA,0 PORTA の bit 0 を 0 にする

bsf PORTB,3 PORTB の bit 3 を 1 にする

◆ クリア（全ての bit を 0 にする）

clrf addr addr ← 0

clrw Wreg ← 0 その結果 Z ← 1

◆ MOVE

movlw B'0001111' Wreg ← '0001111'

movf addr,d d = W(0) : Wreg ← addr

d = F(1) : addr ← addr

Z フラグに影響する

movwf addr Wreg → addr

例 : アドレス DAT に 3 を入れる

movlw D'3' Wreg ← 3

movwf DAT Wreg → DAT

◆ 加算、減算

addwf addr,d d = W(0) : Wreg ← Wreg + addr

d = F(1) : addr ← Wreg + addr

addlw num Wreg ← Wreg + num

subwf addr,d d = W(0) : Wreg ← addr - Wreg

　　　（順番に注意！）

d = F(1) : addr ← addr - Wreg

結果が正または 0 のとき C フラグが 1

addr が 5 ならジャンプ、というようなとき d = W(0) として addr の内容は不変にしてフラグを見てジャンプをすればよい。

sublw num Wreg ← num - Wreg（順番に注意！）

例1 : 加算

movf DATA\_A,W Wreg ← DATA\_A

addwf DATA\_B,W Wreg ← Wreg + DATA\_B

例2 : 減算

movf DATA\_A,W Wreg ← DATA\_A

subwf DATA\_B,W Wreg ← DATA\_B - Wreg

DATA\_B >= DATA\_A のとき　C = 1

◆ インクリメント・デクリメント

decf addr,d d = W(0) : Wreg = addr - 1

d = F(1) : addr = addr - 1

incf addr,d addr の内容を +1 する

◆ ラベル

1 桁目から文字列を書き始める。ラベルの後に : は不要

◆ ジャンプ

goto LABEL 無条件ジャンプ

call LABEL サブルーチンコール

return サブルーチンからリターン

◆ 条件分岐

btfsc addr,b addr のビット b が 0 なら次の命令を

スキップ

btfss addr,b addr のビット b が 1 なら次の命令を

スキップ

例

movf DATA\_B,W Wreg ← DATA\_B

subwf DATA\_A,W Wreg ← DATA\_A - Wreg

btfsc STATUS,C C フラグが 1 なら ( DATA\_A >= DATA\_B ) 次の命令は無視

◆ ループ専用命令

decfsz addr,d addr の内容を -1 した後、結果が 0 のとき

次の命令をスキップする。

例

movlw D'100'

movwf VAR VAR はあらかじめ定義したアドレス

LOOP

何らかの処理

decfsz VAR,F

goto LOOP

◆ 何もしない

nop 1サイクル（4クロック）消費

--------------------------------------------

◆ 主なレジスタ

STATUS 6-5 バンク切り替え

2 Z フラグ

0 C フラグ

PCLATH プログラムカウンタ

INTCON 割り込みの状態

PORTA 入出力用ポート

PORTB 同上

TRISA PORTA の in/out の方向を決める

0 : out 1 : in

TRISB PORTB の in/out の方向を決める