

基礎コンピュータ工学

第5章 機械語プログラミング

(パート6：残りの条件分岐命令と条件判断の演習)

<https://github.com/tctsigemura/TecTextBook>

本スライドの入手：



ジャンプ命令（7種類）の残り3種類

無条件ジャンプ命令： プログラムの流れを指定のアドレスに飛ばす。

- *JMP (Jump)* 命令：いつもジャンプする。

条件ジャンプ命令： ある条件のときだけジャンプする。

- *JZ (Jump on Zero)* 命令： $Z = 1$ ならジャンプ
- *JC (Jump on Carry)* 命令： $C = 1$ ならジャンプ
- *JM (Jump on Minus)* 命令： $S = 1$ ならジャンプ
- *JNZ (Jump on Not Zero)* 命令： $Z = 0$ ならジャンプ
- *JNC (Jump on Not Carry)* 命令： $C = 0$ ならジャンプ
- *JNM (Jump on Not Minus)* 命令： $S = 0$ ならジャンプ

JNZ (Jump on Not Zero) 命令

Zフラグが0なら（計算結果が0でないなら）ジャンプする.

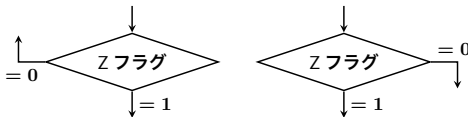
フラグ： 変化しない.

ニーモニック： JNZ EA (if(Z=0) PC \leftarrow EA)

命令フォーマット： 2バイトの長さを持つ.

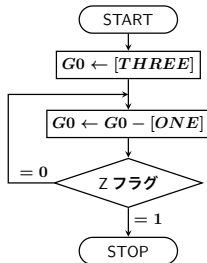
第1バイト		第2バイト
OP	GR XR	
1011 ₂	01 ₂ XR	aaaa aaaa

フローチャート： ある程度，自由にアレンジしてよい.

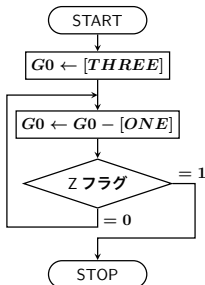


JNZ 命令の使用例 (JZ 命令との比較)

ループを 3 回, 繰り返すプログラム



番地	機械語	ラベル	ニーモニック
00	10 07	LOOP	LD G0, THREE
02	40 08		SUB G0, ONE
04	B4 02		JNZ LOOP
06	FF		HALT
07	03	THREE	DC 3
08	01	ONE	DC 1



番地	機械語	ラベル	ニーモニック
00	10 09	LOOP	LD G0, THREE
02	40 0A		SUB G0, ONE
04	A4 08		JZ STOP
06	A0 02		JMP LOOP
08	FF	STOP	HALT
09	03	THREE	DC 3
0A	01	ONE	DC 1

JNC (Jump on Not Carry) 命令

Cフラグが0なら（オーバーフローしていないなら）ジャンプする.

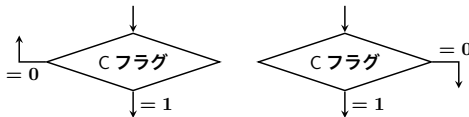
フラグ： 変化しない.

ニーモニック： JNC EA (if(C=0) PC \leftarrow EA)

命令フォーマット： 2バイトの長さを持つ.

第1バイト		第2バイト
OP	GR XR	
1011 ₂	10 ₂ XR	aaaa aaaa

フローチャート： ある程度，自由にアレンジしてよい.



JNM (Jump on Not Minus) 命令

S フラグが 0 なら (正かゼロなら) ジャンプする.

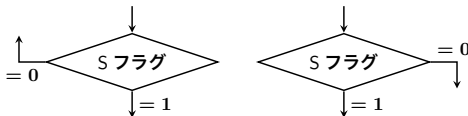
フラグ: 変化しない.

ニーモニック: JNM EA (if(S=0) PC \leftarrow EA)

命令フォーマット: 2 バイトの長さを持つ.

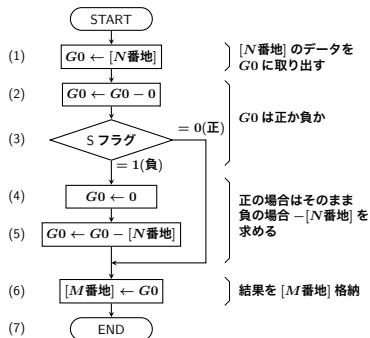
第 1 バイト		第 2 バイト
OP	GR XR	
1011 ₂	11 ₂ XR	aaaa aaaa

フローチャート: ある程度, 自由にアレンジしてよい.



条件判断の例 (JNM 使用)

絶対値を求めるプログラム (例題 5-1 の改良)



注意: [N番地] は、N 番地に格納されているデータのこと

番地	機械語	ラベル	ニーモニック	
00	10 0E	START	LD	G0,N
02	40 0D		SUB	G0,ZERO
04	BC 0A		JNM	L1
06	10 0D		LD	G0,ZERO
08	40 0E		SUB	G0,N
0A	20 0F	L1	ST	G0,M
0C	FF		HALT	
0D	00	ZERO	DC	0
0E	FF	N	DC	-1
0F	00	M	DS	1

- JNM を使用したほうが短くなる.

まとめ

●学んだこと

必須ではないがあると便利な JNZ, JNC, JNM 命令

●条件判断の演習

1. プログラムの作成手順を再度確認

- (1) フローチャートを描く.
- (2) フローチャートを基にニーモニックを書く.
- (3) アドレスを決める.
- (4) 機械語を作る.

2. 演習 (宿題)

- (1) N 番地の値がゼロなら M 番地にゼロを, そうでなければ M 番地に 1 を格納するプログラム
 - LD 命令はフラグを変化させないので...
 - 前回の「条件判断 2」のパターンを利用
- (2) N 番地の値と M 番地の値で, 大きい方を L 番地に格納するプログラム
 - 値は符号付きの数値とする.
 - 比較は引き算でできる.