

x86 系 MPU の機械語とアセンブリ言語（抜粋）

略語一覧

- AL AL レジスタ。8 ビットで加算器、乗算器になる。
- BL BL レジスタ。8 ビットで加算器やアドレスの間接指定に使う。
- imm8 即値バイト値（8 ビットの数値）
- r8 8 ビットのレジスタ（AL,BL,CL,DL,AH,BH,CH,DH）の一つ。
- r/m8 8 ビットのレジスタまたはメモリの一つ。
- rel8 相対アドレス

MOV r8, imm8 : imm8 を r8 に転送する

アセンブリ言語	MOV AL, 1 [数値の]1 を AL に転送する
機械語[16 進数表示]	B0 01
機械語	1011 0000_0000 0001
解 説	1011[転送する] 0[次の 8 ビット]000[AL]_0000 0001[数値の 1]

ADD AL, imm8 : imm8 を AL に加算する

*AL の内容は、AL のもとの内容 + imm8 になる。

アセンブリ言語	ADD AL, 2 [数値の]2 を AL に加算する
機械語[16 進数表示]	04 02
機械語	0000 0100_0000 0010
解 説	0000[加算する] 010[AL]0[次の 8 ビットの数値] _0000 0010[数値の 2]

ADD r/m8, r8 : r/m8[の内容]を r8 に加算する

*r8 の内容は、r8 のもとの内容 + r/m8 の内容 になる。

アセンブリ言語	ADD AL, BL BL[の内容]を AL に加算する
機械語[16 進数表示]	00 D8
機械語	0000 0000_1101 1000
解 説	0000 0000[次の 8 ビットに書かれた r/m8 を r8 に加算する] _11[規定値]01 1[BL]000[AL]

JNE (または JNZ) rel8 : ZF=0 のとき rel8 に分岐し、ZF=1 のとき次の命令を実行する

アセンブリ言語	JNE 000A ZF=0 のとき (演算結果が 0 ではない)、000A 番地の命令に分岐して実行し、 ZF<>0 のときは次の命令を実行する
機械語[16 進数表示]	75 FB * JNE のアドレスによって FB は異なる。
機械語	0111 0101_1111 1011
解 説	ZF=0 のとき[アドレスが] 1111 1011 の命令を実行し、 ZF<>0 のときは次の命令を実行する

- * ZF [ゼロフラグ]はフラグレジスタの中の1ビットで、前の演算の結果=0 のとき 1 にセットされ、それ以外では 0 にリセットされる。
- * 分岐とは、分岐先のアドレスを CPU のプログラムカウンタに転送し、次に実行する命令のアドレスにすることにより、分岐先のアドレスの命令に JUMP すること。

BASIC 言語と JNE (分岐命令の一つ) との対応関係

BASIC のプログラム例		アセンブリ言語によるプログラム	
I=I-1	I-1 を I にする	DEC CX	CX-1 を CX にする *ZF が 1 か 0 になる
IF I<>0 THEN 120	I が 0 でなければ 120 行目の命令に分岐する	JNE 000A	ZF が 0 ならば 000A 行目の命令に分岐する

MOV r/m8, r8 : r8[の内容]を r/m8 に転送する

アセンブリ言語	MOV AL, BL BL[の内容]を AL に転送する (「加算する」と同じ)
機械語[16 進数表示]	88 D8
機械語	1000 1000_1101 1000
解 説	1000 1000[次の 8 ビットに書かれた r8 を r/m8 に転送] _11[既定値]01 1[BL]000[AL]

参考文献

1. 藤本文彦『98 マシン語』技術評論社、1984 年。
2. 田中廣『PC-9801 機械語プログラミング 500 題』日刊工業新聞社、1987 年。
3. 『IA-32 インテル アーキテクチャ・ソフトウェア・デベロッパーズ・マニュアル 中巻 A&B : 命令セット・レファレンス』2004 年