

F² MC-16LX 命令表

FUJITSU 16bit micro-computer

ニーモニク	オペレーション	備考	ニーモニク	オペレーション	備考	ニーモニク	オペレーション	1
MOVX A,#imm	符号付 byte(A) imm8 (AH) (AL)		EXT	word(A)8-15 (A)7 符号拡張		EXTW	long(A)16-31 (A)15	
MOVX A,ea	符号付 byte(A) (ea) (AH) (AL)		ZEXT	word(A)8-15 0		ZEXTW	long(A)16-31 0	
MOVX A,@A	符号付 byte(A) ((A))		MOVX 符号付は、(A)8-15に第7ビットの符号に応じて00,FFが入る					
MOVX A,@RLi+imm	符号付 byte(A) ((i)+) (AH) (AL)							
MOVN A,#imm4	byte(A) imm4 (AH) (AL)							
MOV A,#imm	byte(A) imm8 (AH) (AL)		MOVW A,#imm	word(A) imm16 (AH) (AL)		MOVL A,#imm	long(A) imm32	
MOV A,ea	byte(A) (ea) (AH) (AL)		MOVW A,ea	word(A) (ea) (AH) (AL)		MOVL A,ea	long(A) (ea)	
MOV A,@RLi+imm	byte(A) ((RLi)+imm) (AH) (AL)		MOVW A,@RLi+imm	word(A) ((RLi)+imm)(AH) (AL)		MOVL ea,A	long(ea) (A)	
MOV A,@A	byte(A) ((A))		MOVW A,@A	word(A) ((A))				
MOV ea,A	byte(ea) (A)		MOVW ea,A	word(A) (ea)				
MOV @RLi+imm,A	byte((RLi)+imm) (A)		MOVW @RLi+imm,A	word((RLi)+imm) (A)				
MOV @AL,AH	byte((A)) (AH)		MOVW @AL,AH	word((A)) (AH)				
MOV Ri,ea	byte(Ri) (ea)		MOVW RWi,ea	word(RWi) (ea)				
MOV ea,Ri	byte(ea) (Ri)		MOVW ea,RWi	word(ea) (RWi)				
MOV ea,#imm	byte(ea) imm8		MOVW ea,#imm	word(ea) imm16				
MOV RP,#imm	レジスタバンクポインタ(RP) imm8		MOVEA Rwi,ea	word(RWi) ea				
MOV A,brg	byte(A) (brg) (AH) (AL)		MOVEA A,ea	word(A) ea (AH) (AL)				
MOV brg,A	brgはDTB,ADB,SSB,USB,DPR,PCB,SPB		MOVW SP,A	word(SP) (A)				
MOV ILM,#imm	割込レベル(ILM)		ADDSP #imm	word(SP) imm16				
SWAP	byte(A)0-7 (A)8-15		SWAPW	word(AH) (AL) (AH) (AL)				
XCH A,ea	byte(A) (ea)		XCHW A,ea	word(A) (ea)				
XCH Ri,ea	byte(Ri) (ea)		XCHW RWi,ea	word(RWi) (ea)				
MOVB A,imm:bp	byte(A) bit(imm:第bp) (AH) (AL)							
MOVB imm:bp,A	bit(imm:第bp) (A)							
SETB imm:bp	bit(imm:第bp) 1							
CLRB imm:bp	bit(imm:第bp) 0							

特殊な形式

NOP	ノーオペレーション
ADB	次命令の
DTB	フィックスコードを
PCB	強制してADB,DTB,
SPB	PCB,SPB とみなす
NCC	次命令のフラグ無変化
CMR	次命令のレジスタの
	バンクを RP=0 とみなす

eaは44種類

レジスタ間接

R0	RW0	RL0	@RW0
R1	RW1		@RW1
R2	RW2	RL1	@RW2
R3	RW3		@RW3
R4	RW4	RL2	@RW0+
R5	RW5		@RW1+
R6	RW6	RL3	@RW2+
R7	RW7		@RW3+

偏位値付き

偏位・その他

@RW0+disp8	@RW0+disp16
@RW1+disp8	@RW1+disp16
@RW2+disp8	@RW2+disp16
@RW3+disp8	@RW3+disp16
@RW4+disp8	@RW0+RW7
@RW5+disp8	@RW1+RW7
@RW6+disp8	@PC+disp16
@RW7+disp8	addr16

RW0をカウンタとするストリング命令

MOVSB	byte転送 @AH+ @AL+	MOVSW	word転送 @AH+ @AL+
MOVSD	byte転送 @AH- @AL-	MOVSWD	word転送 @AH- @AL-
SCEQB	byte検索 @AH+ AL	SCWEQB	word検索 @AH+ AL
SCEQD	byte検索 @AH- AL	SCWEQD	word検索 @AH- AL
FIFLBS	byte充填 @AH+ AL	FIFLSW	word充填 @AH+ AL

SPへの退避と復帰

PUSHW A	word(SP) (SP)-2,((SP)) (A)	POPW A	(AH) (AL)
PUSHW AH	word(SP) (SP)-2,((SP)) (AH)	POPW AH	
PUSHW PS	PSは割込レベル(ILM)・レジスタバンク(RP)・フラグ類(CCR)	POPW PS	
PUSHW rlist	rlistはRW7,RW6,...,RW0から任意レジスタを記入	PUSHW rlist	

ニーモニック	オペレーション	備考	ニーモニック	オペレーション	備考	ニーモニック	オペレーション	
INC ea	byte(ea) (ea)+1		INCW ea	word(ea) (ea)+1		INCL ea	long(ea) (ea)+1	
DEC ea	byte(ea) (ea)-1		DECW ea	word(ea) (ea)-1		DECL ea	long(ea) (ea)-1	
ADD A,#imm	byte(A) (A)+imm8		ADDW A,#imm	word(A) (A)+imm16		ADDL A,#imm	long(A) (A)+imm32	
ADD A,ea	byte(A) (A)+(ea)		ADDW A,ea	word(A) (A)+(ea)		ADDL A,ea	long(A) (A)+(ea)	
ADD ea,A	byte(ea) (ea)+(A)		ADDW ea,A	word(ea) (ea)+(A)				
ADDC A	byte(A) (AH)+(AL)+(C)		ADDW A	word(A) (AH)+(AL)				
ADDC A,ea	byte(A) (A)+(ea)+(C)		ADDCW A,ea	word(A) (A)+(ea)+(C)				
ADDDC A	byte(A) (AH)+(AL)+(C) [10進数]							
SUB A,#imm	byte(A) (A)-imm8		SUBW A,#imm	word(A) (A)-imm16		SUBL A,#imm	long(A) (A)-imm32	
SUB A,ea	byte(A) (A)-(ea)		SUBW A,ea	word(A) (A)-(ea)		SUBL A,ea	long(A) (A)-(ea)	
SUB ea,A	byte(ea) (ea)-(A)		SUBW ea,A	word(ea) (ea)-(A)				
SUBC A	byte(A) (AH)-(AL)-(C)		SUBW A	word(A) (AH)-(AL)				
SUBC A,ea	byte(A) (A)-(ea)-(C)		SUBCW A,ea	byte(A) (A)-(ea)-(C)				
SUBDC A	byte(A) (AH)-(AL)-(C) [10進数]							
CMP A	byte(AH)-(AL)		CMPW A	word(AH)-(AL)		CMPL A	long(AH)-(AL)	
CMP A,ea	byte(A)-(ea)		CMPW A,ea	word(A)-(ea)		CMPL A,ea	long(A)-(ea)	
CMP A,#imm	byte(AH)-imm8		CMPW A,#imm	word(AH)-imm16		CMPL A,#imm	long(AH)-imm32	
			1'が見つかるまで					
AND CCR,#imm	CCRは・ISTNZVCのフラグ		ANDW A	word(A) (AH) and (A)				
AND A,#imm	byte(A) (A) and imm8		ANDW A,#imm	word(A) (A) and imm16				
AND A,ea	byte(A) (A) and (ea)		ANDW A,ea	word(A) (A) and (ea)		ANDL A,ea	long(A) (A)and(ea)	
AND ea,A	byte(ea) (ea) and (A)		ANDW ea,A	word(ea) (ea) and (A)				
OR CCR,#imm	CCRは・ISTNZVCのフラグ		ORW A	word(A) (AH) or (A)				
OR A,#imm	byte(A) (A) or imm8		ORW A,#imm	word(A) (A) or imm16				
OR A,ea	byte(A) (A) or (ea)		ORW A,ea	word(A) (A) or (ea)		ORL A,ea	long(A) (A)or(ea)	
OR ea,A	byte(ea) (ea) or (A)		ORW ea,A	word(ea) (ea) or (A)				
XOR A,#imm	byte(A) (A) xor imm8		XORW A,#imm	word(A) (A) xor imm16				
XOR A,ea	byte(A) (A) xor (ea)		XORW A,ea	word(A) (A) xor (ea)		XORL A,ea	long(A) (A)xor(ea)	
XOR ea,A	byte(ea) (ea) xor (A)		XORW ea,A	word(ea) (ea) xor (A)				
			XORW A	word(A) (AH) xor (A)				
NOT A	byte(A) not(A)		NOTW A	word(A) not(A)		ビットのサーチ		
NOT ea	byte(ea) not(ea)		NOTW ea	word(ea) not(ea)		NRML A,R0	long(A) sift(A)	
NEG A	byte(A) 0-(A)		NEGW A	word(A) 0-(A)				
NEG ea	byte(ea) 0-(ea)		NEGW ea	word(ea) 0-(ea)				1が見つかるまで byte(R0) sift回数

ニーモニック	オペレーション	備考	ニーモニック	オペレーション	備考	ニーモニック	オペレーション	備考
RORC A	byte 右ローテート(C)付き	(A)	ASRW A	word 算術右シフト1回	(A)	ASRL A,R0	long 算術	
ROLC A	byte (A) 左ローテート(C)付き		LSRW A	word ロジカル右シフト1回	0 (A)		右シフトR0回	(A)
RORC ea	byte 右ローテート(C)付き	(ea)	LSLW A	word (A) 0 ロジカル左シフト1回		LSRL A,R0	long ロジカル	
ROLC ea	byte (ea) 左ローテート(C)付き		ASRW A,R0	word 算術右シフトR0回	(A)		右シフトR0回	0 (A)
ASR A,R0	byte 算術右シフトR0回	(A)	LSRW A,R0	word ロジカル右シフトR0回	0 (A)	LSLL A,R0	long ロジカル	
LSR A,R0	byte ロジカル右シフトR0回	0 (A)	LSLW A,R0	word (A) 0 ロジカル左シフトR0回			(A) 0 左シフトR0回	
LSL A,R0	byte (A) 0 ロジカル左シフトR0回							

フラグ判断で分岐 (相対番地のラベル rel を記載省略)

(-) 無条件	BRA rel	
(Z) 零フラグ	BZ / BEQ	BNZ / BNE
(C) CYフラグ	BC / BLO	BNC / BHS
(N) 正負フラグ	BN (最上ビット=1)	BP
(V) OVERフラグ	BV (繰り上げ下げ)	BNV
(T) ステッキ	BT	BNT
ASRW ASRL や LSRW LSRL において '1' が1回以上排出されるとフラグが立つ なお, ASR や LSR では Tフラグ無変化		
(V,N) 小 , 大=	BLT (-128 ~ 127)	BGE
(V,N,Z) 小= , 大	BLE (-128 ~ 127)	BGT
(C,Z) 低= , 高	BLS (0 ~ 255)	BHI

絶対分岐・サブルーチン・割込み

JMP @A	word(PC) (A)	(PCB)使用
JMP #imm	word(PC) imm16	(PCB)使用
JMP @ea	word(PC) (ea)	(PCB)使用
JCTX @A	word(PC) (A)	(DTB)使用
CALL #imm	word(PC) imm16	(PC)退避
CALL @ea	word(PC) (ea)	(PC)退避
CALLV #vct4	ベクタ表に従って呼出す命令	
INT #vct8	ベクタ表に従ってソフトウェア割込み	
INT imm	呼出先指定のソフトウェア割込み	
INT9	ベクタ表に従ってソフトウェア割込み	

JMPP #imm	(PCB)word(PC)	imm24
JMPP @ea	(PCB)word(PC)	(ea)
CALLP #imm	(PCB)word(PC)	imm24
CALLP @ea	(PCB)word(PC)	(ea)
サブルーチン・割込みからの復帰		
RETP	CALLPからの復帰	
RET	サブルーチンからの復帰	
RETI	割込みからの復帰	

ビット判断で分岐

BBS imm:bp,rel	if bit(imm:第bp)=1 then	レベルに分岐	WBTS imm:bp	if bit(imm:第bp)=1 then	レベルに分岐	els NOP	但し, immは0000 ~ 00FFまでのI/O
BBC imm:bp,rel	if bit(imm:第bp)=0 then	レベルに分岐	WBTC imm:bp	if bit(imm:第bp)=0 then	レベルに分岐	els NOP	但し, immは0000 ~ 00FFまでのI/O

CMP付き相対分岐

CBNE A,#imm,rel	byte(A) imm8 で	ラベル(rel)へ	CWBNE A,#imm,rel	word(A) imm16 で	ラベルへ
CBNE ea,#imm,rel	byte(ea) imm8 で	ラベル(rel)へ	CWBNE ea,#imm,rel	word(ea) imm16 で	ラベルへ
DBNZ ea,rel	byte(ea)=(ea)-1 if (ea) 0	分岐	DWBNZ ea,rel	word(ea)=(ea)-1 if (ea) 0	分岐
但し, RWi+ は使用禁止			但し, RWi+ は使用禁止		

特殊な命令

LINK #imm8	関数の入口
UNLINK	LINKの出口

符号無し乗除算

DIVU A	word(AH)/byte(AL) 商 (AL) 余り (AH)		
DIVU A,ea	word(A)/byte(ea) 商 (A) 余り (ea)	DIVUW A,ea	long(A)/word(ea) 商 (A) 余り (ea)
MULL A	byte(AH) x byte(AL) word(A)	MULEY A	word(AH) x word(AL) long(A)
MULL A,ea	byte(A) x byte(ea) word(A)	MULEY A,ea	word(A) x word(ea) long(A)

符号付き乗除算

DIV A	word(AH)/byte(AL) 商 (AL) 余り (AH)		
DIV A,ea	word(A)/byte(ea) 商 (A) 余り (ea)	DIVW A,ea	long(A)/word(ea) 商 (A) 余り (ea)
MUL A	byte(AH) x byte(AL) word(A)	MULY A	word(AH) x word(AL) long(A)
MUL A,ea	byte(A) x byte(ea) word(A)	MULY A,ea	word(A) x word(ea) long(A)

