Appendix 2

Instructions

The table in this appendix provides the different assembly language formats of the instructions for the Intel 8086 processor discussed in this book. It also provides machine language encoding information about the opcode and the operands of these instructions.

In this table,

- imm8 is an 8-bit (or 1 byte) immediate data
- imm16 is a 16-bit (1 word) immediate data
- reg8 is an 8-bit (1 byte) register (that means register AL, AH, BL, BH, CL, CH, DL or DH)
- reg16 is a 16-bit register which is not a segment register, the flag register or register IP (that means register AX, BX, CX, DX, SP, BP, SI, or DI)
- Sreg is a segment register (that means register CS, SS, ES, or DS)
- mem8 is a byte memory location
- mem16 is a word memory location, and
- regcd is the 3-bit code of a register operand.

Instructions	Opcode	'mod rm' Byte	Opcode Extension	Exceptions/ Comments
ADD reg8/mem8, reg8	00	yes		
ADD reg16/mem16, reg16	01	yes		
ADD reg8, mem8	02	yes		
ADD reg16, mem16	03	yes		
ADD AL, imm8	04	no		
ADD AX, imm16	05	no		
ADD reg8/mem8, imm8	80	yes	000	Except AL
ADD reg16/momo16, imm16	81	yes	000	Except AX
ADD reg16/mem16, imm8	83	yes	000	Except AX
CALL Selector: Offset	9A	no		Transfer instruction
CALL Offset	E8	no		Relative addressing
CALL reg16/mem16	FF	yes	010	Transfer instruction
CBW	98	no		
CMP reg8/mem8, reg8	38	yes		
CMP reg16/mem16, reg16	39	yes		
CMP reg8, mem8	3A	yes		
CMP reg16, mem16	3B	yes		
CMP AL, imm8	3C	no		
CMP AX, imm16	3D	no		
CMP reg8/mem8, imm8	80	yes	111	Except AL
CMP reg16/mem16, imm16	81	yes	111	Except AX
CMP reg16/mem16, imm8	83	yes	111	Except AX
CWD	99	no		
DEC reg16	48 + regcd			Short Form
DEC reg8/mem8	FE	yes	001	
DEC reg16/mem16	FF	yes	001	
DIV reg8/mem8	F6	yes	110	
DIV reg16/mem16	F7	yes	110	
IDIV reg8/mem8	F6	yes	111	
IDIV reg16/mem16	F7	yes	111	

Instructions	Opcode	'mod rm' Byte	Opcode Extension	Exceptions/ Comments
IMUL reg8/mem8	F6	yes	101	
IMUL reg16/mem16	F7	yes	101	
INC reg16	40 + regcd	no		Short Form
INC reg8/mem8	FE	yes	000	
INC reg16/mem16	FF	yes	000	
INT 3	CC	no		
INT type imm8	CD	no		Except type 3
JB/JNAE ShortOffset	72	no		Relative addressing
JBE/JNA ShortOffset	76	no		-
JCXZ ShortOffset	E3	no		-
JE/JZ ShortOffset	74	no		-
JG/JNLE ShortOffset	7F	no		-
JGE/JNL ShortOffset	7D	no		-
JL/JNGE ShortOffset	7C	no		-
JLE/JNG ShortOffset	7E	no		-
JMP ShortOffset	EB	no		-
JMP Offset	E9	no		-
JMP reg16/mem16	FF	yes	100	Transfer Instruction
JMP Selector: Offset	EA	no		-
JNB/JAE ShortOffset	73	no		Relative addressing
JNBE/JA ShortOffset	77	no		-
JNE/JNZ ShortOffset	75	no		-
JNO ShortOffset	71	no		-
JNP/JPO ShortOffset	7B	no		-
JNS ShortOffset	79	no		-
JO ShortOffset	70	no		-
JP/JPE ShortOffset	7A	no		-
JS ShortOffset	78	no		-
LOOP ShortOffset	E2	no		-
LOOPNZ/LOOPNE ShortOffset	E1	no		-

Instructions	Opcode	'mod rm' Byte	Opcode Extension	Exceptions/ Comments
LOOPZ/LOOPE ShortOffset	E0	no		-
MOV reg8/mem8, reg8	88	yes		Except AL when mem8 is in direct mode
MOV mem8, AL	A2	no		mem8 is in direct mode
MOV reg16/mem16, reg16	89	yes		Except AX when mem16 is in direct mode
MOV mem16, AX	A3	no		mem16 is in direct mode
MOV reg8, mem8	8A	yes		Except AL when mem8 is in direct mode
MOV AL, mem8	A0	no		mem8 is in direct mode
MOV reg16, mem16	8B	yes		Except AX when mem16 is in direct mode
MOV AX, mem16	A1	no		mem16 is in direct mode
MOV reg8, imm8	B0 + regcd	no		
MOV reg16, imm16	B8 + regcd	no		
MOV Sreg, reg16/mem16	8E	yes		Except CS: reg-field = Sreg; r/m-field = reg16/mem16
MOV reg16/mem16, Sreg	8C	yes		reg-field = Sreg; r/m-field = reg16/mem16
MOV mem8, imm8	C6	yes	000	
MOV mem16, imm16	C7	yes	000	
MUL reg8/mem8	F6	yes	100	
MUL reg16/mem16	F7	yes	100	
NEG mem8/reg8	F6	yes	011	
NEG mem16/reg16	F7	yes	011	
POP mem16	8F	yes	000	
POP reg16	58 + regcd	no		
POP Sreg	000regcd111	no		Except CS
PUSH mem16	FF	yes	110	
PUSH reg16	50 + regcd	no		Short Form
PUSH Sreg	000regcd110	no		Short Form
RET	C3	no		

Instructions	Opcode	'mod rm' Byte	Opcode Extension	Exceptions/ Comments
RET imm16	C2	no		
RETF imm16	CA	no		
RETF	CB	no		
SUB mem8/reg8, reg8	28	yes		
SUB mem16/reg16, reg16	29	yes		
SUB reg8, mem8	2A	yes		
SUB reg16, mem16	2B	yes		
SUB AL, imm8	2C	no		Short Form
SUB AX, imm16	2D	no		Short Form
SUB reg8/mem8, imm8	80	yes	101	Except AL
SUB reg16/mem16, imm16	81	yes	101	Except AX
SUB reg16/mem16, imm8	83	yes	101	Except AX
XCHG mem8/reg8, reg8	86	yes		
XCHG reg8, mem8	86	yes		
XCHG mem16, reg16	87	yes		
XCHG reg16, mem16/reg16	87	yes		2^{nd} reg16 ≠ AX
XCHG reg16, AX	90 + regcd	no		reg16 ≠ AX