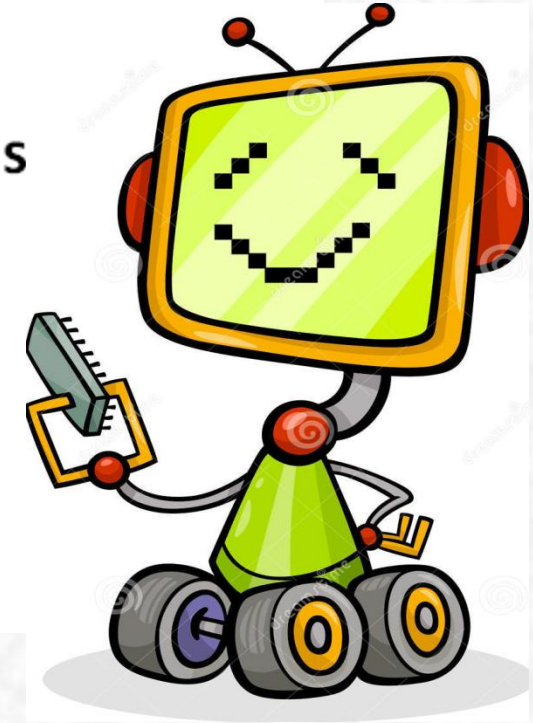
Information Technology Department
Faculty of Computers and Information
Mansoura University

Chapter 4:

Data Movement Instructions

Sara El-Metwally, Ph.D.
Computer Science Department
Faculty of Computers and Information,
Mansoura University, Egypt.

Email: sara.elmetwally.2007@gmail.com
Office: Faculty of CIS, third floor



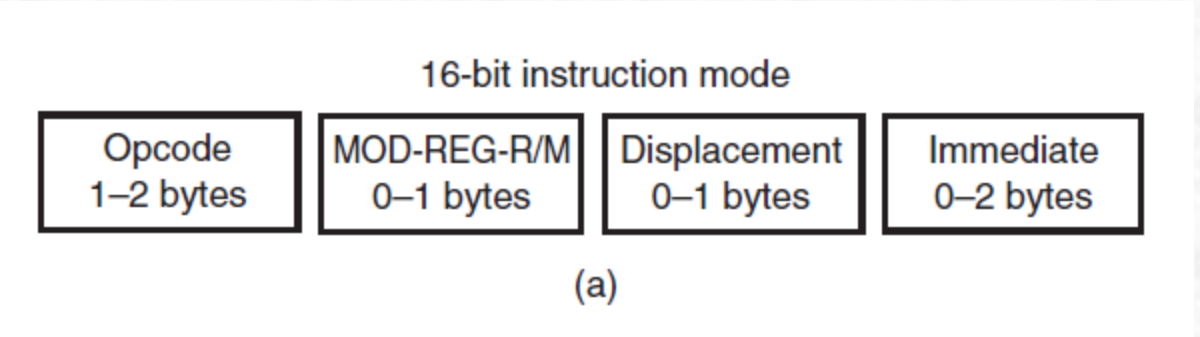
Machine Language

- Machine language is the native binary code that the microprocessor understands and uses as its instructions to control its operation.

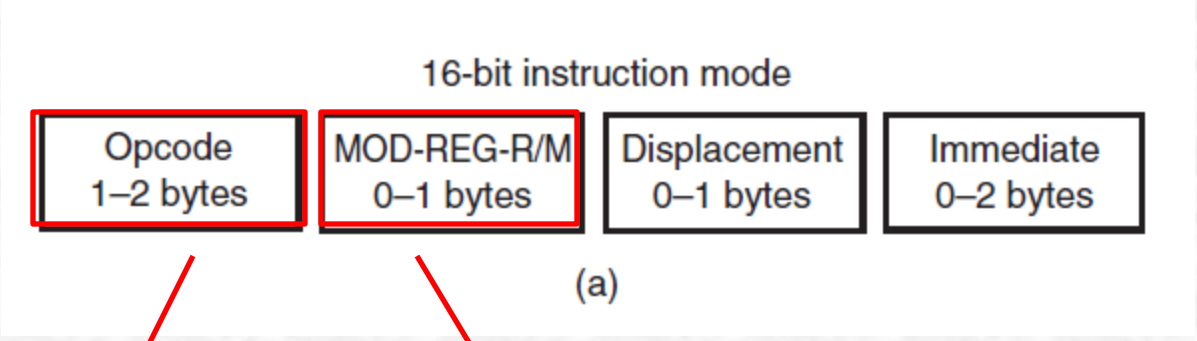
- Machine language instructions for the 8086 through the Core2 vary in length from 1 to as many as 13 bytes.
- There are well over 100,000 variations of machine language instructions and there is no complete list of these variations.

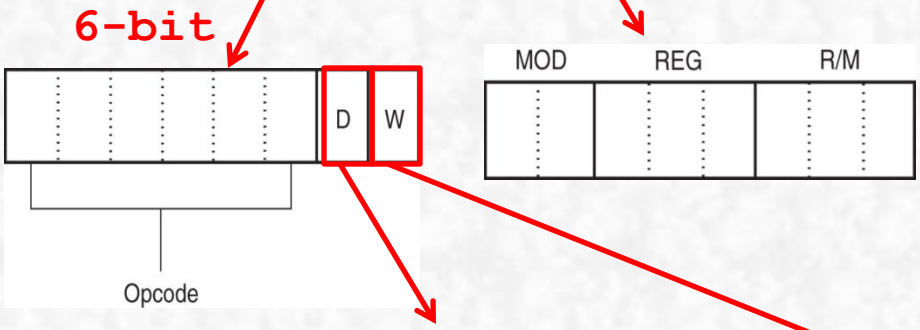
Machine Language

- Instructions for the 8086 through the 80286 are 16-bit mode instructions that take the form found in Figure 4–1(a).



Machine Language





Direction of the data flow

D=1, data flow to REG from R/M

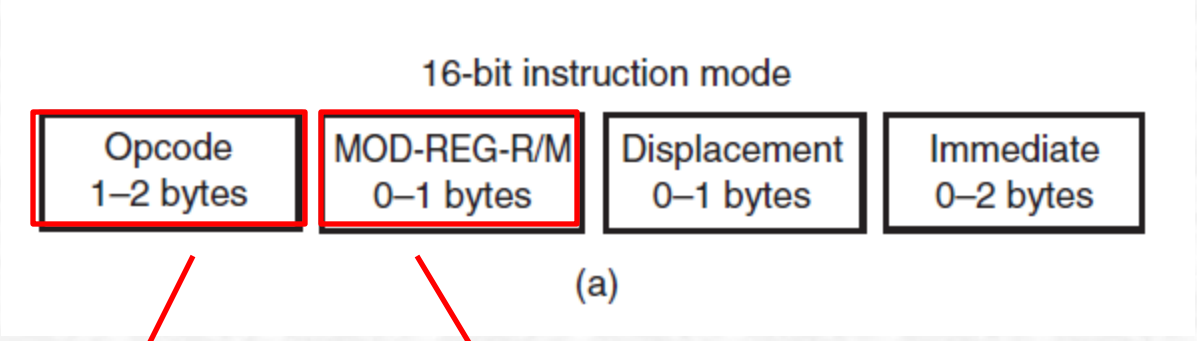
D=0, data flow from REG to R/M

Size of the data flow

W=1, Word and above data flow

W=0, byte data flow

Machine Language



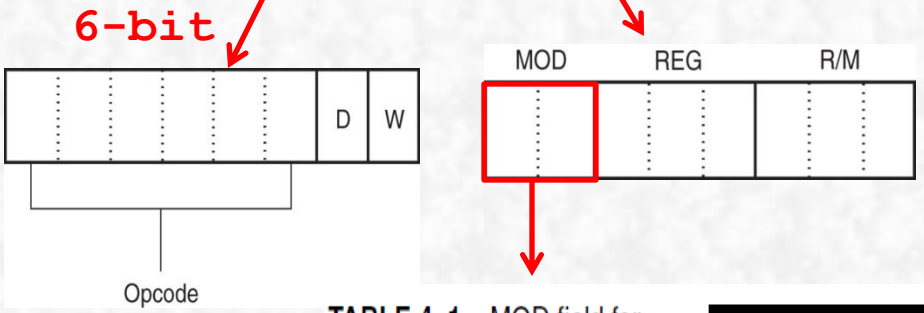


TABLE 4–1 MOD field for the 16-bit instruction mode.

MOD	*Function*
00	No displacement
01	8-bit sign-extended displacement
10	16-bit signed displacement
11	R/M is a register

Machine Language

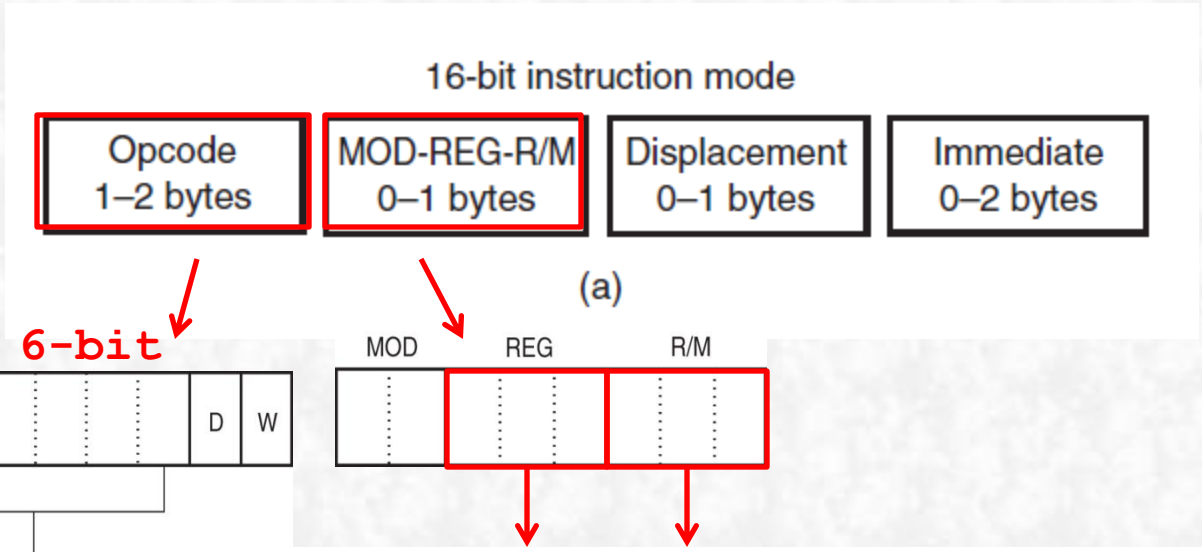
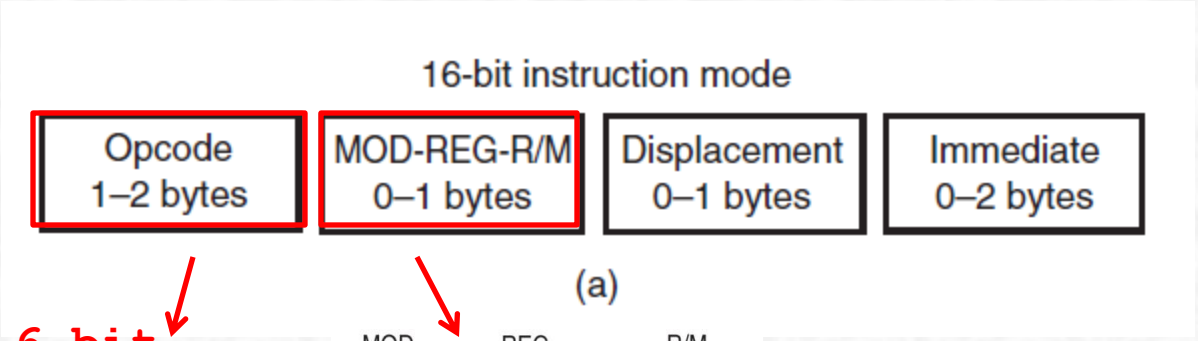


TABLE 4–3 REG and R/M (when) MOD = 11 assignments.

Code	W = 0 *(Byte)*	W = 1 *(Word)*
000	AL	AX
001	CL	CX
010	DL	DX
011	BL	BX
100	AH	SP
101	CH	BP
110	DH	SI
111	BH	DI

Machine Language



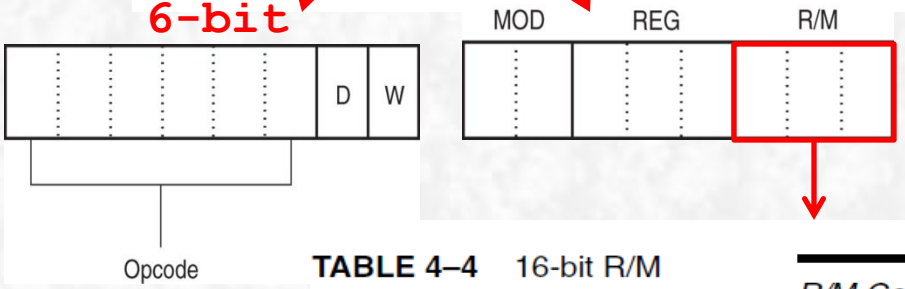


TABLE 4–4 16-bit R/M memory-addressing modes.

R/M Code	*Addressing Mode*
000	DS:[BX+SI]
001	DS:[BX+DI]
010	SS:[BP+SI]
011	SS:[BP+DI]
100	DS:[SI]
101	DS:[DI]
110	SS:[BP]*
111	DS:[BX]

*Note: See text section, Special Addressing Mode.

Machine Language (MOV BP,SP)

- MOV(Opcode = 1 0 0 0 1 0)

MOV BP,SP = 8BEC

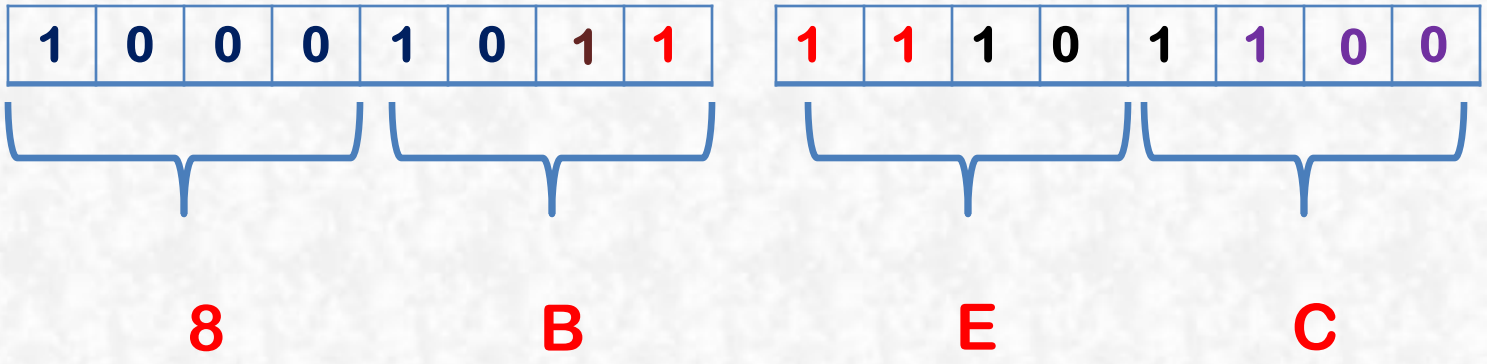


TABLE 4–3 REG and R/M (when) MOD = 11 assignments.

Code	W = 0 *(Byte)*	W = 1 *(Word)*
000	AL	AX
001	CL	CX
010	DL	DX
011	BL	BX
100	AH	SP
101	CH	BP
110	DH	SI
111	BH	DI

Machine Language (MOV AL, BL)

- MOV(Opcode = 1 0 0 0 1 0)

MOV AL,BL = 8AC3

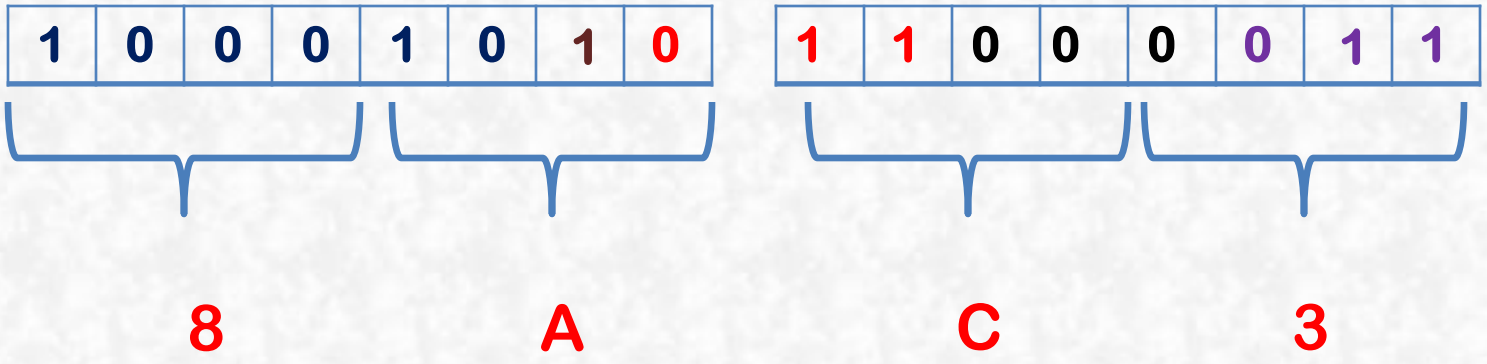


TABLE 4–1 MOD field for the 16-bit instruction mode.

MOD	*Function*
00	No displacement
01	8-bit sign-extended displacement
10	16-bit signed displacement
11	R/M is a register

TABLE 4–3 REG and R/M (when) MOD = 11 assignments.

Code	W = 0 *(Byte)*	W = 1 *(Word)*
000	AL	AX
001	CL	CX
010	DL	DX
011	BL	BX
100	AH	SP
101	CH	BP
110	DH	SI
111	BH	DI

Machine Language (MOV DL,[DI])

- MOV(Opcode = 1 0 0 0 1 0)

MOV DL, [DI] = 8A15

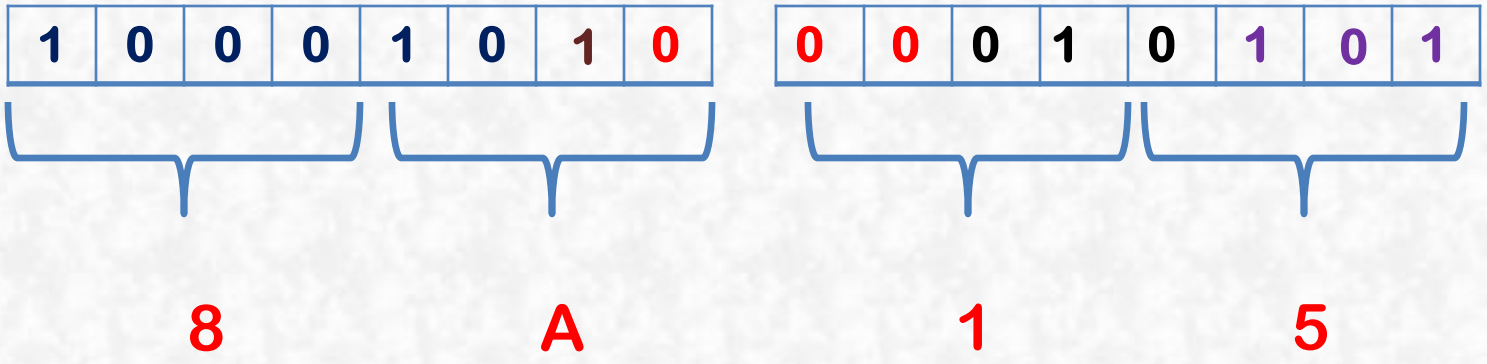


TABLE 4–1 MOD field for the 16-bit instruction mode.

MOD	*Function*
00	No displacement
01	8-bit sign-extended displacement
10	16-bit signed displacement
11	R/M is a register

TABLE 4–4 16-bit R/M memory-addressing modes.

R/M Code	*Addressing Mode*
000	DS:[BX+SI]
001	DS:[BX+DI]
010	SS:[BP+SI]
011	SS:[BP+DI]
100	DS:[SI]
101	DS:[DI]
110	SS:[BP]*
111	DS:[BX]

*Note: See text section, Special Addressing Mode.

TABLE 4–3 REG and R/M (when) MOD = 11 assignments.

Code	W = 0 *(Byte)*	W = 1 *(Word)*
000	AL	AX
001	CL	CX
010	DL	DX
011	BL	BX
100	AH	SP
101	CH	BP
110	DH	SI
111	BH	DI

Machine Language (MOV DL,[DI+1])

TABLE 4–1 MOD field for the 16-bit instruction mode.

MOD	*Function*
00	No displacement
01	8-bit sign-extended displacement
10	16-bit signed displacement
11	R/M is a register

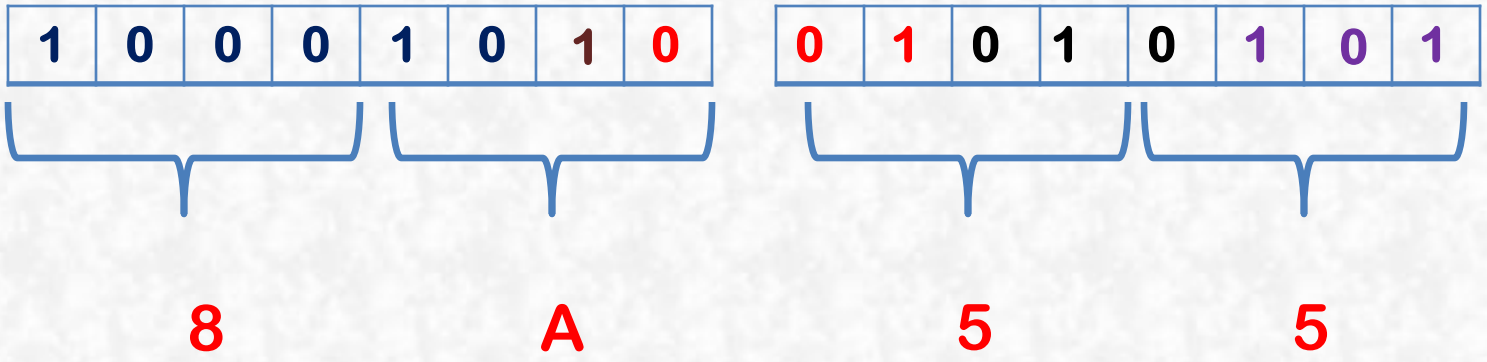
TABLE 4–4 16-bit R/M memory-addressing modes.

R/M Code	*Addressing Mode*
000	DS:[BX+SI]
001	DS:[BX+DI]
010	SS:[BP+SI]
011	SS:[BP+DI]
100	DS:[SI]
101	DS:[DI]
110	SS:[BP]*
111	DS:[BX]

*Note: See text section, Special Addressing Mode.

- MOV(Opcode = 1 0 0 0 1 0)

MOV DL, [DI+1] = 8A5501



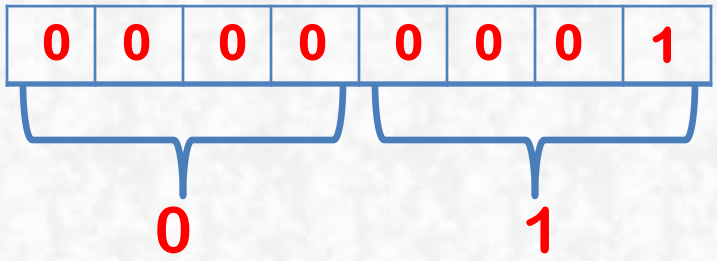
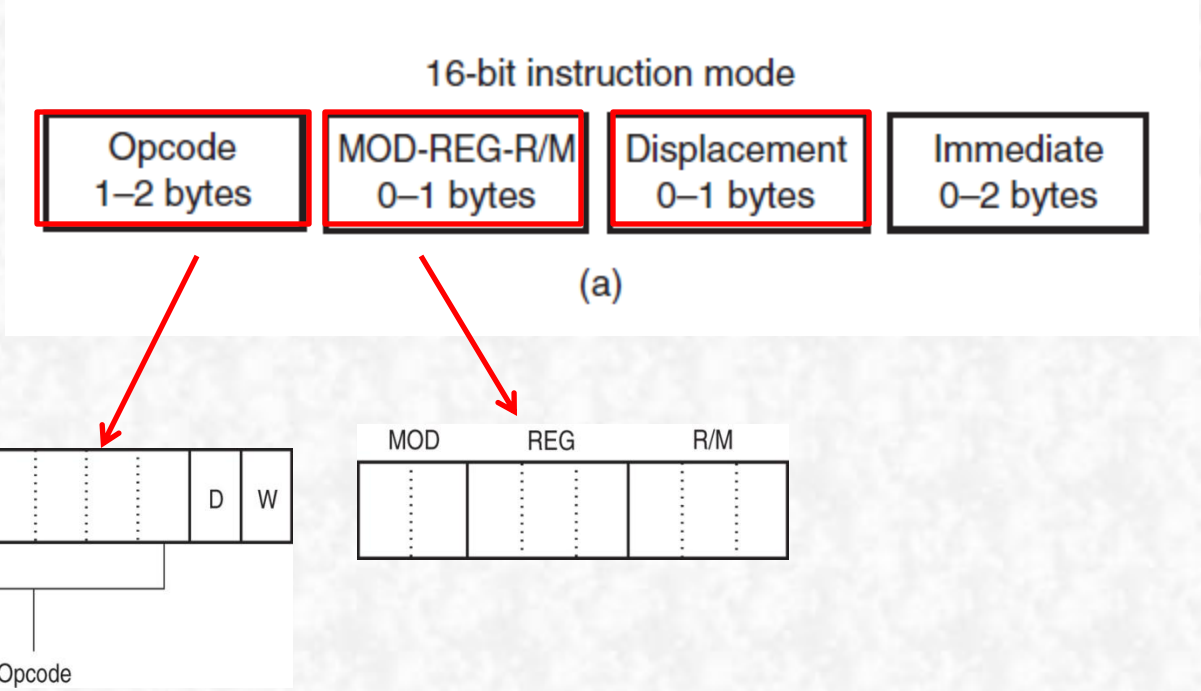


TABLE 4–3 REG and R/M (when) MOD = 11 assignments.

Code	W = 0 *(Byte)*	W = 1 *(Word)*
000	AL	AX
001	CL	CX
010	DL	DX
011	BL	BX
100	AH	SP
101	CH	BP
110	DH	SI
111	BH	DI

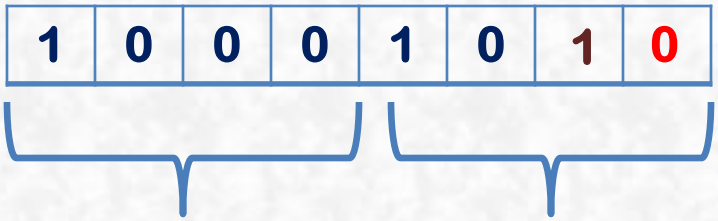
Machine Language



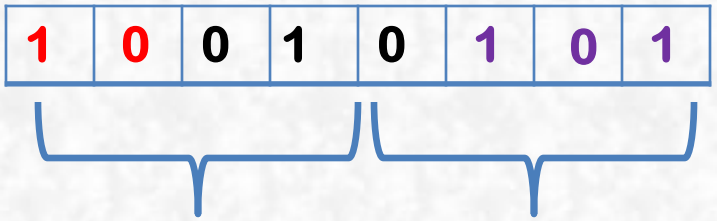
Machine Language (MOV DL,[DI+1000])

- MOV(Opcode = 1 0 0 0 1 0)

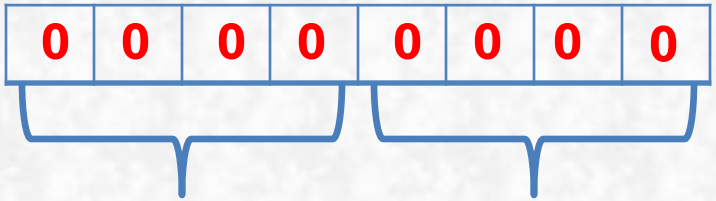
MOV DL, [DI+1000] = 8A950010



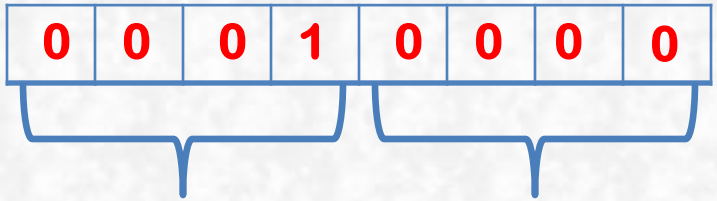
8 A



9 5

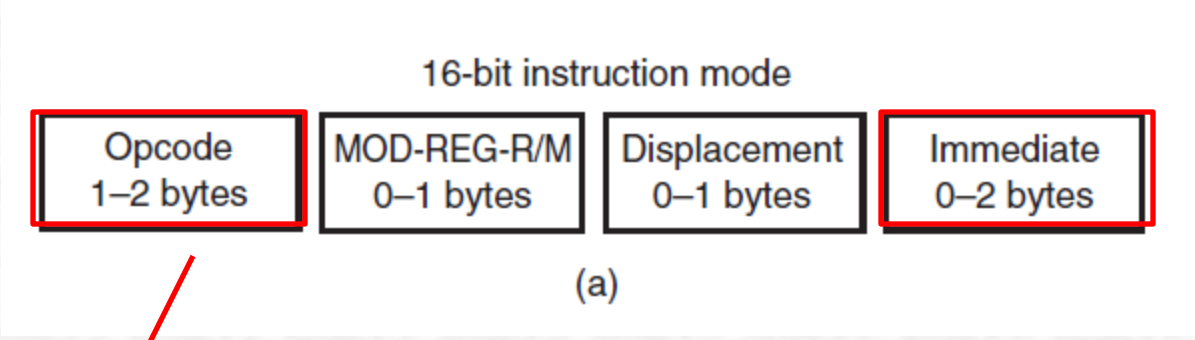


0 0



1 0

Machine Language (Immediate to Register)

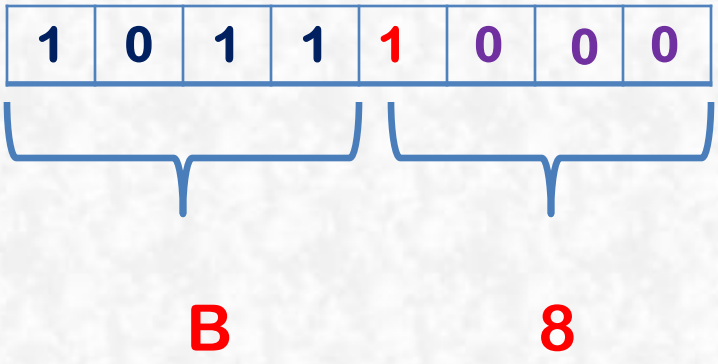


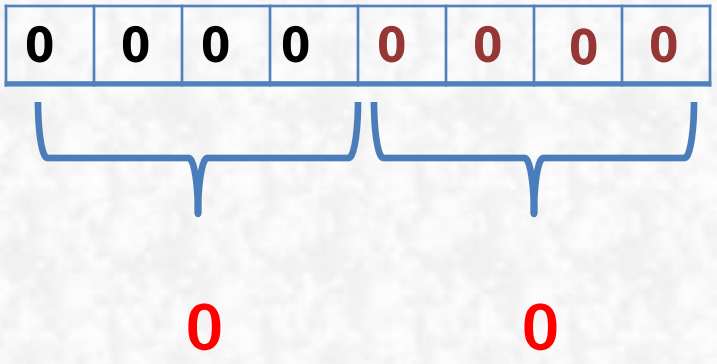
1	0	1	1	w	reg

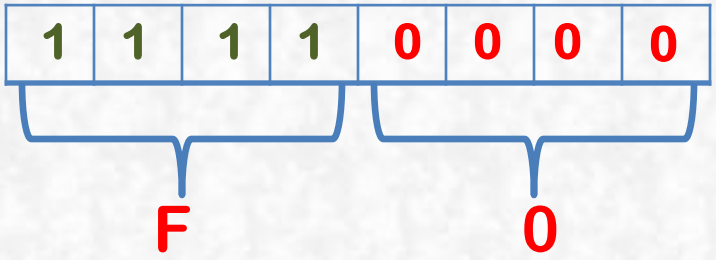
Machine Language (MOV AX,F000H)

- MOV(Opcode = 1 0 11)

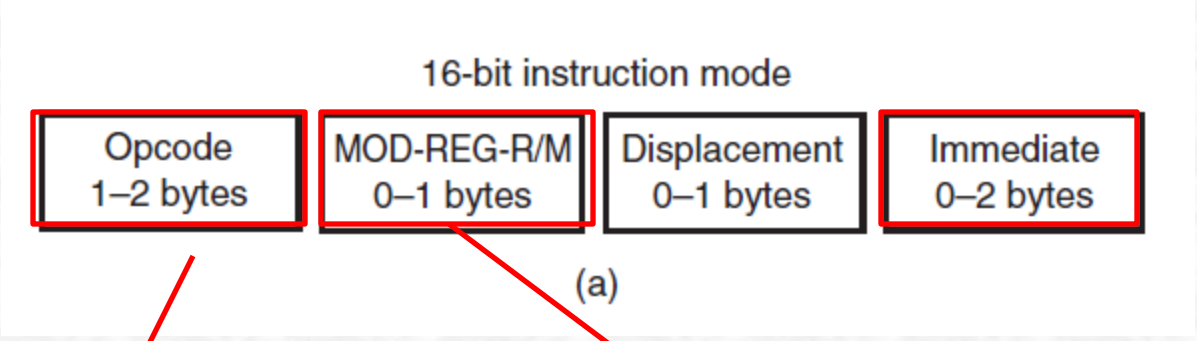
MOV AX, F000H = B800F0







Machine Language (Immediate to R/M)



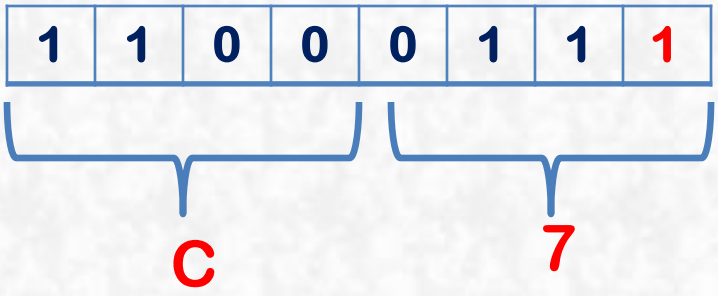
1	1	0	0	0	1	1	W

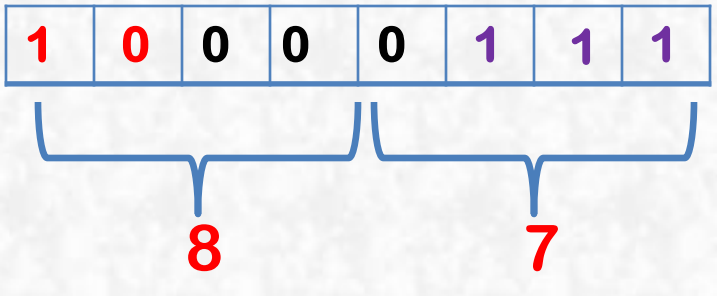
Mod	0	0	0	R/M

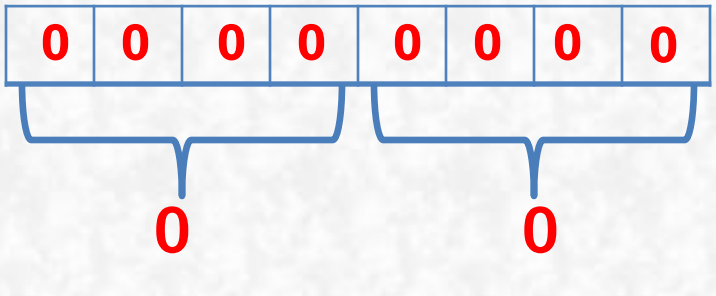
Machine Language

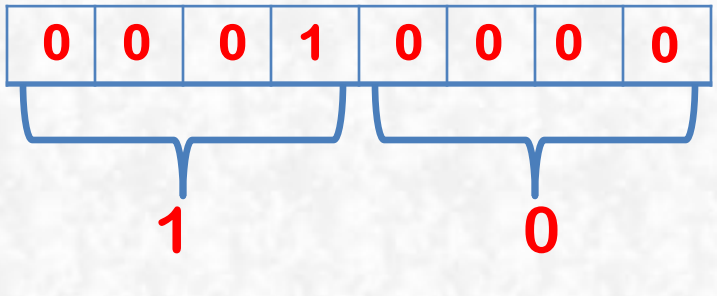
(MOV WORD PTR [BX+1000H], 1234H)

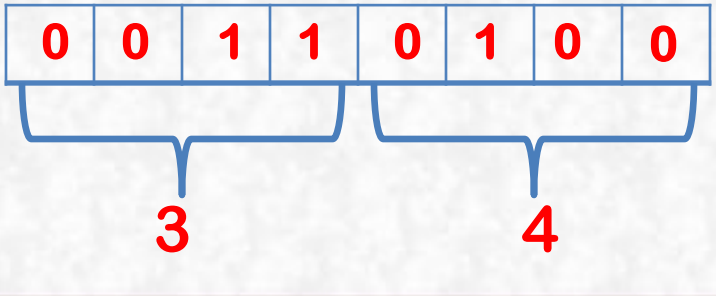
- MOV(Opcode = 1 1 0 0 0 1 1)

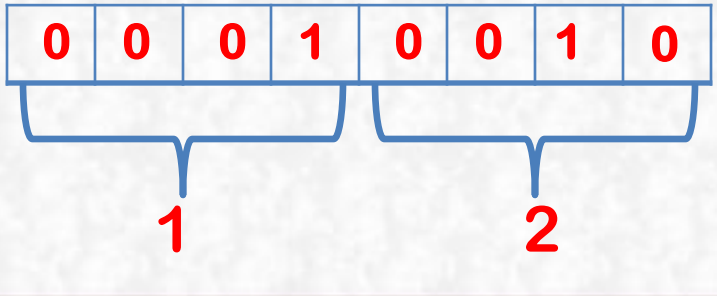




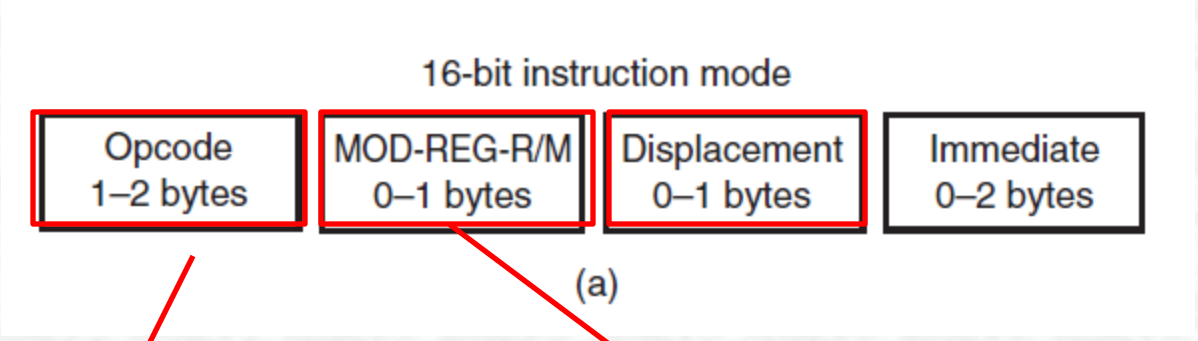








Machine Language (MOV [1000H],DL)



1	0	0	0	1	0	D	W

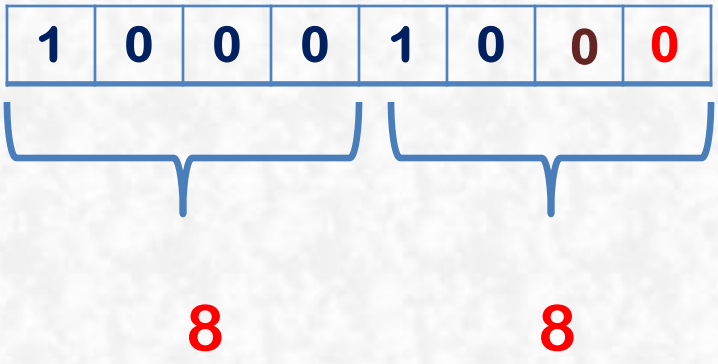
00	R	E	G	110

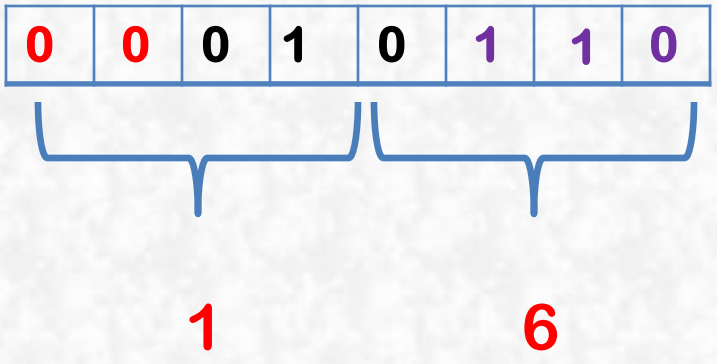
Machine Language (MOV [1000H],DL)

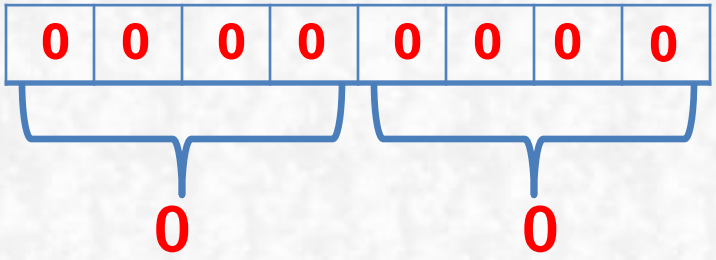
MOV Num,DL

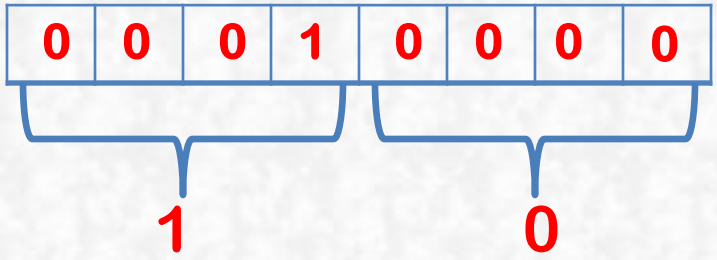
- MOV(Opcode = 1 0 0 0 1 0)

MOV [1000H],DL = 88160010





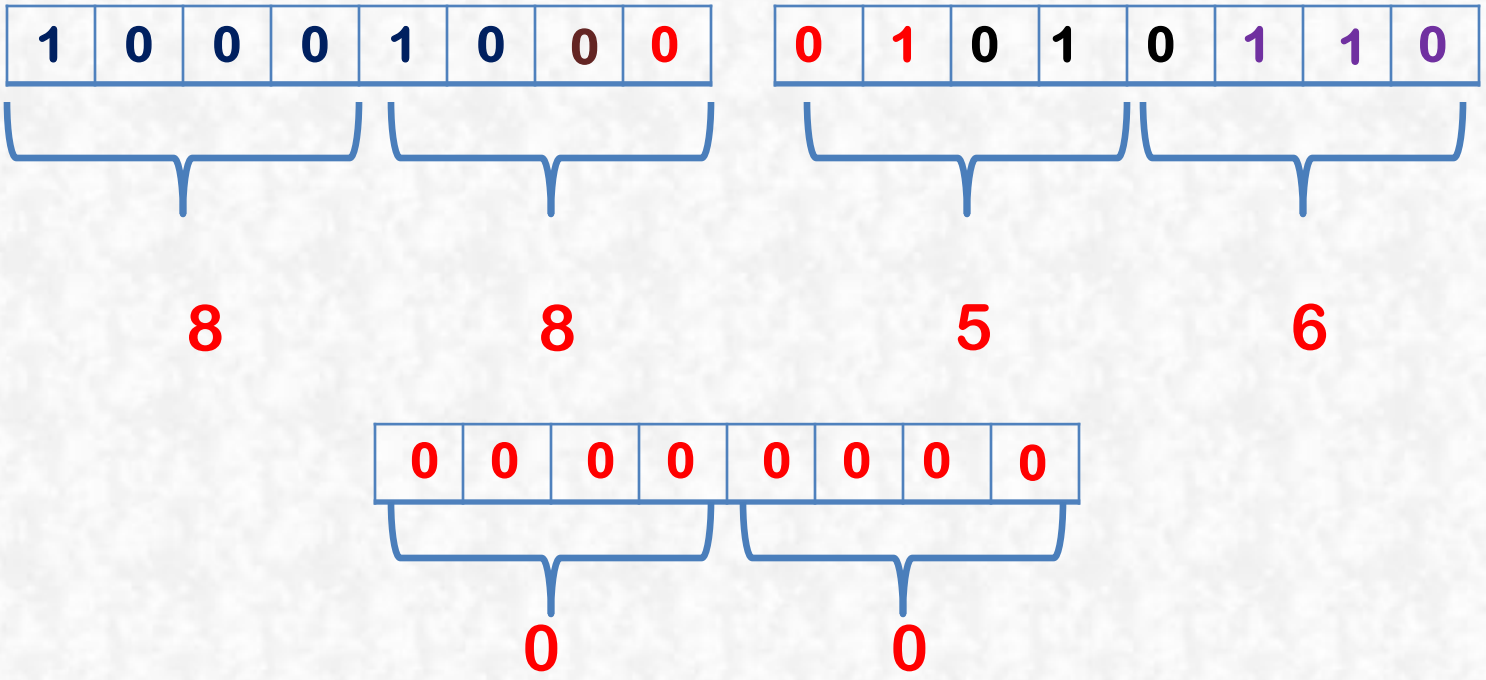




Machine Language (MOV [BP],DL)

- MOV(Opcode = 1 0 0 0 1 0)

MOV [BP],DL = 885600



Machine Language

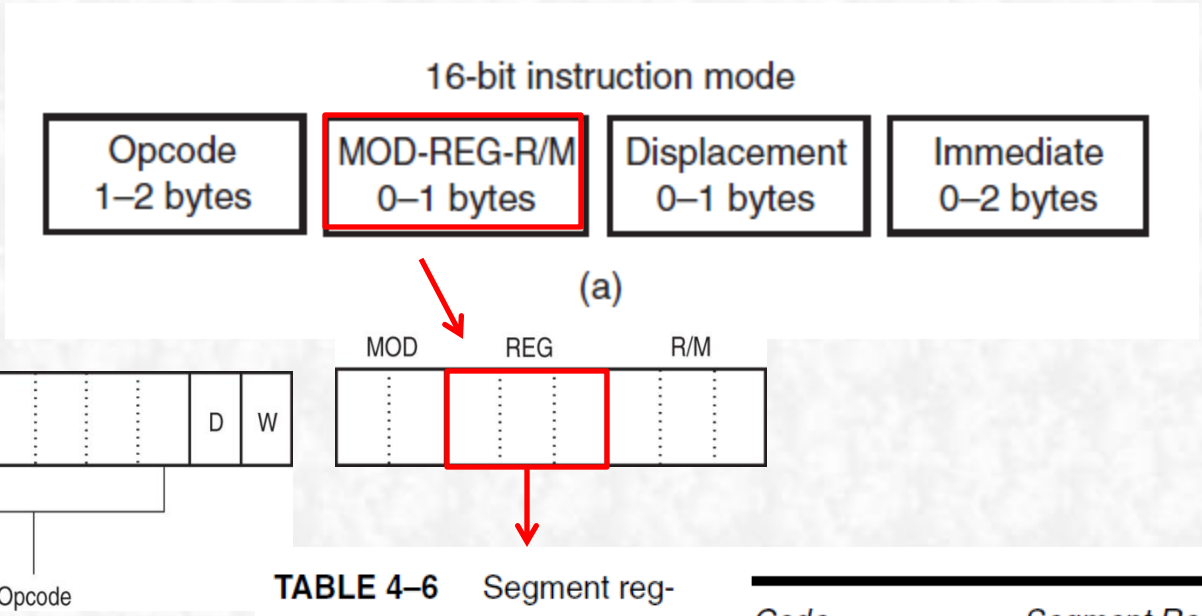


TABLE 4–6 Segment register selection.

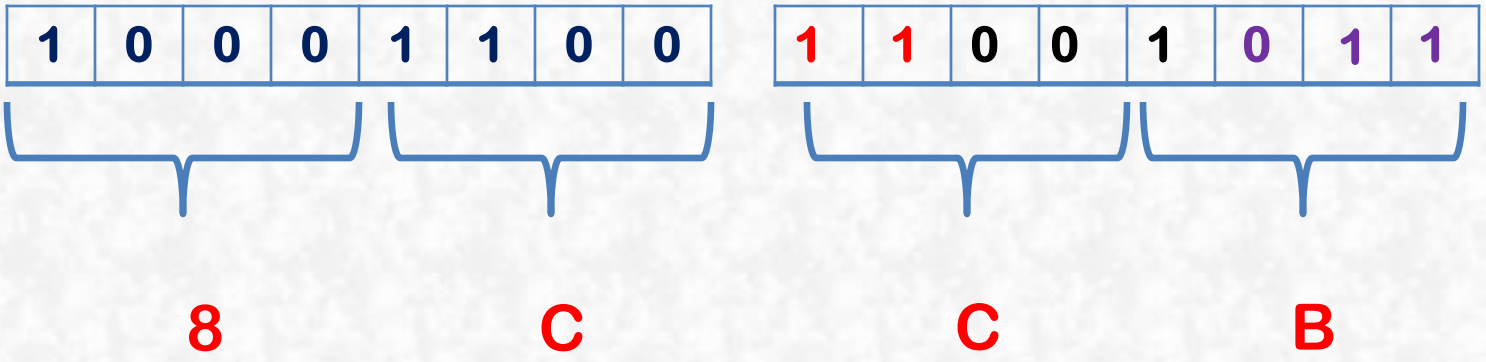
Code	*Segment Register*
000	ES
001	CS*
010	SS
011	DS
100	FS
101	GS

*Note: MOV CS,R/M and POP CS are not allowed.

Machine Language (MOV BX,CS)

- MOV(Opcode = 1 0 0 0 1 1 0 0)

MOV BX,CS = 8CCB



More Machine Language codes (ADD)

R/M with R

0	0	0	0	0	0	d	W

Mod	r	e	g	R/M

Immediate to R/M

1	0	0	0	0	0	s	W

Mod	0	0	0	R/M

Data

Immediate to AC

0	0	0	0	0	1	0	W

Data

More Machine Language codes (ADD AX,4789H)

R/M with R

0	0	0	0	0	0	d	W

Mod	r	e	g	R/M

Immediate to R/M

1	0	0	0	0	0	s	W

Mod	0	0	0	R/M

Data

058947

Immediate to AC

0	0	0	0	0	1	0	W

Data

More Machine Language codes (INC)

R/M

INC AL

FE C0