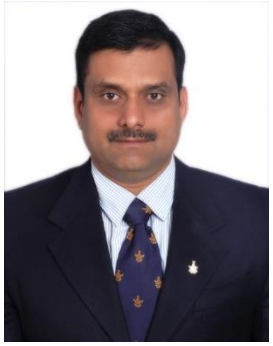
Microprocessors and Interfaces: 2021-22
Lecture 6
8086 Addressing Modes and OP-Code : Examples

By Dr. Sanjay Vidhyadharan



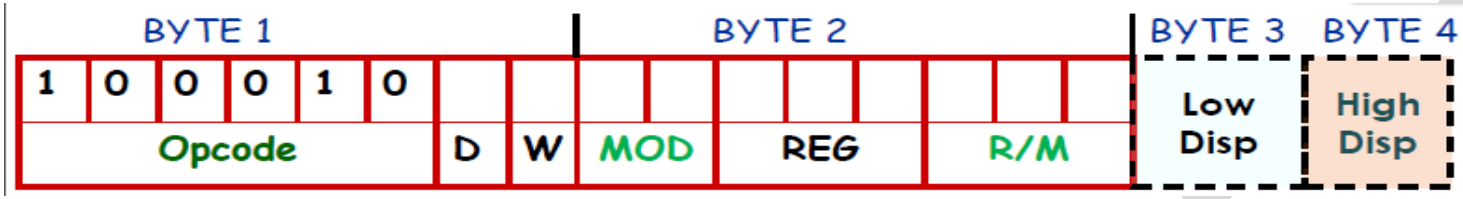
INSTRUCTION FORMAT

BYTE 1								BYTE 2								BYTE 3	BYTE 4
																LOW DISP.	HIGH DISP.
OPCODE						D	W	MOD		REG			R/M				

BYTE 1						BYTE 2	BYTE 3
				1 BIT	3 BITS	LOW DISP.	HIGH DISP.
OPCODE				W	REG		

Register Addressing

MOV AX,BX



1 0 0 0 1 0 0 1 1 1 0 1 1 0 0 0

89D8

1 0 0 0 1 0 1 1 1 1 0 0 0 0 1 1

8BC3

MOD	Interpretation
00	Memory mode with no displacement follows except for 16-bit Displacement when R/M = 110
01	Memory mode with 8-bit displacement
10	Memory mode with 16-bit displacement
11	Register mode (no displacement)

MOV = Move
Register/Memory to/from Register
Immediate to Register/Memory

76543210
100010 dw
1100011 w

REG	W = 0	W = 1
000	AL	AX
001	CL	CX
010	DL	DX
011	BL	BX

Immediate Addressing

MOV CX, $AD4C_H$

BYTE 1						BYTE 2	BYTE 3
				1 BIT	3 BITS	LOW DISP.	HIGH DISP.
OPCODE				W	REG		

1	0	1	1	1	0	0	1

4C AD

B94CAD

MOV = Move
Register/Memory to/from Register
Immediate to Register/Memory
Immediate to Register

76543210
100010 dw
1100011 w
1011 w reg

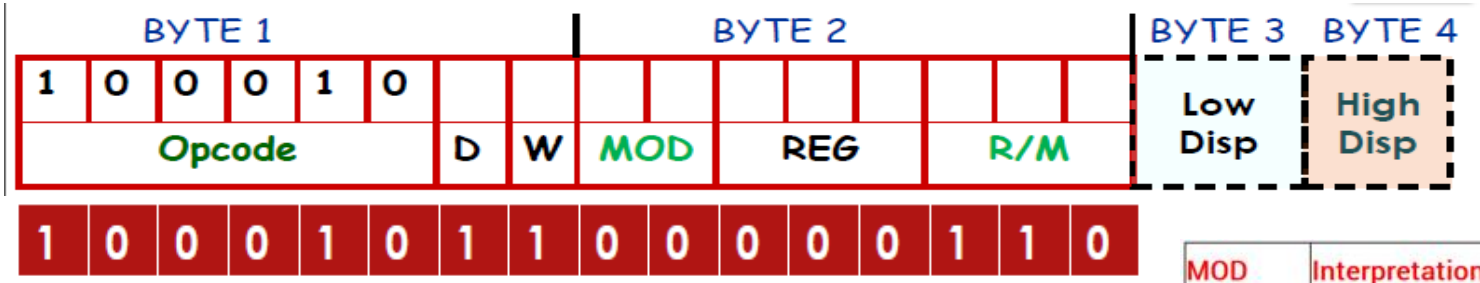
REG	W = 0	W = 1
000	AL	AX
001	CL	CX
010	DL	DX
011	BL	BX

Little vs Big Endian

Little		vs	Big Endian (5627)	
00000	27		00000	56
00001	56		00001	27

Direct Addressing

MOV AX, [1234_H]



1 0 0 0 1 0 1 1 0 0 0 0 0 1 1 0

88 06 34 12

MOD	Interpretation
00	Memory mode with no displacement follows except for 16-bit Displacement when R/M = 110
01	Memory mode with 8-bit displacement
10	Memory mode with 16-bit displacement
11	Register mode (no displacement)

Operands	Memory Operands			Register Operands	
	No Displacement	Displacement 8-bit	Displacement 16-bit		
MOD / R/M	00	01	10	11	
				W = 0	W = 1
000	(BX) + (SI)	(BX) + (SI) + D8	(BX) + (SI) + D16	AL	AX
001	(BX) + (DI)	(BX) + (DI) + D8	(BX) + (DI) + D16	CL	CX
010	(BP) + (SI)	(BP) + (SI) + D8	(BP) + (SI) + D16	DL	DX
011	(BP) + (DI)	(BP) + (DI) + D8		BL	BX
100	(SI)	(SI) + D8	(SI) + D16	AH	SP
101	(DI)	(DI) + D8	(DI) + D16	CH	BP
110	D16	(BP) + D8	(BP) + D16	DH	SI
111	(BX)	(BX) + D8	(BX) + D16	BH	DI

REG	W = 0	W = 1
000	AL	AX
001	CL	CX
010	DL	DX
011	BL	BX

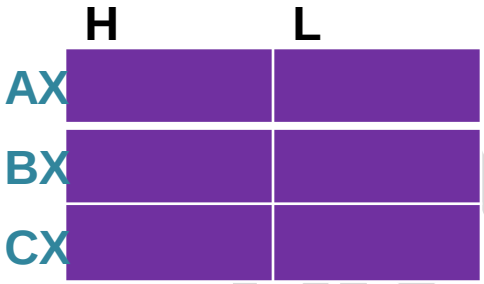
Register indirect Addressing:

MOV [BX], CL - Register Indirect addressing

- Transfers a byte or word between a register and a memory location addressed by an index or base register.

Lookup table with register indirect addressing

```
CS: 0200, DS: 0100
        MOV AL,0
        MOV CL,04H
        MOV BX,0000H
AGAIN:  MOV [BX],AL
        INC AL
        INC BX
        LOOP AGAIN
```

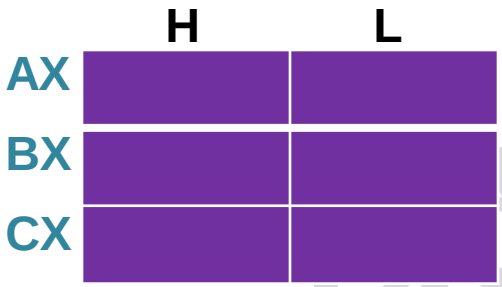


Physical Address	Value

DS = 0100

Lookup table with register indirect addressing

```
CS: 0200, DS: 0100
        MOV AL,0
        MOV CL,04H
        MOV BX,0000H
AGAIN:  MOV [BX],AL
        INC AL
        INC BX
        LOOP AGAIN
```



Physical Address	Value

DS = 0100

Lookup table with register indirect addressing

CS: 0200, DS: 0100

```
        MOV AL,0
        MOV CL,04H
        MOV BX,0000H
AGAIN:  MOV [BX],AL
        INC AL
        INC BX
        LOOP AGAIN
```

Physical Address	Value

DS = 0100

Lookup table with register indirect addressing

```
CS: 0200, DS: 0100
        MOV AL,0
        MOV CL,04H
        MOV BX,0000H
AGAIN:  MOV [BX],AL
        INC AL
        INC BX
        LOOP AGAIN
```

	H	L
AX	xx	00
BX	00	00
CX	xx	04

DS*10H+BX	01000

Physical address

Physical Address	Value
01000	00

DS = 0100

Lookup table with register indirect addressing

```
CS: 0200, DS: 0100
        MOV AL,0
        MOV CL,04H
        MOV BX,0000H
AGAIN:  MOV [BX],AL
        INC AL
        INC BX
        LOOP AGAIN
```

	H	L
AX	xx	01
BX	00	00
CX	xx	04

DS*10H+BX	01000

Physical address

Physical Address	Value
01000	00

DS = 0100

Lookup table with register indirect addressing

```
CS: 0200, DS: 0100
        MOV AL,0
        MOV CL,04H
        MOV BX,0000H
AGAIN:  MOV [BX],AL
        INC AL
        INC BX
        LOOP AGAIN
```

	H	L
AX	xx	01
BX	00	01
CX	xx	04

DS*10H+BX	01000

Physical address

Physical Address	Value
01000	00

DS = 0100

Lookup table with register indirect addressing

```
CS: 0200, DS: 0100
        MOV AL,0
        MOV CL,04H
        MOV BX,0000H
AGAIN:  MOV [BX],AL
        INC AL
        INC BX
        LOOP AGAIN
```

	H	L
AX	xx	01
BX	00	01
CX	xx	03

DS*10H+BX	01000

Physical address

Physical Address	Value
01000	00

DS = 0100

Lookup table with register indirect addressing

```
CS: 0200, DS: 0100
        MOV AL,0
        MOV CL,04H
        MOV BX,0000H
AGAIN:  MOV [BX],AL
        INC AL
        INC BX
        LOOP AGAIN
```

	H	L
AX	xx	01
BX	00	01
CX	xx	03

DS*10H+BX	01001

Physical address

Physical Address	Value
01000	00
01001	01

DS = 0100

Lookup table with register indirect addressing

```
CS: 0200, DS: 0100
        MOV AL,0
        MOV CL,04H
        MOV BX,0000H
AGAIN:  MOV [BX],AL
        INC AL
        INC BX
        LOOP AGAIN
```

	H	L
AX	xx	02
BX	00	01
CX	xx	03

DS*10H+BX	01001

Physical address

Physical Address	Value
01000	00
01001	01

DS = 0100

Lookup table with register indirect addressing

```
CS: 0200, DS: 0100
        MOV AL,0
        MOV CL,04H
        MOV BX,0000H
AGAIN:  MOV [BX],AL
        INC AL
        INC BX
        LOOP AGAIN
```

	H	L
AX	xx	02
BX	00	02
CX	xx	03

DS*10H+BX	01001

Physical address

Physical Address	Value
01000	00
01001	01

DS = 0100

Lookup table with register indirect addressing

CS: 0200, DS: 0100

```
        MOV AL,0
        MOV CL,04H
        MOV BX,0000H
AGAIN:  MOV [BX],AL
        INC AL
        INC BX
        LOOP AGAIN
```

	H	L
AX	xx	02
BX	00	02
CX	xx	03

DS*10H+BX	01001

Physical address

Physical Address	Value
01000	00
01001	01

DS = 0100

Lookup table with register indirect addressing

CS: 0200, DS: 0100

```
        MOV AL,0
        MOV CL,04H
        MOV BX,0000H
AGAIN:  MOV [BX],AL
        INC AL
        INC BX
        LOOP AGAIN
```

	H	L
AX	xx	02
BX	00	02
CX	xx	02

DS*10H+BX	01001

Physical address

Physical Address	Value
01000	00
01001	01

DS = 0100

Lookup table with register indirect addressing

CS: 0200, DS: 0100

```
        MOV AL,0
        MOV CL,04H
        MOV BX,0000H
AGAIN:  MOV [BX],AL
        INC AL
        INC BX
        LOOP AGAIN
```

	H	L
AX	xx	02
BX	00	02
CX	xx	01

DS*10H+BX	01002

Physical address

Physical Address	Value
01000	00
01001	01
01002	02

DS = 0100

Lookup table with register indirect addressing

CS: 0200, DS: 0100

```
        MOV AL,0
        MOV CL,04H
        MOV BX,0000H
AGAIN:  MOV [BX],AL
        INC AL
        INC BX
        LOOP AGAIN
```

	H	L
AX	xx	03
BX	00	02
CX	xx	01

DS*10H+BX	01002

Physical address

Physical Address	Value
01000	00
01001	01
01002	02

DS = 0100

Lookup table with register indirect addressing

CS: 0200, DS: 0100

```
        MOV AL,0
        MOV CL,04H
        MOV BX,0000H
AGAIN:  MOV [BX],AL
        INC AL
        INC BX
        LOOP AGAIN
```

	H	L
AX	xx	03
BX	00	03
CX	xx	01

DS*10H+BX	01002

Physical address

Physical Address	Value
01000	00
01001	01
01002	02

DS = 0100

Lookup table with register indirect addressing

CS: 0200, DS: 0100

```
        MOV AL,0
        MOV CL,04H
        MOV BX,0000H
AGAIN:  MOV [BX],AL
        INC AL
        INC BX
        LOOP AGAIN
```

	H	L
AX	xx	03
BX	00	03
CX	xx	00

DS*10H+BX	01002

Physical address

Physical Address	Value
01000	00
01001	01
01002	02

DS = 0100

Lookup table with register indirect addressing

CS: 0200, DS: 0100

```
        MOV AL,0
        MOV CL,04H
        MOV BX,0000H
AGAIN:  MOV [BX],AL
        INC AL
        INC BX
        LOOP AGAIN
```

	H	L
AX	xx	03
BX	00	03
CX	xx	00

DS*10H+BX	01003

Physical address

Physical Address	Value
01000	00
01001	01
01002	02
01003	03

DS = 0100

Lookup table with register indirect addressing

CS: 0200, DS: 0100

```
        MOV AL,0
        MOV CL,04H
        MOV BX,0000H
AGAIN:  MOV [BX],AL
        INC AL
        INC BX
        LOOP AGAIN
```

	H	L
AX	xx	04
BX	00	03
CX	xx	00

DS*10H+BX	01003

Physical address

Physical Address	Value
01000	00
01001	01
01002	02
01003	03

DS = 0100

Lookup table with register indirect addressing

CS: 0200, DS: 0100

```
        MOV AL,0
        MOV CL,04H
        MOV BX,0000H
AGAIN:  MOV [BX],AL
        INC AL
        INC BX
        LOOP AGAIN
```

	H	L
AX	xx	04
BX	00	04
CX	xx	00

DS*10H+BX	01003

Physical address

Physical Address	Value
01000	00
01001	01
01002	02
01003	03

DS = 0100

Lookup table with register indirect addressing

CS: 0200, DS: 0100

```
        MOV AL,0
        MOV CL,04H
        MOV BX,0000H
AGAIN:  MOV [BX],AL
        INC AL
        INC BX
        LOOP AGAIN
```

Exit from the loop

	H	L
AX	xx	04
BX	00	04
CX	xx	00

DS*10H+BX	01003

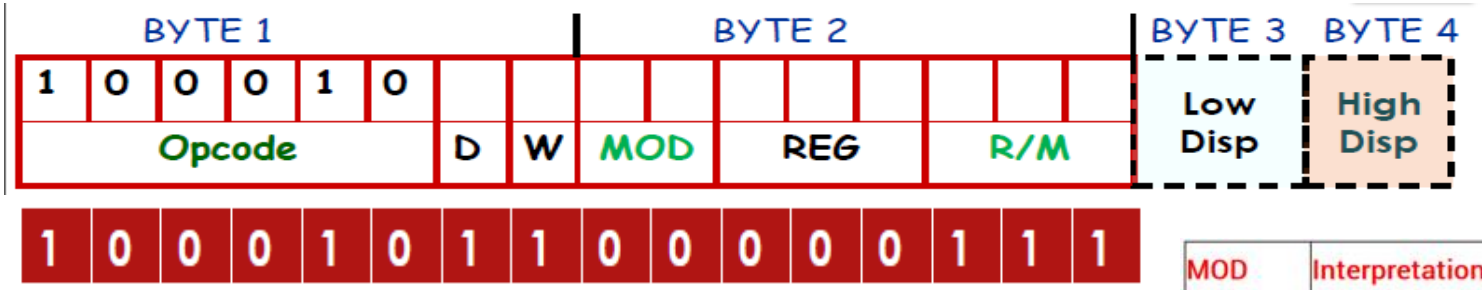
Physical address

Physical Address	Value
01000	00
01001	01
01002	02
01003	03

DS = 0100

Register Indirect Addressing

MOV AX, [BX]



8B07

MOD	Interpretation
00	Memory mode with no displacement follows except for 16-bit Displacement when R/M = 110
01	Memory mode with 8-bit displacement
10	Memory mode with 16-bit displacement
11	Register mode (no displacement)

Operands	Memory Operands			Register Operands	
	No Displacement	Displacement 8-bit	Displacement 16-bit		
MOD / R/M	00	01	10	11	
				W = 0	W = 1
000	(BX) + (SI)	(BX) + (SI) + D8	(BX) + (SI) + D16	AL	AX
001	(BX) + (DI)	(BX) + (DI) + D8	(BX) + (DI) + D16	CL	CX
010	(BP) + (SI)	(BP) + (SI) + D8	(BP) + (SI) + D16	DL	DX
011	(BP) + (DI)	(BP) + (DI) + D8		BL	BX
100	(SI)	(SI) + D8	(SI) + D16	AH	SP
101	(DI)	(DI) + D8	(DI) + D16	CH	BP
110	D16	(BP) + D8	(BP) + D16	DH	SI
111	(BX)	(BX) + D8	(BX) + D16	BH	DI

REG	W = 0	W = 1
000	AL	AX
001	CL	CX
010	DL	DX
011	BL	BX

Base plus Indexed Addressing

MOV AX, [BX+SI]

BX = 1200_H
SI = 0034_H

(AX) ← DS:1200+34

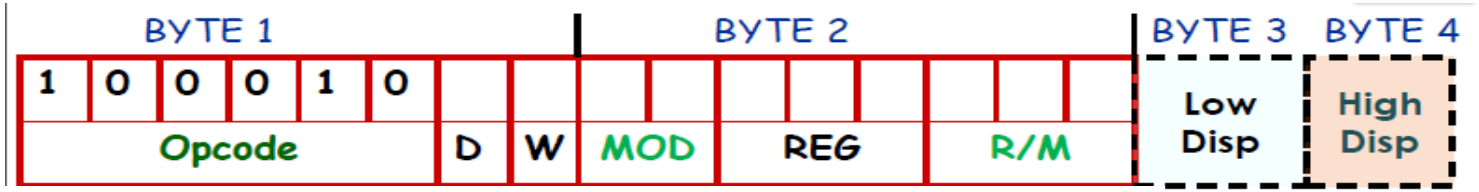
DS = 2000_H
ADDRESS = 20000+1200+34=21234

- **21234_H** 74
- **21235_H** 82

- **AX = 82 74_H**

Base plus Indexed Addressing

MOV AX,[BX+SI]



1	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0

8B00

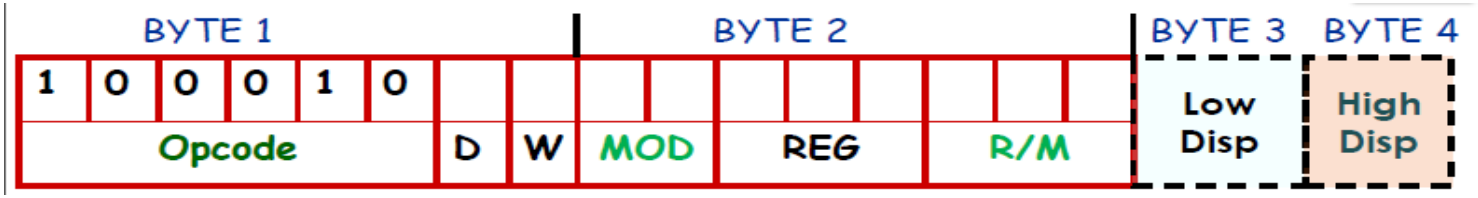
MOD	Interpretation
00	Memory mode with no displacement follows except for 16-bit Displacement when R/M = 110
01	Memory mode with 8-bit displacement
10	Memory mode with 16-bit displacement
11	Register mode (no displacement)

Operands	Memory Operands			Register Operands	
	No Displacement	Displacement 8-bit	Displacement 16-bit		
MOD / R/M	00	01	10	11	
				W = 0	W = 1
000	(BX) + (SI)	(BX) + (SI) + D8	(BX) + (SI) + D16	AL	AX
001	(BX) + (DI)	(BX) + (DI) + D8	(BX) + (DI) + D16	CL	CX
010	(BP) + (SI)	(BP) + (SI) + D8	(BP) + (SI) + D16	DL	DX
011	(BP) + (DI)	(BP) + (DI) + D8		BL	BX
100	(SI)	(SI) + D8	(SI) + D16	AH	SP
101	(DI)	(DI) + D8	(DI) + D16	CH	BP
110	D16	(BP) + D8	(BP) + D16	DH	SI
111	(BX)	(BX) + D8	(BX) + D16	BH	DI

REG	W = 0	W = 1
000	AL	AX
001	CL	CX
010	DL	DX
011	BL	BX

Register Relative Addressing

MOV AX, [BX+34]



1	0	0	0	1	0	1	1	0	1	0	0	0	1	1	1

8B47 34

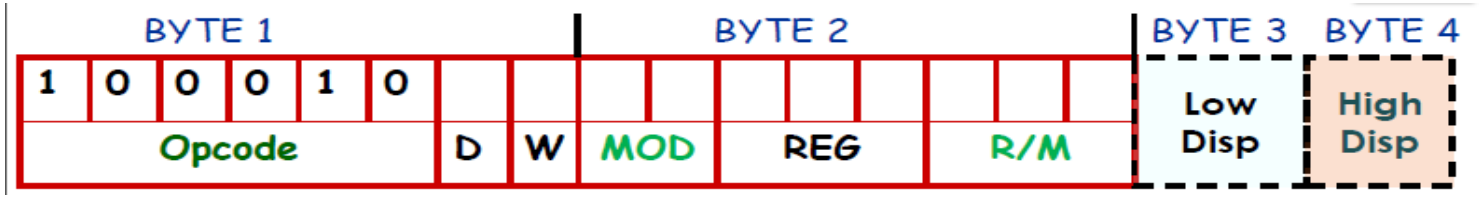
MOD	Interpretation
00	Memory mode with no displacement follows except for 16-bit Displacement when R/M = 110
01	Memory mode with 8-bit displacement
10	Memory mode with 16-bit displacement
11	Register mode (no displacement)

Operands	Memory Operands			Register Operands	
	No Displacement	Displacement 8-bit	Displacement 16-bit		
MOD / R/M	00	01	10	11	
				W = 0	W = 1
000	(BX) + (SI)	(BX) + (SI) + D8	(BX) + (SI) + D16	AL	AX
001	(BX) + (DI)	(BX) + (DI) + D8	(BX) + (DI) + D16	CL	CX
010	(BP) + (SI)	(BP) + (SI) + D8	(BP) + (SI) + D16	DL	DX
011	(BP) + (DI)	(BP) + (DI) + D8		BL	BX
100	(SI)	(SI) + D8	(SI) + D16	AH	SP
101	(DI)	(DI) + D8	(DI) + D16	CH	BP
110	D16	(BP) + D8	(BP) + D16	DH	SI
111	(BX)	(BX) + D8	(BX) + D16	BH	DI

REG	W = 0	W = 1
000	AL	AX
001	CL	CX
010	DL	DX
011	BL	BX

Register Relative Addressing

MOV [SI+600], BH



1	0	0	0	1	0	0	0	1	0	1	1	1	1	0	0

88BC 00 06

MOD	Interpretation
00	Memory mode with no displacement follows except for 16-bit Displacement when R/M = 110
01	Memory mode with 8-bit displacement
10	Memory mode with 16-bit displacement
11	Register mode (no displacement)

Operands	Memory Operands			Register Operands	
	No Displacement	Displacement 8-bit	Displacement 16-bit		
MOD / R/M	00	01	10	11	
				W = 0	W = 1
000	(BX) + (SI)	(BX) + (SI) + D8	(BX) + (SI) + D16	AL	AX
001	(BX) + (DI)	(BX) + (DI) + D8	(BX) + (DI) + D16	CL	CX
010	(BP) + (SI)	(BP) + (SI) + D8	(BP) + (SI) + D16	DL	DX
011	(BP) + (DI)	(BP) + (DI) + D8		BL	BX
100	(SI)	(SI) + D8	(SI) + D16	AH	SP
101	(DI)	(DI) + D8	(DI) + D16	CH	BP
110	D16	(BP) + D8	(BP) + D16	DH	SI
111	(BX)	(BX) + D8	(BX) + D16	BH	DI

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

Thankyou