Examples

1. **MOV [BX + 34h], AL - convert this assembly instruction to Machine Code. Also show the hex representation of the machine code.**
   Sol.

1	0	0	0	1	0	0	0	0	1	0	0	0	1	1	1
OPCODE						D	W	MOD		REG			R/M		

0	0	1	1	0	1	0	0								
Low byte displacement								Higher byte displacement							

2. **MOV AX, [BX + 1234h] - convert this assembly instruction to Machine Code. Also show the hex representation of the machine code.**
   Sol.

1	0	0	0	1	0	1	1	1	0	0	0	0	1	1	1
OPCODE						D	W	MOD		REG			R/M		

0	0	1	1	0	1	0	0	0	0	0	1	0	0	1	0
Low byte displacement								Higher byte displacement							

3. **MOV CL, DH - convert this assembly instruction to Machine Code. Also show the hex representation of the machine code.**
   Sol.

1	0	0	0	1	0	0	0	1	1	1	1	0	0	0	1
OPCODE						D	W	MOD		REG			R/M		

Low byte displacement								Higher byte displacement							

4. MOV CX, [BP + 437AH] - convert this assembly instruction to Machine Code. Also show the hex representation of the machine code.
   Sol.

1	0	0	0	1	0	1	1	1	0	0	0	1	1	1	0
OPCODE						D	W	MOD		REG			R/M		

0	1	1	1	1	0	1	0	0	1	0	0	0	0	1	1
Low byte displacement								Higher byte displacement							

5. 89879140h convert this into an assembly instruction.
   Sol.

8 9 8 7 9 1 4 0

1000 1001 1000 0111 1001 0001 0100 0000

1	0	0	0	1	0	0	1	1	0	0	0	0	1	1	1
OPCODE						D	W	MOD		REG			R/M		

1	0	0	1	0	0	0	1	0	1	0	0	0	0	0	0
Low byte displacement								Higher byte displacement							

Assembly instruction:

- As opcode 100010 its a MOV instruction
  MOV __, __;
- As D = 0 so we can say the register is in source
  MOV __, _a register_;
- As W = 1 we will go to w = 1 column in the table and match reg = 000 value which matches the register AX
  MOV __, AX;
- As MOD = 10 so there is 16 bit displacement in the offset now we go to the 10 column of the table and match the R/M = 111
  MOV [BX + d16], AX;
- D16 = 16 bit displacement - higher bit = 40, lower bit 91
  So the instruction: MOV [BX + 4091H], AX

6. **8A34h convert this into an assembly instruction.**

8A34h
1000 1010 0011 0100

1	0	0	0	1	0	1	0	0	0	1	1	0	1	0	0
OPCODE						D	W	MOD		REG			R/M		

Instruction: MOV reg, memory with no displacement;
MOV DH, [SI];

MOD / RM	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	[BP] + [DI] + d16	BL	BX
100	[SI]	[SI] + d8	[SI] + d16	AH	SP
101	[DI]	[DI] + d8	[DI] + d16	CH	BP
110	d16 (direct address)	[BP] + d8	[BP] + d16	DH	SI
111	[BX]	[BX] + d8	[BX] + d16	BH	DI

MODE	OPERAND NATURE	
00	Memory with no displacement	→ *MOV AX, [BX]*
01	Memory with 8-bit displacement	→ *MOV AX, [BX + 12h]*
10	Memory with 16-bit displacement	→ *MOV AX, [BX + 1234h]*
11	Both are registers	→ *MOV AX, BX*