

Chapter-02

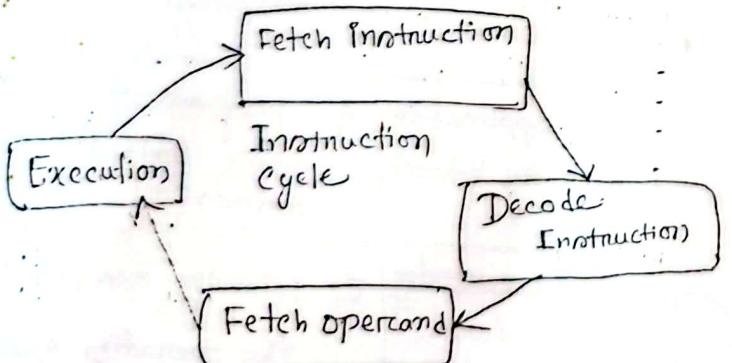
- 1) Write down the steps to execute a machine instruction.

Machine instruction are executed in 4 steps.

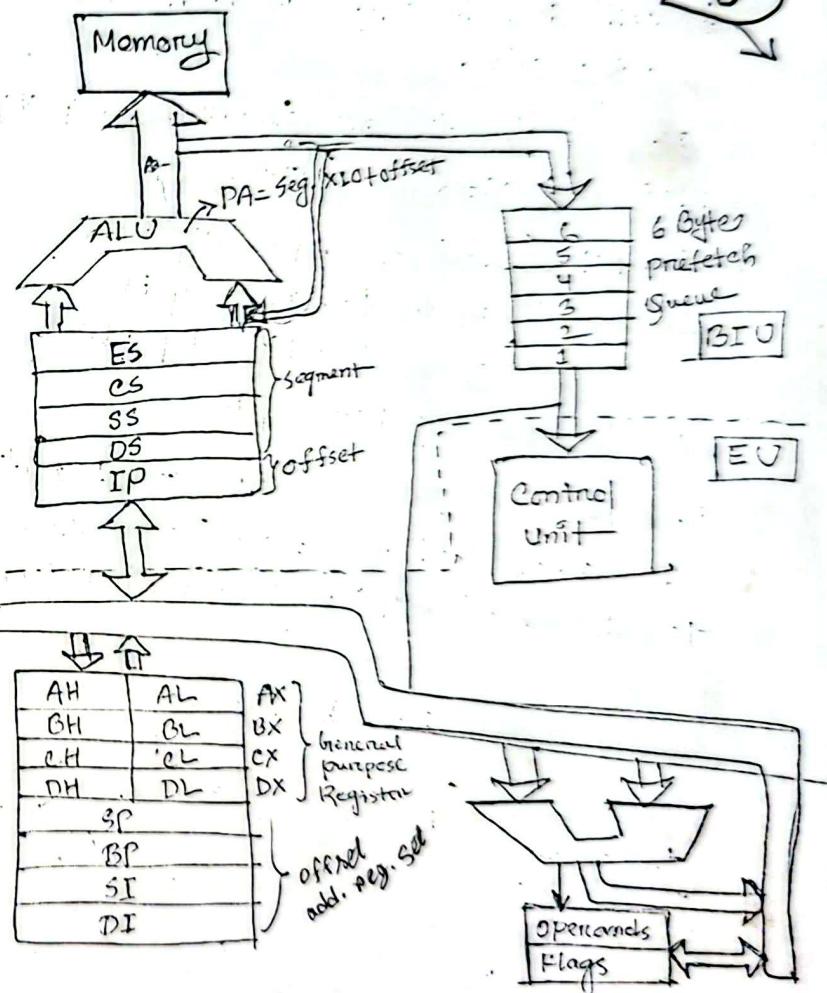
Which are listed as follows.

1. Fetch instruction: Done by BIU
2. Decode instruction: Done by EU
3. Fetch operands: Done by BIU
4. Execution: Done by EU

Both Bus Interface unit (BIU) and Execution Unit (EU) works mutually exclusive to each other to execute the machine instruction.



- 2) Illustrate the Intel 8086 Microprocessor Organization.



Q) Define memory Segment.

Memory segment:

Memory Segment is the process in which the ~~memory~~ main memory of the computer is logically divided into different segments and each segment has its own base address.

Q) Write down the features of 80286 microprocessor.

(i) 80286 is a high performance 16 bit microprocessor.

(ii) 80286 base address is a 24 bit address, so it could address upto 16 MB of physical memory.

(iii) 80286 supports protected mode memory addressing.

(iv) Multitasking ability

(v) 80286 has improved bus interface

Q) Write down the difference between physical and logical memory?

RAM	Physical memory	Logical memory
	1. Physical memory refers to the actual hardware components where data and instructions are stored in a computer system. It consists of RAM modules.	1. Logical memory is the virtual address space that a program or process uses during execution.
	2. Physical hardware components storing data.	2. Virtual representation of available memory.
	3. Direct interaction by CPU.	3. Accessed through logical addresses.
	4. Users and applications don't directly interact.	4. Program and users work with logical address.
	5. Hardware and operating Managed by hardware and operating system	5. primarily managed by the operating system.

Sat / Sun / Mon / Tue / Wed / Thu / Fri

 Date: / /

- ⑥ A memory location has physical address $80FD2h$. In what segment does it have offset $BFD2h$?

$$PA = \text{segment address} \times 10 + \text{offset address}$$

or, $80FD2h = \text{Segment} \times 10h + BFD2h$

$$\begin{aligned} & 80FD2h - BFD2h \\ \text{or, Segment} &= \frac{80FD2h - BFD2h}{10h} \\ &= 7500h \end{aligned}$$

(Ans)

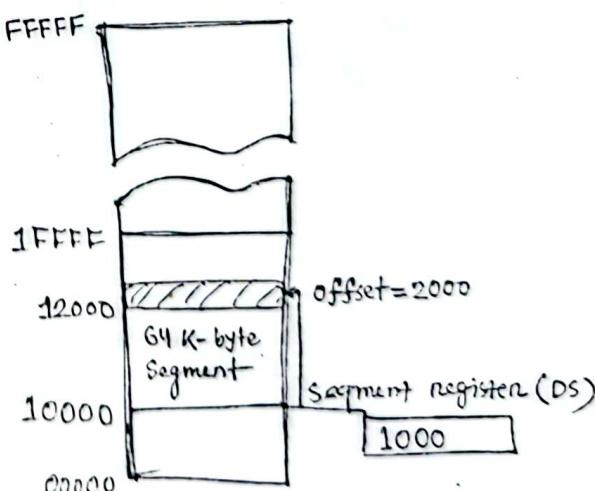
Sat / Sun / Mon / Tue / Wed / Thu / Fri

 Date: / /

- ⑦ Determine the memory location addressed by the following real mode 8086 register combinations: $DS = 1000H$ and $DI = 2000H$ also draw the diagram of memory access.

Given - $DS = 1000H$
 $DI = 2000H$

$$\begin{aligned} \text{Memory location Accessed} &= DS \times 10H + DI \\ &= 1000 \times 10 + 2000 \\ &= 12000H \end{aligned}$$



(5) How is the local descriptor table addressed in the memory system?

The location of the local descriptor table is selected from the global descriptor table. One of the global descriptors is set up to address the local descriptor table. To access the local descriptor table, the LDTR (Local Descriptor Table Register) is loaded with a selector, just as a segment register is loaded with a selector. This selector access the global descriptor table and ~~then~~ loads the address, limit, access rights of the local descriptor table into the cache portion of the LDTR.

(6) Which Intel microprocessor has 1T of memory?

~~Intel Xeon Scalable processor family~~

~~Xeon processor supports up to 6 terabytes of system memory.~~

Pentium 4 and Core 2 ~~has~~ Intel microprocessor addresses 1T of memory.

(10) What is the purpose of the microprocessor in a microprocessor based computer?

Microprocessors contain the arithmetic, logical and control circuit required to perform the function of a CPU. It is responsible for processing the unique set of instructions and processes that make up a computer program.

The microprocessor is a multipurpose clock driven, register based, digital integrated circuit that accepts binary data as input processes it according to instructions stored in its memory and provides results as output.

Sat / Sun / Mon / Tue / Wed / Thu / Fri

Date: _____/_____/_____

Sat / Sun / Mon / Tue / Wed / Thu / Fri

Date: _____/_____/_____

(11) What will be the CS:IP of physical addressBCDEFh where CS = FFF0?

Given,

$$PA = BCDEF\text{h}$$

Segment, CS = FFF0
address

We know,

$$\cdot PA = CS \times 10 + \text{offset(IP)}$$

$$\therefore IP = PA - CS \times 10$$

$$= BCDEF\text{h} - FFF0 \times 10\text{h}$$

$$= BCDEF\text{h} - FFF0\text{h}$$

$$= ACFE\text{Fh}$$

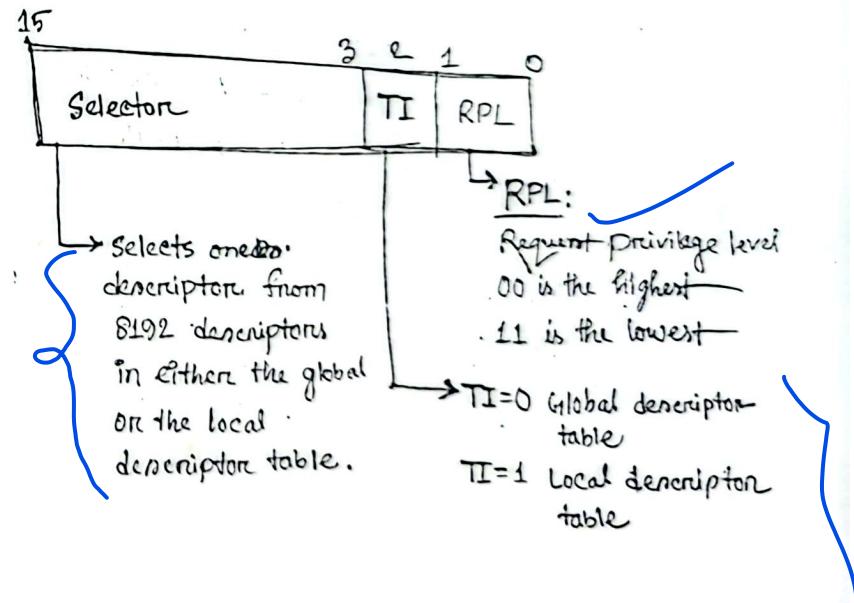
Ans

(12) Which register locates the global descriptor table?

GDTR (Global descriptor table register)

(13) Describe the content of the segment register at protected mode memory addressing.

Segment register contains a selector at protected mode memory addressing. Selector that selects a descriptor from a descriptor table.



14) What are program-visible registers?

(Program visible register are the registers that are directly used in an instruction.)

The programming model of the 8086 through the pentium 4 is considered to be program visible because its register satisfy above condition.

15) In the real mode, show the starting and ending addresses of each segment located by the following segment register values:

i) 1000H

Starting address = 1000×10 H
 $= 10000$ H

ending address = 10000 H + FFFFH
 $= 1FFFFH$

11) 1234H

Starting address = 1234×10 H
 $= 12340$ H

ending address = 12340 H + FFFFH
 $= 2233F$ H

11) 2300H

Starting address = 2300×10 H
 $= 23000$ H

Ending address = 23000 H + FFFFH
 $= 32FFFFH$

16) Protected mode memory addressing allows access to which area of the memory in the 80286 microprocessor?

Protected mode memory addressing allows access to data and programs located above the first 1 M byte of memory as well as within the first 1 M byte of memory.

Q7) Determine the memory location addressed by the following real mode Cane 2 register combinations:

(i) DS = 2000H and EAX = 00003000H

$$\therefore \text{memory location} = 2000H \times 10H + 3000H \\ = 23000H$$

(ii) DS = 1A00H and ECX = 00002000H

$$\therefore \text{memory location} = 1A00H \times 10H + 2000H \\ = 1C000H$$

(iii) DS = C000H and ESI = 0000A000H

$$\therefore \text{memory location} = C000H \times 10H + A000H \\ = CA000H$$

Q8) Which registers are used as an offset address for the string instruction destination in the microprocessor?

The Destination Index (DI) register is used as an offset address for the String instruction destination in the microprocessor.

DI is a 16 bit register.

Q9) Find the memory address of the next instruction executed by the microprocessor when operated in the real mode, for the following CS:IP combinations:

(i) CS = 1000H and IP = 2000H

$$\text{PA} = (\text{CS} \times 10H) + \text{offset} \\ = 10000 + 2000 \\ = 12000H$$

∴ Next instruction memory address: 12000H

(ii) CS = 3456H and IP = ABEDH

$$\text{PA} = (3456 \times 10H) + ABEDH \\ = BF12DH$$

∴ Next instruction memory address: BF12DH

Chapter-03

Sat / Sun / Mon / Tue / Wed / Thu / Fri

Date:

- ① What is the difference between an intersegment and intrasegment jump?

The intersegment jump allows jumps between segments or to anywhere in the memory system.

The intrasegment jump allows a jump allows a jump to any location within the current code segment.

- ② Show which jmp instruction assembles (short, near, or far) if the jmp THERE instruction is stored at memory address 10000H and the address of THERE is:

i) 10020H

$$\underline{10020H} - 10000H = 20H \text{ (8 bit)}$$

so, short jump

8 bit - short jump
16 bit - near jump
intragsegment jump

ii) 11000H

$$11000H - 10000H = 1000H \text{ (16 bit)}$$

near jump

Sat / Sun / Mon / Tue / Wed / Thu / Fri
Date:

- ③ ~~What is wrong with the Mov DS, SS instruction?~~

iii) OFFFE H

$$(10000H - OFFFEH) = 2H \text{ (4 bit)}$$

Short jump

iv) 30000 H

$$(30000H - 10000H) = 20000 \text{ (20 bit)} > 16 \text{ bit}$$

Far jump

- ③ ~~What is wrong with Mov DS, SS instruction?~~

Mov DS, SS

Hence, DS and SS both are segment.

In the instruction segment, segment to segment transfer is not allowed.

④ What do the following MOV instructions accomplish?

i) $MOV AX, BX$

The contents of BX are copied into AX .

ii) $MOV BX, AX$

The contents of AX are copied into BX .

iii) $MOV ESP, EBP$

The contents of EBP are copied into ESP .

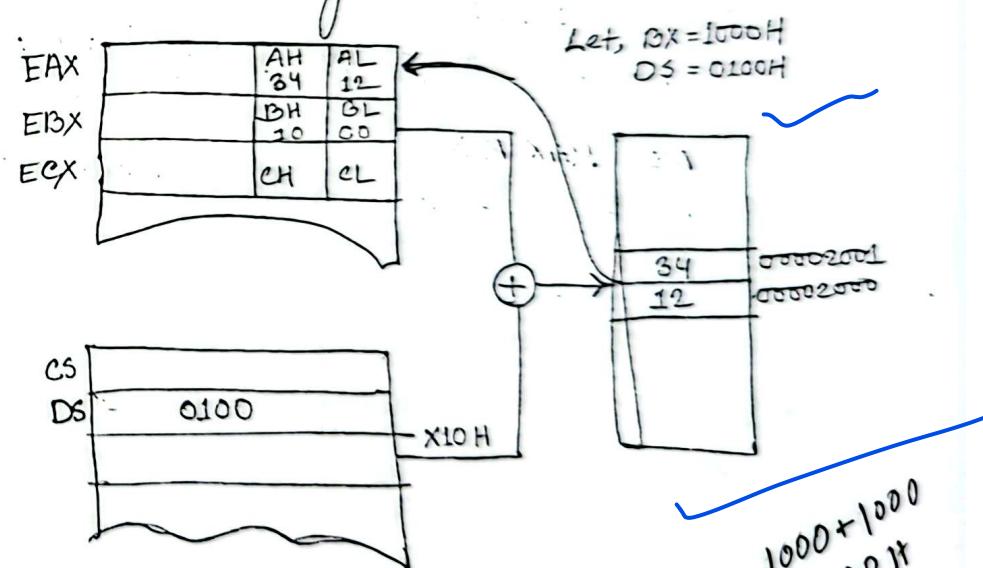
iv) $MOV RAX, RCX$

The contents of RCX are copied into RAX .

⑤ Explain the instruction with respect to 8085 microprocessor. $MOV AX, [BX]$.

The instruction $MOV AX, [BX]$ is used to move the contents of the memory location pointed to by the BX register into AX register.

This instruction is an example of register indirect addressing mode.



6) What is displacement?

In microprocessors displacement typically refers to a value that is added to a base address to calculate the effective address of an operand in memory.

7) How does it determine the memory address in a MOV DS:[2000H], AL instruction?

The instruction MOV DS:[2000H], AL is used to move the contents of the AL register into the memory location specified by the segment register DS and the offset address 2000H.

The DS is one of the segment registers that is used to specify the segment portion of the memory address. The content of the DS register is left shifted by 4 bits. So the memory address is determined by combining the segment and offset as follows—

$$\text{Memory address} = (\text{DS register left shifted by 4}) + \text{offset}$$



8) Explain the difference between the **MOV BX, DATA** instruction and the **MOV BX, OFFSET DATA** instruction.

DATA DW 1234H

MOV BX, DATA

In this code DATA is a label that is used to identify the memory location where the value 1234H is stored.

The instruction **MOV BX, DATA** copies the word from the memory location pointed to by the label DATA into the BX register.

On the other hand, the instruction **MOV BX, OFFSET DATA** copies the offset address of the label DATA into the BX register.

9) How many bytes are stored on the stack by a **PUSH AX**?

When we **PUSH AX**, 2 bytes are stored on the stack.

Sat / Sun / Mon / Tue / Wed / Thu / Fri

Date / /

Sat / Sun / Mon / Tue / Wed / Thu / Fri

Date / /

- (10) LIST the 16 bit registers that are used for register addressing.

AX, BX, CX, DX, SP, BP, SI, DI,
CS, DS, ES, SS, FS and GS

- (11) What is wrong with a $MOV [BX], [DI]$ instruction?

Memory to memory transfers are not allowed with the MOV instructions.

- (12) If a near jump uses a signed 16 bit displacement, how can it jump to any memory location within the current code segment?

16 bit displacement (near) has a range of $\pm 32K$ bytes
so, it can jump to any memory location within the current code segment by $\pm 32 K$ bytes.

- (13) Suppose that DS = 1300H, SS = 1400H,

BP = 1500H and SI = 0100H. Determine

the address accessed by each of the following instructions, assuming real mode operation:

- i) $MOV EAX, [BP+200H]$

$$\text{Address} = SS \times 10H + BP + 200H$$

$$= 1400H \times 10H + 1500H + 200H$$

$$= 15700H$$

- ii) ~~MOV AL, [SI-0100H]~~

- a. $MOV AL, [BP+SI-200H]$

$$\text{address} = SS \times 10H + BP + SI - 200H$$

$$= 1400H \times 10H + 1500H + 100H - 200H$$

$$= 15400H$$

Ans

14) Suppose that DS=1000H, BX=0300H, and SI=0000H. Determine the memory address accessed by each of the following instructions, assuming real mode operation.

(i) MOV AL, [BX]

$$\begin{aligned} & \text{DS} \times 10H + 1234H \\ & = 2000H + 1234H \\ & = 3234H \end{aligned}$$

(ii) MOV EAX, [BX]

$$\begin{aligned} & \text{DS} \times 10H + BX \\ & = 200H \times 10H + 300H \\ & = 2300H \end{aligned}$$

(iii) MOV [DI], AL

$$\begin{aligned} & \text{DS} \times 10H + DI \\ & = 200H \times 10H + 400H \\ & = 2400H \end{aligned}$$

15) What, if anything, is wrong with the MOV AL, [BX][SI] instruction?

Given instruction MOV AL, [BX][SI] where [BX][SI] indicate separate index that is defined two dimensional array.

In this case BX would be treat as the row index and SI as the column index. This kind of syntax is not standard in 8086 CPU.
It should be MOV AL, [BX+SI] to properly access the memory location.

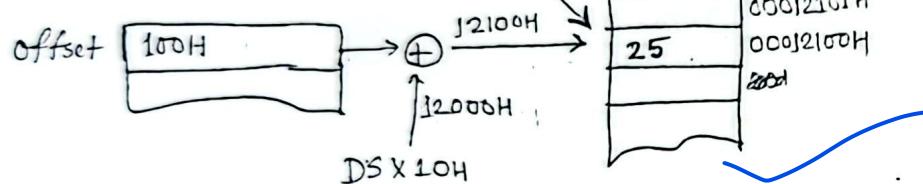
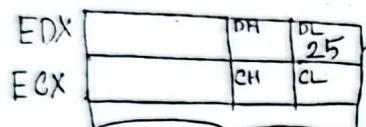
16) Suppose that DS=1200H, BX=0100H, and SI=0250H. Determine the address accessed by each of the following instructions, assuming real mode operation:

(i) MOV [100H], DL

$$\begin{aligned} & \text{If indicate direct addressing mode} \\ & \text{address} = \text{DS} \times 10H + 100H \\ & = 1200H \times 10H + 100H \\ & = 12100H \end{aligned}$$

Sat / Sun / Mon / Tue / Wed / Thu / Fri

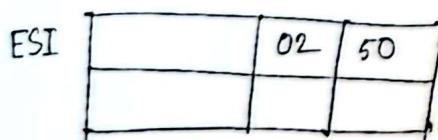
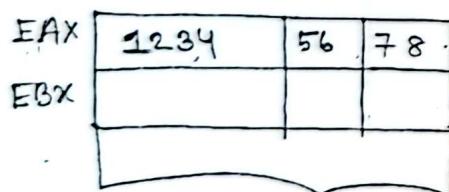
Date / /



ii) MOV [SI+100H], EAX

It indicate Register relative addressing mode.

$$\begin{aligned} \text{address} &= DS \times 10H + SI + 100 \\ &= 1200H \times 10H + 250 + 100 \\ &= 12350H \end{aligned}$$



displacement, $100H$ → \oplus
 $1200H \times 10H$ → \oplus
 12350

Address: $00012353H$
 12
 34
 56
 78
 $00012351H$
 $00012352H$
 $00012350H$

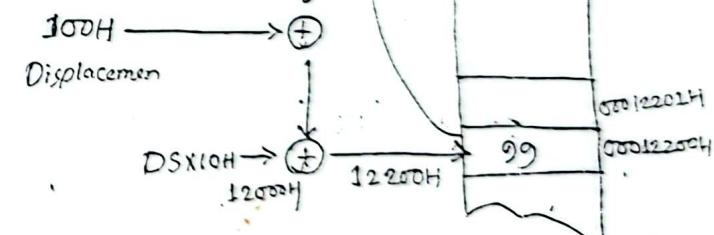
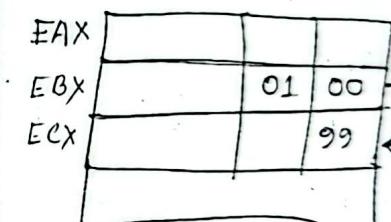
Sat / Sun / Mon / Tue / Wed / Thu / Fri

Date / /

III) MOV DL, [BX+100H]

This instruction indicate register relative addressing mode.

$$\begin{aligned} \text{address} &= 1200H \times 10H + 100H + 100H \\ &= 12200H \end{aligned}$$



Chapter-04

Sat / Sun / Mon / Tue / Wed / Thu / Fri

 Date _____

Sat / Sun / Mon / Tue / Wed / Thu

 Date _____

- ① Convert an 8B9E004CH from machine language to assembly language.

MOD	Function
00	No displacement
01	8 bit sign displacement
10	16 bit sign displacement
11	R/M is a Register

REG:

code	W=0 (Byte)	W=1 (Word)	W=3 (Double word)
0 0 0	AL	AX	EAX
0 0 1	CL	CX	ECX
0 1 0	DL	DX	EDX
0 1 1	BL	BX	EBX
1 0 0	AH	SP	ESP
1 0 1	CH	BP	EBP
1 1 0	DH	SI	EST
1 1 1	BH	DI	EDI

Memory

Code	Addressing mode (16 bit)
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]

Given,

8B9E004CH
displacement

1000 1011 1001 1110
opcode D4 MOD REG R/M

MOD is 10

so, R/M is ... 16 bit sign displacement

W=1 means it's an 16 bit word

D=1 means ... R/M → REG

MOV REG, R/M

REG 011 means BX

R/M ... 110 means SS: [BP]

so,
Assembly language is:

MOV BX, [BP+4000H]

519383800
100000000

Sat / Sun / Mon / Tue / Wed / Thu / Fri
□ □ □ □ □ □

Date: _____

- ② If a MOV SI, [BX+2] instruction appears in a program, what is its machine language equivalent?

MOV SI, [BX+2]
displacement

opcode for MOV is 100010

REG code for SI is 110

Memory code for BX is 111

Though R/M is 8 bit displacement MOD=01

R/M → REG so D=1 and W=1

10001011 01110111

8B7702H

(Ans)

- ③ Convert machine code 8BEC to equivalent assembly instruction.

8BEC

1000 1011 1110 1100
D W MOD REG R/M

→ opcode for mov

W=1 and it's 16 bit

D = 1, so MOV REG, R/M

MOD=11, so R/M is a REG

REG = 101, so register is BP

R/M = 100 is a register so register is SP

So, Assembly instruction is,

MOV BP, SP

④ 8B07H

1000 1011 0000 0111
MOV D W MOD REG R/M

MOD=00, No displacement in memory

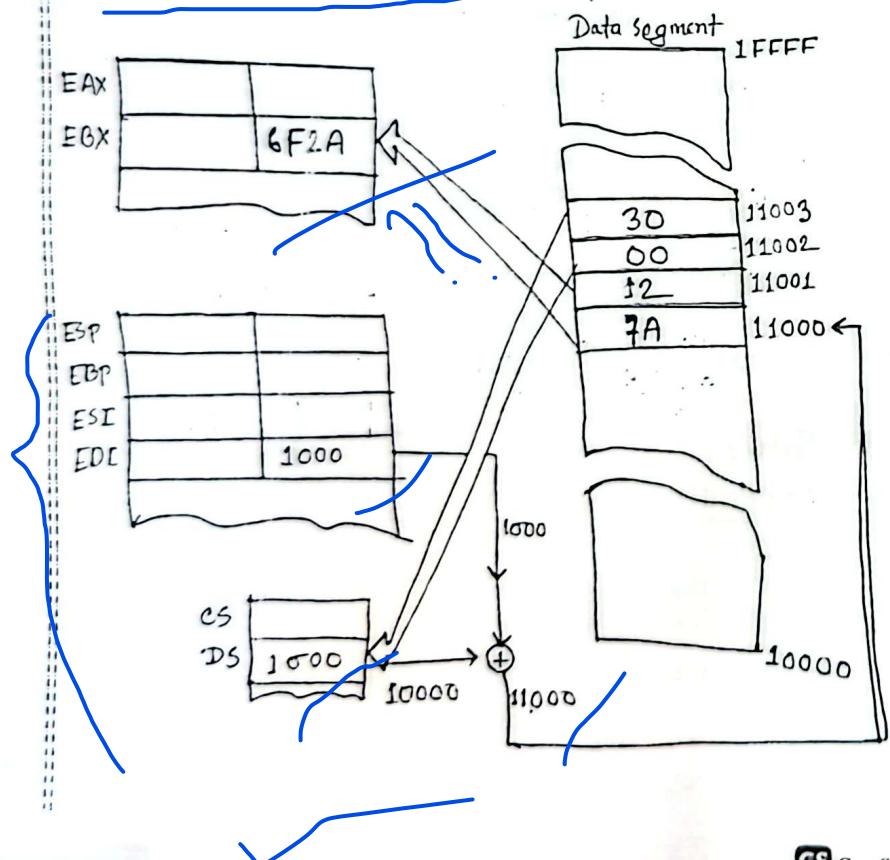
REG=000, so Register is AX

R/M = 111. so memory is DS:[BX]

So, Assembly instruction is

MOV AX, [BX]

(4) Draw a diagram and Show the LDS BX, [DI] instruction loads register BX from address 11000H and 11003H and register DS from locations 11002H and 11003H. This instruction shows at the point just before DS changes to 3000H and BX changes to 127AH. The initial value of DS = 1000 and EDI = 1000.



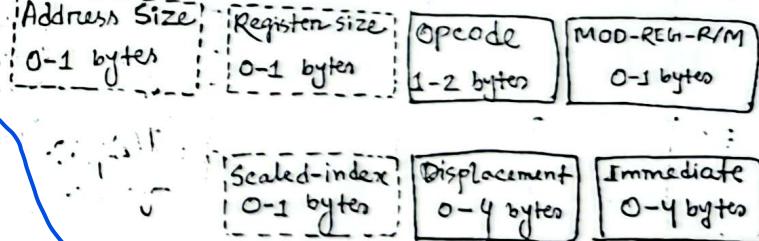
(5) Write down the formats of the 8086 - Core2 instructions.

(a) The 16 bit form.



16 bit instruction mode

(b) The 32 bit form.



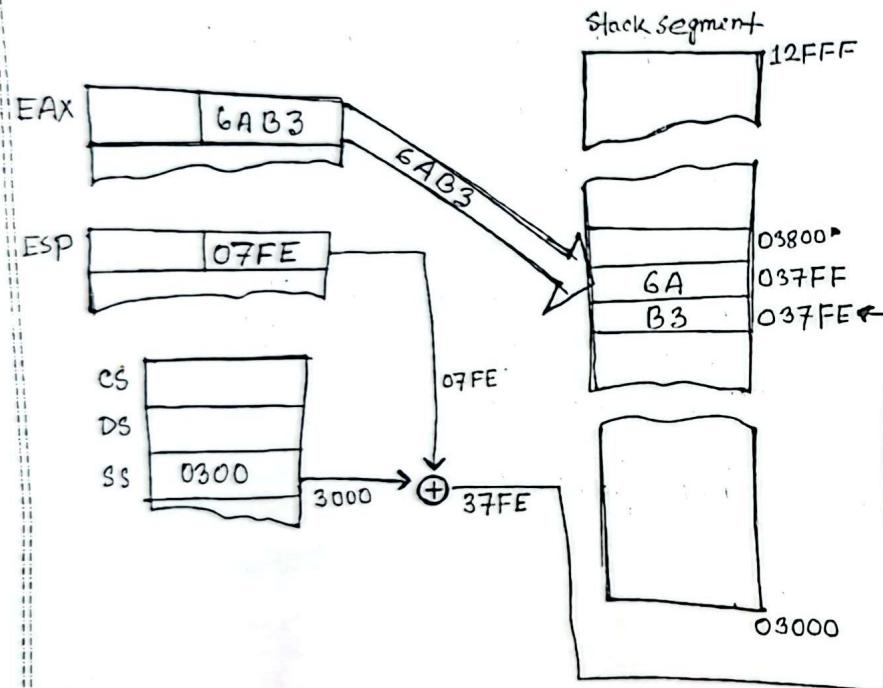
Sat / Sun / Mon / Tue / Wed / Thu / Fri

 Date: _____

Sat Sun Mon Tue Wed Thu Fri

 Date: _____

- ⑥ The effect of the PUSH AX instruction on
 ESP and stack memory locations 37FFF
 and 37FEH. [Assume SS=0300, ESP=07FE]



- ⑦ If the start of a segment is identified
 with .DATA, what type of memory organization
 is in effect?

Model type of organization is in
 effect, if the start of a segment
 is identified with .DATA.

- ⑧ If the segment directive identifies the
 start of a segment, what type of memory
 organization is in effect?

Full segment definitions

⑨ What values appear in sp and ss if the stack is addressed at memory location 02200H?

Stack is addressed using the stack segment (ss) register and the stack pointer (sp) register.

If the stack addressed at memory location 02200H. The value of the stack segment (ss) register would be 0200H.

The value of the stack pointer (sp) register depends on the size of the stack. The stack pointer (sp) register would be 0200H

$$\therefore SS \times 10H + SP$$

$$= 0200H \times 10H + 0200H$$

= 02200H ; Which addressed at memory location 02200H.

⑩ Develop a sequence of instructions that exchanges the contents of AX with BX, ECX with EDX and SI with DI.

XCHG AX, BX

XCHG ECX, EDX

XCHG SI, DI

Sat / Sun / Mon / Tue / Wed / Thu / Fri

 Date: / /

Sat / Sun / Mon / Tue / Wed / Thu / Fri

 Date: / /

- ① Define DAA and DAS.

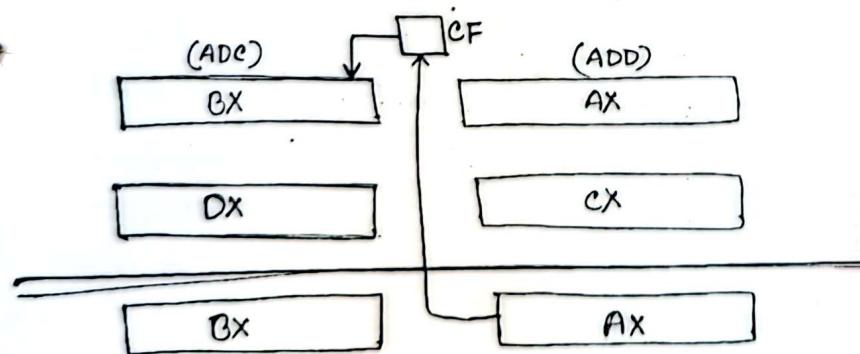
DAA: (Decimal Adjust after addition)

DAA instruction follows the ADD or ADC instruction to adjust the result into a BCD result.

DAS: (Decimal adjust after subtraction)

DAS instruction follows the SUB or SBB instruction to adjust the result into a BCD result.

- ② Show the process of addition with carry, how the carry flag (C) links the two 16 bit additions into one 32 bit addition.



We add 32 bit number in BX and AX to.

the 32 bit number in DX and CX.

Without carry we cannot done this addition.

because 8086-80286 only adds 8 or 16 bit numbers. Contents of AX and CX add to

from LSB and carry appears in the carry flag if sum is greater than FFFFH. Most significant 16 bits of this addition are added with carry flag using the ADC instruction.

This programs adds ~~BX+AX~~ BX-AX to DX-EX , with the sum appearing BX-AX .

- ③ What is wrong with the ADD RCX, AX instruction?

ADD RCX, AX

RCX is 64 bit and AX is 16 bit.

In assembly language mixed sizes not allowed.

- (4) If $AX = 1001H$ and $DX = 20FFH$,
list the sum and the contents of each flag
register bit (C, A, S, Z, and O) after the
ADD AX,DX instruction executes.

$$\begin{array}{r} AX = 1001H = 0001\ 0000\ 0000\ 0001 \\ DX = 20FFH = 0010\ 0000\ 1111\ 1111 \\ \hline & 00110001\ 0000\ 0000 \end{array}$$

$$AX = 3100H$$

Flag register bits:

zero flag (Z) = 0 (result not zero)

Carry flag (C) = 0 (no carry),

Auxiliary carry flag (A) = 1 (~~no~~ half carry)

Sign Flag (S) = 0 (positive)

Parity Flag (P) = 0 (odd parity)

Overflow Flag (O) = 0 (No overflow)

- (5) What is the difference between the
NOT and the NEG instructions?

Logical inversion on the one's complement
is "NOT".

Arithmetic sign inversion on the two's
complement is "NEG".

The NOT instruction inverts all bits of
a byte, word or double word. The NEG
instruction two's complement a number,
which means that the arithmetic sign of
signed numbers changes from positive to negative
or from negative to positive.

Arithmetic sign number

$$\begin{array}{r} (-6) \ 1010 \\ 0101 \\ +1 \\ \hline (+6) \ 0110 \longrightarrow \text{NEG} \end{array}$$

logical inversion NOT

Sat / Sun / Mon / Tue / Wed / Thu / Fri

Date

Sat Sun Mon Tue Wed Thu Fri

Date

- ⑥ Is it possible to add CX to DS with the ADD instruction?

Here DS is the Segment register.

There is no instruction available to add a segment register.

- ⑦ Write an instruction that adds BX to DX and adds the ~~the~~ contents of the carry flag (C) to the result.

ADC (Add with Carry) instruction that adds two operands along with the value of the Carry Flag (C) to the destination operand.

DX → Destination

BX → Source

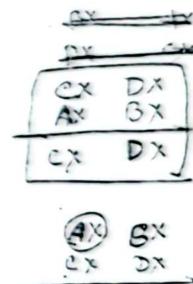
Assembly Instruction:

ADC DX, BX

This instruction add the value of BX and DX together and add the contents of the Carry Flag to the result.

- ⑧ Develop a sequence of instructions that adds the 8 digit BCD numbers in AX and CX to the 8-digit BCD numbers in BX and DX. (AX and CX are the most significant registers. The result must be found in BX and DX after the addition.)

PUSH AX
MOV AL, BL
Add AL, DL
DAA
MOV AL, BH
ADC AL, DH
DAA
MOV BX, AX
POP AX
ADC AL, CL
DAA
XCHG AH, AL
ADC AL, CH
DAA
XCHG AH, AL



③ Which type of Jmp instruction (short, near or far) is assembler for the following:

i) if the distance is 0210H bytes.

Distance equivalent to 12 ~~bit~~ bits.

Length of near jump is 16 bits.

Range in between -32768 to +32767.

so, jt indicates "near jump".

ii) If the distance is 0020 H bytes.

Distance equivalent to 8 bits.

Jt indicates "Short jump". Length of the

Short jump is 8 bits and Range in between

-128 to +127.

iii) If the distance is 10000H bytes.

It indicates far jump, because total size is 20 bit.

Length of the far jump is 32 bits and

Range in between -2147483648 to +2147483647.

③ Which Conditional jump instructions test both the Z and C flag bits?

JA (Jump if above) and JBE (Jump if below, or equal) test both the Z and C flag bits.

When JA tested condition is Z=0 and C=0

And JBE tested condition is Z=1 and C=1

③ Explain how the ~~the~~ LOOPE instruction operates.

The LOOPE (loop while equal) instruction jumps if CX!=0 while an equal condition exists.

It will exit the loop if the condition is not equal or if the CX register decrement to zero.

④ Contrast the operation of a `JMP BX` with a `JMP [DI]`.

A `JMP DI` copies the contents of DI into the instruction address register.

A `JMP [DI]` copies the 16 bit number from the data segment memory location addressed by DI into the instruction address register.

⑤ Explain how the near and far CALL instructions function.

The main difference between

When the near `CALL` executes, it first pushes the offset address of the next instruction on the stack. The offset address of the next instruction appears in the instruction pointer (IP). After saving this return address, it then adds the displacement from bytes 2 and 3 to the IP to transfer control to the procedure.

The 'far' call instruction places the contents of both IP and CS on the stack before jumping to the address indicated by bytes 2-5 of the instruction. This allows the far `CALL` to call a procedure located anywhere in the memory and return from the procedure.

A label followed by a single colon is a short of near address and a double colon denotes a far address.

Q
A
+1