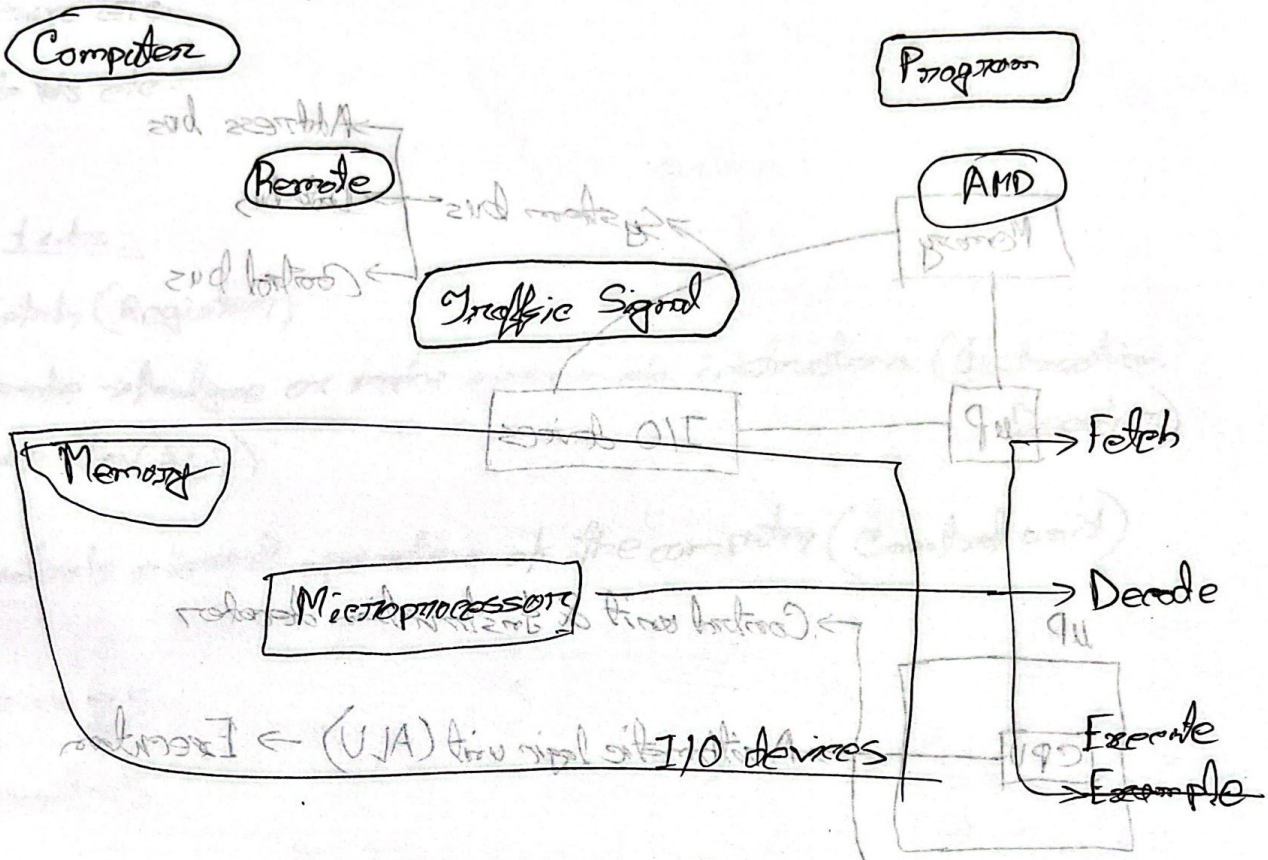
CSE-341

Microprocessor

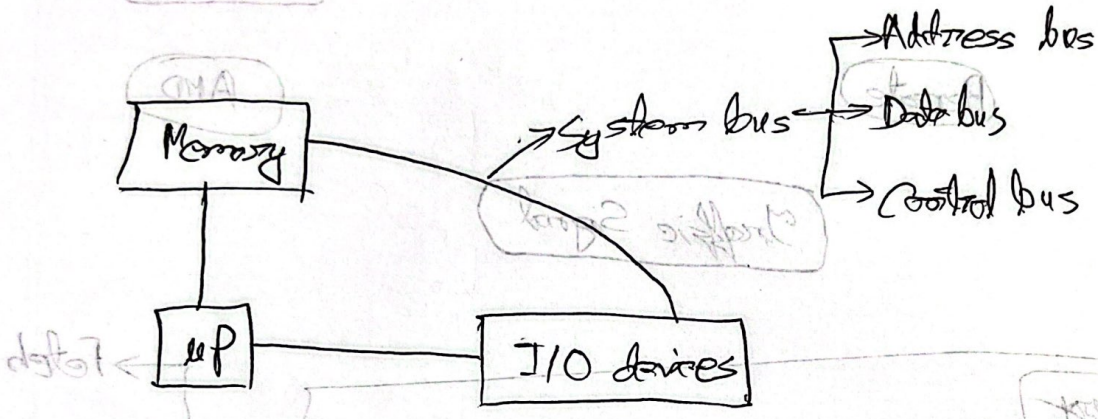


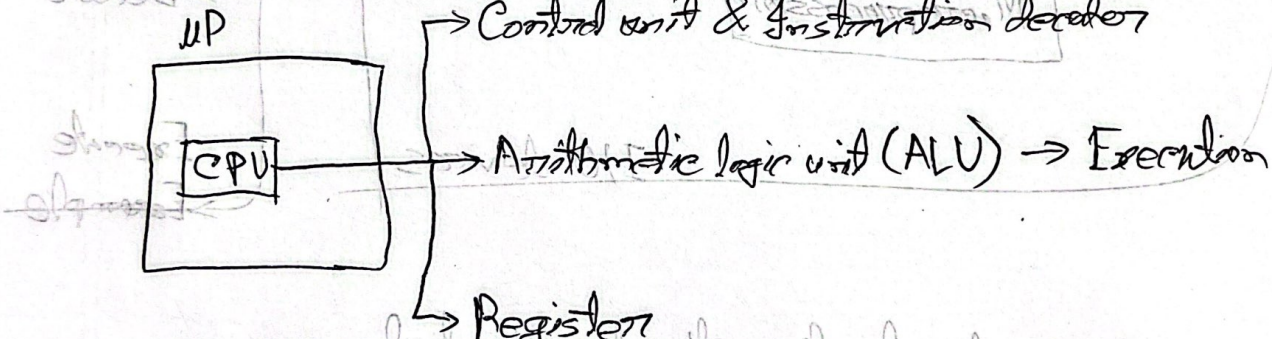
Decoding means understanding the low level language.

⇒ Meaning of binary language.

① Finding out the

We use microprocessor where programming is involved.





Control unit

↳ Synchronization

Important

Block diagram of a Microcontroller.

Write & Read operation (Def)

µP vs. µC

Expensive etc.

Multiple bits etc.

CPU tasks:

Fetch (Register)

Decode → Analyze or make sense of instructions. (Instruction Decoder)

Execute (ALV)

Controls overall operation of the computer (Control unit)

Attendance → 5

Assignment → 5

Quiz → 10

LAB → 25

Mid → 25

Final → 30

CSE-341 Date: 09-10-22

4 parts of CPU register:

Instruction pointer / Program counter:

Fetching steps:

Memory

Represent Hex numbers in 4 bit binary

$2^{23} \Rightarrow 2^{20} \times 2^3$

$\Rightarrow 1MB \times 8$

$\Rightarrow 8MB$

20 bit system → 5 bit Hex

10 bit system → 2^{10} → 1024 bytes

1 kB

$2^{20} = 1MB$ $2^{30} = 1GB$

$= 2^{20} \times 1^{10}$

$= \frac{1 \times 1024 \times 1 \times 1024}{1024 \times 1024}$

$= 1MB$

Address bus size → Memory size

bit → byte

8086 → 16 bit μP

Bus → Address bus → 20 bit = 1MB → CPU memory size
↳ 5 digit hex

Bus → Data → 16 bit → Architecture system.

Bus → Control → 16 bit
↳ 4 digit hex

Address (5 Hex digits)

FFFFFh

20 bit

1MB ↳ 5 hex digit

00000h

1 segment

Segmentation:

$$\frac{1MB}{64kB} = \frac{1 \times 1024 \times 1024}{64 \times 1024}$$

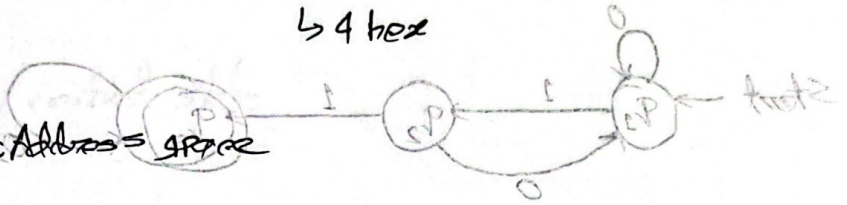
= 16 segments

$2MB = 2^{20} \times 2^1$

16 bit
2 MB

32 bit
2 MB

per segment → 64kB
↳ 16 bit
↳ 4 hex



Memory segment: Address range

per. location → 8bit

CSE-341(LAB)

Date: 01-10-22

Registers:

GPR 16bit	8bit	8bit
AX	AH	AL
BX	BH	BL
CX	CH	CL
DX	DH	DL

Segment registers

CS	64kb
DS	64kb
SS	64kb
ES	64kb

CSE-342

Date: 06-10-22

20 bit → Address Bus → 2^{20} → 1 MB → 5 hex digit

Data Bus / Control / Registers → 2^{16}

Logical Address : A4FB : 4872

Segment, Offset

Physical Address = Logical Segment × 10h + Offset

Types of segmentation.

Overlapping segmentation

Non-overlapping segmentation

October - 15th → Quiz-1

Last - 1, 2, 3 [including 8086 architecture]

If 24 bit → Address bus

Q. Calculate the total memory size that it can access

Q. Determine the first and last address of that memory.

Ans: $2^{24} = 2^{20} \times 2^4$

$= 1\,MB \times 16$

$= 16\,MB$

Ans: 24 bit, hex $= \frac{24}{4} = 6$ hex

Start: 000000h

End: FFFFFFh

Logical Address:

1234:5678h, 05:23A4

Calculate the physical address.

$PA = 1234 \times 10h + 5678$

$= 17988h$

$PA = 0005 \times 10h + 23A4$

$= 023A4h$

Assuming Non-overlapping Segmentation:

Physical Address: 12345h

What is the segment no./which segment does the address belong to? (Segment-1)

What is the first and last address? (10000h → F, 1FFFFh → L)

Calculate 3 logical addresses given 12345h.

1. 1000:2345 → 1000X10+2345=12345h

2. 1001:2335 → 1001X10+2335=12345h

3. 1002:2325 → 1002X10+2325=12345h

023JA

1. 0005:232Ah

2. 0003:234Ah

PA = 23427h

What is the highest segment address and lowest segment address for the given physical address?

** segment address should be divisible by 10:

Logical A:

2342 : 0007h ← Highest LA

1343 : 9997h ← Lowest LA

23427 - FFFF

=13428h → 13430h → Lowest

→ 13420h

389 7AH

Logical Address:

Seg no : Offset

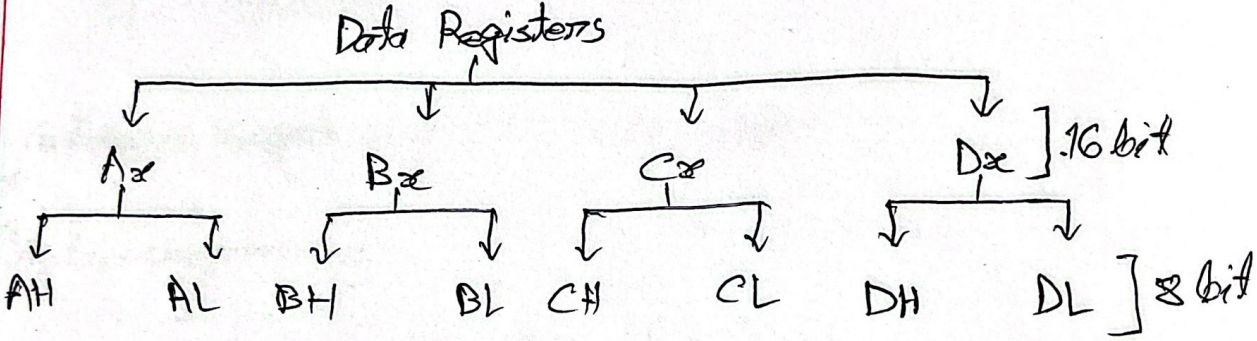
↳ 4 hex digit ↳ 4 hex digit

Seg no. → starting address of that seg

Alg of seg. Faster

Less search

Registers



Segment Registers } 16 bit

Offset registers

CS → Code Segment → IP - Instruction pointer

DS → Data Segment → DI, SI, Bx → Base (stores address)

Destination index

Source index

ES → Extra segment → DI, SI, Bx

SS → Stack segment → SP, BP

stack pointer

Base Pointer

Flag registers

(Later)

Offset Registers

Eg: Seg
DS = 1234h
SI = 2000h
↳ offset

Physical Address = DS X 10 + SI

Practice

Q) A microprocessor which contains 28 bit address bus.

a. What is the total memory size that it can access?

b. " " " first address and last address?

20

∴ Memory size = $2^{20} \times 2^8$ ← hex digit

= (1 X 256) MB

↳ start address: 0000000
End address: FFFFFFF

Q) Assuming non-overlapping:

34728h → Physical Address

What is the segment address / base address?

↳ 30000h
↳ Segment Address

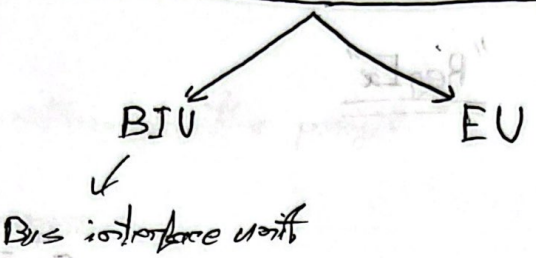
Q) Given,

2743Ah

Determine 3 possible segment addresses and their corresponding logical address.

	Logical Address
Base address : 27430h	2743:000Ah
" " : 27420h	
" " : 27400h	

Internal Architecture of 8086



CSE-331 Date: 11-10-22

Regular Language

Three operations OR Concatenation Kleene closure

$\Sigma = \{0, 1\}$ $\Sigma = \{a, b, c\}$

$0^* = \{\epsilon, 0, 00, \ldots, \infty\}$

$01^* = \{0, 01, 011, \ldots, \infty\}$

$0|1$ $a|b|c$

$0|11$ $a|bc$

$(11)^* = \{\epsilon, 11, 1111, \ldots, \infty\}$

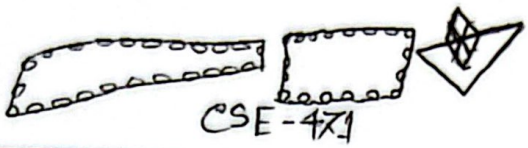
$1(11)^*$

$(0|1)^* = \{\epsilon, 0, 1, 00, 11, \ldots, \infty\}$ $(001)^* = \{\epsilon, 001, 001001, \ldots, \infty\}$

$= \{\epsilon, 0, 1,$

$(0|1)^* = \{\epsilon, 0, 1, 00, 01, 10, 11, \ldots, \infty\}$

Σ^* → dotted kleene closure



CSE-471

Date: 13-10-22

Scrum → Most widely used methodology

Scrum master → Manages the project

Project master/owner

Product backlog ☆☆☆☆

↓

Sprint backlog ☆☆☆

↳ Smaller part of product Backlog.

Burndown chart is used to keep track of daily progress.

CSF-341

16-bit 8086 Architecture

Bus Interface Unit (BIU) → Fetch

EU → Execution Unit

↓

Decode & Execute

Instruction Queue

↳ Prefetch

↳ 6 bytes of instruction prefetch.

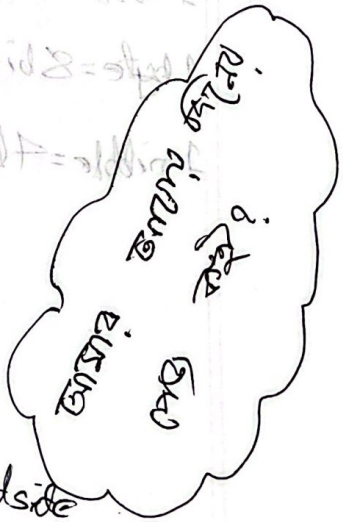
Max length

Min 2 byte

Pipelining → Instruction queue

↳ Fetch, Decode, Execute, Execute

Disadvantage: Data needed to execute the instruction in one of instruction queue is outside of instruction queue will cause error.



8086 Flag Registers

Control Flag

Status Flag → Overflow Flag → OF

Sign Flag → SF

Zero Flag → ZF

Auxiliary Flag → AF

Parity Flag → PF

Carry Flag → CF

MOV AL, 50h 01010000

MOV BL, 32h 00110010

ADD AL, BL 10000010

1 nibble

All bit output ↓

ZF = 0 [Output is not 0]

CF = 0 [No carry out from 8th bit]

SF = 1 [MSB is 1]

AF = 0 [No carry out from 1 nibble]

OF = 1

PF = 1 (Even number of 1's in output)

1 word = 16 bit

1 byte = 8 bit

1 nibble = 4 bit

If there is a carry from 7th bit in to 8th bit = OF = 1

If there is a carry out from 8th bit to 9th bit = OF = 1

OR operation

```
                1 1 1 1 1 1 1
MOV AL, FFh     1 1 1 1 1 1 1 1      ZF = 1
MOV AL, 01h     0 0 0 0 0 0 0 1      CF = 1
                ---------------      SF = 0
ADD AL, BL      1 0 0 0 0 0 0 0 0    AF = 1
                                     OF = 0
                                     PF = 1
```

```
                  1   1   1 1
2346h   0 0 1 0 0 0 1 1 0 1 0 0  0 1 1 0
1256h   0 0 0 1 0 0 1 0 0 1 0 1  0 1 1 0
        -----------------------------------
        0 0 1 1 0 1 0 1 1 0 0 1  1 1 0 0
```

ZF = 0

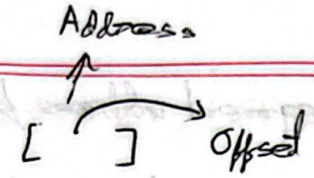
CF = 0

SF = 0

AF = 0

OF = 1

PF = 1 ; [Low byte - 8]



Suppose,

DS = 2246h IP = 32h

SS = 12FFh SP = 2300h

CS = 2243h BP = 14264h

SJ = 2200h

DI = 1246h

BX = 20h

MOV AX, [BX+SJ]

Q. Calculate the physical address from which data will be accessed.

P.A = (DS X 10) + (BX + SJ)

= 2246 X 10 + 2220

= ~~4476~~ 24680h

Ans.

Q. Calculate the 2nd highest and 3rd lowest segment address from the physical address calculated. Determine the respective logical address.

24680h ⟶ 24670h → 2467:0010

2nd highest = 24680 − FFFF (least)

= 14681 → 14690 → 14680

= ~~14680~~ → 14670

1st 2nd

Logical = Segment : Offset

Note from pic ↓

3rd 14670

2nd 14680

MOV CX, [SP+22h]

Q. Calculate the physical address.

P.A = SSX10 + (SP+22h)

= 12FFhX10 + (2300h+22h)

= 15312h

MOV [BX+DI+2000], AX → 4000h

Q. Calculate the physical address where data will be stored.

P.A = DSX10 + (BX+DI+2000)

= (2296X10) + (20+1246+2000)

= 256C6

2nd highest = 256C0 → 256B0 → 256A0 → 256A:0026

256C6 - FFFF = 156C7

156D0 → 156F0 → 156F0 → 15700

15710

1571:FFB6h

MOV AX, 2356h

MOV 2356, AX

ADD [4356], [2222]

Date: 16-10-22

20-10-22

CSF-341

CS ↑ IP ↑

JMP 2000:2346 → ~~direct addressing~~

P.A = CS x 10 + IP

Addressing Program codes ~~from~~ (instruction)!

CS = 2000h

DS = 4000h

Bx, AX = 2236h = IP

JMP AX

P.A = CS x 10 + AX

JMP [BX]

P.A = (DS x 10) + BX

= 40000 + 2236

= 42236h

42237 → 42h

42236 → 30h

IP = 4230h

P.A = CS x 10 + IP

Call [BX]

CS = 2322h

DS = 12464h

BX = 2200h

P.A = DS x 10 + BX

= 12464 x 10 + 2200

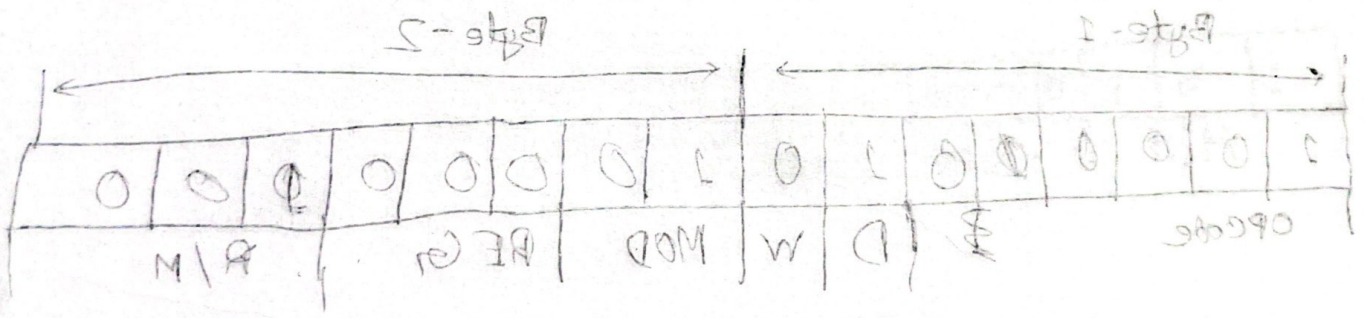
= 14660

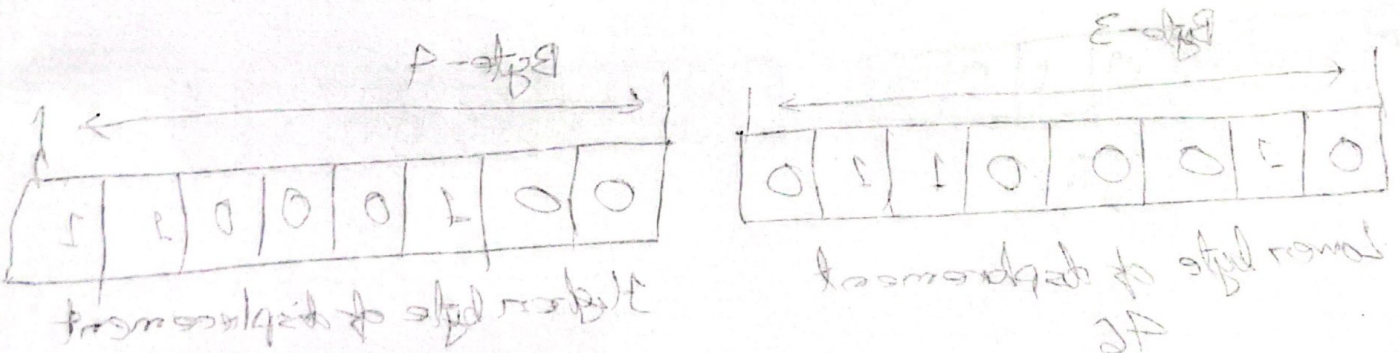
14661h	35h
14660h	20h

IP = 3520h

P.A = CS x 10 + IP

=





Scanned with CamScanner

Call [BX]

CS = 2322h

DS = 12464h

BX = 2200h

P.A = DS x 10 + BX

= 12464 x 10 + 2200

= 14660

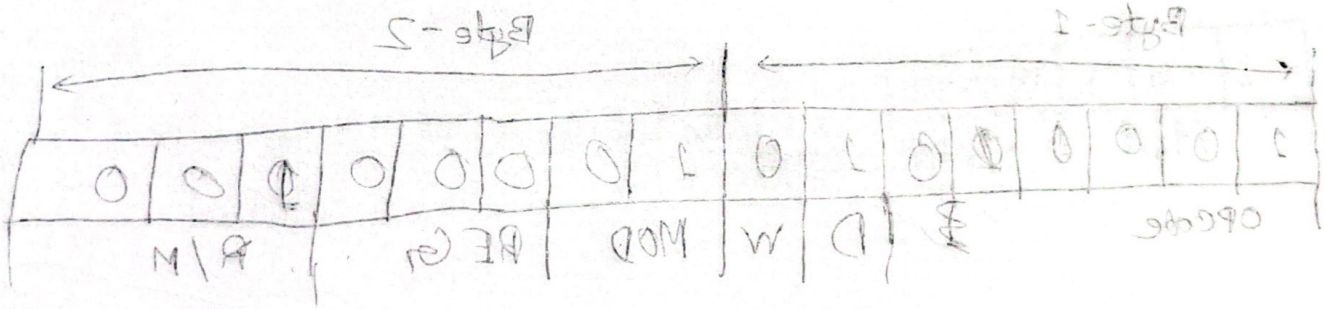
14661h | 35h

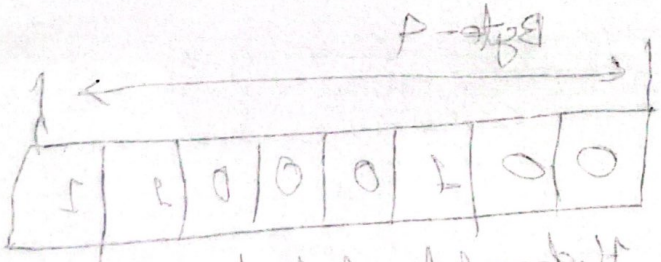
14660 | 20h

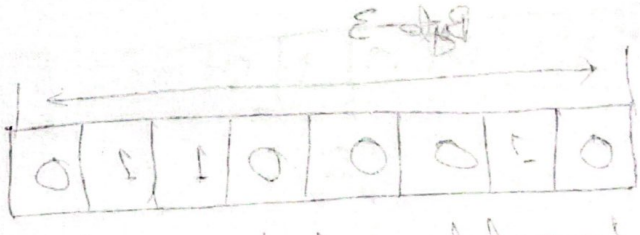
IP = 3520h

P.A = CS x 10 + IP

=







Instruction queue minimum 2 bytes, maximum 6 bytes

Opcode

MOV AX, 24

100010 → Opcode

Operand

Opcode = 6 bits

W = 1 ; for 16 bit

W = 0 ; for 8 bit

D = 1 ; Destination

D = 0 ; Source

data register

Reg = 3 bit data register value

000 → AL

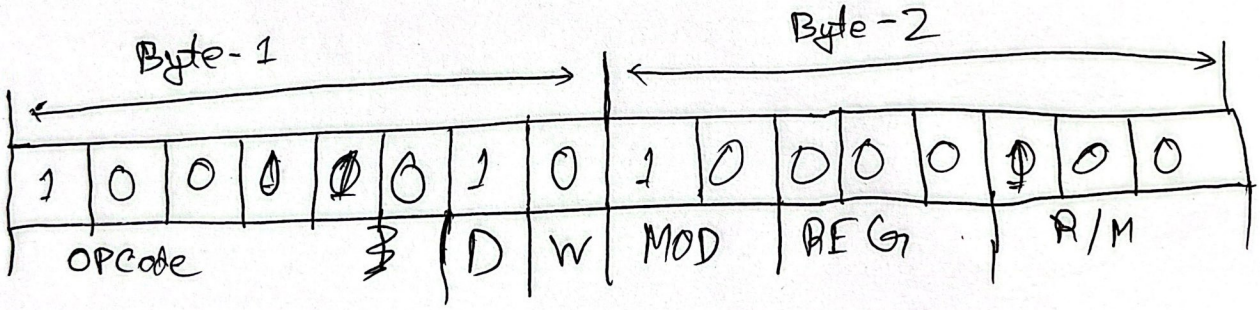
MOV AL, [SI + 2346h]

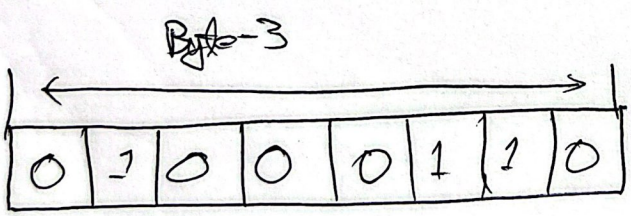
higher lower

RM = depends on MOD

MOD depends on displacement of offset

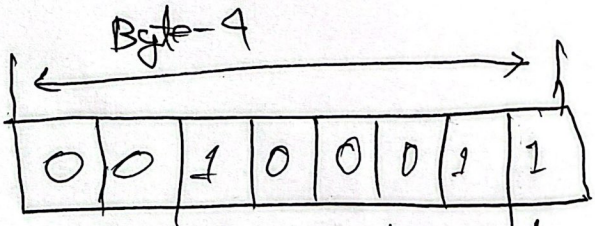
AL, [SI + 1234h]





Lower byte of displacement

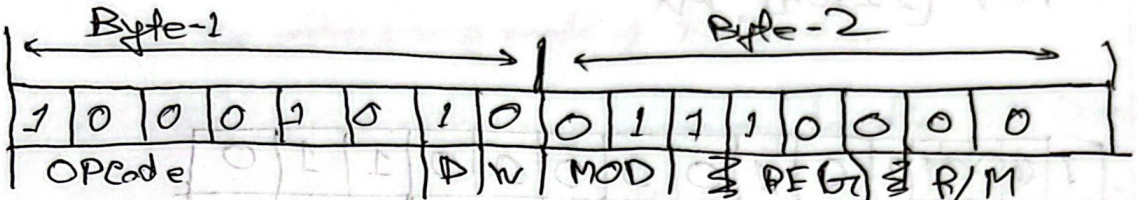
46



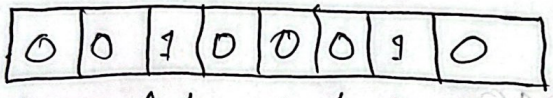
Higher byte of displacement

23

MOV DH, [BX+SI+22h]



Byte-3



Lower byte of displacement

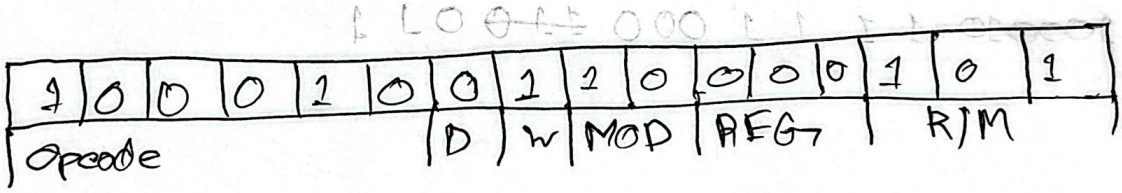
22h

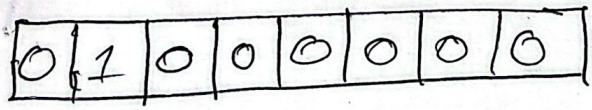
8A7022h

↳ Hexadecimal representation of the solved machine language

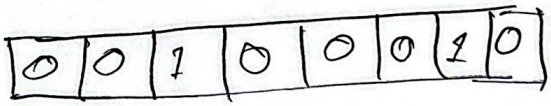
MOV [DI + 2240], AX

↳ source





40

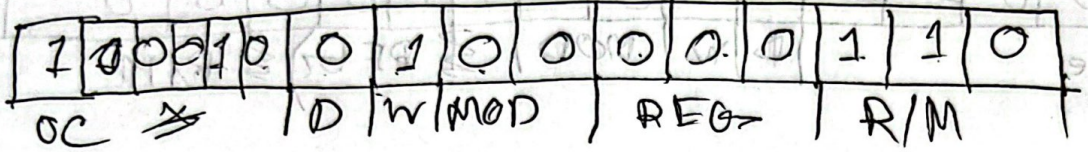


22

89854022

↳ Hex representation.

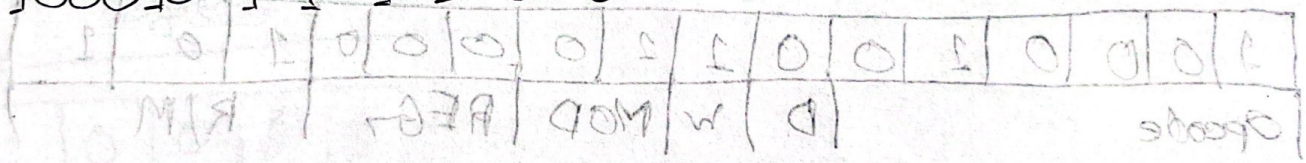
MOV [2240h], AX

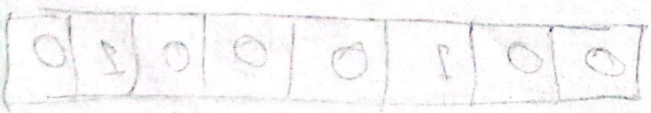


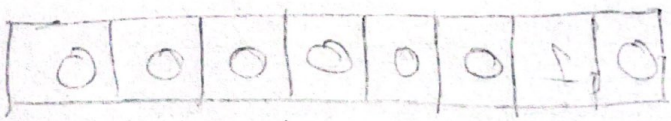
01000000 00100010

MOV (AX), BX

100010 1 1 1 1 000 110 1 1







MOV CX, [BX+SI+2246h]

a. State the addressing mode of the function.

b. Find the physical address of memory location if DS=2240h, BX = 12FFh, SI = 2346h → (DSX10) + (BX+SI+2246h)

c. Find the 3rd highest and 3rd lowest.

d. Find the machine code of the instruction.

e. Hexadecimal representation.

(8B 84 43 8B)h

100010 00 10 1 1 01 100

opcode D W MOD REG R/M

01000011 10001011

MOV [SI+8B43], DH

Date: 27-10-22

CSE-341

*** MOV AX, [BP+2740] (SP) } Exceptional

DSX10 + BP + 2240

→x←

DIP → Parallel and both have can take input

min mode - solo

CLK → Synchronization (duty cycle indicates complete high period)

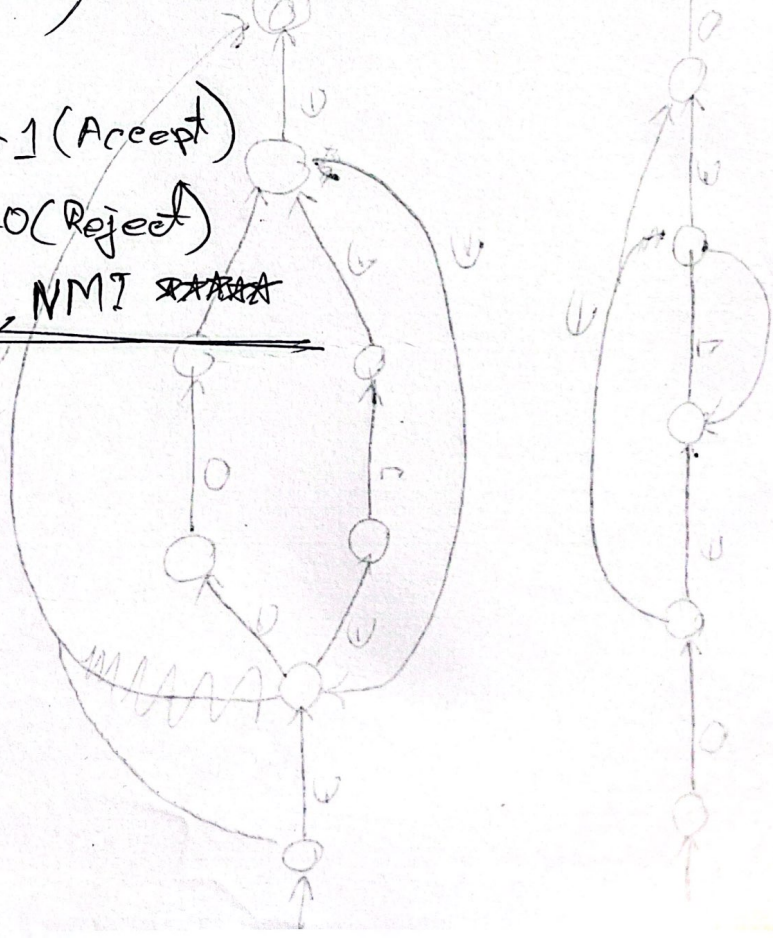
" active "

(Address Bus + Data bus) → ~~MM~~ Multiplexed

S_3 → Interrupt Flag = 1 (Accept)

" , = 0 (Reject)

INTR, NMI ~~XXXXX~~



1 2 3 4 5 6 | 7 | 8 | 9 | 10 | 11 12 13 | 14 15 16 | =

Opcode | D | w | MOD | REG | R/M

MOV AX, [BX+SI+2246h]

1 0 0 0 1 0 1 1 | 1 0 | 0 0 0 | 0 0 0

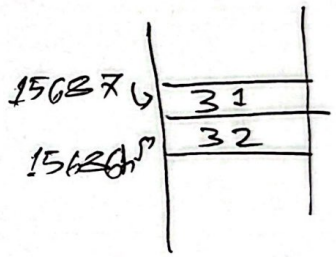
0 1 0 0 | 0 1 1 0 | 0 0 1 0 0 0 1 0

JMP [BX]

DS = 1234h

BX = 3346h

CS = 2200h



PA = DSX10 + 3346

= 15686h

I_P = 3132h

PA of that instruction = CSX10 + IP

= _____