Name: Syeda Nowshin Ibnat

ID: 17183103020

Intake: 39 (1)

Course title: Micropnocessor and Microcontroller

Course code: CSE 315

Semester: Summer 2020

Name: Syeda Nowshin Ibnat

ID: 17183103020

Answer to the question 1(a)

Part-1

Soln: Segmentation is the process in which the main memory of the computer is logically divided into different segments and each segment has its own base address. It is basically use to enhance the speed of execution of the computer system, so that the processor is able to fetch and execute the data from the memory easily and fast. one of the reasons to divide the physical memory of 8086 into segment is that different type of segment can do different works. More reasons -

1) It provides a powerfull memory management mechanism.

2) Data related or stack related operations can be performed in different segments.

3) It allows to extend the address ability of the processor, i.e. segmentation allows the use of 16 bit registers to give an addressing capability of 1 Megabytes. Without segmentation, it would require 20 bit registors.

4) It allows to processes to easily share data.

ID: 171831030 20

Part - 2

Given, DS = 0200H , BX = 0300H , DI = 0400H

(i) MOV AL, [1234H] → DS X 10H + 1234H

= 0200H X 10H + 1234H

= 02000 H + 1234H

= 03234H

(ii) MOV AX, [BX] → 0200H X 10H + 0300H (DS X 10H + 0300H)

= 02000H + 0300H

= 02300H

(iii) MOV [DI], AL → 0200H X 10H + 0400 H (DS X 10H + 0400H)

= 02000H + 0400H

= 02400H

Ans:

Answer to the question 1(b)

Part - 1

Given that, the address bus size is 22 bits.

With 32 bits, we can address 2^{32} bytes, thus we have

2^{2230} = 4 GB of address space. But here we only need

22 bits for the address.

The size of the memory is = 2^{22} = ~~4194 304~~ 4 MB

Ans:

ID: 17183103020

Part-2 :

Given, Physical address 4A37Bh.

(i) Segment number = 40FFh

Soln: Formula: Physical address = Segment X 10H + Offset

$\Rightarrow$ 4A37Bh = 40FFh x 10h + offset

$\Rightarrow$ 4A37Bh = 40FF0h + offset

$\Rightarrow$ 4A37Bh - 40FF0h = offset

$\therefore$ offset = 938Bh

(ii) Offset address = 123Bh

Soln: Formula: Physical address = segment X 10h + offset

$\Rightarrow$ 4A37Bh = segment X 10h + 123Bh

$\Rightarrow$ 4A37Bh - 123Bh = segment x 10h

$\Rightarrow$ 49140h = segment x 10h

$\Rightarrow$ 49140h / 10 = segment

$\therefore$ segment = 4914h

Ans:

Answer to the question 2(a)

Soln: From the question I find that,

Pin P1.0 works as ⟶ Motion Sensor

Pin P2 → light connected

Corridor light off → "0"
↳ when no ones there

Any movement P1.0 = high →
↳ light connected to P2 on → "1"

Now,
corresponding 8051 assembly code :-

```
        SETB P1.0   ; make P1.0 an input
AGAIN : MOV C, P1.0 ; read MS status in CF.    (here, MS = Motion Sensor)
        SETB P2 ;
        MOV P2, C   ; send MS status to lights.

        SJMP AGAIN  ; Keep repeating
```

~~Here, P2 is using as output pin. In 8051 micc micro~~

ID: 17183103020

In ~~80~~ microcontroller we can use switch of a light in one pin and light in another pin. And we can controll it by using switch. Here, ~~we have~~ I have to use a motion sensor instead of switch. And if i use the motion sensor here, and connect it with P1.0 pin. So, when Pin P1.0 is high (1) (motion sensor sense any movement) then P2 will be turned on (lights on). And when P1.0 pin is low (0) (motion sensor ~~sense~~ does not sense any movement) then P2 will be turned OFF (lights off). So, to do this I have to know the status of the motion sensor. That's why I use a carry flag (CF) here. And then sending the status to P2 pin. So, the given code can be use as it sense the motion sensor status first then send it to the lights.

Ans

ID: 17183103020

Answer to the question 2(b)

Soln: Given,

AX = A35FH

BX = F535H

And Instructions, MOV cx, AX

ADD AX, cx

SUB AX, BX

Now,

1st Instruction

For → MOV cx,AX

No flag bits are affected. But value of AX copies to cx.

So, cx = A35FH

2nd Instruction

For → ADD AX, cx

For this instruction all the flag bits ~~of~~ will be affected.

Here,
AX = A35FH
CX = A35FH

MSB

Ax → A35FH = 1 0 1 0 0 0 1 1 0 1 0 1 1 1 1 1

(+) cx → A35FH = 1 0 1 0 0 0 1 1 0 1 0 1 1 1 1 1

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

carry ↲ ↳ MSB of result

Now,

Status of the flag bits:

Here, Carry Flag (CF) = 1 (here, carry = 1)

Sign Flag (SF) = 0 (positive number)

Zero Flag (ZF) = 0 (All the bits are not 0)

Parity Flag (PF) = 0 (number of 1 is odd)

Auxiliary Flag (AF) = 1 (when, causes a carry or borrow bit 3 to bit 4)

Overflow Flag (O) = 1 (there is an overflow)

Now, AX = 46BE (Hence, AX = AX + CX)

3rd Instruction

For → SUB AX, BX

All Flag bits will be affected because of this instruction.

So,

		MSB	
	AX → 46BE =	$\boxed{0}$	1000110 10111110
(−)	BX → F535H =	$\boxed{1}$	1110101 00110101
			1010111 00111011 1

Now,

Status of the flag bits:

Here,
- CF = 0 (no carry)
- SF = 1 (negative number)
- ZF = 0 (All the bits are not 0)
- PF = 1 (number of 1 is even)
- AF = 1 (when causes a carry or borrow → bit 3 to 4)
- O = 0 (no overflow here)

Ans:

Answer to the question 3(a)

(i)

Soln: Given, MOV [SI+2000H], BP

Opcode = MOV = 100010

D = Transfer to R/M (memory) = 0

W = 16 bit / word = 1

so, byte 1 of machine code will be = 10001001

MOD = 16 ~~bid~~ bit displacement = 10

REG = Destination = BP = 101

R/M = source = DS:[SI] = 100

so, byte 2 of machine code will be = 10101100

Here, Displacement = 2000 H

so, byte 3 of machine code will be = 0000 0111 1101 1010

∴ The machine code. ~~will be~~ is =

10001001 10101100 0000 0111 1101 1010 B

Ans:

ID: 17183103020

(ii)

Soln:
Given, 8B872DXXH:

Last 2 digit of my ID = 20

so, it is = 8B872D20H

Let's take it as a 16-bit instruction mode. It is not prefixed by 66H or ~~66~~ 67H. so, first byte 8B is opcode.

1st byte

OPCODE						D	W
1	0	0	0	1	0	1	1

- Opcode = 100010 = MOV
- D = 1 = R/M → REG
- W = 1 = WORD

2nd byte

Second byte is = 87

MOD		REG			R/M		
1	0	0	0	0	1	1	1

ID: 17183103020

- MOD = 10 ⇒ 16 bit displacement
- REG = 000 = AL = Destination
- R/M = 111 = DS:[BX] = source

3rd and 4th byte

Third and fourth byte is = 2D20

If we convert 2D20 to Binary →

2D20 = 0010 1101 0010 0000 B

∴ Displacement = 11552H

Let's putt all together. The instruction (8B87 2D20H) moves a byte from memory address to register

i.e: MOV AL, [BX + 11552H]

Ans:

ID: 17183103020

ID: ~~171830~~.

Answer to the question 3(b)

Soln: Given, Crystal frequency = 15.0592 MHz

Now $15.0592 \text{ MHz} = \frac{15.0592 \times 10^3}{12}$

$= 1254.93333 \text{ KHz}$

$= 1254.93 \text{ KHz}$

Machine cycle is $= \frac{1}{1254.933 \text{ KHz}}$

$= 0.7968 \ \mu s$

For HERE LOOP,

$= (2 \times 250) \times 0.91632 \ \mu s$

$= 458.16 \ \mu s$

For AGAIN LOOP, (Here again loop repeats 220 times)

So, $220 \times 458.16 \ \mu s$

$= 100795.2 \ \mu s$

MOV R3, #250 and DJNZ R2, AGAIN .. At the start and end of the again loop" add

$= (6 \times 220 \times 0.91632) \ \mu s$

$= 1209.54 \ \mu s$

As a result $= (100795.2 + 1209.54)$

$= 102004.74 \ \mu s$

Delay increased by 15%

$$\text{so,} \quad 0.7968 \times \frac{15}{100}$$

$$= 0.11952$$

After increasing 15%, size of the delay is now,

$$0.7968 + 0.11952$$

$$= 0.91632$$

		Machine cycle
~~Dela~~ DELAY :	MOV R2, # 220	1
AGAIN :	MOV R3, # 250	1
	MOV A, # 0A5H	1
HERE :	DJNZ R3, HERE	2
	MOV P1, A	1
	NOP	1
	DJNZ R2, AGAIN	2
	RET	2

Am: