National University
Of Computer and Emerging Sciences

Assignment 3:

Name – **Muhammad Taha**

Roll NO – **23p-0559**

SECTION – **BCS(3A)**

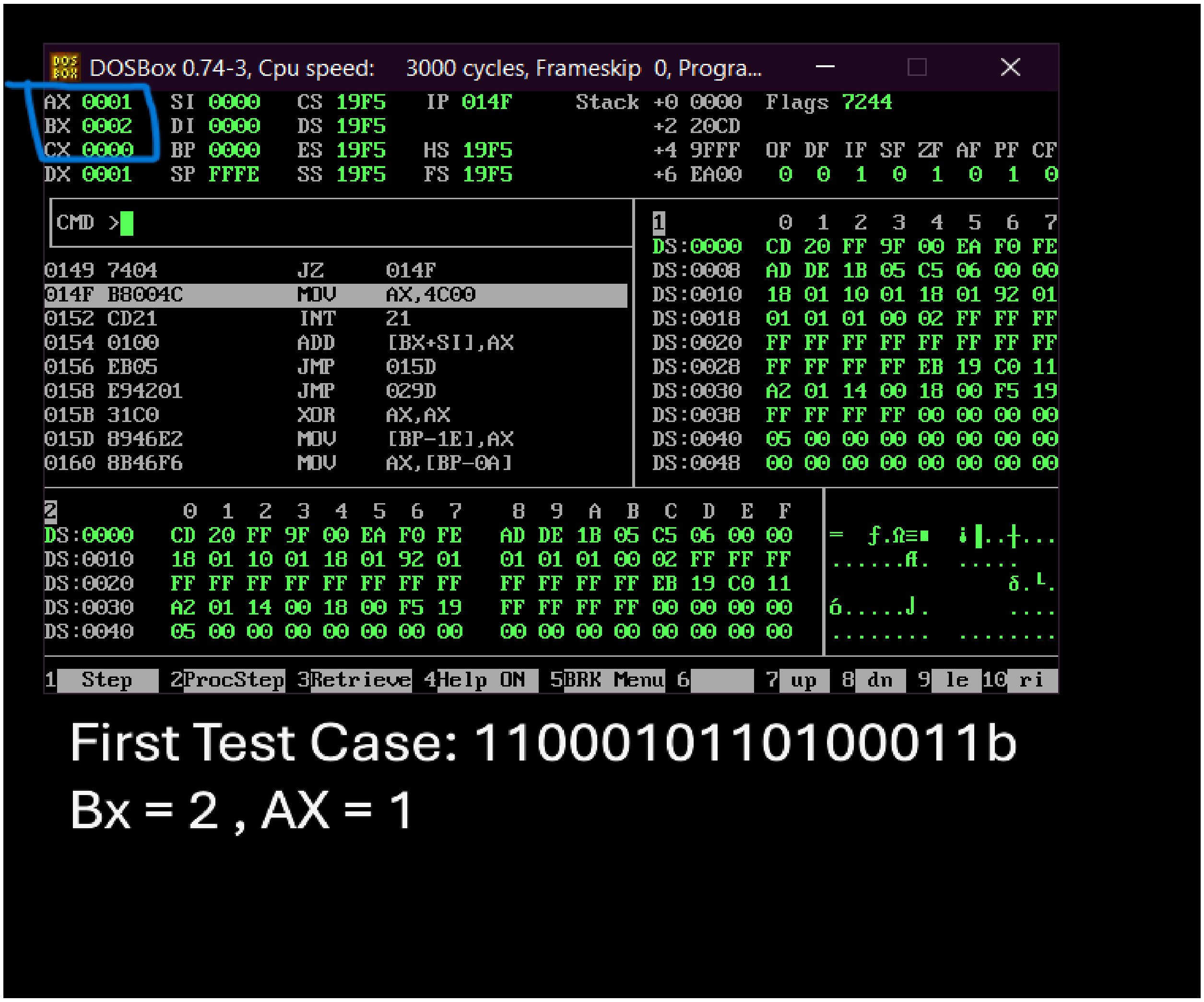
Subject - **Computer Organization and Assembly Language**

Question No. 1:

```
[org 0x0100]

mov ax, 1100010110100011b   ; Test case 1

;mov ax, 1111111111100000b    ; Test case 2

mov bx, 0                   ; iteration counter

mov cx, 0                   ; counter for ones

start:

   mov dx, 0

   shr ax, 1               ; Shift AX right by 1

   jnc start               ; If no carry, jump back to start


   add cx, 1               ; Increment CX for a counted one

   cmp ax, 0                ; Check if AX is 0

   jnz start               ; If not, repeat


   mov ax, cx               ; Move CX (count of ones) to AX

   add bx, 1               ; Increment BX (iteration counter)


   cmp ax, 1                ; Check if AX equals 1

   jz end                  ; If yes, jump to end


   mov cx, 16               ; Reset CX to 16 for 16-bit binary length
```

```
l1:
    sub cx, 1                ; Decrement CX
    rol ax, 1               ; Rotate AX left by 1
    jnc l2                  ; If no carry, jump to l2
    add dx, 1                ; Increment DX for a counted one

l2:
    cmp cx, 0                ; Check if CX is 0
    jnz l1                  ; If not, repeat l1

    mov cx, 0                ; Reset CX to 0
    cmp dx, 2                ; Check if DX >= 2
    jge start               ; If yes, jump back to start

    add bx, 1                ; Increment BX for another iteration

l3:
    cmp ax, 1                ; Check if AX equals 1
    je end                  ; If yes, jump to end
    shr ax, 1               ; Shift AX right by 1
    jmp l3                  ; Repeat l3
```

```
end:

    mov ax, 0x4c00

    int 0x21
```

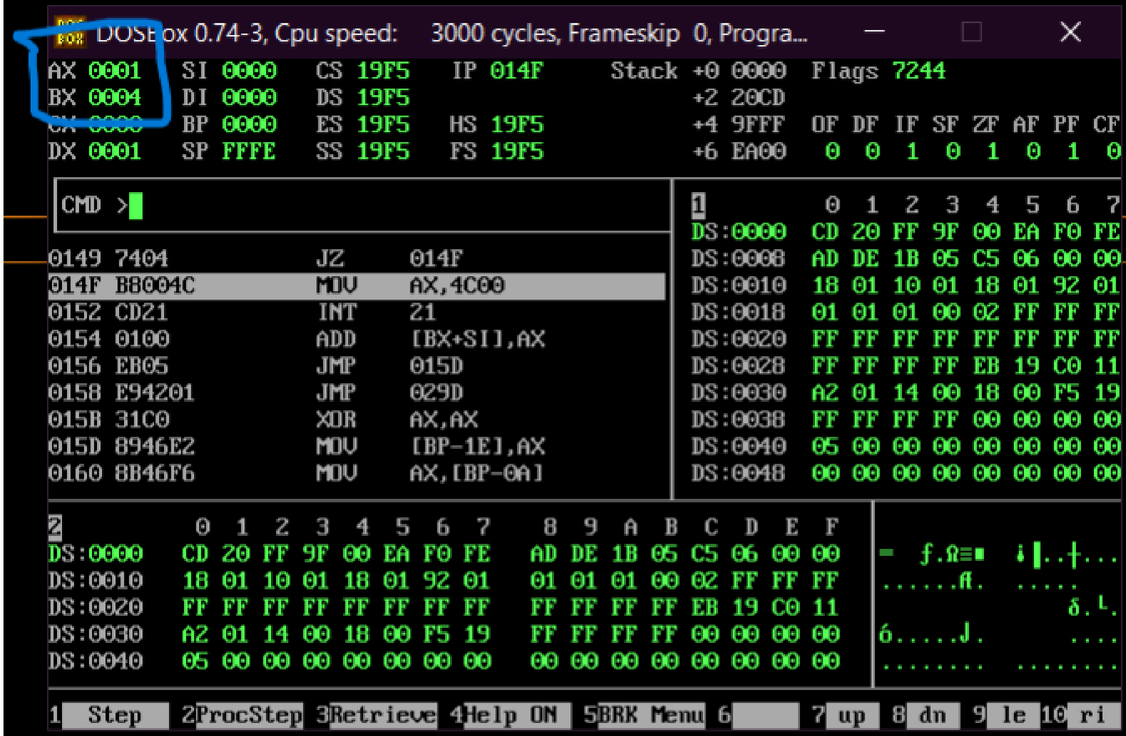


DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Progra...

```
AX 0001   SI 0000   CS 19F5   IP 014F     Stack +0 0000  Flags 7244
BX 0002   DI 0000   DS 19F5                     +2 20CD
CX 0000   BP 0000   ES 19F5   HS 19F5           +4 9FFF  OF DF IF SF ZF AF PF CF
DX 0001   SP FFFE   SS 19F5   FS 19F5           +6 EA00   0  0  1  0  1  0  1  0

 CMD >                                     1           0  1  2  3  4  5  6  7
                                           DS:0000  CD 20 FF 9F 00 EA F0 FE
0149 7404            JZ     014F           DS:0008  AD DE 1B 05 C5 06 00 00
014F B8004C          MOV    AX,4C00        DS:0010  18 01 10 01 18 01 92 01
0152 CD21            INT    21             DS:0018  01 01 01 00 02 FF FF FF
0154 0100            ADD    [BX+SI],AX     DS:0020  FF FF FF FF FF FF FF FF
0156 EB05            JMP    015D           DS:0028  FF FF FF FF EB 19 C0 11
0158 E94201          JMP    029D           DS:0030  A2 01 14 00 18 00 F5 19
015B 31C0            XOR    AX,AX          DS:0038  FF FF FF FF 00 00 00 00
015D 8946E2          MOV    [BP-1E],AX     DS:0040  05 00 00 00 00 00 00 00
0160 8B46F6          MOV    AX,[BP-0A]     DS:0048  00 00 00 00 00 00 00 00

2          0  1  2  3  4  5  6  7   8  9  A  B  C  D  E  F
DS:0000   CD 20 FF 9F 00 EA F0 FE  AD DE 1B 05 C5 06 00 00   =  ƒ.Ω≡■  ¡▌..┼...
DS:0010   18 01 10 01 18 01 92 01  01 01 01 00 02 FF FF FF   ......Æ.  .....
DS:0020   FF FF FF FF FF FF FF FF  FF FF FF FF EB 19 C0 11             δ.└.
DS:0030   A2 01 14 00 18 00 F5 19  FF FF FF FF 00 00 00 00   ó.....⌡.      ....
DS:0040   05 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ........  ........

1 Step  2ProcStep 3Retrieve 4Help ON  5BRK Menu 6       7 up  8 dn  9 le 10 ri
```

First Test Case: 1100010110100011b

Bx = 2 , AX = 1



DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Progra...

```
AX 0001   SI 0000   CS 19F5   IP 014F      Stack +0 0000  Flags 7244
BX 0004   DI 0000   DS 19F5                      +2 20CD
CX 0000   BP 0000   ES 19F5   HS 19F5            +4 9FFF  OF DF IF SF ZF AF PF CF
DX 0001   SP FFFE   SS 19F5   FS 19F5            +6 EA00   0  0  1  0  1  0  1  0

 CMD >                                          1          0  1  2  3  4  5  6  7
                                                DS:0000    CD 20 FF 9F 00 EA F0 FE
0149 7404          JZ      014F                 DS:0008    AD DE 1B 05 C5 06 00 00
014F B8004C        MOV     AX,4C00              DS:0010    18 01 10 01 18 01 92 01
0152 CD21          INT     21                   DS:0018    01 01 01 00 02 FF FF FF
0154 0100          ADD     [BX+SI],AX           DS:0020    FF FF FF FF FF FF FF FF
0156 EB05          JMP     015D                 DS:0028    FF FF FF FF EB 19 C0 11
0158 E94201        JMP     029D                 DS:0030    A2 01 14 00 18 00 F5 19
015B 31C0          XOR     AX,AX                DS:0038    FF FF FF FF 00 00 00 00
015D 8946E2        MOV     [BP-1E],AX           DS:0040    05 00 00 00 00 00 00 00
0160 8B46F6        MOV     AX,[BP-0A]           DS:0048    00 00 00 00 00 00 00 00

2           0  1  2  3  4  5  6  7   8  9  A  B  C  D  E  F
DS:0000    CD 20 FF 9F 00 EA F0 FE  AD DE 1B 05 C5 06 00 00   =  ƒ.Ω≡■  ↓▌..┼...
DS:0010    18 01 10 01 18 01 92 01  01 01 01 00 02 FF FF FF   .......Æ.  .....
DS:0020    FF FF FF FF FF FF FF FF  FF FF FF FF EB 19 C0 11              δ.└.
DS:0030    A2 01 14 00 18 00 F5 19  FF FF FF FF 00 00 00 00   ó.....J.      ....
DS:0040    05 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ........  ........

1 Step  2ProcStep  3Retrieve  4Help ON  5BRK Menu  6       7 up  8 dn  9 le  10 ri
```

2nd Test Case:

1111111111100000b

BX = 4, AX = 1

Question 2:

```
[org 0x0100]


mov ax,6

mov bx,32        ;bx = 0000 0000 0010 0000



mov dx,1         ;dx = 0000 0000 0000 0001


mov cx,ax

sub cx,1

l1:


shl dx,1

sub cx,1

jnz l1


xor bx,dx     ;dx = 0000 0000 0010 0000


mov ax,0x4c00

int 0x021
```

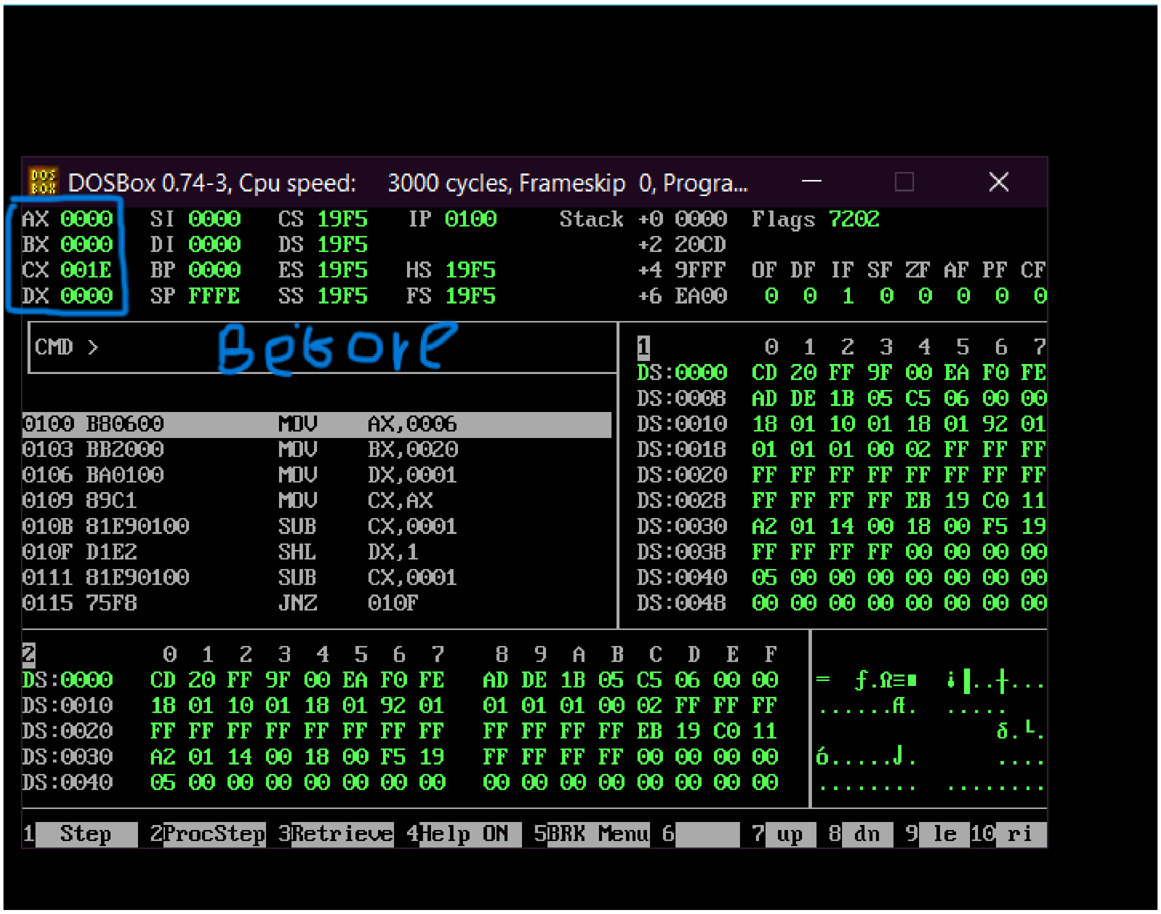
;bx = 0000 0000 0010 0000

;dx = 0000 0000 0010 0000

;XOR----------------------XOR

;   0000 0000 0000 0000

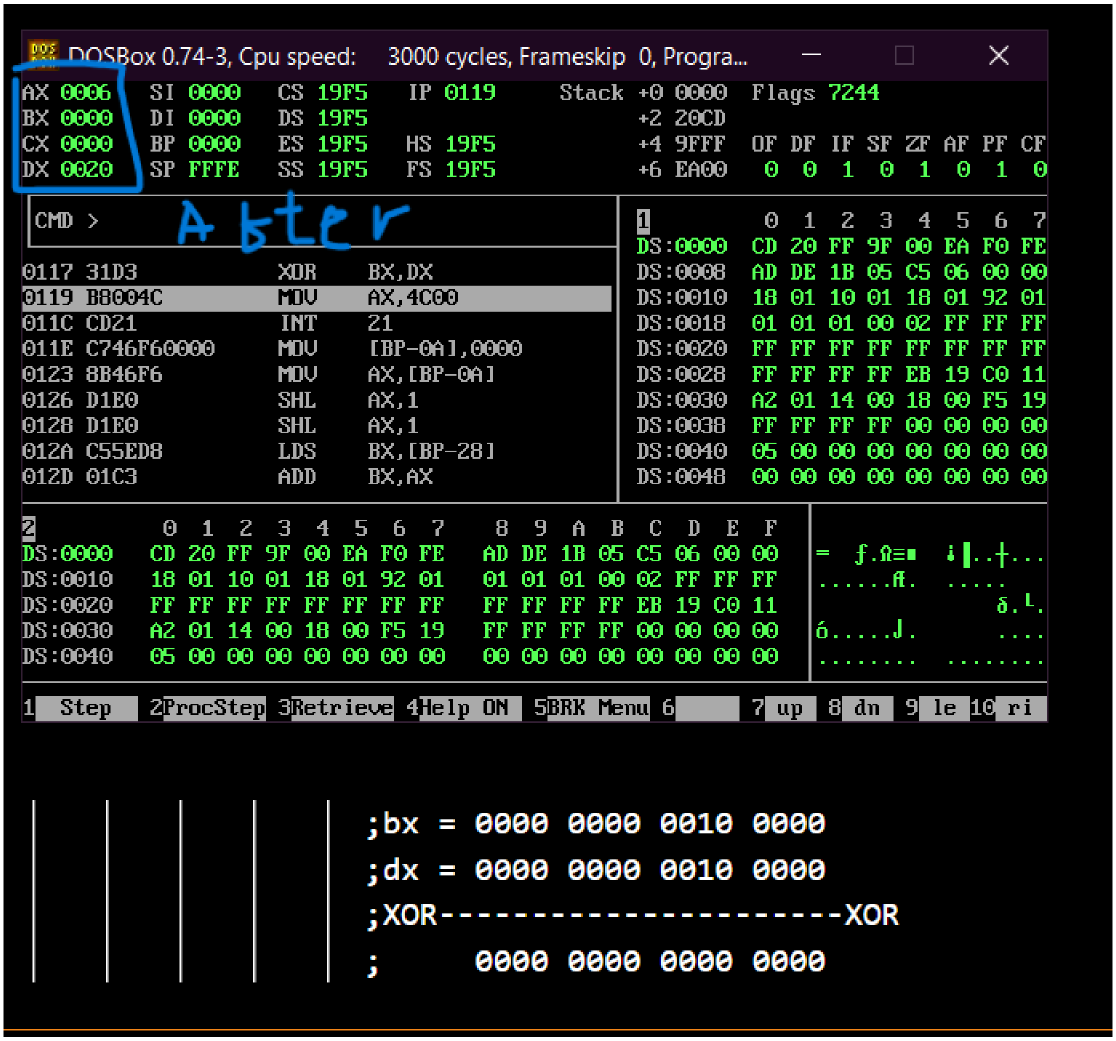
Before:



DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Progra...
Before

```
AX 0000   SI 0000   CS 19F5   IP 0100      Stack +0 0000  Flags 7202
BX 0000   DI 0000   DS 19F5                      +2 20CD
CX 001E   BP 0000   ES 19F5   HS 19F5            +4 9FFF  OF DF IF SF ZF AF PF CF
DX 0000   SP FFFE   SS 19F5   FS 19F5            +6 EA00   0  0  1  0  0  0  0  0

 CMD >                                           1         0  1  2  3  4  5  6  7
                                                 DS:0000  CD 20 FF 9F 00 EA F0 FE
                                                 DS:0008  AD DE 1B 05 C5 06 00 00
0100 B80600          MOV      AX,0006            DS:0010  18 01 10 01 18 01 92 01
0103 BB2000          MOV      BX,0020            DS:0018  01 01 01 00 02 FF FF FF
0106 BA0100          MOV      DX,0001            DS:0020  FF FF FF FF FF FF FF FF
0109 89C1            MOV      CX,AX              DS:0028  FF FF FF FF EB 19 C0 11
010B 81E90100        SUB      CX,0001            DS:0030  A2 01 14 00 18 00 F5 19
010F D1E2            SHL      DX,1               DS:0038  FF FF FF FF 00 00 00 00
0111 81E90100        SUB      CX,0001            DS:0040  05 00 00 00 00 00 00 00
0115 75F8            JNZ      010F               DS:0048  00 00 00 00 00 00 00 00

2          0  1  2  3  4  5  6  7    8  9  A  B  C  D  E  F
DS:0000   CD 20 FF 9F 00 EA F0 FE   AD DE 1B 05 C5 06 00 00   =  ƒ.Ω≡▪  ¡▌..┼...
DS:0010   18 01 10 01 18 01 92 01   01 01 01 00 02 FF FF FF   ......ñ.  .....
DS:0020   FF FF FF FF FF FF FF FF   FF FF FF FF EB 19 C0 11             δ.└.
DS:0030   A2 01 14 00 18 00 F5 19   FF FF FF FF 00 00 00 00   ó.....┘.      ....
DS:0040   05 00 00 00 00 00 00 00   00 00 00 00 00 00 00 00   ........  ........

1 Step  2ProcStep 3Retrieve 4Help ON 5BRK Menu 6      7 up  8 dn  9 le 10 ri
```

After:



DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Progra...

After

```
AX 0006   SI 0000   CS 19F5   IP 0119      Stack +0 0000  Flags 7244
BX 0000   DI 0000   DS 19F5                      +2 20CD
CX 0000   BP 0000   ES 19F5   HS 19F5            +4 9FFF  OF DF IF SF ZF AF PF CF
DX 0020   SP FFFE   SS 19F5   FS 19F5            +6 EA00   0  0  1  0  1  0  1  0

 CMD >

0117 31D3            XOR     BX,DX
0119 B8004C          MOV     AX,4C00
011C CD21            INT     21
011E C746F60000      MOV     [BP-0A],0000
0123 8B46F6          MOV     AX,[BP-0A]
0126 D1E0            SHL     AX,1
0128 D1E0            SHL     AX,1
012A C55ED8          LDS     BX,[BP-28]
012D 01C3            ADD     BX,AX

1              0  1  2  3  4  5  6  7
DS:0000       CD 20 FF 9F 00 EA F0 FE
DS:0008       AD DE 1B 05 C5 06 00 00
DS:0010       18 01 10 01 18 01 92 01
DS:0018       01 01 01 00 02 FF FF FF
DS:0020       FF FF FF FF FF FF FF FF
DS:0028       FF FF FF FF EB 19 C0 11
DS:0030       A2 01 14 00 18 00 F5 19
DS:0038       FF FF FF FF 00 00 00 00
DS:0040       05 00 00 00 00 00 00 00
DS:0048       00 00 00 00 00 00 00 00

2          0  1  2  3  4  5  6  7    8  9  A  B  C  D  E  F
DS:0000   CD 20 FF 9F 00 EA F0 FE   AD DE 1B 05 C5 06 00 00   = ƒ.Ω≡■  ↓▌..┼...
DS:0010   18 01 10 01 18 01 92 01   01 01 01 00 02 FF FF FF   ......Æ.  .....
DS:0020   FF FF FF FF FF FF FF FF   FF FF FF FF EB 19 C0 11             δ.└.
DS:0030   A2 01 14 00 18 00 F5 19   FF FF FF FF 00 00 00 00   ó.....⌡.      ....
DS:0040   05 00 00 00 00 00 00 00   00 00 00 00 00 00 00 00   ........  ........

1 Step  2ProcStep 3Retrieve 4Help ON  5BRK Menu 6       7 up  8 dn  9 le 10 ri
```

```
;bx = 0000 0000 0010 0000
;dx = 0000 0000 0010 0000
;XOR---------------------XOR
;     0000 0000 0000 0000
```

Question #3:

Instruction	AX register (value)	Carry Flag (value)
stc	0000	1
mov ax, 0x0559	022F	1
adc ah,'T'	572F	0
cmc	572F	1
xor ah, al	782F	0
mov cl, 4	782F	0
shr al, cl	7802	1
rcr ah, cl	1702	1