

Lab (i): Addressing data in memory and segments

Part I

1. How to open the debug program?
 -
 -
 -

2. What is the DEBUG command to perform the following operations?
 - a) Begin assembling statements in assembly language, and that will be converted and saved in machine language.
 - b) Display the contents of registers.
 - c) Execute, then display contents of registers.
 - d) Enter machine instructions into memory.
 - e) Display the contents of memory segments.

3. Use DEBUG to enter the following commands:
 - a 100
 - mov BL, 42
 - mov DL, 2A
 - add BL, DL
 - jmp 100

What you can see when the following command is typed?

- a) U 100,107
- b) D CS:100
- c) E CS:100 A2 00 02 03 06 02 02
- d) U 100,106

4. Use DEBUG to enter the following commands:

```
A 100
MOV AX, 0123
ADD AX, 0025
MOV BX, AX
ADD BX, AX
MOV DX, BX
SUB DX, AX
SUB AX, AX
JMP 100
```

What you can see when the following command is typed?

- a) U 100, 111
- b) R
- c) T (repeat 8 times)

What is the value of AX,BX,CX,DX after executing 8 times from the address 100?

What is the alternative to execute 8 times in a single command?

5. Use DEBUG to enter the following command:

```
E CS:100 12 43 56 78 9A BC
```

Now, what you see after D CS:100 ?

The hexadecimal value 43 at address CS:101 was supposed to be 34. Code another E command to correct only the one byte that is incorrect, that is change 43 to 34 directly.

6. Assume that you have used DEBUG to enter the following E command:

```
E CS:100 B8 05 1B 00 2C EB F8
```

Use U command to find what are the 3 assembly/symbolic instruction represented here.

Part II

1. Enter these machine language instructions into the code segment address 100.

B0 1C D0 E0 B3 12 F6 E3 EB F6

Determine which DEBUG command to use. Which memory byte above performs the following operations?

(Hint: you need to convert the above machine language to assembly language)

- a) Move hex value 1C to the AL register.
- b) Shift the contents of AL one bit to the left.
- c) Move the hex value 12 to BL.
- d) Multiply AL by BL.
- e) Jump back to 100

Execute the program and identify what is the final product in AX?

2. What is the output in AH and AL? Why?

```
A 100
MOV AH, 0
MOV AL, 7
MOV BL, 10
MUL BL           ;BL is 8-bit register
JMP 100
```

Which arithmetic operation is performed when you execute MUL BL?

- AL x BL
- AX x BX
- AX x BL
- AL x BX

Why? Would you be able to explain?

3. What is the output in AH and AL? Why?

```
A 100
MOV AH, 0
MOV AL, 83
MOV BL, 2
DIV BL          ;BL is 8-bit register
JMP 100
```

Which arithmetic operation is performed when you execute DIV BL?

- AL / BL
- AX / BX
- AX / BL
- AL / BX

Why? Would you be able to explain?

4. What is the output in AX and DX? Why?

```
A 100
MOV DX, 0
MOV AX, 8003
MOV CX, 100
DIV CX          ;CX is 16-bit register
JMP 10
```

Which arithmetic operation is performed when you execute DIV CX?

- AL / CL
- AX / CX
- AX / CL
- AL / CX

Why? Would you be able to explain?

5. Enter into AL and BL register so that AL contains AA and that an item named BL contains F0. Determine the result on AL for the following unrelated operations by using debug program:

- a) AND AL, BL
- b) OR AL, BL
- c) XOR AL, BL
- d) NOT AL

6. What is the output in AX?

```
A 100  
MOV AL, 8  
SHR AL, 1  
MOV BL, 8  
SHL BL, 1  
JMP 100
```

	AX
MOV AL, 8	
SHR AL, 1	
MOV BL, 8	
SHL BL, 1	

Part III: Independent Practical

1. Update CS register to 116E (Using R command), and enter the following instructions into DEBUG program.

```
-A 100
116E:0100 MOV AX, 0010
116E:0103 MOV BX, 0020
116E:0106 MOV CX, 0030
116E:0109 ADD AX, BX
116E:010B INC BX
116E:010C SUB CX, AX
116E:010E DEC CX
116E:010F JMP 0100
116E:0111 <enter>
```

What is the content of register AX, BX, CX and IP?

AX	BX	CX	IP

2. Trace the content of the registers used in the following program segment:

```
MOV AX, 1
MOV BX, 1
MOV CX, 5
MOV DX, 0
```

A10:

```
ADD AX, BX
MOV DX, AX
MOV AX, BX
MOV BX, DX
LOOP A10
```

	AX	BX	CX	DX
Before Loops				
After 1 st loop				
After 2 nd loop				
After 3 rd loop				
After 4 th loop				
After 5 th loop				

3. Trace the execution of the following instructions and record the values of the register.

```
MOV AX,010
MOV BX,020
MOV CX,030
ADD AX,BX
INC BX
SUB CX,AX
DEC CX
JMP 100
```

AX	BX	CX

4. What is the value of AX and BX?

```
MOV AX,00
MOV BX,00
MOV CX,3           ;Initialize for 3 loops
A20:
INC AX
ADD BX,AX
LOOP A20           ;Decrement CX ;Repeat if nonzero
```

AX	BX	CX

5. What is the final value of AX and BX?

```
MOV AX,0           ;Initialize AX and
MOV BX,0           ;BX to zero,
MOV CX,4           ;CX for 4 loops
A20:
INC AX             ;Add 01 to AX
ADD BX,AX          ;Add AX to BX
LOOP A20           ;Decrement CX, loop if nonzero
```

AX	BX