

# Lab # 4

Name : M.Shafeen

Roll No : 22P-9278

Debugger :

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: AFD

AX	SI	CS	IP	Stack	Flags
0000	0000	000F	0100	+0 0000	7202
BX	0000	DI	0000	+2 20CD	
CX	0012	BP	0000	+4 9FFF	OF DF IF SF ZF AF PF CF
DX	0000	SP	FFFE	+6 EA00	0 0 1 0 0 0 0
		ES	19F5		
		SS	19F5		
		FS	19F5		

CMD > █

Address	Disassembly	Comment
0100 0000	ADD	[BX+SI],AL
0102 0000	ADD	[BX+SI],AL
0104 0000	ADD	[BX+SI],AL
0106 0000	ADD	[BX+SI],AL
0108 0000	ADD	[BX+SI],AL
010A 0000	ADD	[BX+SI],AL
010C 0000	ADD	[BX+SI],AL
010E 0000	ADD	[BX+SI],AL

Address	Disassembly	Comment
DS:000F	F0 60 10 00	F0 60 11 00
DS:0017	F0 60 10 00	F0 60 17 00
DS:001F	C0 60 10 00	F0 60 10 00
DS:0027	F0 80 11 00	F0 60 10 00
DS:002F	F0 60 10 00	F0 60 10 00
DS:0037	F0 60 10 00	F0 60 16 00
DS:003F	F0 60 10 00	F0 60 10 00
DS:0047	F0 60 10 00	F0 60 10 00
DS:004F	F0 60 10 00	F0 60 10 00
DS:0057	F0 60 10 00	F0 60 10 00

Address	Disassembly	Comment
DS:000F	F0 60 10 00	F0 60 11 00
DS:001F	C0 60 10 00	F0 80 11 00
DS:002F	F0 60 10 00	F0 60 16 00
DS:003F	F0 60 10 00	F0 60 10 00
DS:004F	F0 60 10 00	F0 60 10 00

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

Step 1 :

Assigning the address 000F to DS register

```

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: AFD
AX 0000 SI 0000 CS 000F IP 0100 Stack +0 0000 Flags 7202
BX 0000 DI 0000 DS 000E +2 20CD
CX 0012 BP 0000 ES 19F5 HS 19F5 +4 9FFF 0F 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

{R} reg=value
CMD >DS=000F

0100 0000 ADD [BX+SI],AL
0102 0000 ADD [BX+SI],AL
0104 0000 ADD [BX+SI],AL
0106 0000 ADD [BX+SI],AL
0108 0000 ADD [BX+SI],AL
010A 0000 ADD [BX+SI],AL
010C 0000 ADD [BX+SI],AL
010E 0000 ADD [BX+SI],AL

1 F 0 1 2 3 4 5 6
DS:000F F0 60 10 00 F0 60 10 00
DS:0017 F0 60 10 00 F0 60 10 00
DS:001F F0 60 10 00 F0 60 11 00
DS:0027 F0 60 10 00 F0 00 17 00
DS:002F C0 60 10 00 F0 60 10 00
DS:0037 F0 80 11 00 F0 60 10 00
DS:003F F0 60 10 00 F0 60 10 00
DS:0047 F0 60 10 00 F0 00 16 00
DS:004F F0 60 10 00 F0 60 10 00
DS:0057 F0 60 10 00 F0 60 10 00

2 F 0 1 2 3 4 5 6 7 8 9 A B C D E
DS:000F F0 60 10 00 F0 60 10 00 F0 60 10 00 F0 60 10 00
DS:001F F0 60 10 00 F0 60 11 00 F0 60 10 00 F0 00 17 00
DS:002F C0 60 10 00 F0 60 10 00 F0 80 11 00 F0 60 10 00
DS:003F F0 60 10 00 F0 60 10 00 F0 60 10 00 F0 00 16 00
DS:004F F0 60 10 00 F0 60 10 00 F0 60 10 00 F0 60 10 00

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

```

## Step 2 :

Assigning the address **000E** to **CS** register

```

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: AFD
AX 0000 SI 0000 CS 000F IP 0100 Stack +0 0000 Flags 7202
BX 0000 DI 0000 DS 000E +2 20CD
CX 0012 BP 0000 ES 19F5 HS 19F5 +4 9FFF 0F 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

{R} reg=value
CMD >CS=000E

1
DS:000F F0 60 10 00 F0 60 10 00
DS:0017 F0 60 10 00 F0 60 10 00
DS:001F F0 60 10 00 F0 60 11 00
DS:0027 F0 60 10 00 F0 00 17 00
DS:002F C0 60 10 00 F0 60 10 00
DS:0037 F0 80 11 00 F0 60 10 00
DS:003F F0 60 10 00 F0 60 10 00
DS:0047 F0 60 10 00 F0 00 16 00
DS:004F F0 60 10 00 F0 60 10 00
DS:0057 F0 60 10 00 F0 60 10 00

2
F 0 1 2 3 4 5 6 7 8 9 A B C D E
DS:000F F0 60 10 00 F0 60 10 00 F0 60 10 00 F0 60 10 00
DS:001F F0 60 10 00 F0 60 11 00 F0 60 10 00 F0 00 17 00
DS:002F C0 60 10 00 F0 60 10 00 F0 80 11 00 F0 60 10 00
DS:003F F0 60 10 00 F0 60 10 00 F0 60 10 00 F0 00 16 00
DS:004F F0 60 10 00 F0 60 10 00 F0 60 10 00 F0 60 10 00

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

```

## Step 3 :

By adding the **offset** we get to the same **physical address**

```

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: AFD
AX 0000 SI 0000 CS 000F IP 0100 Stack +0 0000 Flags 7202
BX 0000 DI 0000 DS 000E +2 20CD
CX 0012 BP 0000 ES 19F5 HS 19F5 +4 9FFF 0F 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

{R} reg=value
CMD >DS=000E+0001 60

0100 0000 ADD [BX+SI],AL
0102 0000 ADD [BX+SI],AL
0104 0000 ADD [BX+SI],AL
0106 0000 ADD [BX+SI],AL
0108 0000 ADD [BX+SI],AL
010A 0000 ADD [BX+SI],AL
010C 0000 ADD [BX+SI],AL
010E 0000 ADD [BX+SI],AL

1 F 0 1 2 3 4 5 6
DS:000F F0 60 10 00 F0 60 10 00
DS:0017 F0 60 10 00 F0 60 10 00
DS:001F F0 60 10 00 F0 60 11 00
DS:0027 F0 60 10 00 F0 00 17 00
DS:002F C0 60 10 00 F0 60 10 00
DS:0037 F0 80 11 00 F0 60 10 00
DS:003F F0 60 10 00 F0 60 10 00
DS:0047 F0 60 10 00 F0 00 16 00
DS:004F F0 60 10 00 F0 60 10 00
DS:0057 F0 60 10 00 F0 60 10 00

2 F 0 1 2 3 4 5 6 7 8 9 A B C D E
DS:000F F0 60 10 00 F0 60 10 00 F0 60 10 00 F0 60 10 00
DS:001F F0 60 10 00 F0 60 11 00 F0 60 10 00 F0 00 17 00
DS:002F C0 60 10 00 F0 60 10 00 F0 80 11 00 F0 60 10 00
DS:003F F0 60 10 00 F0 60 10 00 F0 60 10 00 F0 00 16 00
DS:004F F0 60 10 00 F0 60 10 00 F0 60 10 00 F0 60 10 00

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

```

## Step 4 :

As you can see both are pointing to the same address

```

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: AFD
AX 0000 SI 0000 CS 000F IP 0100 Stack +0 0000 Flags 7202
BX 0000 DI 0000 DS 000F +2 20CD
CX 0012 BP 0000 ES 19F5 HS 19F5 +4 9FFF 0F 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 >

0100 0000 ADD [BX+SI],AL
0102 0000 ADD [BX+SI],AL
0104 0000 ADD [BX+SI],AL
0106 0000 ADD [BX+SI],AL
0108 0000 ADD [BX+SI],AL
010A 0000 ADD [BX+SI],AL
010C 0000 ADD [BX+SI],AL
010E 0000 ADD [BX+SI],AL

1 F 0 1 2 3 4 5 6
DS:000F F0 60 10 00 F0 60 11 00
DS:0017 F0 60 10 00 F0 00 17 00
DS:001F C0 60 10 00 F0 60 10 00
DS:0027 F0 80 11 00 F0 60 10 00
DS:002F F0 60 10 00 F0 60 10 00
DS:0037 F0 60 10 00 F0 00 16 00
DS:003F F0 60 10 00 F0 60 10 00
DS:0047 F0 60 10 00 F0 60 10 00
DS:004F F0 60 10 00 F0 60 10 00
DS:0057 F0 60 10 00 F0 60 10 00

2 F 0 1 2 3 4 5 6 7 8 9 A B C D E
DS:000F F0 60 10 00 F0 60 11 00 F0 60 10 00 F0 00 17 00
DS:001F C0 60 10 00 F0 60 10 00 F0 80 11 00 F0 60 10 00
DS:002F F0 60 10 00 F0 60 10 00 F0 60 10 00 F0 00 16 00
DS:003F F0 60 10 00 F0 60 10 00 F0 60 10 00 F0 60 10 00
DS:004F F0 60 10 00 F0 60 10 00 F0 60 10 00 F0 60 10 00

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

```

Showing Memory now :

As you can see both memory are the same

— ×

CMD >			60	1	F	0	1	2	3	4	5	6
0100	0000	ADD	[BX+SI],AL	DS:000F	F0	60	10	00	F0	60	11	00
0102	0000	ADD	[BX+SI],AL	DS:0017	F0	60	10	00	F0	00	17	00
0104	0000	ADD	[BX+SI],AL	DS:001F	C0	60	10	00	F0	60	10	00
0106	0000	ADD	[BX+SI],AL	DS:0027	F0	80	11	00	F0	60	10	00
0108	0000	ADD	[BX+SI],AL	DS:002F	F0	60	10	00	F0	60	10	00
010A	0000	ADD	[BX+SI],AL	DS:0037	F0	60	10	00	F0	00	16	00
010B	0000	ADD	[BX+SI],AL	DS:003F	F0	60	10	00	F0	60	10	00
010A	0000	ADD	[BX+SI],AL	DS:0047	F0	60	10	00	F0	60	10	00
010C	0000	ADD	[BX+SI],AL	DS:004F	F0	60	10	00	F0	60	10	00
010E	0000	ADD	[BX+SI],AL	DS:0057	F0	60	10	00	F0	60	10	00

2	F	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E			
DS:000F	F0	60	10	00	F0	60	11	00	F0	60	10	00	F0	00	17	00	≡	...	≡
DS:001F	C0	60	10	00	F0	60	10	00	F0	80	11	00	F0	60	10	00	L	...	≡
DS:002F	F0	60	10	00	F0	60	10	00	F0	60	10	00	F0	00	16	00	≡	...	≡
DS:003F	F0	60	10	00	F0	60	10	00	F0	60	10	00	F0	60	10	00	≡	...	≡
DS:004F	F0	60	10	00	F0	60	10	00	F0	60	10	00	F0	60	10	00	≡	...	≡

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