

## PRACTICAL: 3(A3)

**AIM:** Write a program to perform selective clear operation on data stored at 4000H corresponding to the data stored at 4001H and store the result at 4002H. Verify the result and write bite wise operation of this program. ( A AND B')

### CODE:

```
org 100h
```

```
MOV [4000H], 10010101B ;149
```

```
MOV [4001H], 01010010B ;82
```

```
MOV AL, [4000H]
```

```
MOV BL, [4001H]
```

```
NOT BL
```

```
ADD AL, BL
```

```
MOV [4002H], AL
```

```
ret
```

**OUTPUT:**

emulator: PRAC\_3A3.com\_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	42
BX	00	AD
CX	00	19
DX	00	00
CS	F400	
IP	0154	
SS	0700	
SP	FFFA	
BP	0000	
SI	0000	
DI	0000	
DS	0700	
ES	0700	

F400:0154

F4150:	FF	255	RES
F4151:	FF	255	RES
F4152:	CD	205	=
F4153:	20	032	SPA
F4154:	CF	207	±
F4155:	00	000	NULL
F4156:	00	000	NULL
F4157:	00	000	NULL
F4158:	00	000	NULL
F4159:	00	000	NULL
F415A:	00	000	NULL
F415B:	00	000	NULL
F415C:	00	000	NULL
F415D:	00	000	NULL
F415E:	00	000	NULL
F415F:	00	000	NULL
F4160:	FF	255	RES
F4161:	FF	255	RES
F4162:	CD	205	=
F4163:	1A	026	→
F4164:	CF	207	±
F4165:	00	000	NULL

F400:0154

BIOS DI  
INT 020h  
IRET

ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD BH, BH
DEC BP
SBB CL, BH
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD [BX + SI], AL
ADD BH, BH
DEC BP
ADD BH, CL
ADD [BX + SI], AL
ADD [BX + SI], AL
...

screen source reset aux vars debug stack flags

Random Access Memory

4002 update table list

0700:4002	42	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	B.....
0700:4012	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
0700:4022	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
0700:4032	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
0700:4042	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
0700:4052	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
0700:4062	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
0700:4072	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....

**CONCLUSION:**

We learned about NOT command and its implementation.