

PRACTICAL: 2(C6)

AIM: Subtract the 16-bit number in memory locations 4002H and 4003H from the 16-bit number in memory locations 4000H and 4001H. The most significant eight bits of the two numbers are in memory locations 4001H and 4003H. Store the result in memory locations 4004H and 4005H with the most significant byte in Memory location 4005H.

CODE:

org 100h

MOV [4000H], 1234H

MOV [4002H], 1111H

MOV AX, [4000H]

MOV BX, [4002H]

SBB AX, BX

MOV [4004H], AX

ret

OUTPUT:

emulator: PRAC_2C6.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	01	23
BX	11	11
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
I RET
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

4004 update table list

0700:4004	23	01	00	00	00	00	00	00	00	00	00	00	00	00	00	00	#0.....
0700:4014	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0700:4024	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0700:4034	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0700:4044	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0700:4054	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0700:4064	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0700:4074	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

CONCLUSION:

We learned about SBB command and its implementation for 16-Bit Numbers.