

## PRACTICAL – 28

**AIM:** Write an assembly language program to multiply two 16-bit numbers in memory and store the result in memory.

**CODE:**

org 100h

MOV [4000h],0034h

MOV [4002h],175Ch

MOV AX,[4000H]

MOV BX,[4002H]

MUL BX

MOV [4004H],DH

MOV [4005H],DL

MOV [4006H],AH

MOV [4007H],AL

ret

**OUTPUT:**

emulator: Practical-28.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	BE	B0
BX	17	5C
CX	00	24
DX	00	04
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

0700:4004 update table list

0700:4004	00	04	BE	B0	00	00	00	00	00-00	00	00	00	00	00	00	00	00	00	00
0700:4014	00	00	00	00	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	00	00	00	00
0700:4034	00	00	00	00	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	00	00	00	00
0700:4054	00	00	00	00	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	00	00	00	00
0700:4074	00	00	00	00	00	00	00	00	00-00	00	00	00	00	00	00	00	00	00	00

**CONCLUSION:** In this Practical we learnt how to multiply two 16 bit number and store the result in memory.