Fall Semester 2021-2022

Microprocessor and Interfacing

Lab Report

Digital Assignment-3

Experiment No: 4 Task No: 3

Course Code: CSE2006

Slot: L7+L8

Submitted By: Alokam Nikhitha

Reg. Numb: 19BCE2555

Submitted To: Dr. Abdul Majed KK

EXPERIMENT 4:

Programs to find Factorial of a number

Aim:

1. Write an Assembly Language Programme (ALP) to find the factorial of a number 6.

2. Write an Assembly Language Programme (ALP) to find the factorial of a number 7.

Factorial of a number

Algorithm:

1) Move the value of whose factorial to be found in to the Accumulator.

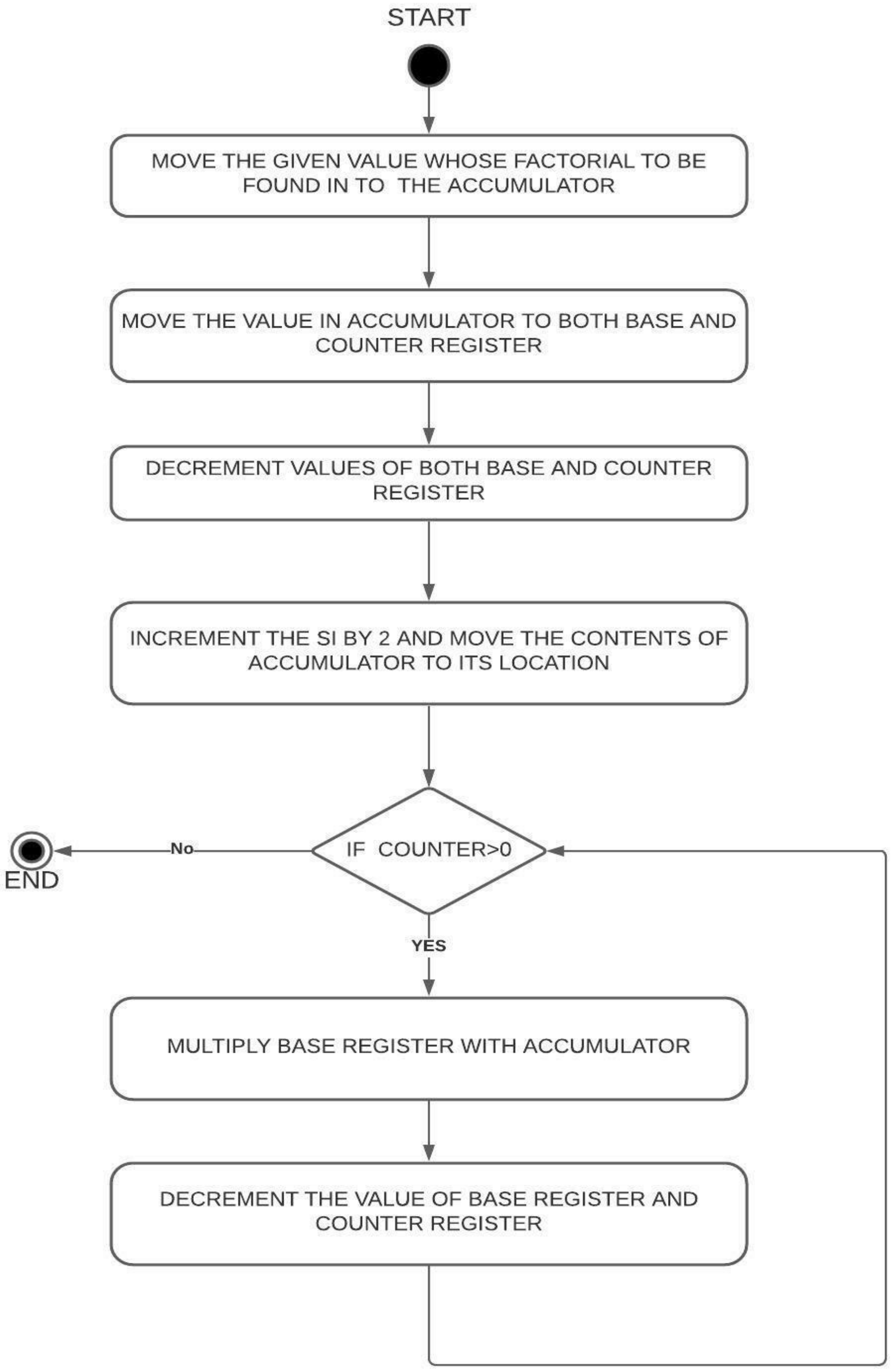
2) Move the value in Accumulator to both Base Register and Counter Register.

3) Decrement both Base Register and Counter Register
4) Multiply the Base Register to Accumulator and decrement the Base Register inside a loop

5) Loop is repeated and Base register is multiplied to Accumulator till the counter becomes 0.

6)The Factorial of the Given value is stored in Accumulator.

FLOW CHART


START
MOVE THE GIVEN VALUE WHOSE FACTORIAL TO BE FOUND IN TO THE ACCUMULATOR
MOVE THE VALUE IN ACCUMULATOR TO BOTH BASE AND COUNTER REGISTER
DECREMENT VALUES OF BOTH BASE AND COUNTER REGISTER
INCREMENT THE SI BY 2 AND MOVE THE CONTENTS OF ACCUMULATOR TO ITS LOCATION
IF COUNTER>0
No
END
YES
MULTIPLY BASE REGISTER WITH ACCUMULATOR
DECREMENT THE VALUE OF BASE REGISTER AND COUNTER REGISTER

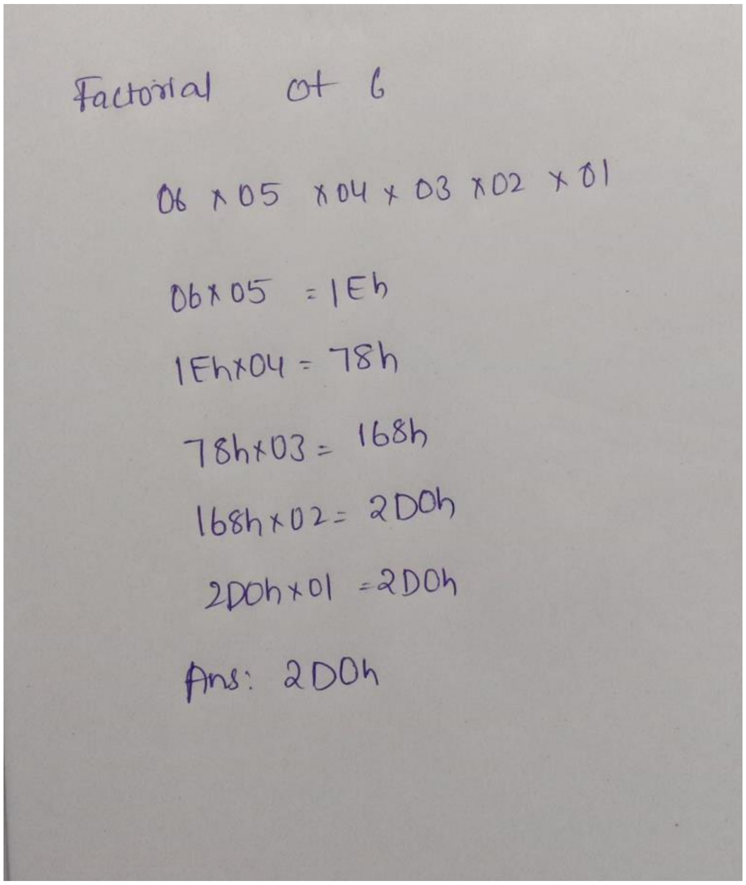
1. Write an Assembly Language Programme (ALP) to find the factorial of a number 6.

Design and Calculations:

Here we input the value ‘6’ in to the accumulator and Move the value from Accumulator to Base Register and Counter Register and Decrement the value of both Base and Counter registers and run a loop and Multiply the Base register . And decrement the Base Register and loop is repeated until the counter becomes 0

Calculations:

6*5*4*3*2*1=2D0[Hex] (720[Dec])



Factorial of 6

06 x 05 x 04 x 03 x 02 x 01

06 x 05 = 1Eh

1Eh x 04 = 78h

78h x 03 = 168h

168h x 02 = 2D0h

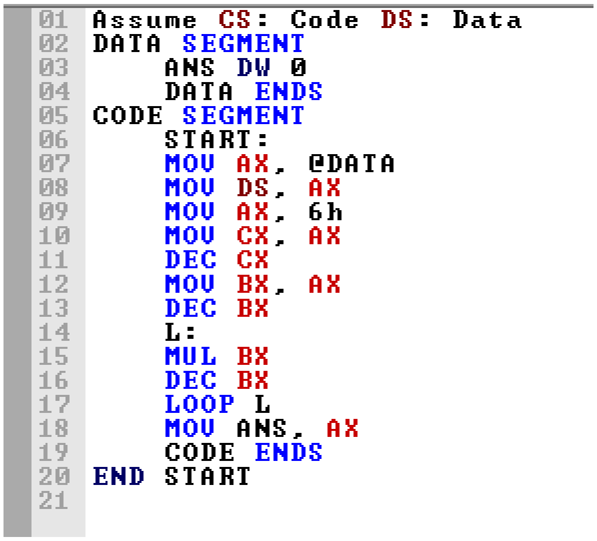
2D0h x 01 = 2D0h

Ans: 2D0h

Program Code:

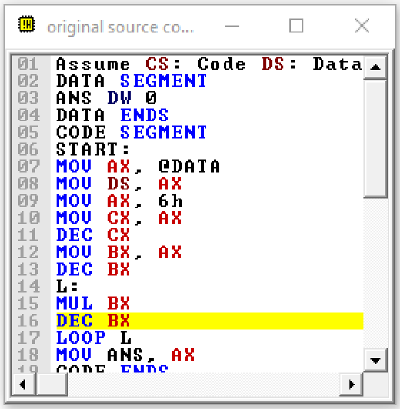
```
Assume CS: Code DS: Data
DATA SEGMENT
    ANS DW 0
    DATA ENDS
CODE SEGMENT
    START:
    MOV AX, @DATA
    MOV DS, AX
    MOV AX, 6h
    MOV CX, AX
    DEC CX
    MOV BX, AX
    DEC BX
    L:
    MUL BX
    DEC BX
    LOOP L
    MOV ANS, AX
    CODE ENDS
```

END START

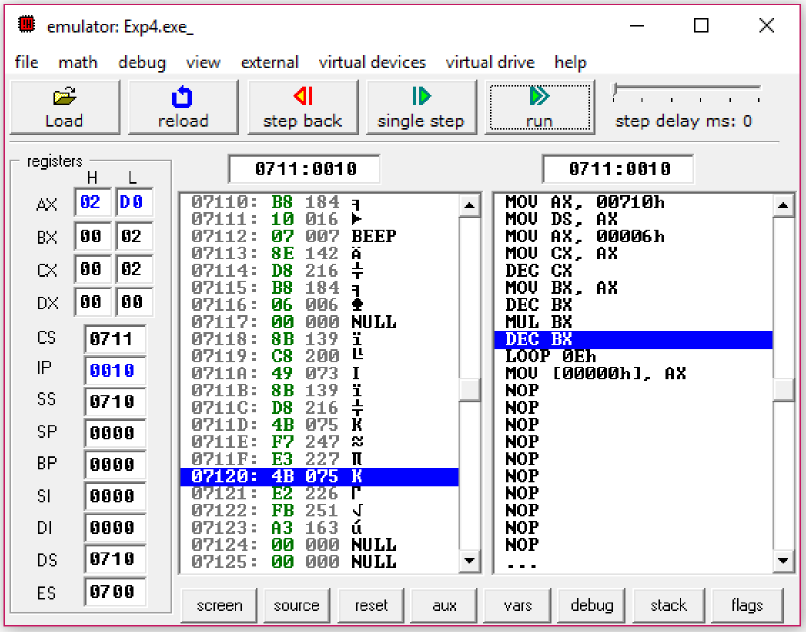


```
Assume CS: Code DS: Data
DATA SEGMENT
     ANS DW 0
     DATA ENDS
CODE SEGMENT
     START:
     MOV AX, @DATA
     MOV DS, AX
     MOV AX, 6h
     MOV CX, AX
     DEC CX
     MOV BX, AX
     DEC BX
     L:
     MUL BX
     DEC BX
     LOOP L
     MOV ANS, AX
     CODE ENDS
END START
```

Output:



```
Assume CS: Code DS: Data
DATA SEGMENT
ANS DW 0
DATA ENDS
CODE SEGMENT
START:
MOV AX, @DATA
MOV DS, AX
MOV AX, 6h
MOV CX, AX
DEC CX
MOV BX, AX
DEC BX
L:
MUL BX
DEC BX
LOOP L
MOV ANS, AX
CODE ENDS
```



```
    H  L
AX  02 D0
BX  00 02
CX  00 02
DX  00 00
CS  0711
IP  0010
SS  0710
SP  0000
BP  0000
SI  0000
DI  0000
DS  0710
ES  0700
```

```
0711:0010

07110: B8 184 ╕
07111: 10 016 ►
07112: 07 007 BEEP
07113: 8E 142 Ä
07114: D8 216 ╪
07115: B8 184 ╕
07116: 06 006 ♠
07117: 00 000 NULL
07118: 8B 139 ï
07119: C8 200 ╚
0711A: 49 073 I
0711B: 8B 139 ï
0711C: D8 216 ╪
0711D: 4B 075 K
0711E: F7 247 ≈
0711F: E3 227 π
07120: 4B 075 K
07121: E2 226 Γ
07122: FB 251 √
07123: A3 163 ú
07124: 00 000 NULL
07125: 00 000 NULL
```

```
0711:0010

MOV AX, 00710h
MOV DS, AX
MOV AX, 00006h
MOV CX, AX
DEC CX
MOV BX, AX
DEC BX
MUL BX
DEC BX
LOOP 0Eh
MOV [00000h], AX
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...
```

Result and Inference:

-The Factorial of 6 (720[dec]-> 2D0[hex]) 2D0 is stored in Accumulator .

-Both Base and Counter Registers becomes 0.

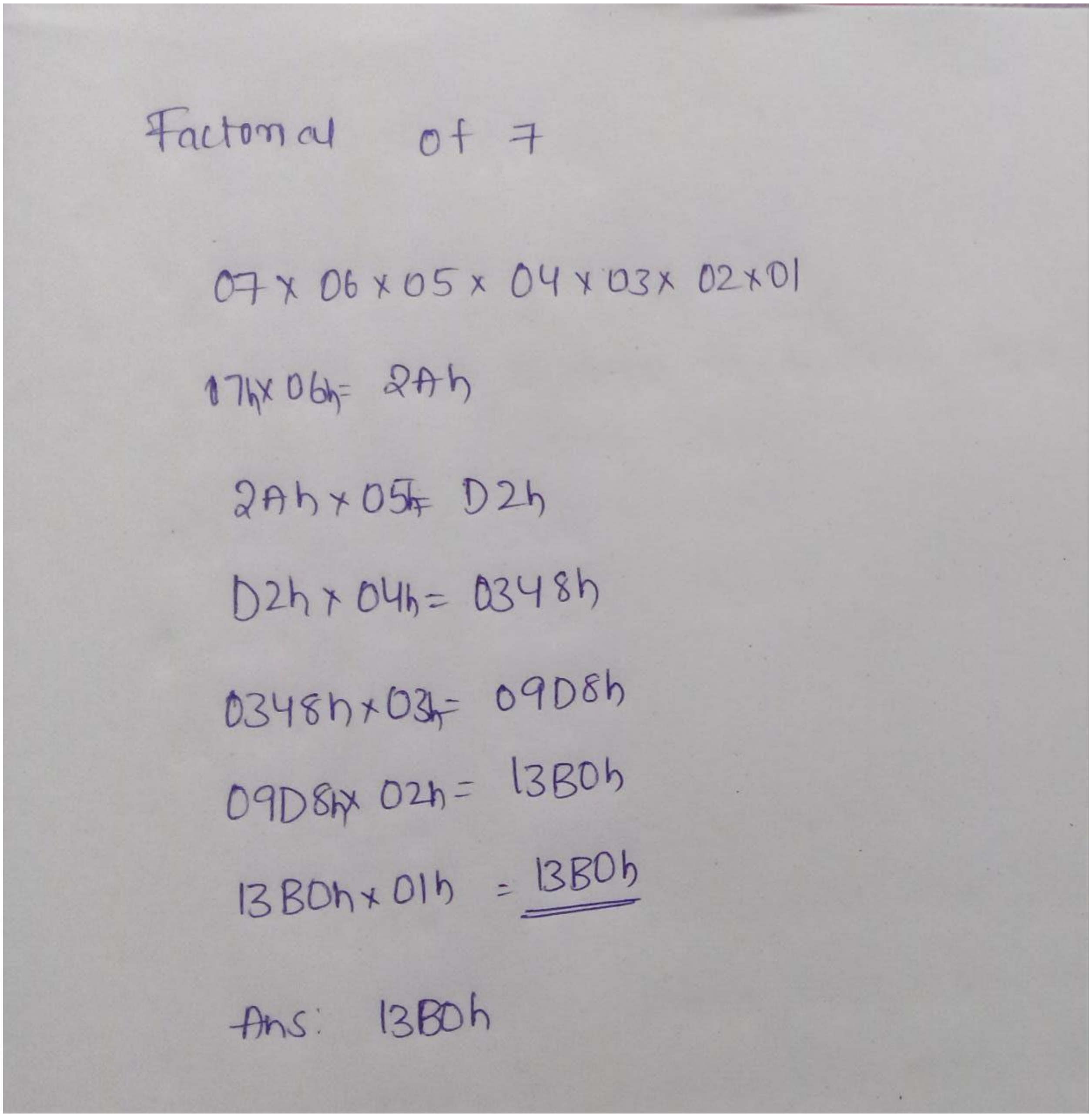
2. Write an Assembly Language Programme (ALP) to find the factorial of a number 7.

Design and Calculations:

Here we input the value ‘7’ in to the accumulator and Move the value from Accumulator to Base Register and Counter Register and Decrement the value of both Base and Counter registers and run a loop and Multiply the Base register . And decrement the Base Register and loop is repeated until the counter becomes 0

Calculations:

7*6*5*4*3*2*1=13B0[Hex] (5040[Dec])



Factorial of 7

07 × 06 × 05 × 04 × 03 × 02 × 01

07h × 06h = 2Ah

2Ah × 05h = D2h

D2h × 04h = 0348h

0348h × 03h = 09D8h

09D8h × 02h = 13B0h

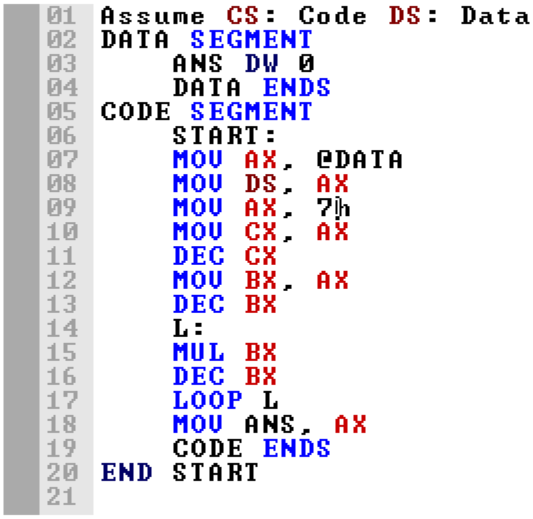
13B0h × 01h = $\underline{\underline{\text{13B0h}}}$

Ans: 13B0h

Program Code:

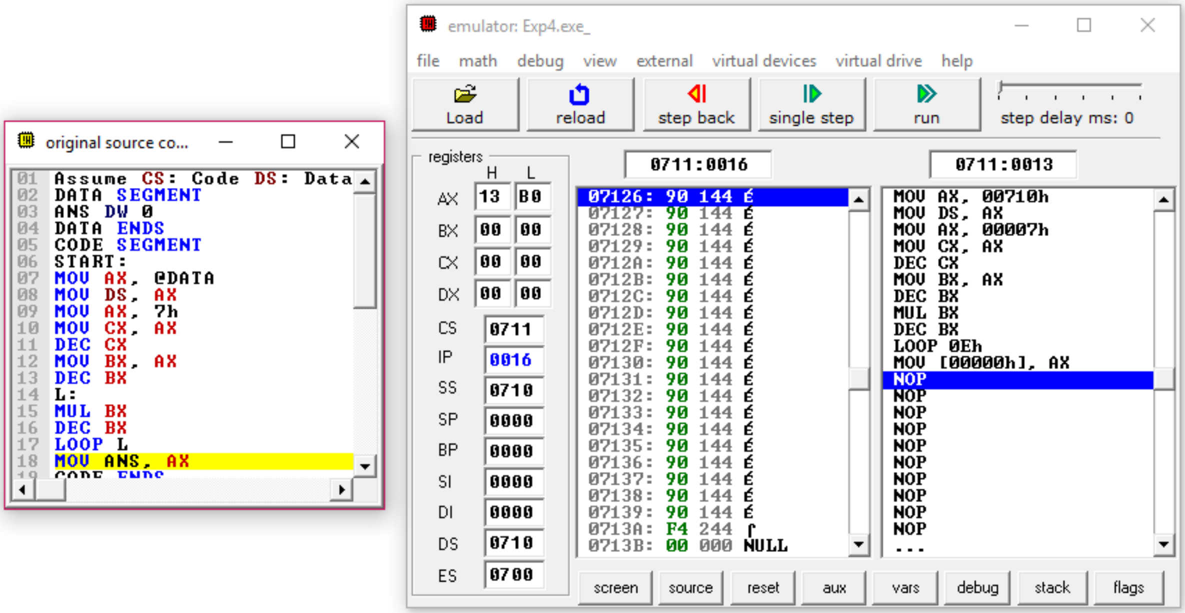
```
Assume CS: Code DS: Data
DATA SEGMENT
    ANS DW 0
    DATA ENDS
CODE SEGMENT
    START:
    MOV AX, @DATA
    MOV DS, AX
    MOV AX, 7h
    MOV CX, AX
    DEC CX
    MOV BX, AX
    DEC BX
    L:
    MUL BX
    DEC BX
    LOOP L
    MOV ANS, AX
    CODE ENDS
```

END START



```
Assume CS: Code DS: Data
DATA SEGMENT
     ANS DW 0
     DATA ENDS
CODE SEGMENT
     START:
     MOV AX, @DATA
     MOV DS, AX
     MOV AX, 7h
     MOV CX, AX
     DEC CX
     MOV BX, AX
     DEC BX
     L:
     MUL BX
     DEC BX
     LOOP L
     MOV ANS, AX
     CODE ENDS
END START
```

OUTPUT:



```
Assume CS: Code DS: Data
DATA SEGMENT
ANS DW 0
DATA ENDS
CODE SEGMENT
START:
MOV AX, @DATA
MOV DS, AX
MOV AX, 7h
MOV CX, AX
DEC CX
MOV BX, AX
DEC BX
L:
MUL BX
DEC BX
LOOP L
MOV ANS, AX
CODE ENDS
```

	H	L
AX	13	B0
BX	00	00
CX	00	00
DX	00	00
CS	0711	
IP	0016	
SS	0710	
SP	0000	
BP	0000	
SI	0000	
DI	0000	
DS	0710	
ES	0700	

```
0711:0016
07126: 90 144 É
07127: 90 144 É
07128: 90 144 É
07129: 90 144 É
0712A: 90 144 É
0712B: 90 144 É
0712C: 90 144 É
0712D: 90 144 É
0712E: 90 144 É
0712F: 90 144 É
07130: 90 144 É
07131: 90 144 É
07132: 90 144 É
07133: 90 144 É
07134: 90 144 É
07135: 90 144 É
07136: 90 144 É
07137: 90 144 É
07138: 90 144 É
07139: 90 144 É
0713A: F4 244 ſ
0713B: 00 000 NULL
```

```
0711:0013
MOV AX, 00710h
MOV DS, AX
MOV AX, 00007h
MOV CX, AX
DEC CX
MOV BX, AX
DEC BX
MUL BX
DEC BX
LOOP 0Eh
MOV [00000h], AX
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...
```

Result and Inference:

-The Factorial of 7 (5040[dec]-> 13B0[hex]) 13B0 is stored in Accumulator .

-Both Base and Counter Registers becomes 0.