Fall Semester 2021-2022

Microprocessor and Interfacing

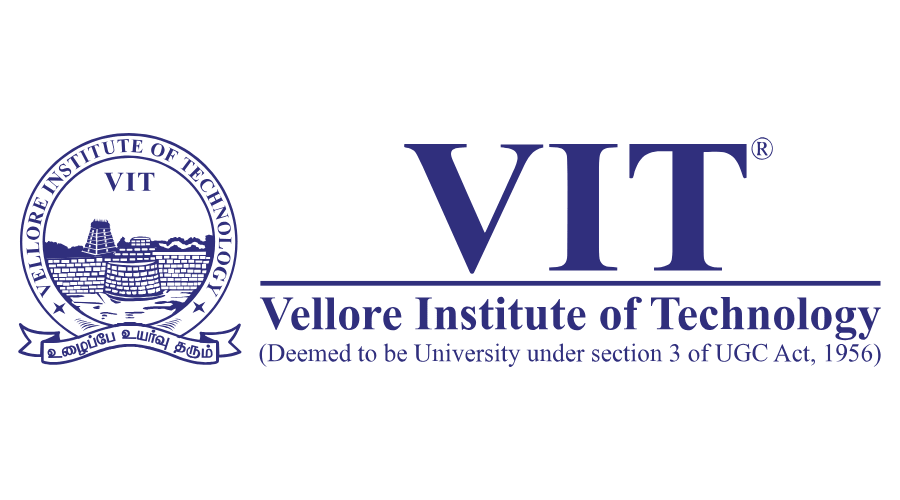
Lab Report

Digital Assignment-4

Experiment No: 5 Task No: 4

Course Code: CSE2006

Slot: L7+L8



Submitted By: Alokam Nikhitha

Reg. Numb: 19BCE2555

Submitted To: Dr. Abdul Majed KK

**EXPERIMENT 5:**

**Aim:**

**1. Write an ALP Program to find LCM of a given numbers**

**2. Write an ALP program to find the average of N numbers.**

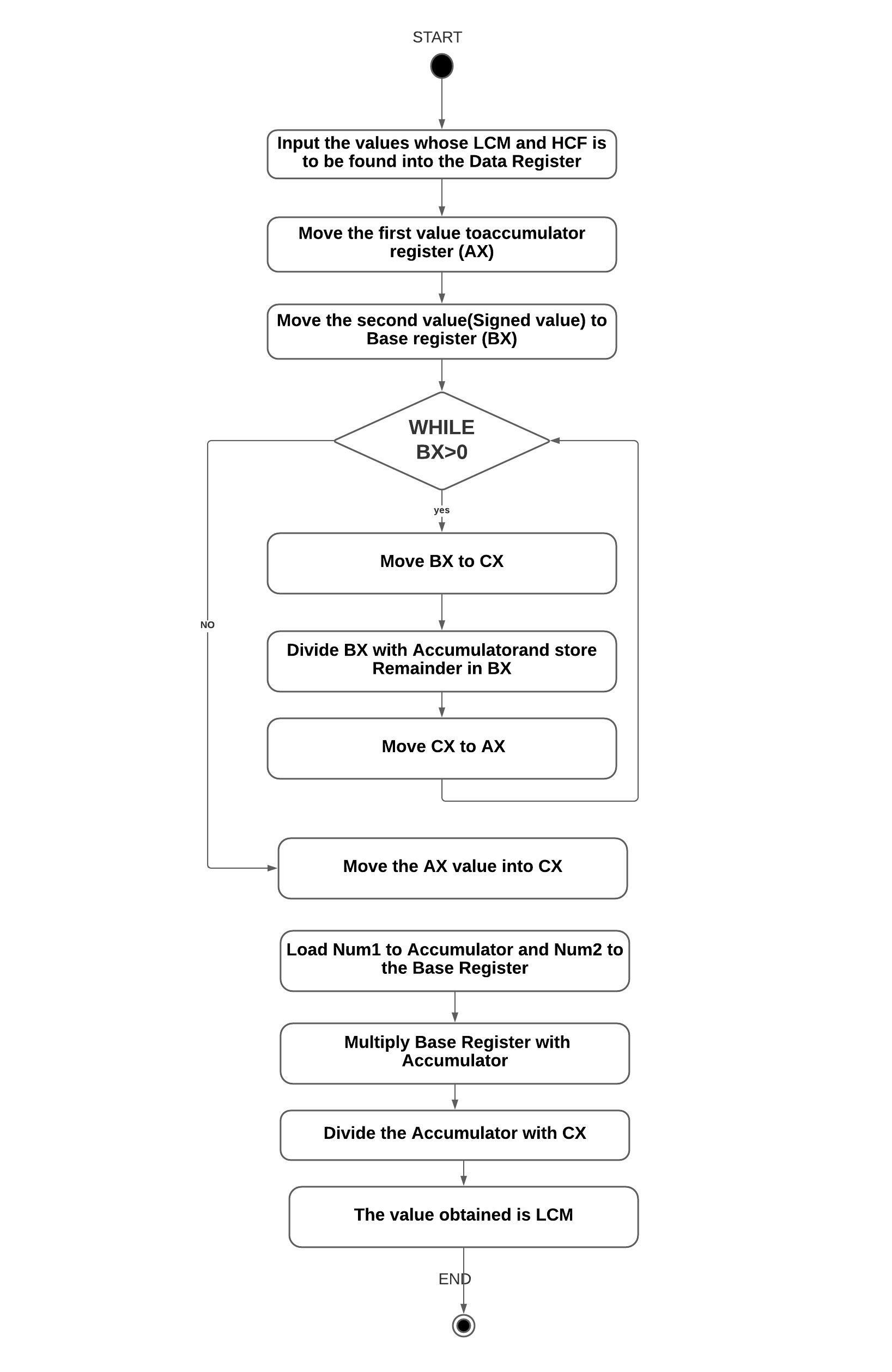
**3. Write an ALP to find the greatest among two numbers.**

**Tool Used: emu8086 simulator**

**1. Write an ALP Program to find LCM of a given numbers**

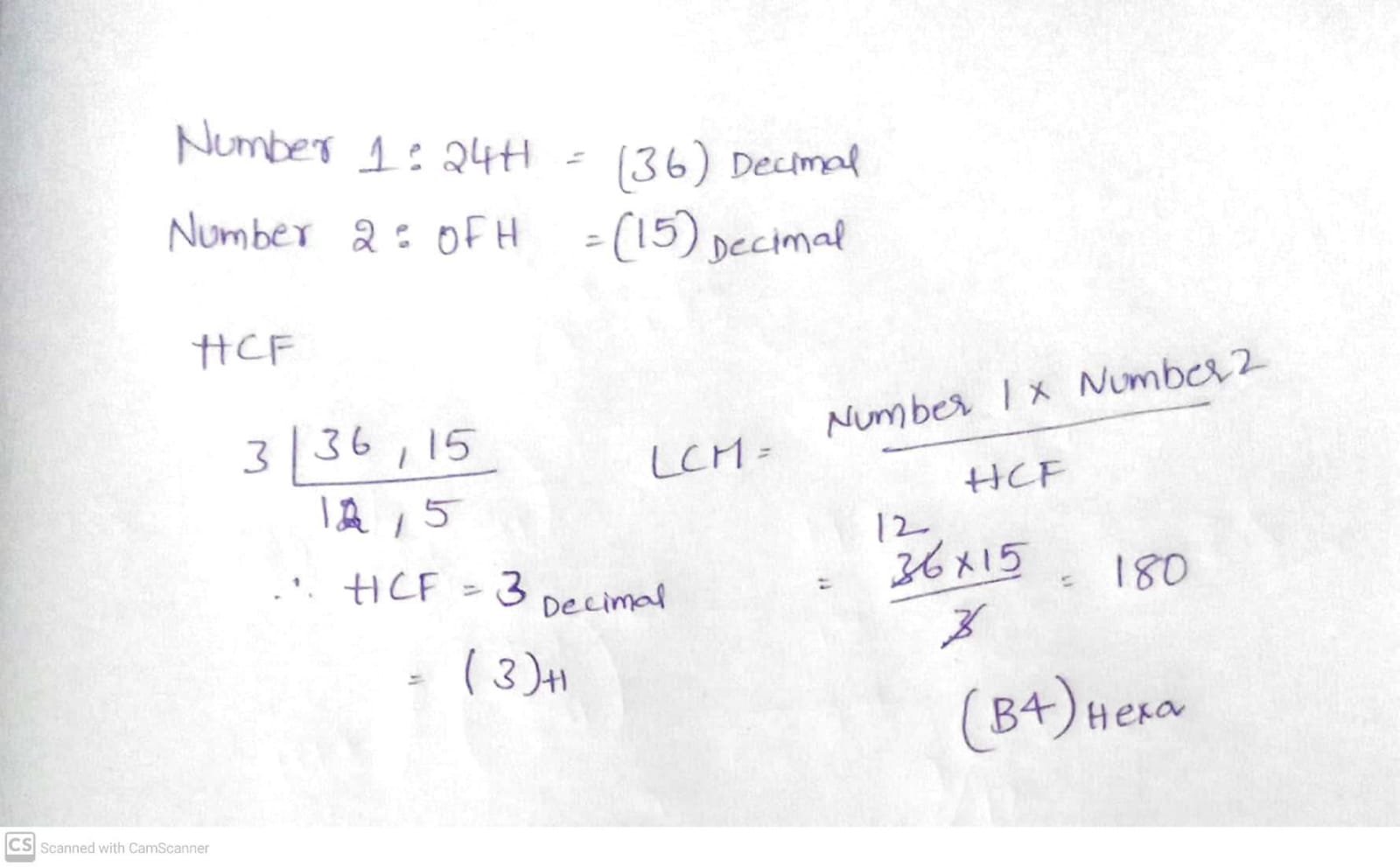
**Algorithm:**

* **Input the values whose LCM and HCF is to be found into the Data Register**
* **Move the first value to accumulator register (AX).**
* **Move the second value(Signed value) to Base register (BX).**
* **Create a while loop until until BX becomes 0**
* **Move BX to CX.**
* **Divide BX with Accumulator and store Remainder in BX**
* **Move CX to AX and repeat the While loop**
* **After the While loop ends the value in AX is HCF**
* **Move the AX value into CX**
* **Load Num1 to Accumulator and Num2 to the Base Register**
* **Multiply Base Register with Accumulator**
* **Divide the Accumulator with CX(HCF)**
* **The value obtained is LCM(stored in AX)**
* **Halt the overall process.**

**Flow Chart: **

**Design and Calculations:**

**Input the values whose LCM and HCF is to be found into the Data Register.Move the first value to accumulator register (AX).Move the second value(Signed value) to Base register (BX).Create a while loop until until BX becomes 0.Move BX to CX.**

**Divide BX with Accumulator and store Remainder in BX.Move CX to AX and repeat the While loop.After the While loop ends the value in AX is HCF.Move the AX value into CX.Load Num1 to Accumulator and Num2 to the Base Register .Multiply Base Register with Accumulator.Divide the Accumulator with CX(HCF).The value obtained is LCM(stored in AX) **

**Program Code:**

**ASSUME CS:CODE DS:DATA**

**DATA SEGMENT**

**NUM1 DW 24H**

**NUM2 DW 0FH**

**HCF DW ?**

**LCM DW ?**

**ENDS**

**CODE SEGMENT**

**ASSUME DS:DATA CS:CODE**

**START:**

**MOV AX,DATA**

**MOV DS,AX**

**MOV AX,NUM1**

**MOV BX,NUM2**

**WHILE:MOV DX,0**

**MOV CX,BX**

**DIV BX**

**MOV BX,DX**

**MOV AX,CX**

**CMP BX,0**

**JNE WHILE**

**MOV HCF,AX**

**MOV CX,AX**

**MOV AX,NUM1**

**MOV BX,NUM2**

**MUL BX**

**DIV CX**

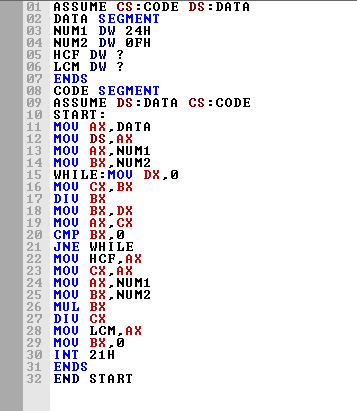
**MOV LCM,AX**

**MOV BX,0**

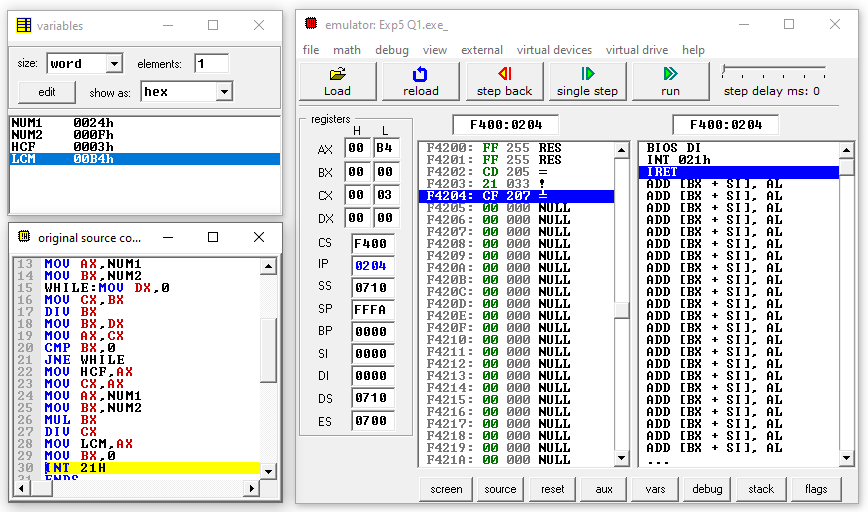
**INT 21H**

**ENDS**

**END START**

****

**Output:**

****

**Result and Inference:**

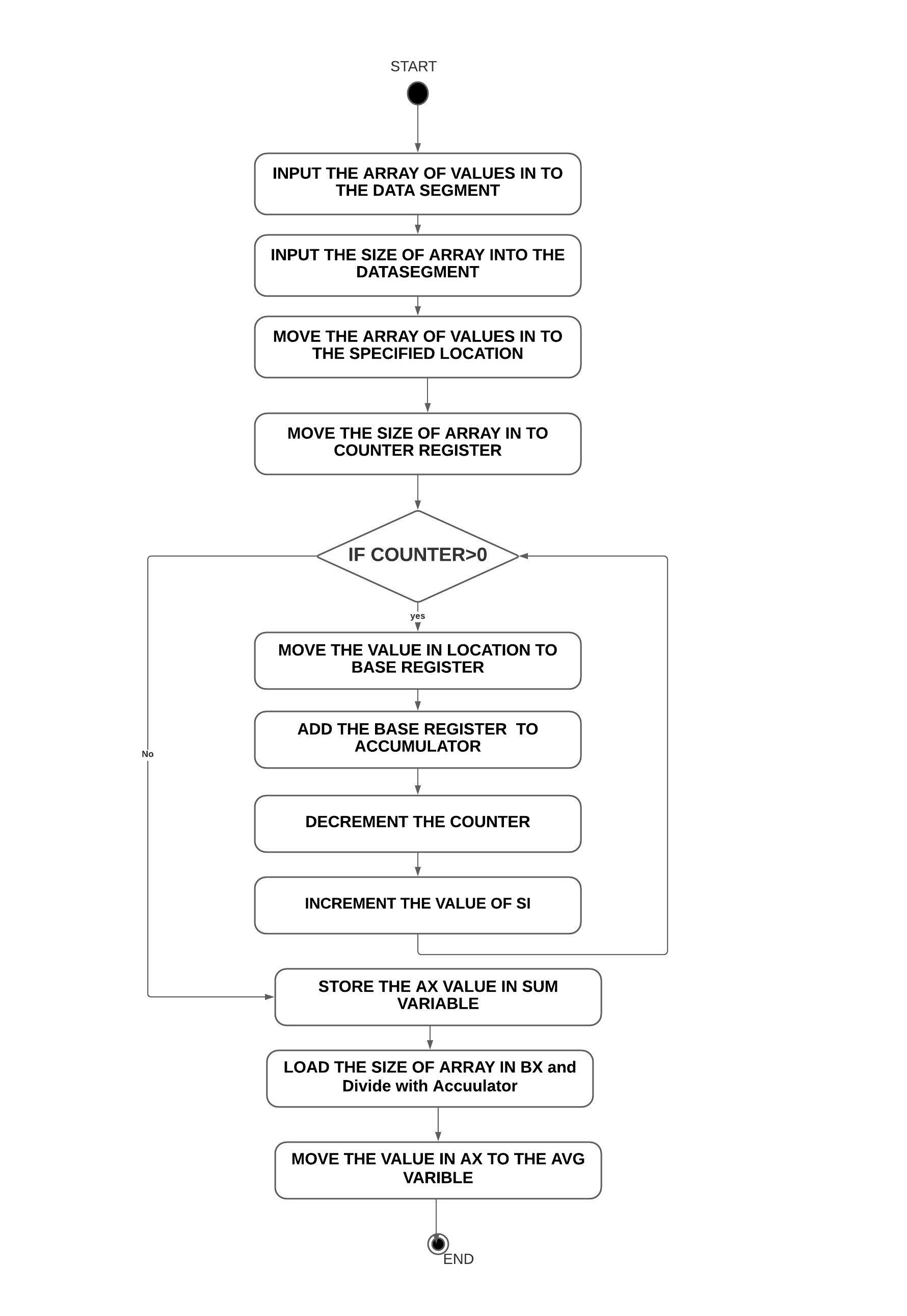
**-The HCF of the 2 input values in Stored in CX and LCM is stored in AX and Variable we can see that HCF is 0003H and LCM is 00B4H**

**2) Write an ALP program to find the average of N numbers.**

**Algorithm**

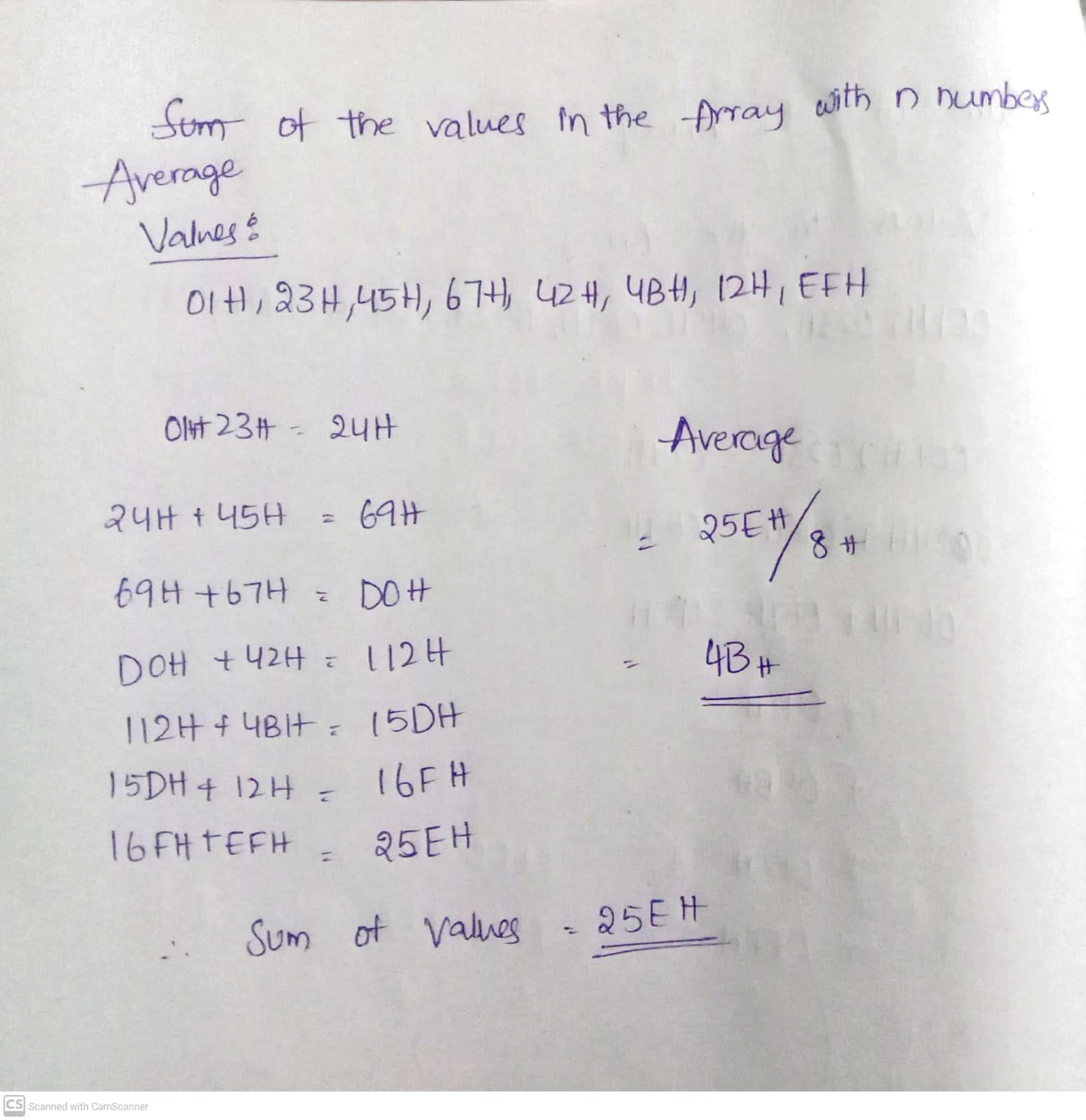
* **Input the Array of Values in Data Segment**
* **Input the Size of the Array in Data segment**
* **Take SUM and AVG in DataSegment**
* **Move all the values of the Array to a specified Memory Location(SI) in Code Segment**
* **Move the size of Array from Data segment in to Counter Register**
* **Now Run a loop and Add move the values in SI location of to Base Register and add that to Accumulator .Increment the SI value and**
* **Decrement the Counter Register**
* **Repeat the Process until the Counter becomes 0**
* **The Sum is stored in Accumulator.**
* **Move Size of Array in to BX**
* **-Divide the Sum with BX**
* **-Move the value to AVG**

**Flow chart:**

****

**Design and Calculations:**

**Input the Array of Values(001H, 023H, 045H, 067H, 042H, 04BH, 012H, 0EFH ) in Data Segment .Input the Size of the Array(8) in Data segment.Move all the values of the Array to a specified Memory Location(SI) in Code Segment.Move the size of Array from Data segment in to Counter Register.Now Run a loop and Add move the values in SI location of to Base Register and add that to Accumulator .Increment the SI value and Decrement the Counter Register.Repeat the Process until the Counter becomes 0 .The Sum is stored in Accumulator. Move Size of Array in to BX.Divide the Sum with BX.Move the value to AVG**

****

**Program Code:**

**ASSUME CS: CODE ,DS: DATA**

**DATA SEGMENT**

**ARR DB 001H, 023H, 045H, 067H, 042H, 04BH, 012H, 0EFH**

**N DW 08H**

**SUM DW 01 DUP (?)**

**AVG DW 01 DUP (?)**

**DATA ENDS**

**CODE SEGMENT**

**START:**

**MOV AX, @DATA**

**MOV DS, AX**

**MOV CX, N**

**MOV AX, 0000H**

**MOV SI, OFFSET ARR**

**ABC:**

**MOV BL, [SI]**

**INC SI**

**ADD AX, BX**

**DEC CX**

**JNZ ABC**

**MOV SUM, AX**

**MOV CX, AX**

**MOV BX, N**

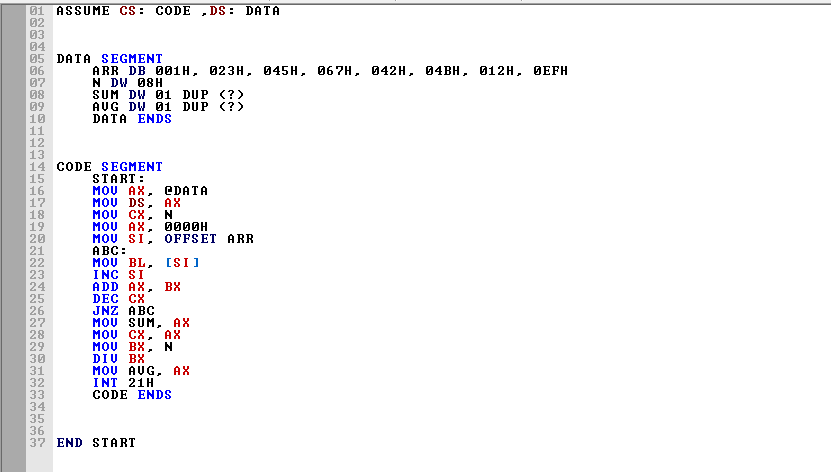
**DIV BX**

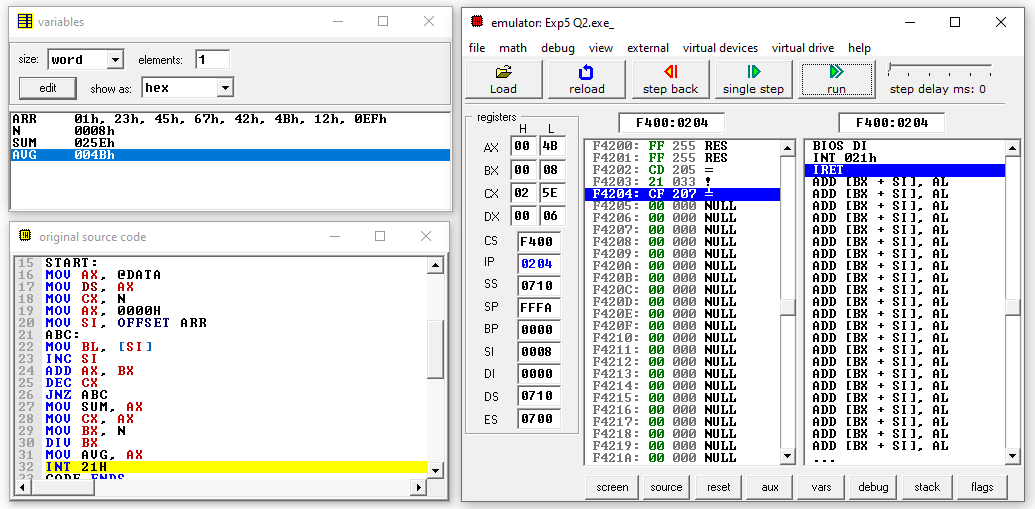
**MOV AVG, AX**

**INT 21H**

**CODE ENDS**

**END START**

****

**Output:** 

**Result and Inference:**

**-The sum of the Values is stored in CX, and Average is Stored in Accumulator. In the Variables We can see the value of Sum(25EH) and Average(4BH)**

**3) Write an ALP to find the greatest among two numbers.**

**ALGORITHM**

**FLOWCHART**

**Design and Calculations:**

**Program Code:**

**Output:**

**Result and Inference:**

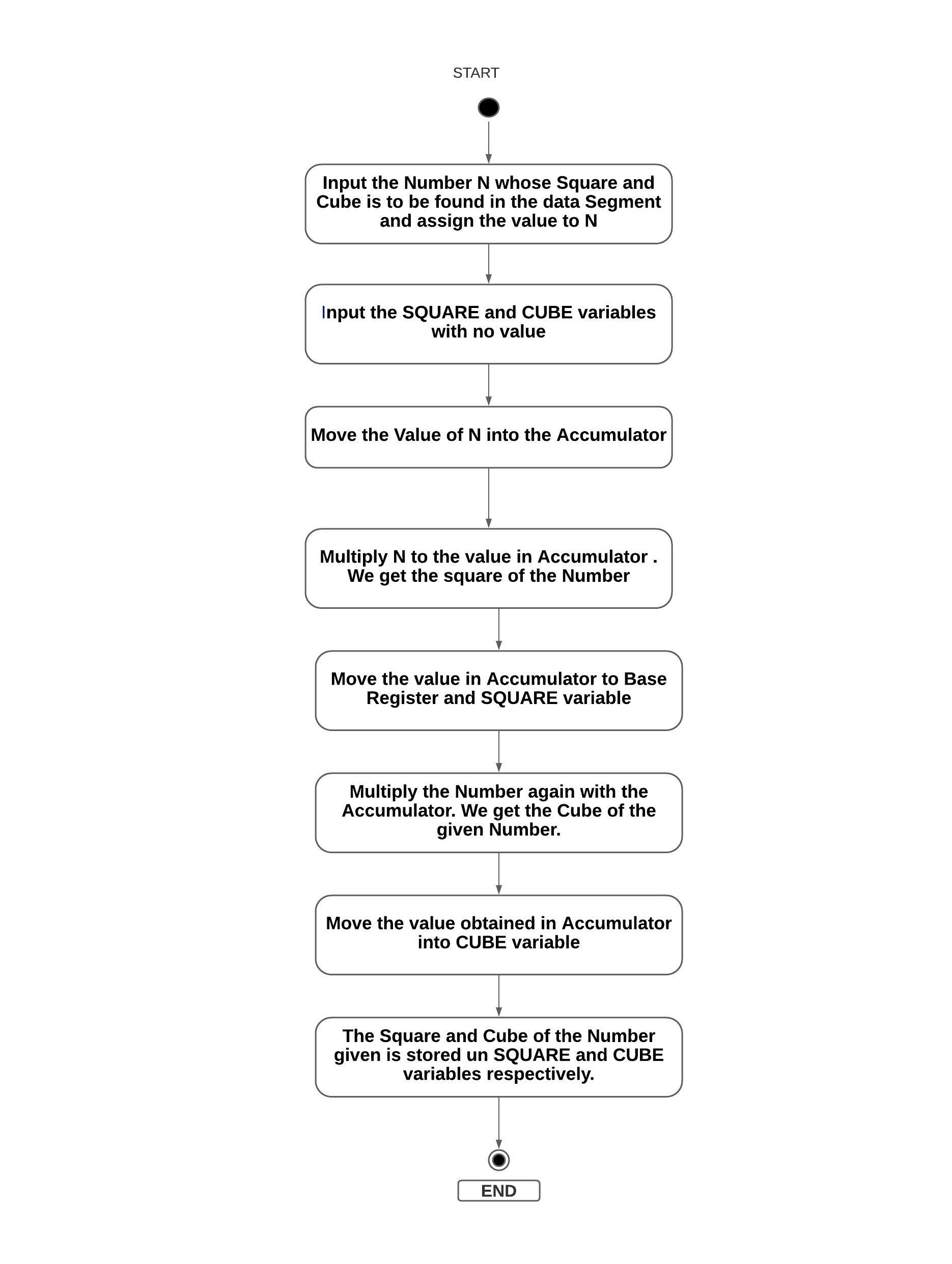
**Experiment 6:**

**Question 1) Write an ALP program to find square and cube of a given number**

**ALGORITHM**

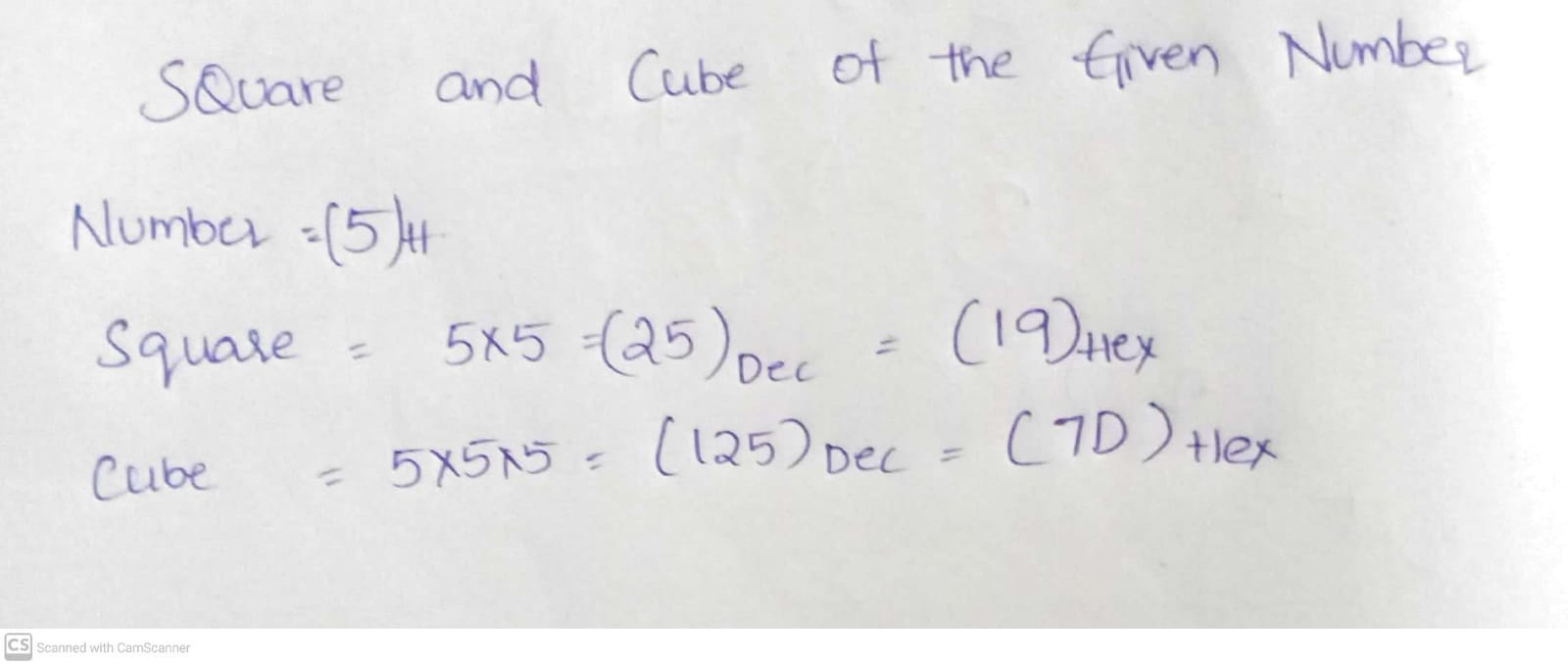
* **Input the Number N whose Square and Cube is to be found in the data Segment and assign the value to N**
* **Input the SQUARE and CUBE variables with no value**
* **Move the Value of N into the Accumulator**
* **Multiply N to the value in Accumulator . We get the square of the Number**
* **Move the value in Accumulator to Base Register and SQUARE variable**
* **Multiply the Number again with the Accumulator. We get the Cube of the given Number.**
* **Move the value obtained in Accumulator into CUBE variable**
* **The Square and Cube of the Number given is stored in SQUARE and CUBE variables respectively.**

**FLOWCHART**

****

**Design and Calculations:**

**Input the Number N whose Square and Cube is to be found in the data Segment and assign the value to N.Input the SQUARE and CUBE variables with no value.Move the Value of N into the Accumulator.Multiply N to the value in Accumulator . We get the square of the Number.Move the value in Accumulator to Base Register and SQUARE variable.Multiply the Number again with the Accumulator. We get the Cube of the given Number.Move the value obtained in Accumulator into CUBE variable.The Square and Cube of the Number given is stored in SQUARE and CUBE variables respectively.**

****

**Program Code:**

**DATA SEGMENT**

**A DW 5H**

**SQUARE DW ?**

**CUBE DW ?**

**DATA ENDS**

**CODE SEGMENT**

**ASSUME DS:DATA,CS:CODE**

**START:**

**MOV AX,DATA**

**MOV DS,AX**

**MOV AX,A**

**MUL A**

**MOV SQUARE, AX**

**MOV BX, AX**

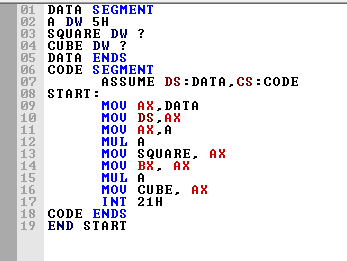
**MUL A**

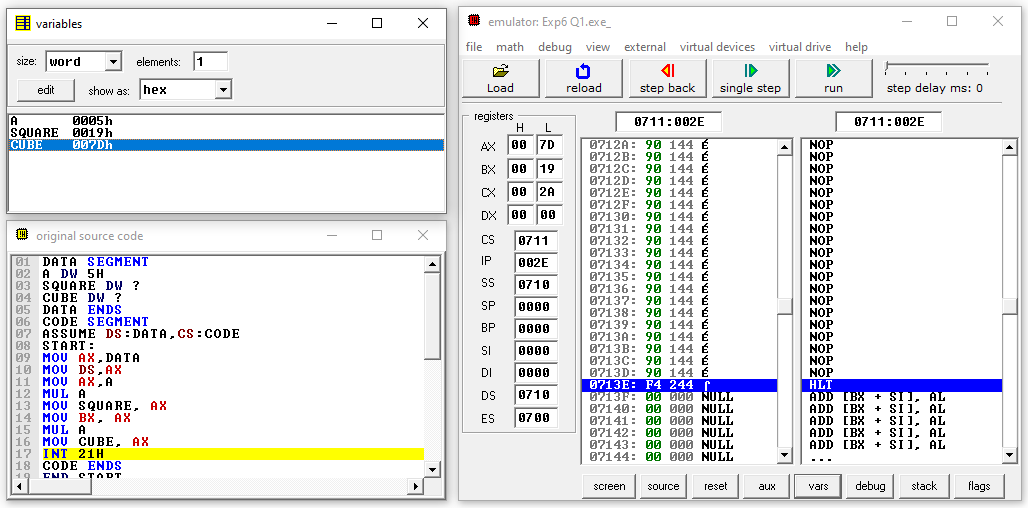
**MOV CUBE, AX**

**INT 21H**

**CODE ENDS**

**END START**

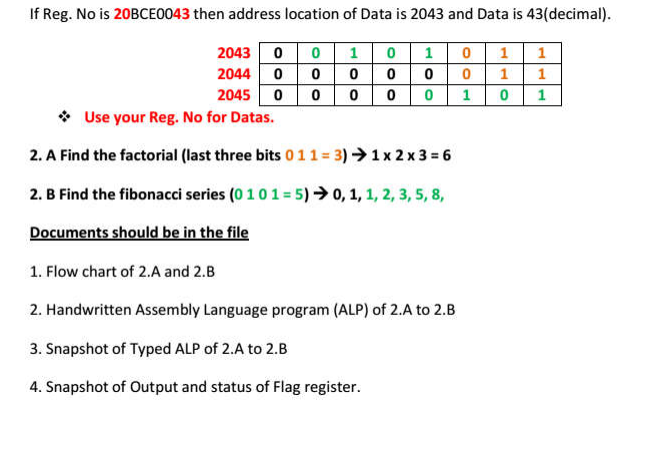
****

**Output: **

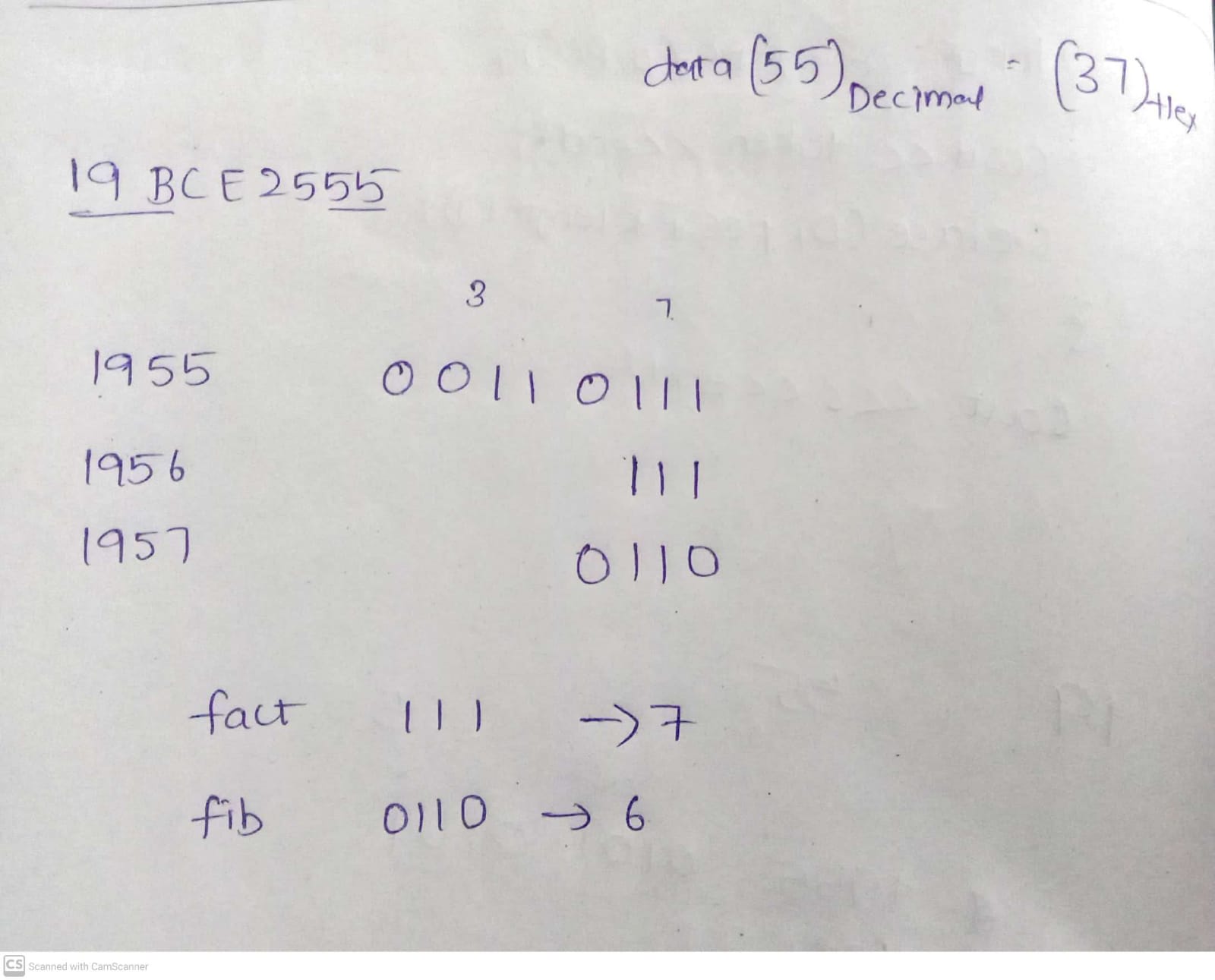
**Result and Inference:**

**The value in the Accumulator is the Cube of the number given and Square of the number is stored in Counter Register. We can see that SQUARE(0019H) and CUBE (007DH) variables are filled with values of square andd cube of the number(0005H) given**

**Question 2)**

****

**INITIAL CACLUTIONS WITH REGISTRATION NUMBER:**

****

**Factorial Input: 7**

**Fibanocci Input: 6**

**ALGORITHM**

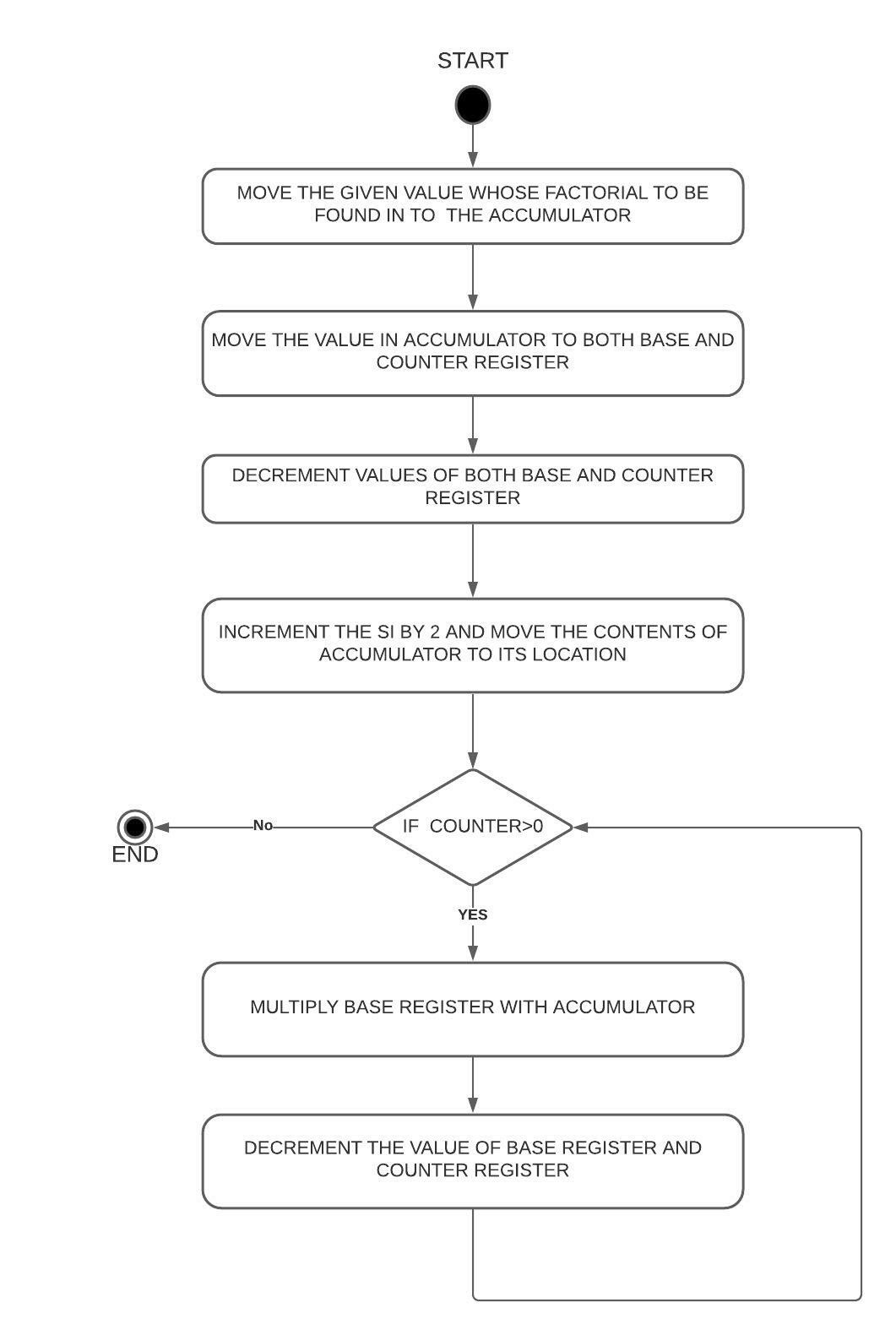
**Factorial:**

* **Move the value of whose factorial to be found in to the Accumulator.**
* **Move the value in Accumulator to both Base Register and Counter Register.**
* **Decrement both Base Register and Counter Register**
* **Multiply the Base Register to Accumulator and decrement the Base Register inside a loop**
* **Loop is repeated and Base register is multiplied to Accumulator till the counter becomes 0.**
* **The Factorial of the Given value is stored in Accumulator.**

**Fibanocci:**

**FLOWCHART**

**Factorial:**

****

**Fibanocci:**

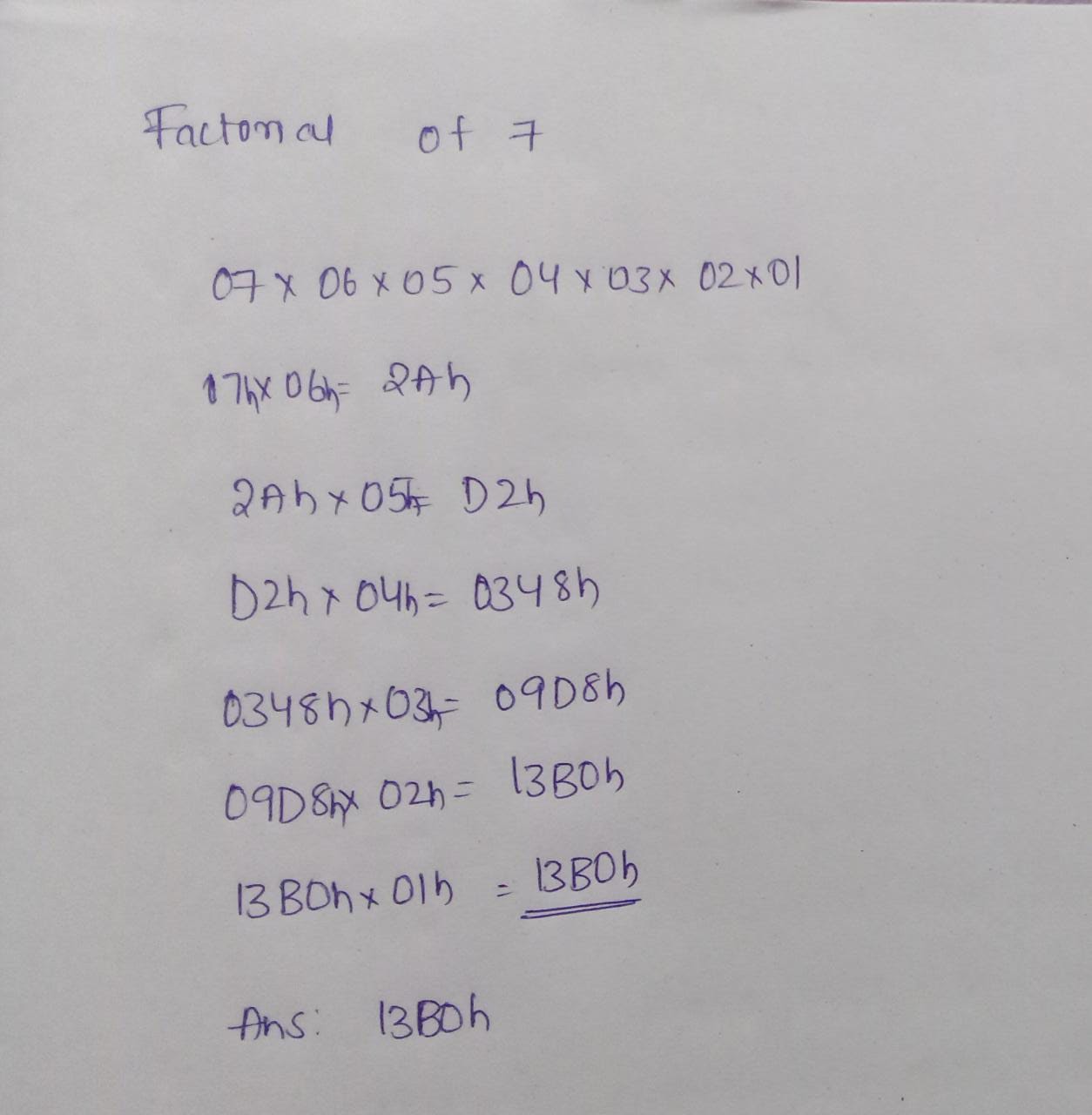
**Factorial**

**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])**

****

**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**

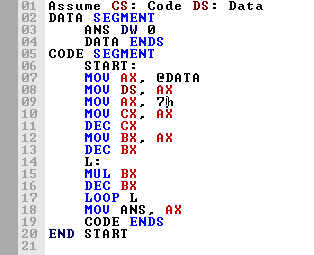
**DEC BX**

**LOOP L**

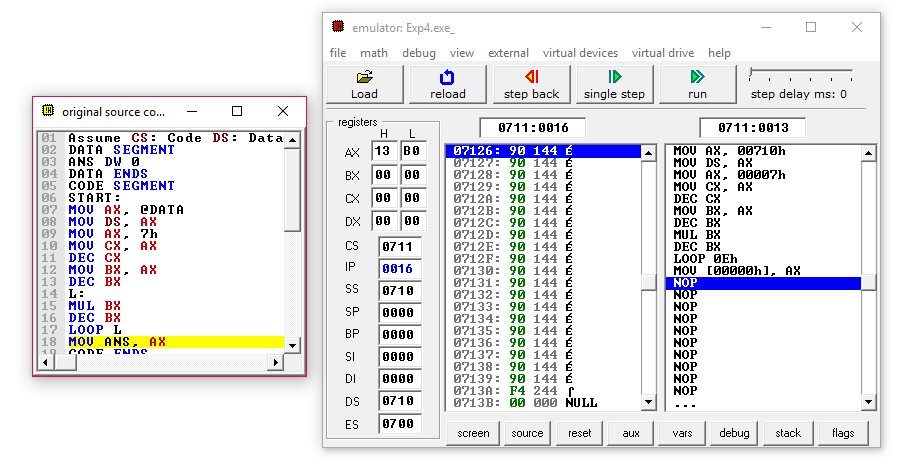
**MOV ANS, AX**

**CODE ENDS**

**END START**

****

**OUTPUT:**

****

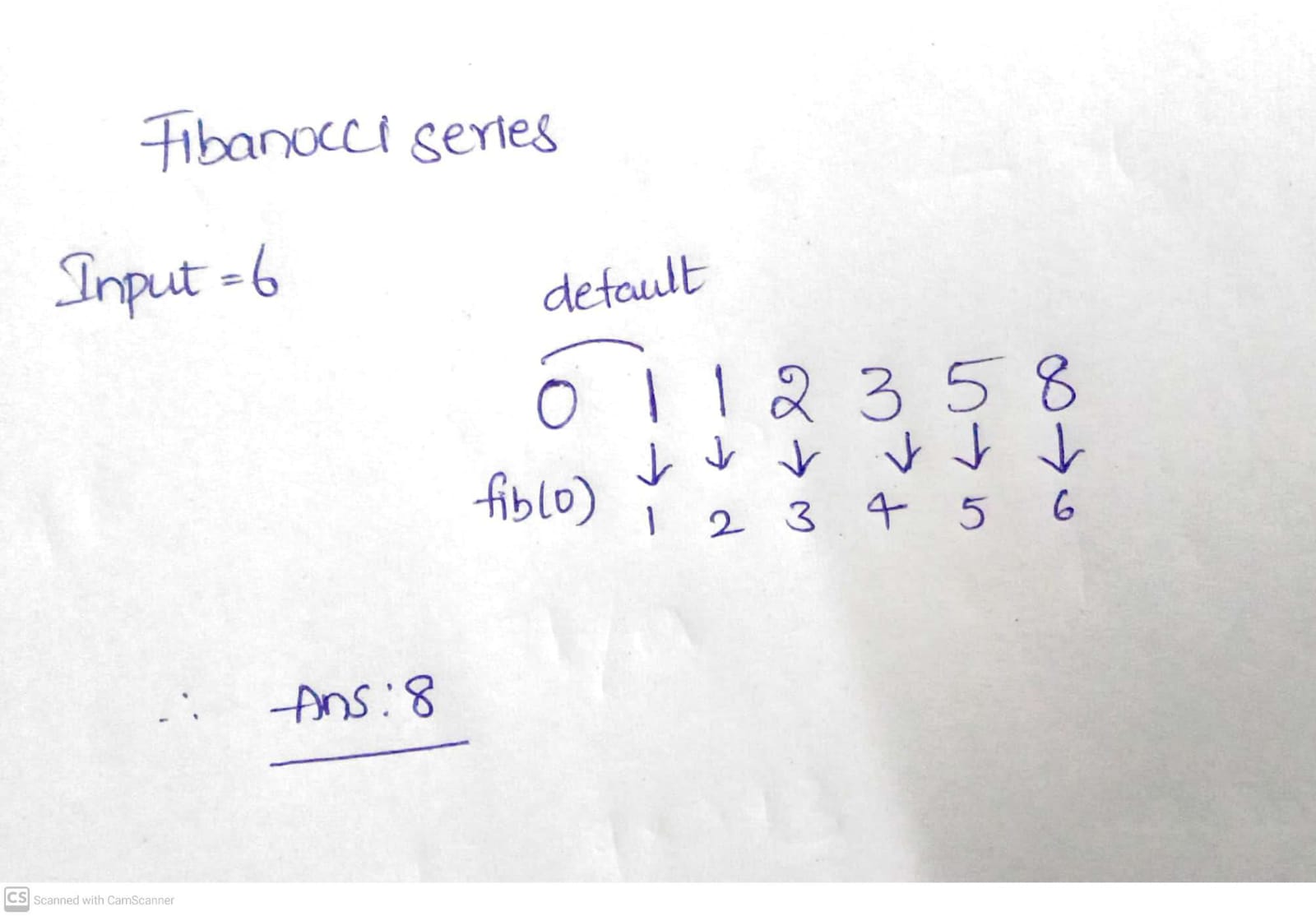
**Result and Inference:**

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

**-Both Base and Counter Registers becomes 0.**

**Fibanocci:**

**Design and Calculations:**

****

**Program Code:**

**Output:**

**Result and Inference:**