Fall Semester 2021-2022

Microprocessor and Interfacing

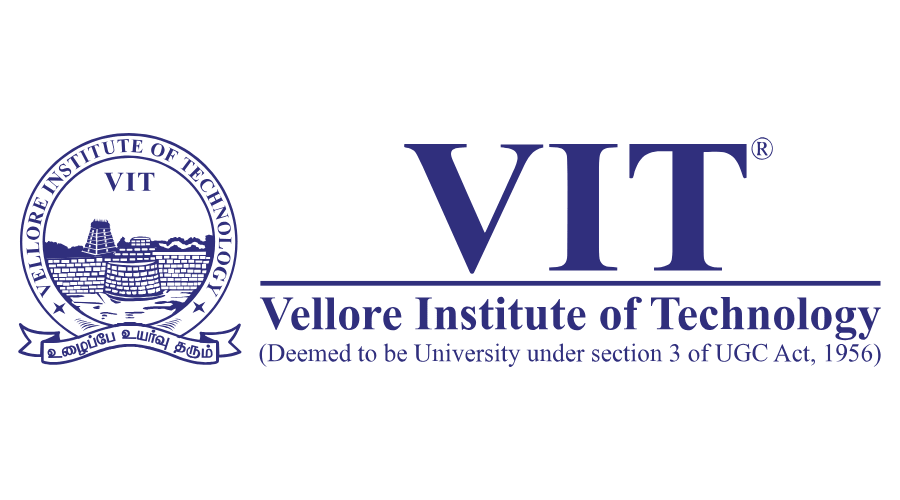
Lab Report

Digital Assignment-5

Experiment No: 6,7 Task No: 5

Course Code: CSE2006

Slot: L7+L8



Submitted By: Alokam Nikhitha

Reg. Numb: 19BCE2555

Submitted To: Dr. Abdul Majed KK

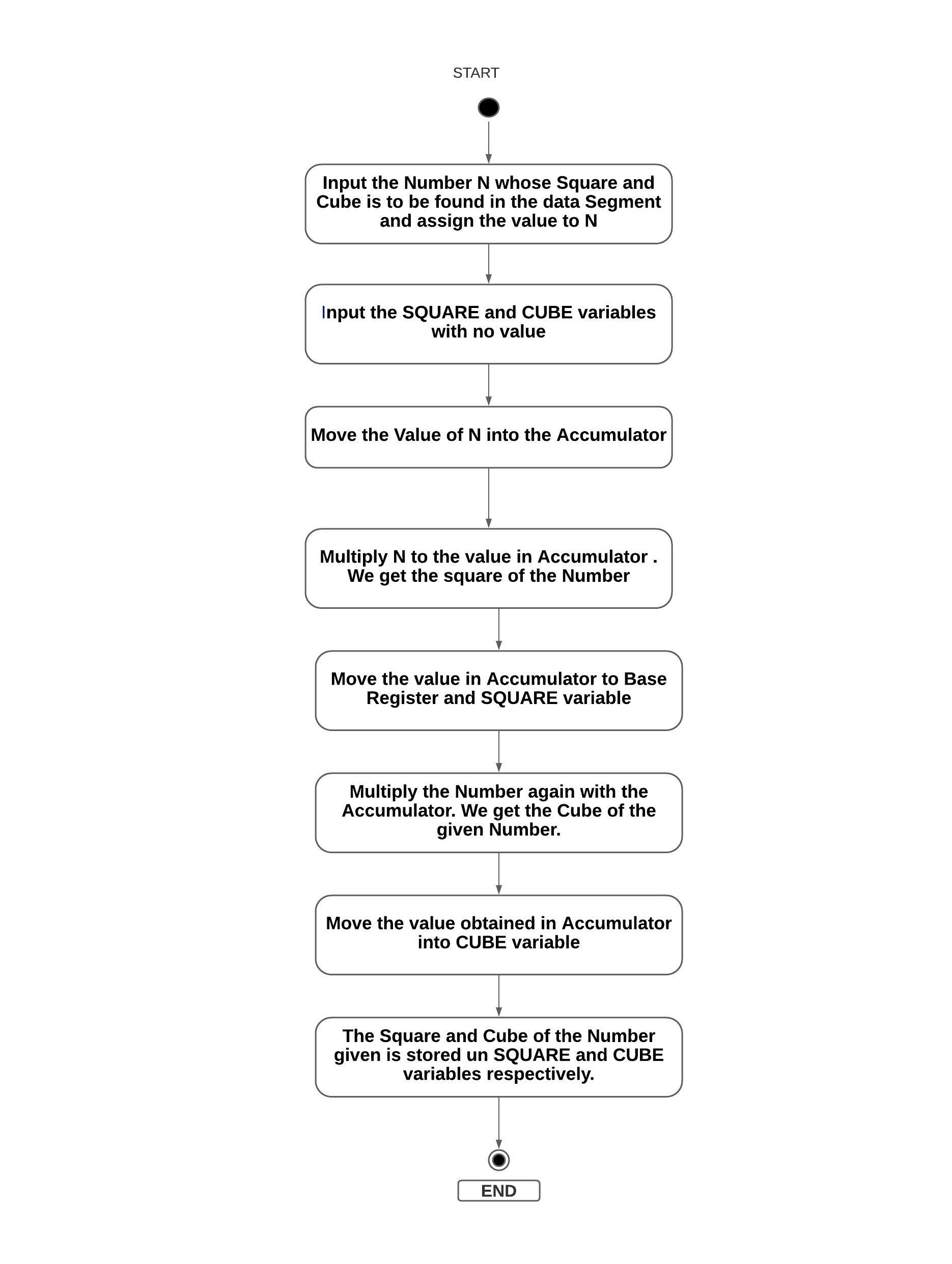
**Experiment 6:**

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

**ALGORITHM**

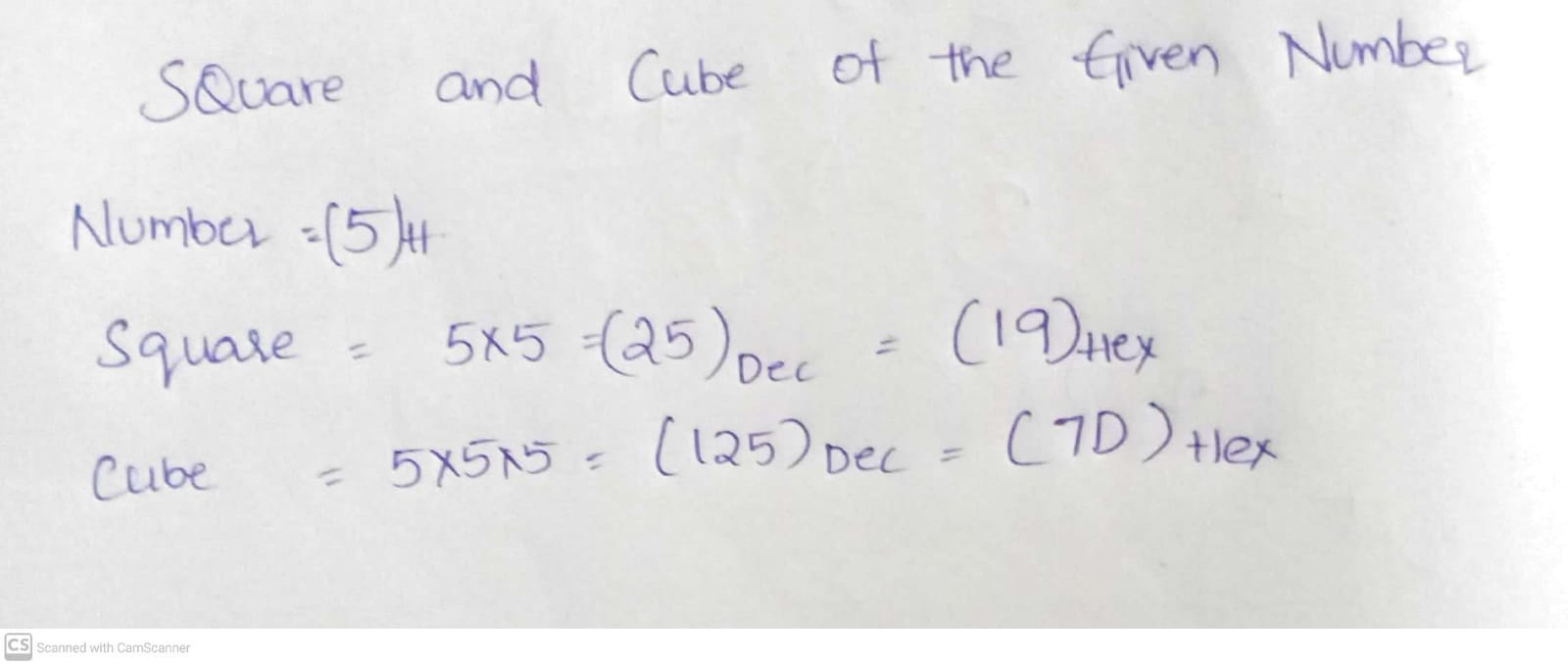
* **Input the Number N whose Square and Cube is to be found in the data Segment and assign the value to N**
* **Input the SQUARE and CUBE variables with no value**
* **Move the Value of N into the Accumulator**
* **Multiply N to the value in Accumulator . We get the square of the Number**
* **Move the 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**

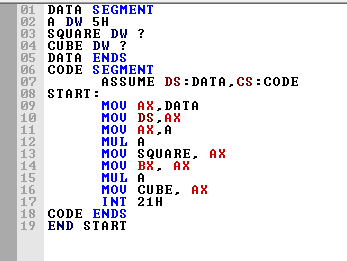
**MUL A**

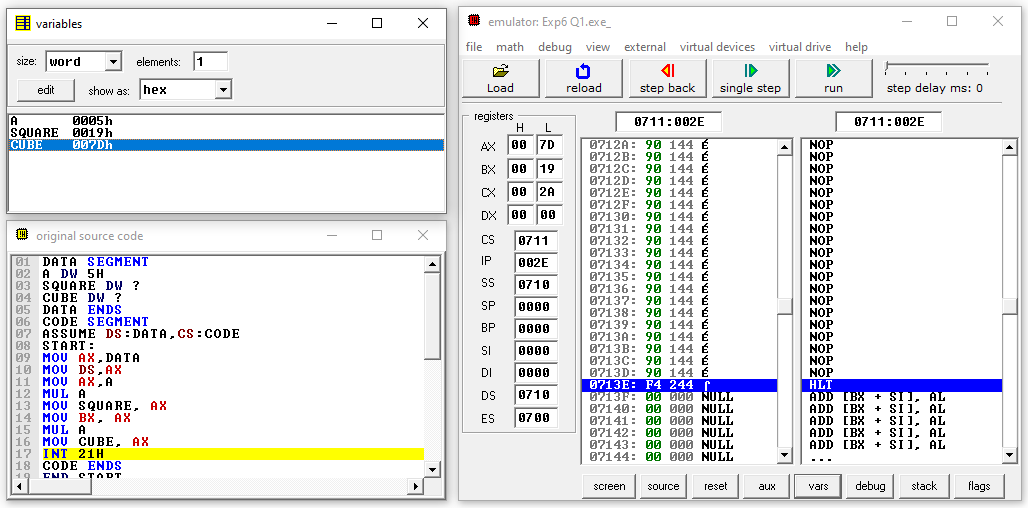
**MOV CUBE, AX**

**INT 21H**

**CODE ENDS**

**END START**

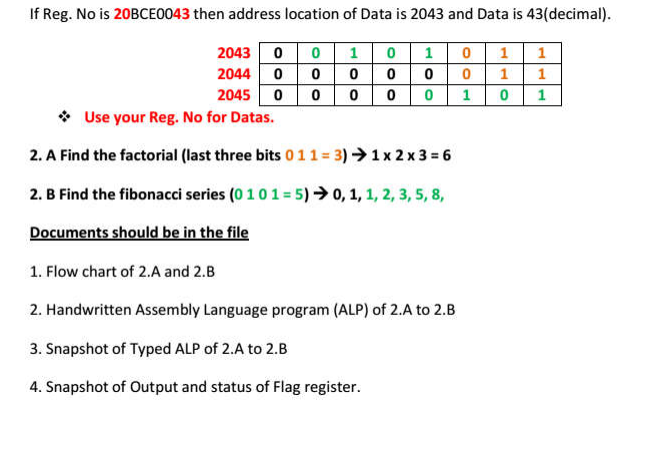
****

**Output: **

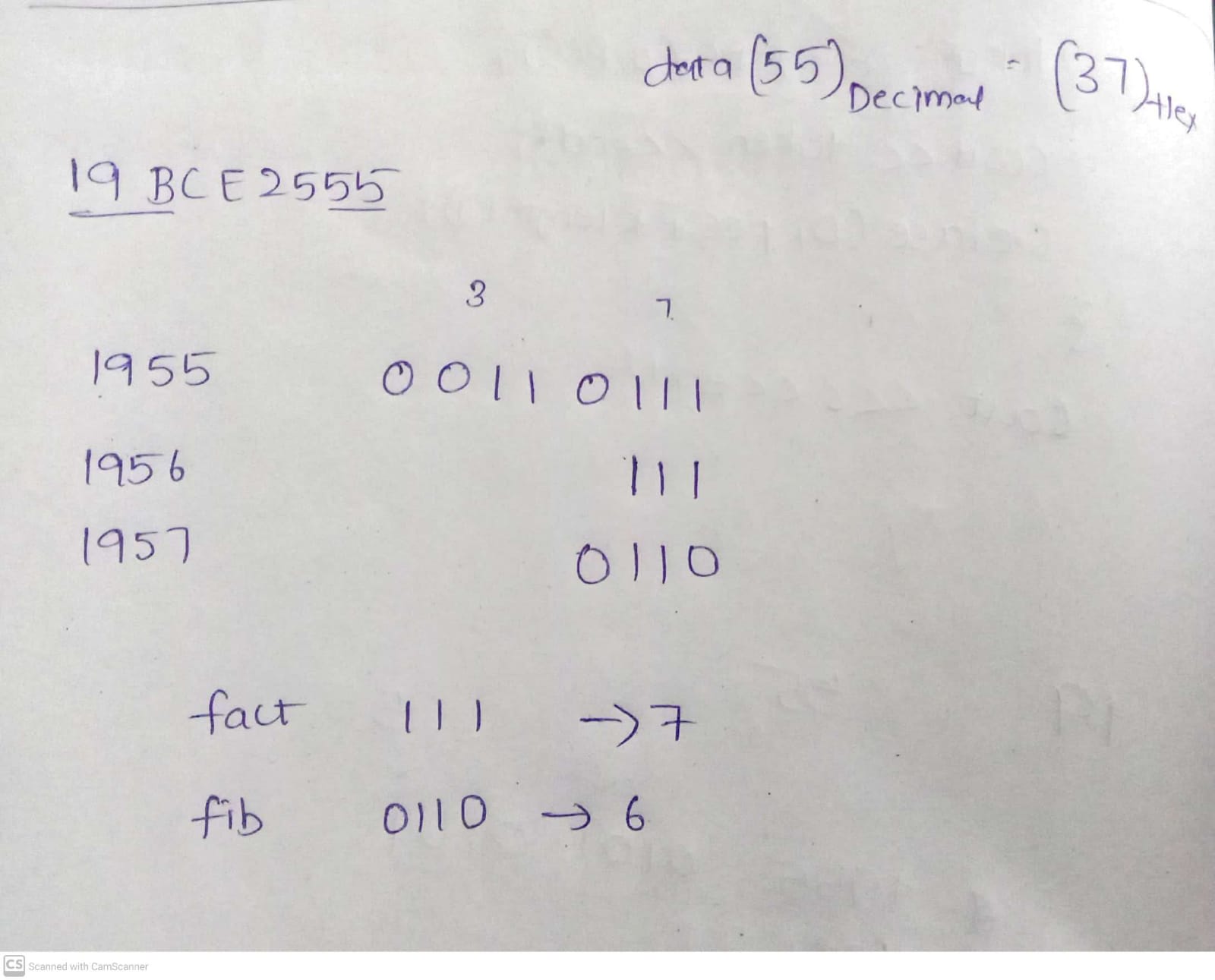
**Result and Inference:**

**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:**

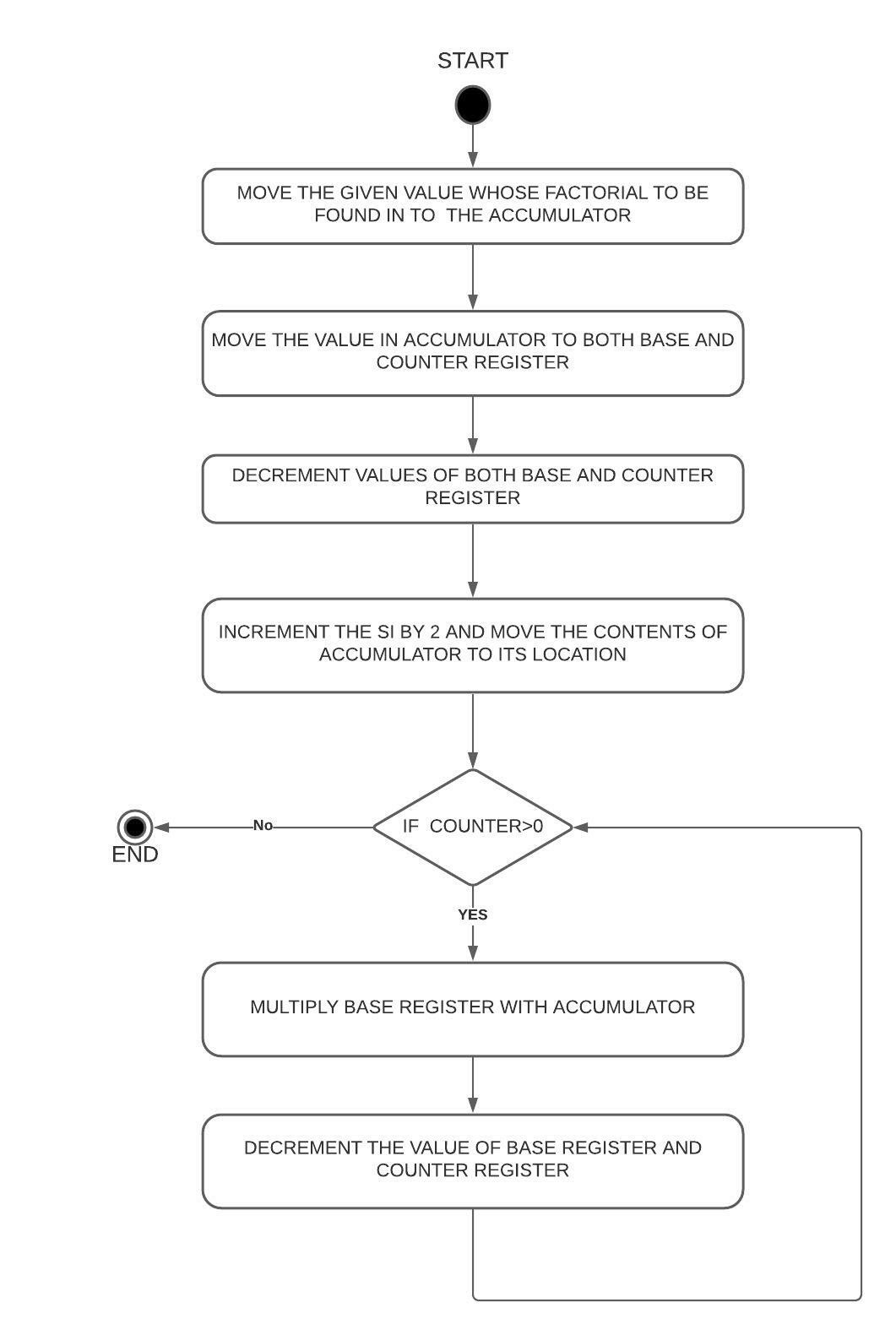
* **Move the value of whose factorial to be found in to the Accumulator form the given Location.**
* **Move the value in Accumulator to both Base Register and Counter Register.**
* **Decrement both Base Register and Counter Register**
* **Multiply the Base Register to Accumulator and decrement the Base 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:**

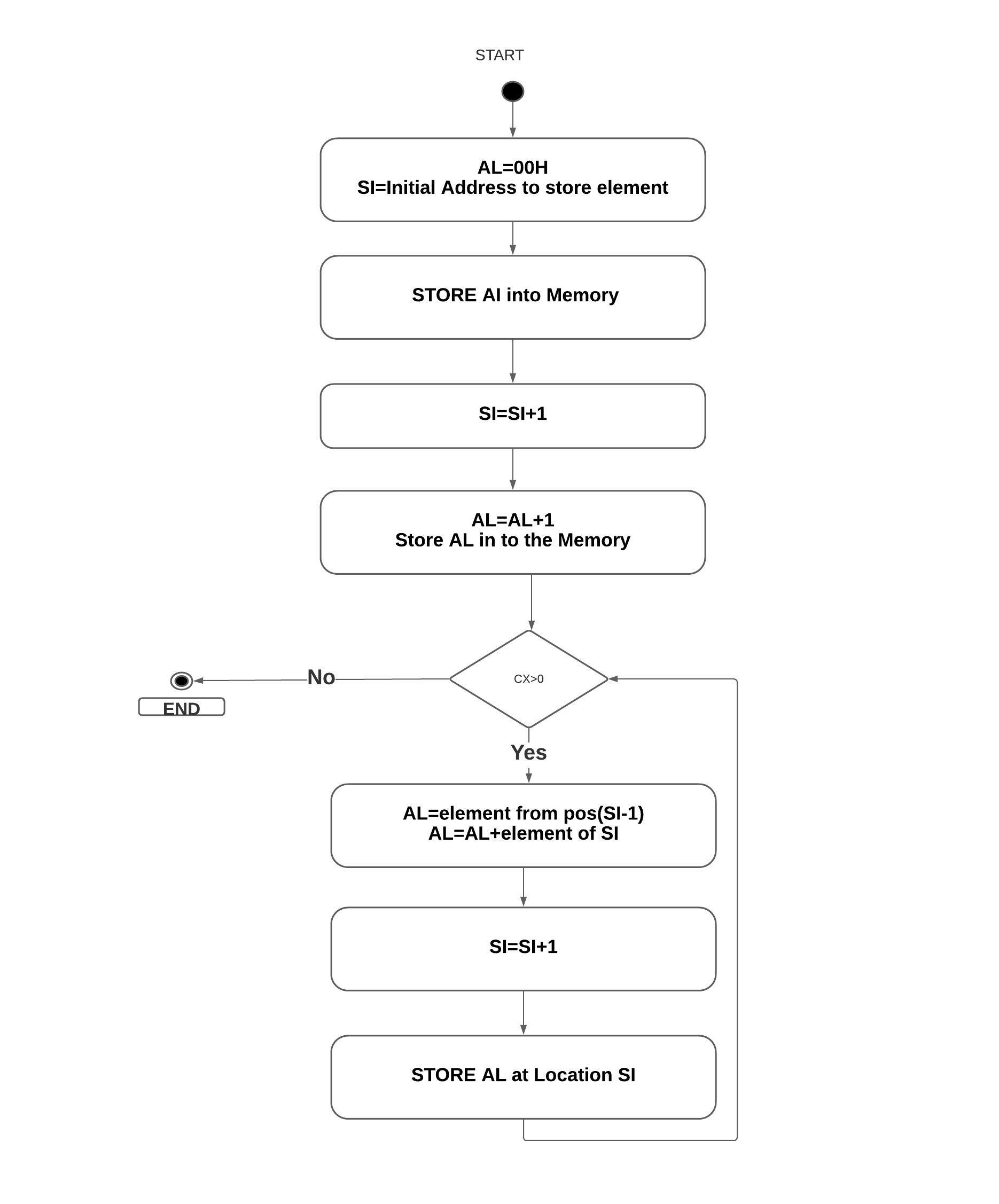
* **Input the value of the whose Fibanocci series to be found from the given Location.**
* **Move 500h to SI and input 00h to that location([SI])**
* **Increment the SI value and input 01h to it**
* **Move the N value into counter**
* **Run a loop if counter >0**
* **Move the value at SI-1 location to Accumulator**
* **Add the Value at Location SI to the Accumulator**
* **Increment the SI value**
* **Move the value in Accumulator to location of SI**
* **Decrement the counter**
* **Repeat Loop**

**FLOWCHART**

**Factorial:**

****

**Fibanocci:**

****

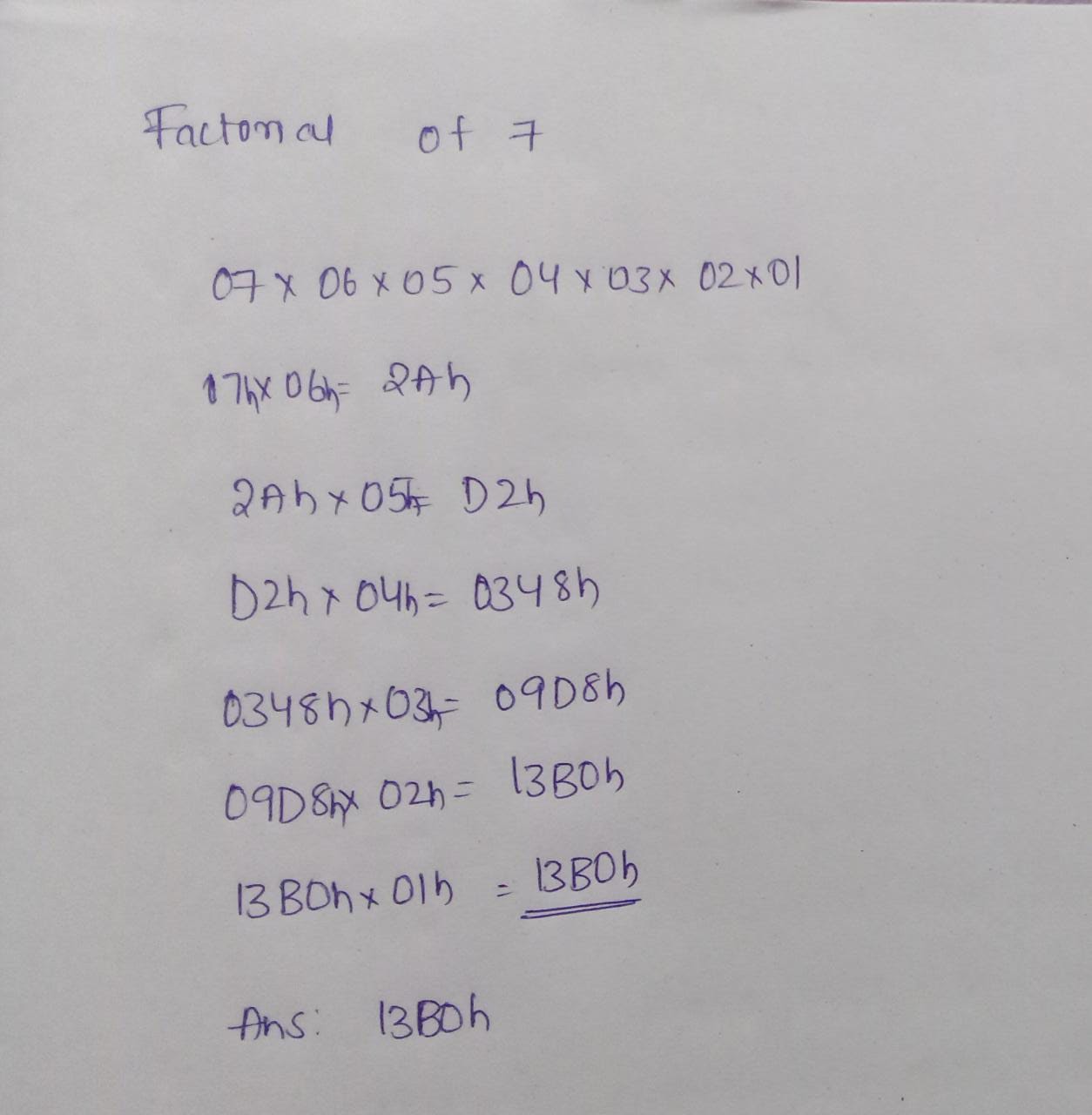
**Factorial**

**Design and Calculations:**

**Here we input the value from location 1956h 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**

**N DW 1956h**

**Fact DW ?**

**DATA ENDS**

**CODE SEGMENT**

**START:**

**MOV AX, @DATA**

**MOV DS, AX**

**MOV SI, N**

**MOV AX, [SI]**

**MOV CX, AX**

**DEC CX**

**MOV BX, AX**

**DEC BX**

**L:**

**MUL BX**

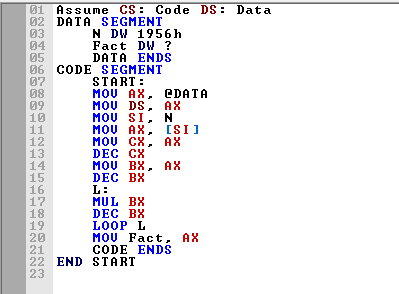
**DEC BX**

**LOOP L**

**MOV Fact, AX**

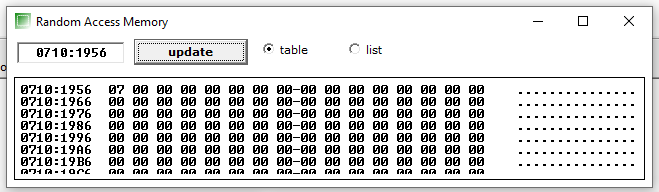
**CODE ENDS**

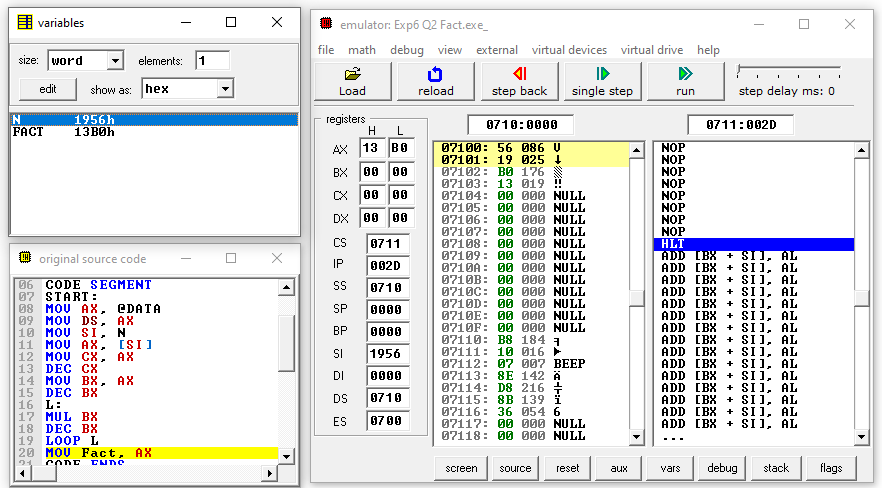
**END START**

****

**OUTPUT:**

**Memory Location:**

****

****

**Result and Inference:**

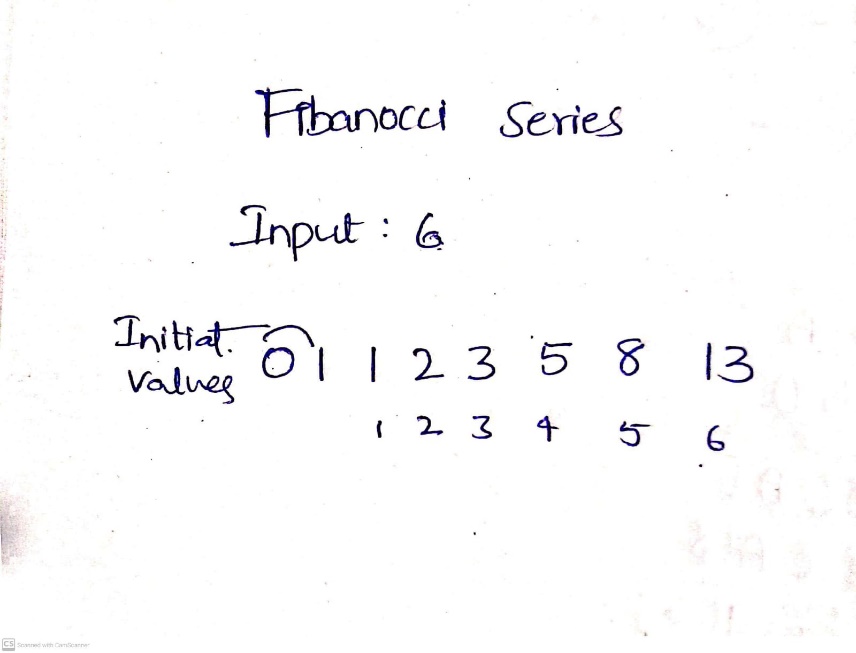
**-The Factorial of 7 (5040[dec]-> 13B0[hex]) 13B0 is stored in Accumulator and in Fact Variable.**

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

**Fibanocci:**

**Design and Calculations:**

**We can observe that DS=0710h from fig1.Then we collected data from Memory location 1957h.To generate Fibonacci sequence, we are putting the 00H and 01H into memory at first. Then we are taking the limit from location offset 500. We input the N value in to the counter and.Now we are taking number from previous location, then add it with the value of current location, after that storing the result into next location, till the counter becomes 0**

****

**Program Code:**

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

**DATA SEGMENT**

**N DW 1957h**

**FIB DB ?**

**DATA ENDS**

**CODE SEGMENT**

**START:**

**MOV AX,DATA**

**MOV DS,AX**

**MOV AX, N**

**MOV SI, AX**

**MOV CX, [SI]**

**MOV AL, 00H**

**MOV SI, 500H**

**MOV [SI], AL**

**INC SI**

**ADD AL, 01H**

**MOV [SI], AL**

**L1: MOV AL, [SI-1]**

**ADD AL, [SI]**

**INC SI**

**MOV [SI], AL**

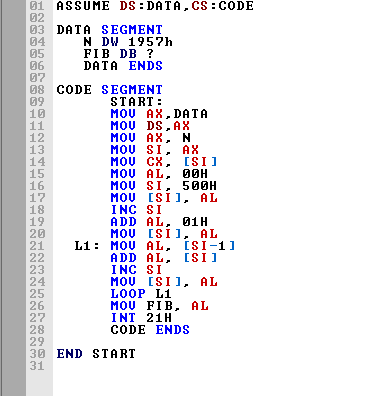
**LOOP L1**

**MOV FIB, AL**

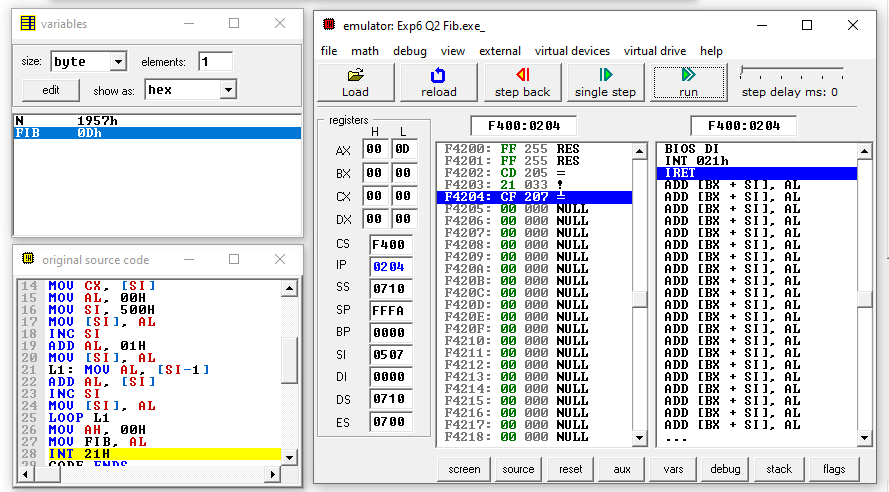
**INT 21H**

**CODE ENDS**

**END START**

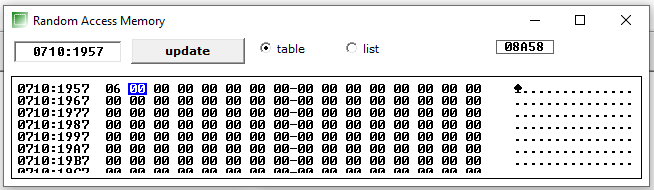
****

**Output:**

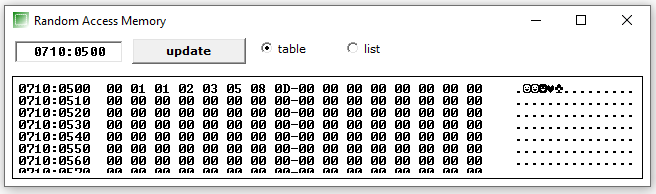
****

**Memory Location:**

**BEFORE (INPUT):**

****

**AFTER:**

****

**Result and Inference:**

**We can observe that DS=0710h.**

**Then we collected data from Memory location 1957h**

**We have the list of Fibanocci series for N=6 we have DS=0710 and they are located from 0710:0500 to 0710:0507(00,01,01,02,03,05,08,0D)**

**Experiment 7:**

**Question 1) Convert Binary number corresponding to 0109H to BCD number.**

**ALGORITHM**

**Step I              :**Initialize the data segment.

**Step II             :**Initialize BX = 0000 H and DH = 00H.

**Step III           :**    Load the number in AX.

**Step IV           :**Compare number with 10000 decimal. If below goto step VII else goto step V.

**Step V             :**Subtract 10,000 decimal from AX and add 1 decimal to DH

**Step VI           :**Jump to step IV.

**Step VII          :**Compare number in AX with 1000, if below goto step X else goto step VIII.

**Step VIII        :**Subtract 1000 decimal from AX and add 1000 decimal to BX.

**Step IX           :**Jump to step VII.

**Step X             :**Compare the number in AX with 100 decimal if below goto step XIII

**Step XI           :**Subtract 100 decimal from AX and add 100 decimal to BX.

**Step XII         :**Jump to step X

**Step XIII        :**Compare number in AX with 10. If below goto step XVI

**Step XIV        :**Subtract 10 decimal from AX and add 10 decimal to BX..

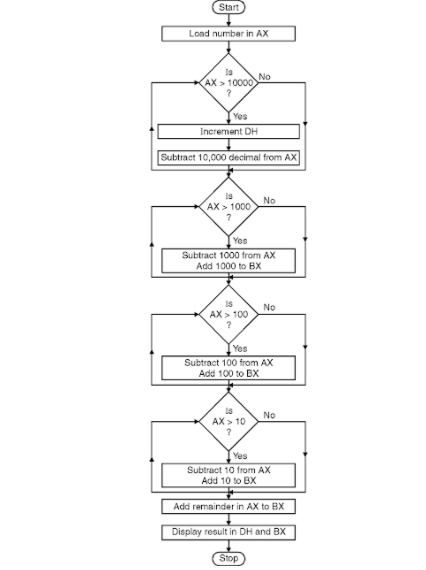
**Step XV          :**Jump to step XIII.

**Step XVI        :**Add remainder in AX with result in BX.

**Step XVII      :**Display the result in DH and BX.

**Step XVIII     :**Stop.

**Flow Chart:**

****

**Design and Calculations:**

**Program Code:**

**Output:**

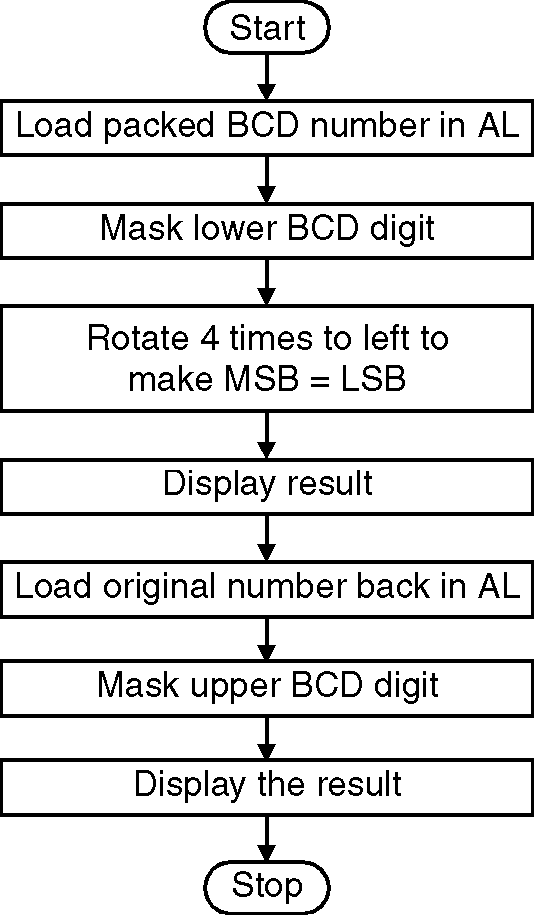
**Result and Inference:**

**Question 2) Convert BCD number corresponding to 27H to Binary number.**

**ALGORITHM**

* Assign value 27H to BCD in Datset
* Move BCD into AH(Accumulator)
* Move the contents of [SI] in BL.
* Use AND instruction to calculate AND between 0F and contents of BL.
* Move the contents of [SI] in AL.
* Use AND instruction to calculate AND between F0 and contents of AL.
* Move 04 in CL.
* Use ROR instruction on AL.
* Move 0A in DL.
* Use MUL instruction to multiply AL with DL.
* Use ADD instruction to add AL with BL.
* Move the contents of AL in [DI].
* Halt the program.

**Flow Chart:**



**Design and Calculations:**

Assign value 27H to BCD in Datset.Move the contents of [SI] in BL.Use AND instruction to calculate AND between 0F and contents of BL.Move the contents of [SI] in AL.Use AND instruction to calculate AND between F0 and contents of AL.Move 04 in CL.Use ROR instruction on AL.Move 0A in DL.Use MUL instruction to multiply AL with DL.Use ADD instruction to add AL with BL.Move the contents of AL in [DI].Halt the program.

**Program Code:**

**DATA\_SEG SEGMENT**

**BCD DB 27H ; STORAGE FOR A BCD VALUE**

**BIN DB ? ; STORAGE FOR BINARY VALUE**

**DATA\_SEG ENDS**

**CODE\_SEG SEGMENT**

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

**START:**

**MOV AX,DATA\_SEG**

**MOV DS,AX**

**MOV AH,BCD**

**MOV BH,AH**

**AND BH,0FH**

**AND AH,0F0H**

**ROR AH,04**

**MOV CL,10**

**MOV AL,AH**

**AND AX,00FFH**

**MUL CL**

**ADD AL,BH**

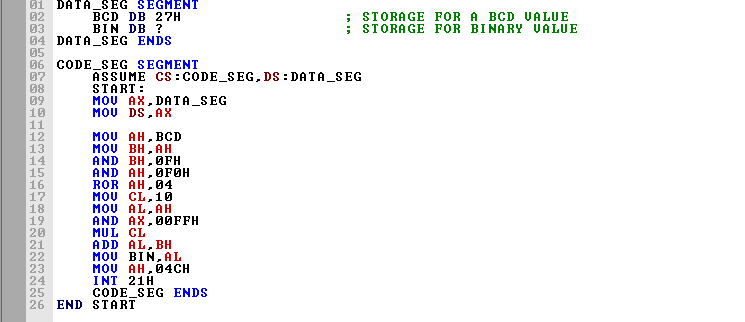
**MOV BIN,AL**

**MOV AH,04CH**

**INT 21H**

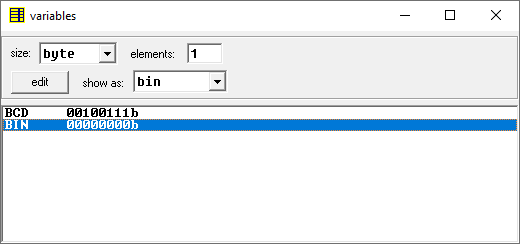
**CODE\_SEG ENDS**

**END START**

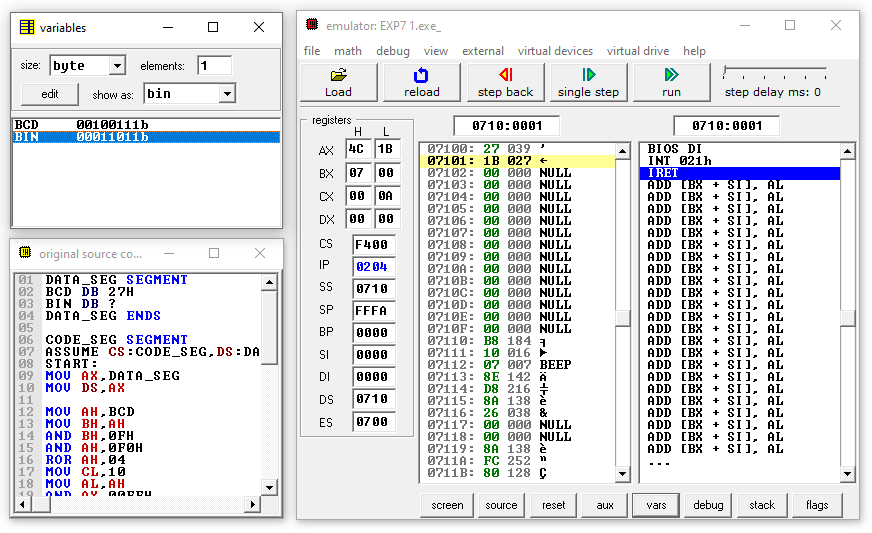
****

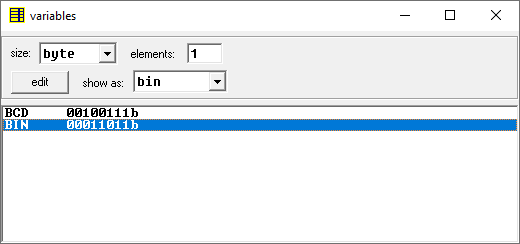
**Output:**

**Before Execution:**

****

**After Execution:**

****

****

**Result and Inference:**

**On execution we got the Binary of the given BCD(00100111b) 27H as 00011011b which is 1BH and the answer is stored in BIN variable.**