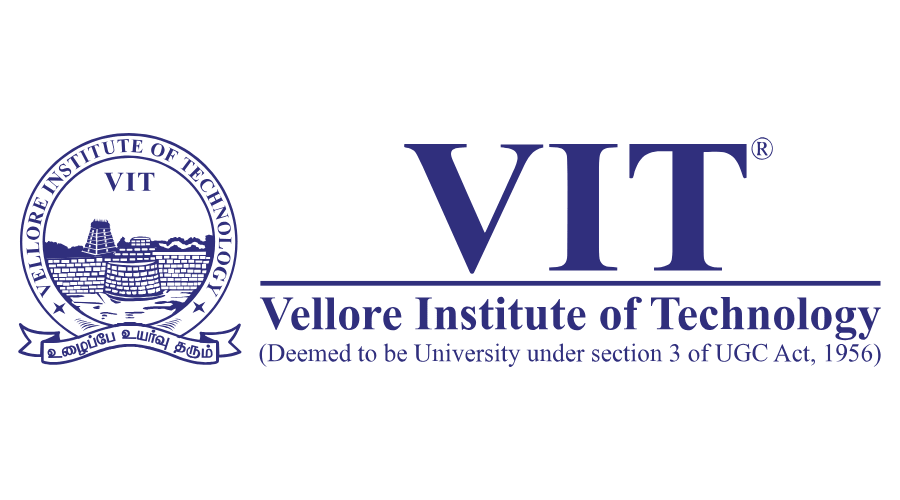
Fall Semester 2021-2022

Microprocessor and Interfacing

LAB FAT

Course Code: CSE2006

Slot: L7+L8

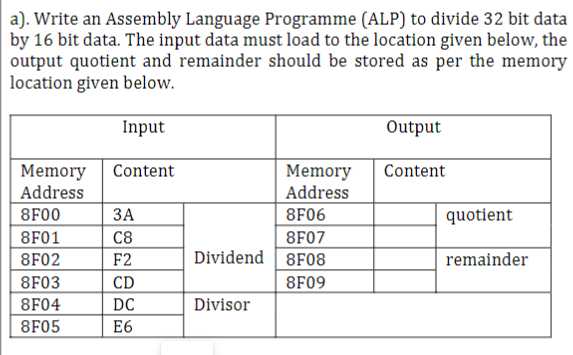


Submitted By: Alokam Nikhitha

Reg. Numb: 19BCE2555

Submitted To: Dr. Abdul Majed KK

**SNO . 36**



Aim

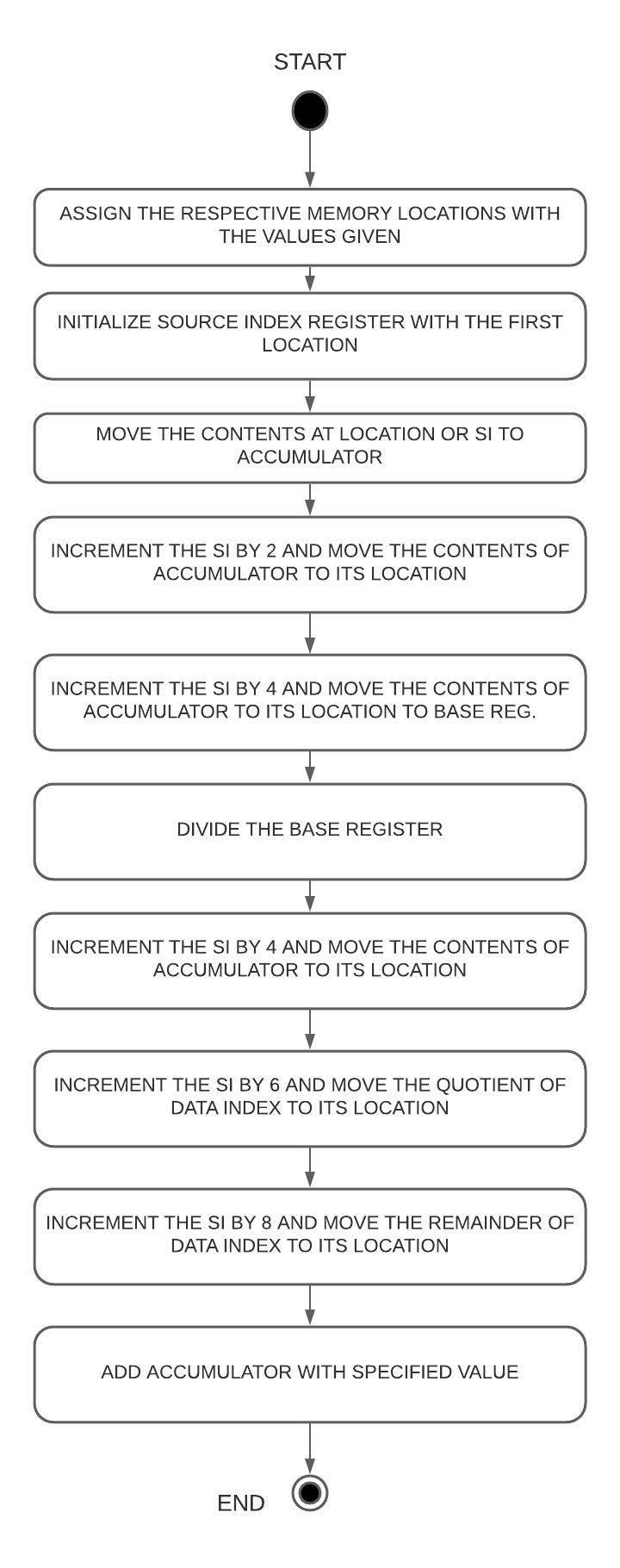
To write an Assembly code for Divison of 32 bit number with 16 bit number and store vaklues in given memory location

Algorithm:

1. Move all the values in the specified memory locations.
2. Move the starting memory to SI register for reference.
3. Move the contents at location SI to accumulator register (AX).
4. Increment the SI value by 2 in order to point to the next memory location.
5. Increment the SI value by 4 and move the next contents to base register (BX).
6. Divide the base register. This will store the quotient in AX.
7. Move the contents of accumulator register to specified memory location

8)The remainder is stored in Data register and we move it to the specified location

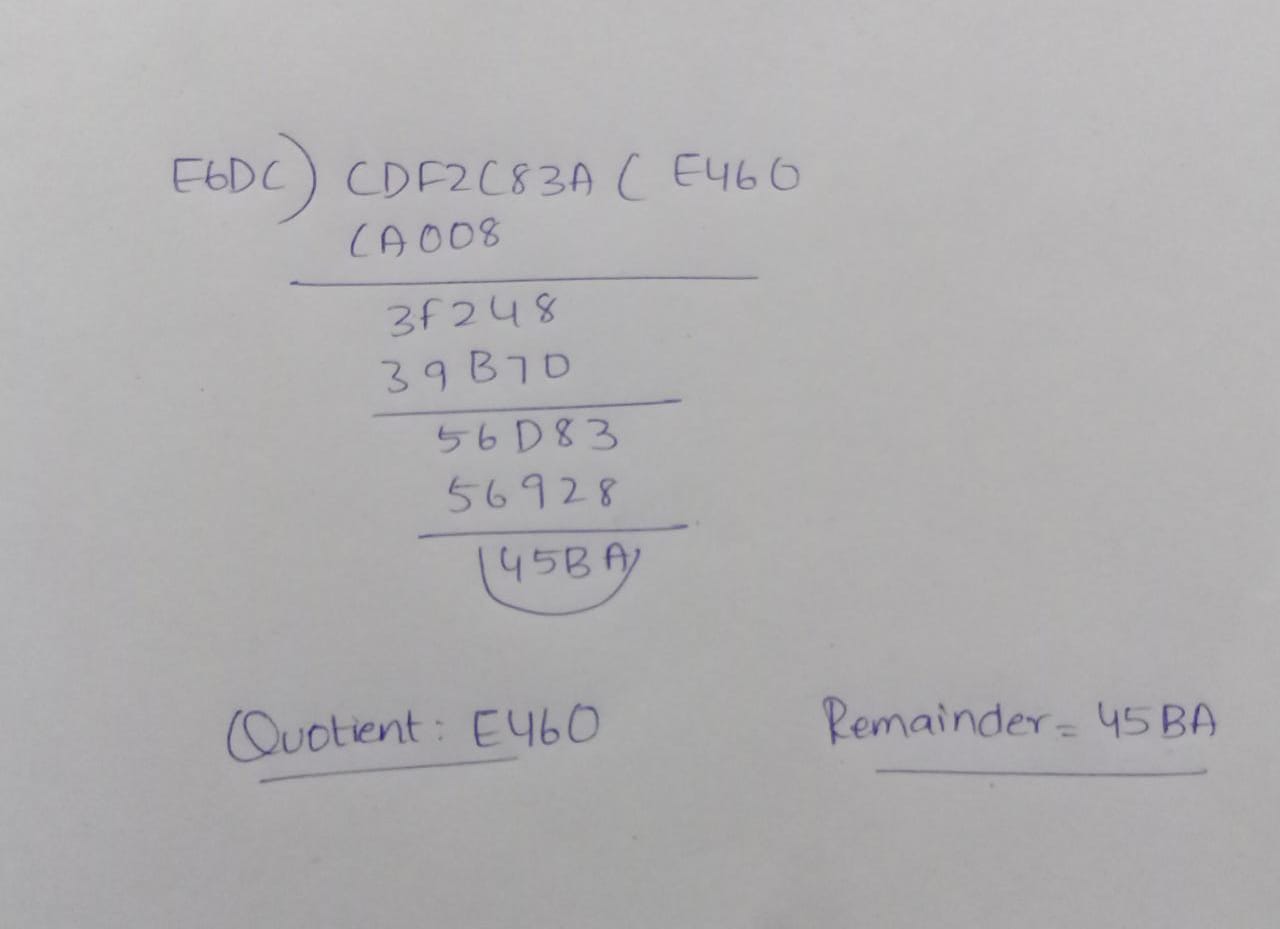
Flow Chart:



Design and Calculations:

Here we are going to need Source index register, accumulator, base register and the data register. The source index register is used as reference for the location to point at and stores the memory location of 8F00H. We then store the values mentioned to the specific locations of 8F00H, 8F01H, 8F02H, 8F03H, 8F04H, 8F05H. We then move the data to accumulator

We then divide the base register which stores the quotient in accumulator and the remainder in data register. Hence, we move the contents of accumulator and data register to the specified location of 8F06H and 8F08H.



Program Code:

DATA\_SEG SEGMENT

DIVIDEND1 DW 0CDF2H

DIVIDEND2 DW 0C83AH

DIVISOR DW 0E6DCH

QUOTIENT DW ?

REMAINDER DW ?

DATA\_SEG ENDS

CODE\_SEG SEGMENT

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

START:

MOV AX,DATA\_SEG

MOV DS,AX

MOV [8F00H],3AH

MOV [8F01H],0C8H

MOV [8F02H],0F2H

MOV [8F03H],0CDH

MOV [8F04H],0DCH

MOV [8F05H],0E6H

MOV SI, 8F00H

MOV AX, [SI]

MOV DX, [SI+2]

MOV BX, [SI+4]

DIV BX

MOV QUOTIENT, AX

MOV [SI+6], AX

MUL BX

MOV CX,[SI]

SUB AX, CX

MOV [SI+7], AH

MOV [SI+8], AL

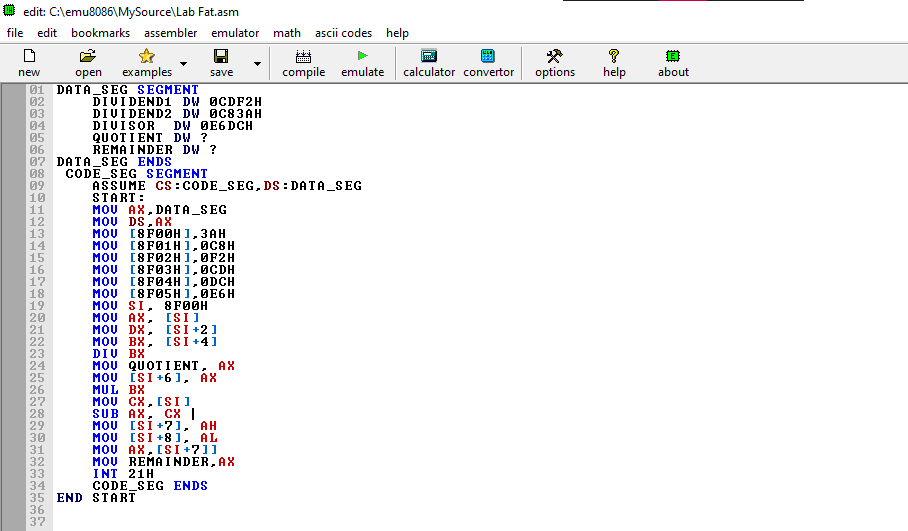
MOV AX,[SI+7]]

MOV REMAINDER,AX

INT 21H

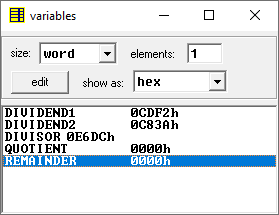
CODE\_SEG ENDS

END START

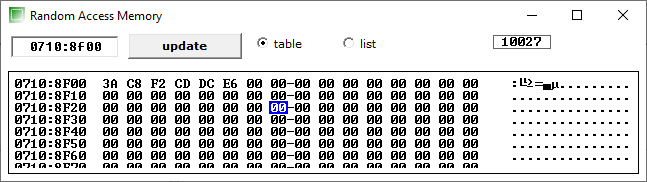


Before Emulation

**Variables:**

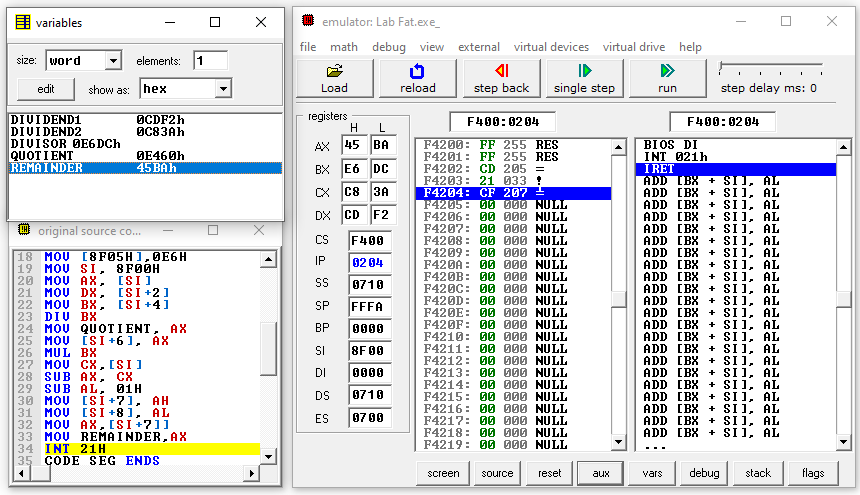
****

**Memory before Emulation**

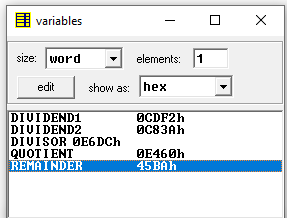


**OUTPUT**

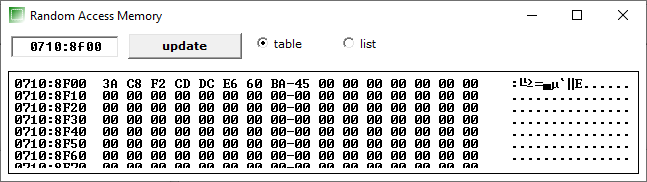
**After Emulation**



Variables after Emulation



Memory after emulation



Result and Inference:

* The accumulator initially had CDF2.
* The data register initially had C83A.
* The base register initially had E6DC
* The expected quotient of E460is stored in the memory location of 8F06H.
* The expected remainder of 45BAis stored in the memory location of 8F08H.
* Hence the quotient is E460 and remainder is 45BA as expected.

1 B)

