EXPERIMENT--01-ALP-FOR-8086

```
Name : Sana Fathima H
Register no :212223240145
Date of experiment :19-08-2025
```

Aim: To Write and execute ALP on fundamental arithmetic and logical operations

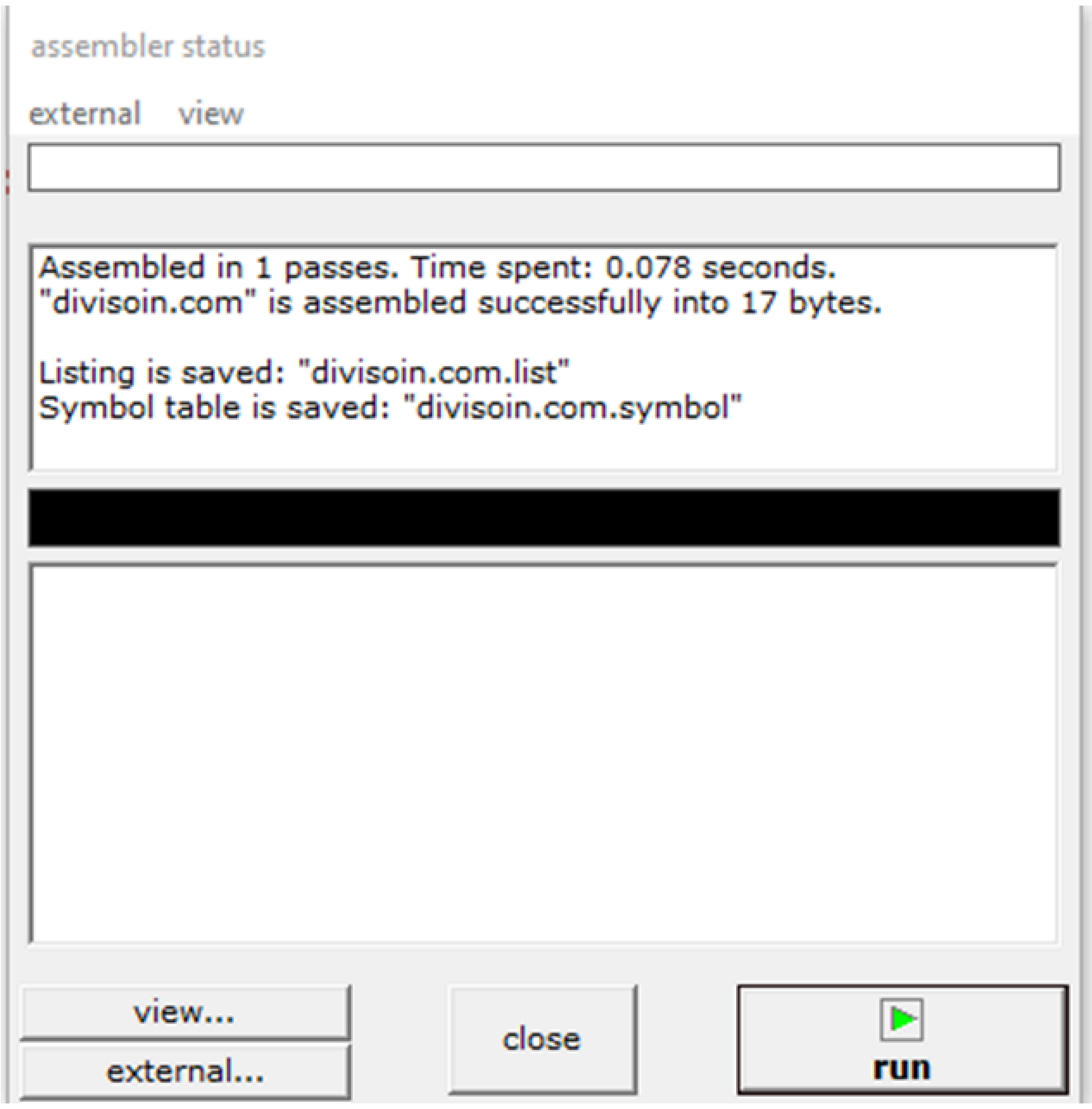
Components required: 8086 emulator

Theory

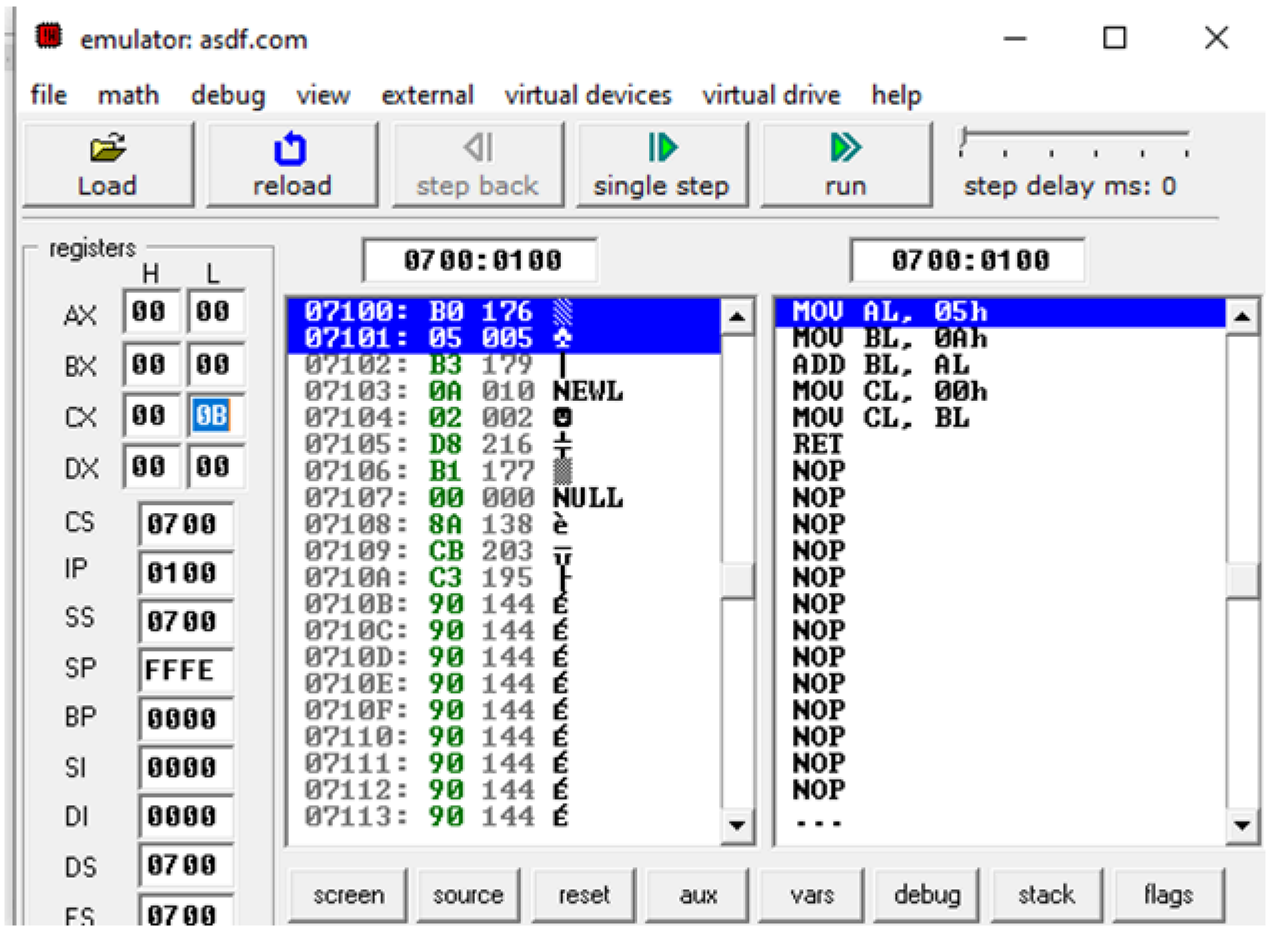
Running The Emulator (emu8086) Intro 8086 Microprocessor Emulator, also known as EMU8086, is an emulator of the program 8086 microprocessor. It is developed with a built-in 8086 assembler. This application is able to run programs on both PC desktops and laptops. This tool is primarily designed to copy or emulate hardware. These include the memory of a program, CPU, RAM, input and output devices, and even the display screen. There are instructions to follow when using this emulator. It can be executed into one of the two ways: backward or forward. There are also examples of assembly source code included. With this, it allows the programming of assembly language, reverse engineering, hardware architecture, and creating miniature operating system (OS). The user interface of 8086 Microprocessor Emulator is simple and easy to manage. There are five major buttons with icons and titles included. These are “Load”, “Reload”, “Step Back”, “Single Step”, and “Run”. Above those buttons is the menu that includes “File”, “View”, “Virtual Devices”, “Virtual Drive”, and “Help”. Below the buttons is a series of choices that are usually in numbers and codes. At the leftmost part is an area called “Registers” with an indication of either “H” or “L”. The other side is divided into two, which enables users to manually reset, debug, flag, etc. What is 8086 emulator emu8086 is an emulator of Intel 8086 (AMD compatible) microprocessor with integrated 8086 assembler and tutorials for beginners. Emulator runs programs like the real microprocessor in step-by-step mode. it shows registers, memory, stack, variables and flags.

Running the Emulator :

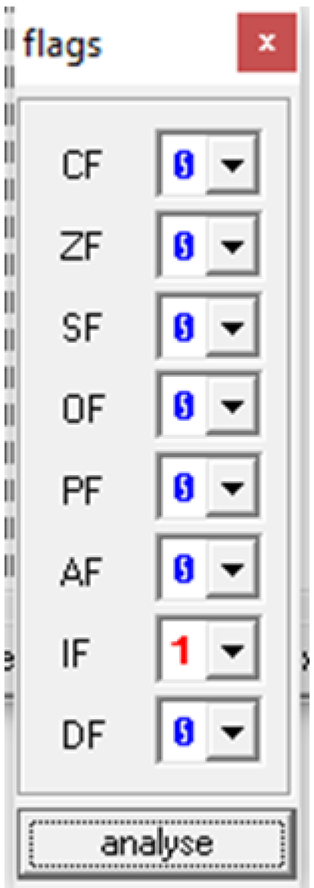
1. Download and install emu8086 (www.emu8086.com) It is usually installed in C:\EMU8086 subfolder in the “Windows” directory
2. Run emu8086 icon (on the desktop or in the c:\EMU8086 folder of window) It has green color
3. ```
   write the code for the appropriate program for ADDITION,SUBTRACTION, MULTIPLICATION,  DIVISION operations
   ```
4. Compile the program and check for the errors
5. Run (once there is no syntax error)
6. Click OK to see/view the output of your program on the Emulator screen.
7. After running the program, another menu screen will be displayed, where you have the option to “View” symbol table,
8.
   ```



9. Click on emulate to start emulation



10. If no errors are found click on run the program and check the status of various flags in the flags tab as shown below

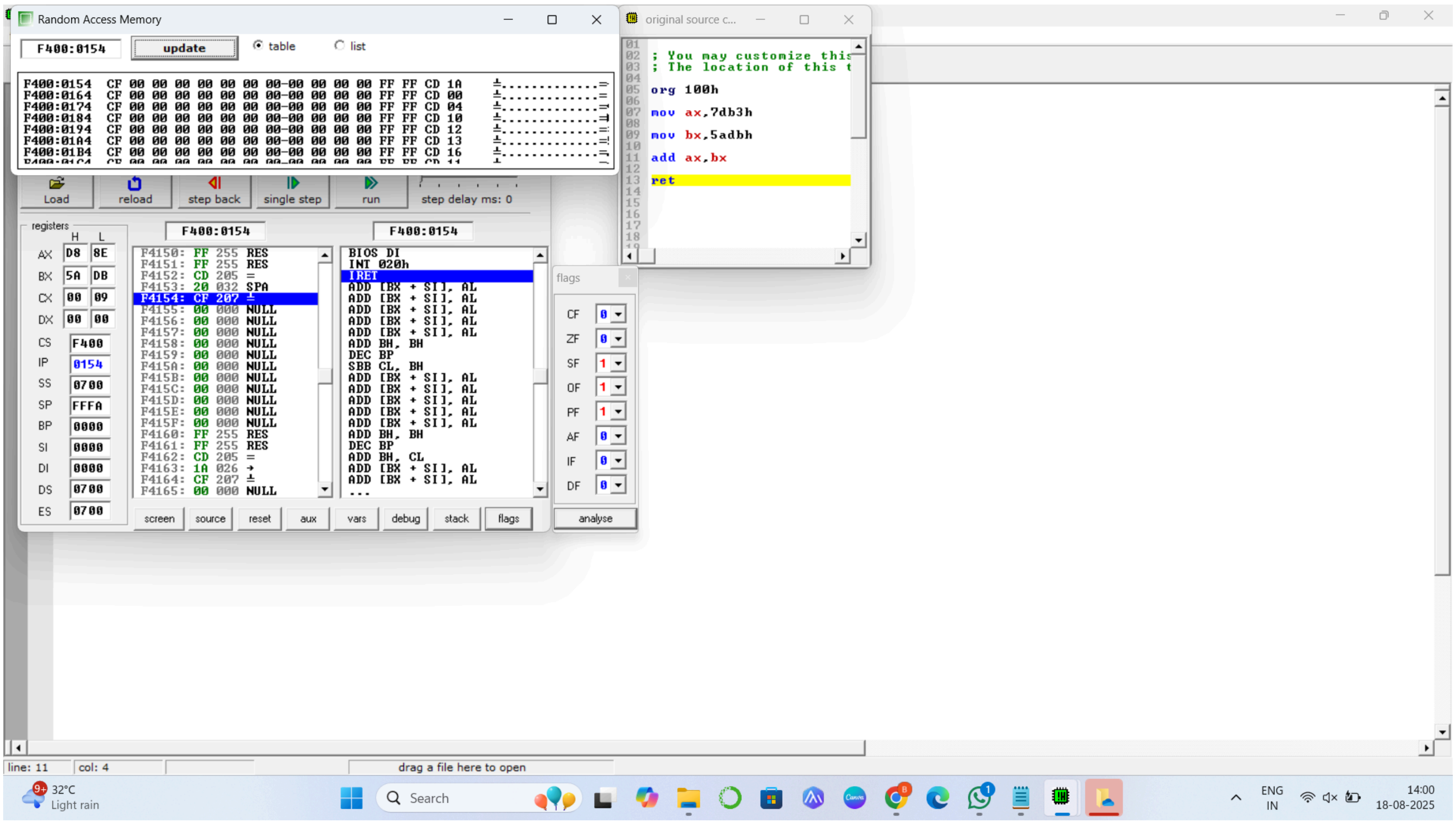


Programs for arithmetic operations

Addition of 8 bit ALP

```
org 100h
mov ax,7db3h
mov bx,5adbh
add ax,bx
ret
```

Output

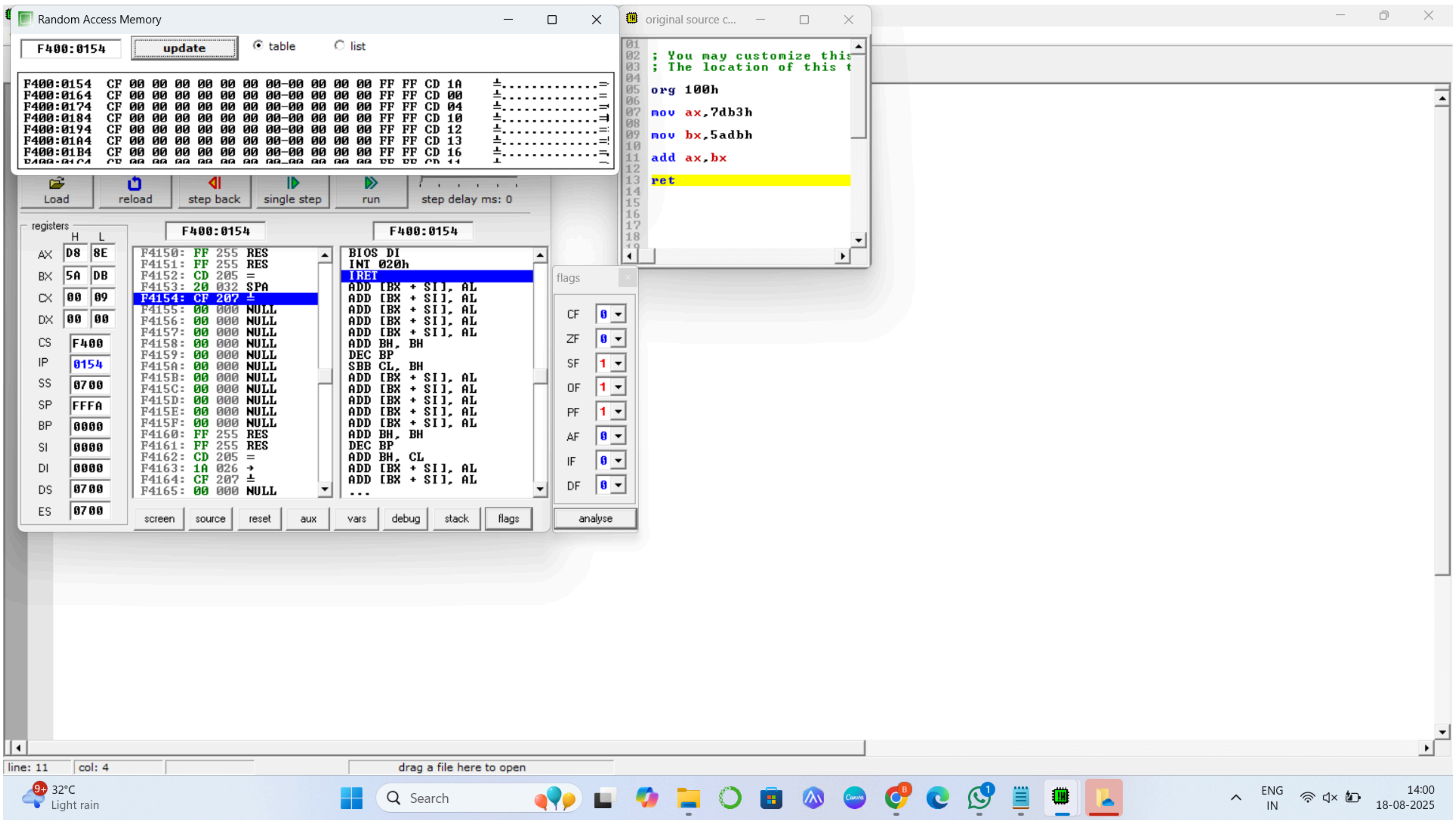


Subtraction of 8 bit numbers ALP

```
org 100h
mov ax,7db3h
```

```
mov bx,5adbh
sub ax,bx
ret
```

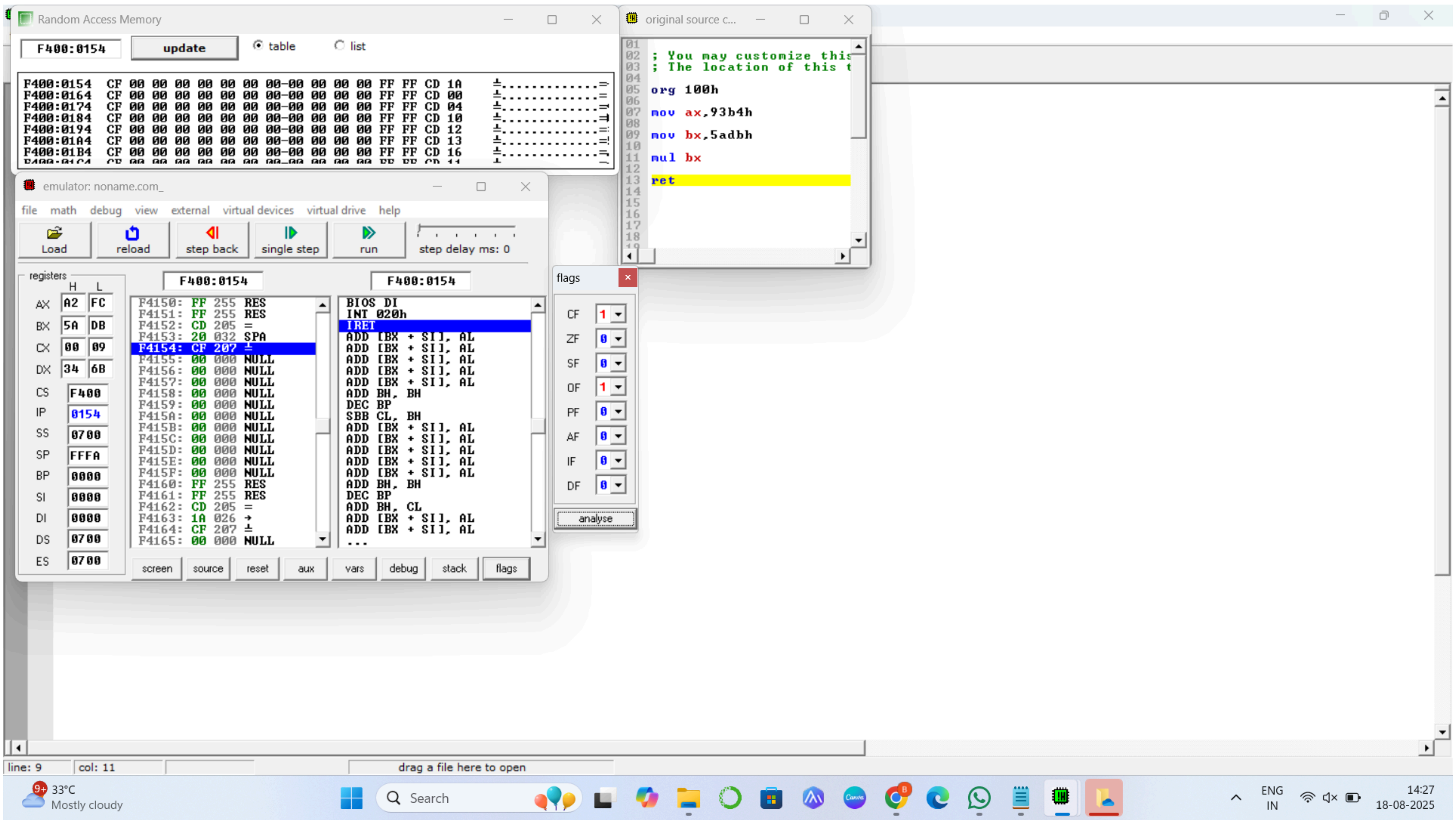
Output



Multiplication alp

```
org 100h
mov ax,93b4h
mov bx,5adbh
mul bx
ret
```

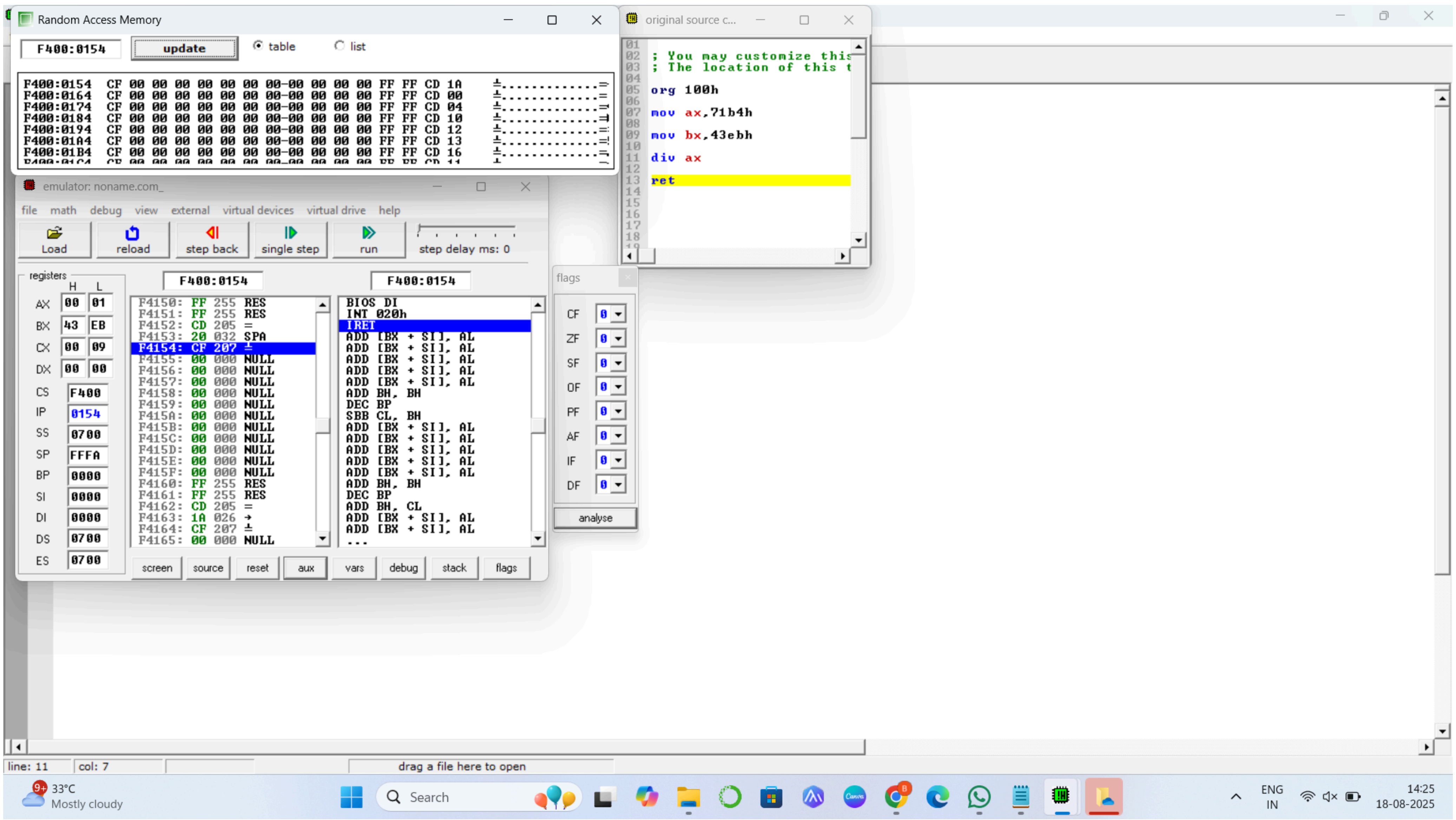
Output



Division alp

```
org 100h
mov ax,71b4h
mov bx,43ebh
div ax
ret
```

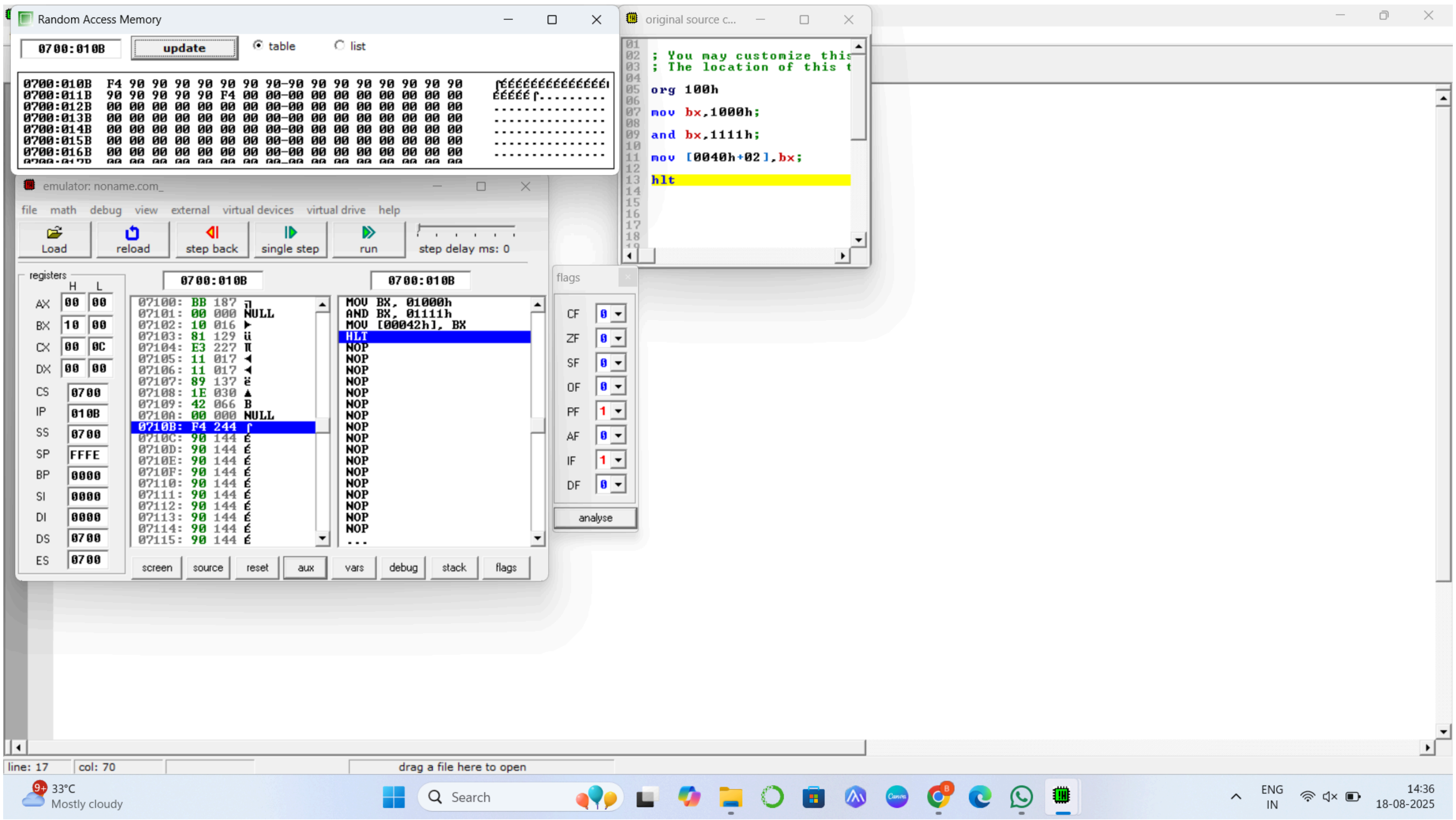
Output



AND

```
org 100h
mov bx,1000h;
and bx,1111h;
mov [0040h+02],bx;
hlt
```

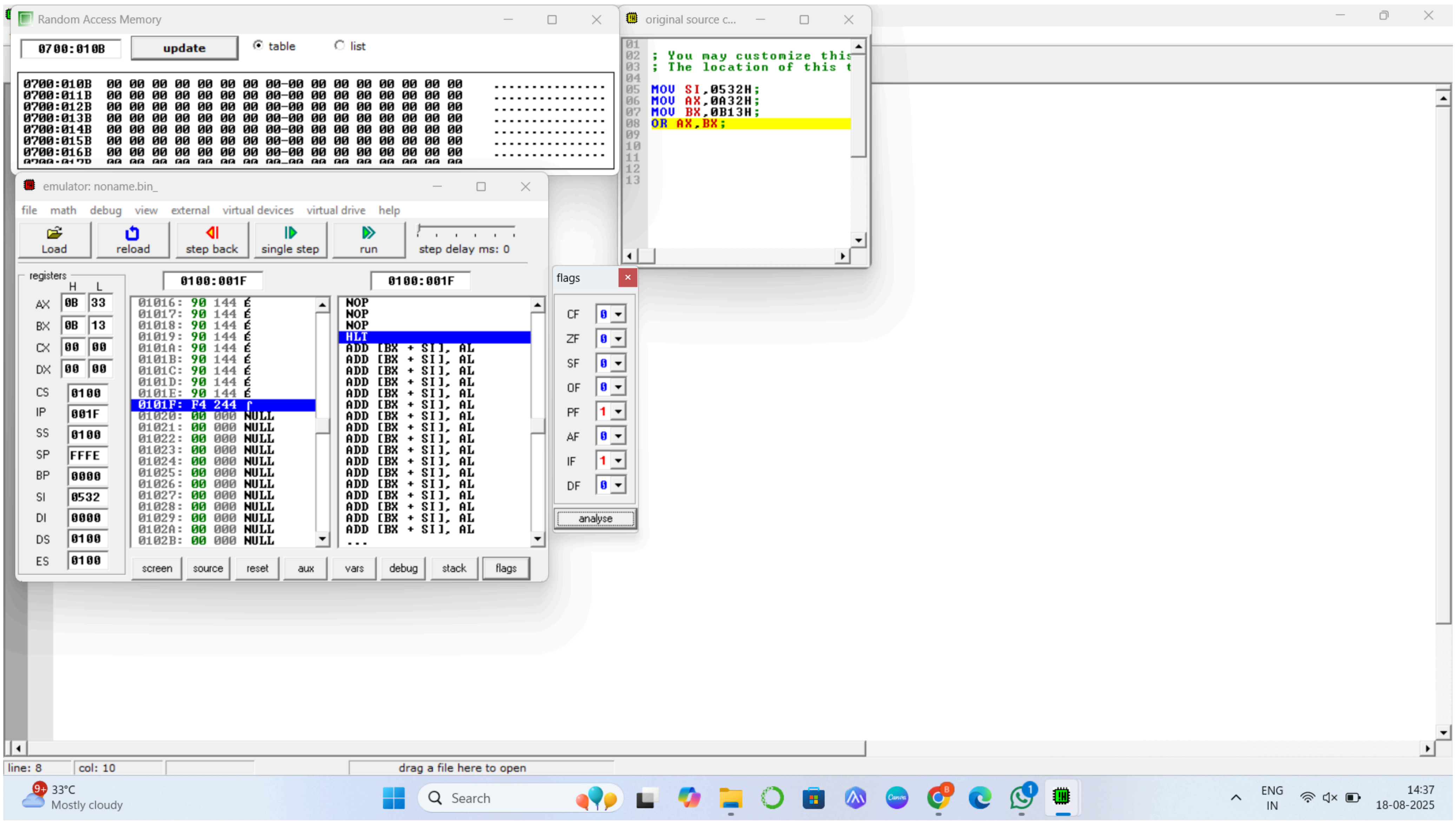
Output:



OR

```
MOV SI,0532H;
MOV AX,0A32H;
MOV BX,0B13H;
OR AX,BX;
```

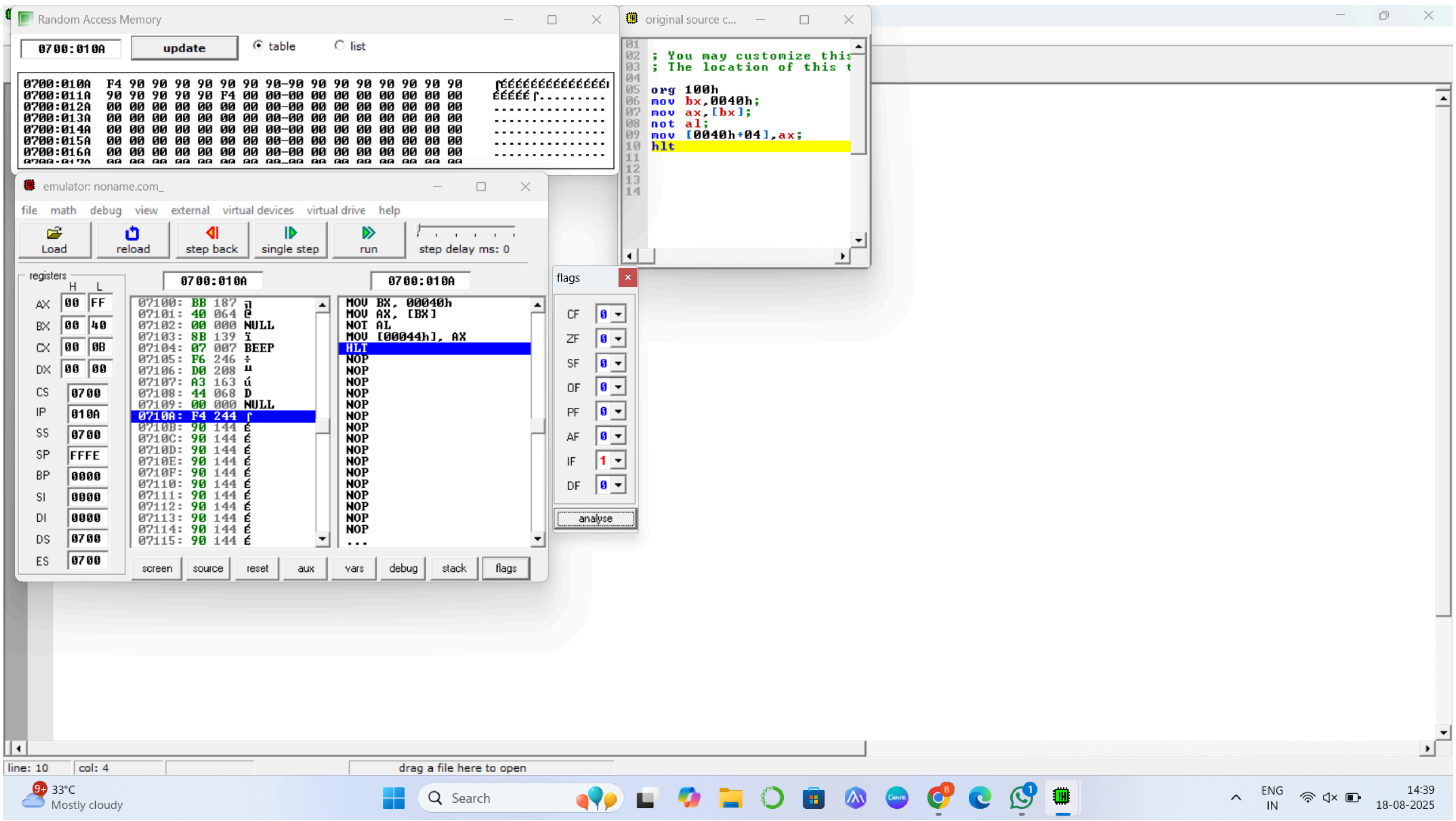
Output:



NOT

```
org 100h
mov bx,0040h;
mov ax,[bx];
not al;
mov [0040h+04],ax;
hlt
```

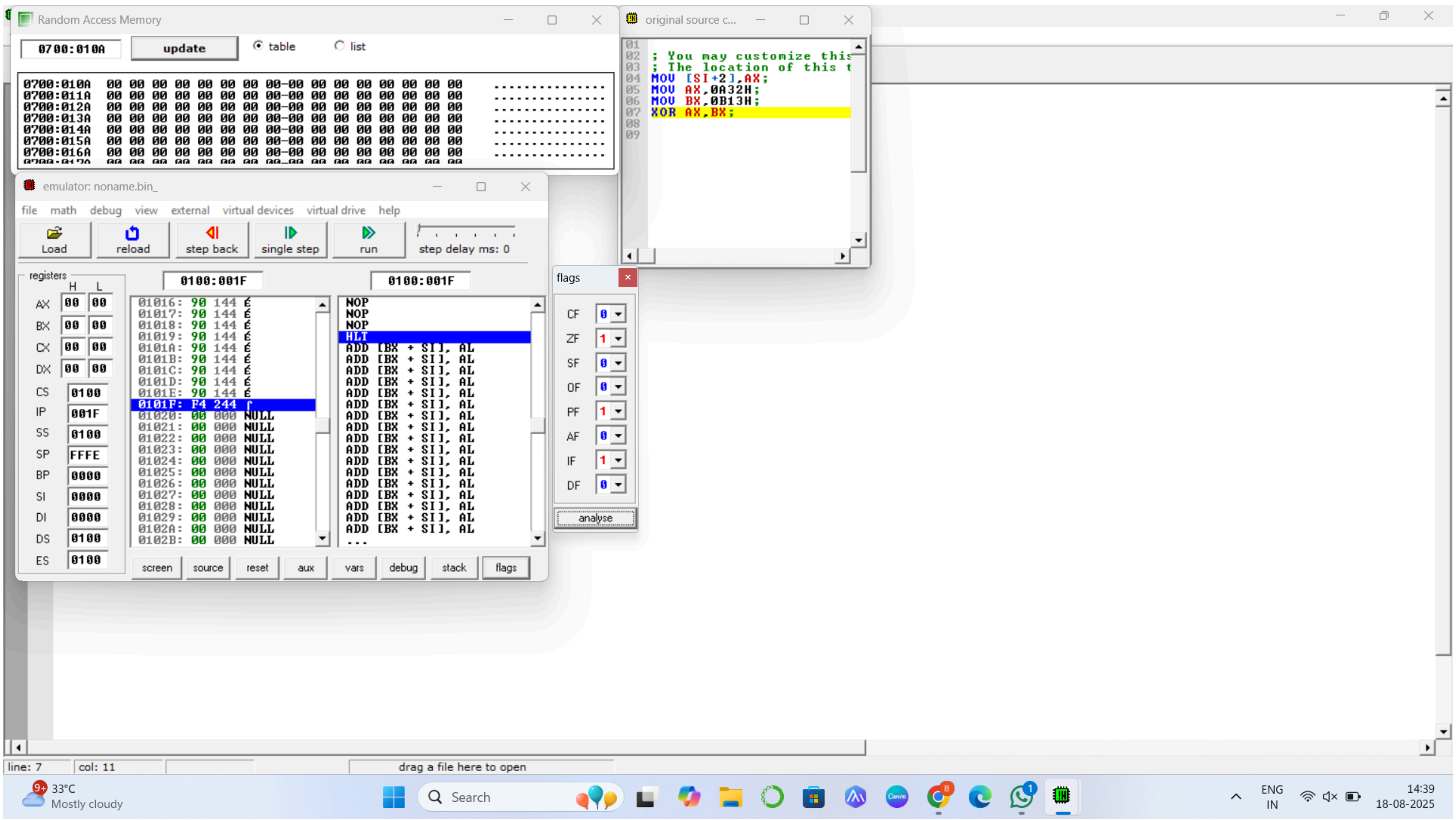
Output:



XOR

```
MOV [SI+2],AX;
MOV AX,0A32H;
MOV BX,0B13H;
XOR AX,BX;
```

Output:



Result :

Thus, to write and execute ALP on fundamental arithmetic operations and Logical operations is successful.