EXPERIMENT 10

AIM: Find the largest number in a set of 16, 8 bit numbers.

Algorithm:

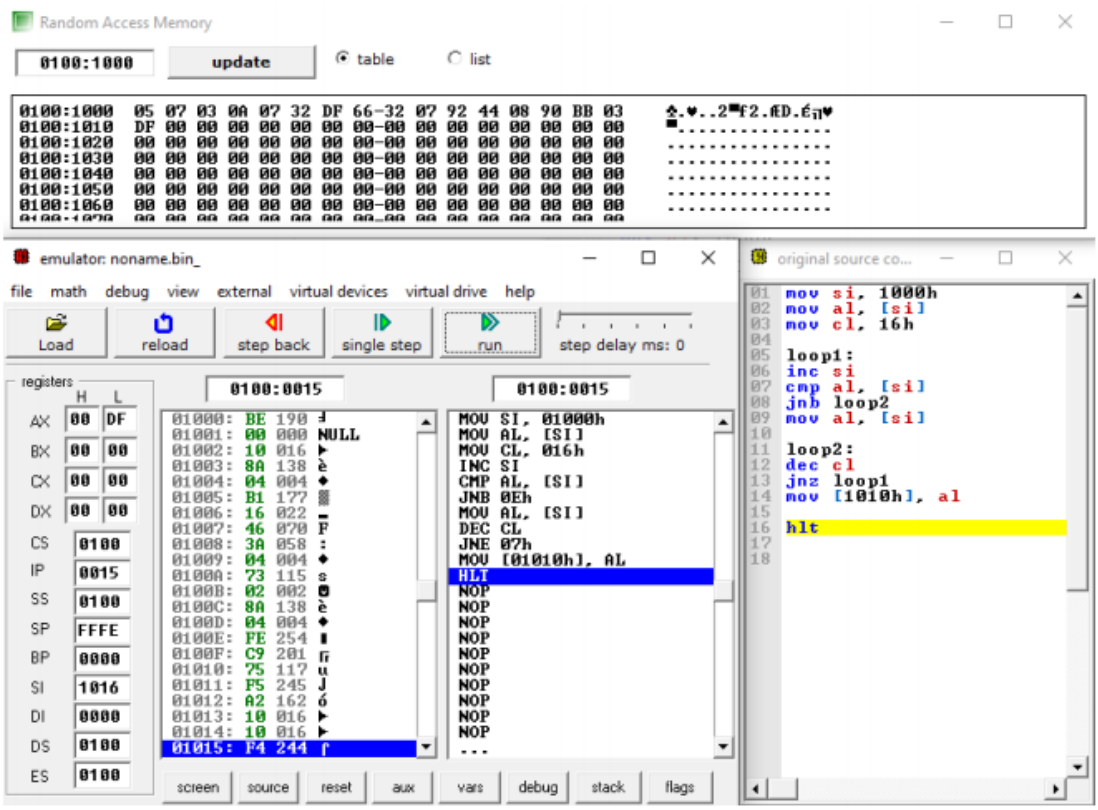
1) Assigns source index a value to point at an address '0100:1000h' in RAM.
2) Moving AL register to source index and fetching the 8-bit number in the register.
3) Initialising CL register with value 16h (0Fh), to run the loop 16 number of time (as the number of elements).
4) Now in loop1 value of SI is increased by one to move forward, and fetch value in AL register.
5) Then the fetched value is compared to the previously stored value and if the value is greater than the previous one, it get stored in AL register.
6) And if the fetched value is less than the previous value/value stored in AL, using JNB it jumps to loop2, without altering the value stored in AL.
7) The loop will inevitably come to loop2, doesn't matter if the value of AL is updated or not.
8) In loop2, the value of CL is decreased by one, as one element of the array has been scanned.
9) Using JNZ command, it moves back to loop1, if value of CL is not zero, to scan remaining elements.

10) When all the elements are scanned, we are left with the largest value in the AL register, which is then shifted to 1010h (just for clarity and understanding) .
11) Hence, the desired result is obtained and program is halted.

Code:

```
 mov SI, 1000h
mov AL, [SI]mov CL, 16h
loop1:
inc SI
cmp AL, [SI]
JNB loop2
mov AL, [SI]
loop2:
dec CL
JNZ loop1
mov [1010h], AL
hlt
```

Output:



(**ARKAJYOTI 2K19/EP/022)**