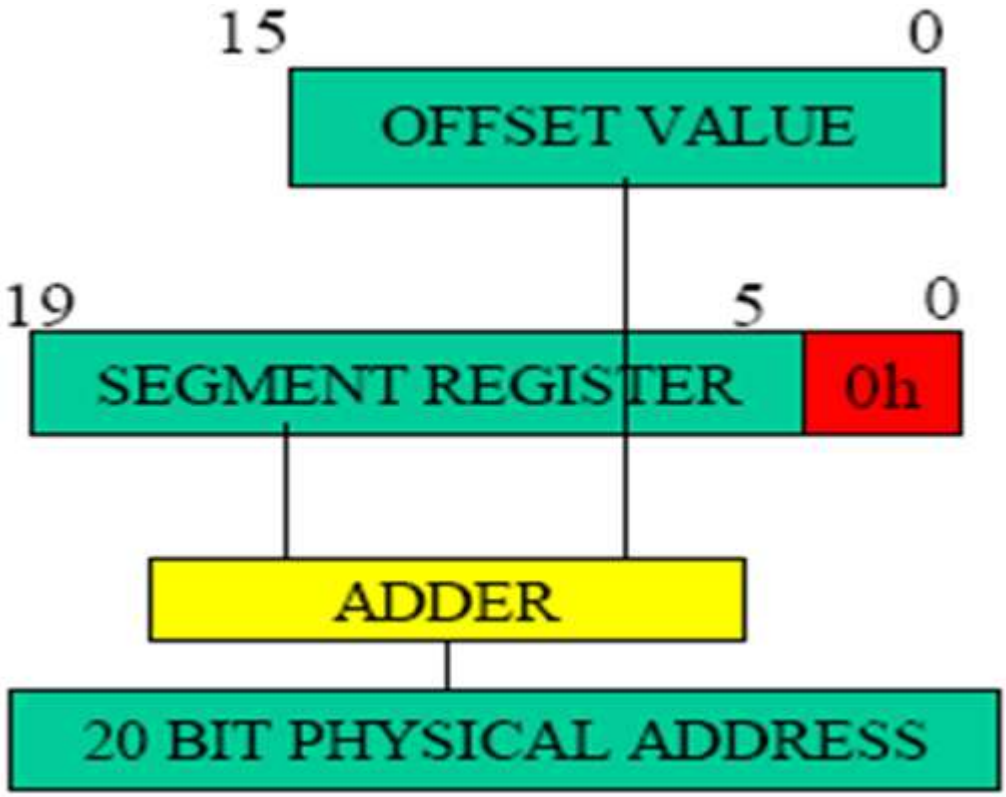
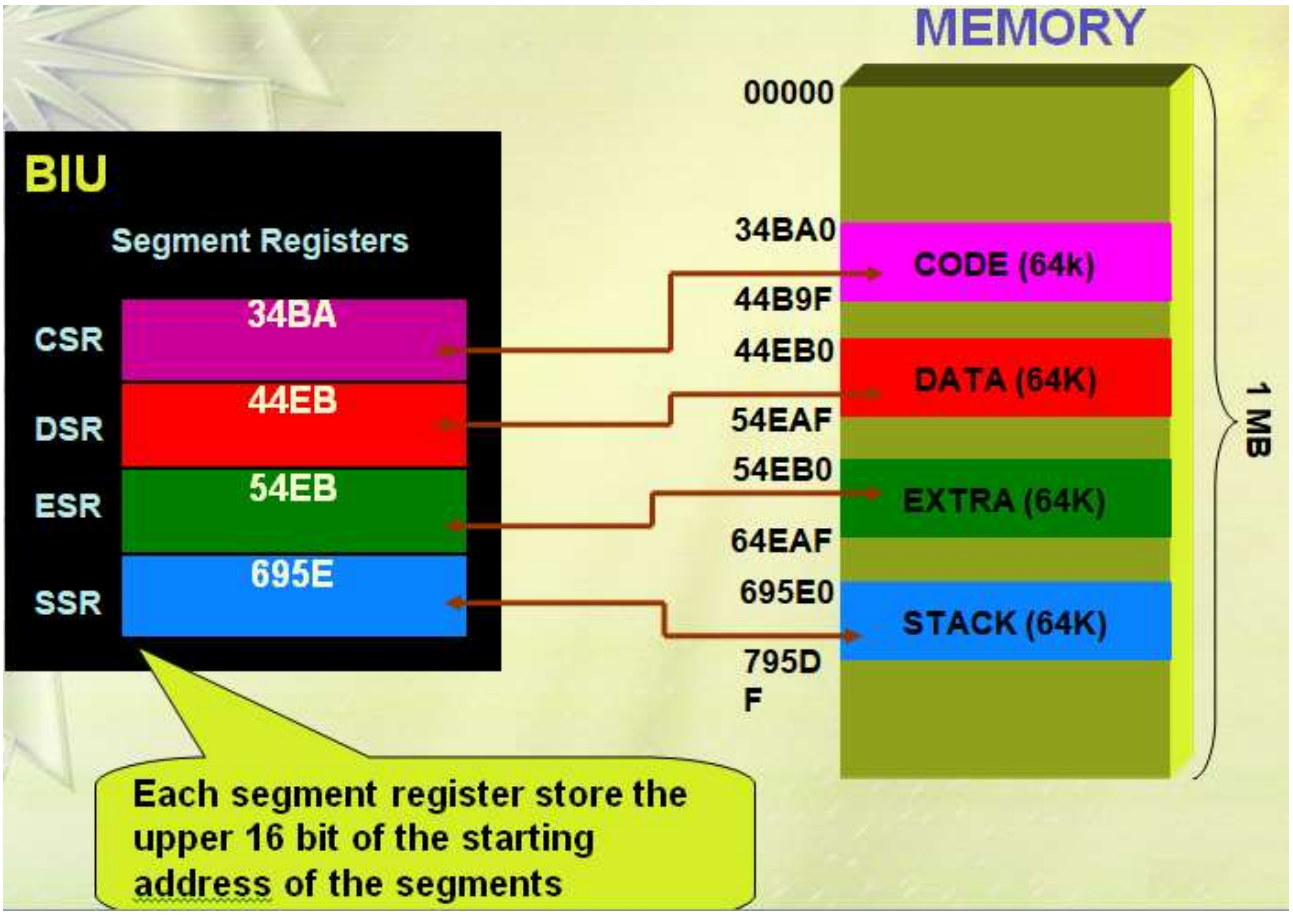
Convert a logic address to physical address

Instruction pointer & summing block

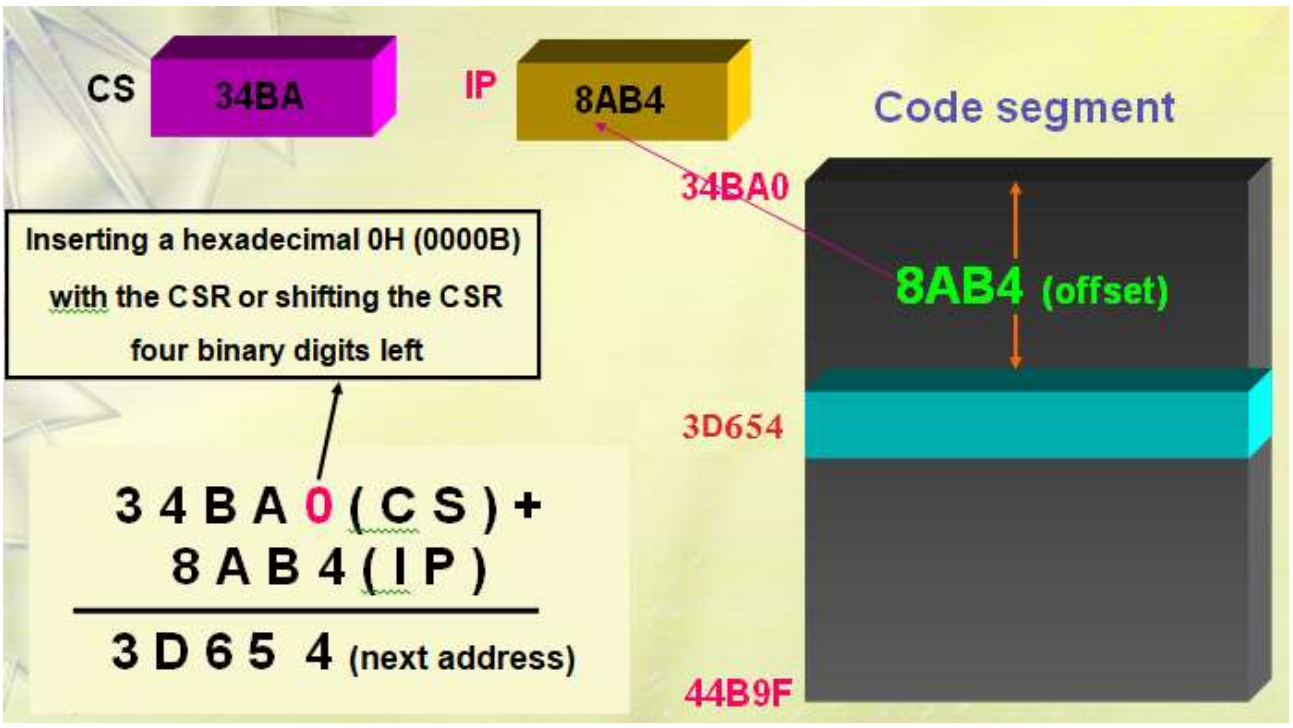
- **The instruction pointer register contains a 16-bit offset address of instruction that is to be executed next.**
- **The IP always references the Code segment register (CS).**
- **The value contained in the instruction pointer is called as an offset because this value must be added to the base address of the code segment, which is available in the CS register to find the 20-bit physical address.**
- **The value of the instruction pointer is incremented after executing every instruction.**
- **To form a 20bit address of the next instruction, the 16 bit address of the IP is added (by the address summing block) to the address contained in the CS , which has been shifted four bits to the left.**

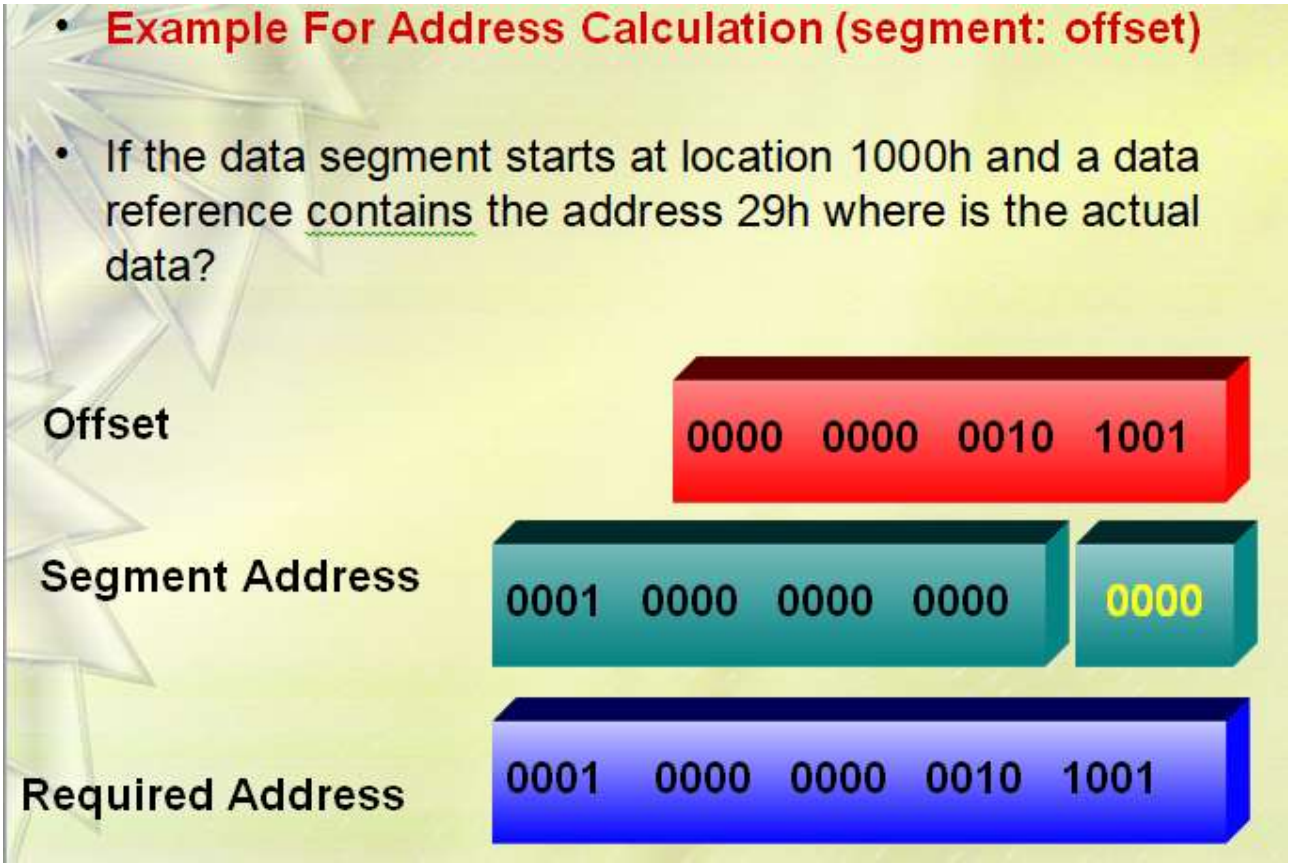


- **Starting and ending address of the segment in the memory**



- **The following examples shows the CS:IP scheme of address formation:**



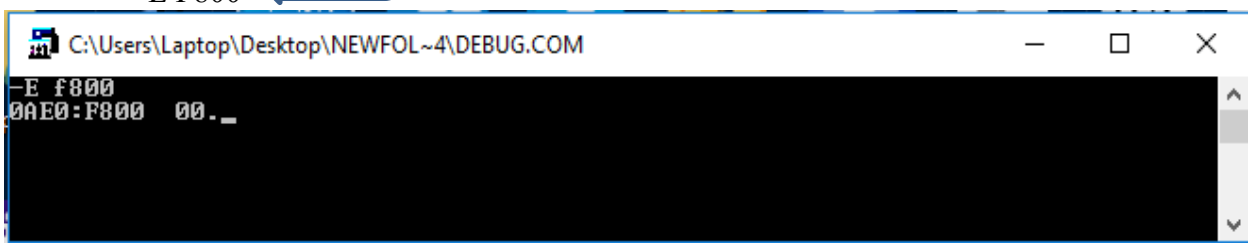


Enter: **E address [List]**

How to display the contents of one Byte in the specific location of memory? By using command (E):

Enter an ‘E or e ' at the first DEBUG prompt with a specific address, DEBUG will display something similar to this:

- E F800



‘00’ is the value stored in location F800

How to change a specific value of location in the memory? Also by using debugger command (E):

- E F800 ↵

Enter the new value

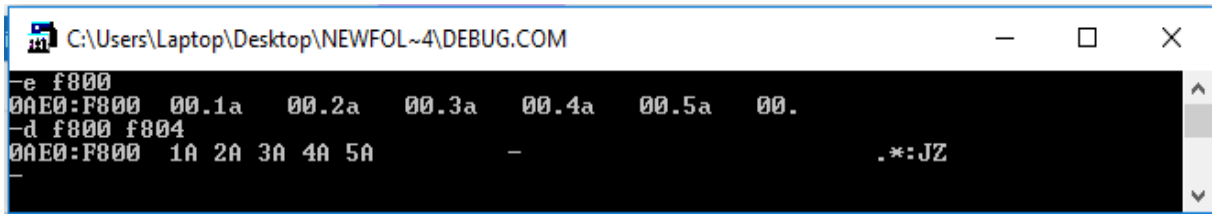
How to change the value of number of bytes? Also by using debugger command (E):

- E address [List] ↵
- E F800 1a 2a 3a 4a 5a 6a

How to display the contents of number of Bytes in the memory? By using command (E):

- E F800 ↵ after then pressing a **spaceBar** Key to disply next byte

Or by using D command as follows:



Hex add/sub: H value1 value2

How to add/sub two values in debugger? By using command (H):

- **H BBBB 1111** ↵