**Assignment 1**

**Name:** Vaghani Smit Dhirubhai

**Roll No:** 166

**College Id:** 19CEUEG022

1. **Use the FILL command (F) to initialize the 10h storage locations starting at DS:10 with the value 11h, the 10h storage locations starting at address DS:30 with 22h, the 10h storage locations starting at address DS:50 with 33h, and the 10h storage locations starting at address DS:70 with 44h**

**Ans:**

* initialize the 10h storage locations starting at DS:10 with the value 11h

-f 10L 0A 11

* initialize the 10h storage locations starting at address DS:30 with 22h

-f 30L 0A 22

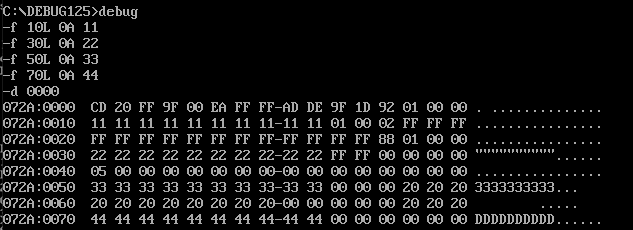
* initialize the 10h storage locations starting at address DS:50 with 33h

-f 50L 0A 33

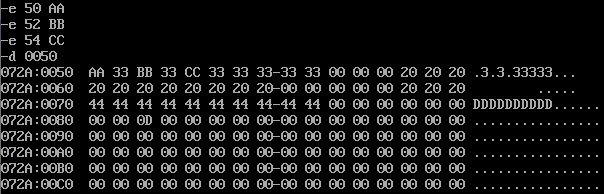
* initialize the 10h storage locations starting at address DS:70 with 44h

-f 70L 0A 44

**2. Verify the result of step 6 using the DUMP command.**

**Ans:**

**3.Use the ENTER command (E) to load locations CS:50, CS:52, and CS:54 with AA, BB, and CC, respectively.**

**Ans:**

**4. What is the extension of the file produced by the linker?**

**Ans:** .MAP is theextension of the file produced by the linker.

**5. Which debug commands allows us to see the memory contents?**

**Ans:**  dump(-d) debug commands allows us to see the memory contents.

**6. What is the difference between a logical address and a physical address?**

**Ans:** The fundamental difference between logical and physical address is that logical address is generated by CPU during a program execution whereas, the physical address refers to a location in the memory unit.

**7. Show how a physical address is generated from a logical address.**

**Ans:** physical address= logical address +offset value

**8. What are the following registers used for: DS, CS, SS, SP, IP, AX**

**Ans:**

* **Data segment register (DS):**it points to the data segment of the memory where the data is stored.
* **Code segment Register (CS):** It points to the segment of the running program.
* **Stack Segment Register(SS):**It points to stack segment.
* **Stack Pointer Register(SP):**SP is used to point the current stack.
* **Instruction Pointer Register(IP):**IP denotes the current pointer of the running program. .
* **Accumulator Register(AX)(General Purpose Register):**

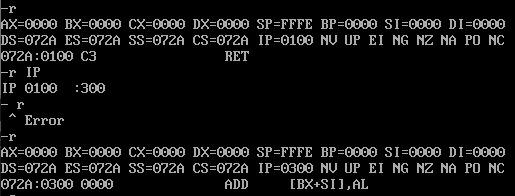
Most of the arithmetic operation is done with AX.

**9. Define the function each of the following flag bits in the flag register: Overflow, Carry, Sign, and Zero.**

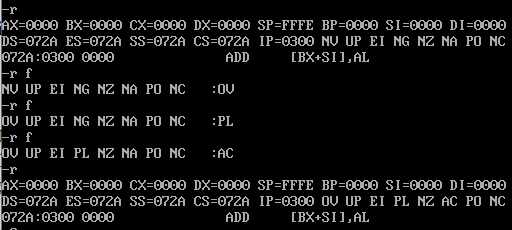
**Ans:**

* **Overflow flag:** this flag register set if arithmetic operation of two number is overflow otherwise reset.
* **Carry flag:** this flag register set if addition or subtraction of two number and carry or borrow generated otherwise reset.
* **Sign flag:** this flag register set if last operation result is negative otherwise reset.
* **Zero:** this flag register set if last operation result is zero otherwise reset.

**10. Use a REGISTER command to first display the current contents of IP and then change this value to 0300h.**

**Ans:**

**11. Use a REGISTER command to first display the current contents of the flag register and then reset the overflow, sign, and auxiliary flags.**

**Ans:**

**12. Using the ASSEMBLE command (A), load the program shown below into memory starting at address CS: 0100.**

**a. program**

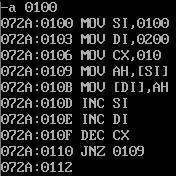
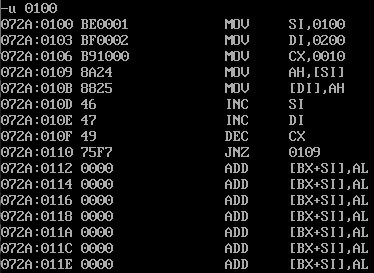
**b. Verify the loading of the program by displaying it with the UNASSEMBLE (U) command.**

**c. How many bytes of memory does the program take up?**

**d. What is the machine code for the DEC CX instruction?**

**e. What is the address offset for the label BACK?**

**Ans:**

* A.
* B.
* C. Program takes 101E(11E-100) bytes memory take up.
* D. Machine code for DEC CX is 072A:010F.
* E. the address offset for the label BACK is 0110

**13.What are the difference between T, G and P debug commands?**

**Ans:**

* **T (Trace command):**

while Go executes a whole block of code at one time, the Trace command executes instructions one at a time, displaying the registers after each instruction.

* **G (Go command):**

The Go command is used to start program execution. It can be used to start the execution at any point in the program, and optionally stop at any of points (breakpoints) in the program. If no breakpoints are set, program execution continues until termination.

* **P (Proceed command):**

The P command executes one or more instructions or subroutines. Whereas the T command trace into subroutine calls, the P command simply executes subroutines.

**Ex:**

P 100 5 Execute 5 instructions starting at CS:0100

P 3 Execute the next 3 instructions