**Computer organization and Architecture**

**Lab Report 1**

**Submitted by: Saad Masood**

**Submitted to: Lab Engr. Nausheen**

**Date : 21/2/2012**

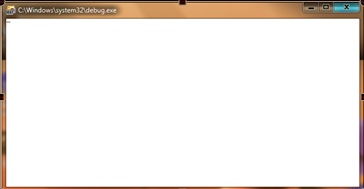
NC-Saad Masood

Course:- BESE 16A

# Q1:- Show how you can invoke DEBUG?

Ans: - To invoke DEBUG, Click Start button, write *debug* in run prompt and press Enter Key.

It shows me this window that is given below.

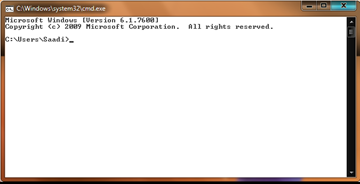


The hyphen prompt indicates that DEBUGis ready to accept commands in an interactive environment.

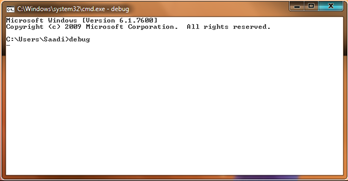
### Alternate:-

Click Start button, write *cmd* in run prompt and press Enter Key.

This window is appeared;

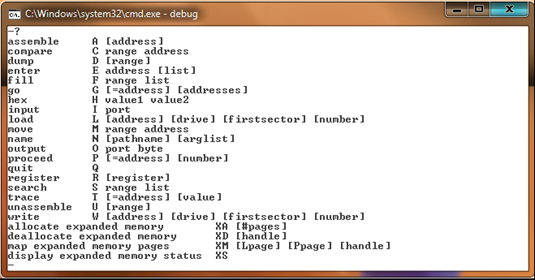


Type debug and this window is appeared.



# Q2:- How can you display list of all valid commands? Also, show the list of commands displayed? What information is displayed in the list against each command?

Ans: - Type “?” to display list of all valid commands.

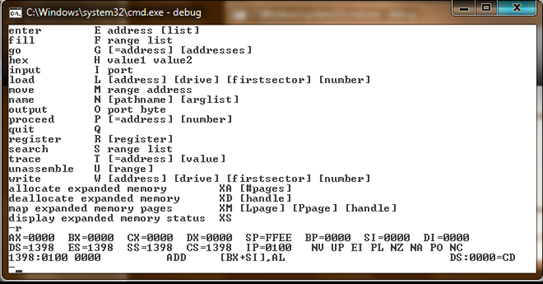


* Program Creation and Debugging
  + A: Assemble a program using instruction mnemonics(reminders)
  + G : Execute the program currently in memory
  + R: Display the contents of register and flags
  + P: Trace past an instruction, procedure, or loop
  + T: Trace a single instruction
  + U: Disassemble memory into assembler mnemonics
* Memory Manipulation
  + C: Compare one memory range with another
  + D: Dump (display) the contents of memory
  + E: Enter bytes into memory.
  + F: Fill a memory range with a single value
  + M: Move bytes from one memory range to another
  + S: Search a memory range for specific value
* Miscellaneous
  + H: Perform hexadecimal addition and subtraction
  + Q: Quit DEBUG and return to DOS
* Input-Output
  + I: Input a byte from port
  + L: Load data from disk
  + O: Output a byte to a port
  + N: Create a filename for use by the L and W Commands
  + W: Write data from memory to disk.

# Q3:- Show how you can execute r command in different formats?

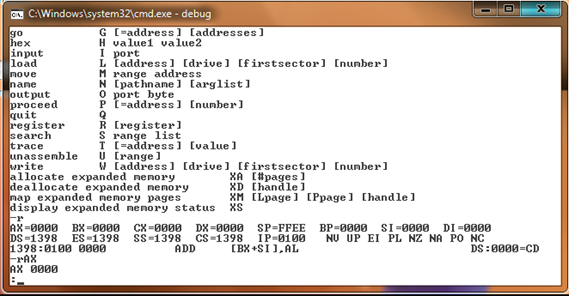
Ans:- r command is for registers.

We can simple type r to see all the registers.



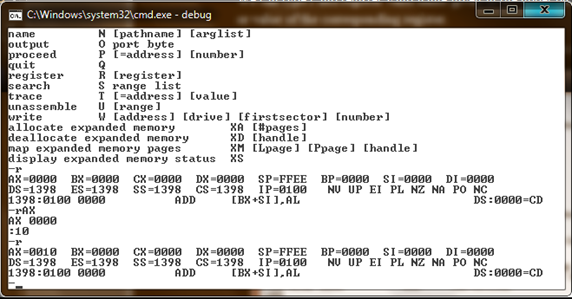
We can type r and register valid name given in the upper picture to change the default value or value of the corresponding register.

For example I want to change value of AX register. I type r AX and press Enter



And after “:” I enter changed value. Like 10 (this value is in hexadecimal minimum value is 0000 and maximum value is FFFF)

And again type r to check.



And AX value is changed to 0010.

# Q4:- What is a register?

Ans:- Registers are high-speed storage locations inside the CPU, designed to be accessed at much higher speed than conventional memory.

* **Types of registers**
  + General Purpose/Data Registers
    - Hold data for an operation to be performed
    - There are 4 data registers (AX, BX, CX, DX)
  + Segment Registers
    - Hold the address of an instruction or data element
    - Segment registers (CS, DS, ES, SS)
  + Index and Pointer Registers
    - Pointer registers (SP, BP, IP)
    - Index registers (SI, DI)
  + Status/Flags Register (FLAGS)
    - Keeps the current status of CPU or results of arithmetic operations

# Q5:- What is the difference in registers and flags?

Ans:- **Registers** are high-speed storage locations inside the CPU, designed to be accessed at much higher speed than conventional memory.

**Flag** register is 16-bit special register .Each bit position is assigned to show the status of CPU or the results of arithmetic operations. Each relevant bit position is given a name; other positions are undefined.

* **Two Types:**
  + **Control Flags:** Individual bits may be set to control the CPU’s Operation
  + **Status Flags:** The Status bits reflect the outcome of arithmetic and logical operations performed by the CPU.

|  |  |
| --- | --- |
| **Registers** | **Flag Registers** |
| High-speed storage locations inside the CPU. | 16-bit special register. They are types of registers. |
| They are designed to be accessed at much higher speed than conventional memory. | Each bit position is assigned to show the status of CPU. |
| They have four types. data register, segment register, index register, flag registers | They have two types. Control flag, status flag. |

# 

# Q:-6 Briefly explain the purpose of each register. For Example, ax (accumulator) is a data register used by CPU for arithmetic operations.

Ans:- **AX (Accumulator):** Used by CPU for arithmetic operations.

* **BX (Base):** It can hold a memory address that points to a variable.
* **CX (Counter):** Act as a counter for repeating or looping instructions. These instructions automatically repeat and decrement CX and quit when equals to 0.
* **DX (Data):** It has a special role in multiply and divide operations.
* **CS (Code)**: Defines the starting address of the section of memory holding code.
* **DS (Data)**: Defines the section of memory that holds most of the data used by programs.
* **ES (Extra)**: This is an additional data segment that is used by some of the string instructions.
* **SS (Stack**): It defines the area of the memory used for stack.
* **IP - instruction pointer**: Always points to next instruction to be executed. IP register always works together with CS segment register and it points to currently executing instruction.
* **SI - source index register**: Can be used for pointer addressing of data. Offset address relative to DS
* **DI - destination index register**: Can be used for pointer addressing of data . Offset address relative to ES
* **SI** and **DI** used in string movement instructions.
* **SP** and **BP** are used to access data inside the stack segment
* **BP - base pointer**: Primarily used to access parameters passed via the stack. Offset address relative to SS
* **SP – stack pointer**: Always points to top item on the stack. Offset address relative to SS
* The 80386/80486 processor contain 32-bit registers which greatly improve the efficiency of program that take advantage of them.
  + EAX, EBX, ECX, EDX,
  + EFLAGS
  + EIP
  + EBP, ESP, ESI, EDI.