Executing Computer Instructions

Dr . Shahenda sarhan

Lecturer @ CS Department

DEBUG

- DEBUG is a DOS utility used for checking/changing the content of memory, for executing programs instruction by instruction, for storing and executing small programs.

Debug Commands

- **A** command allows you to assemble symbolic instructions into machine code

Ex: A 100 <Enter> allows you to enter assembly instructions starting at offset 100 from the code segment in CS register

```
A 100              <enter>
MOV CL ,42         <enter>
MOV DL , 2A        <enter>
MOV CL ,DL         <enter>
NOP                <enter>     <enter>
```

The debug will display :

```
Xxxx:100    MOV CL ,42
Xxxx:102    MOV DL , 2A
Xxxx: 104   MOV CL ,DL
Xxxx:106    NOP
```

Debug Commands

- U :Unassemble the machine code into symbolic code

ex: U 100,106 <enter>

```
xxxx:100   B142   MOV CL ,42
xxxx:102   B22A   MOV DL , 2A
xxxx: 104  00D1   MOV CL ,DL
xxxx:106   90      NOP
```

- D :displays content of memory locations.

D **DS:00 displays the content of memory starting at address stored in data segment DS with offset 00.**

- E :enter data into memory beginning at specific location

E CS:100 B8 23 01 05 25 00

- R : Displays the content of all registers

-R

AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000
DI=0000 DS=17C1 ES=17C1 SS=17C1 CS=17C1 IP=0100 NV UP EI PL
NZ NA PO NC

Debug Commands

- N :name a program
- P : proceed or execute a set of related instruction .
- Q :quit the debug session
- T :trace the execution of one instruction

```
-R
AX=0000  BX=0000  CX=0000  DX=0000  SP=FFEE  BP=0000  SI=0000  DI=0000
DS=17C1  ES=17C1  SS=17C1  CS=17C1  IP=0100   NV UP EI PL NZ NA PO NC
17C1:0100 B8FF00       MOV     AX,0123
-T
AX=0123  BX=0000  CX=0000  DX=0000  SP=FFEE  BP=0000  SI=0000  DI=0000
DS=17C1  ES=17C1  SS=17C1  CS=17C1  IP=0103   NV UP EI PL NZ NA PO NC
17C1:0103 B900FF       MOV     CX,FF00
-T
AX=0123  BX=0000  CX=FF00  DX=0000  SP=FFEE  BP=0000  SI=0000  DI=0000
DS=17C1  ES=17C1  SS=17C1  CS=17C1  IP=0106   NV UP EI PL NZ NA PO NC
17C1:0106 01C8         ADD     AX,CX
```

- G : run executable program into memory
- W :write a program onto disk

Ex: to check system equipments using machine language

- Two bytes in BIOS at address 410H and 411H contain the number of serial and parallel ports, status of diskette device, initial video mode, if coprocessor is present or not.
- Type D 40:10 and copy the first two bytes(6344). Reverse the order(4463). Write content in binary and number the bits from 0 to 15 Decode the content as follows:
  - bits 15, 14 – number of parallel ports attached.
  - bits 11 – 9 – number of serial ports attached.
  - bits 7, 6 - number of diskette devices.
  - bits 5, 4 – initial video mode (= 10 means 80 X 25 color).
  - bit 1 is one when coprocessor is present.
  - bit 0 is one if diskette drive is present.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0	1	1	0	0	0	1	1

Practice (machine language)

- Checking check ROM BIOS date starting at address FFFF5H
- Checking memory size starting at address 413H
- Checking serial number and copyright notice starting at address FE000H
- Checking model ID starting at address FFFFEH

Requesting information about system using assembly commands

- Getting current date

```
A  100
MOV AH,2A
INT   21
NOP
R
T
```

- Size of memory

```
 A  100
INT  12
NOP
 R
 T
```

Examples

- **Print your name**

```
A   100
MOV AH, 09
MOV DX, 109
INT 21
NOP
DB ‘your name’,’$’
```


good
Luck
U
U