

Executing Computer Instructions

Objective

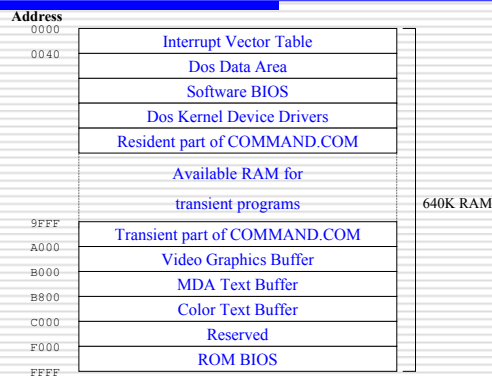
■ DEBUG Program

- DEBUG Commands
- Rules of DEBUG Commands
- DEBUG Display
- Viewing Memory Locations

■ Machine and Assembly Language

- Keying in program instructions and data
- Execute program instructions

Map of the First MB of Memory



Registers

■ High-speed storage directly inside the CPU (CPU has internal data bus that is twice as wide as external data bus)

- Segment registers
- Pointer registers
- General-purpose registers
- Index registers
- Flags register

Segment Registers

- Used as base locations (addresses) for program instructions, data, and stack
 - CS (code segment) -- holds the base location for all executable instruction (code) in a program
 - DS (data segment) -- default base location for variables
 - SS (stack segment) -- holds the stack base location
 - ES (extra segment) -- additional base location

Pointer Registers

Contain the offsets of data instructions

- IP (instruction pointer)
 - contains the offset address of the next instruction to be executed
 - associated with CS register
- BP (base pointer)
 - contains an assumed offset from SS register
 - used by subroutine to locate variables
- SP (stack pointer)
 - contains the offset of the top of the stack
 - SS and SP combine to form a complete address

Instruction Pointer Register

Segment address in CS	39B40h
Offset address in IP	+514h
Address of next instruction	3A054h

General Purpose Registers

- Data registers -- used for arithmetic and data movement
- can be addressed as either an 8-bit or a 16-bit
- AX (accumulator) -- arithmetic operations
- BX (base) -- hold address of a procedure or variable
- CX (counter) -- a counter for repeating or looping instructions
- DX (data) -- a special role in multiply and divide operations

Index Registers

Contain the offsets of data instructions

- **SI (source index)** -- source string is pointed to by the SI register
- **DI (destination index)** -- acts as the destination for string movement instruction

Register Names

8-Bit	16-Bit	32-Bit
AL	AX	EAX
AH		
BL	BX	EBX
BH		
CL	CX	ECX
CH		
DL	DX	EDX
DH		
	SI	ESI
	DI	EDI
	BP	EBP
	SP	ESP

Flags Register

O = Overflow -- indicate overflow of the left most bit following arithmetic

D = Direction -- determine left or right direction for moving or comparing data

I = Interrupt -- indicate that all interrupts to be processed or ignored

T = Trap -- permit operation of the processor in single-step-mode

S = Sign -- indicate the resulting sign of an arithmetic operation, 0 (negative), 1 (positive)

Z = Zero -- indicate the resulting sign of an arithmetic or comparison operation, 0 (nonzero), 1 (zero) result

A = Auxiliary carry -- contain a carry out of bit 3 on 8-bit data

P = Parity -- indicate even or odd parity of a low-order 8-bit data operation

C = Carry -- contain the leftmost bit

x = undefined

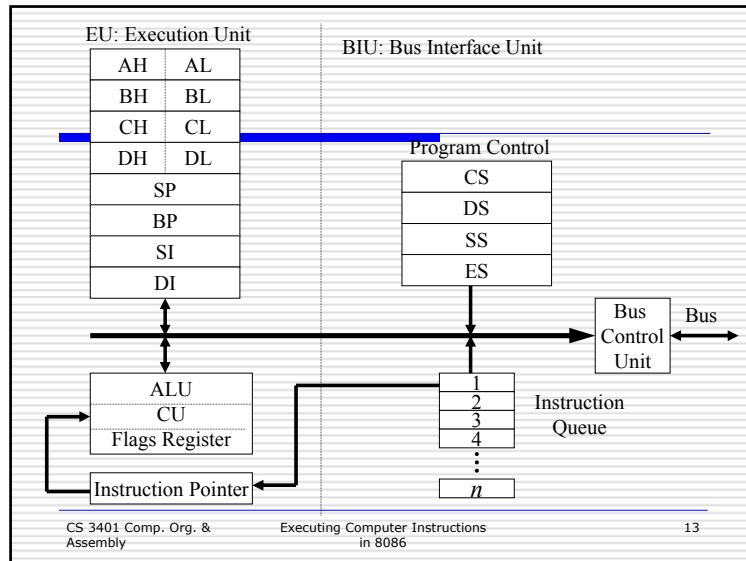
Flags Registers

```

MS-DOS Prompt - DEBUG
Auto
AX=1000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=0102 NU UP EI PL NZ NA PO NC
OF6C:0102 CD16 INT 16
    
```

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

x x x x 0 D I T S Z x A x P x C



DEBUG Program

- DEBUG Commands
- Rules of DEBUG Commands
- DEBUG Display
- Viewing Memory Locations

DEBUG Program

DEBUG is a DOS program that is used to

- view memory
- enter machine code or assembly code in memory
- enter data
- trace code execution
- single step tracing

DEBUG Commands

- A:** Assemble symbolic instructions into machine code
- D:** Display contents of memory at a specific address
- E:** Enter data/instructions into memory
- U:** Unassemble machine code into symbolic code
- G:** Run the program
- T:** Trace the execution of one instruction
- P:** Proceed or execute a set of instructions
- R:** display contents of Registers
- W:** Save a program onto disk
- Q:** Quit

Rules of DEBUG Commands

- Not case insensitive
- use colon to specify segment and offset
- use hexadecimal numbers
- use a space to separate parameters in a command

The DEBUG Display

Display contents of memory at offset 200H in DS using D command

Address	HEX Representation	ASCII
C:\> eg		
-d ds:200		
0F6C:0200	B8 00 42 33 C9 8B D1 CD-21 EB 24 3D 05 00 F9 75	..B3...!.\$=...u
0F6C:0210	03 E9 5B FF BE E7 04 33-C9 2E A1 D1 E4 BB 22 00	..[...3.....".
0F6C:0220	BA 12 01 BF 01 00 CD 21-73 03 E9 42 FF 8B D8 B0[s...B....
0F6C:0230	FF 86 47 18 A2 19 00 C3-50 33 C9 FC AC 41 0A C0	..G.....P3...A..
0F6C:0240	75 FA 2B F1 58 C3 73 FD-9C 53 51 56 57 55 06 1E	u.+X.s...SQVWU..
0F6C:0250	50 52 B4 59 CD 21 59 5B-BA A1 80 3D 41 00 74 04	PR.Y.IY[...=A.t.
0F6C:0260	8B C3 8B D1 1F 07 5D 5F-5E 59 5B 9D C3 E8 D6 FF]^[.....
0F6C:0270	CB 56 57 51 BF CB D7 33-C9 8B C1 57 AC 3C 00 74	.VWQ...3...W.<.t

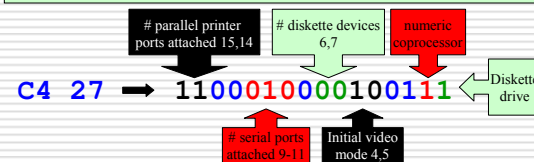
Viewing Memory Locations

- System equipment
- memory size
- serial number and copyright notice
- ROM BIOS Date
- Model ID

System Equipment

Location of equipment status word in the BIOS data area is at 410h-411h

```
-d 40:10
0040:0010 27 C4 00 80 02 20 00 00-00 00 32 00 32 00 33 04  '.....2.2.3.
0040:0020 0D 1C 64 20 20 39 34 05-30 0B 3A 27 31 02 30 0B  '...d 94.0.:1.0.
```



Based Memory Size

Size of based memory is at location 413h and 414h

```
-d 40:13
0040:0010      80 02 20 00 00-00 00 22 00 22 00 33 04      ....."3.
0040:0020 0D 1C 08 0E 08 0E 30 0B-30 0B 3A 27 30 0B 0D 1C  ....0.:'0...
```

02 80 ↔ 640

ROM BIOS Date

ROM BIOS manufacture date begins at location FFFF5h

```
-d FFFF:5
FFFF:0000      31 30 2F-31 36 2F 39 36 00 FC 56      10/16/96..V
FFFF:0010 00 00 00 56 44 49 53 4B-33 2E 33 80 00 01 01 00  ...VDISK3.3.....
```

Machine and Assembly Language

- Key in program instructions
- Execute program instructions
- Save a program

Machine Language Example

Machine Code	Assembly Code
B82301	MOV AX, 0123
052500	ADD AX, 0025
8BD8	MOVE BX, AX
03D8	ADD BX, AX
8BCB	MOV CX, BX
2BC8	SUB CX, AX
90	NOP

```

MS-DOS Prompt - DEBUG
Auto
C:\>DEBUG
-E CS:100 B8 23 01 05 25 00
-E CS:106 8B D8 03 08 8B CB
-E CS:10C 2B C8 90
-R
AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=0100  NU UP EI PL NZ NA PO NC
0F6C:0100 B82301      MOV     AX,0123
-T
AX=0123 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=0103  NU UP EI PL NZ NA PO NC
0F6C:0103 052500      ADD     AX,0025
-T
AX=0148 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=0106  NU UP EI PL NZ NA PE NC
0F6C:0106 8B08      MOV     BX,AX
-T
AX=0148 BX=0148 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=0108  NU UP EI PL NZ NA PE NC
0F6C:0108 0308      ADD     BX,AX
-

```

CS 3401 Comp. Org. & Assembly Executing Computer Instructions in 8086 25

```

MS-DOS Prompt - DEBUG
Auto
AX=0148 BX=0148 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=0108  NU UP EI PL NZ NA PE NC
0F6C:0108 0308      ADD     BX,AX
-T
AX=0148 BX=0290 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=010A  NU UP EI PL NZ AC PE NC
0F6C:010A 8BCB      MOV     CX,BX
-T
AX=0148 BX=0290 CX=0290 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=010C  NU UP EI PL NZ AC PE NC
0F6C:010C 2BC8      SUB     CX,AX
-T
AX=0148 BX=0290 CX=0148 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=010E  NU UP EI PL NZ AC PE NC
0F6C:010E 90      NOP
-T
AX=0148 BX=0290 CX=0148 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=010F  NU UP EI PL NZ AC PE NC
0F6C:010F C0      DB     C0
-

```

CS 3401 Comp. Org. & Assembly Executing Computer Instructions in 8086 26

Code Segment Display

```

MS-DOS Prompt - DEBUG
Auto
-D CS:100
0F6C:0100 B8 23 01 05 25 00 8B D8 03 08 8B CB 2B C8 90 C0  #..%.....+..
0F6C:0110 90 F9 75 04 3C 3B 75 F6 4E C3 1E 52 34 00 5B 0F  .u.<;u.N..B%.
0F6C:0120 57 2E 8E 1E DE 00 80 3E 43 04 00 75 0D F6 06 21  W.....>C..u..!
0F6C:0130 04 FF 75 06 E8 0B 00 E8 59 00 5F 5E 59 5B 58 5A  .u.....V..^V[XZ
0F6C:0140 1F C3 2E 80 3E 77 E0 00 74 F7 1E 0E 1F BE 77 E0  .>w..t.....w.
0F6C:0150 E8 91 02 2E A1 01 E4 BB 40 00 BA 01 00 33 FF CD  .....@....3..
0F6C:0160 21 1F 72 0B 8B D8 B0 FF 86 47 18 A2 18 00 C3 0E  !.r.....6.....
0F6C:0170 1F E8 02 00 3D 41 00 74 07 0B FF 74 06 B8 8F 80  ....=A.t...t....
-D CS:100,110
0F6C:0100 B8 23 01 05 25 00 8B D8 03 08 8B CB 2B C8 90 C0  #..%.....+..
0F6C:0110 90
-

```

CS 3401 Comp. Org. & Assembly Executing Computer Instructions in 8086 27

Debug Operations

- Keying in program instructions
- Executing program instructions
- Displaying memory contents
- Correcting an entry

Machine Language Example with Defined Data

DS Offset	Hex Contents
0200h	2301h
0202h	2500h
0204h	0000h
0206h	2A2A2Ah

Program Instructions with Defined Data

Machine Code	Assembly Code
A10002	MOV AX, [0200]
03060202	ADD AX, [0202]
A30402	MOV [0204], AX
90	NOP

```

MS-DOS Prompt - DEBUG
Auto
C:\>DEBUG
-E DS:200 23 10 25 00 00 00
-E DS:206 2A 2A 2A
-E CS:100 A1 00 02 03 06 02 02
-E CS:107 A3 04 02 90
-D DS:200,208
DF6C:0200 23 10 25 00 00 00 2A 2A-2A          #.Z...***
-
-D CS:100,10A
DF6C:0100 A1 00 02 03 06 02 02 A3-04 02 90
-R
AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=0100  NV UP EI PL NZ NA PO NC
DF6C:0100 A10002      MOV     AX,[0200]          DS:0200=1023
-I
AX=1023 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=0103  NV UP EI PL NZ NA PO NC
DF6C:0103 03060202    ADD     AX,[0202]          DS:0202=0025
-I
AX=1048 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=0107  NV UP EI PL NZ NA PE NC
DF6C:0107 A30402      MOV     [0204],AX        DS:0204=0000
-I
AX=1048 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=010A  NV UP EI PL NZ NA PE NC
DF6C:010A 90              NOP
-
CS 3401 Comp. Org. & Assembly
Executing Computer Instructions in 8086
31
  
```

Assembly Language Example

Assemble Command (A)

Use assemble command to key in assembly instructions

```

MOV CL, 42
MOV DL, 2A
ADD CL, DL
NOP
  
```

Unassemble Command (U)

Unassemble command displays the machine code for assembly language instructions

MS-DOS Prompt - DEBUG

```

-A 100
0F6C:0100 MOV CL,42
0F6C:0102 MOV DL,2A
0F6C:0104 ADD CL,DL
0F6C:0106 NOP
0F6C:0107
-U
0F6C:0100 B142      MOV     CL,42
0F6C:0102 B22A      MOV     DL,2A
0F6C:0104 D0D1      ADD     CL,DL
0F6C:0106 90        NOP
0F6C:0107 A3D402      MOV     [0204],AX
0F6C:010A 90        NOP
0F6C:010B CB        RETF
0F6C:010C 2BC8      SUB     CX,AX
0F6C:010E 90        NOP
0F6C:010F CD        DB     CD
0F6C:013A 5F        POP    DI
0F6C:013B 5E        POP    SI
0F6C:013C 59        POP    CX
0F6C:013D 5B        POP    BX
0F6C:013E 58        POP    AX
0F6C:013F 5A        POP    DX
  
```

CS 3401 Comp. Org. & Assembly Executing Computer Instructions in 8086 33

Another Example

```

MOV AX,5
ADD AX,10
ADD AX,20
MOV [0102],AX
  
```

CS 3401 Comp. Org. & Assembly Executing Computer Instructions in 8086 34

INT (Interrupt) Instruction

Current date - INT 21h

MS-DOS Prompt - DEBUG

```

C:\>DEBUG
-A 100
0F6C:0100 MOV AH,2A
0F6C:0102 INT 21
0F6C:0104 NOP
0F6C:0105
-B
AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=0100  NU UP EI PL NZ NA PO NC
0F6C:0100 B42A      MOV     AH,2A
-I
AX=2A02 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=0102  NU UP EI PL NZ NA PO NC
0F6C:0102 CD21      INT     21
-P
AX=2A02 BX=0000 CX=07CF DX=0811 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=0104  NU UP EI PL NZ NA PO NC
0F6C:0104 90        NOP
  
```

CS 3401 Comp. Org. & Assembly Executing Computer Instructions in 8086 35

Size of Memory

INT 12h

MS-DOS Prompt - DEBUG

```

C:\>DEBUG
-A 100
0F6C:0100 INT 12
0F6C:0102 NOP
0F6C:0103
-R
AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=0100  NU UP EI PL NZ NA PO NC
0F6C:0100 C012      INT     12
-P
AX=0280 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=0102  NU UP EI PL NZ NA PO NC
0F6C:0102 90        NOP
  
```

CS 3401 Comp. Org. & Assembly Executing Computer Instructions in 8086 36

Display with INT

```

C:\>DEBUG
-A 100
0F6C:0100 MOV AH,09
0F6C:0102 MOV DX,108
0F6C:0105 INT 21
0F6C:0107 NOP
0F6C:0108 DB "ONGARD", '$'
0F6C:010F
-B
AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=0100  NU UP EI PL NZ NA PO NC
0F6C:0100 6409      MOV     AH,09
-T
AX=0900 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=0102  NU UP EI PL NZ NA PO NC
0F6C:0102 B80801    MOV     DX,0108
-T
AX=0900 BX=0000 CX=0000 DX=0108 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=0105  NU UP EI PL NZ NA PO NC
0F6C:0105 CD21      INT     21
-P
ONGARD
AX=0924 BX=0000 CX=0000 DX=0108 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=0107  NU UP EI PL NZ NA PO NC
0F6C:0107 90          NOP

```

CS 3401 Comp. Org. & Assembly

Executing Computer Instructions in 8086

37

INT for Keyboard Input

```

C:\>DEBUG
-A 100
0F6C:0100 MOV AH,10
0F6C:0102 INT 16
0F6C:0104 JMP 100
0F6C:0106 NOP
0F6C:0107
-B
AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=0100  NU UP EI PL NZ NA PO NC
0F6C:0100 B410      MOV     AH,10
-T
AX=1000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=0102  NU UP EI PL NZ NA PO NC
0F6C:0102 CD16      INT     16
-P
AX=0231 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=0104  NU UP EI PL NZ NA PO NC
0F6C:0104 EBFA      JMP     0100

```

CS 3401 Comp. Org. & Assembly

Executing Computer Instructions in 8086

38

```

MS-DOS Prompt - DEBUG
Auto
AX=1000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=0102  NU UP EI PL NZ NA PO NC
0F6C:0102 CD16      INT     16
-P
AX=0231 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=0104  NU UP EI PL NZ NA PO NC
0F6C:0104 EBFA      JMP     0100
-T
AX=0231 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=0100  NU UP EI PL NZ NA PO NC
0F6C:0100 B410      MOV     AH,10
-T
AX=1031 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=0102  NU UP EI PL NZ NA PO NC
0F6C:0102 CD16      INT     16
-P
AX=0332 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0F6C ES=0F6C SS=0F6C CS=0F6C IP=0104  NU UP EI PL NZ NA PO NC
0F6C:0104 EBFA      JMP     0100

```

CS 3401 Comp. Org. & Assembly

Executing Computer Instructions in 8086

39