
上海交通大学
网络学院

微机原理与应用

The Principle & Application of Microcomputer

王春香　副教授

wangcx@sjtu.edu.cn

第四章 汇编语言程序设计

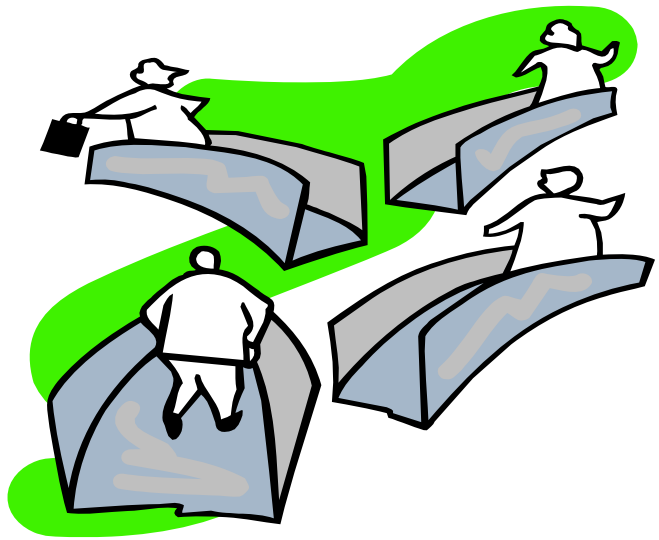
主要内容

4.1 程序设计语言概述

4.2 8086/8088汇编语言的基本语法

4.3 8086/8088汇编语言程序设计基本方法

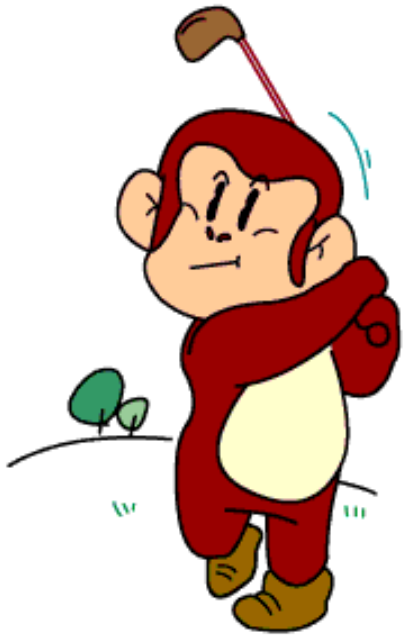
4.4 软件调试技术



4.4 软件调试技术

掌握软件技术不仅应具有很强的抽象能力、分析能力与综合能力，还应具有较强的调试能力及排错能力。

DEBUG调试软件，是分析、调试与排错的基本软件工具。



4.4 软件调试技术

```
STACK   SEGMENT STACK
    DB      200 DUP(0)
STACK   ENDS
DATA    SEGMENT
    BUF     DB 'Hello Everybody! $'
DATA    ENDS
CODE    SEGMENT
        ASSUME CS:CODE,DS:DATA,SS:STACK
BEGIN:  MOV   AX,DATA
        MOV   DS,AX
        LEA   DX,BUF
        MOV   AH,9
        INT   21H
        MOV   AH,4CH
        INT   21H
CODE  ENDS
END BEGIN
```

4.4 软件调试技术

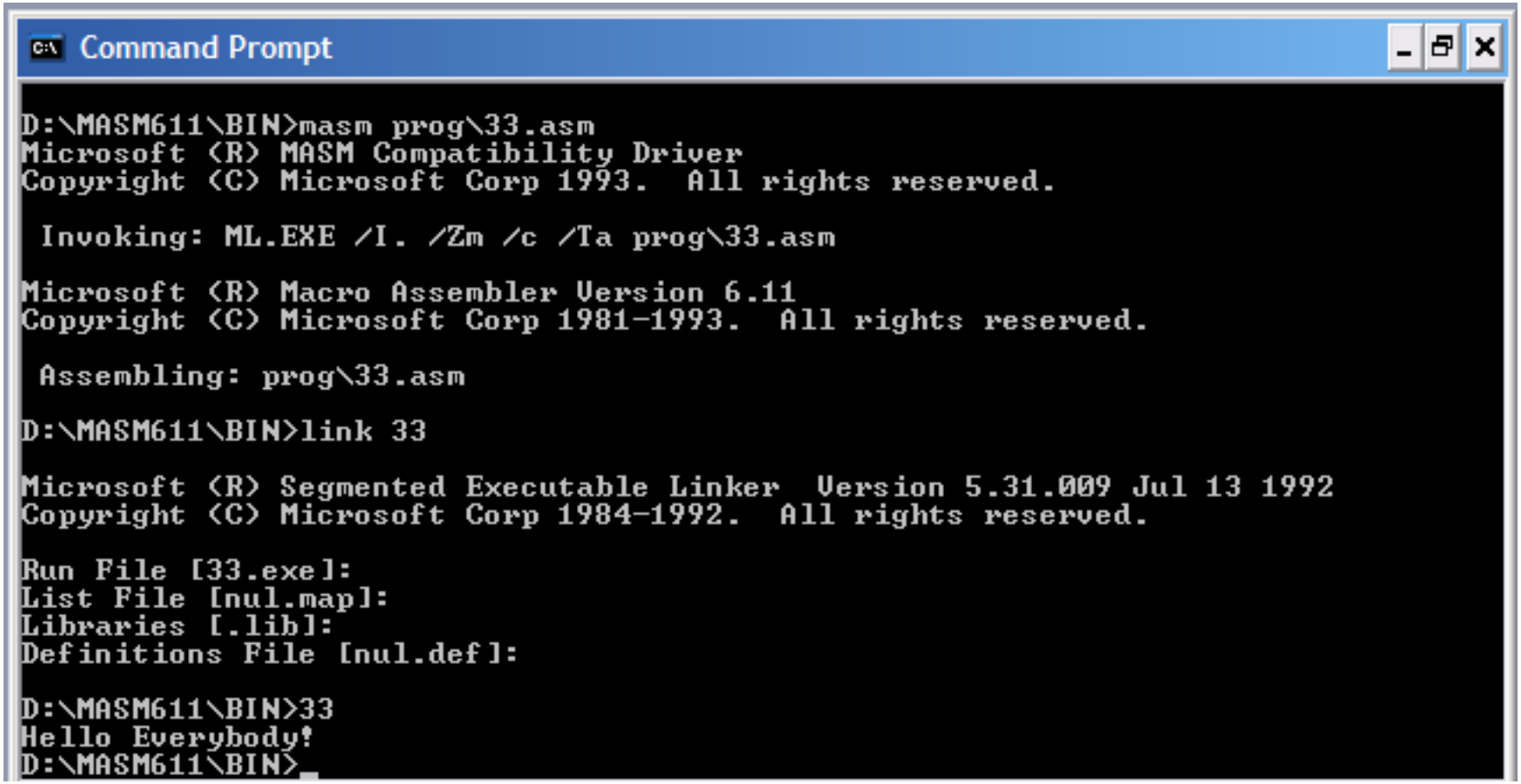
1. 利用记事本输入汇编源程序，然后将扩展名改为ASM。
   如，文件名为 33.asm

2. 利用宏汇编程序MASM对汇编源程序33.asm进行汇编，产生 33.obj文件。

3. 利用LINK对33.obj 进行连接，产生 33.exe文件。

4. 利用debug调试33.exe文件。

4.4 软件调试技术


Command Prompt
D:\MASM611\BIN>masm prog\33.asm
Microsoft (R) MASM Compatibility Driver
Copyright (C) Microsoft Corp 1993. All rights reserved.
Invoking: ML.EXE /I. /Zm /c /Ta prog\33.asm
Microsoft (R) Macro Assembler Version 6.11
Copyright (C) Microsoft Corp 1981-1993. All rights reserved.
Assembling: prog\33.asm
D:\MASM611\BIN>link 33
Microsoft (R) Segmented Executable Linker Version 5.31.009 Jul 13 1992
Copyright (C) Microsoft Corp 1984-1992. All rights reserved.
Run File [33.exe]:
List File [nul.map]:
Libraries [.lib]:
Definitions File [nul.def]:
D:\MASM611\BIN>33
Hello Everybody!
D:\MASM611\BIN>

4.4 软件调试技术

4.4.1 调试软件

DEBUG是DOS操作系统提供的一个调试汇编语言程序的工具软件，各种版本DOS都带有该程序，利用它可以动态地调试汇编语言程序。

检查与修改CPU各寄存器、标志位及内存单元内容；

输入汇编指令(或程序)到指定的内存，或把要调试的程序(通常以可执行的文件形式存储在磁盘上)**调入**到内存中；

控制CPU按单步、断点设置等方式**执行程序**；

检查程序运行过程中中间结果，以便查找程序出错原因等。

4.4 软件调试技术

4.4.1 调试软件

启动 DEBUG 的格式如下：

C〉DEBUG [d:][path][filename[.exe]][parm1] [parm2] ↵

C>　　— DOS下的提示符；

d:　　— DEBUG.EXE 程序所在的盘符；

Path　　— filename 的目录路径；

Filename —要分析或调试的二进制程序文件名；

exe　　—程序文件的扩展名；

parm1　— 被调试程序约定的第1参数文件名；

parm2　— 被调试程序约定的第2参数文件名。

4.4 软件调试技术

该命令是在DOS下把DEBUG.EXE程序调入内存，并运行DEBUG程序。

输入命令后，屏幕上将出现提示符“-”，表示当前已进入DEBUG的命令状态，可执行DEBUG程序的命令。

所有DEBUG命令均为单一字母，其后跟着一个或多个参数。命令中参数之间必须用空格或逗号分隔，每个命令都以回车键作结束符。

4.4 软件调试技术

在DEBUG命令中，使用的地址格式为：

段基值：偏移量

其中，段基值可以用段寄存器名（如CS，SS，DS，ES）表示，也可以是一个十六进制数。

在DEBUG状态下，命令参数中的数据和机器显示的数据都是十六进制数，而且不再以“H”结尾。

4.4 软件调试技术

4.4.1 调试软件

常用的DEBUG命令。

1. 显示存储单元内容：

格式1：D [起始地址]

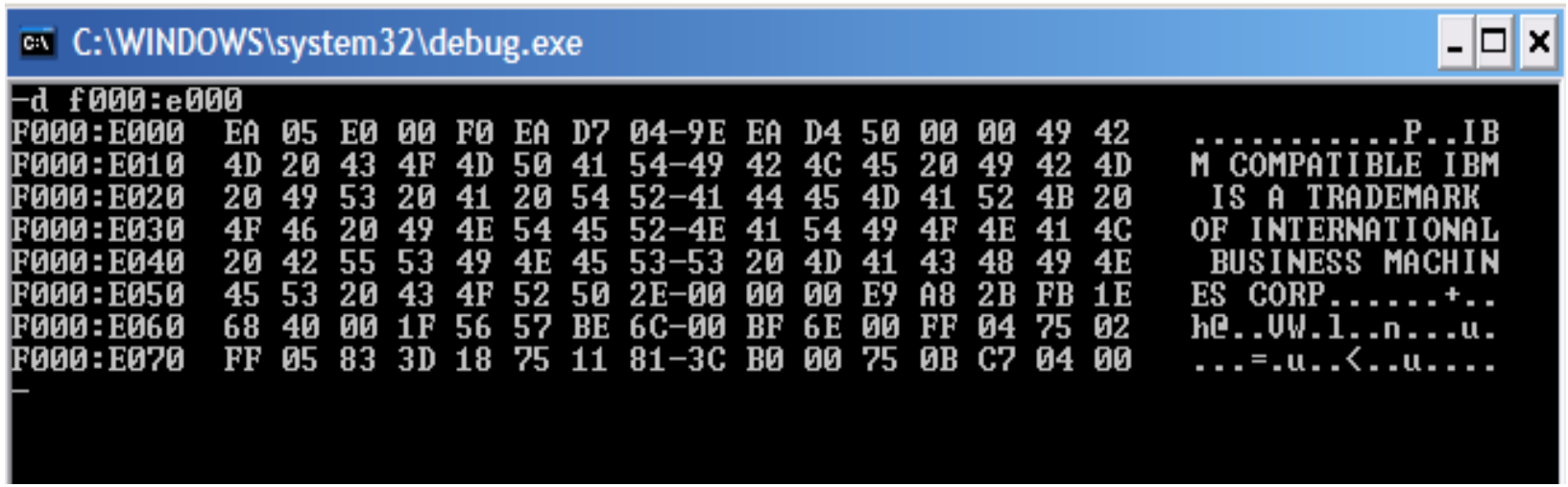
从起始地址开始按十六进制显示80个单元内容，每行16个单元。每行右侧还显示该16个单元的ASCII码字符。对于无字符对应的ASCII 码则显示“ ”。

格式2：D [地址范围]

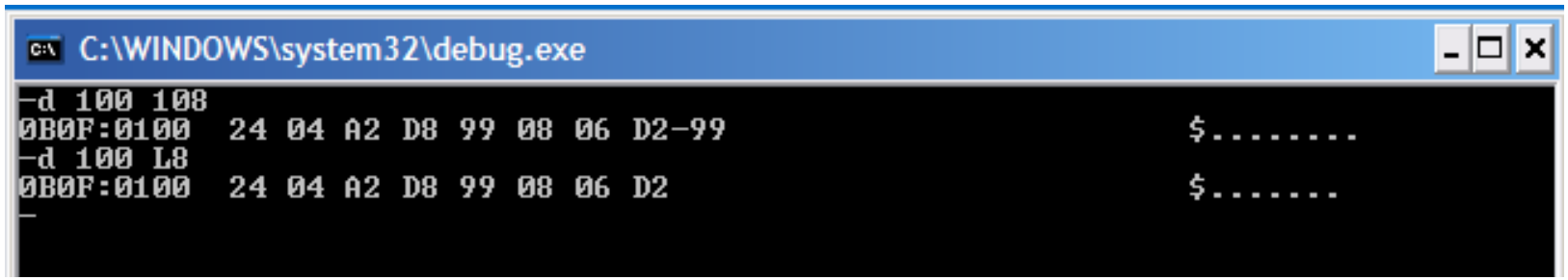
显示指定范围存储单元中内容，每行16个单元。每行右侧还显示该16个单元的ASCII码字符，无字符对应的ASCII则显示“ ”。如果不给出起始地址或地址范围，则从当前地址开始按格式 1 操作。

4.4 软件调试技术

例1： －D f000: e000



例2： －D 100 108（等价于- D 100 L8）



4.4 软件调试技术

4.4.1 调试软件

常用的DEBUG命令。

2. 修改存储单元内容

格式1： E 起始地址 ［列表］

按列表内容（包括字符或数值串）修改从起始地址开始的多个存储单元内容。

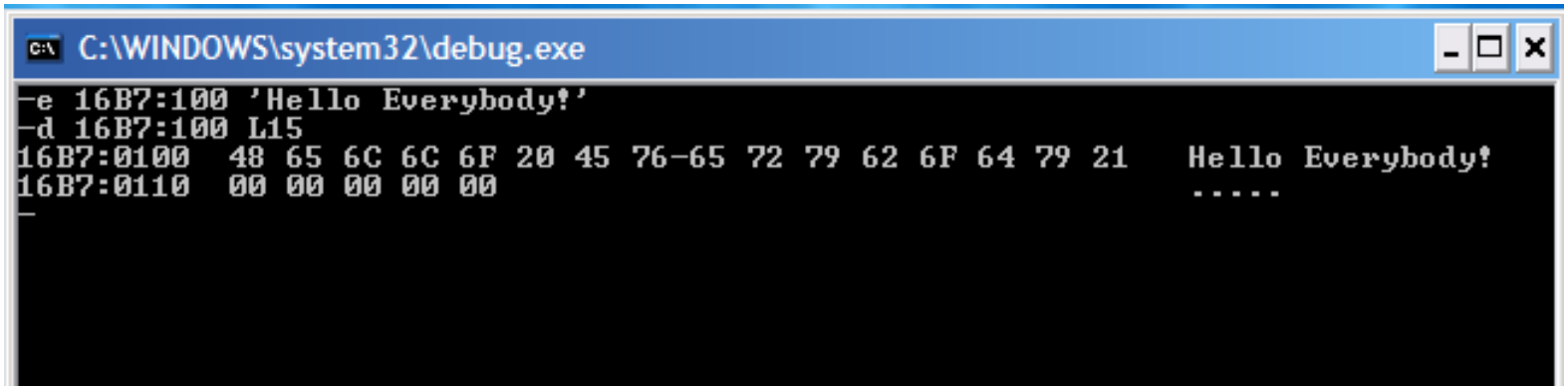
格式2： E 地址

用于逐个修改某指定地址单元内容。

4.4 软件调试技术

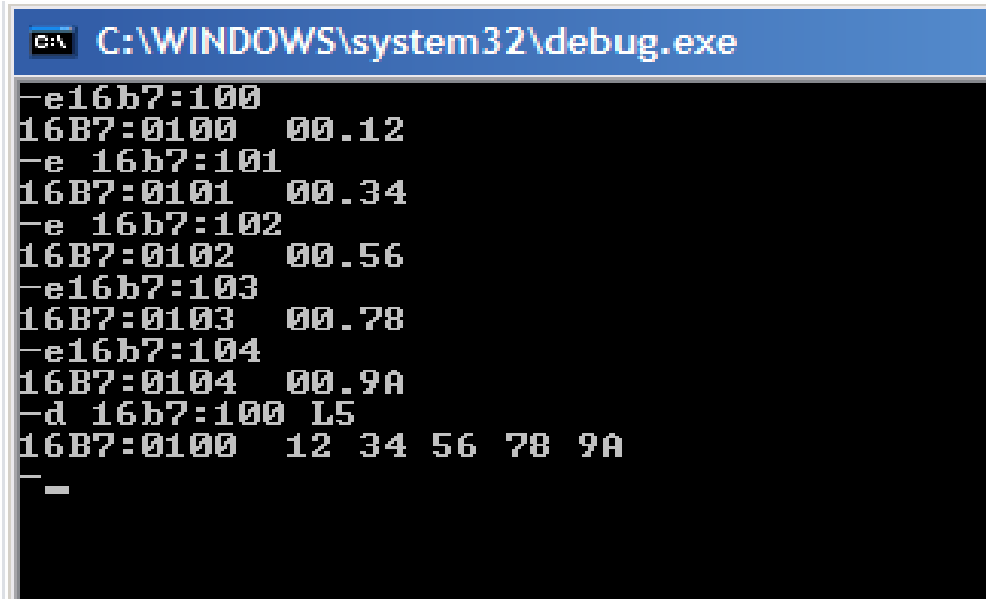
例3：用字符串 ‘Hello, Everybody’ 修改从 16B7: 100H 开始的15个内存单元内容。

- E 16B7:100 ‘Hello, Everybody’
- D 16B7:100 L15



4.4 软件调试技术

例4：用数值12H, 34H, 56H, 78H, 9AH修改 16B7: 0100H 开始的5个内存单元内容。



4.4 软件调试技术

4.4.1 调试软件

常用的DEBUG命令。

3. 显示、修改寄存器内容

格式：R［寄存器名］

如果指定了寄存器名，则显示寄存器的内容，并允许修改。
如果不指出寄存器名，则按如下格式显示CPU内部各寄存器（包括通用寄存器、段寄存器、标志寄存器）的内容。

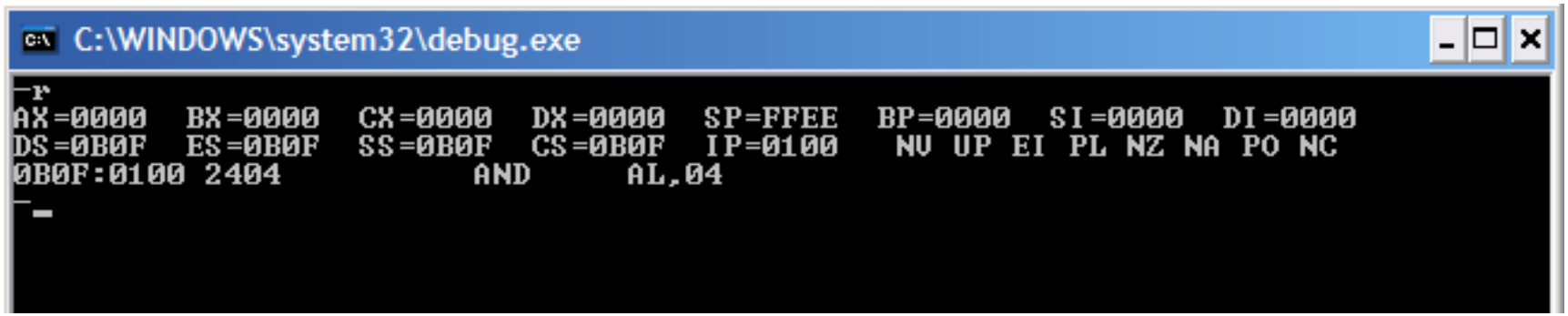
4.4 软件调试技术

■ 标志寄存器各标志位的显示字符

标志位		置位	复位
溢出位	OF	OV	NV
方向位	DF	DN	UP
中断位	IF	EI	DI
符号位	SF	NG	PL
零值位	ZF	ZR	NZ
辅助进位位	AF	AC	NA
奇偶位	PF	PE	PO
进位位	CF	CY	NC

4.4 软件调试技术

例5：-r ；显示CPU内所有寄存器内容和标志位状态



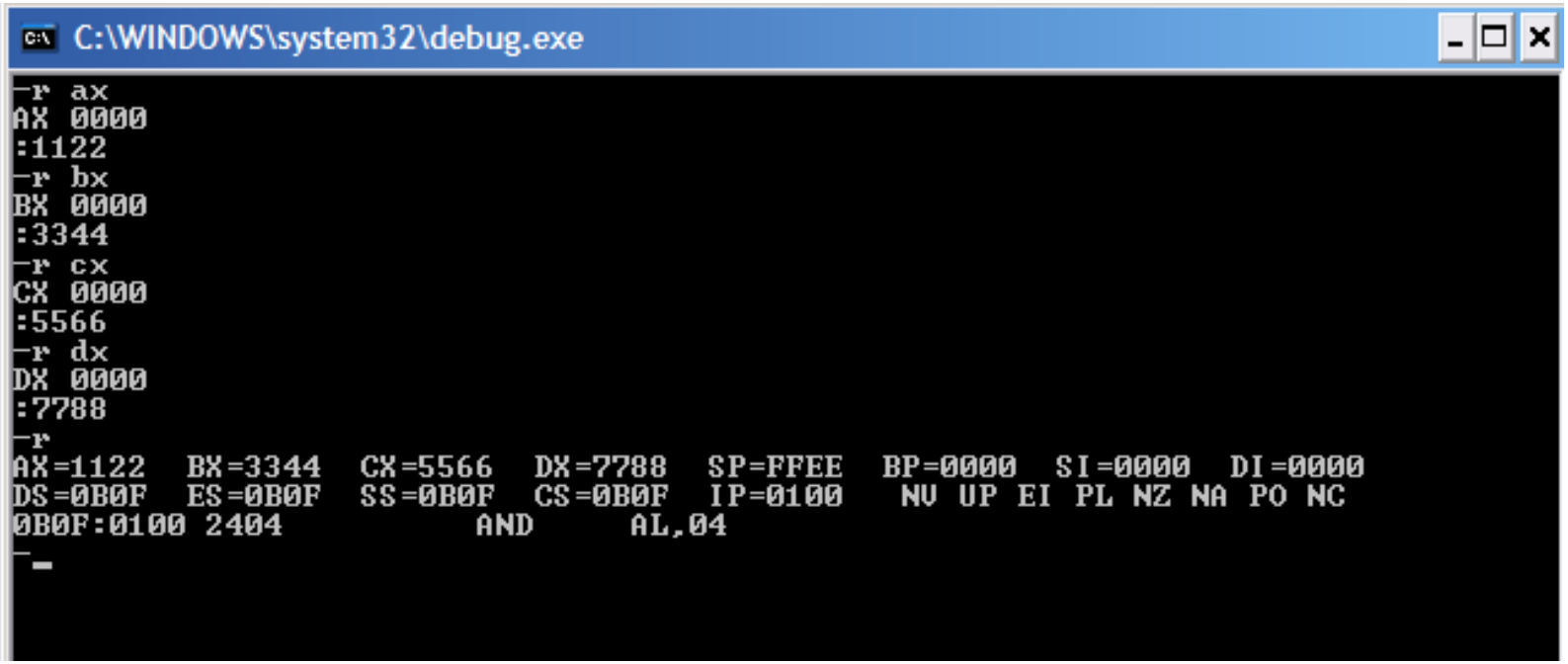
各通用寄存器显示了其中的内容，均以4位十六进制数形式来显示；

标志寄存器中各标志位（除TF外）的值均以符号形式显示；

CS:IP所指向的内存内容反汇编成一条指令，可视为将要执行的指令。

4.4 软件调试技术

例6：－r 寄存器名　；显示和修改某个寄存器内容



4.4 软件调试技术

例7： －ｒｆ　；显示和修改标志位状态

C:\WINDOWS\system32\debug.exe

```
-r ax
AX 0000
:1122
-r bx
BX 0000
:3344
-r cx
CX 0000
:5566
-r dx
DX 0000
:7788
-r
AX=1122  BX=3344  CX=5566  DX=7788  SP=FFEE  BP=0000  SI=0000  DI=0000
DS=0B0F  ES=0B0F  SS=0B0F  CS=0B0F  IP=0100   NV UP EI PL NZ NA PO NC
0B0F:0100 2404          AND     AL,04
-r f
NV UP EI PL NZ NA PO NC  -
-r f
NV UP EI PL NZ NA PO NC  -ponzdinv
-r
AX=1122  BX=3344  CX=5566  DX=7788  SP=FFEE  BP=0000  SI=0000  DI=0000
DS=0B0F  ES=0B0F  SS=0B0F  CS=0B0F  IP=0100   NV UP DI PL NZ NA PO NC
0B0F:0100 2404          AND     AL,04
-
```

4.4 软件调试技术

4.4.1 调试软件

常用的DEBUG命令。

4. 运行命令

格式：G [=起始地址] [第1断点地址 [第2断点地址…]]

功能：

CPU从指定起始地址开始执行，依次在第1、第2等断点处中断。

若命令格式中不给出起始地址，则从当前CS：IP指示的地址开始执行。

4.4 软件调试技术

当程序运行至其中任一断点时，便立即停下来，并显示CPU各寄存器内容和下一次将要执行的指令。

断点地址参数只对本次G命令有效，再次使用G命令时，仍需重新指定断点地址参数。

若不设断点地址参数，那么程序就运行至结束，并显示“Program terminated normally”（程序正常结束）。

4.4 软件调试技术

例8： －G=2180:100 116

C:\WINDOWS\system32\debug.exe

```
-r ax
AX 0000
:1122
-r bx
BX 0000
:2233
-g 2180:100 116

Program terminated normally
-g=2180:100 116

AX=1122  BX=2233  CX=0000  DX=0000  SP=FFEE  BP=0000  SI=0000  DI=0000
DS=0B0F  ES=0B0F  SS=0B0F  CS=2180  IP=0100   NV UP EI PL NZ NA PO NC
2180:0100 CC              INT     3
-
```

4.4 软件调试技术

4.4.1 调试软件

常用的DEBUG命令。

5. 跟踪命令

格式：T［=起始地址］［正整数］

从指定地址开始执行‘正整数’条指令。

如不给出‘正整数’，则按 1 处理；

如不给起始地址，则从当前CS：IP指示的地址开始执行。

4.4 软件调试技术

例9：- t=0b0f:100 3； 执行从0b0F:100H 开始的3条指令。

C:\WINDOWS\system32\debug.exe

```
0B0F:0102 A2D899        MOV     [99D8],AL
0B0F:0105 0806D299      OR      [99D2],AL
0B0F:0109 A0D899        MOV     AL,[99D8]
0B0F:010C 0AC0          OR      AL,AL
0B0F:010E C3            RET
0B0F:010F 803EB798FF    CMP     BYTE PTR [98B7],FF
0B0F:0114 750C          JNZ     0122
0B0F:0116 BFE194        MOV     DI,94E1
0B0F:0119 BDB798        MOV     BP,98B7
0B0F:011C 3400          XOR     AL,00
0B0F:011E FE0A          DEC     BYTE PTR [BP+SI]
-t=0b0f:0100 3

AX=0000  BX=0000  CX=0000  DX=0000  SP=FFEE  BP=0000  SI=0000  DI=0000
DS=0B0F  ES=0B0F  SS=0B0F  CS=0B0F  IP=0102   NV UP EI PL ZR NA PE NC
0B0F:0102 A2D899        MOV     [99D8],AL                       DS:99D8=00

AX=0000  BX=0000  CX=0000  DX=0000  SP=FFEE  BP=0000  SI=0000  DI=0000
DS=0B0F  ES=0B0F  SS=0B0F  CS=0B0F  IP=0105   NV UP EI PL ZR NA PE NC
0B0F:0105 0806D299      OR      [99D2],AL                       DS:99D2=00

AX=0000  BX=0000  CX=0000  DX=0000  SP=FFEE  BP=0000  SI=0000  DI=0000
DS=0B0F  ES=0B0F  SS=0B0F  CS=0B0F  IP=0109   NV UP EI PL ZR NA PE NC
0B0F:0109 A0D899        MOV     AL,[99D8]                       DS:99D8=00
-
```

4.4 软件调试技术

4.4.1 调试软件

常用的DEBUG命令。

6. 汇编命令

格式： A［起始地址］

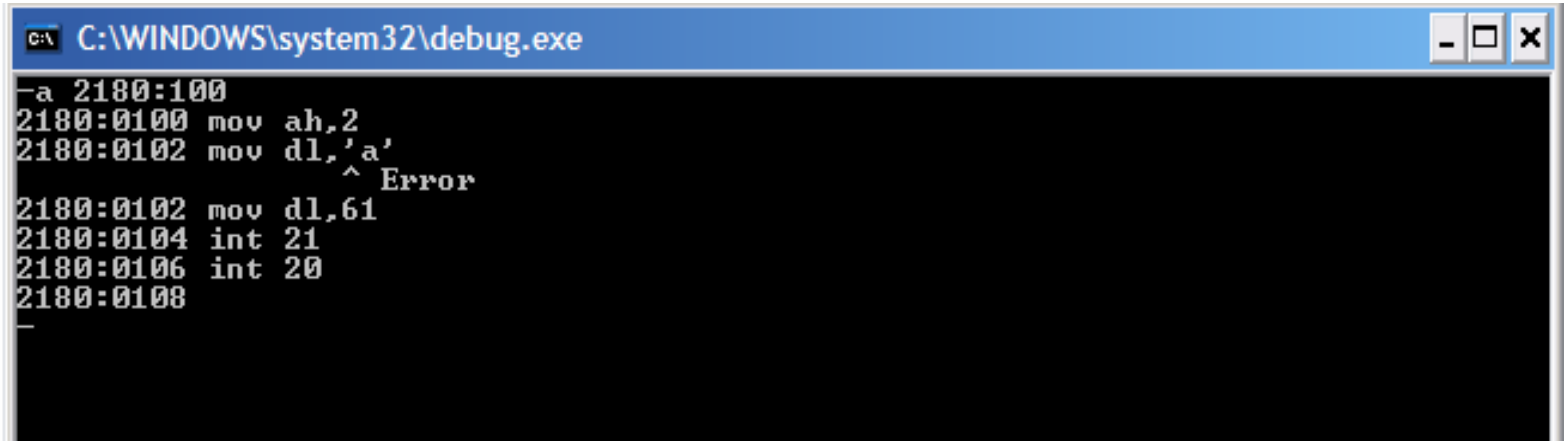
从指定地址开始接受汇编指令，并将汇编指令译成机器码，存入内存。

如不给出起始地址，则从当前地址开始接受，或从当前代码段的16进制100表示的相对地址处接受汇编指令。

如输入汇编指令过程中，在某行不作任何输入而直接回车，则结束A命令，回到接受命令状态“—”处。

4.4 软件调试技术

例10： 将利用DOS功能2 显示字符a的一段小程序汇编到2180: 100H开始的内存中。



由于A命令不支持字符方式，当在0102地址后输入指令 MOV DL, 'a' 后，显示出错信息，然后仍然提示当前汇编地址 2180：0102，此时可重新输入指令 MOV DL,61，其中61H为a的ASCII码。

4.4 软件调试技术

4.4.1 调试软件

常用的DEBUG命令。

7. 反汇编命令

格式1： U［起始地址］

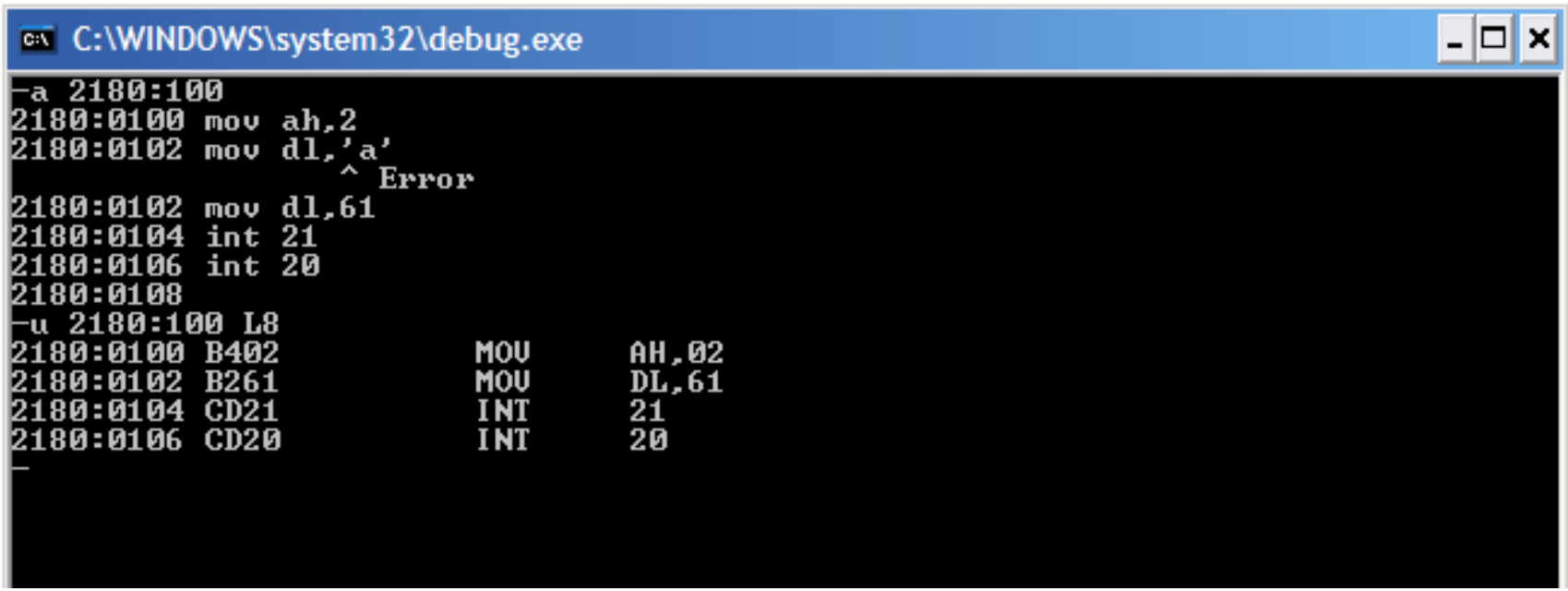
从指定起始地址处开始对32个字节内容转换成汇编指令形式，如不给出起始地址，则从当前地址开始。

格式2： U 地址范围

将指定范围内的存储内容转换成汇编指令。

4.4 软件调试技术

例11： 反汇编从2180:100开始长度为8个字节的部分程序段。



4.4 软件调试技术

4.4.1 调试软件

常用的DEBUG命令。

8. 指定文件名命令

格式：N文件名及扩展名

指出即将调入内存或从内存中存盘的文件名。
这条命令要配合L或W命令一起使用。

4.4 软件调试技术

4.4.1 调试软件

常用的DEBUG命令。

9. 装入命令

格式1： L 起始地址 驱动器号 起始扇区 扇区数

根据指定驱动器号（0表示A驱，1表示B驱，2表示C驱），指定起始逻辑扇区号和扇区数将相应扇区内容装入到指定起始地址的存储区中。

格式2： L［起始地址］

将N命令指出的文件装入到指定起始地址的存储区中，若没有指定起始地址，则装入到CS:100 处或按原来文件定位约定装入到相应位置。

4.4 软件调试技术

4.4.1 调试软件

常用的DEBUG命令。

10. 写磁盘命令

格式1：W起始地址 驱动器号 起始扇区 扇区数

与L命令格式1的功能正好相反，将程序由指定的存储区写入指定驱动器中指定的扇区或文件中。

格式2：W［起始地址］

将起始地址开始的BX*10000H+CX个字节内容存放到由N命令指定的文件中。

执行这条命令前注意给BX、CX中置上恰当的值。

4.4 软件调试技术

例12： 装入文件33.exe。

C:\WINDOWS\system32\debug.exe

```
-n d:\masm611\bin\33.exe
-l
-u
0B79:0000 B8770B        MOV     AX,0B77
0B79:0003 8ED8          MOV     DS,AX
0B79:0005 8D160000      LEA     DX,[0000]
0B79:0009 B409          MOV     AH,09
0B79:000B CD21          INT     21
0B79:000D B44C          MOV     AH,4C
0B79:000F CD21          INT     21
0B79:0011 26            ES:
0B79:0012 8A4701        MOV     AL,[BX+01]
0B79:0015 32E4          XOR     AH,AH
0B79:0017 40            INC     AX
0B79:0018 D1E0          SHL     AX,1
0B79:001A 03D8          ADD     BX,AX
0B79:001C 26            ES:
0B79:001D 8A07          MOV     AL,[BX]
0B79:001F 32E4          XOR     AH,AH
-
```

4.4 软件调试技术

4.4.1 调试软件

常用的DEBUG命令。

11. 退出命令

格式： Q

退出DEBUG，返回到操作系统。

执行这条命令时不存盘，如需要存盘应先用W命令，然后再退出。

4.4 软件调试技术

4.4.1 调试软件

以上是常用的DEBUG命令，此外，还有一般命令：

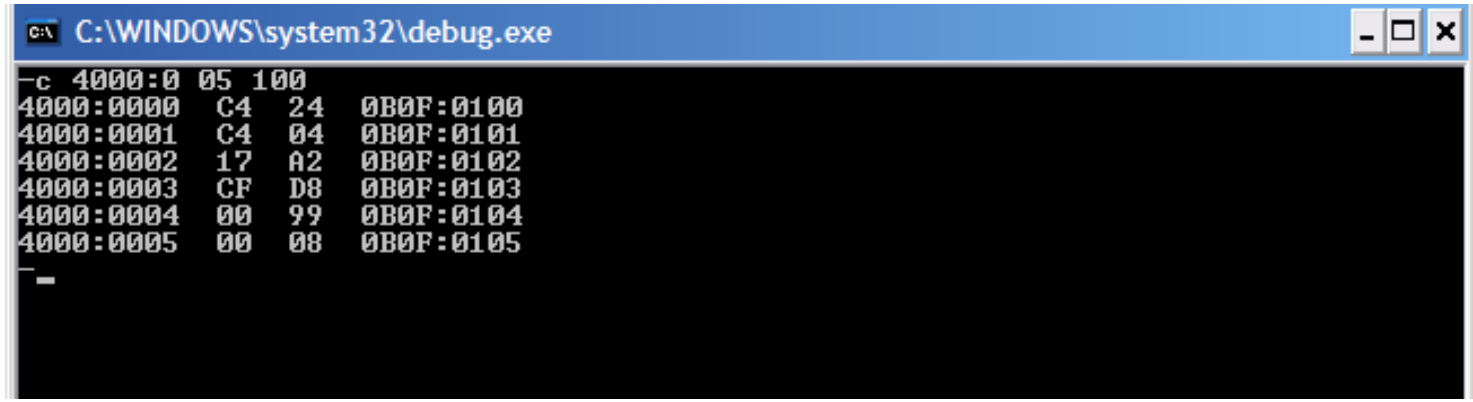
12. 比较命令

格式： C 源地址范围 目标起始地址

比较两段内存区的内容，并列出相异式。

4.4 软件调试技术

例13：比较4000:0 05 100 内容的差异。



0B0F为DS段寄存器内容

4.4 软件调试技术

4.4.1 调试软件

以上是常用的DEBUG命令，此外，还有一般命令：

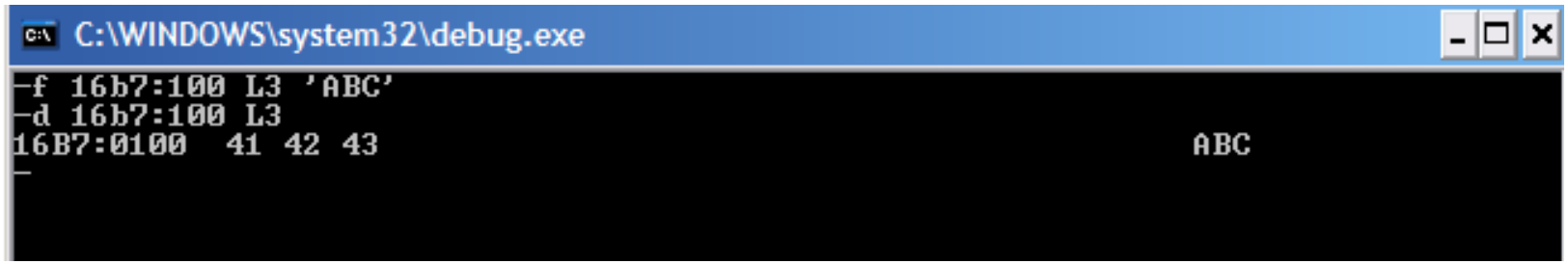
13. 填充命令

格式： F 地址范围 要填入的字节或字符串

将要填充的数据填入到指定的地址范围。

4.4 软件调试技术

例14： 将ABC字符串的ASCII码填入到16B7:0100开始的3个单元中。



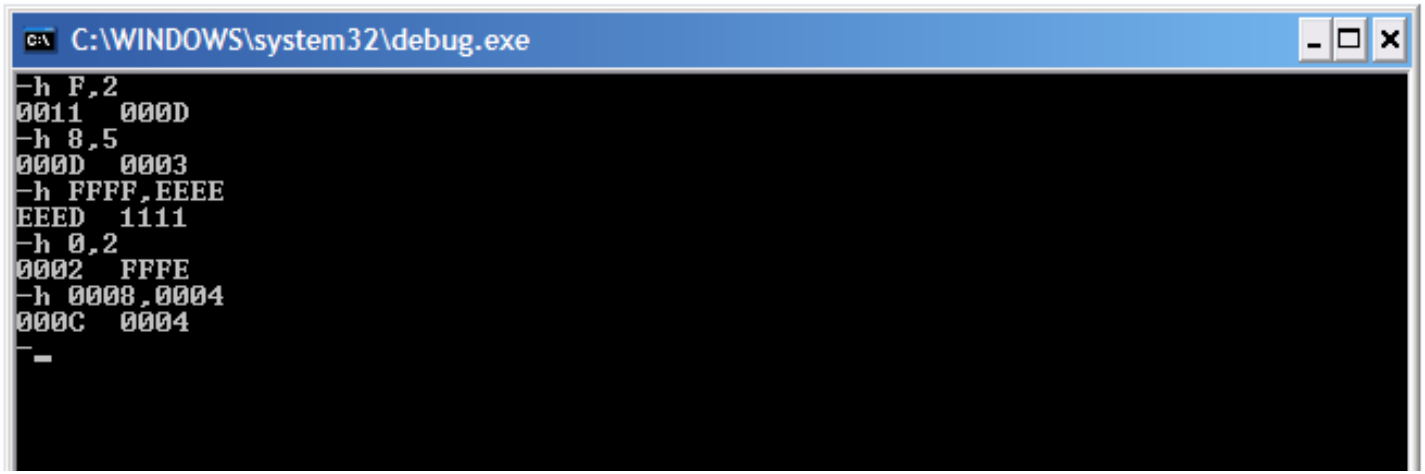
4.4 软件调试技术

4.4.1 调试软件

以上是常用的DEBUG命令，此外，还有一般命令：

14. 计算十六进制的和与差

格式：H数1，数2



4.4 软件调试技术

4.4.1 调试软件

以上是常用的DEBUG命令，此外，还有一般命令：

15. 从指定端口输入并显示

格式：端口地址

16. 移动存储器内容

格式：M源地址范围　目标起始地址

4.4 软件调试技术

4.4.1 调试软件

以上是常用的DEBUG命令，此外，还有一般命令：

17. 向指定端口输出字节

格式： O 端口地址

18. 搜索字符或字符串

格式： S 地址范围 要搜索的字节或字节串

4.4 软件调试技术

4.4.2 调试软件基本方法

利用调试软件 DEBUG 装入二进制执行程序，通过连续运行、分段运行、单步运行，可以实现对软件剖析、查错或修改。

将COM文件装入后，指令指针IP设置成十六进制的100，即为程序入口的相对地址。

首先从此处开始连续运行，考察程序的功能是否达到。如果出错，则可用分段运行方式，缩小错误所在程序段的范围，然后，再用单步方式找出错误确切所在处。

4.4 软件调试技术

C:\WINDOWS\system32\debug.exe

```
-n d:\masm611\bin\33.exe
-l
-t

AX=0B77  BX=0000  CX=0101  DX=0000  SP=00C8  BP=0000  SI=0000  DI=0000
DS=0B5A  ES=0B5A  SS=0B6A  CS=0B79  IP=0003   NV UP EI PL NZ NA PO NC
0B79:0003 8ED8          MOV     DS,AX
-t

AX=0B77  BX=0000  CX=0101  DX=0000  SP=00C8  BP=0000  SI=0000  DI=0000
DS=0B77  ES=0B5A  SS=0B6A  CS=0B79  IP=0005   NV UP EI PL NZ NA PO NC
0B79:0005 8D160000      LEA     DX,[0000]                     DS:0000=6548
-t

AX=0B77  BX=0000  CX=0101  DX=0000  SP=00C8  BP=0000  SI=0000  DI=0000
DS=0B77  ES=0B5A  SS=0B6A  CS=0B79  IP=0009   NV UP EI PL NZ NA PO NC
0B79:0009 B409          MOV     AH,09
-g
Hello Everybody!
Program terminated normally
-
```


Question？
第四章结束！