

# 汇编语言程序设计

## *Assembly Language Programming*

### DEBUG的使用



# 进入Debug

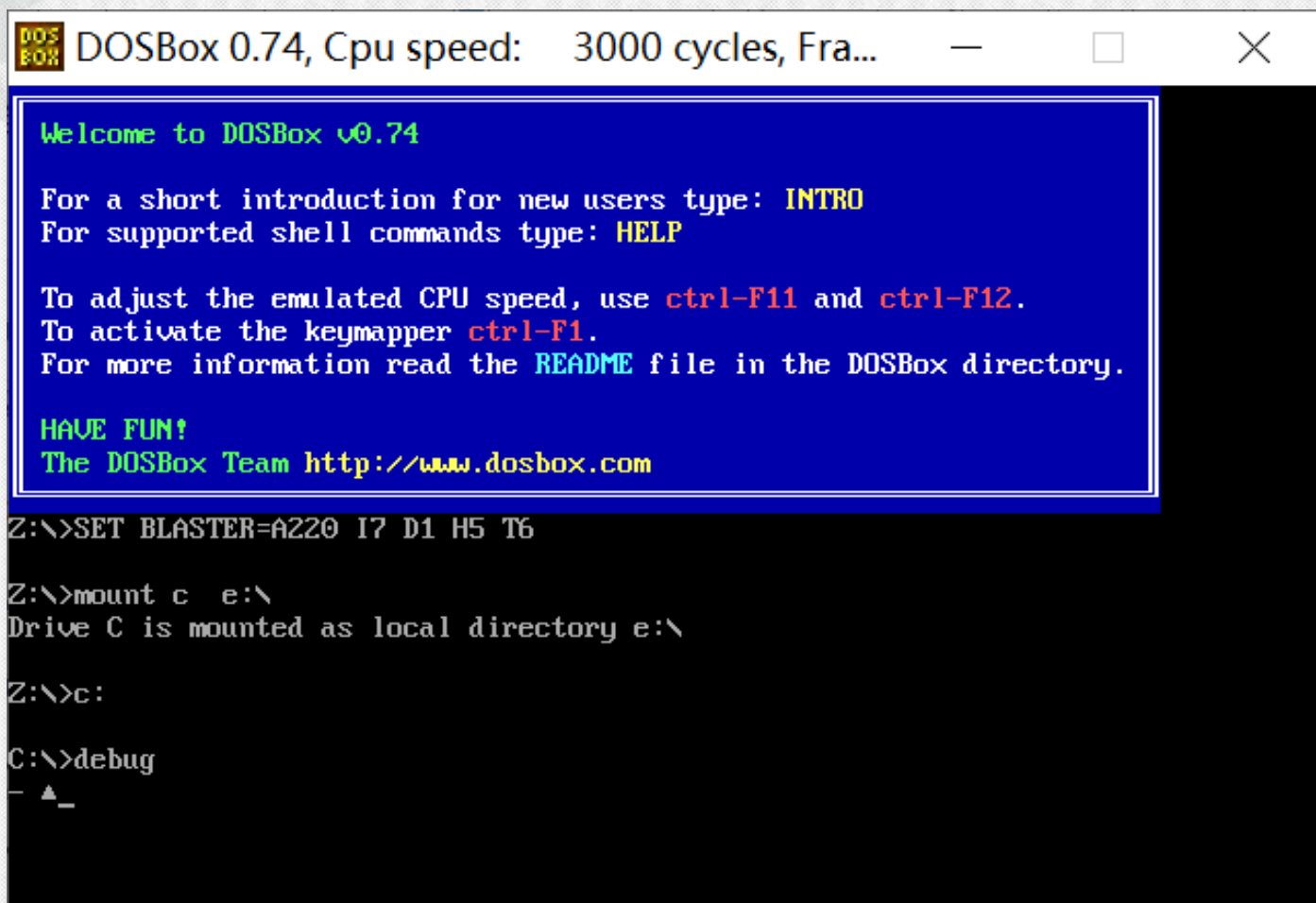
♥ 安装DOSBOX

♥ 将Debug、Masm和Link文件拷贝到某个文件夹下，如E盘

♥ 双击运行DOSBOX

- ❖ mount c e:\ 虚拟C盘
- ❖ C: 进入C盘
- ❖ DEBUG 进入DEBUG

# 进入Debug环境



DOSBox 0.74, Cpu speed: 3000 cycles, Fra... — □ ×

```
Welcome to DOSBox v0.74

For a short introduction for new users type: INTRO
For supported shell commands type: HELP

To adjust the emulated CPU speed, use ctrl-F11 and ctrl-F12.
To activate the keymapper ctrl-F1.
For more information read the README file in the DOSBox directory.

HAVE FUN!
The DOSBox Team http://www.dosbox.com
```

```
Z:\>SET BLASTER=A220 I7 D1 H5 T6

Z:\>mount c e:\
Drive C is mounted as local directory e:\

Z:\>c:

C:\>debug
- ▲ _
```



# Debug调试环境

- ♥ 大小写不敏感：不区分
- ♥ 只有16进制数：-1直接为FF
- ♥ 以空格或逗号作为命令各项之间的分隔符
- ♥ 个别指令不支持：SAL，同SHL
- ♥ 跳转指令使用：没有编号，直接使用偏移地址
- ♥ 书上P328



# 命令行

## ♥ 程序调用命令

❖ DEBUG [D: \PATH\FILENAME.EXE]  
[PARM1] [PARM2]

## ♥ Debug 命令的参数

### ❖ 地址

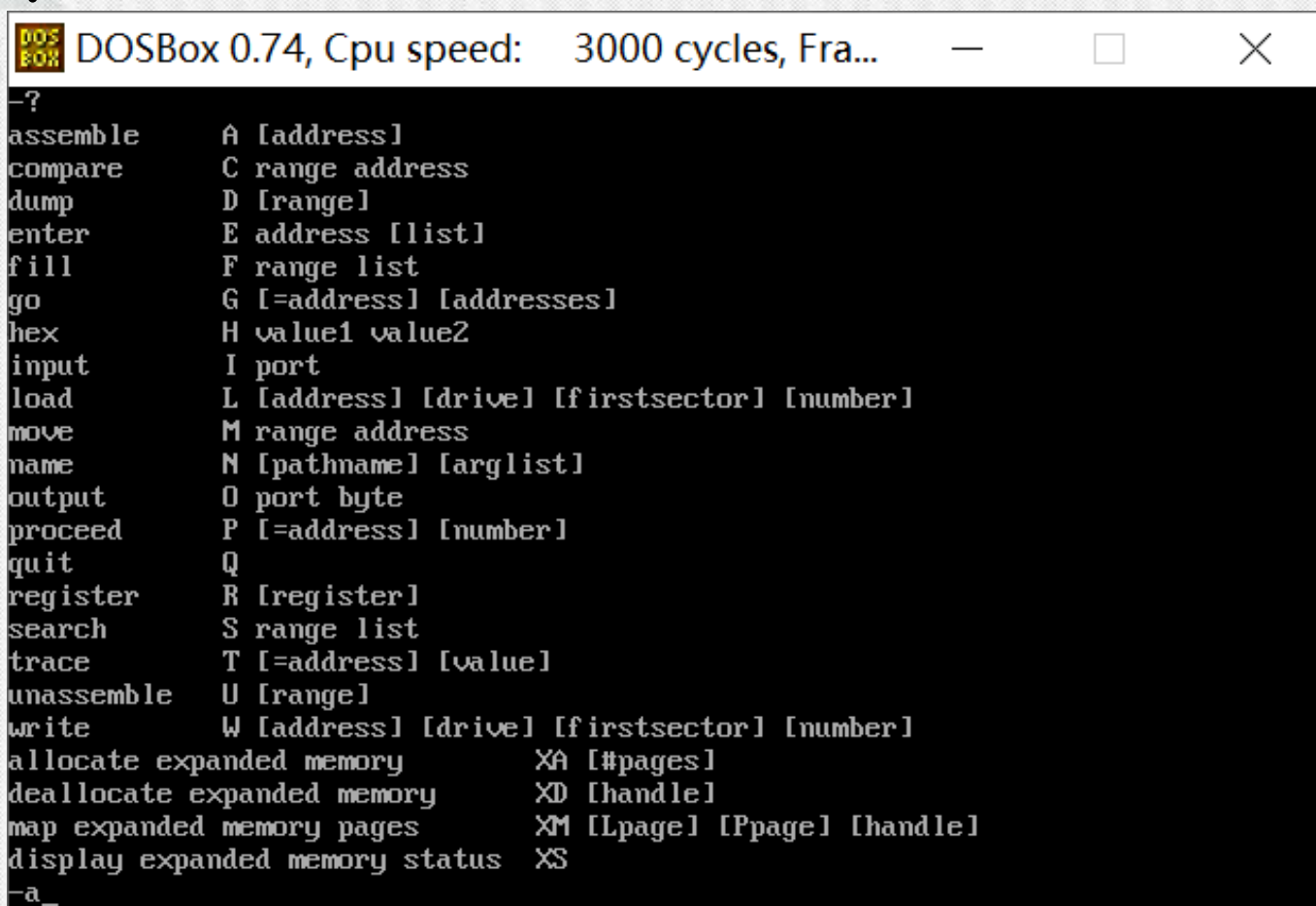
- 段地址：偏移地址
- 偏移地址

### ❖ 地址范围

- 开始地址 结束地址
- 开始地址 L 长度

♥帮助:

❖ ?

A screenshot of a DOSBox 0.74 window showing a list of commands and their syntax. The window title is "DOSBox 0.74, Cpu speed: 3000 cycles, Fra...". The list includes commands like assemble, compare, dump, enter, fill, go, hex, input, load, move, name, output, proceed, quit, register, search, trace, unassemble, write, allocate expanded memory, deallocate expanded memory, map expanded memory pages, and display expanded memory status.

```
-?
assemble      A [address]
compare       C range address
dump          D [range]
enter         E address [list]
fill          F range list
go            G [=address] [addresses]
hex           H value1 value2
input         I port
load          L [address] [drive] [firstsector] [number]
move          M range address
name          N [pathname] [arglist]
output        O port byte
proceed       P [=address] [number]
quit          Q
register       R [register]
search        S range list
trace         T [=address] [value]
unassemble    U [range]
write         W [address] [drive] [firstsector] [number]
allocate expanded memory    XA [#pages]
deallocate expanded memory  XD [handle]
map expanded memory pages   XM [Lpage] [Ppage] [handle]
display expanded memory status XS
-a_
```



## ♥ 显示存储单元命令

- ❖ -D [ADDRESS] 或 [RANGE]
- ❖ 默认：前面没用过D命令，则地址为CS:IP，否则从前一个D结束地址显示。

DOSBox 0.74, Cpu speed: 3000 cycles, Fra... — □ ×

To adjust the emulated CPU speed, use **ctrl-F11** and **ctrl-F12**.  
To activate the keymapper **ctrl-F1**.  
For more information read the **README** file in the DOSBox directory.

**HAVE FUN!**  
The DOSBox Team <http://www.dosbox.com>

```
Z:\>SET BLASTER=A220 I7 D1 H5 T6

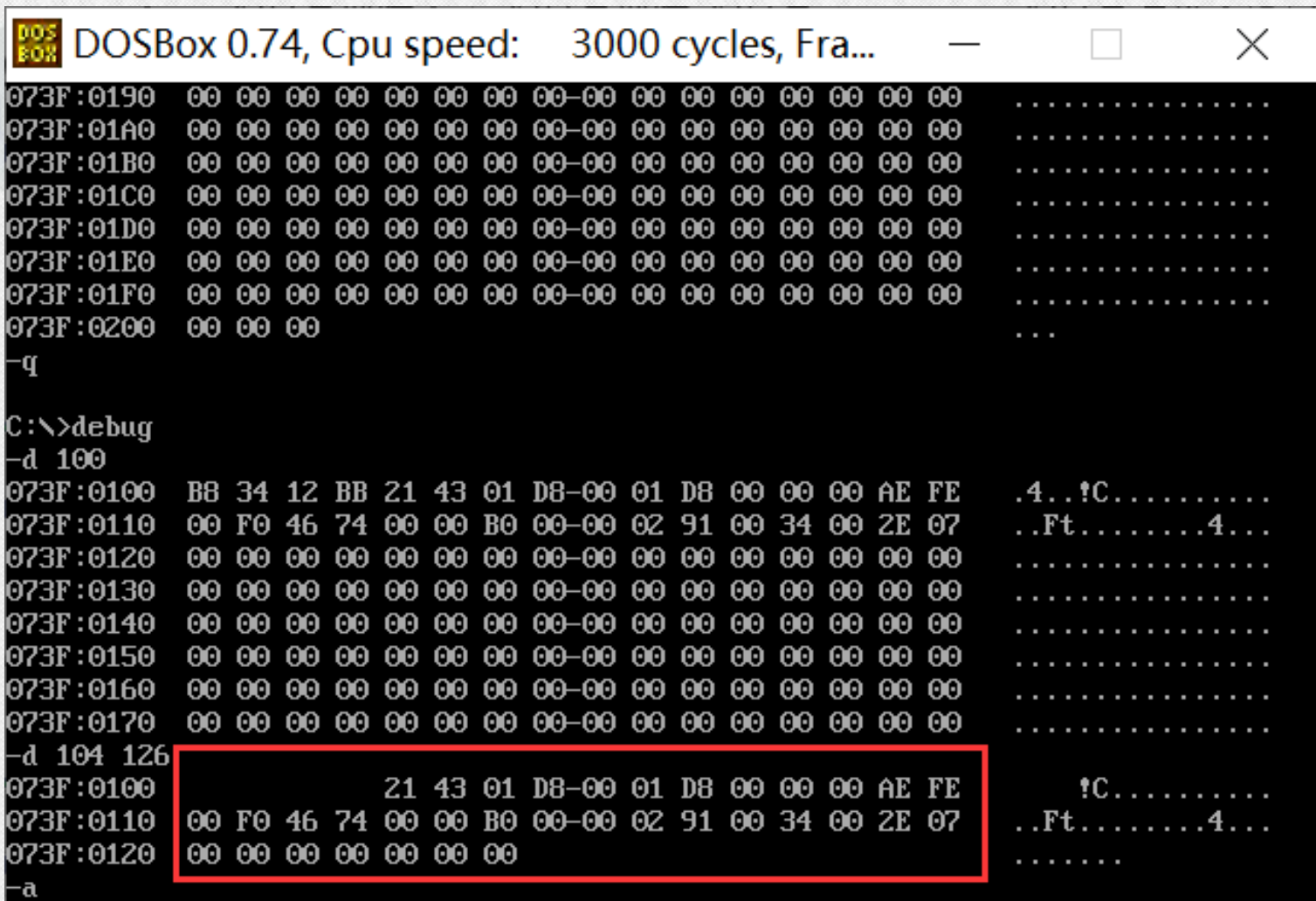
Z:\>mount c e:\
Drive C is mounted as local directory e:\

Z:\>c:

C:\>debug
-d
073F:0100  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00
073F:0110  00 00 00 00 00 00 00 00-00 00 00 00 34 00 2E 07
073F:0120  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00
073F:0130  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00
073F:0140  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00
073F:0150  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00
073F:0160  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00
073F:0170  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00
-a_
```

.....4.....

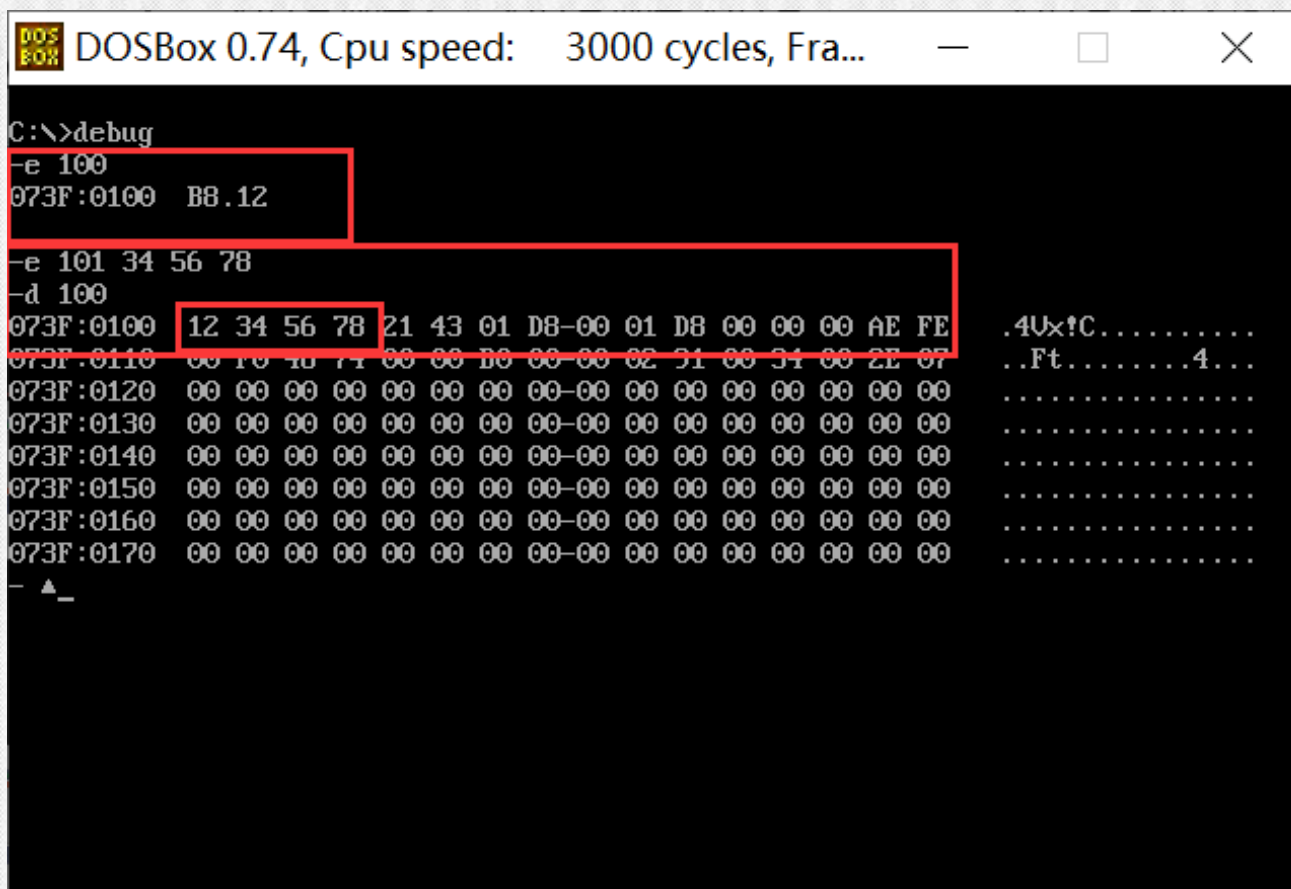




```
DOSBox 0.74, Cpu speed: 3000 cycles, Fra...  
073F:0190 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....  
073F:01A0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....  
073F:01B0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....  
073F:01C0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....  
073F:01D0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....  
073F:01E0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....  
073F:01F0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....  
073F:0200 00 00 00 .....  
-q  
C:\>debug  
-d 100  
073F:0100 B8 34 12 BB 21 43 01 D8-00 01 D8 00 00 00 AE FE .4...!C.....  
073F:0110 00 F0 46 74 00 00 B0 00-00 02 91 00 34 00 2E 07 ..Ft.....4...  
073F:0120 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....  
073F:0130 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....  
073F:0140 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....  
073F:0150 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....  
073F:0160 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....  
073F:0170 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....  
-d 104 126  
073F:0100 21 43 01 D8-00 01 D8 00 00 00 AE FE !C.....  
073F:0110 00 F0 46 74 00 00 B0 00-00 02 91 00 34 00 2E 07 ..Ft.....4...  
073F:0120 00 00 00 00 00 00 00 .....  
-a
```

## ♥ 修改存储单元内容命令

❖ -E ADDRESS [LIST] ;



```
DOS BOX DOSBox 0.74, Cpu speed: 3000 cycles, Fra...
C:\>debug
-e 100
073F:0100  B8.12
-e 101 34 56 78
-d 100
073F:0100  12 34 56 78 21 43 01 D8-00 01 D8 00 00 00 AE FE .4Ux!C.....
073F:0110  00 10 40 70 00 00 B0 00-00 02 31 00 34 00 2E 07 ..Ft.....4...
073F:0120  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....
073F:0130  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....
073F:0140  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....
073F:0150  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....
073F:0160  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....
073F:0170  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....
- _
```



## ♥ 修改存储单元内容命令

❖ -E ds:100 "This is the text example" 01 02 03

```

DOS
BOX
DOSBox 0.74, Cpu speed: 3000 cycles, Fra...
Drive C is mounted as local directory e:\
Z:\>c:
C:\>debug
-d
073F:0100  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  .....
073F:0110  00 00 00 00 00 00 00 00-00 00 00 00 34 00 2E 07  .....4...
073F:0120  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  .....
073F:0130  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  .....
073F:0140  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  .....
073F:0150  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  .....
073F:0160  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  .....
073F:0170  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  .....
-e ds:100 "This is the text example" 01 02 03
-d ds:100
073F:0100  54 68 69 73 20 69 73 20-74 68 65 20 74 65 78 74  This is the text
073F:0110  20 65 78 61 6D 70 6C 65-01 02 03 00 34 00 2E 07  example....4...
073F:0120  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  .....
073F:0130  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  .....
073F:0140  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  .....
073F:0150  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  .....
073F:0160  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  .....
073F:0170  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  .....
-a

```

## ♥ 检查和修改寄存器内容命令

❖ -R [REGISTER NAME] ; Register name 寄存器名字

```

DOSBox 0.74, Cpu speed: 3000 cycles, Fra...
C:\>debug
-r
AX=0000 BX=0000 CX=0000 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=073F IP=0100 NU UP EI PL NZ NA PO NC
073F:0100 54 PUSH SP
-r ax
AX 0000
:1234
-r
AX=1234 BX=0000 CX=0000 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=073F IP=0100 NU UP EI PL NZ NA PO NC
073F:0100 54 PUSH SP
-d

```

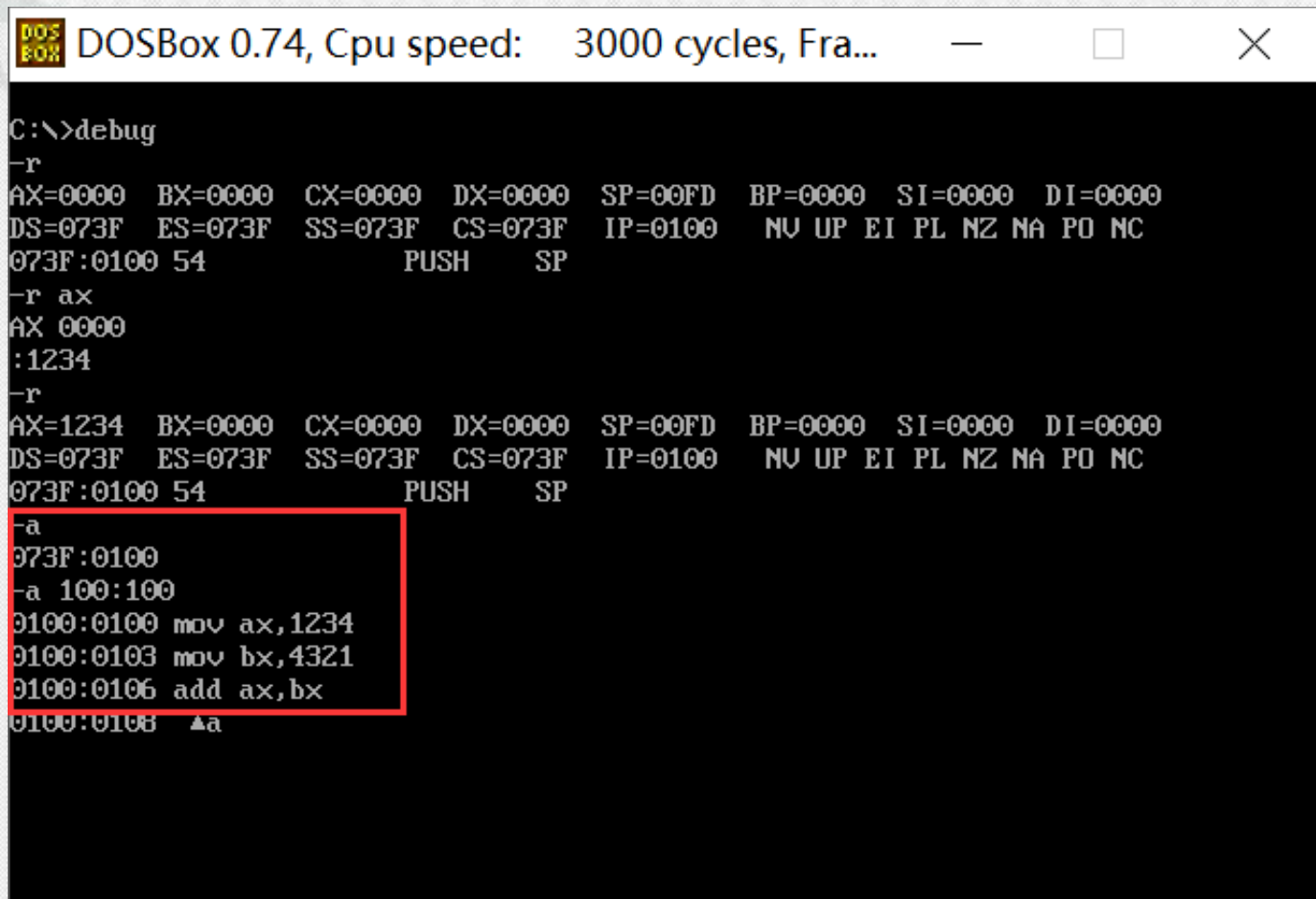
DEBUG调试中的标志表示

OF	溢出(是/否)	OV	NV
DF	方向(减/增)	DN	UP
IF	中断(允许/关闭)	EI	DI
SF	符号(负/正)	NG	PL
ZF	零(是/否)	ZR	NZ
AF	辅助进位(有/无)	AC	NA
PF	奇偶(偶/奇)	PE	PO
CF	进位(是/否)	CY	NC



# ♥ 汇编命令

## ❖ -A [ADDRESS]

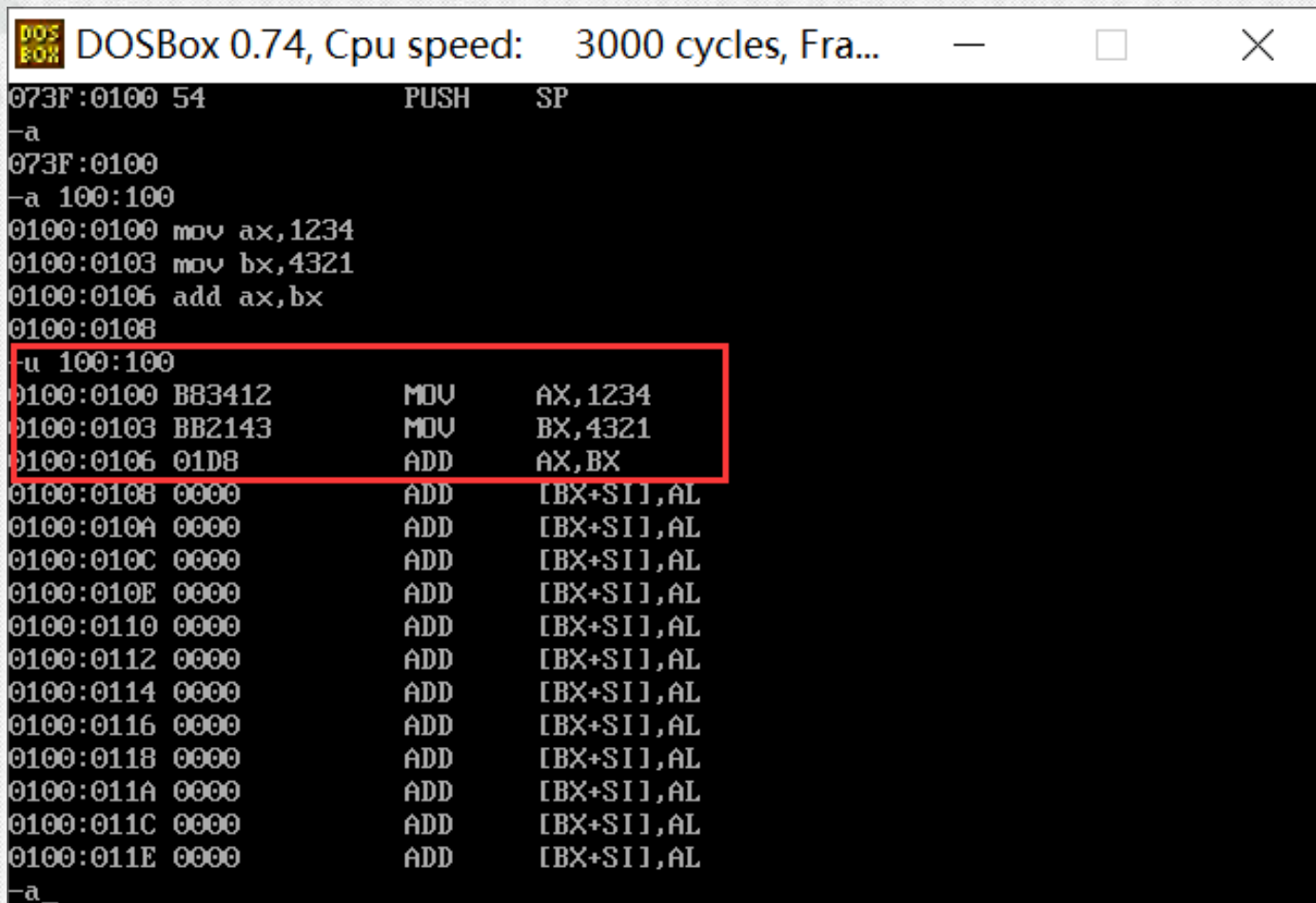


DOSBox 0.74, Cpu speed: 3000 cycles, Fra... — □ ×

```
C:\>debug
-r
AX=0000 BX=0000 CX=0000 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=073F IP=0100  NU UP EI PL NZ NA PO NC
073F:0100 54          PUSH    SP
-r ax
AX 0000
:1234
-r
AX=1234 BX=0000 CX=0000 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=073F IP=0100  NU UP EI PL NZ NA PO NC
073F:0100 54          PUSH    SP
-a
073F:0100
-a 100:100
0100:0100 mov ax,1234
0100:0103 mov bx,4321
0100:0106 add ax,bx
0100:0108 ▲a
```

## ♥ 反汇编命令

❖ -U [ADDRESS]或[RANGE]



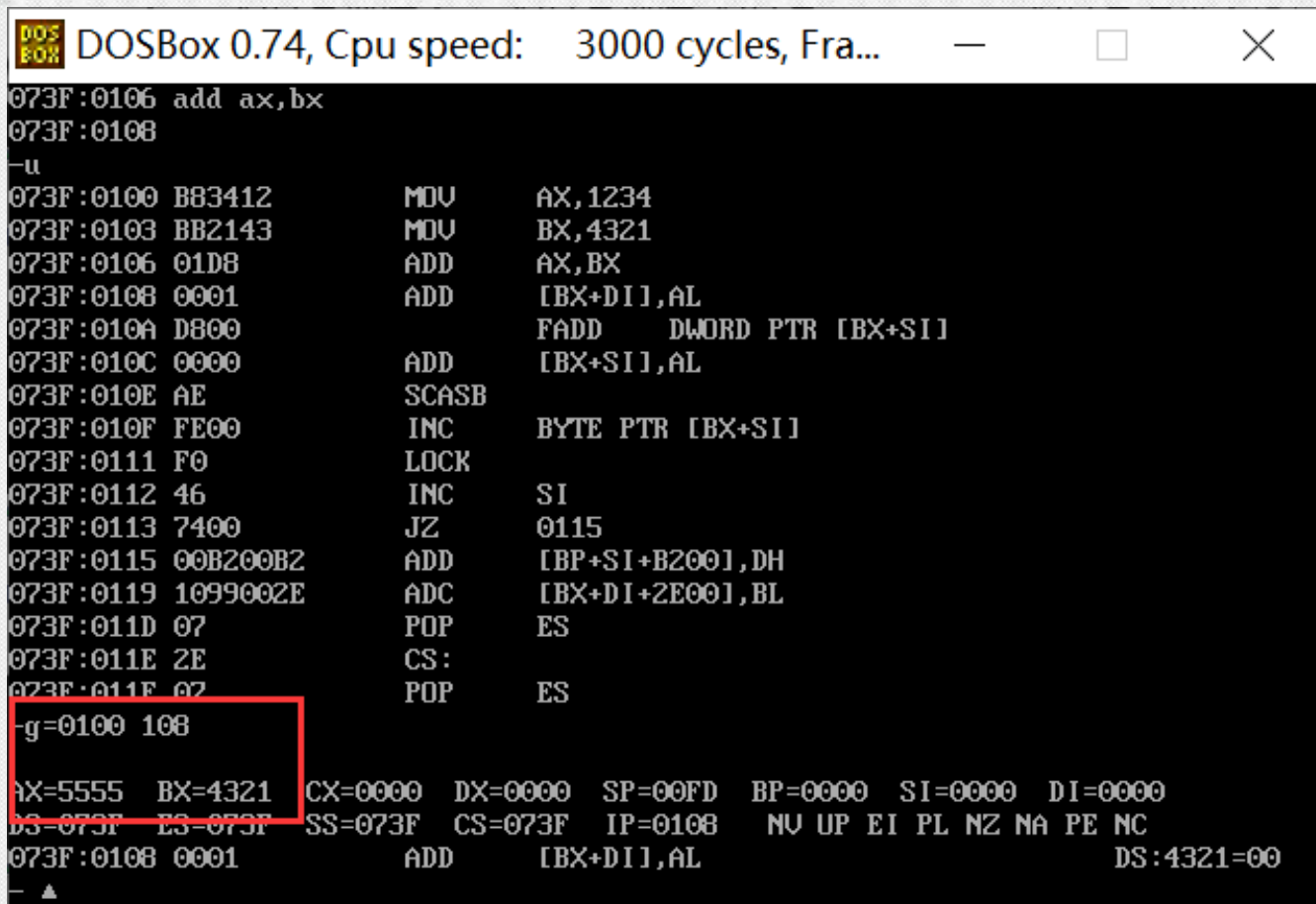
DOSBox 0.74, Cpu speed: 3000 cycles, Fra...

```
073F:0100 54          PUSH    SP
-a
073F:0100
-a 100:100
0100:0100 mov ax,1234
0100:0103 mov bx,4321
0100:0106 add ax,bx
0100:0108
-u 100:100
0100:0100 B83412      MOV     AX,1234
0100:0103 BB2143      MOV     BX,4321
0100:0106 01D8        ADD     AX,BX
0100:0108 0000        ADD     [BX+SI],AL
0100:010A 0000        ADD     [BX+SI],AL
0100:010C 0000        ADD     [BX+SI],AL
0100:010E 0000        ADD     [BX+SI],AL
0100:0110 0000        ADD     [BX+SI],AL
0100:0112 0000        ADD     [BX+SI],AL
0100:0114 0000        ADD     [BX+SI],AL
0100:0116 0000        ADD     [BX+SI],AL
0100:0118 0000        ADD     [BX+SI],AL
0100:011A 0000        ADD     [BX+SI],AL
0100:011C 0000        ADD     [BX+SI],AL
0100:011E 0000        ADD     [BX+SI],AL
-a_
```



## ♥ 运行命令

❖ -G [=ADDRESS ] [ADDRESS2 [ADDRESS3]

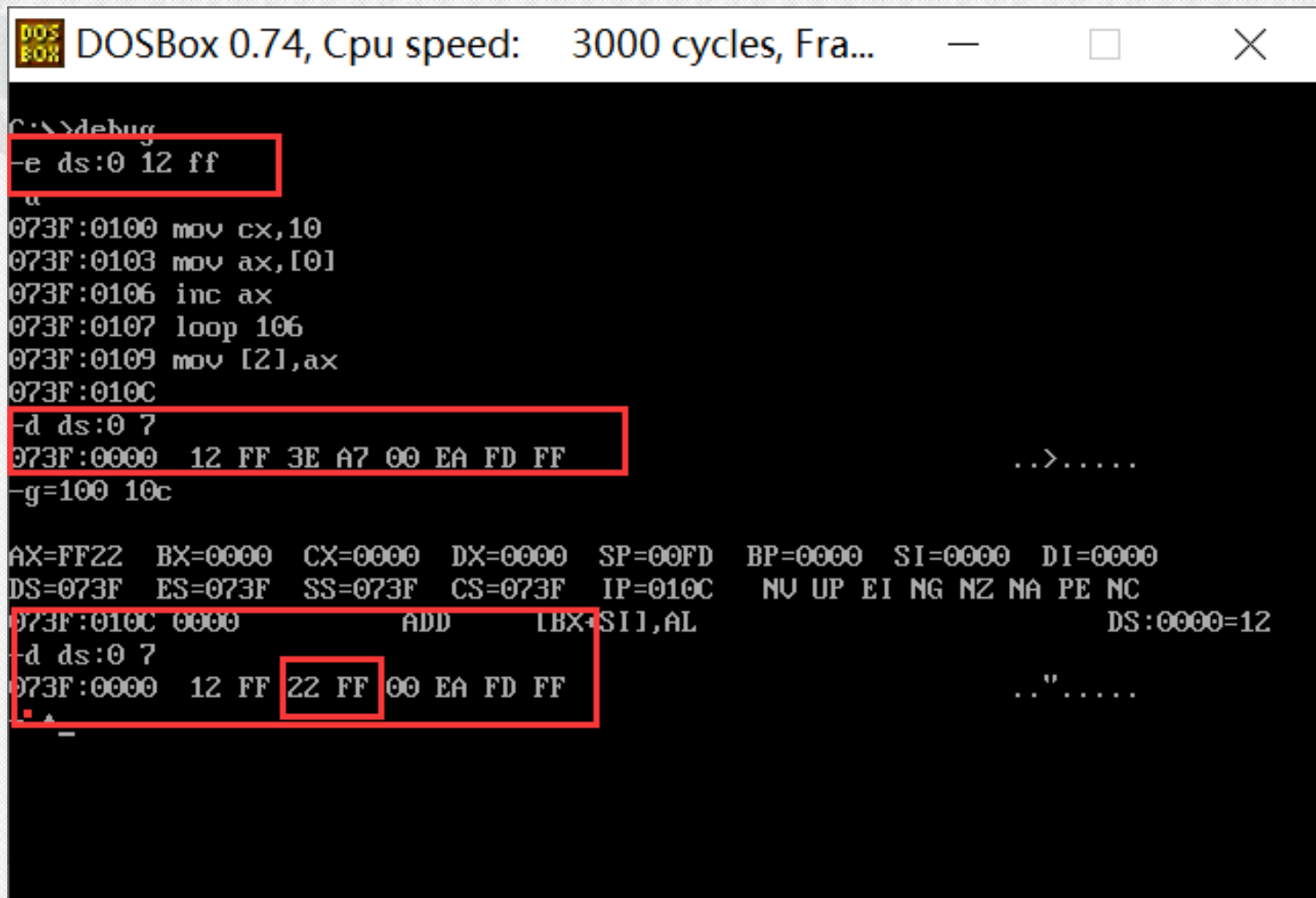


```

DOS BOX DOSBox 0.74, Cpu speed: 3000 cycles, Fra...
073F:0106 add ax,bx
073F:0108
-u
073F:0100 B83412      MOV     AX,1234
073F:0103 BB2143      MOV     BX,4321
073F:0106 01D8        ADD     AX,BX
073F:0108 0001        ADD     [BX+DI],AL
073F:010A D800        FADD    DWORD PTR [BX+SI]
073F:010C 0000        ADD     [BX+SI],AL
073F:010E AE          SCASB
073F:010F FE00        INC     BYTE PTR [BX+SI]
073F:0111 F0          LOCK
073F:0112 46          INC     SI
073F:0113 7400        JZ      0115
073F:0115 00B200B2     ADD     [BP+SI+B200],DH
073F:0119 1099002E     ADC     [BX+DI+2E00],BL
073F:011D 07          POP     ES
073F:011E 2E          CS:
073F:011F 02          POP     ES
-g=0100 108
AX=5555 BX=4321 CX=0000 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=073F IP=0108  NU UP EI PL NZ NA PE NC
073F:0108 0001        ADD     [BX+DI],AL                      DS:4321=00
- ▲

```

## ♥例子：循环实现



```
DOSBox 0.74, Cpu speed: 3000 cycles, Fra...  
C:\>debug  
-e ds:0 12 ff  
u  
073F:0100 mov cx,10  
073F:0103 mov ax,[0]  
073F:0106 inc ax  
073F:0107 loop 106  
073F:0109 mov [2],ax  
073F:010C  
-d ds:0 7  
073F:0000 12 FF 3E A7 00 EA FD FF ...>.....  
-g=100 10c  
AX=FF22 BX=0000 CX=0000 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000  
DS=073F ES=073F SS=073F CS=073F IP=010C  NU UP EI NG NZ NA PE NC  
073F:010C 0000 ADD [BX+SI],AL DS:0000=12  
-d ds:0 7  
073F:0000 12 FF 22 FF 00 EA FD FF ...".....
```



## ♥跟踪命令

- ❖ -T [=ADDRESS] [VALUE] ; VALUE运行的条数

## ♥继续命令

- ❖ -P [=ADDRESS] [VALUE]
- ❖ CALL等指令当成一条。

## ♥退出命令 -Q



## 习题一

1. 利用DEBUG程序中的“E”命令，将两个多字节数“12345678H”和“FEDCBA98H”分别送入起始地址为DS:0200H和DS:0204H两个单元中。
2. 将DS:0200H单元和DS:0204H单元中的数据相加，并将运算结果存放在DS:0208H单元中



DOS BOX DOSBox 0.74, Cpu speed: 3000 cycles, Fra...

C:\&gt;debug

-e ds:200 12345678fedcba98

^ Error

-e ds:200 12 34 56 78 fe dc ba 98

-d ds:200 210

073F:0200 12 34 56 78 FE DC BA 98-00 00 00 00 00 00 00 00 .4Ux.....

073F:0210 00

-e ds:200 78 56 34 12 98 ba dc fe

-d ds:200 210

073F:0200 78 56 34 12 98 BA DC FE-00 00 00 00 00 00 00 00 xU4.....

073F:0210 00

- ▲ ▲ \_

```

DOSBox 0.74, Cpu speed: 3000 cycles, Fra...
-d ds:200 210
073F:0200 12 34 56 78 FE DC BA 98-00 00 00 00 00 00 00 00 .4Ux.....
073F:0210 00
-e ds:200 78 56 34 12 98 ba dc fe
-d ds:200 210
073F:0200 78 56 34 12 98 BA DC FE-00 00 00 00 00 00 00 00 xU4.....
073F:0210 00
- a
^ Error
-a
073F:0100 mov ax,[200]
073F:0103 add ax,[204]
073F:0107 mov bx,[202]
073F:010B add bx,[206]
073F:010F mov [208],ax
073F:0112 mov [20a],bx
073F:0116
-g=100 116

AX=1110 BX=1110 CX=0000 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=073F IP=0116  NV UP EI PL NZ AC PO CY
073F:0116 B200          MOV     DL,00
-d ds:200 20f
073F:0200 78 56 34 12 98 BA DC FE-10 11 10 11 00 00 00 00 xU4.....
- ▲a

```



## 习题二

♥ 3. 从DS:0000H开始的4个16位带符号数分别记为x,y,z,w, 用E命令初始化为: 540,1,-1, 0, 试求 $w=x*y+z-540$

```

DOS
BOX
DOSBox 0.74, Cpu speed: 3000 cycles, Fra...
-e ds:0 1c 02 01 00 ff ff 00 00 00 00
-a
073F:0100 mov ax,[0]
073F:0103 imul word ptr[2]
073F:0107 mov cx,ax
073F:0109 mov bx,dx
073F:010B mov ax,[4]
073F:010E cwd
073F:010F add ax,cx
073F:0111 adc dx,bx
073F:0113 sub ax,21c
073F:0116 sbb dx,0
073F:0119 mov [6],ax
073F:011C mov [8],dx
073F:0120
-d ds:0 f
073F:0000 1c 02 01 00 ff ff 00 00-00 00 4f 03 a3 01 8a 03 .....0.....
g-100 120

AX=FFFF BX=0000 CX=021C DX=FFFF SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=073F IP=0120 NU UP EI NG NZ AC PE CY
073F:0120 0800 OR [BX+SI],AL DS:0000=1C
-d ds:0 f
073F:0000 1c 02 01 00 ff ff ff ff-ff ff 4f 03 a3 01 8a 03 .....0.....
- ▲

```