

bx偏移地址

- `mov ax,[bx]`
- `mov al,[bx]`

loop循环指令

- `(cx)=(cx)-1`
- 判断 `(cx)`，不为零则转至标号处执行，如果是零则向下执行

段前缀

用于显式的指明内存单元的段地址。

- `mov ax,ds:[bx]`
- `mov ax,cs:[bx]`
- `mov ax,ss:[bx]`
- `mov ax,es:[bx]`

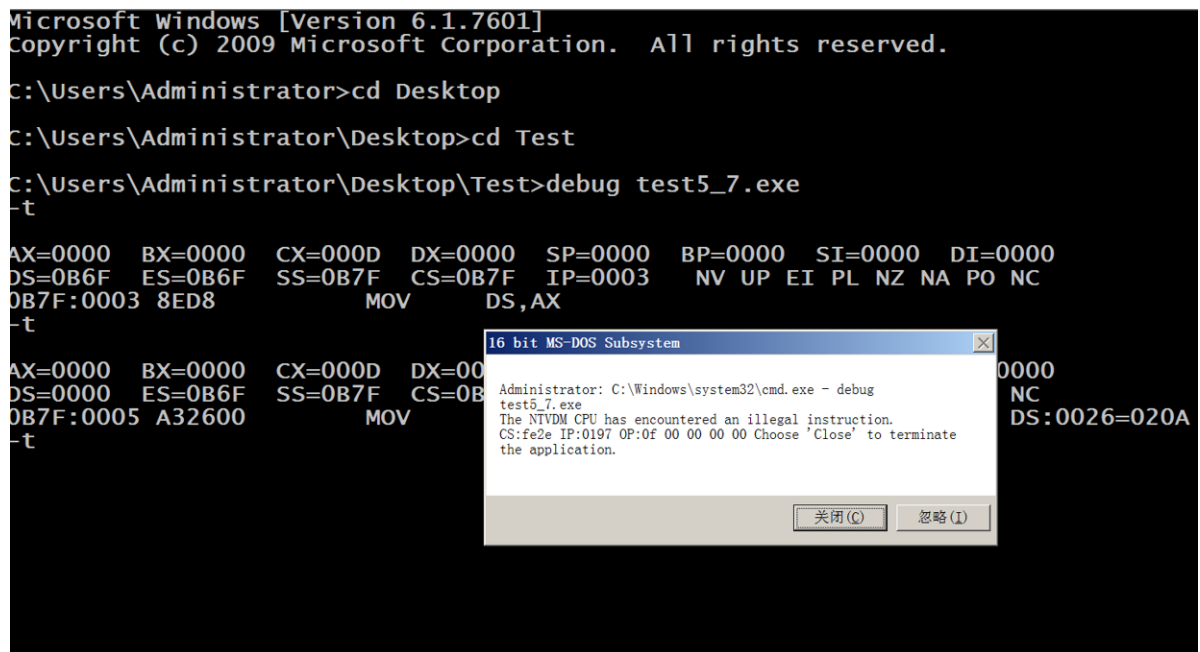
debug和masm对指令的不同处理

- debug将 `mov ax,[0]` 中的 `[0]` 解释成内存单元，而masm汇编源程序将其解释成 `0`
- 解决方案：使用bx或者显示给出段寄存器

一段安全的空间

在8086模式中，随意向一段内存空间写入内容是很危险的，因为这段空间可能存放着重要的系统数据或代码。

```
1  assume cs:code
2  code segment
3  start:mov ax,0
4          mov ds,ax
5          mov ds:[26h],ax
6
7          mov ax,4c00h
8          int 21h
9  code ends
10 end start
```



练习

计算 ffff:0006 单元中的数乘以123，结果存储在dx中。

- 汇编源程序中，数字不能以字母开头
- 跳出循环 `g + IP`

```
1  assume cs:code
2  code segment
3
4      mov ax,0ffffH
5      mov ds,ax
6
7      mov bx,6
8
9      mov al,[bx]
10     mov ah,0
11
12     mov dx,0
13     mov cx,123
14
15     s:add dx,ax
16     loop s
17
18     mov ax,4c00H
19     int 21H
20
21 code ends
22 end
```

