

中包含的是转移的位移，就不存在这个问题了，因为，无论 s 处的指令的实际地址是多少，loop 指令的转移位移是不变的。

## 9.10 编译器对转移位移超界的检测

注意，根据位移进行转移的指令，它们的转移范围受到转移位移的限制，如果在源程序中出现了转移范围超界的问题，在编译的时候，编译器将报错。

比如，下面的程序将引起编译错误：

```
assume cs:code

code segment
    start: jmp short s
           db 128 dup (0)
           s: mov ax, 0ffffh
code ends
end start
```

jmp short s 的转移范围是-128~127，IP 最多向后移动 127 个字节。

**注意：**我们在第 2 章中讲到的形如“jmp 2000:0100”的转移指令，是在 Debug 中使用的汇编指令，汇编编译器并不认识。如果在源程序中使用，编译时也会报错。

## 实验 8 分析一个奇怪的程序

分析下面的程序，在运行前思考：这个程序可以正确返回吗？

运行后再思考：为什么是这种结果？

通过这个程序加深对相关内容的理解。

```
assume cs:codesg
codesg segment

    mov ax, 4c00h
    int 21h

start: mov ax, 0
      s: nop
        nop

        mov di, offset s
        mov si, offset s2
```

```
    mov ax,cs:[si]
    mov cs:[di],ax

s0: jmp short s

s1: mov ax,0
    int 21h
    mov ax,0

s2: jmp short s1
    nop

codesg ends
end start
```

## 实验9 根据材料编程

这个编程任务必须在进行下面的课程之前独立完成，因为后面的课程中，需要通过这个实验而获得的编程经验。

编程：在屏幕中间分别显示绿色、绿底红色、白底蓝色的字符串 'welcome to masm!'。

编程所需的知识通过阅读、分析下面的材料获得。

80×25 彩色字符模式显示缓冲区(以下简称为显示缓冲区)的结构：

内存地址空间中，B8000H~BFFFFH 共 32 KB 的空间，为 80×25 彩色字符模式的显示缓冲区。向这个地址空间写入数据，写入的内容将立即出现在显示器上。

在 80×25 彩色字符模式下，显示器可以显示 25 行，每行 80 个字符，每个字符可以有 256 种属性(背景色、前景色、闪烁、高亮等组合信息)。

这样，一个字符在显示缓冲区中就要占两个字节，分别存放字符的 ASCII 码和属性。80×25 模式下，一屏的内容在显示缓冲区中共占 4000 个字节。

显示缓冲区分为 8 页，每页 4 KB(≈4000 B)，显示器可以显示任意一页的内容。一般情况下，显示第 0 页的内容。也就是说通常情况下，B8000H~B8F9F 中的 4000 个字节的内容将出现在显示器上。

在一页显示缓冲区中：

偏移 000~09F 对应显示器上的第 1 行(80 个字符占 160 个字节)；

偏移 0A0~13F 对应显示器上的第 2 行；

偏移 140~1DF 对应显示器上的第 3 行。

依此类推，可以偏移 F00~F9F 对应显示器上的第 25 行。

在一行中, 一个字符占两个字节的存储空间(一个字), 低位字节存储字符的 ASCII 码, 高位字节存储字符的属性。一行共有 80 个字符, 占 160 个字节。

即在一行中:

00~01 单元对应显示器上的第 1 列。

02~03 单元对应显示器上的第 2 列。

04~05 单元对应显示器上的第 3 列。

依此类推, 可知, 9E~9F 单元对应显示器上的第 80 列。

例如: 在显示器的 0 行 0 列显示黑底绿色的字符串'ABCDEF'  
( 'A' 的 ASCII 码值为 41H, 02H 表示黑底绿色)

显示缓冲区里的内容为:

	00	01	02	03	04	05	06	07	08	09	0A	0B	...	0E	0F
B800:0000	41	02	42	02	43	02	44	02	45	02	46	02	...	...	..
:															
:															
B800:00A0	..	..	..	..	..	..	..	..	..	..	..	..	...	...	..

可以看出, 在显示缓冲区中, 偶地址存放字符, 奇地址存放字符的颜色属性。

一个在屏幕上显示的字符, 具有前景(字符色)和背景(底色)两种颜色, 字符还可以以高亮度和闪烁的方式显示。前景色、背景色、闪烁、高亮等信息被记录在属性字节中。

属性字节的格式:

	7	6	5	4	3	2	1	0
含义	<u>B</u> L	<u>R</u>	<u>G</u>	<u>B</u>	<u>I</u>	<u>R</u>	<u>G</u>	<u>B</u>
	闪烁	背景	高亮	前景				

R: 红色

G: 绿色

B: 蓝色

我们可以按位设置属性字节, 从而配出各种不同的前景色和背景色。

比如: 红底绿字, 属性字节为: 01000010B;

红底闪烁绿字, 属性字节为: 11000010B;

红底高亮绿字, 属性字节为: 01001010B;

黑底白字, 属性字节为: 00000111B;

白底蓝字, 属性字节为: 01110001B。

例: 在显示器的 0 行 0 列显示红底高亮闪烁绿色的字符串'ABCDEF'  
(红底高亮闪烁绿色, 属性字节为: 11001010B, CAH)

显示缓冲区里的内容为:

	00	01	02	03	04	05	06	07	08	09	0A	0B	...	9E	9F
B800:0000	41	CA	42	CA	43	CA	44	CA	45	CA	46	CA	...	..	..
:															
:															
B800:00A0	..	..	..	..	..	..	..	..	..	..	..	..	...	..	..

注意, 闪烁的效果必须在全屏 DOS 方式下才能看到。