

```

1      ;代码清单5-1
2      ;文件名: c05_mbr.asm
3      ;文件说明: 硬盘主引导扇区代码
4      ;创建日期: 2011-3-31 21:15
5
6      mov ax,0xb800                ;指向文本模式的显示缓冲区
7      mov es,ax
8
9      ;以下显示字符串"Label offset:"
10     mov byte [es:0x00], 'L'
11     mov byte [es:0x01], 0x07
12     mov byte [es:0x02], 'a'
13     mov byte [es:0x03], 0x07
14     mov byte [es:0x04], 'b'
15     mov byte [es:0x05], 0x07
16     mov byte [es:0x06], 'e'
17     mov byte [es:0x07], 0x07
18     mov byte [es:0x08], 'l'
19     mov byte [es:0x09], 0x07
20     mov byte [es:0x0a], ' '
21     mov byte [es:0x0b], 0x07
22     mov byte [es:0x0c], 'o'
23     mov byte [es:0x0d], 0x07
24     mov byte [es:0x0e], 'f'
25     mov byte [es:0x0f], 0x07
26     mov byte [es:0x10], 'f'
27     mov byte [es:0x11], 0x07
28     mov byte [es:0x12], 's'
29     mov byte [es:0x13], 0x07
30     mov byte [es:0x14], 'e'
31     mov byte [es:0x15], 0x07
32     mov byte [es:0x16], 't'
33     mov byte [es:0x17], 0x07
34     mov byte [es:0x18], ':'
35     mov byte [es:0x19], 0x07
36
37     mov ax,number                ;取得标号number的偏移地址
38     mov bx,10
39
40     ;设置数据段的基地址
41     mov cx,cs
42     mov ds,cx
43
44     ;求个位上的数字
45     mov dx,0
46     div bx
47     mov [0x7c00+number+0x00],dl  ;保存个位上的数字
48
49     ;求十位上的数字
50     xor dx,dx
51     div bx
52     mov [0x7c00+number+0x01],dl  ;保存十位上的数字
53

```

```

54      ;求百位上的数字
55      xor dx,dx
56      div bx
57      mov [0x7c00+number+0x02],dl      ;保存百位上的数字
58
59      ;求千位上的数字
60      xor dx,dx
61      div bx
62      mov [0x7c00+number+0x03],dl      ;保存千位上的数字
63
64      ;求万位上的数字
65      xor dx,dx
66      div bx
67      mov [0x7c00+number+0x04],dl      ;保存万位上的数字
68
69      ;以下用十进制显示标号的偏移地址
70      mov al,[0x7c00+number+0x04]
71      add al,0x30
72      mov [es:0x1a],al
73      mov byte [es:0x1b],0x04
74
75      mov al,[0x7c00+number+0x03]
76      add al,0x30
77      mov [es:0x1c],al
78      mov byte [es:0x1d],0x04
79
80      mov al,[0x7c00+number+0x02]
81      add al,0x30
82      mov [es:0x1e],al
83      mov byte [es:0x1f],0x04
84
85      mov al,[0x7c00+number+0x01]
86      add al,0x30
87      mov [es:0x20],al
88      mov byte [es:0x21],0x04
89
90      mov al,[0x7c00+number+0x00]
91      add al,0x30
92      mov [es:0x22],al
93      mov byte [es:0x23],0x04
94
95      mov byte [es:0x24],'D'
96      mov byte [es:0x25],0x07
97
98      infi: jmp near infi                ;无限循环
99
100     number db 0,0,0,0,0
101
102     times 203 db 0
103             db 0x55,0xaa

```