

```

1      ;代码清单8-1
2      ;文件名: c08_mbr.asm
3      ;文件说明: 硬盘主引导扇区代码(加载程序)
4      ;创建日期: 2011-5-5 18:17
5
6      app_lba_start equ 100      ;声明常数(用户程序起始逻辑扇区号)
7      ;常数的声明不会占用汇编地址
8
9  SECTION mbr align=16 vstart=0x7c00
10
11     ;设置堆栈段和栈指针
12     mov ax,0
13     mov ss,ax
14     mov sp,ax
15
16     mov ax,[cs:phy_base]      ;计算用于加载用户程序的逻辑段地址
17     mov dx,[cs:phy_base+0x02]
18     mov bx,16
19     div bx
20     mov ds,ax      ;令DS和ES指向该段以进行操作
21     mov es,ax
22
23     ;以下读取程序的起始部分
24     xor di,di
25     mov si,app_lba_start      ;程序在硬盘上的起始逻辑扇区号
26     xor bx,bx      ;加载到DS:0x0000处
27     call read_hard_disk_0
28
29     ;以下判断整个程序有多大
30     mov dx,[2]      ;曾经把dx写成了ds,花了二十分钟排错
31     mov ax,[0]
32     mov bx,512      ;512字节每扇区
33     div bx
34     cmp dx,0
35     jnz @1      ;未除尽,因此结果比实际扇区数少1
36     dec ax      ;已经读了一个扇区,扇区总数减1
37 @1:
38     cmp ax,0      ;考虑实际长度小于等于512个字节的情况
39     jz direct
40
41     ;读取剩余的扇区
42     push ds      ;以下要用到并改变DS寄存器
43
44     mov cx,ax      ;循环次数(剩余扇区数)
45 @2:
46     mov ax,ds
47     add ax,0x20      ;得到下一个以512字节为边界的段地址
48     mov ds,ax
49
50     xor bx,bx      ;每次读时,偏移地址始终为0x0000
51     inc si      ;下一个逻辑扇区
52     call read_hard_disk_0
53     loop @2      ;循环读,直到读完整个功能程序

```



```

107
108         inc dx                      ;0x1f7
109         mov al,0x20                 ;读命令
110         out dx,al
111
112     .waits:
113         in al,dx
114         and al,0x88
115         cmp al,0x08
116         jnz .waits                  ;不忙，且硬盘已准备好数据传输
117
118         mov cx,256                   ;总共要读取的字数
119         mov dx,0x1f0
120     .readw:
121         in ax,dx
122         mov [bx],ax
123         add bx,2
124         loop .readw
125
126         pop dx
127         pop cx
128         pop bx
129         pop ax
130
131         ret
132
133 ;-----
134 calc_segment_base:                  ;计算16位段地址
135                                     ;输入：DX:AX=32位物理地址
136                                     ;返回：AX=16位段基地址
137         push dx
138
139         add ax,[cs:phy_base]
140         adc dx,[cs:phy_base+0x02]
141         shr ax,4
142         ror dx,4
143         and dx,0xf000
144         or ax,dx
145
146         pop dx
147
148         ret
149
150 ;-----
151         phy_base dd 0x10000          ;用户程序被加载的物理起始地址
152
153     times 510-($-$$) db 0
154                                     db 0x55,0xaa

```