

## 备份和清空硬盘分区表

December 14, 1999

Sinbad Technical Publications  
Website: <http://sinbad.zhoubin.com>  
© Copyright 1999, All Rights Reserved

---

## 一. 新系统之前的必做功--备份 MBR

经常有人提问：

装完 linux 之后，重新启动，win98 没了？

或者重装一遍 98,linux 又没了。。

解决方法一般是 fdisk /mbr,lilo -u,或者重装 rh 选择 update 了，这些都是治标不治本的方法。只有从根本上理解了他们的原理，做到自己心里有数，才不会被计算机故障急的团团转。

启动一个操作系统，首先执行硬盘第一个扇区上的代码部分，之后再把控制权交给相应的操作系统。fdisk.exe 里面有一部分是引导 DOS 的代码备份，通过 fdisk /mbr 恢复被破坏的 MBR；装上 lilo 到第一个扇区后，lilo 的代码就覆盖了原来本机的，同时再写一个备份，它是通过 lilo -u 命令恢复的。

事实上很简单，多个系统装来装去，就是在争夺 MBR 这块宝地。完整的一个扇区是 512 字节，刨去 64 字节的分区信息和 2 字节的校验码，称下的大部分就是代码。如果我们每次在新装系统前把这扇区做个备份，就能解决重装原来系统的麻烦。想想看，就 512 个字节，宏观分配着整个硬盘的分区，哪怕你是 25G 的银盘也是如此。再扯远点，问个问题，“怎样以最快的速度删除 BBS 里面某个版的所有文章？”

斑竹说：“大 D!”。

版务总管说：“砍掉讨论区!”。

站长说：“删掉该版的.DIR 文件。嘿嘿!”

当然站长是最牛的了，呵呵，他知道.DIR 文件的重要地位，没了他，那些长篇大论的文章用户一片都看不到。举这个例子就是说明，有些东东虽然很小，但没了就不行，你必须把它保养好。

不说废话了，下面简要讲一下如何备份主引导扇区：

用 debug，不要用 kv300 相应的功能，kv300 不是个好东西。:)

```
C:\>debug
```

```
-a
```

```
1C7E:0100 mov ax,201
```

```
1C7E:0103 mov bx,200
```

```
1C7E:0106 mov cx,1
```

```
1C7E:0109 mov dx,80
```

```
1C7E:010C int 13
```

```
1C7E:010E int 20
```

```
1C7E:0110
```

```
-g
```

```
Program terminated normally
```

```
-rcx
```

```
CX 0000
```

```
:200
```

```
-n a:\backup.mbr
```

```
-w200
```

```
-q
```

这样 512 字节的 MBR 就备份到软盘上的 `backup.mbr` 文件中了。以后想恢复的时候，用 `l` 命令把备份调入内存，再用 `int13` 的写命令（就是 `mov ax,301`）写入。注意，写入的操作要在纯 DOS 下进行的说。ok，只要你有这张软盘，就可以放心大胆的尝试安装 Linux 了。

## 二. 清除硬盘分区表信息

这段是很久以前写给几个爱折腾机器的朋友看的，以协助解决 DOS 下怎么删除 Linux 等 Non-DOS 分区、如何重整硬盘等问题，给个一了百了的解决方法，大家都省事。

在 pure dos 下面：

```
c:\>debug
```

```
-a //开始代码
0ACD:0100 mov ax,201 //ah=2 是读功能
0ACD:0103 mov bx,200 //把第一扇区的信息读入 200 开始的缓冲区
0ACD:0106 mov cx,1 //指定读柱面和磁道
0ACD:0109 mov dx,80 //读第一块硬盘，如果想读第二块硬盘，就 mov dx,81
0ACD:010C int 13 //int 13 中断
0ACD:010E int 20 //中止程序
0ACD:0110
-g //执行
```

```
Program terminated normally
```

```

-d380                //看读入内存中的信息
0ACD:0380
0ACD:0390    //这里略去若干。。。
0ACD:03A0    //这些字节是我凑出来的。 :)
0ACD:03B0    00 00 00 00 00 00 00 00-00 00 00 00 00 80 10    .>...u$.....u...
0ACD:03C0    74 19 0B 0E D5 96 E3 13-B0 1A 06 33 FF 8E 00 B4    t.....3....
0ACD:03D0    96 F2 05 07 75 05 4F 89-3E D5 96 BB BA 97 00 00    ....u.O.....>
0ACD:03E0    00 00 00 00 00 00 00 00-00 00 00 00 00 00 00    ...t....w..>...
0ACD:03F0    00 00 00 00 00 00 00 00-00 00 00 00 00 00 55 AA    .....< t.....

```

可以看出，从:03BE 开始的地方就是硬盘的分区表信息，最后的 55AA 是有效表示符，所以分区表信息一共有 4 项，共 64 字节。把这 64 字节清空，硬盘就几乎空空如也了。

下面往这 64 个字节填充零。

```
-f3be 3fd 0
```

写入硬盘，生效：

```

-a100                //修改刚才的代码
0ACD:0100 mov ax,301 //调用 int13 的写功能
0ACD:0103
-g=100                //从 100 处开始执行

```

搞定，fdisk 看看，啥都没有了！

这其实就是对硬盘主引导扇区的修改，可要千万小心，如果往里面写入了一大堆垃圾，你的硬盘就 over 了。找第三方工具来修复也可以，不过操作的时候还是小心为好。