

硬盘分区的划分与保护方法

陈志斌 李 群 (抚顺矿务局信息中心)

摘要:本文描述了微机硬盘的分区划分及其保护方法。了解硬盘分区的划分与保护方法,对保护硬盘的数据十分重要。

一、硬盘分区

一个硬盘通常划分为几个分区,原因可能有以下两种。

1.为了容纳多个操作系统。比如可以将硬盘划分成两个分区,分别给 MS-DOS 及 UNIX 使用,当然机器启动时只能有一个活动分区,只能有一种操作系统取得控制权。

2.可能是受当时操作系统的限制,对于大容量的硬盘不便于管理,以前的 PC/XT 机上一般都配备有 10MB 或 20MB 硬盘,随着技术的发展,现在大容量的硬盘已经很普及,如现在的 286 或 386 等高档微机基本上是 40MB 硬盘,有的达到 80MB,110MB 或 200MB 的硬盘。

以 MS-DOS 为例,在 MS-DOS3.30 以前的版本最大能够管理 32MB 硬盘,原因是 DOS 要访问本分区的扇区通常使用的是逻辑扇区号,用一个字来表示,扇区号范围是 $0-2^{16}$,即最多可有 65535 个扇区。每扇区通常为 $2^9(512)$ 字节,所以一个分区的最大容量为 $2^{16} \times 2^9 = 2^{20} \times 2^5 = 1\text{MB} \times 32 = 32\text{MB}$ 字节,超出这一范围则无法通过逻辑扇区号进行访问。

因此超过 32MB 的硬盘都要划分分区。如 AST386 有 80MB 的硬盘,可划分为一个 32MB 及一个 32MB 和 16MB 的逻辑硬盘。

二、硬盘分区的划分

1.建立分区

硬盘分区的划分是由 DOS 的外部命令 FDISK.COM 来完成的。

FDISK 使用户在硬盘上建立、激活、删除及显示分

区。FDISK.COM 与 DOS 的其它命令不同,它采用菜单操作,启动方式为 $A> \text{FDISK}$,之后根据菜单及用户的要求建立分区,先建立初始 DOS 分区 (PRIMARY DOS PARTITION) 之后建立扩展 DOS 分区 (EXTENDED DOS PARTITION) 及逻辑驱动器 (LOGICAL DRIVE)。在 FDISK 划分硬盘后,通常应用 FORMAT.COM 对各分区及逻辑驱动器进行格式化。

具体用法在一般的 DOS 手册上有详细介绍,这里不在讲述。

2.建立分区后,主引导扇区的结构

执行了 FDISK 之后,在硬盘 0 面 0 柱面 1 扇区便建立了主引导程序及硬盘分区有。主引导程序长 240 个字节,位于偏移 $0 \sim 1\text{BDH}$ 片,主要用于硬盘自举。硬盘分区表从偏移 1BE 处开始,长度为 64 个字节其中每个分区 16 个字节,描述了硬盘分区状况,该扇区最后两个字节为 55AA,这是自举程序的有效标志字。

3.每个分区 16 个字节的含义

+0: 活动分区标志

00: 不可自举分区(非活动分区)

80H: 可自举分区(活动分区),同时 80H 也是硬盘 \$

机 1 物理驱动器号

+1: 本分区开始磁头号

+2: 开始扇区号

+3: 开始柱面号

+4: DOS 系统指示字节。

其中: 01—本分区 FAT(文件分配表)每项占 12Bit (1.5 字节)

04—本分区 FAT 每项占 16Bit (2 字节)

05—当前分区表项为链接项

+5: 本分区结束磁头号

+6: 结束扇区号

+7: 结束柱面号

+8-0B: 首扇区的相对扇区号(相对于本分区范围内的第一个物理扇区,这时首扇区是分区 BOOT 所在的扇区,它是本分区内可以用 DOS 功能调用(INT 21H)访问到的第一个扇区,但却不是本分区物理的第一个扇区)

+0C-0F: 本分区的总扇区数。

三、读取硬盘分区表

硬盘分区表的读出是靠 ROM BIOS INT 13H 来实现的,DOS 的功能调用访问不到这部分扇区,因此有必要介绍一下硬盘 I/O 操作 INT 13H 的使用。

1. INT 13H 的使用

入口参数: AH = 功能号, AL = 读/写的扇区数

ES: BX = 读/写了缓冲区地址(校验操作不需要)

CH = 磁道柱面号低 8 位(高 2 位在 CL 中)

CL = 高 2 位为柱面号高 2 位,低 6 位为扇区号

DH = 磁头号

DL = 物理驱动器号(80 或 81)

出口参数: CF = 0 操作成功(AH = 0)

CF = 1 操作失败, AH = 出错状态。

在这里经常要用到 AH = 2 读扇区和 AH = 3 写扇区。

2. 利用 INT 13H 读出主引导扇区

可直接用 DEBUG 编写下面一段小程序

C>DEBUG

-A

121F:0100 MOV AX,0201

121F:0103 MOV BX,0200

121F:0106 MOV CX,0001

121F:0109 MOV DX,0090

121F:010C INT 13

121F:010E INT 3

-G

分区表链首总是固定为主引导扇区中的分区表。表中+3BE~3CD 为 C 盘的分区表项,+3BE 为 80H 表示它为活动分区,开始柱面为 0 柱面,结束柱面为 7CH 柱面。BOOT 位于相对扇区 2BH,表明隐含扇区数为

2BH,即 0 柱面 0 头下的 2BH 扇区保留,在后面谈到引导扇区分区表的保护要用到这部分扇区。

显示主引导扇区中的分区表(C 盘所在分区的分区表)

```
-03BE 3FF
121F:0380          80 01 .....
121F:03C0 01 00 04 03 88 7C 28 00-00 00 01 FF 00 00 00 00 .....K1+
121F:03D0 41 70 05 03 E8 C8 FC FF-00 00 14 80 01 00 00 00 .....A)
121F:03E0 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....
121F:03F0 00 00 00 00 00 00 00-00 00 00 00 00 55 AA .....J.
```

3CE~3D0 为链接项,它开始于 7DH 柱面 0 头 41H 扇区,它所代表的扩展分区的 BOOT 位于相对扇区 FFFCH,从扩展分区的首扇区至分区最末扇区的总扇区数为 18D14H。

3. 根据上面 C 盘链接项的指定可读出分区表链第 2 项(D 盘所在分区的分区的表)

```
-03BE 3FF
121F:0380          00 01 .....
121F:03C0 41 70 04 03 AB F8 28 00-00 00 01 FF 00 00 00 00 .....A)
121F:03D0 51 FA 05 03 E8 C8 FC FF-00 00 16 80 00 00 00 00 .....
121F:03E0 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....U.
121F:03F0 00 00 00 00 00 00 00-00 00 00 00 00 55 AA .....U.
```

同理可读出分区表链末项 E 盘所在分区的分区表

```
-03BE 3FF
121F:0380          00 01 .....
121F:03C0 81 FA 04 03 E8 C8 28 00-00 00 ED 3C 00 00 00 00 .....
121F:03D0 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....
121F:03E0 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....
121F:03F0 00 00 00 00 00 00 00-00 00 00 00 00 55 AA .....U.
```

根据上面可知这个表中无链接项,故该分区表是分区表链的最后一项。

根据上面的分析,硬盘分区表是一种链接结构,如果 C 盘的分区表被破坏掉,那么其它分区也就自然丢失了,最后导致硬盘数据的丢失。所以说保护硬盘分区表非常重要。

四、硬盘分区表的保护与恢复

1. 利用已有工具 MIRROR.COM 与 REBUILD.COM 方法

在 PCTOOLS 工具箱中,提供了一对强有力的盘信息保护工具 MIRROR 与 REBUILD。这组工具不但可对硬盘中的 FAT 和 FDT 进行保护,而且还可保护硬盘分区表。

使用带有 /PARTN 参数的 MIRROR 程序可以把硬盘分区表存储到一个名为 PARTNSAV.FIL 文件中。具体操作应由以下各步完成:

(1) C>MIRROR /PARTN

MIRROR 程序提问你想把硬盘分区表信息存储到

哪个驱动器上(A:是缺省驱动器)。

(2)把一张经过格式化的软盘插入到你选择的驱动器中按一下回车键。

这样分区表信息被写到这张软盘上并存储在 PARTNSAV.FIL 文件中。

当出现无效的驱动器说明出错信息时(INVALID DRIVE SPECIFICATION),一般是硬盘分区表被破坏造成的。这时就需要由 REBUILD / PARTN 来恢复硬盘分区表,具体应由以下两步完成:

(1)A>REBUILD / PARTN

于是,REBUILD 程序将提示你把一个软盘插入到驱动器(A:是缺省驱动器)。

(2)将带有 PARTNSAV.FIL 文件的软盘插入到选择的驱动器中,输入驱动器名称并按一下回车键,于是 REBUILD / PARTN 开始执行且该程序将提示你输入进一步信息的屏幕提示。

2.手工方法

手工保护与恢复硬盘分区表信息主要由 DEBUG.COM 来完成的。

(1)硬盘分区表信息的备份

```
A>DEBUG
-A
121F:0100 MOV AX,0201
121F:0103 MOV BX,0200
121F:0106 MOV CX,0001
121F:0109 MOV DX,0080
121F:010C INT 13
121F:010E INT 3
-G
-M200 2FF 100
-N A:BOOT.SAV
-RBX
```

```
:0
-RCX
:200
-W
-Q
```

(2)当出现 INVALID DRIVE SPECIFICATION 提示信息时,就可进行恢复硬盘分区信息。

```
A>DEBUG BOOT.SAV
-M100 300 300
-A
121F:0100 MOV AX,0301
121F:0103 MOV BX,0300
121F:0106 MOV CX,0001
121F:0109 MOV DX,0080
121F:010C INT 13
121F:010E INT 3
-G
-Q
```

3.利用硬盘中的保留扇区,将硬盘分区信息写入保留扇区中

这样每一个硬盘都存在两个分区信息表,当硬盘分区表被破坏时,就可以从保留扇区中恢复,这样既方便,又可减少错误。但有一点需注意:用这种方法硬盘必须存在保留扇区,否则可能引起硬盘文件的丢失!

实现框图略。

将程序存放在一张软盘中,可将每一台需保护的机器都运行一次此程序,当硬盘分区表被破坏时,可用一张干净的系统盘引导机器,之后运行一下此程序被破坏的硬盘分区表就被恢复完成。

在计算机的使用中,经常会遇到硬盘分区被破坏的现象,利用上面介绍的方法,笔者成功的恢复了许多硬盘分区表被破坏的机器,减少了许多不应有的损失,因此对