

第零章 磁盘与存储器的 结构与管理

本章学习目标

1. 读者可认识各种磁盘格式。
2. 读者可了解磁盘内部资料，例如，引导扇区 (Boot Sector)，目录 (Directory)，FAT (文件分配表) 和簇 (Cluster)。
3. 读者可了解存储器的基本定义。

在学习 PC Tools 之前若是能对磁盘的结构有基本认识，将可使各位更容易了解 PC Tools 各功能的意义，因此本书特在第零章讲解磁盘的知识。

0-1 软盘

DOS 所使用的磁盘的基本结构如图 0-1 所示。

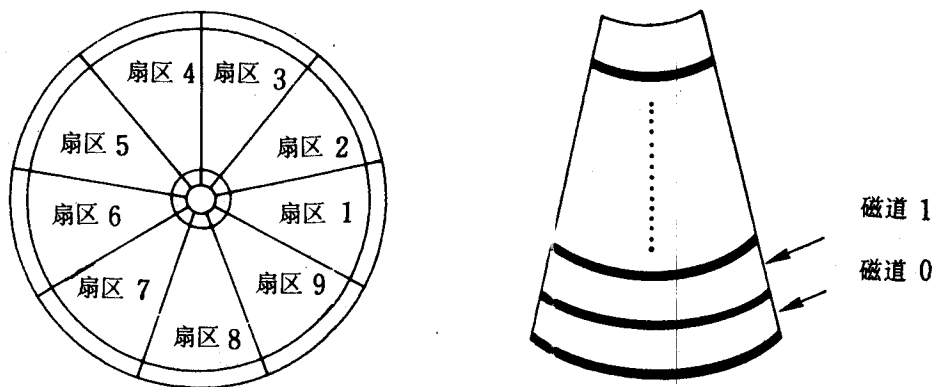


图 0-1 磁盘基本结构

数据储存在同心圆上，这些同心圆又称为磁道 (track)，磁盘驱动器的读写头会利用磁盘旋转，从一条磁道移动到另一指定磁道，读取或写入数据。

磁道又可分成若干段落，一般我们称这些段落为扇区 (sector)。扇区上容量的大小是用字节 (Byte) 来衡量的，每个扇区可容纳 512 个字节，每一个字节可储存一个字符 (character)。

以下是常用磁盘容量规格的说明：

1. 单面 160KB 磁盘：每盘有 40 道。
 (5 1/4 英寸) 每道有 8 扇区。
 每扇区有 512Byte。
 $40 \times 8 \times 512 = 163840 \text{Byte} = 160 \text{KB}$
2. 单面 180KB 磁盘：每盘有 40 道。
 (5 1/4 英寸) 每道有 9 扇区。
 每扇区有 512Byte。
 $40 \times 9 \times 512 = 184320 \text{Byte} = 180 \text{KB}$
3. 双面 320KB 磁盘：每盘有两面。
 (5 1/4 英寸) 每面有 40 道。
 每道有 8 扇区。
 每扇区有 512Byte。
 $2 \times 40 \times 8 \times 512 = 327680 \text{Byte} = 320 \text{KB}$
4. 双面 360KB 磁盘：每盘有两面。
 (5 1/4 英寸) 每面有 40 道。
 每道有 9 扇区。
 每扇区有 512Byte。
 $2 \times 40 \times 9 \times 512 = 368640 \text{Byte} = 360 \text{KB}$
5. 双面 1.2MB 磁盘：每盘有两面。
 (5 1/4 英寸) 每面有 80 道。
 每道有 15 个扇区。
 每扇区有 512Byte。
 $2 \times 80 \times 15 \times 512 = 1228800 \text{Byte} = 1.2 \text{MB}$
6. 双面 720KB 磁盘：每盘有两面。
 (3 1/2 英寸) 每面有 80 道。
 每道有 9 个扇区。
 每扇区有 512Byte。
 $2 \times 80 \times 9 \times 512 = 737280 \text{Byte} = 720 \text{KB}$
7. 双面 1.44MB 磁盘：每盘有两面。
 (3 1/2 英寸) 每面有 80 道。
 每道有 18 个扇区。
 每扇区有 512Byte。
 $2 \times 80 \times 18 \times 512 = 1474560 \text{Byte} = 1.44 \text{MB}$
8. 双面 2.88MB 磁盘：每盘有两面。
 (3 1/2 英寸) 每面有 80 道。
 每道有 36 个扇区。
 每扇区有 512Byte。
 $2 \times 80 \times 36 \times 512 = 2949120 \text{Byte} = 2.88 \text{MB}$

从前面叙述可知有的磁盘是单面，有的磁盘是双面，若是磁盘为双面则我们又称其一面为第 0 面，另一面为第 1 面。

0-2 逻辑扇区与物理扇区

DOS 在正式处理扇区数据时又将扇区分为两种定义，一是逻辑扇区(logical sector)，另一是物理扇区 (Physical sector)。

所谓的逻辑扇区是以一系列数字来执行扇区的编号，从 0 开始，如果某个磁盘共有 720 个扇区（例如，360K 磁盘），则它的扇区编号从 0 至 719。如果某个磁盘共有 2400 个扇区（例如，1.2M 磁盘），则它的扇区编号从 0 至 2399。如果某个磁盘共有 2880 个扇区（例如，1.44M 磁盘），则它的扇区编号从 0 至 2879。

物理扇区则完全是以实际扇区位置代表其位置。若要知道某个扇区位置需了解三个信息，一是第几面，二是第几道，三是第几扇区。表 0-1 是 360K 磁盘逻辑扇区与物理扇区的对应表。表 0-2 是 1.2M 磁盘逻辑扇区与物理扇区的对应表。表 0-3 是 1.44M 磁盘逻辑扇区与物理扇区的对应表。

表 0-1 360K 磁盘的逻辑扇区和物理扇区对应表

物 理 扇 区			逻辑扇区编号
磁 面	磁 道	扇 区	
0	0		0
0	0		1
0	0		2
⋮	⋮	⋮	⋮
0	0		8
1	0		9
1	0		10
⋮	⋮	⋮	⋮
1	39		717
1	39		718
1	39		719

表 0-2 1.2M 磁盘的逻辑扇区和物理扇区对应表

物 理 扇 区			逻辑扇区编号
磁 面	磁 道	扇 区	
0	0	1	0
0	0	2	1
0	0	3	2
⋮	⋮	⋮	⋮
0	0	15	14
1	0	1	15
⋮	⋮	⋮	⋮
1	0	15	29
0	1	1	30
⋮	⋮	⋮	⋮
1	79	14	2398
1	79	15	2399

表 0-3 1.44M 磁盘的逻辑扇区和物理扇区对应表

物 理 扇 区			逻辑扇区编号
磁 面	磁 道	扇 区	
0	0	1	0
0	0	2	1
0	0	3	2
⋮	⋮	⋮	⋮
0	0	18	17
1	0	1	18
⋮	⋮	⋮	⋮
1	0	18	35
0	1	1	36
⋮	⋮	⋮	⋮
1	79	17	2878
1	79	18	2879

0-3 硬盘

硬盘的基本外观如图 0-2 所示。

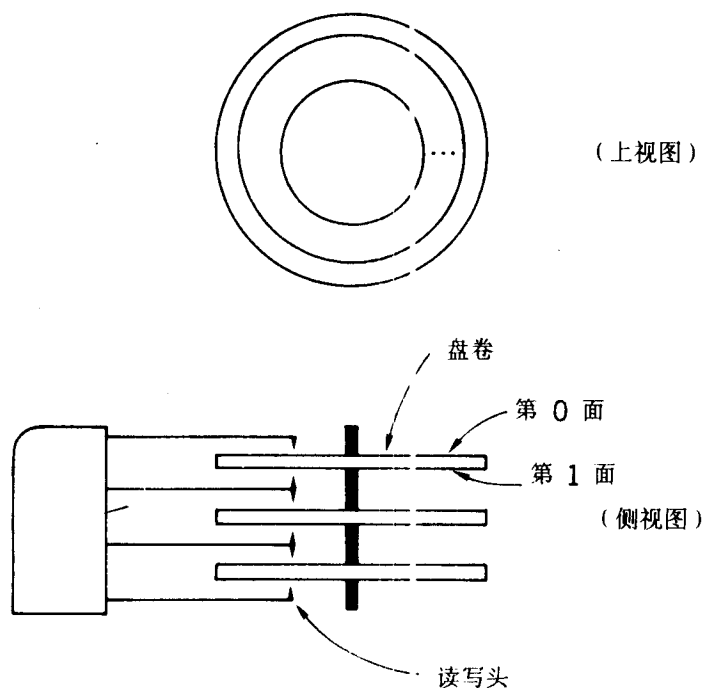


图 0-2 硬盘基本外观

硬盘主要是由许多盘卷 (platters) 所组合而成, 至于是由多少盘卷组合而成则完全视硬盘的容量而定。每个硬盘包含许多盘卷, 我们将一系列盘卷中相同半径的磁道称之为柱面 (cylinder), 因此我们可以说所谓第 0 柱面指的便是第 0 道所成的集合, 第 1 柱面是第 1 道所成的集合……等等。

由于硬盘内有许多盘卷, 通常我们又将最上面的盘卷称之为第 0 面和第 1 面, 下一个盘卷称第 2 面和第 3 面, ……等, 其它请依此类推。

0-4 簇的定义

簇 (cluster) 是 DOS 为数据分配扇区的最基本单位, 不同的磁盘其簇的大小并不一样, 如表 0-4 所示。

表 0-4

磁 盘 类 型	每一簇的扇区数
360K	2
1.2M	1
1.44M	1

实例：观察 360K 磁盘，每一簇所占的扇区数（如图 0-3）。

```

A: \HISTORY>dir
Volume in drive A has no label
Directory of A: \HISTORY
.           <DIR>          11-24-92   1 : 25f
. .         <DIR>          11-24-92   1 : 25f
2 file (s)                0bytes
52224 bytes free←原先磁盘容量

A: \HISTORY>copy con test.txt
This is cluster testing. ←建立此文件
^ z
1 file (s) copied

A: \HISTORY>dir
Volume in drive A has no label
Directory of A: \HISTORY
.           <DIR>          11-24-92   1 : 25f
. .         <DIR>          11-24-92   1 : 25f
TEST       TXT           11-27-92   9 : 30a
3 file (s)                26 bytes
51200 bytes free←磁盘剩余容量

A: \HISTORY>

```

所建文件长度

26

图 0-3

从这个实验可以看到原先磁盘尚剩 52224 字节，但建立一个仅含 26 个字符的文件之后，再观察此磁盘时，磁盘容量仅剩 51200 字节，DOS 居然以 1024 字节（52224－51200＝

1024) 存放该文件，这表示存放该文件共耗用了 2 个扇区。从以上实验可以得到结论，对 360K 磁盘而言，每个簇占 2 个扇区，也就是当你建立文件时，DOS 会以 1024 字节为单位分配空间给文件。例如，若是文件长度是 26 字节，则分配 1024 字节空间给它，若是文件长度是 1050 字节，则分配 2048 字节空间给它。

实例：观察 1.2M 磁盘，每一簇所占的扇区数（如图 0-4）。

```
A: \TMP>dir
Volume in drive A has no label
Directory of A: \TMP
.                <DIR>          11-27-92    9: 46:1
..               <DIR>          11-27-92    9: 46:1
2 file (s)              0 bytes
902144 bytes free<--原先磁盘容量

A: \TMP>copy con test. txt
This is cluster testing. <--建立此文件
^ z
1 file (s) copied

A: \TMP>dir
Volume in drive A has no label
Directory of A: \TMP
.                <DIR>          11-27-92    9: 46:1
..               <DIR>          11-27-92    9: 46:1
TEST            TEST          26 11-27-92    9: 46:1
3 file (s)              26 bytes
901632 bytes free<--磁盘剩余容量

A: \TMP>
```

图 0-4

从这个实验可以看到原先磁盘尚余 902144 字节，但建立一个 26 字节的文件之后，磁盘剩余容量是 901632 字节。因此可以得到结论，对 1.2M 磁盘而言，当你建立文件时，DOS 会以 512 字节为单位分配给文件，这相当于一个扇区，因此，我们可以知道对 1.2M 磁盘而言，每个簇占 1 个扇区。

实例：观察 1.44M 磁盘，每一个簇所占的扇区数（如图 0-5）。

```
B: \TMP>dir
Volume in drive B is NUDE2
Volume serial Number is 0B2F-1B00
Directory of B: \TMP
.                <DIR>          11-27-92    4: 24p
. .              <DIR>          11-27-92    4: 24p
                2 file (s)          0 bytes
                82944 bytes free←原先磁盘容量

B: \TMP>copy con test. txt
This is cluster testing. ←建立此文件
^ Z

                1 file (s) copied

B: \TMP>dir
Volume in drive B is NUDE2
Volume serial Number is 0B2F-1B00
Directory of B: \TMP
.                <DIR>          11-27-92    4: 24p
. .              <DIR>          11-27-92    4: 24p
TEXT            TXT            26 11-27-92    4: 24p
                3 file (s)          26 bytes
                82432 bytes free←磁盘剩余容量

B: \TMP>
```

图 0-5

从这个实验可以看到原先磁盘尚余 82944 字节，但建立一个 26 字节的文件之后，磁盘剩余容量是 82432 字节。因此可以得到结论，对 1.44M 磁盘而言，当你建立文件时，DOS 会以 512 字节为单位分配给文件，这相当于一个扇区，因此，我们可以知道对 1.44M 磁盘而言，每个簇占 1 个扇区。

0-5 引导扇区

不管你是使用哪一种磁盘，磁盘第 0 面第 0 道第 1 扇区（又称第 0 逻辑扇区）又称引导扇区（Boot Sector），而这引导扇区又包含三部分，一是跳转指令，二是 BIOS 参数区段（BIOS Parameter Block 简称 BPB），三是系统引导程序（Bootstrap），如下所示：

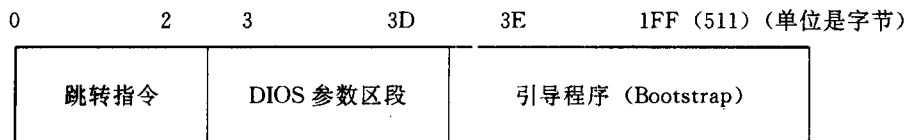


图 0-6 是利用 PC Tools 扇区编辑功能 (参考第 8 章) 所得的引导扇区的 16 位参考图 (以 DOS 5.0 为例)。

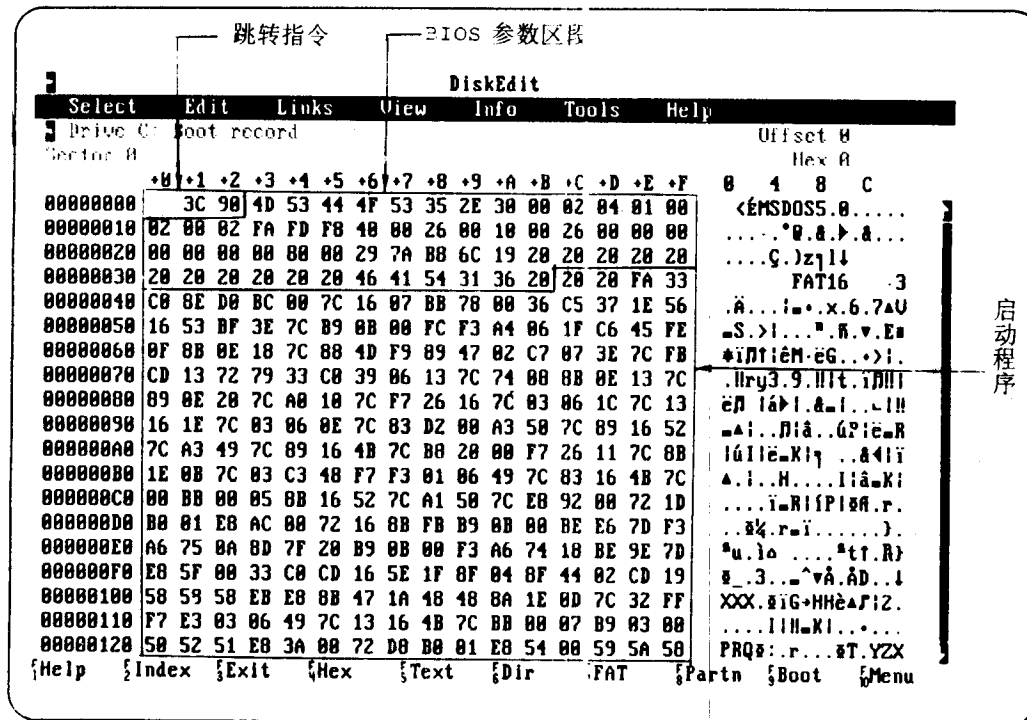


图 0-6 硬盘 C 引导扇区图

有关上图的产生方式将在本书第 8 章作说明。

跳转指令

引导扇区的前三个指令是跳转指令，它的意义如下：

EB……这是汇编语言指令“JMP NEAR (近程)”近程跳转的机器码。

3C……若和 EB 配合，表示将跳至距离目前地址 3C 差距的地址执行启动程序。由于目前地址在 01，所以相当于跳至 3D 地址，然后便可以执行下一个地址指令，相当于执行启动程序。

90……这是汇编语言指令“NOP”的机器码。

BIOS 参数区段

本部分包含目前这份磁盘的基本使用信息，以便供未来其它公用程序读取本磁盘时使用，它的说明请参考表 0-5。

表 0-5 BIOS 参数区段

扇区地址	说 明
第 3~0A 字节	OEM 制造厂商名称 (DOS 厂商) 及 MS-DOS 版本信息
第 0B~0C 字节	每个扇区含多少字节
第 0D 字节	每个簇含多少扇区
第 0E~0F 字节	保留扇区数，通常是指引导扇区的个数，通常此值是 1
第 10 字节	指明文件分配表 (FAT) 的个数，通常是 2 份，若第 1 份损坏时可使用第 2 份做数据复原
第 11~12 字节	指明根目录 (root directory) 可储存文件或子目录的数量
第 13~14 字节	指明本磁盘的扇区数，如果磁盘容量大于 32MB 则此部分是 0，将于另外一个域指明扇区数
第 15 字节	媒体描述表，描述磁盘格式，请参考表 0-6 媒体描述表。
第 16~17 字节	指明每个 FAT 占多少扇区
第 18~19 字节	指明每一道占多少扇区
第 1A~1B 字节	指明本磁盘可使用磁面有几个
第 1C~1F 字节	指明本磁盘有多少隐藏扇区
第 20~23 字节	本域仅供磁盘容量大于 32M 时使用，本域将指明本磁盘的扇区数。
第 24 字节	本域仅 DOS 内部使用，指明此磁盘是否是第 1 个硬盘，如果是则此值是 80H，如果不是则此值是 00H
第 25 字节	保留未使用
第 26 字节	此值是 29H，指明引导扇区记号
第 27~2A 字节	磁盘序列编号 (ID)
第 2B~35 字节	磁盘标题名称
第 36~3D 字节	记载 FAT 的使用格式，如果是 12 位的 FAT 则此域是 FAT12，如果是 16 位的 FAT 则此域是 FAT16，剩余 3 个字节则填空白，ASCII 值是 20H，详情可参看 0-6 节。

表 0-6 媒体描述表

值	媒 体 名 称
0F0H	3 1/2 吋 1.44M 磁盘 3 1/2 吋 2.88M 磁盘
0F8H	适用任何型式的硬盘
0F9H	3 1/2 吋 720K 磁盘 5 1/4 吋 1.2M 磁盘
0FAH	5 1/4 吋 320K 磁盘
0FBH	3 1/2 吋 640K 磁盘
0FCH	5 1/4 吋 180K 磁盘
0FDH	5 1/4 吋 360K 磁盘
0FEH	5 1/4 吋 160K 磁盘
0FFH	5 1/4 吋 320K 磁盘

0-6 FAT 的基本定义

FAT 的全名是 File Allocation Table, 它的中文意义是文件分配表, 通常在格式化 (FORMAT) 新的磁盘时, DOS 会在磁盘上建立 2 份 FAT, 这两份 FAT 内容完全相同, 不过 DOS 通常只使用第 1 份, 第 2 份 FAT 主要是用在若第 1 份 FAT 有损坏时, 则可使用第 2 份做挽救文件之用。

FAT 是紧接在引导扇区之后的, 也就是从第 0 面第 0 道第 2 扇区起是 FAT 扇区, 至於 FAT 占多少扇区, 且在哪些位置, 则视磁盘的不同而不同, 如表 0-7 所示。

表 0-7 常用磁盘的 FAT 在磁盘的位置

磁 盘 格 式	FAT 所占扇区数	第 1 份 FAT 物理位置	第 2 份 FAT 物理位置
5 1/4 英寸 360K	2	0 面 0 道 2-3 扇区	0 面 0 道 4-5 扇区
5 1/4 英寸 1.2M	7	0 面 0 道 2-8 扇区	0 面 0 道 9-15 扇区
3 1/2 英寸 1.44M	12	0 面 0 道 2-7 扇区 0 面 0 道 16-18 扇区 1 面 0 道 1-3 扇区	1 面 0 道 4-15 扇区

有关 FAT 的用法将在 0-8 节中做综合说明。

0-7 目录的基本定义

当执行格式化磁盘时，DOS 也将在磁盘上建立目录区，而此目录在磁盘内占多少扇区则视磁盘容量的不同而有所不同，如表 0-8 所示。

表 0-8 常用磁盘的目录区在磁盘的位置

磁 盘 格 式	目录所占扇区数	目录区的物理位置
5 1/4 英寸 360K	7	0 面 0 道 6-9 扇区 1 面 0 道 1-3 扇区
5 1/4 英寸 1.2M	14	1 面 0 道 1-14 扇区
3 1/2 英寸 1.44M	8	0 面 0 道 8-15 扇区

0-8 目录、簇和 FAT 的综合说明

在 DOS 的磁盘中，有二种类型的目录：一是具有固定大小的根目录（root directory），另一是可具有任意大小的子目录（subdirectory）（DOS 1. X 版没有子目录类型）。而每一个磁盘有一个（也只有一个）根目录，但却可以有零个到多个子目录（直到磁盘装满为止）。而子目录之间又可以有其它的子目录，这也是所谓的层次目录结构（又称树型目录），如图 0-7 所示。

磁盘内文件目录就像是磁盘内容的表格，磁盘内每个文件在目录内都有一个对应的目录项。每个目录项长度是 32 字节，从表 0-8 可知 360K 磁盘有 7 个目录扇区，所以：

$7 \times 512 = 3584 \cdots \cdots$ 目录所占空间数
 $3584 / 32 = 112 \cdots \cdots$ 每个磁盘的目录容量

对 1.2M 的磁盘而言，每个目录项长度和 360K 磁盘一样是 32 字节，从表 0-8 可知 1.2M 磁盘有 14 个目录扇区，所以：

$14 \times 512 = 7168 \cdots \cdots$ 目录所占空间数
 $7168 / 32 = 224 \cdots \cdots$ 每个磁盘的目录容量

对 1.44M 的磁盘而言每个目录项长度和 360K 磁盘一样是 32 字节，从表 0-8 可知 1.44M 磁盘有 8 个目录扇区，所以，

$8 \times 512 = 4096 \cdots \cdots$ 目录所占空间数
 $4096 / 32 = 128 \cdots \cdots$ 每个磁盘的目录容量

而 32 bytes 长的目录项，又可分成 8 个域，如图 0-8 所示。

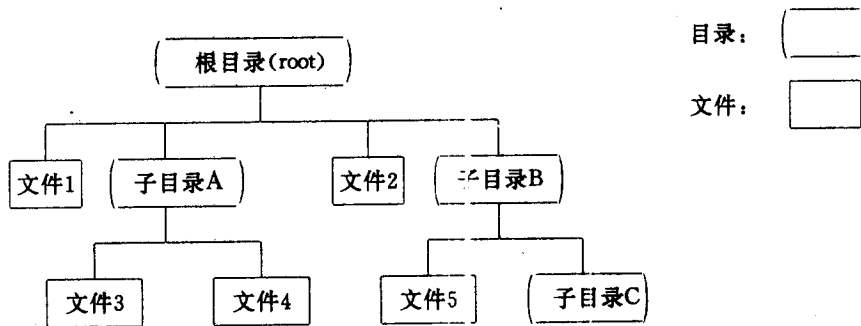


图 0-7 DOS 的层次目录结构

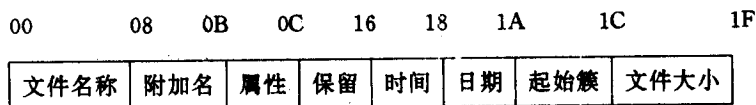


图 0-8 文件目录项的结构图

图 0-8 中的第一个域是 (0-7 字节) 文件名称, 这个域占 8 字节, 如果文件名称少于 8 个字符则其余的以空白填充, 然而目录项中, 文件名称域的第一个字节, 有其特殊意义:

00H: 代表这个目录项从未被使用过。

05H: 文件名称的第一个字符实际上是 E5H, 但 E5H 有其特殊的意义 (代表某文件已被删除)。因此 DOS 会将 E5H 开头的文件名称, 更改成 05H。

2EH: 表示这是一个子目录。此时起始簇会指向目录本身所在的簇。

若是下一个字节也是 2EH, 则起始簇会指向父目录所在的簇, 若父系目录是根目录, 则其值为零。

E5H: 表示文件已被删除。其实删除, 并不是将整个文件删除, 只是把第一个字节设成此值, 所以要挽回被删除的文件, 可利用此特性。

目录项目的第 2 个域 (8-0AH) 名称是附加名, 如果附加名少于 3 个字符, 则以空白填充。

第 3 个域 (0BH) 是文件属性, 其中每个位代表一个属性, 若某位为 1 表示该文件具有此属性, 反之则没有。下列假设各位值是 1。

位 0: 只读文件。只能读取, 无法执行写入和删除。

位 1: 隐含文件。使用 DIR 无法看到此文件。

位 2: 系统文件。启动 DOS 系统时所使用的文件。

位 3: 磁盘标记。这个项目一定要在根目录内, 且一个磁盘仅能有一个这种项目。

位 4: 子目录。设定此位时, 该目录项目表示是子目录。

位 5: 处理位。只要文件被修改, 这个位都会被设定。

位 6: 保留

位 7: 保留

第 4 个域 (0CH~15H) 是保留, 预备给较新版本的 DOS 使用。

第 5 个域 (16H~17H) 是代表文件建立或最后一次修改的时间，其使用方式如下：

位 0~4：以 2 秒为增量单位 (0~29 相当於 0~58 秒)

位 5~0AH：表示分 (0~59)

位 0BH~0FH：表示时 (0~23)

第 6 个域是代表文件建立或最后一次修改的日期。其使用方式如下：

位 0~4H：表示日期 (1~31)

位 5~8H：表示月份 (1~12)

位 9~0FH：表示年，以 1980 为基底 (即以 0 表 1980) 的二进制数值。

第 7 个域 (1AH~1BH) 是起始簇，表示文件是从磁盘哪里开始存放。

第 8 个域 (1CH~1FH) 是文件大小，这个域有 4 个字节长，储存时先存入低两个字节，放在低地址处，然后再存入高两个字节，放在高地址处。

文件分配表 (FAT) 主要是记录磁盘空间 (簇) 使用的情况。如表 0-7 所示，若以 360K 磁盘为例，每个 FAT 占用二个扇区，对大多数磁盘格式而言，DOS 都拥有两个一模一样的 FAT，用以预防一个损坏或不可读时当备份用，但在一般情况，DOS 并不会使用多余的那一个 FAT。

FAT，从字义上可知道是一个表格，表格上则被划分成许多域，每个域恰有一个簇与之配合，如图 0-9 所示。

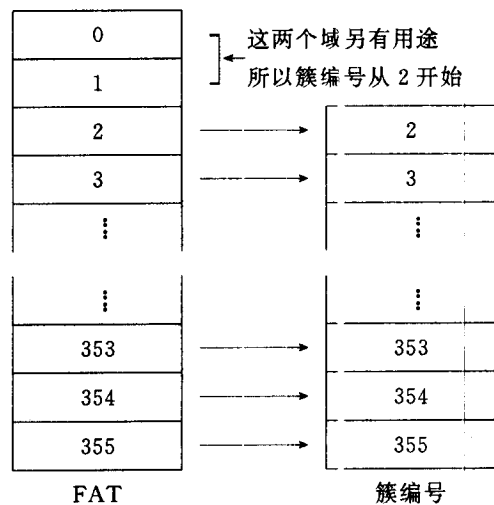


图 0-9 360K 磁盘 FAT 与互相对应的簇

各簇编号相对於扇区的位置如图 0-10 所示。

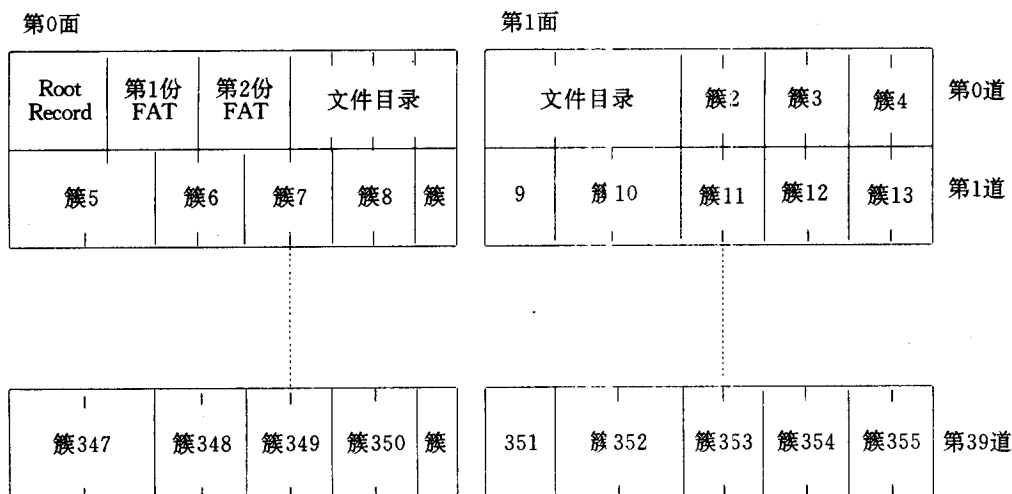


图 0-10 各簇在扇区内分布情形

FAT 域有 12 位和 16 位两种，12 位的 FAT 格式，较常用且较复杂（例 360K，1.2M，1.44M 磁盘使用此格式），至於 16 位的 FAT 格式则一般用在硬盘上。所谓的 12 位 FAT 格式是指 3 个字节（ $3 \times 8 = 24$ ）共 24 个位，其中前 12 个位代表前一个 FAT 域，后 12 个位代表另一个 FAT 域。而所谓 16 位 FAT 格式是指 2 个字节代表一个 FAT 域。

由於 FAT 域的前二个域（3 个字节）代表特殊意义，所以并不指向簇编号，这也是为什么我们从 2 开始进行簇编号。FAT 第一个域的第一个字节又被称为是磁盘的 ID，主要用於辨别磁盘格式。其所代表意义如表 0-9 所示。

表 0-9 FAT 第 1 域值与磁盘格式表

值	磁盘格式
0F0H	3 1/2 英寸 1.44M 磁盘
0F8H	硬盘
0F9H	1.2M 磁盘
0FDH	360K 磁盘

其它 FAT 内含值的意义如下：

- (0) 000H：表示此域所对应的簇尚未使用。
- (F) FF7H：表示此域所对应的簇已损坏，不可使用。
- (F) FF8H～(F) FFFH：表示这是文件的最后一个簇。
- (F) FF60～(F) FF6H：保留簇。

其他值：此值为下一个簇的编号。

先前已知，目录项 1AH～1CH 域会指向该文件起始簇的编号。假设有一文件，其目录项，FAT，簇关系如图 0-11 所示。则磁盘在读取该文件时，首先会从目录项得知文件 A 起始地址在第 3 簇，由於第 3 个 FAT 是 4，因此可知此文件另一部分内容放在第 4 簇内，由

於第 4 个 FAT 值是 5，所以第 5 个簇也存有文件 A 的内容，因为第 5 个 FAT 值是 FFE（介於 FF8H~FFFH 间）表示这是最后一个簇，所以可知文件 A 是由 3、4、5 簇所组成。

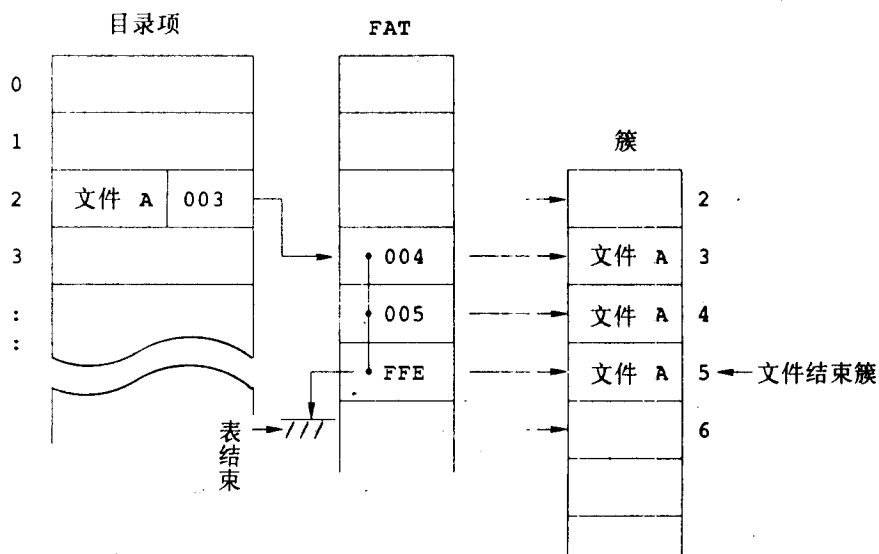
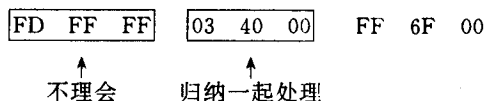


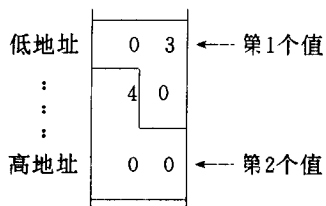
图 0-11

从前面内容相信读者已经了解目录、FAT 和簇内基本使用目的及原则了。接下来笔者要讨论的是如何判读 12 位和 16 位的 FAT 值。

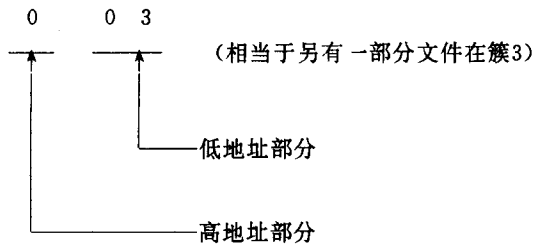
首先要讲解的是判读 12 位 FAT 的方式，这也是比较复杂的方式，处理 12 位 FAT 时由於 3 个字节代表 2 个 FAT 域值，因此最好的方式是将 3 个字节归纳在一起再做处理。例如若是 FAT 域内容如下：



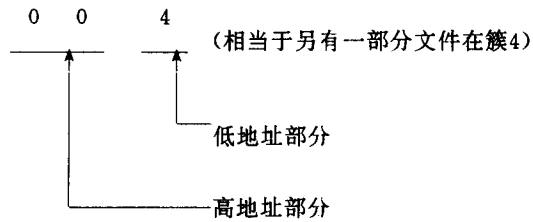
则 03 40 00 的存储器内容如下：



在处理上述值时尚要有一个观念是低地址代表低位数，高地址代表高位数，因此第 1 个值为如下所示：



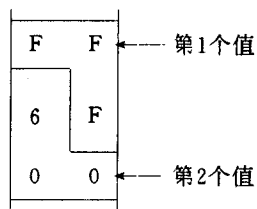
同理可得第 2 个 FAT 域值如下：



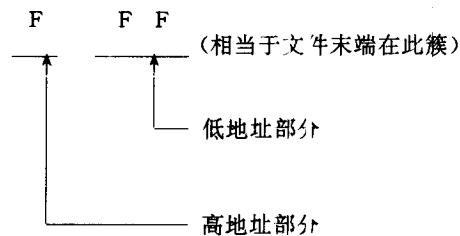
接下来我们来探讨下两个 FAT 域内容

FF 6F 00

则其存储器内容如下：



因此第 1 个值如下：



第 2 个值如下：

