

外存空闲空间的管理

3. 位示图法

位示图是利用二进制的一位来表示磁盘中一个盘块的使用情况，磁盘上所有的盘块都有一个二进制位与之对应。当其值为“0”时，表示对应的盘块空闲；为“1”时，表示已分配。这样，一个 $m \times n$ 位组成的位示图就可用来表示 $m \times n$ 个盘块的使用情况，如图 4.20 所示。

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	1	1	0	0	0	1	1	1	0	0	1	0	0	1	1	0
2	0	0	0	1	1	1	1	1	1	0	0	0	0	1	1	1
3	1	1	1	0	0	0	1	1	1	1	1	1	0	0	0	0
4																
⋮																
16																

图 4.20 位示图法示意图

注意：本书如无特别提示，所使用的位示图法中行和列都从 1 开始编号。特别注意，若题目中指明从 0 开始编号，则上述计算方法要进行相应调整。

真题 2015.12

2. 用一张 8 个 16 位字长的字组成的“位示图”来管理一个高速存储器，现规定
字号，位号和块号均从 1 开始计，试问：

(1) 该位示图可表示多少块？

(2) 字号为 7，位号 13 所对应的块号是多少？

(3) 块号 55 对应的字号和位号分别是多少？

页式存储管理+位示图

三、综合应用题

1. 页式存储管理中，主存空间按页分配，可用一张“位示图”构成主存分配表。假设主存容量为 2M 字节，页面长度为 512 字节，若用字长为 32 位的字作主存分配的“位示图”需要多少个字？如页号从 1 开始，字号和字内位号（从高位到低位）均从 0 开始，试问：第 2999 页对应于何字何位；99 字 19 位又对应于第几页？

页式存储管理+位示图

- 2) 页式存储管理有无零头存在? 若有, 会存在什么零头? 为该作业分配内存后, 会产生零头吗? 若产生, 大小为多少? (提示: 这里的零头是指一页中未被使用的部分。)
- 3) 假设一个 64MB 内存容量的计算机, 其操作系统采用页式存储管理(页面大小为 4KB), 内存分配采用位示图方式管理, 请问位示图将占用多大的内存?

1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	0	1	1	1	1	0	0	0	1	1
1	1	0	0	0	0	0	0	0	0	0	0	1	1	1
1	1	1	1	1	0	0	0	0	1	0	0	0	1	0
0	1	0	1	1	0	1	1	0	1	0	1	1	0	1
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	1	1	1	1	1	0	0	0	0	0	0	0	0
1														
.....														

页号	块号 (从 0 开始编址)

3.1.7 答案与解析

练习

3. 问答题

(1) 设某系统的磁盘空间共有 5000 块，若用位示图管理磁盘空间，位示图的每个字有 32 位，并且物理块号、字号、位号均从 1 开始。试问：

- 1) 位示图需要多少个字构成？
- 2) 计算位示图第 9 个字第 22 位对应的物理块号。
- 3) 求物理块号 106 对应的字和位。

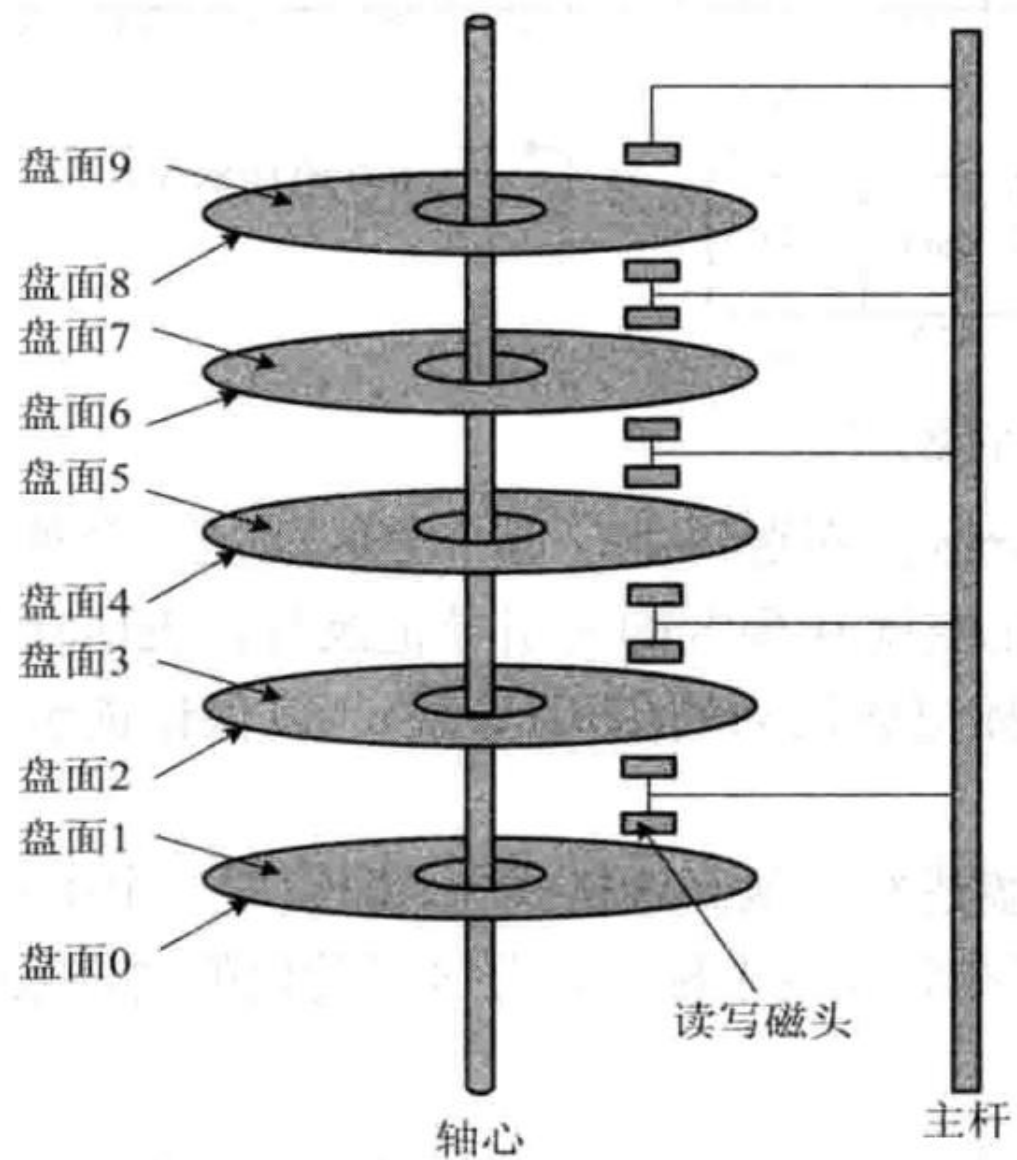
真题 2014.11

磁盘结构

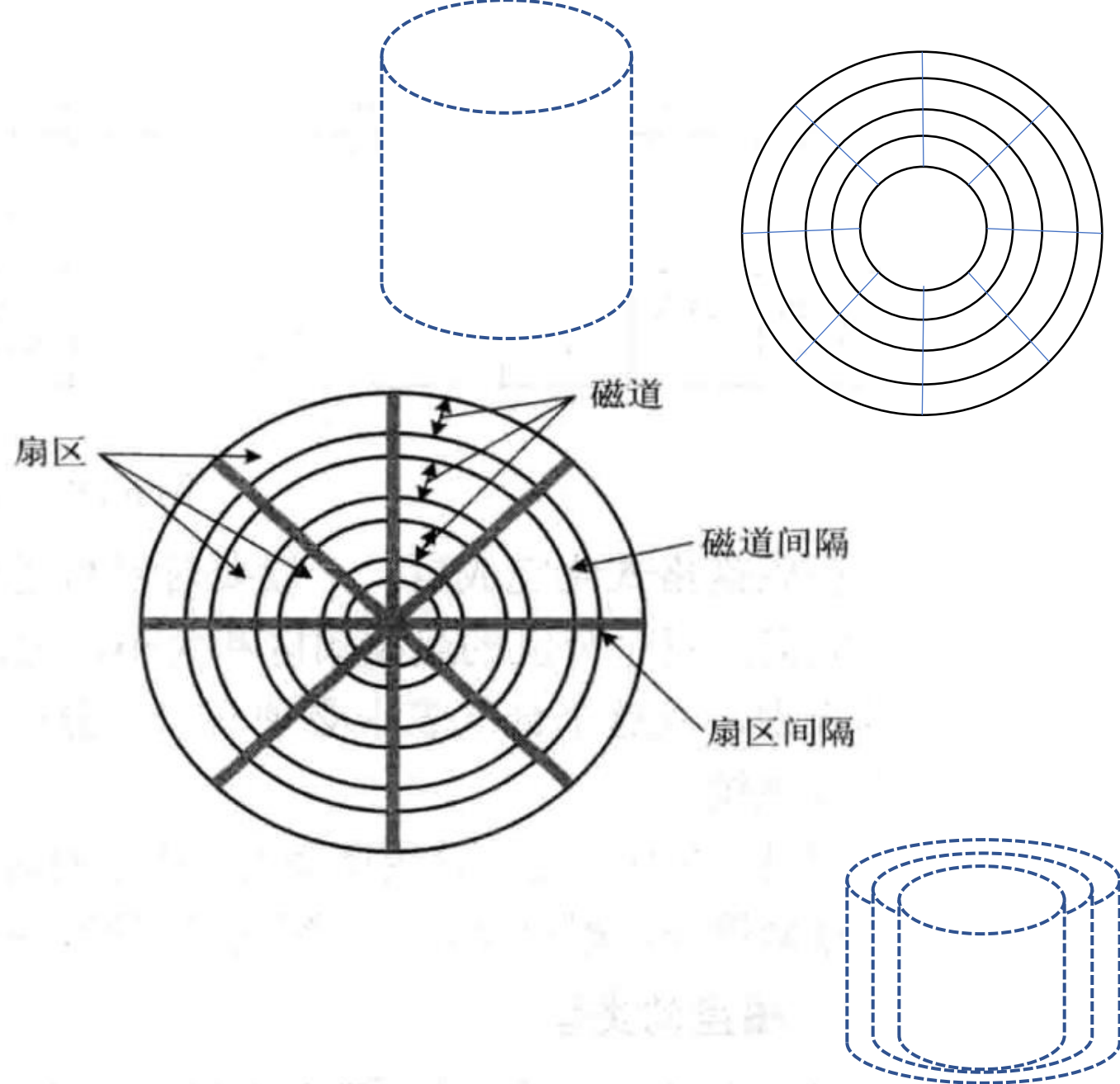
5.3.1 磁盘

磁盘（Disk）是由表面涂有磁性物质的物理盘片，通过一个称为磁头的导体线圈从磁盘存取数据。在读/写操作期间，磁头固定，磁盘在下面高速旋转。如图 5.13 所示，磁盘盘面上的数据存储在—组同心圆中，称为磁道。每个磁道与磁头一样宽，一个盘面有上千个磁道。磁道又划分为几百个扇区，每个扇区固定存储大小，一个扇区称为一个盘块。相邻磁道及相邻扇区间通过—定的间隙分隔开，以避免精度错误。注意，由于扇区按固定圆心角度划分，所以密度从最外道向里道增加，磁盘的存储能力受限于最内道的最大记录密度。

磁盘安装在一个磁盘驱动器中，它由磁头臂、用于旋转磁盘的主轴和用于数据输入/输出的电子设备组成。如图 5.14 所示，多个盘片垂直堆叠，组成磁盘组，每个盘面对应一个磁头，所有磁头固定在一起，与磁盘中心的距离相同且一起移动。所有盘片上相对位置相同的磁道组成柱面。扇区是磁盘可寻址的最小单位，磁盘上能存储的物理块数目由扇区数、磁道数及磁盘面数决定，磁盘地址用“柱面号·盘面号·扇区号”表示。



(a) 磁盘驱动器的结构



(b) 磁盘的数据布局

真题 2016.9

2. 假设一个磁盘组有 100 个柱面，编号为 0-99，每个柱面有 32 个磁道，编号为 0-31，每个盘面有 16 个扇区，编号为 0-15，现采用位示图方法管理磁盘空间，令磁盘块号按柱面顺序和盘面顺序编排，请回答下列问题：

(1) 若采用 32 位的字组成位示图，共需要多少个字？

(2) 第 40 字的第 18 位对应于哪个柱面，哪个读写磁头和哪个扇区？

01. 假定有一个磁盘组共有 100 个柱面，每个柱面有 8 个磁道，每个磁道划分成 8 个扇区。现有一个 5000 条逻辑记录的文件，逻辑记录的大小与扇区大小相等，该文件以顺序结构被存放在磁盘组上，柱面、磁道、扇区均从 0 开始编址，逻辑记录的编号从 0 开始，文件信息从 0 柱面、0 磁道、0 扇区开始存放。试问，该文件编号为 3468 的逻辑记录应存放在哪个柱面的第几个磁道的第几个扇区上？

显式链接是指把用于链接文件各物理块的指针，从每个物理块的末尾中提取出来，显式地存放在内存的一张链接表中。该表在整个磁盘中仅设置一张，称为文件分配表 (File Allocation Table, FAT)。每个表项中存放链接指针，即下一个盘块号。文件的第一个盘块号记录在目录项“物理地址”字段中，后续的盘块可通过查 FAT 找到。例如，某磁盘共有 100 个磁盘块，存放了两个文件：文件“aaa”占三个盘块，依次是 2→8→5；文件“bbb”占两个盘块，依次是 7→1。其余盘块都是空闲盘块，则该磁盘的 FAT 表如图 4.8 所示。

不难看出，文件分配表 FAT 的表项与全部磁盘块一一对应，并且可以用一个特殊的数字 -1 表示文件的最后一块，可以用 -2 表示这个磁盘块是空闲的（当然也可指定为 -3, -4）。因此，FAT 不仅记录了文件各块之间的先后链接关系，同时还标记了空闲的磁盘块，操作系统也可以通过 FAT 对文件存储空间进行管理。当某进程请求操作系统分配一个磁盘块时，操作系统只需从 FAT 中找到 -2 的表项，并将对应的磁盘块分配给进程即可。

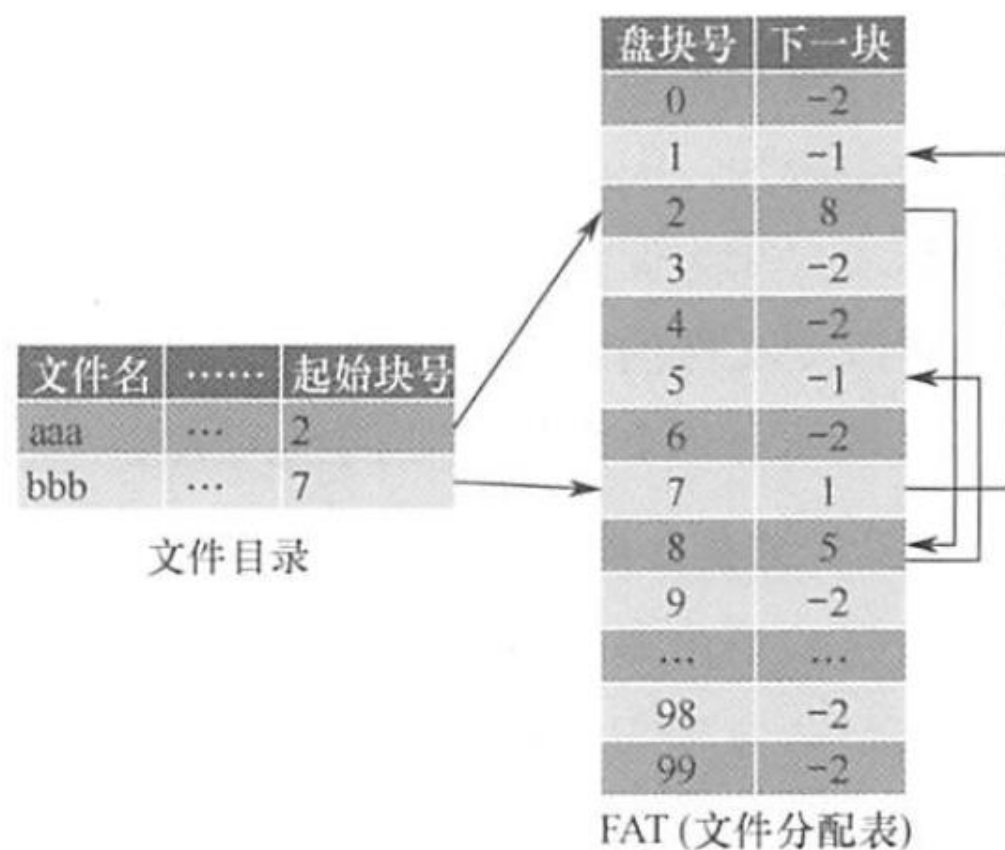


图 4.8 文件分配表

现在我们来计算以盘块为分配单位时所允许的最大磁盘容量。由于每个 FAT 表项为 12 位，因此，在 FAT 表中最多允许有 $4096(2^{12})$ 个表项；如果采用以盘块作为基本分配单位，每个盘块(也称扇区)的大小一般是 512 字节，那么，每个磁盘分区的容量为 2 MB($4096 \times 512 \text{ B}$)；一个物理磁盘能支持 4 个逻辑磁盘分区，所以相应的磁盘最大容量仅为 8 MB。这对最早时期的硬盘还可应付，但很快磁盘的容量就超过了 8 MB，FAT12 是否还可继续用呢，回答虽是肯定的，但需要引入一个新的分配单位——簇。

2) 以簇为单位的 FAT12 文件系统

稍加分析便可看出，如果把每个盘块(扇区)的容量增大 n 倍，则磁盘的最大容量便可增加 n 倍。但要增加盘块的容量是不方便和不灵活的。为此，引入了簇(cluster)的概念。簇是一组相邻的扇区，在 FAT 中它是作为一个虚拟扇区。在进行盘块分配时，是以簇作为分配的基本单位。簇的大小一般是 $2n(n$ 为整数)个盘块，在 MS-DOS 的实际运用中，簇的容量可以仅有一个扇区(512 B)、两个扇区(1 KB)、四个扇区(2 KB)、八个扇区(4 KB)等。一个簇应包含扇区的数量与磁盘容量的大小直接有关。例如，当一个簇仅有一个扇区时，磁盘的最大容量为 8MB；当一个簇包含了八个扇区时，磁盘的最大容量便可达到 64 MB。

FAT12 存在的主要问题是，随着支持硬盘的容量的增加，相应的簇内碎片也将随之成倍地增加，限制了磁盘的最大容量，通常只能是数十 MB。此外，FAT12 只能支持短文件名，即 8+3 格式的文件名。

408真题 2019

- 44.** (7 分) 某计算机系统磁盘有 300 个柱面，每个柱面有 10 个磁道，每个磁道有 200 个扇区，扇区大小为 512B。文件系统的每个簇包含 2 个扇区。请回答下列问题：
- 1) 磁盘的容量是多少？
 - 2) 假设磁头在 85 号柱面上，此时有 4 个磁盘访问请求，簇号分别为 100 260、60 005、101 660 和 110 560。若采用最短寻道时间优先 (SSTF) 调度算法，则系统访问簇的先后次序是什么？

王道题目

- P282 1
- P283 2

一次磁盘读写操作的时间由寻找（寻道）时间、旋转延迟时间和传输时间决定。

- 1) 寻找时间 T_s 。活动头磁盘在读写信息前，将磁头移动到指定磁道所需要的时间。这个时间除跨越 n 条磁道的时间外，还包括启动磁臂的时间 s ，即

$$T_s = m \times n + s$$

式中， m 是与磁盘驱动器速度有关的常数，约为 0.2ms，磁臂的启动时间约为 2ms。

- 2) 旋转延迟时间 T_r 。磁头定位到某一磁道的扇区所需要的时间，设磁盘的旋转速度为 r ，则

$$T_r = \frac{1}{2r}$$

对于硬盘，典型的旋转速度为 5400 转/分，相当于一周 11.1ms，则 T_r 为 5.55ms；对于软盘，其旋转速度为 300~600 转/分，则 T_r 为 50~100ms。

- 3) 传输时间 T_t 。从磁盘读出或向磁盘写入数据所经历的时间，这个时间取决于每次所读/写的字节数 b 和磁盘的旋转速度：

$$T_t = \frac{b}{rN}$$

式中， r 为磁盘每秒的转数， N 为一个磁道上的字节数。

在磁盘存取时间的计算中，寻道时间与磁盘调度算法相关；而延迟时间和传输时间都与磁盘旋转速度相关，且为线性相关，所以在硬件上，转速是磁盘性能的一个非常重要的参数。

总平均存取时间 T_a 可以表示为

$$T_a = T_s + \frac{1}{2r} + \frac{b}{rN}$$

虽然这里给出了总平均存取时间的公式，但是这个平均值是没有太大实际意义的，因为在实际的磁盘 I/O 操作中，存取时间与磁盘调度算法密切相关。

例题

04. 某软盘有 40 个磁道，磁头从一个磁道移至相邻磁道需要 6ms。文件在磁盘上非连续存放，逻辑上相邻数据块的平均距离为 13 磁道，每块的旋转延迟时间及传输时间分别为 100ms 和 25ms，问读取一个 100 块的文件需要多少时间？若系统对磁盘进行了整理，让同一文件的磁盘块尽可能靠拢，从而使逻辑上相邻数据块的平均距离降为 2 磁道，这时读取一个 100 块的文件需要多少时间？

(1) 先来先服务 (First Come First Served, FCFS) 算法

FCFS 算法根据进程请求访问磁盘的先后顺序进行调度, 这是一种最简单的调度算法, 如图 5.16 所示。该算法的优点是具有公平性。若只有少量进程需要访问, 且大部分请求都是访问簇聚的文件扇区, 则有望达到较好的性能; 若有大量进程竞争使用磁盘, 则这种算法在性能上往往接近于随机调度。所以, 实际磁盘调度中会考虑一些更为复杂的调度算法。

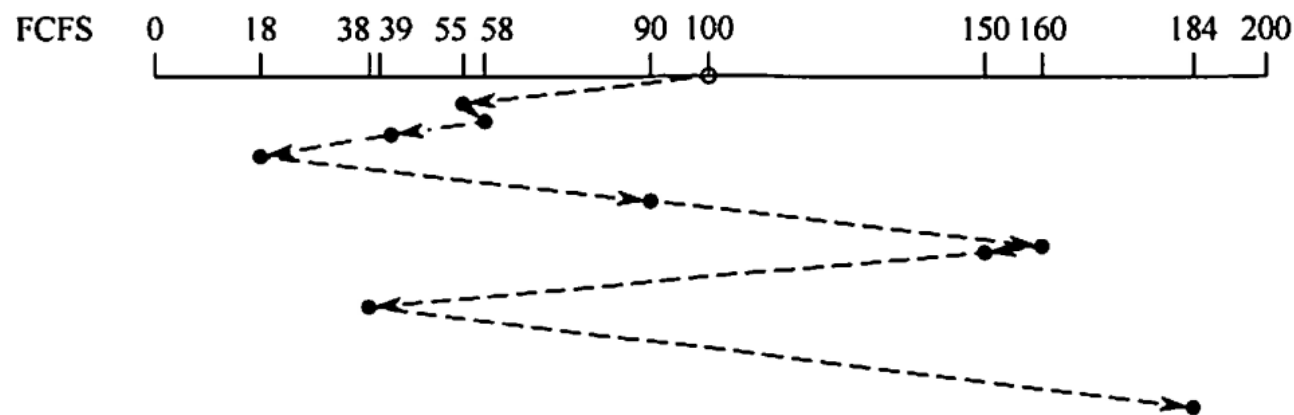


图 5.16 FCFS 磁盘调度算法

例如, 磁盘请求队列中的请求顺序分别为 55, 58, 39, 18, 90, 160, 150, 38, 184, 磁头的初始位置是磁道 100, 采用 FCFS 算法时磁头的运动过程如图 5.16 所示。磁头共移动了 $(45 + 3 + 19 + 21 + 72 + 70 + 10 + 112 + 146) = 498$ 个磁道, 平均寻找长度 $= 498/9 = 55.3$ 。

(2) 最短寻找时间优先 (Shortest Seek Time First, SSTF) 算法

SSTF 算法选择调度处理的磁道是与当前磁头所在磁道距离最近的磁道，以便使每次的寻找时间最短。当然，总是选择最小寻找时间并不能保证平均寻找时间最小，但能提供比 FCFS 算法更好的性能。这种算法会产生“饥饿”现象。如图 5.17 所示，若某时刻磁头正在 18 号磁道，而在 18 号磁道附近频繁地增加新的请求，则 SSTF 算法使得磁头长时间在 18 号磁道附近工作，将使 184 号磁道的访问被无限期地延迟，即被“饿死”。

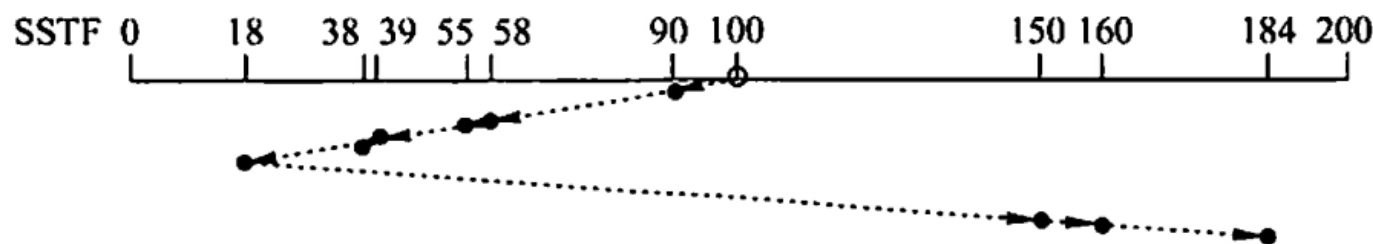


图 5.17 SSTF 磁盘调度算法

例如，磁盘请求队列中的请求顺序分别为 55, 58, 39, 18, 90, 160, 150, 38, 184，磁头初始位置是磁道 100，采用 SSTF 算法时磁头的运动过程如图 5.17 所示。磁头共移动了 $10 + 32 + 3 + 16 + 1 +$

对于给定的一个磁道访问序列，先响应最近的请求

(3) 扫描 (SCAN) 算法 (又称电梯调度算法)

SCAN 算法在磁头当前移动方向上选择与当前磁头所在磁道距离最近的请求作为下一次服务的对象, 实际上就是在最短寻找时间优先算法的基础上规定了磁头运动的方向, 如图 5.18 所示。由于磁头移动规律与电梯运行相似, 因此又称电梯调度算法。SCAN 算法对最近扫描过的区域不公平, 因此它在访问局部性方面不如 FCFS 算法和 SSTF 算法好。

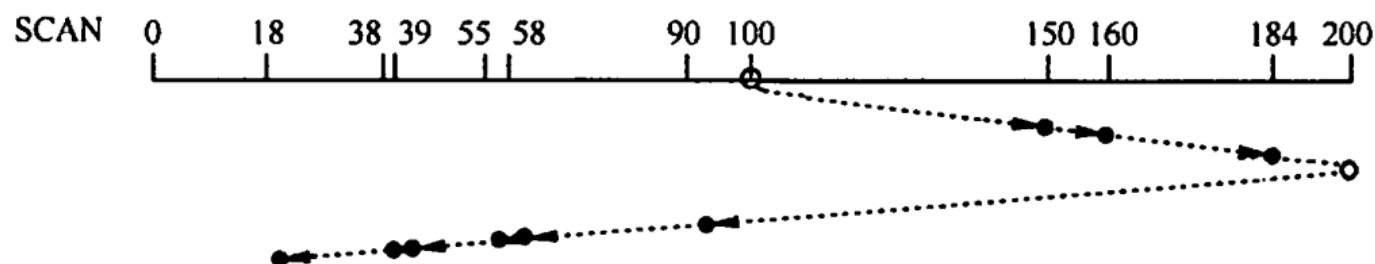


图 5.18 SCAN 磁盘调度算法

例如, 磁盘请求队列中的请求顺序分别为 55, 58, 39, 18, 90, 160, 150, 38, 184, 磁头初始位置是磁道 100。采用 SCAN 算法时, 不但要知道磁头的当前位置, 而且要知道磁头的移动方向, 假设磁头沿磁道号增大的顺序移动, 则磁头的运动过程如图 5.18 所示。移动磁道的顺序为 100, 150, 160, 184, 200, 90, 58, 55, 39, 38, 18。磁头共移动了 $(50 + 10 + 24 + 16 + 110 + 32 + 3 + 16 + 1 + 20) =$

SCAN 往返迂回

(4) 循环扫描 (Circular SCAN, C-SCAN) 算法

在扫描算法的基础上规定磁头单向移动来提供服务, 回返时直接快速移动至起始端而不服务任何请求。由于 SCAN 算法偏向于处理那些接近最里或最外的磁道的访问请求, 所以使用改进型的 C-SCAN 算法来避免这个问题, 如图 5.19 所示。

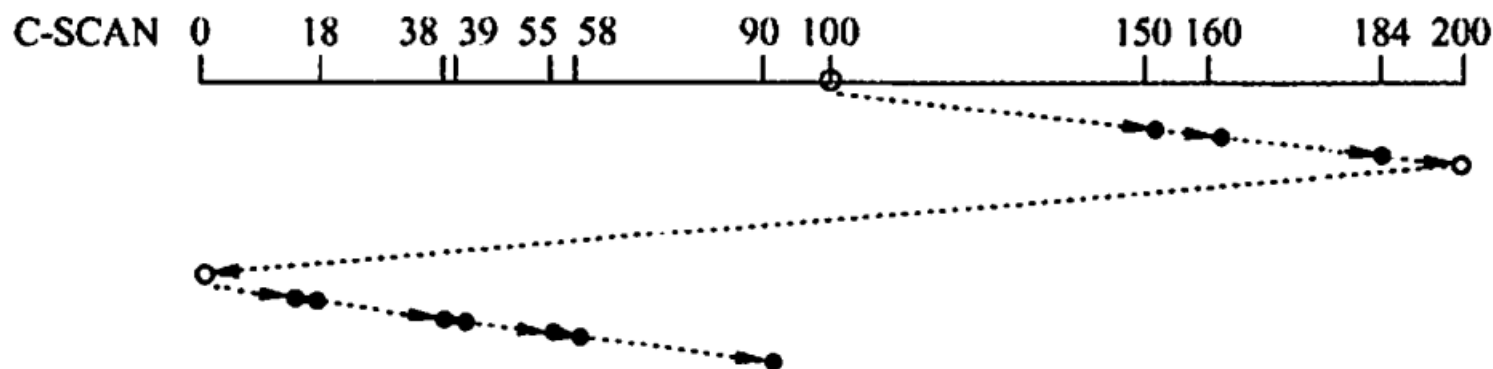


图 5.19 C-SCAN 磁盘调度算法

C-SCAN 单向循环

采用 SCAN 算法和 C-SCAN 算法时，磁头总是严格地遵循从盘面的一端到另一端，显然，在实际使用时还可以改进，即磁头移动只需要到达最远端的一个请求即可返回，不需要到达磁盘端点。这种形式的 SCAN 算法和 C-SCAN 算法称为 LOOK 调度（见图 5.20）和 C-LOOK（见图 5.21）调度，因为它们在朝一个给定方向移动前会查看是否有请求。

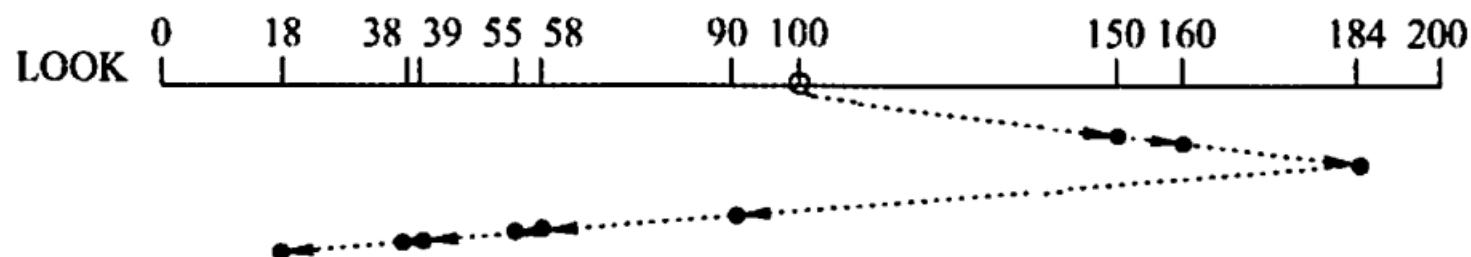


图 5.20 LOOK 磁盘调度算法

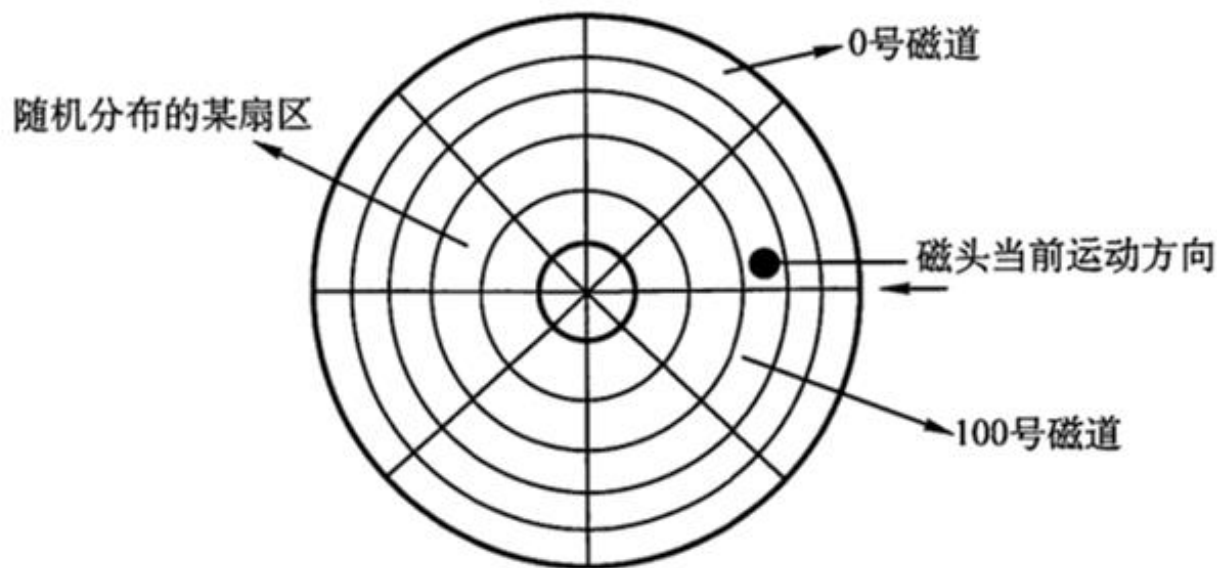
SCAN --- LOOK 往返迂回 不触碰边缘

C-SCAN --- C-LOOK 单向循环 不触碰边缘

408真题 2010

45. (7分) 假设计算机系统采用 CSCAN (循环扫描) 磁盘调度策略, 使用 2KB 的内存空间记录 16384 个磁盘块的空闲状态。

- 1) 请说明在上述条件下如何进行磁盘块空闲状态的管理。
- 2) 设某单面磁盘旋转速度为 6000rpm, 每个磁道有 100 个扇区, 相邻磁道间的平均移动时间为 1ms。若在某时刻, 磁头位于 100 号磁道处, 并沿着磁道号增大的方向移动 (见下图), 磁道号请求队列为 50, 90, 30, 120, 对请求队列中的每个磁道需读取 1 个随机分布的扇区, 则读完这 4 个扇区点共需要多少时间? 要求给出计算过程。



练习

4. 应用题

某活动头磁盘有 200 个磁道，编号为 0~199。磁头当前在 143 道服务。对于请求序列 86、147、91、177、94、150、102、175、130，求在下列调度策略下的寻道顺序及寻道长度。

(1) FCFS

(2) SSTF

(3) SCAN (磁头移动方向先从小到大)

(4) CSCAN (磁头移动方向先从小到大)

王道题目

- P323 1, 2, 3, 4, 6, 7
- 细水长流

408真题 2018

46. (7 分) 某文件系统采用索引结点存放文件的属性和地址信息，簇大小为 4KB。每个文件索引结点占 64B，有 11 个地址项，其中直接地址项 8 个，一级、二级和三级间接地址项各 1 个，每个地址项长度为 4B。请回答下列问题。
- 1) 该文件系统能支持的最大文件长度是多少？（给出计算表达式即可。）
 - 2) 文件系统用 1M ($1\text{M} = 2^{20}$) 个簇存放文件索引结点，用 512M 个簇存放文件数据。若一个图像文件的大小为 5600B，则该文件系统最多能存放多少个图像文件？

索引|结点

文件目录通常存放在磁盘上，当文件很多时，文件目录会占用大量的盘块。在查找目录的过程中，要先将存放目录文件的第一个盘块中的目录调入内存，然后用给定的文件名逐一比较，若未找到指定文件，就还需要不断地将下一盘块中的目录项调入内存，逐一比较。我们发现，在检索目录的过程中，只用到了文件名，仅当找到一个目录项（其中的文件名与要查找的文件名匹配）时，才需从该目录项中读出该文件的物理地址。也就是说，在检索目录时，文件的其他描述信息不会用到，也不需要调入内存。因此，有的系统（如 UNIX，图 4.2）便采用了文件名和文件描述信息分开的方法，使文件描述信息单独形成一个称为索引结点的数据结构，简称 i 结点（inode）。在文件目录中的每个目录项仅由文件名和指向该文件所对应的 i 结点的指针构成。

文件名	索引结点编号
文件名 1	
文件名 2	
...	...

0 13 14 15

图 7-8 UNIX 的文件目录

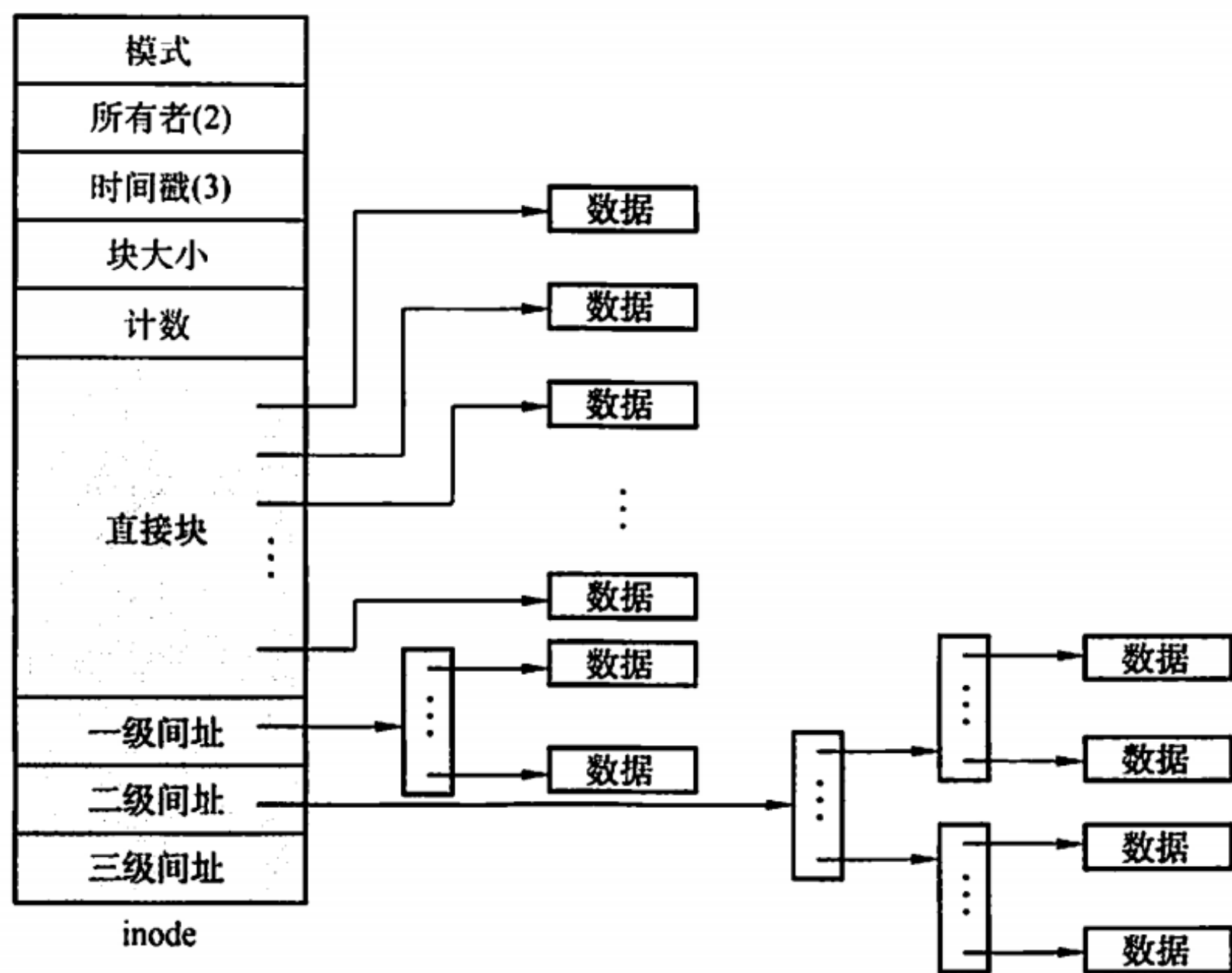


图 4.10 UNIX 系统的 inode 结构示意图

王道题目

- P252 1, 4, 5

3. 问答题

(1) 设某系统的磁盘空间共有 5000 块，若用位示图管理磁盘空间，位示图的每个字有 32 位，并且物理块号、字号、位号均从 1 开始。试问：

1) 位示图需要多少个字构成？

157

2) 计算位示图第 9 个字第 22 位对应的物理块号。

$8*32+22=278$

3) 求物理块号 106 对应的字和位。

第 4 个字 第 10 个位

(1)

FCFS 先来先服务

86、147、91、177、94、150、102、175、130

当前磁道 143	移动距离
86	57
147	61
91	56
177	86
94	83
150	56
102	48
175	73
130	45
平均寻道长度	62.78

(2)

86、147、91、177、94、150、102、175、130

SSTF 最短服务时间优先

当前磁道 143	移动距离
147	4
150	3
130	20
102	28
94	8
91	3
86	5
175	89
177	2
平均寻道长度	18

(3)



SCAN 扫描算法

86、147、91、177、94、150、102、175、130

当前磁道 143	移动距离
147	4
150	3
175	25
177	2
130	47
102	28
94	8
91	3
86	5
平均寻道长度	13.89

(4)

CSCAN 循环扫描算法

86、147、91、177、94、150、102、175、130

当前磁道 143	移动距离
147	4
150	3
175	25
177	2
86	91
91	5
94	3
102	8
130	28
平均寻道长度	18.78