

## 5.3 外存管理

本节有两个核心：磁盘和固态硬盘，它们是两种典型的外存设备。本节的概念较为琐碎，以理解和记忆为主，难度不大。考生在学习本节内容时，可以先尝试对磁盘结构有一个初步的认识，并尝试回答下列问题：

- (1) 什么是磁盘？什么是固态硬盘？
- (2) 磁盘和固态硬盘各有什么样的性能表现？

磁盘从出厂到使用需要经过低级格式化、分区和高级格式化，其中会涉及引导块和坏块的知识。磁盘调度算法是对磁盘寻道时间的优化，固态硬盘无需寻道，但是需要进行磨损均衡。请考生在学习过程中尝试回答下列问题（题目可以在各个小节中找到答案）：

- (1) 磁盘由哪些部件组成？它们是如何相互配合工作的？
- (2) 如何计算磁盘读取时间？
- (3) 四种磁盘调度算法分别是如何进行磁盘调度的？它们有哪些优缺点？
- (4) 为什么固态硬盘需要进行磨损均衡？动态磨损均衡和静态磨损均衡各有哪些优缺点？

### 5.3.1 磁盘

磁盘是一种利用磁记录技术存储数据的存储器，具有记录密度高、容量大、速度快等优点，它是目前计算机存储系统中使用最普遍的一种外部存储器。磁盘是以数据块为信息交换单位的设备，每次可与内存交换一个或多个数据块的信息，且支持操作系统直接存取磁盘上的任意物理块。

**【提示】**操作系统对磁盘等辅存的访问是以块为单位的，而对主存的访问是以字节或字为单位的。例如在分页存储管理方式中，访问主存不仅需要页号，而且需要页内偏移才可以访问。

#### 1. 磁盘结构

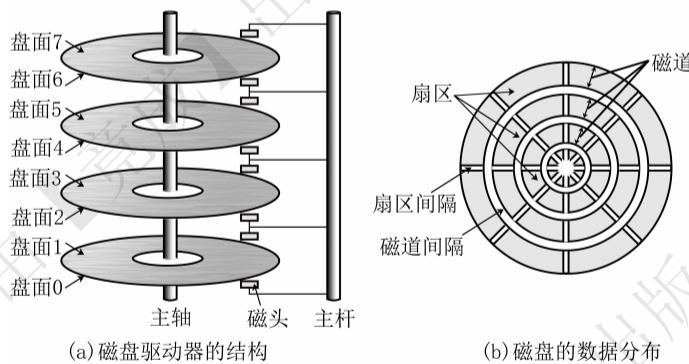


图 5.19 磁盘驱动器的结构

磁盘结构如图5.19所示（4个盘片，8个磁头，每个磁道8个扇区），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。传统磁盘通常是盘面旋转，而磁头在一定范围内移动。磁盘的相关概念如下：

- (1) **盘片：**由铝合金材料或玻璃构成的基片，盘片的表面涂有磁性物质，每个磁盘有一个或多个盘片。
- (2) **盘面：**盘片上涂有磁性物质的一面，每个盘片有一个或两个盘面。

(3) 磁头：对磁盘进行读写的部件，每个盘面对应一个磁头，因此盘面号又叫磁头号，所有磁头连在同一磁臂上，因此所有磁头只能“共进退”。

(4) 磁道：盘面上以盘片中心为圆心，不同半径的同心圆，每个盘面有若干个磁道，各磁道之间有必要的间隙。

(5) 柱面：所有盘面上由相同水平位置的磁道从上到下组成的圆柱体。

(6) 扇区：每个磁道可以被分为若干个弧段，一个弧段就是一个扇区，各扇区之间有必要的间隙。扇区是磁盘的最小组成单元，每个扇区存放的数据量相等，一般为 512 字节。

磁盘上的任意一个物理地址可以用（柱面号，盘面号，扇区号）来表示。磁盘的存储容量 = 磁头数（盘面数）× 磁道数（柱面数）× 每个磁道上的扇区数 × 每个扇区的字节数。

【提示】由于每个扇区存放的数据量相等，因此扇区所在的磁道越往内侧，扇区面积越小，扇区内的数据密度就越大，因此磁盘的存储能力由最内侧磁道的最大记录密度决定。

通常对磁盘扇区进行错位编号、对磁盘盘面进行错位命名，如图5.20和图5.21所示。磁头读完一个扇区的数据时，还需要一部分处理时间。由于此时盘片仍在旋转，无法连续读取下一个磁盘块，需等到盘片再转一圈才能够读取，因此错位编号和错位命名可以减少磁盘的旋转延迟时间。

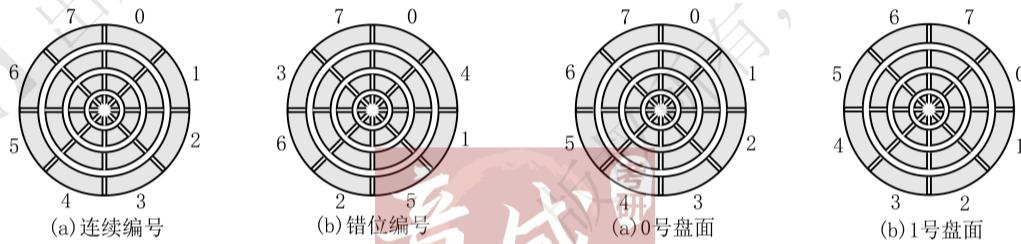


图 5.20 磁盘扇区的连续编号与错位编号

图 5.21 磁盘盘面的错位命名

## 2. 磁盘格式化和分区

磁盘格式化可以分为低级格式化和高级格式化两种，磁盘分区则是磁盘高级格式化前的必要步骤。

### (1) 低级格式化

低级格式化，又称物理格式化，是指对一个新磁盘划分出柱面和磁道，再将磁道划分为若干个扇区。每个扇区又划分为头部、尾部和数据区域，其中头部和尾部包含了一些磁盘控制器的信息，用于保存扇区号、进行扇区校验。

低级格式化一般由磁盘生产厂家在出厂前完成。如果某个扇区在出厂前已经损坏，且无法通过纠错算法恢复，则低级格式化过程将对该扇区进行标记，防止操作系统进行读写。低级格式化只能针对整块磁盘，不能对某个分区单独进行。

【拓展】当磁盘出现大量坏扇区时，可以通过低级格式化来对扇区重新划分，从而修复磁盘。但低级格式化是一种损耗性操作，会对磁盘寿命产生一定的负面影响，非必要应避免。

### (2) 分区

分区是指把一个物理磁盘的存储空间划分为几个相互独立的部分，由分区起始和终止的磁头、柱面、扇区、总的扇区数等信息来控制分区的边界。在逻辑上，每个分区就是一个独立的逻辑磁盘。

一个磁盘可以有一个或多个分区，不同的分区可以使用不同的文件系统。对磁盘进行分区有利于文件的分类管理，将系统文件和数据文件隔离开。如果操作系统需要从磁盘引导启动，就必须有一个分区被标记为活动分区。

③ 竟成针对26考研推出答疑班、全程班（包含全套课程、答疑、以及一对一的指导），报班联系qq客服：3491399761。  
④ 竟成书籍的勘误，书籍相关问题的反馈，关注竟成官方B站（账号：竟成408计算机考研）及官方公众号（账号：竟成408）。

### (3) 高级格式化

高级格式化，又称逻辑格式化，是对磁盘分区创建文件系统，包括创建文件系统的根目录、初始化存储空间管理所用的数据结构（如位示图、空闲分区表）等，使操作系统能够对磁盘进行读写。高级格式化会删除硬盘驱动器上的数据，生成引导区信息、初始化文件分配表、标注逻辑坏道、校验数据等。高级格式化的方法因操作系统的不同而有所差异。

【提示】低级格式化针对的是硬盘，高级格式化针对的是分区，一块新磁盘要经过低级格式化、分区、高级格式化，才可以正常使用。

## 3. 磁盘引导块和坏块

### (1) 引导块

引导块包含引导操作系统的代码。每个分区都有一个引导块，引导块中的程序将装载该分区中的操作系统。为统一起见，每个分区都从引导块开始，如果分区不包含操作系统，则引导块的内容可以为空。

计算机启动时需要先运行一个自举程序（初始化程序），该程序通常位于只读存储器（ROM）中。由于改变 ROM 中的引导代码需要改变 ROM 的硬件芯片，因此 ROM 中只包含自举程序中固定不变的一部分，而把完整的自举程序放在磁盘固定位置的引导块中，以便引导程序的更新。

### (2) 坏块

坏块是无法正常使用的扇区，属于硬件故障。磁盘在生产时就有可能存在坏块，出厂前的低级格式化会对磁盘上的坏块进行标记，并用磁盘末尾的一些空闲块进行替换，操作系统无法感知。在出厂后，仍然会有正常的磁盘块变成坏块，这种情况则可以被操作系统检测到。

坏块只能进行屏蔽，不能进行修复，可以在磁盘控制器或从操作系统层面对坏块进行处理。磁盘控制器可以将坏块重新映射到出厂预留的空闲块中，而操作系统可以在逻辑格式化时对整个磁盘进行坏块检查，标记哪些块属于坏块，例如在 FAT 表上标明以避免使用。

## 4. 磁盘读取时间

磁盘的读取时间由寻道时间、旋转延迟、传输时间共同决定。

(1) 寻道时间  $T_s$ ：磁头移动到数据所在磁道需要的时间，该时间是启动磁臂的时间  $s$  与磁头移动  $n$  条磁道所花费的时间之和，即：

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

其中  $m$  为常数，与磁盘驱动器的速度有关，一般磁盘的  $m = 0.2\text{ms}$ ，磁臂的启动时间  $s$  约为  $2\text{ms}$ 。

(2) 旋转延迟时间  $T_r$ ：磁头移动到指定磁道后，扇区移动到磁头下面需要的时间，设磁盘的旋转速度为  $r$ ，则：

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

如对于转速为  $7200\text{r/min}$  的硬盘， $T_r = 1/2 \times 7200\text{r/min} \approx 4.17\text{ms}$

(3) 传输时间  $T_t$ ：把数据从磁盘读出，或向磁盘写入数据需要的时间， $T_t$  的大小与每次所读写的字节数  $b$  和旋转速度  $r$  有关：

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

其中， $r$  为磁盘每秒钟的转数； $N$  为一条磁道上的字节数。因此，访问时间  $T_a$  可表示为：

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

③62竟成针对26考研推出答疑班、全程班（包含全套课程、答疑、以及一对一的指导），报班联系qq客服：3491399762

④ 竟成书籍的勘误，书籍相关问题的反馈，关注竟成官方B站（账号：竟成408计算机考研）及官方公众号（账号：竟成408）。

## 5. 磁盘调度算法

由于磁盘的读写涉及磁头的移动，磁头移动产生的寻道时间是磁盘读取时间的主要部分，而旋转延迟时间和传输时间几乎无法从操作系统层面优化，因此磁盘调度的目标是使磁盘的平均寻道时间尽可能少。为了减少寻道时间，应尽可能找到一种磁盘调度的最佳算法，目前常用的磁盘调度算法有先来先服务、最短寻道时间优先和扫描等算法。

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

先来先服务算法依据进程请求访问磁盘的先后次序进行调度。该算法实现简单，对进程公平，不会导致进程饥饿。但该算法未对寻道进行优化，平均寻道时间可能较长。

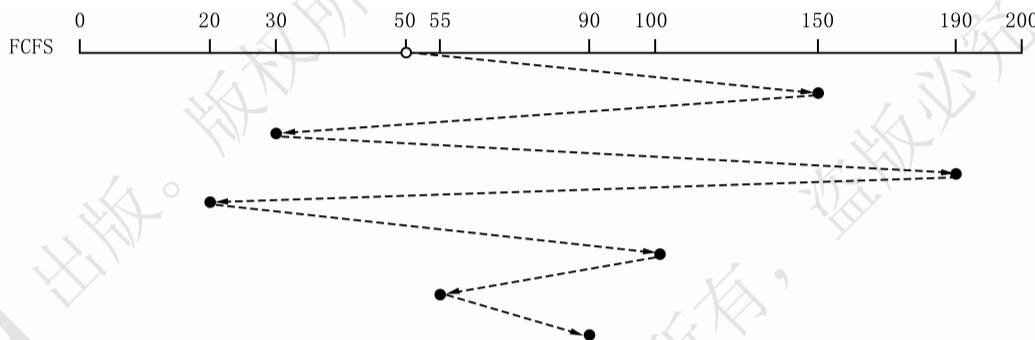


图 5.22 FCFS 磁盘调度算法

如图5.22所示，磁盘请求队列中的请求顺序分别为：50, 150, 30, 190, 20, 100, 55, 90，磁头的初始位置是磁道50，访问磁盘的实际顺序与请求顺序相同。采用该算法，磁头移动磁道总数 $=(150-50)+(150-30)+(190-30)+(190-20)+(100-20)+(100-55)+(90-55)=710$ ，平均寻道长度 $=710/8=88.75$ 。

### (2) 最短寻道时间优先 (Short Seek Time First, SSTF) 算法

最短寻道时间优先算法每次寻找磁盘访问请求中与当前磁头最近的磁道，使每次寻道时间最短。但该算法会导致进程饥饿，且仍然不能保证平均寻道时间最短。假设有大量靠近当前磁道的磁盘访问请求，则请求时间早且距离当前磁头较远的磁盘访问请求将长期无法得到满足，从而出现“饥饿”现象。

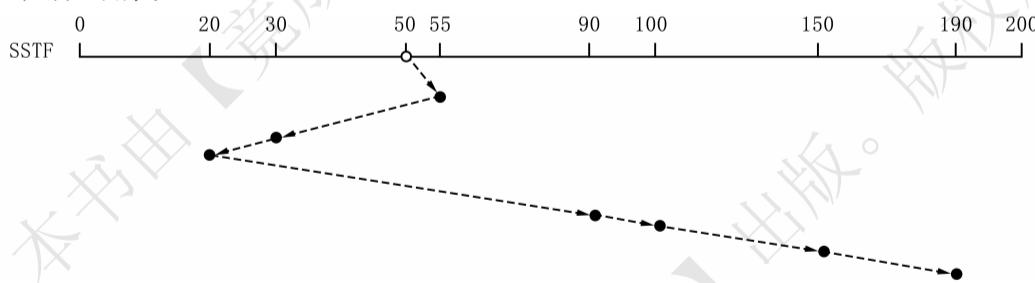


图 5.23 SSTF 磁盘调度算法

如图5.23所示，磁盘请求队列中的请求顺序分别为：50, 150, 30, 190, 20, 100, 55, 90，磁头的初始位置是磁道50，访问磁盘的实际顺序分别为：50, 55, 30, 20, 90, 100, 150, 190。采用该算法，磁头移动磁道总数 $=(55-50)+(55-30)+(30-20)+(90-20)+(100-90)+(150-100)+(190-150)=210$ ，平均寻道长度 $=210/8=26.25$ 。

### (3) 扫描 (SCAN) 算法

扫描算法，又称电梯算法。每次在磁头移动方向上，选择与当前磁头最近的磁道作为访问对象。直至到达这个方向上的最后一个磁道，再沿反方向按照上述规则选择访问对象。该算法不会

③ 竟成针对26考研推出答疑班、全程班（包含全套课程、答疑、以及一对一的指导），报班联系qq客服：3491399763。

④ 竟成书籍的勘误，书籍相关问题的反馈，关注竟成官方B站（账号：竟成408计算机考研）及官方公众号（账号：竟成408）。

导致进程饥饿，且平均寻道时间较短，但对最近访问过的区域不公平。例如：当磁头刚越过某个磁道，而新的磁盘访问请求位于该磁道时，进程可能会面临较长时间的等待。

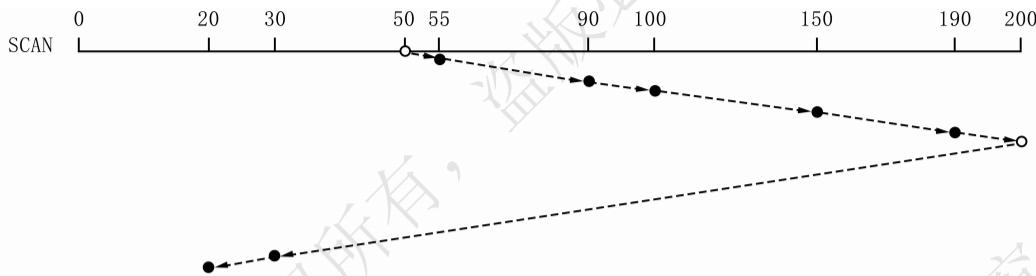


图 5.24 SCAN 磁盘调度算法

如图5.24所示，磁盘请求队列中的请求顺序分别为：50, 150, 30, 190, 20, 100, 55, 90，磁头的初始位置是磁道50，访问磁盘的实际顺序分别为：50, 55, 90, 100, 150, 190, 30, 20。采用该算法，磁头移动磁道总数 =  $(55 - 50) + (90 - 55) + (100 - 90) + (150 - 100) + (190 - 150) + (200 - 190) + (200 - 30) + (30 - 20) = 330$ ，平均寻道长度 =  $330/8 = 41.25$ 。

【提示】按照 SCAN 磁盘调度算法的规则，即使没有访问请求也须扫描到最后一个可扫描的磁道，因此到达 190 后还需继续扫描到 200。

#### (4) 循环扫描 (Circular SCAN, CSCAN) 算法

循环扫描算法在 SCAN 算法的基础上，加了磁头只能在一个方向上提供服务这个限定条件。当磁头访问某个方向的最后一个磁道时，磁头臂回到反方向的起始端重新扫描（回返时不响应请求）。这减少了新请求的最大延迟，避免了两侧磁道的长时间等待。

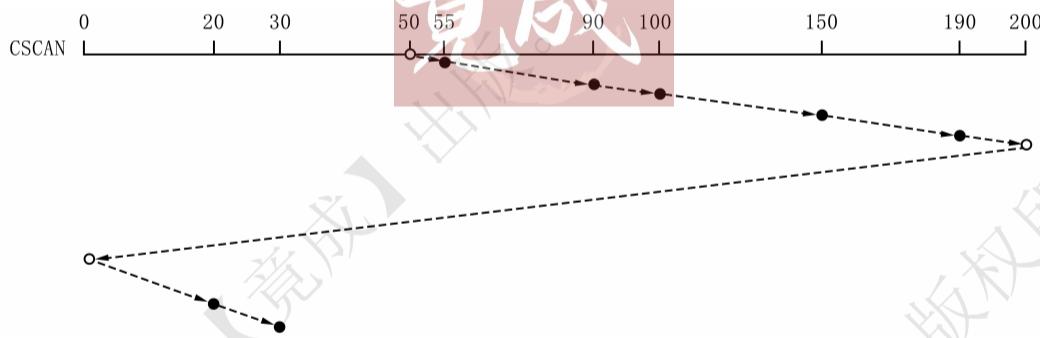


图 5.25 CSCAN 磁盘调度算法

如图5.25所示，磁盘请求队列中的请求顺序分别为：50, 150, 30, 190, 20, 100, 55, 90，磁头的初始位置是磁道50，访问磁盘的实际顺序分别为：50, 55, 90, 100, 150, 190, 20, 30。采用该算法，磁头移动磁道总数 =  $(55 - 50) + (90 - 55) + (100 - 90) + (150 - 100) + (190 - 150) + (200 - 190) + (200 - 0) + (20 - 0) + (30 - 20) = 380$ ，平均寻道长度 =  $380/8 = 47.5$ 。

对于 SCAN 算法和 CSCAN 算法，无论是否有访问请求，磁头都要移动到最后一个磁道。因此对这两种算法做了改进，磁头只需移动到最远端的一个请求即可返回，不需要到达最后一个磁道，改进后的算法称为 **LOOK** 算法和 **CLOOK** 算法。

【提示】如无特殊说明，一般默认 SCAN 算法和 CSCAN 算法为改进后的 LOOK 算法和 CLOOK 算法。

四种磁盘调度算法的优缺点如表5.4所示：

表 5.4 四种磁盘调度算法的优缺点

算法	优点	缺点
FCFS 算法	公平、简单	未对寻道进行优化，适合磁盘 I/O 较少的情况
SSTF 算法	相比 FCFS 算法有更好的性能	不能保证平均寻道时间最短，可能出现饥饿现象
SCAN 算法	解决了 SSTF 的饥饿现象	不利于远离磁头一端出现的磁盘 I/O 请求
CSCAN 算法	解决了 SCAN 的不公平现象	若某个磁道出现连续请求，磁臂将不再移动

### 5.3.2 固态硬盘

固态硬盘 (**Solid State Disk, SSD**) 的接口规范和使用方法与磁盘几乎相同。相比于磁盘而言，固态硬盘不是磁表面存储器，而是由 NAND 闪存组成的存储设备，具有读写快、质量轻、能耗低、体积小等优点。

#### 1. 读写性能特性

固态硬盘一般使用 **NAND 闪存** 来作为存储介质，一块固态硬盘中一般有多个 NAND 闪存，每个 NAND 闪存中包含多个块 (**Block**)，每个块包含多个页 (**Page**)。由于 NAND 闪存的特性，固态硬盘必须以页为单位进行存取。

NAND 闪存还有一个重要的特性：只能读写单个页，不允许覆盖写入。如果需要写入某个页，必须先清空里面的内容才能写入。由于清空内容需要的电压较高，又必须是以块为单位来清空。因此，在没有空闲的页时，必须要找到不存在有效内容的块并对其清空，然后再选择空闲的页进行写入。

固态硬盘一般会维护一个逻辑地址到物理地址的映射表，在每次读写时，可以通过逻辑地址直接查表找到物理地址。这与磁盘相比省去了寻道时间和旋转时间，因此固态硬盘可以取得很好的读写性能。

从 NAND 闪存的原理可以看出，固态硬盘和磁盘的主要区别如下：

(1) **定位数据块**：磁盘需要经过寻道和旋转，才能定位到要读写的数据块，而固态硬盘只需通过映射表计算即可找到数据块。

(2) **读取速度快**：磁盘的速度取决于旋转速度，而固态硬盘的读写不涉及机械部件的移动，因此普遍快于磁盘。

#### 2. 磨损均衡

虽然固态硬盘拥有读写快、体积小等优点，但是 NAND 闪存中的块是有擦写次数限制的。当某个块超过一定的擦写次数后，这个块就只能读取，无法写入。为了使得固态硬盘的寿命最大化，需要在擦除次数上保持平衡，而固态硬盘具有的“磨损均衡”机制则可以实现这一目标。

**磨损均衡** 是一种平衡机制，用于均衡固态硬盘内部各个块的使用程度，优先对擦写次数少的块进行擦写，从而延长整个固态硬盘的使用寿命。对于固态硬盘，一个好的损耗均衡算法就很有必要，主流的方法有动态磨损均衡和静态磨损均衡两种，表 5.5 列举了它们的优缺点。

表 5.5 动态磨损均衡与静态磨损均衡的优缺点

机制	优点	缺点
动态磨损均衡	实现简单，读写占用资源小	优化不足，磨损均衡的效果有限
静态磨损均衡	优化全面，提高整体寿命	实现困难，会增加 NAND 擦写次数

③ 竟成针对26考研推出答疑班、全程班（包含全套课程、答疑、以及一对一的指导），报班联系qq客服：349139976365。

④ 竟成书籍的勘误，书籍相关问题的反馈，关注竟成官方B站（账号：竟成408计算机考研）及官方公众号（账号：竟成408）。