

第五章

设备管理

方 钰



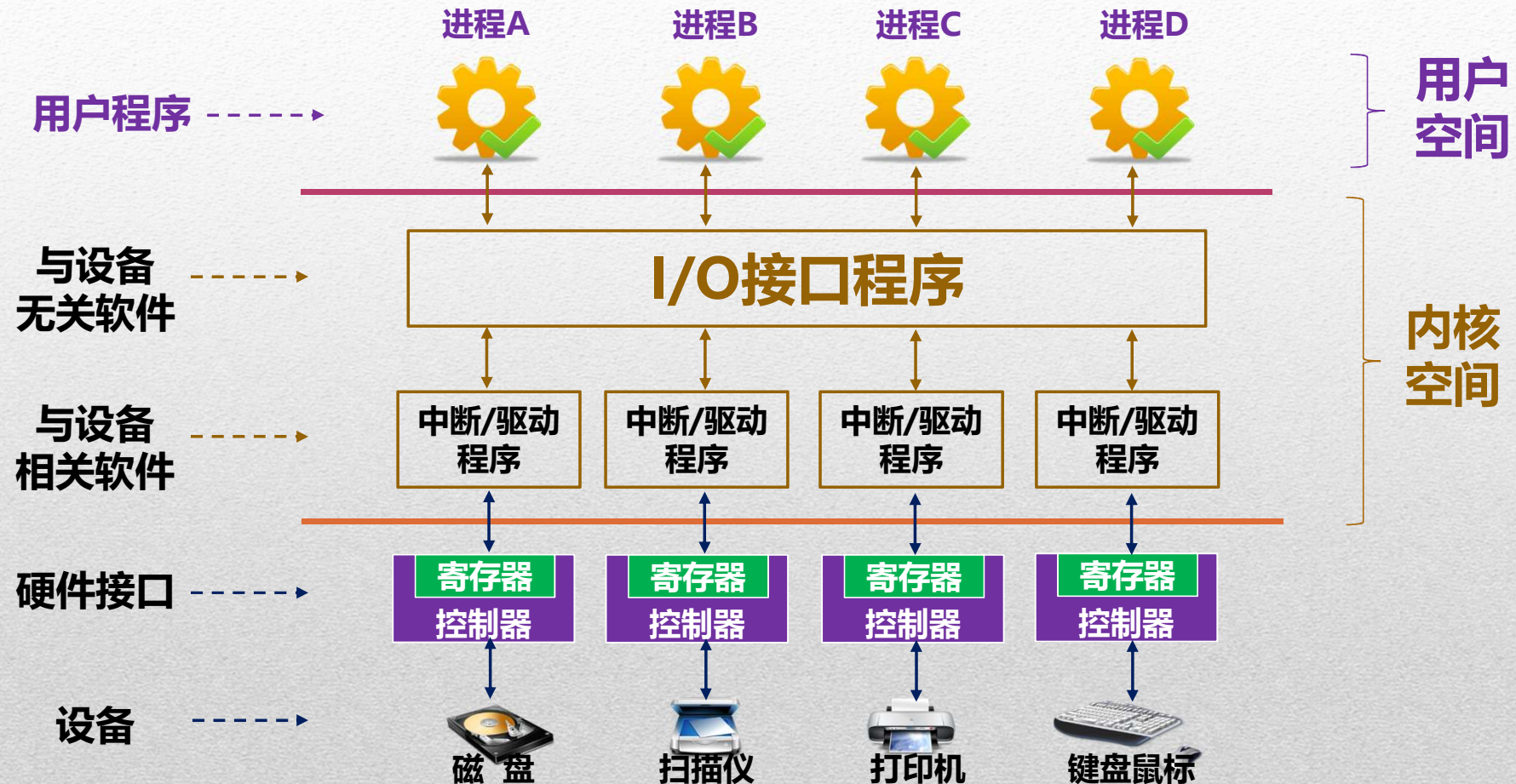
主要内容

5.1 I/O系统

5.2 磁盘存储器管理

5.3 UNIX字符块设备管理

I/O系统的组织结构



I/O系统的组织结构





I/O设备的分类

按设备的使用特性分类

- **存储设备**：外存、后备存储器。存取速度较内存慢，但容量比内存大得多，价格便宜。
- **输入/输出设备**：键盘、鼠标、扫描仪、视频摄像；打印机、绘图仪、显示器、音像输出设备等。

按设备的传输速率分类

- **低速设备**：传输速率为每秒几个字节至数百个字节。典型设备有键盘、鼠标器、语音的输入和输出等。
- **中速设备**：传输速率在每秒钟数千个字节至数万个字节。典型设备有行式打印机、激光打印机等。
- **高速设备**：传输速率在数万个字节至数十兆字节。典型的高速设备有磁带机、磁盘机、光盘机等。



I/O设备的分类

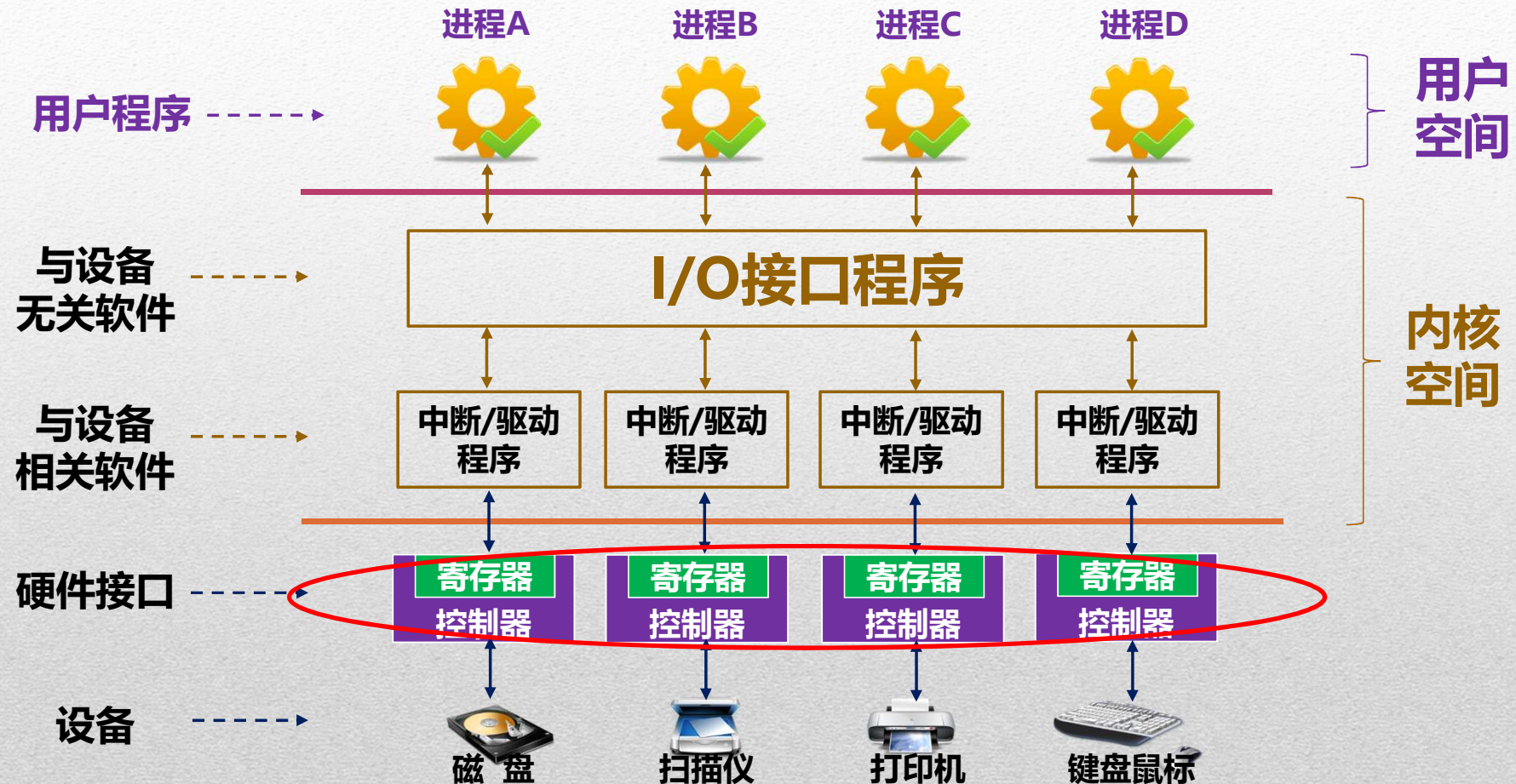
按信息交换的单位分类

- **块设备**：信息的存取总是以数据块为单位。典型的块设备是磁盘。其**传输速率较高**，通常每秒钟为几兆位；**可寻址**，即可随机地读/写任一块。
- **字符设备**：信息的存取基本单位是字符。**传输速率较低**，通常每秒几个字节至数千个字节；**不可寻址**。

按设备的共享属性分类

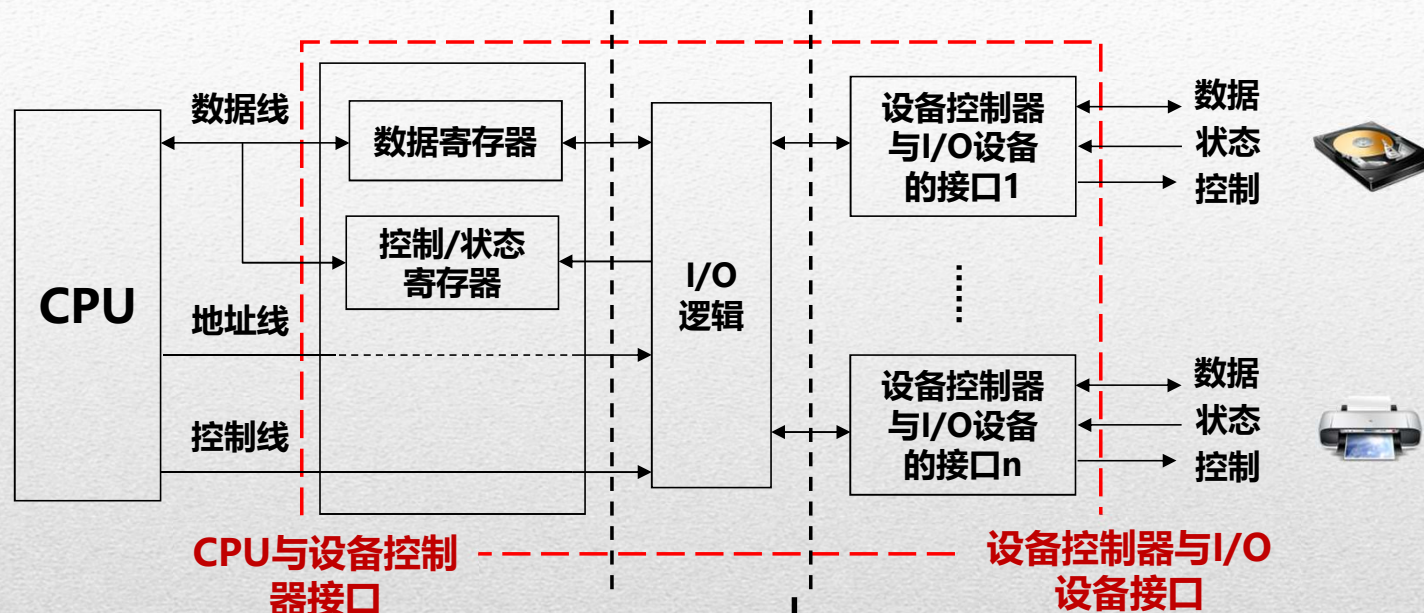
- **独占设备**：临界资源，即一段时间内只允许一个用户（进程）访问。
- **共享设备**：在一段时间内允许多个进程并发访问的设备。
- **虚拟设备**：利用大容量辅助存储器把独享设备**改造**成为能被多个进程共享的设备，以提高独享设备的利用率。是一种逻辑上的I/O设备。

I/O系统的组织结构



设备控制器

在微机中，它通常是一块可插入主板扩展槽的电路板，也叫接口。是CPU与I/O设备之间的硬件接口，接收从CPU发来的命令，去控制一个或多个设备。



- 接收识别命令（控制线、控制寄存器）
- 数据交换与缓冲（数据线、数据寄存器、缓冲器）
- 标识和报告设备的状态（状态寄存器）
- 差错控制
- 地址识别（地址译码器）

- 数据信号：双向，有缓存
- 控制信号：控制器给设备，要求其完成相关操作
- 状态信号：设备给控制器的当前状态信号（ready, busy, error 等）

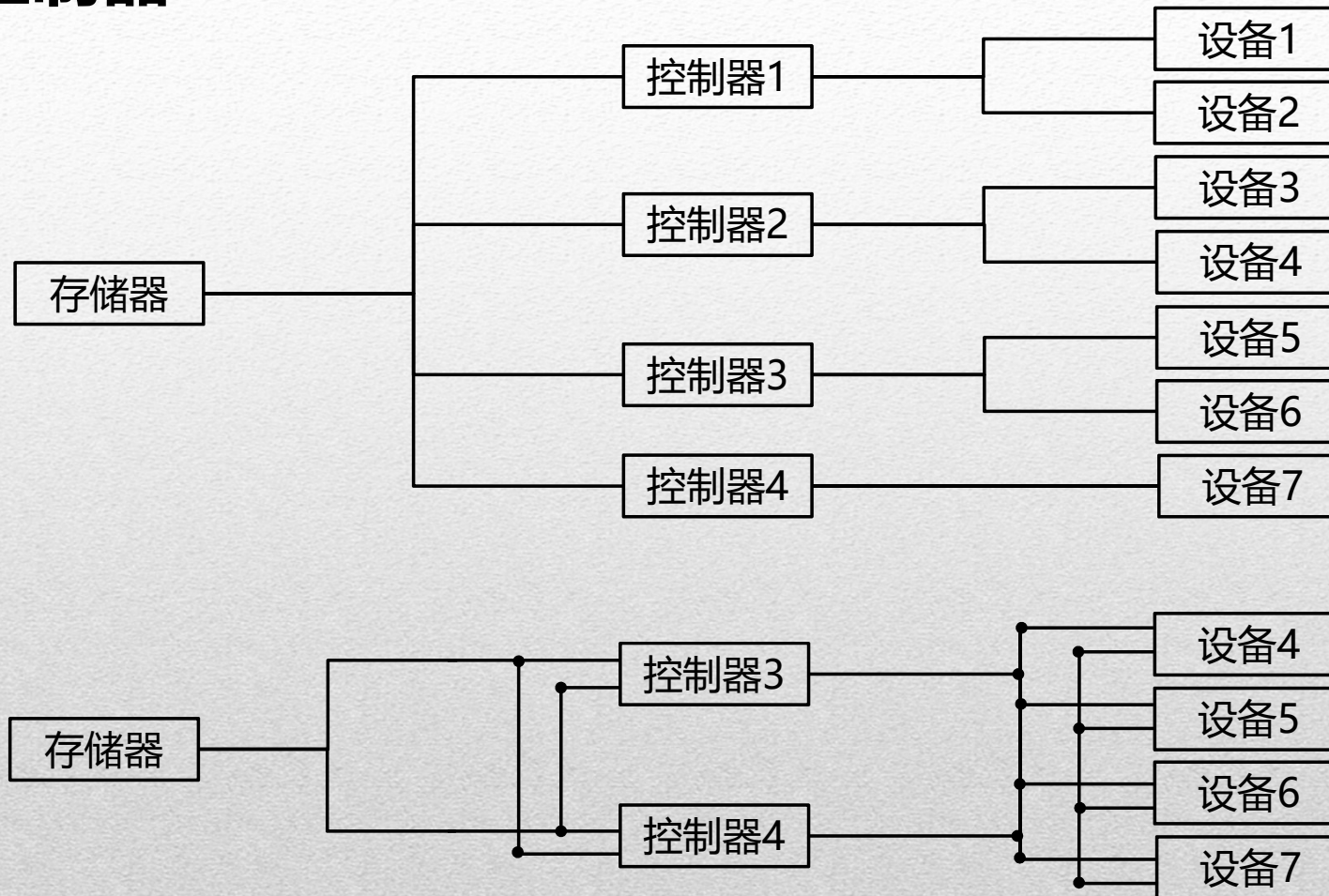


设备控制器





设备控制器



I/O系统的组织结构

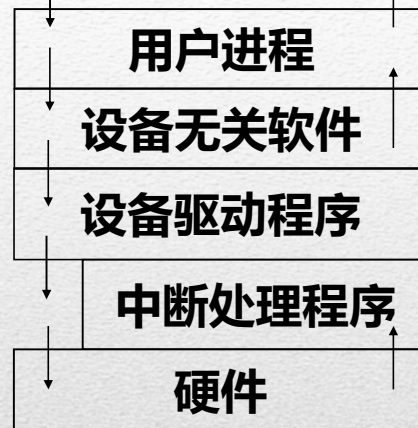




I/O软件层次结构

I/O请求

I/O应答



进行I/O调用; 格式化I/O; SPOOLING

映射; 保护; 阻塞; 缓冲; 分配

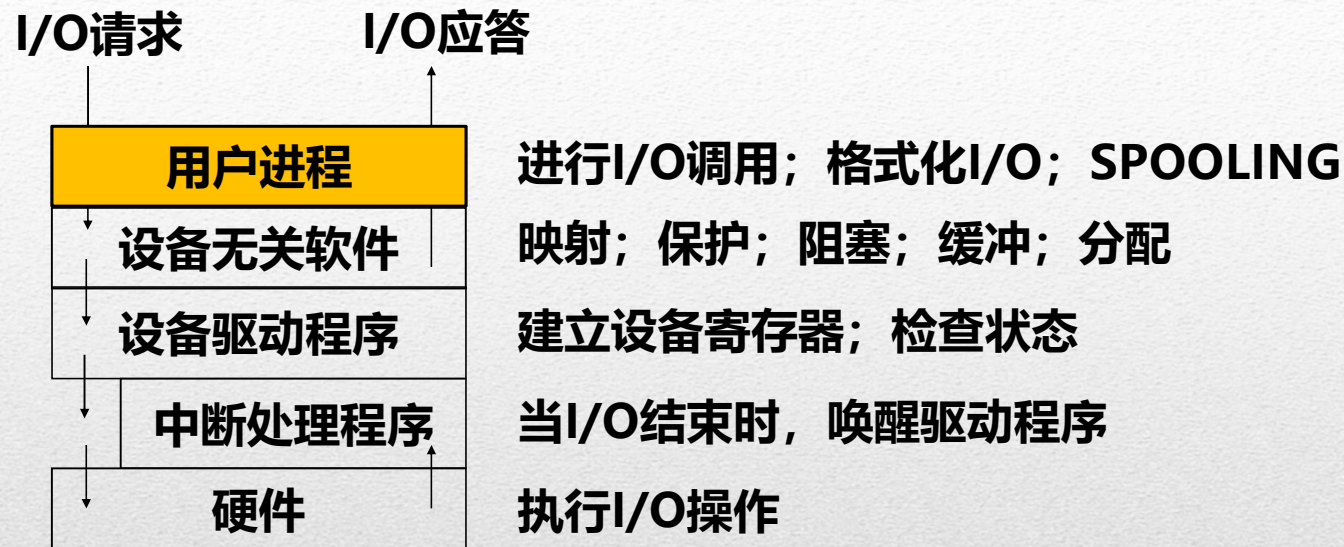
建立设备寄存器; 检查状态

当I/O结束时, 唤醒驱动程序

执行I/O操作



I/O软件层次结构



两种方式向内核提交I/O请求:

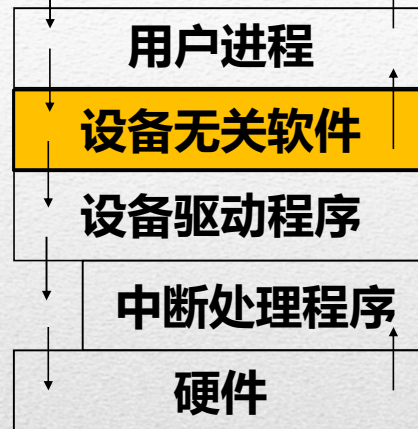
- 库函数
- I/O系统调用

用户空间的I/O软件

I/O软件层次结构

I/O请求

I/O应答



进行I/O调用；格式化I/O；SPOOLING

映射；保护；阻塞；缓冲；分配

建立设备寄存器；检查状态

当I/O结束时，唤醒驱动程序

执行I/O操作

- 逻辑设备名到物理设备名的映射，确定相应物理设备的驱动程序
- 设备的分配和释放
- 设备保护，禁止用户直接访问设备
- 缓冲管理与差错控制
- 向用户层软件提供统一接口

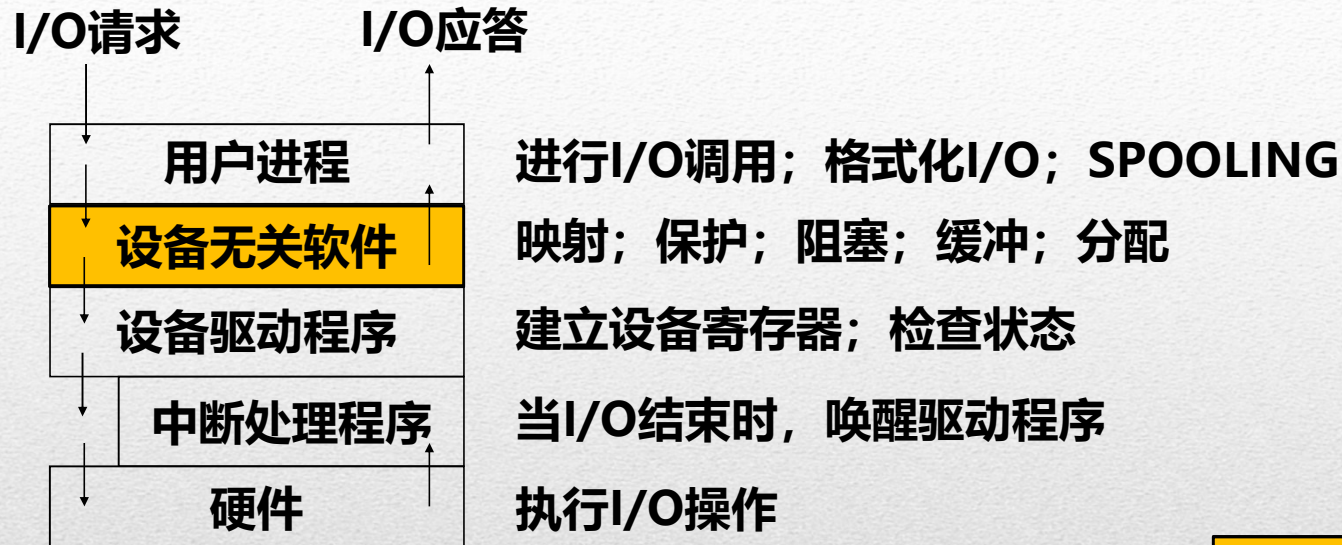
设备无关软件

设备独立性：应用程序独立于具体使用的物理设备

- 逻辑设备 *v.s.* 物理设备
- 设备分配灵活性
- 易于I/O重定向



I/O软件层次结构



- 逻辑设备名到物理设备名的映射，确定相应物理设备的驱动程序
- 设备的分配和释放
- 设备保护，禁止用户直接访问设备
- 缓冲管理与差错控制
- 向用户层软件提供统一接口

设备无关软件

设备独立性：应用程序独立于具体使用的物理设备

- 逻辑设备 *v.s.* 物理设备
- 设备分配灵活性
- 易于I/O重定向



逻辑设备到物理设备的映射

逻辑名到物理名映射的实现:

逻辑设备表: LUT (Logical Unit Tables) 用于名称映射

逻辑设备名	物理设备名	驱动程序入口地址
/dev/tty	3	1024
/dev/printer	5	2046
...

逻辑设备名	系统设备表指针
/dev/tty	3
/dev/printer	5
...	...

LUT的设置方式:

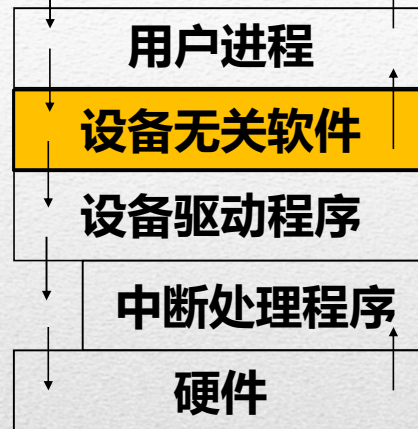
- 整个系统设置一张LUT: 简单, 但多用户系统中, 不允许重名。
- 每个用户一张LUT: 用户登录时, 为该用户创建一个进程, 同时建立一张LUT。



I/O软件层次结构

I/O请求

I/O应答



进行I/O调用；格式化I/O；SPOOLING

映射；保护；阻塞；缓冲；分配

建立设备寄存器；检查状态

当I/O结束时，唤醒驱动程序

执行I/O操作

- 逻辑设备名到物理设备名的映射，确定相应物理设备的驱动程序
- 设备的分配和释放
- 设备保护，禁止用户直接访问设备
- 缓冲管理与差错控制
- 向用户层软件提供统一接口

设备无关软件

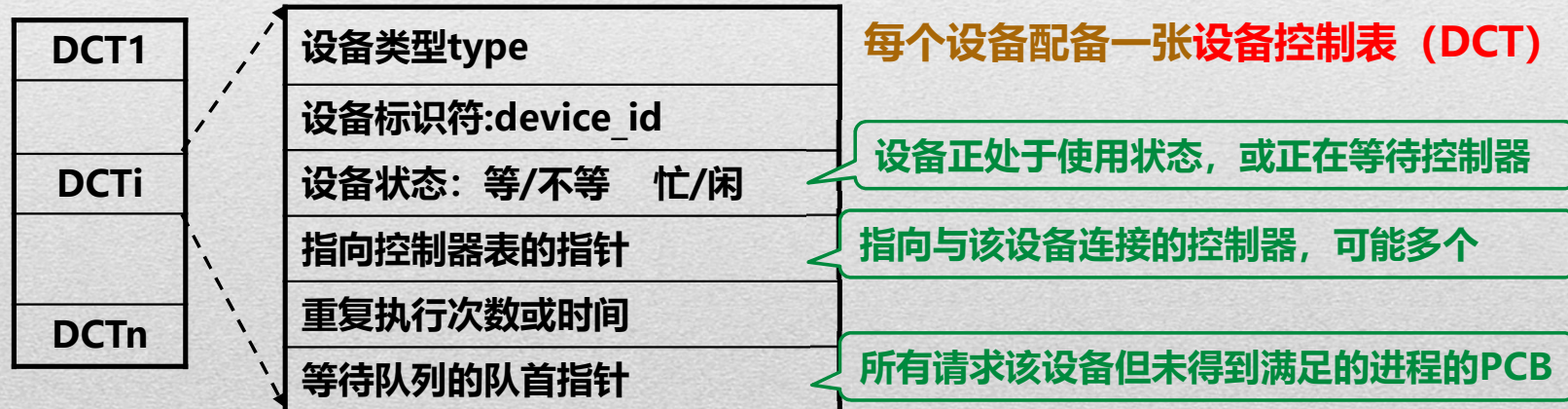
设备独立性：应用程序独立于具体使用的物理设备

- 逻辑设备 v.s. 物理设备
- 设备分配灵活性
- 易于I/O重定向



设备分配

1. 在多道程序环境下，系统中的设备供所有进程共享。为防进程对系统资源无序的竞争，规定系统设备不允许用户自行使用，须由系统统一分配。
2. 进程首先向设备管理程序提出资源申请，然后，由设备分配程序根据设备的固有属性（**独占设备、共享设备、可虚拟设备**），相应的分配算法（**先来先服务、优先级**）和系统安全性考虑（**死锁?**）为进程分配资源（**设备和控制器**），形成一条数据传输通路。如果资源暂时无法获得，进程将被放入相应的资源等待队列。





设备分配

控制器标识符: controller id
控制器状态: 忙/闲
控制器队列的队首指针
控制器队列的队尾指针

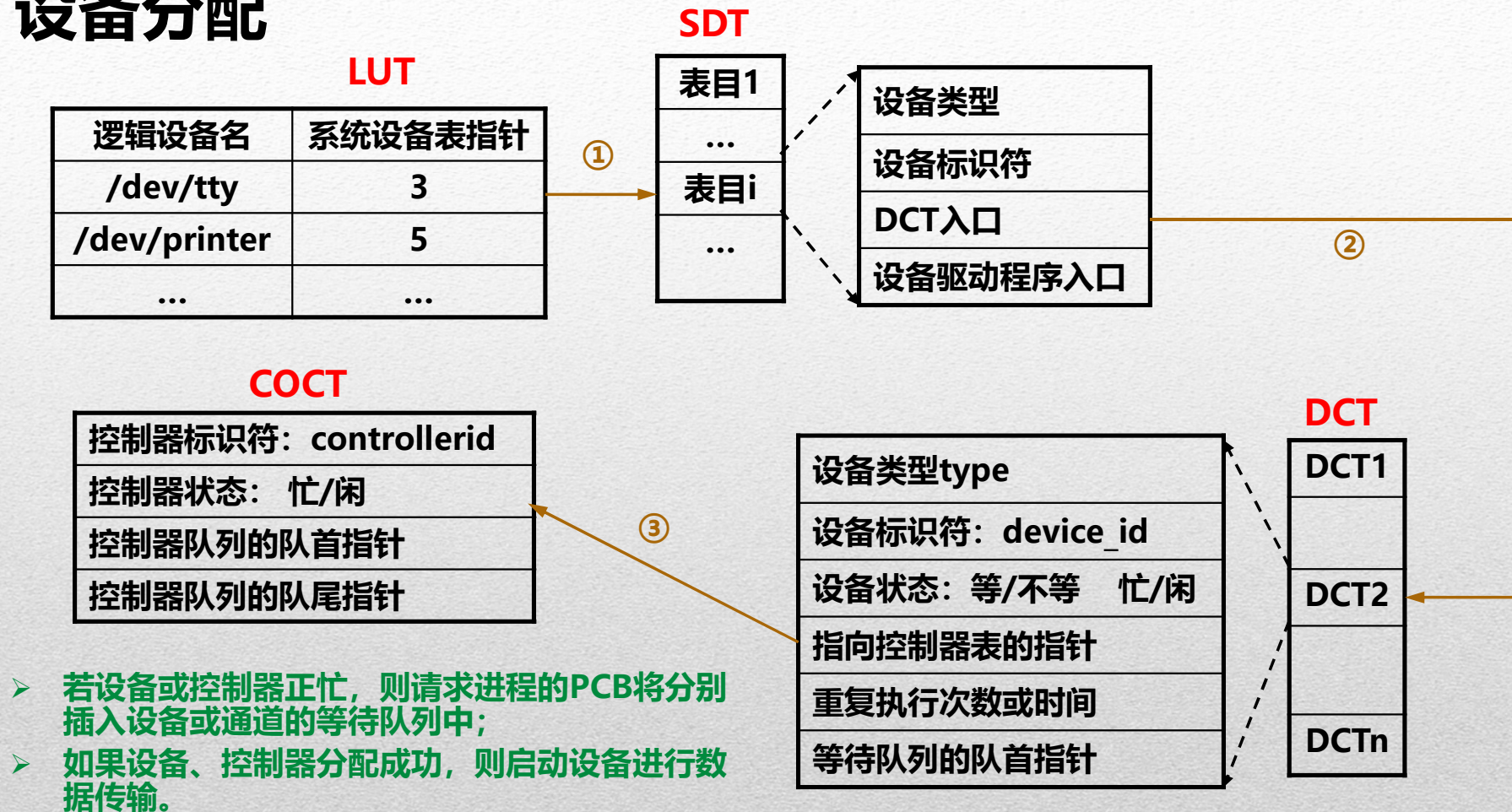
控制器表COCT: 每个控制器一张, 反映I/O控制器使用状态以及和通道的连接情况等。



系统设备表SDT: 整个系统一张表, 记录系统中所有I/O设备的信息, 每个设备占用一个表目。

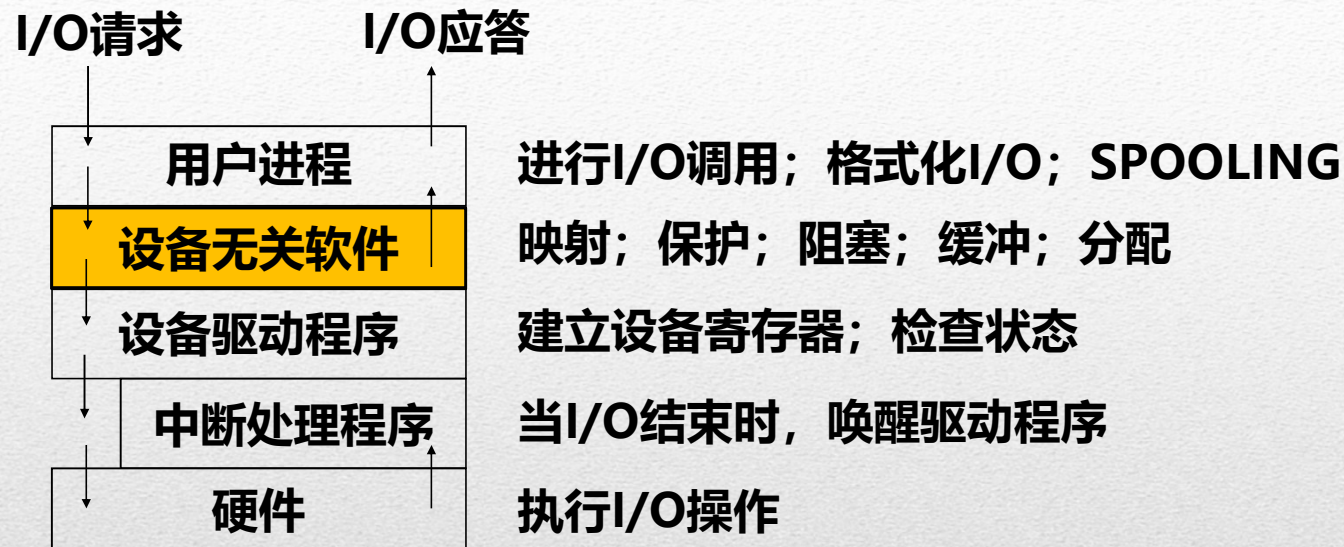


设备分配





I/O软件层次结构



- 设备的分配和释放
- 逻辑设备名到物理设备名的映射，确定相应物理设备的驱动程序
- 设备保护，禁止用户直接访问设备
- **缓冲管理**与差错控制
- 向用户层软件提供统一接口



高速缓存

- (1) 缓和CPU与I/O设备间速度不匹配的矛盾
- (2) 减少对CPU的中断频率
- (3) 提高CPU和I/O设备之间的并行性

```
while ( ... )
```

```
{
```

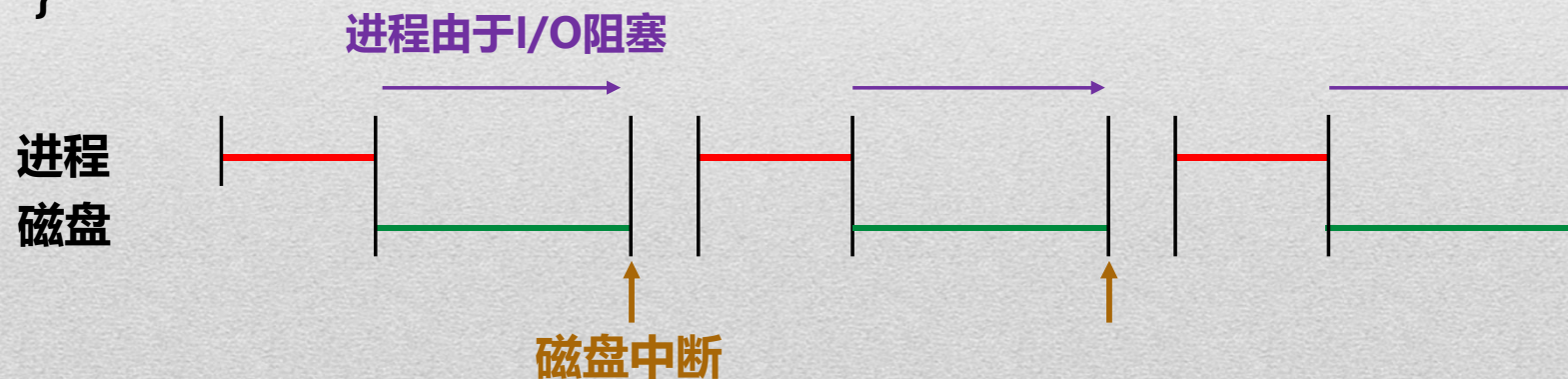
```
    计算 a 数组;
```

```
    保存 a 数组;
```

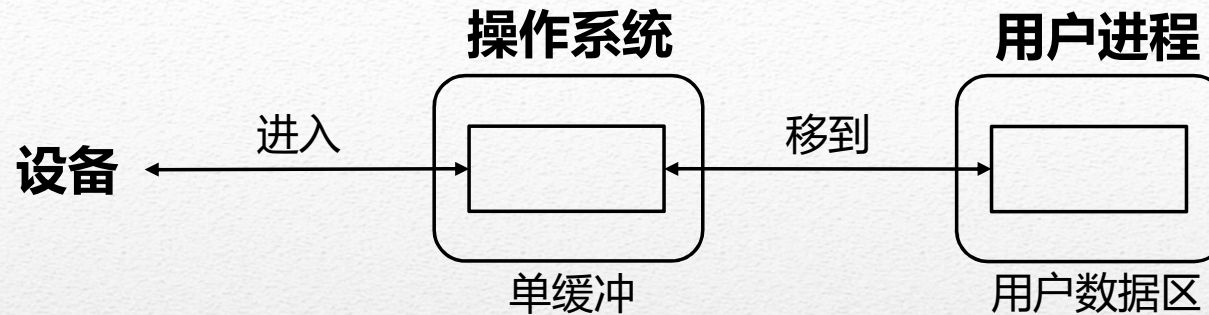
```
}
```

如果不采用缓存，在真正打印完之前不能开始下一轮计算！

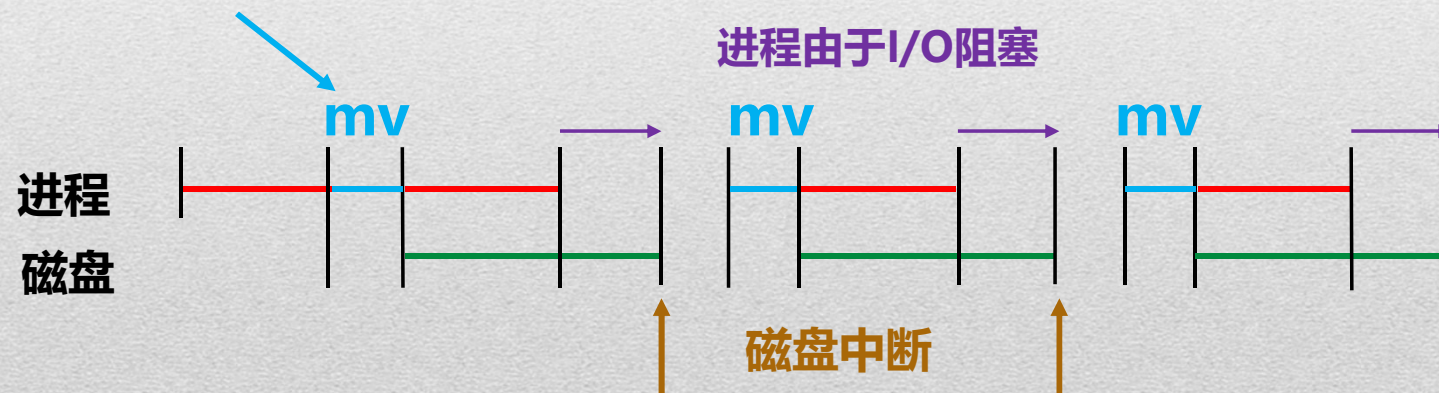
无缓冲



高速缓存

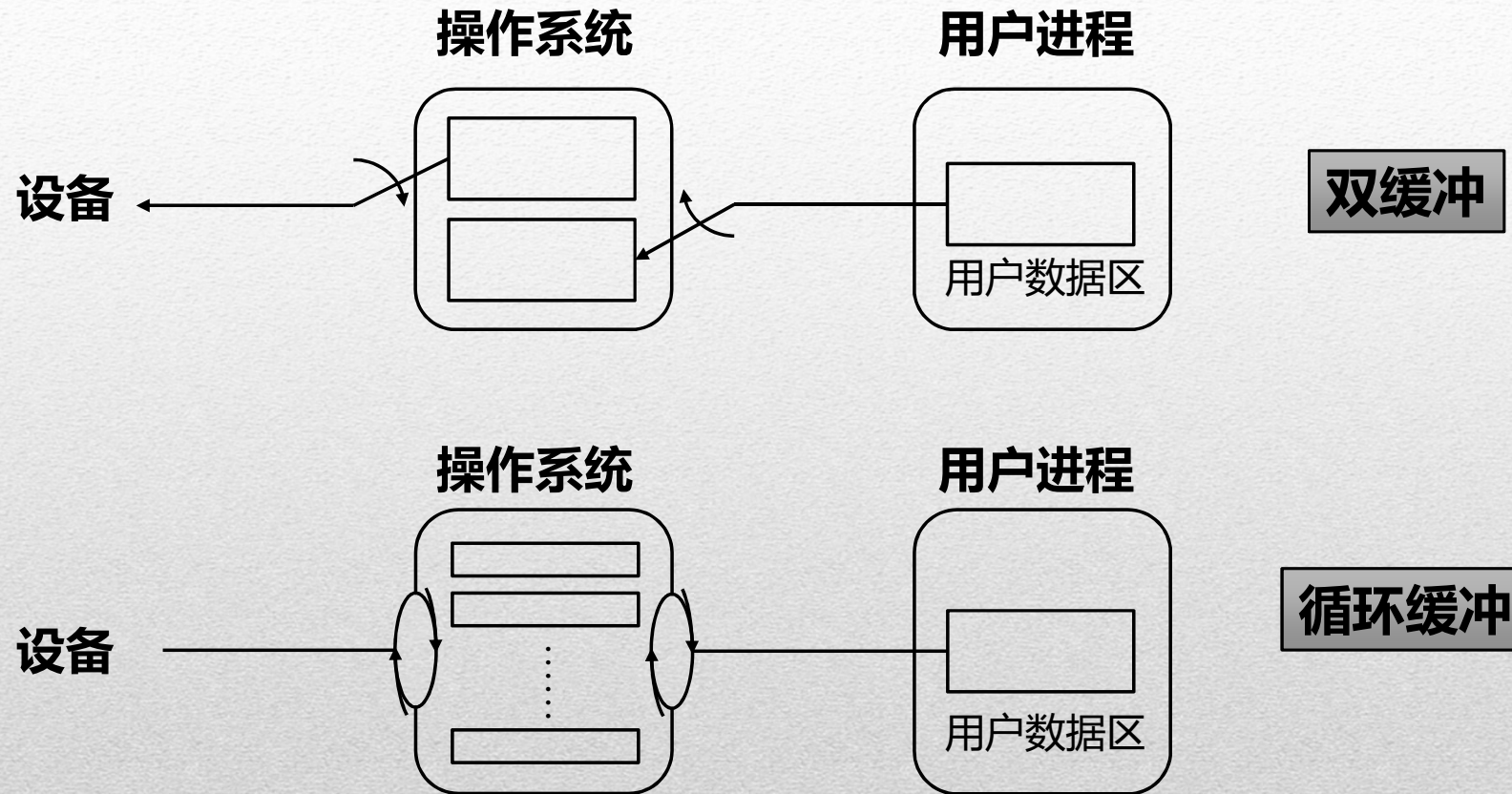


将数据送入缓冲区



单缓冲

高速缓存





高速缓存

系统开辟一个内存空间作为I/O的**共享缓冲区**，不将缓冲区与具体设备固定在一起，而是集中管理。

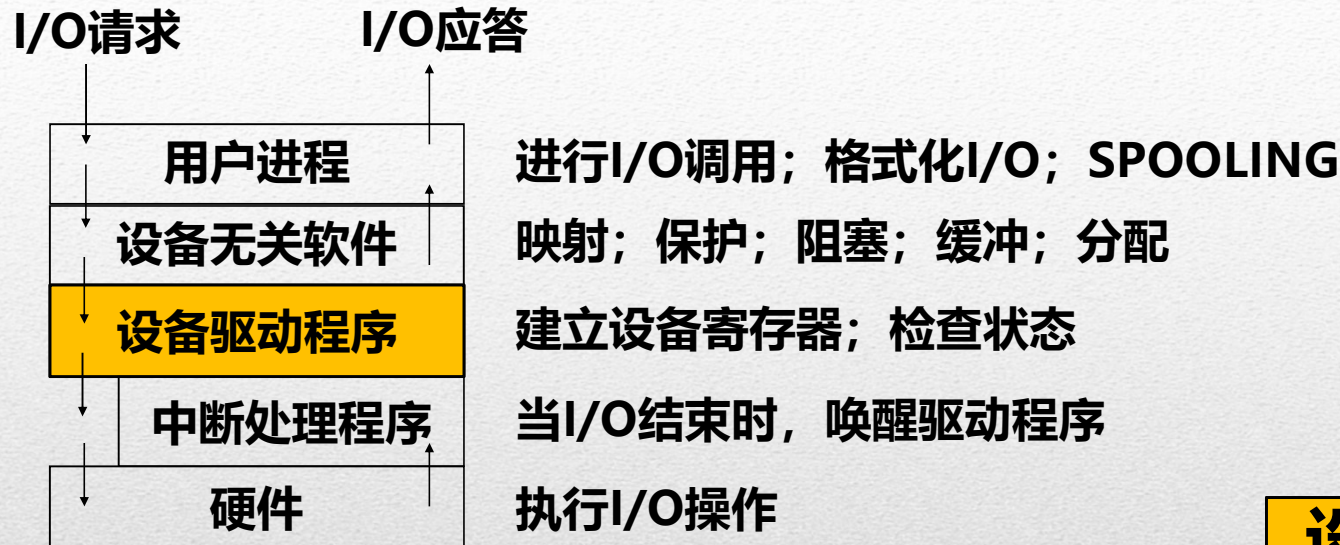
当进程请求I/O时，根据需要从缓冲池里申请缓冲区；I/O处理完毕，系统收回缓冲区归还给缓冲池。

缓冲池

提高了效率，但系统却要为此付出复杂管理的代价



I/O软件层次结构



- 接收上层软件发来的抽象命令和参数, 转换为具体I/O操作要求;
- 检查I/O请求合法性, 了解设备状态, 传递参数, 设置设备工作方式
- 发出I/O命令。(空闲, 启动; 忙碌, 等待)

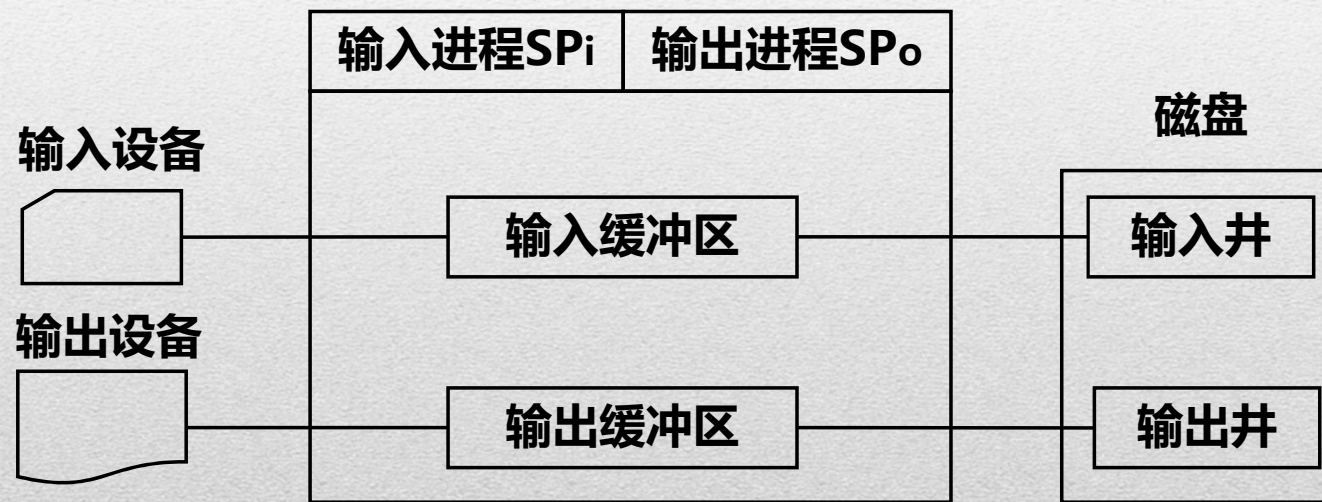
设备驱动程序

- 为每类设备设置一个进程;
- 在整个系统中设置一个I/O进程
- 专用的设备处理程序模块供调用



虚拟设备的分配: SPOOLING技术

(Simultaneous Peripheral Operating On Line, 假脱机操作) 将一台I/O设备虚拟成多台逻辑I/O设备





主要内容

5.1 I/O系统

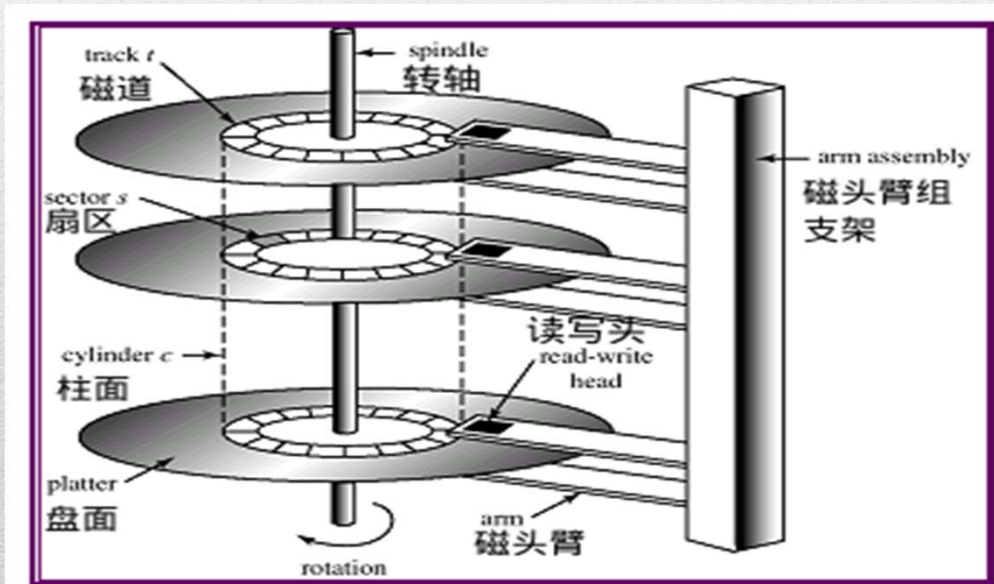
5.2 磁盘存储器管理

5.3 UNIX字符块设备管理

磁盘结构

磁盘存储器是由磁盘、驱动器机构和控制器三部分构成

- **磁盘**：在铝合金盘或塑料盘上的一层磁性材料作为信息存储媒体
- **驱动器机构**：读写磁头、可移动磁头臂以及驱动磁盘转动的机构。
- **控制器**：包括控制读写电路、驱动磁头臂移动和磁盘转动的控制电路等。



在大容量磁盘中，一个驱动机构上安装由若干个盘片组成的盘组。

数据是按柱面来存放的，同一柱面上各磁道放满后，再存到下一柱面

磁盘的编址方式与地址转换

盘地址空间是三维地址:

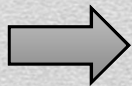
< 磁道号*i*, 盘面号 *j*, 扇区号 *k* >

文件系统使用一维线性地址:

< 盘块号 >

由盘设备驱动程序
完成地址转换

磁盘驱动器工作时, 以恒定的速度进行旋转。只有当磁头位于指定的磁道和该磁道中指定的扇区开始处时, 才能够进行读或写操作。



磁盘的访问时间分为三个部分:

$T_a = T_s$ (寻道时间) + T_r (旋转延迟时间) + T_t (传送时间)



提高磁盘I/O速度的方法

1. 磁盘调度算法 (减少寻道时间)



磁盘调度算法

当前磁头所在磁道号=100

先来先服务 FCFS

磁道号	移动距离 (磁道数)
19	81
376	357
205	171
134	71
18	116
56	38
192	136
396	204
29	367
3	26
19	16
40	21
磁头移动总距离=1604磁道	

简单，但平均寻道距离较大

最短查找时间优先调度 SSTF

磁道号	移动距离 (磁道数)
134	34
192	58
205	13
56	149
40	16
29	11
19	10
19	0
18	1
3	15
376	373
396	20
磁头移动总距离=700磁道	

可能导致进程饿死



磁盘调度算法

当前磁头所在磁道号=100

扫描SCAN (电梯算法)

磁道号	移动距离 (磁道数)
134	34
192	58
205	13
376	171
396	20
56	340
40	16
29	11
19	10
19	0
18	1
3	15
磁头移动总距离=689磁道	

磁头向磁道数增加方向运动

循环扫描 CSCAN (单向电梯)

磁道号	移动距离 (磁道数)
134	34
192	58
205	13
376	171
396	20
3	393
18	15
19	1
19	0
29	10
40	11
56	16
磁头移动总距离=742磁道	

避免进程饿死



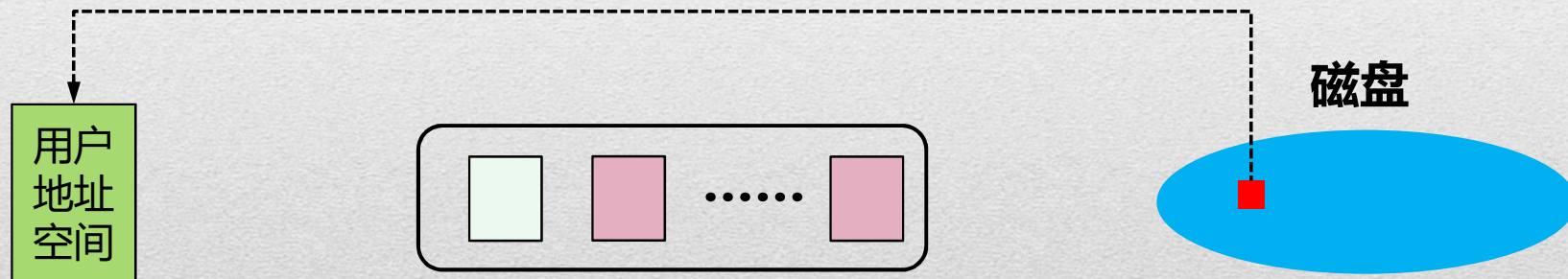
提高磁盘I/O速度的方法

1. 磁盘调度算法 (减少寻道时间)
2. 磁盘高速缓存

提高磁盘I/O速度的方法

1. 磁盘调度算法 (减少寻道时间)
2. 磁盘高速缓存

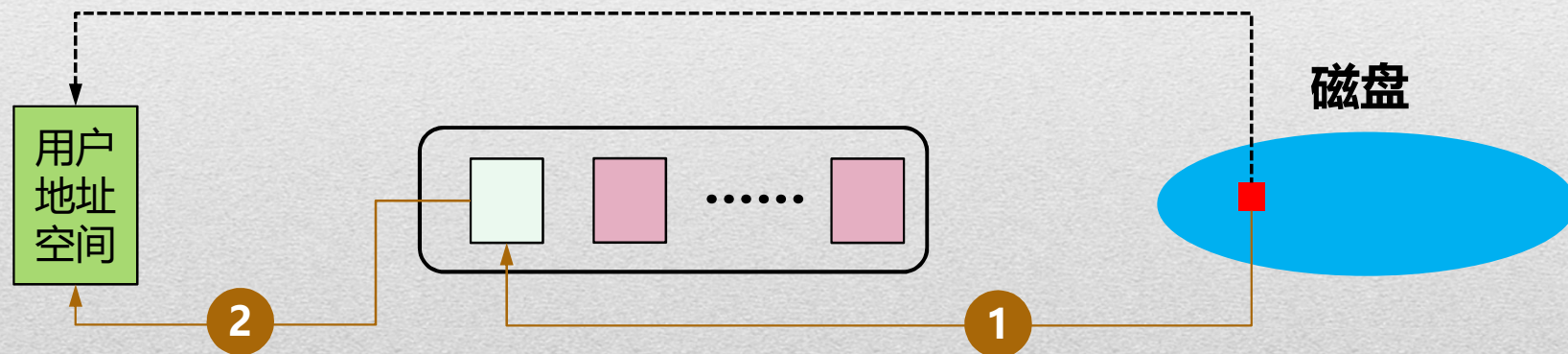
读操作



提高磁盘I/O速度的方法

1. 磁盘调度算法 (减少寻道时间)
2. 磁盘高速缓存

读操作

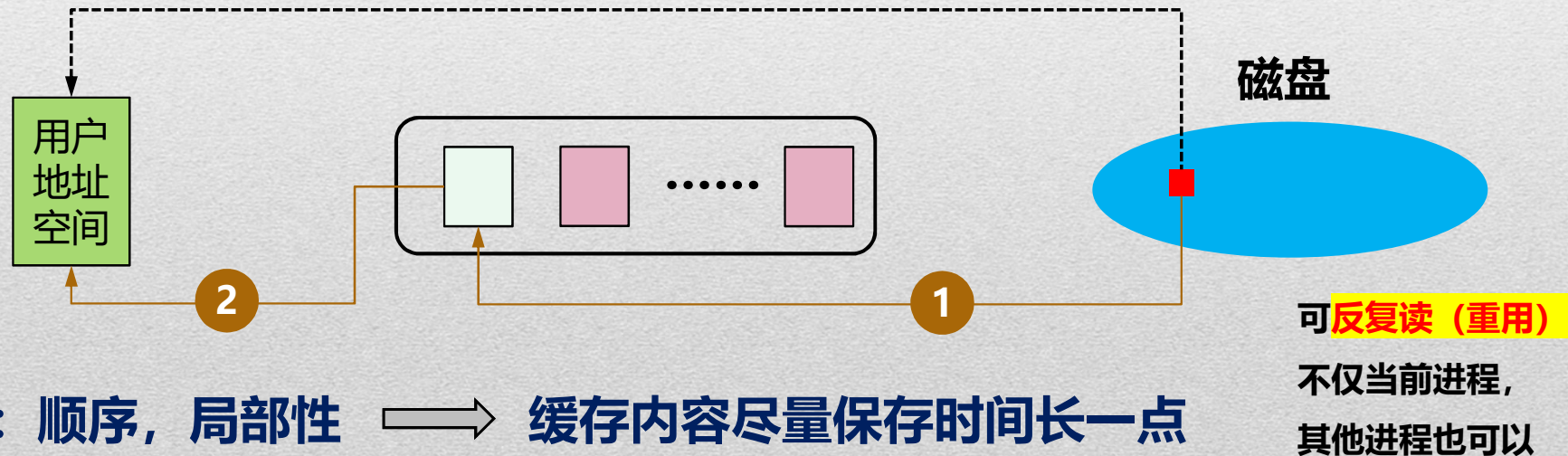


特征：顺序，局部性 \Rightarrow 缓存内容尽量保存时间长一点

提高磁盘I/O速度的方法

1. 磁盘调度算法 (减少寻道时间)
2. 磁盘高速缓存

读操作

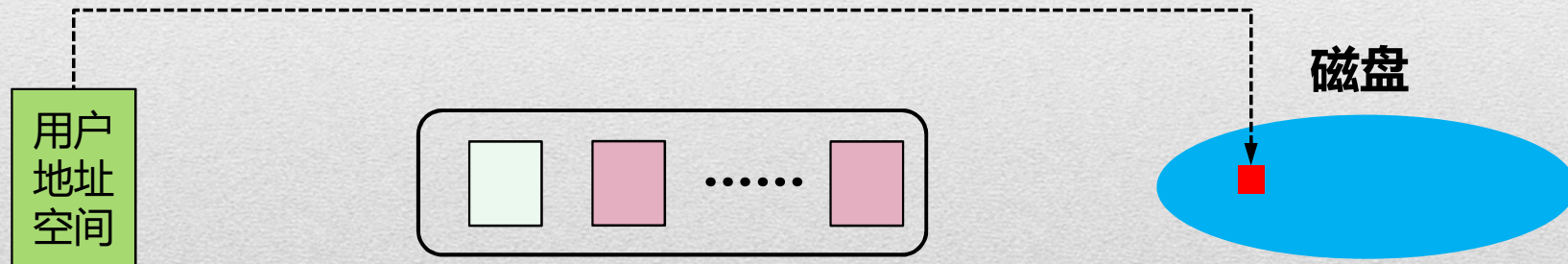


特征：顺序，局部性 \Rightarrow 缓存内容尽量保存时间长一点

提高磁盘I/O速度的方法

1. 磁盘调度算法 (减少寻道时间)
2. 磁盘高速缓存

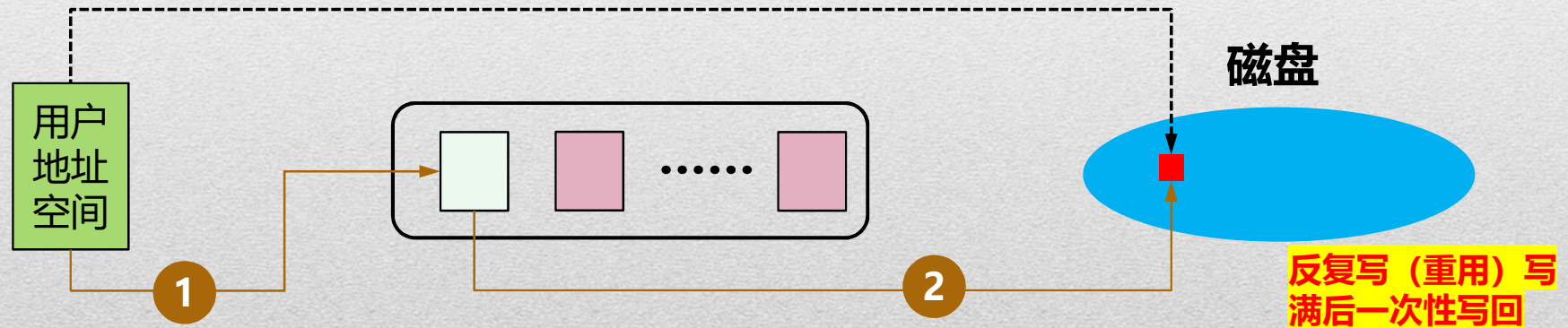
写操作



提高磁盘I/O速度的方法

1. 磁盘调度算法 (减少寻道时间)
2. 磁盘高速缓存

写操作



特征：顺序，局部性 \Rightarrow 缓存内容尽量保存时间长一点

不仅当前进程，
其他进程也可以

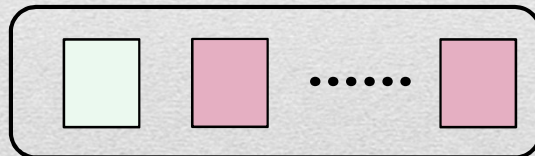


提高磁盘I/O速度的方法

1. 磁盘调度算法 (减少寻道时间)

2. 磁盘高速缓存

所以，对每一块缓存，需记录：



关联的盘块： 盘块地址

当前的状态： 忙？ 闲？
闲了多久？

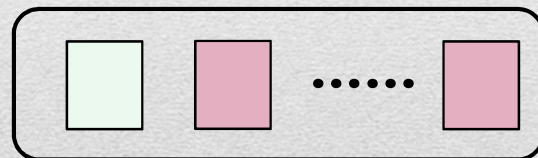
操作的类型： 读？ 写？

提高磁盘I/O速度的方法

1. 磁盘调度算法 (减少寻道时间)

2. 磁盘高速缓存

所以，对每一块缓存，需记录：



关联的盘块：

盘块地址

当前的状态：

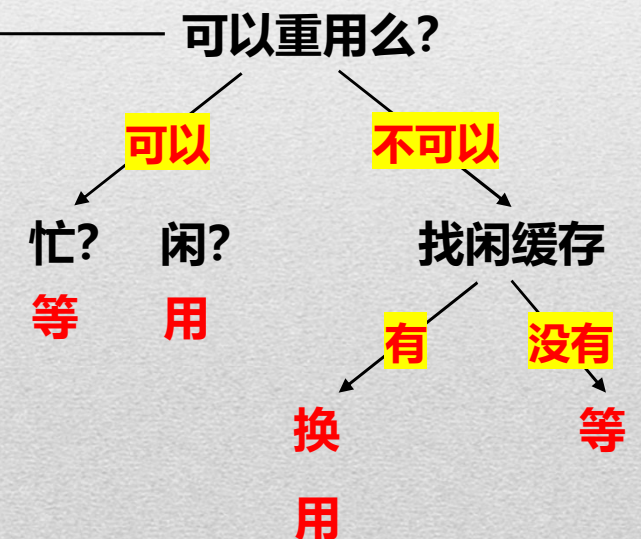
忙？ 闲？

闲了多久？

操作的类型：

读？ 写？

申请缓存时：





本节小结:

- 1 了解I/O软件系统和硬件系统
 - 2 掌握常用的磁盘调度算法
-