

计算机组成原理

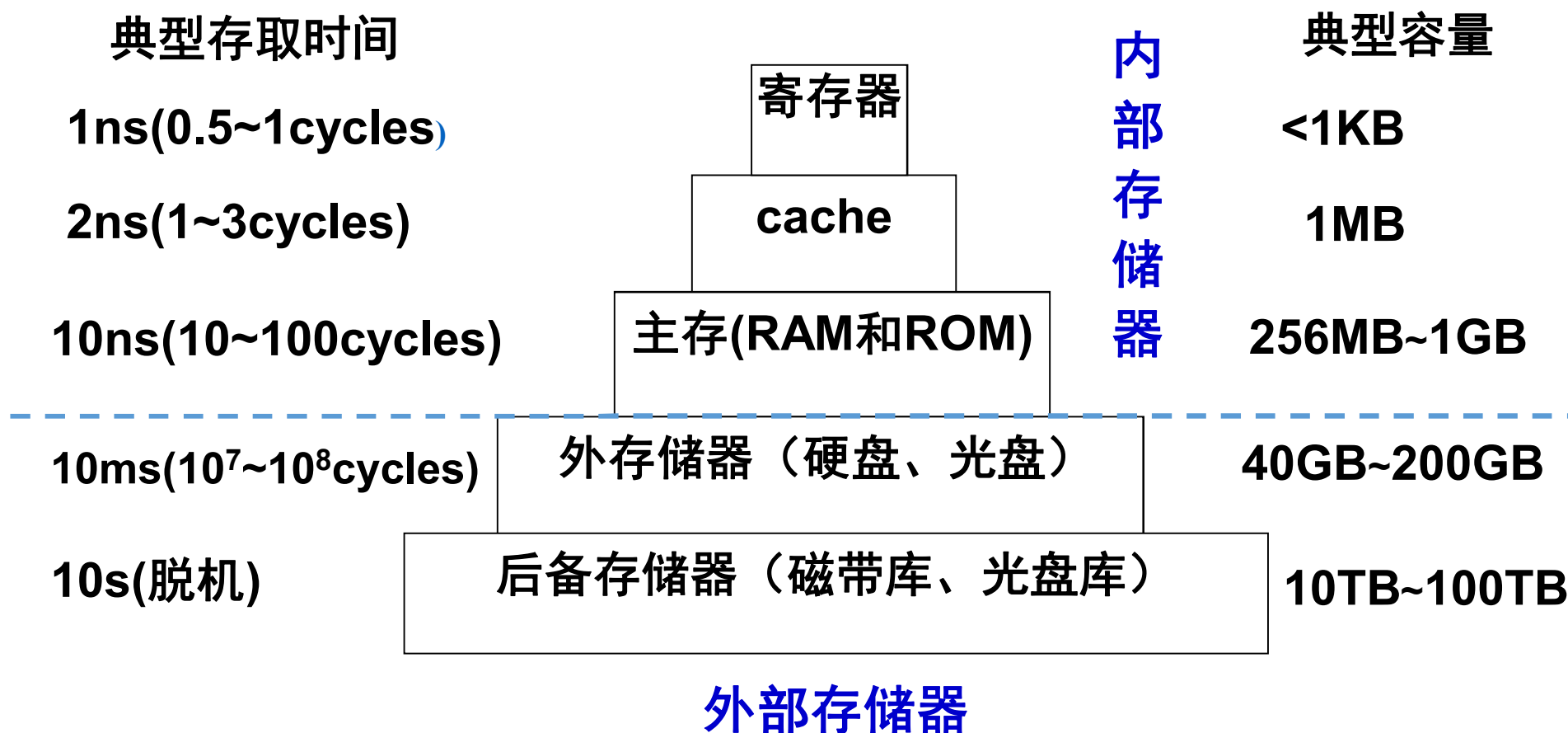
翁睿

哈尔滨工业大学

4.4 虚拟存储器

- 虚拟存储器概述
- 页式虚拟存储器及页表管理
- 虚拟存储器地址映射与变换
- 虚拟存储器举例

存储器的层次结构



列出的时间和容量会随时代变化，但数量级相对关系不变。

虚拟存储系统的基本概念

- 虚拟存储技术的引入用来解决一对矛盾：

有限的主存空间 ⇔ 运行较多的较大的用户程序

- 虚拟存储技术的实质

- 程序员在比实际主存空间大得多的逻辑地址空间中编写程序
- 程序执行时，把当前需要的程序段和相应的数据块调入主存，其他暂不用的部分存放在磁盘上
- 指令执行时，通过硬件将逻辑地址（也称虚拟地址或虚地址）转化为物理地址（也称主存地址或实地址）
- 在发生程序或数据访问失效(缺页)时，**由操作系统**进行主存和磁盘之间的信息交换

虚拟存储器机制由**硬件**和**操作系统**共同协作实现

虚拟存储器基本原理

4.4

虚存管理方式--页式管理

- 页式虚存把主存与外存均划分成等长的页面。常用页大小为**4KB~64KB**。
- 便于维护，便于生成页表，类似于**cache**的块表
- 不容易产生碎块，存储空间浪费小
- 页不是独立的实体，处理、保护和共享不方便

虚拟存储器基本原理

4.4

虚存管理方式--段式管理

- 段式虚存把空间划分为可变长的块，称为段，段的分界与程序的自然分界相对应，段最小长度为1个字节，最大因机器而异，常为 $2^{16}\text{B} \sim 2^{32}\text{B}$ 。
- 段的独立性—易于编译、管理、修改和维护，也便于多道程序共享
- 各段的长度不同，给主存的分配带来麻烦
- 段式管理容易产生碎块，浪费主存空间

虚拟存储器基本原理

4.4

虚存管理方式--段页式管理

- 程序按模块分段
- 段再分成长度固定的页。
- 程序调入调出按页面来进行
- 程序共享保护按段进行
- 兼备段式，页式管理的优点
- 在地址映像中需要多次查表

“Cache—主存”与“主存—辅存”层次的区别 4.4

存储层次 比较项目	“Cache —主存”层次	“主存—辅存”层次
目的	帮助主存提升速度	帮助主存扩展容量
存储管理实现	主要由专用硬件实现	主要由软件实现
访问速度的比值 (第一级和第二级)	几比一	几百比一 必须让失效率降到极低
典型的块(页)大小	几十个字节	几百到几千个字节
CPU对第二级的 访问方式	可直接访问	均通过第一级
失效时CPU是否切换	不切换	切换到其他进程

4.4 虚拟存储器

- 虚拟存储器概述
- 页式虚拟存储器及页表管理
- 虚拟存储器地址映射与变换

分页（Paging）

4.4

- 基本思想：
 - 内存被分成固定长且比较小的存储块（页框、实页、物理页）
 - 每个进程也被划分成固定长的程序块（页、虚页、逻辑页）
 - 程序块(页)可装到存储器中可用的存储块(页框)中
 - 无需用连续页框来存放一个进程
 - 操作系统为每个进程生成一个页表
 - 通过页表(page table)实现逻辑地址向物理地址转换（Address Mapping， $VA \Leftrightarrow PA$ ）

分页（Paging）

4.4

问题：是否需要将一个进程的全部页面都装入内存？

答：不需要！

根据程序访问局部性可知：可把当前活跃的页面调入主存，其余留在磁盘上！

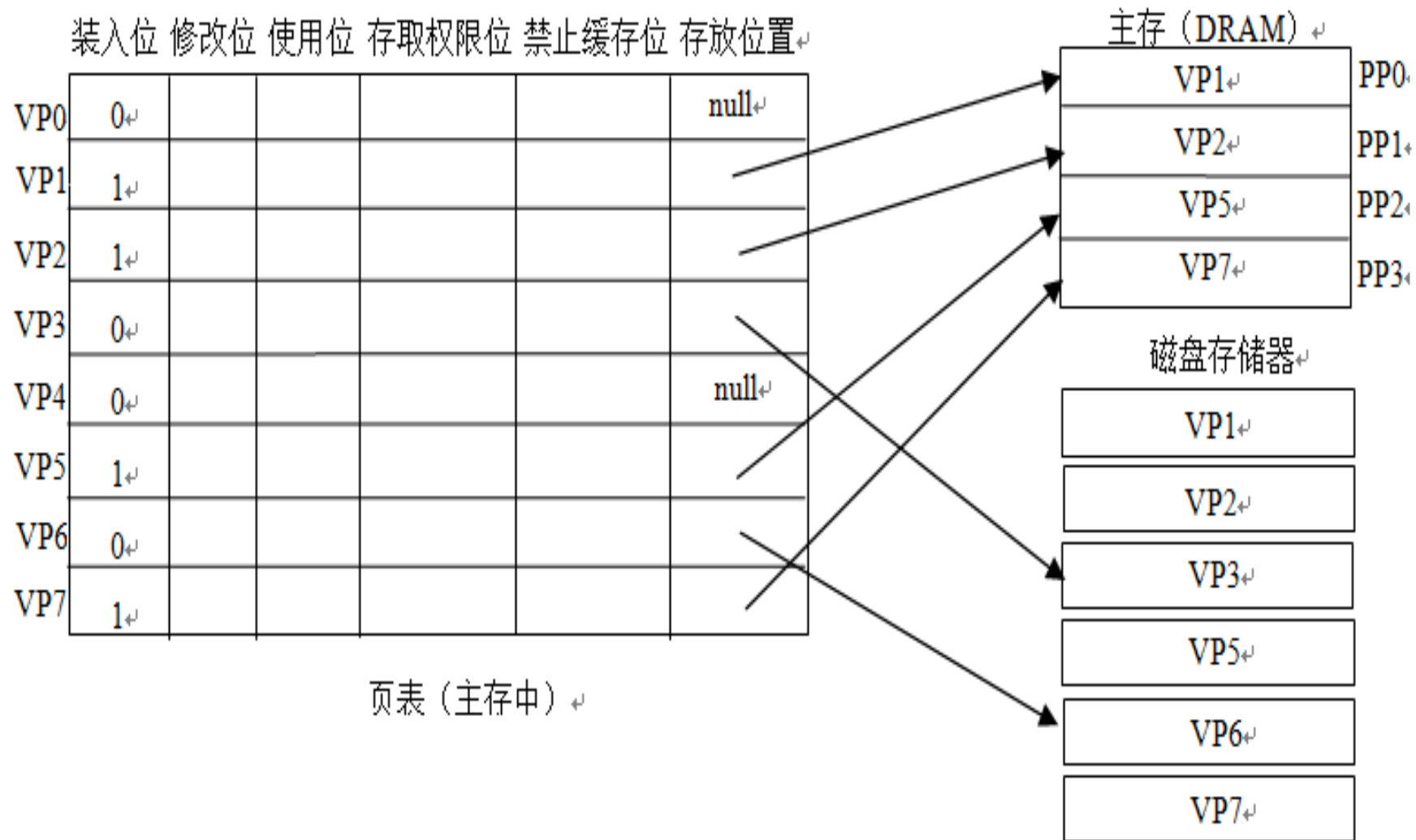
采用“按需调页 Demand Paging”方式分配主存！这就是虚拟存储管理概念

问题：每条指令及其操作数的虚拟地址何时确定？

答：链接时确定虚拟地址；

装入时生成页表，以建立虚拟地址与物理地址之间的映射！

主存中的页表示例



- ◆ **未分配页**: 进程的虚拟地址空间中“空洞”对应的页 (如VP0、VP4)
- ◆ **已分配的缓存页**: 有内容对应的已装入主存的页 (如VP1、VP2、VP5等)
- ◆ **已分配的未缓存页**: 有内容对应但未装入主存的页 (如VP3、VP6)

信息访问中可能出现的异常情况

4.4

可能有两种异常情况：

1) 缺页 (page fault)

产生条件：当Valid (有效位 / 装入位) 为 0 时

相应处理：从磁盘读到内存，若内存没有空间，则还要从内存选择一页替换到磁盘上，替换算法类似于Cache，采用回写法，淘汰时，根据“dirty”位确定是否要写磁盘

当前指令执行被阻塞，当前进程被挂起，处理结束后回到原指令继续执行

2) 保护违例 (protection_violation_fault) 或访问违例

产生条件：当Access Rights (存取权限)与所指定的具体操作不相符时

相应处理：在屏幕上显示“内存保护错”或“访问违例”信息

当前指令的执行被阻塞，当前进程被终止



0x00007FF7E25836C9 指令引用了 0x0000000000000000 内存。该内存不能为 read。

Access Rights (存取权限)可能的取值有哪些？

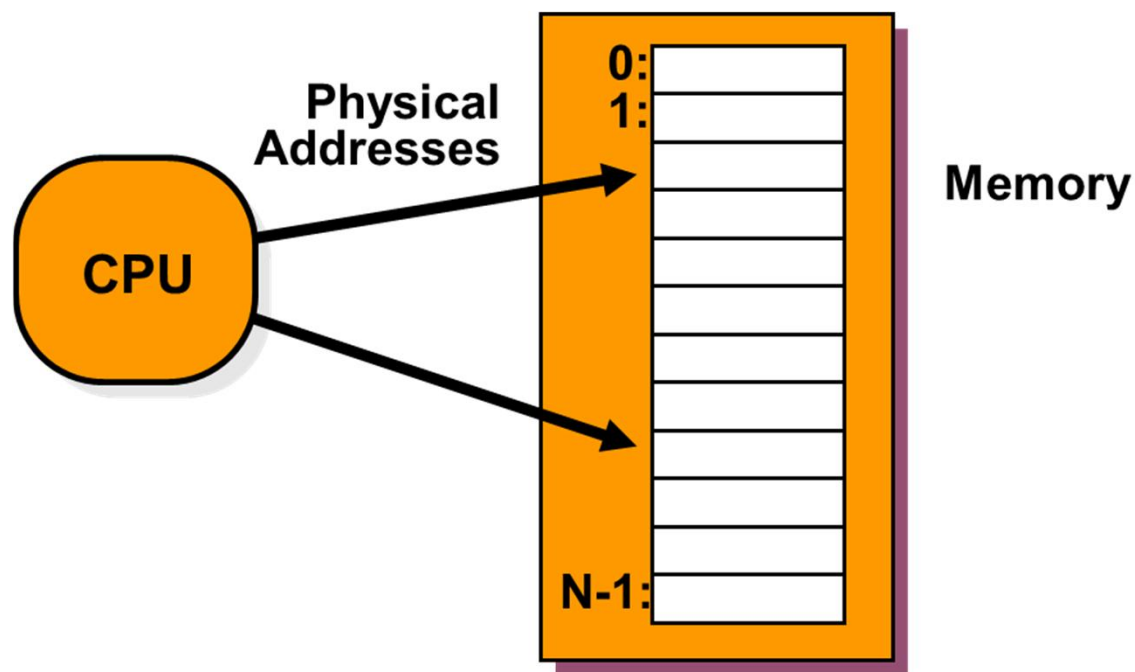
R = Read only, R/W = read/write, X = execute only

4.4 虚拟存储器

- 虚拟存储器概述
- 页式虚拟存储器及页表管理
- 虚拟存储器地址映射与变换

实地址计算机系统

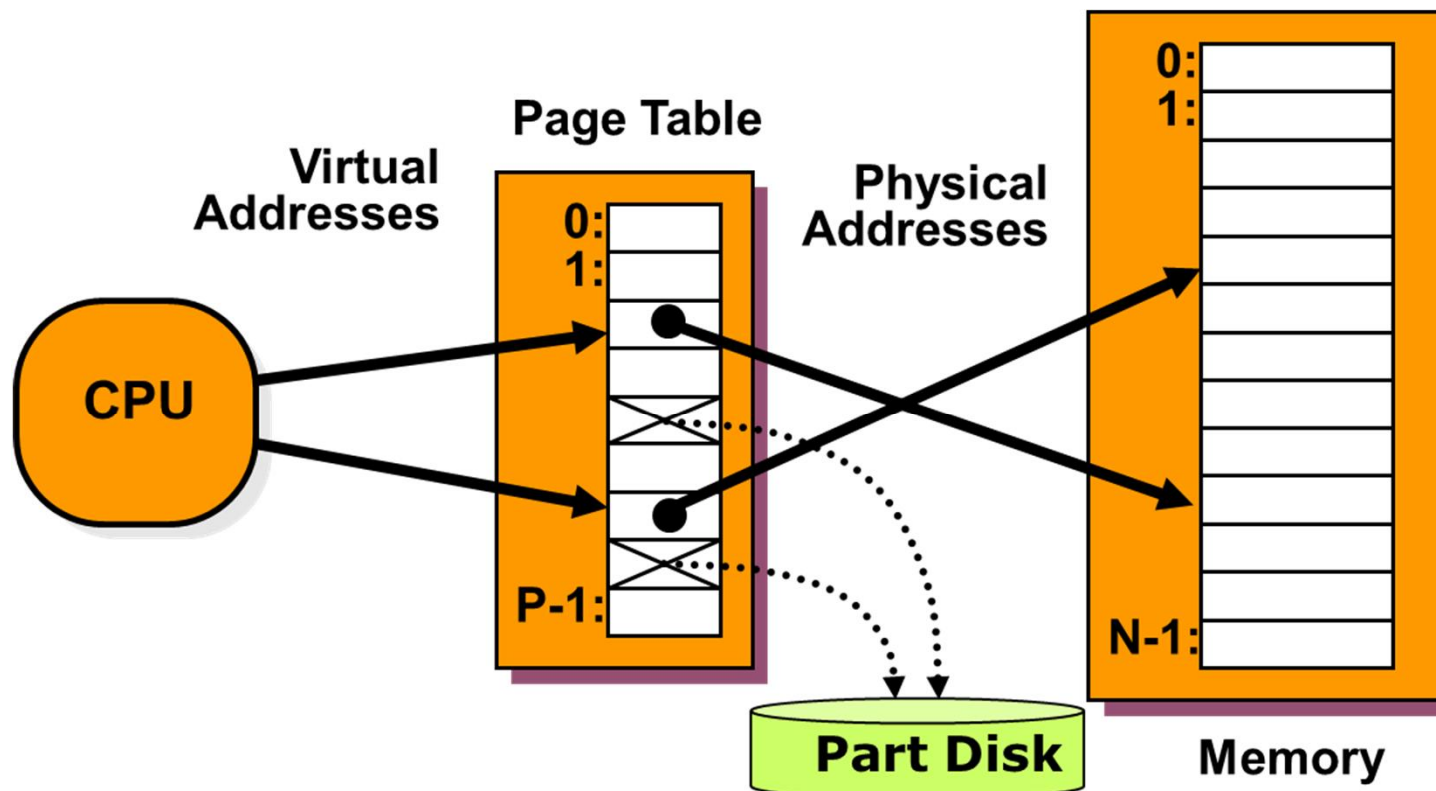
- **CPU地址：** 物理内存地址
 - 大多数Cray计算机，早期PC，大多数嵌入式系统



虚地址计算机系统

4.4

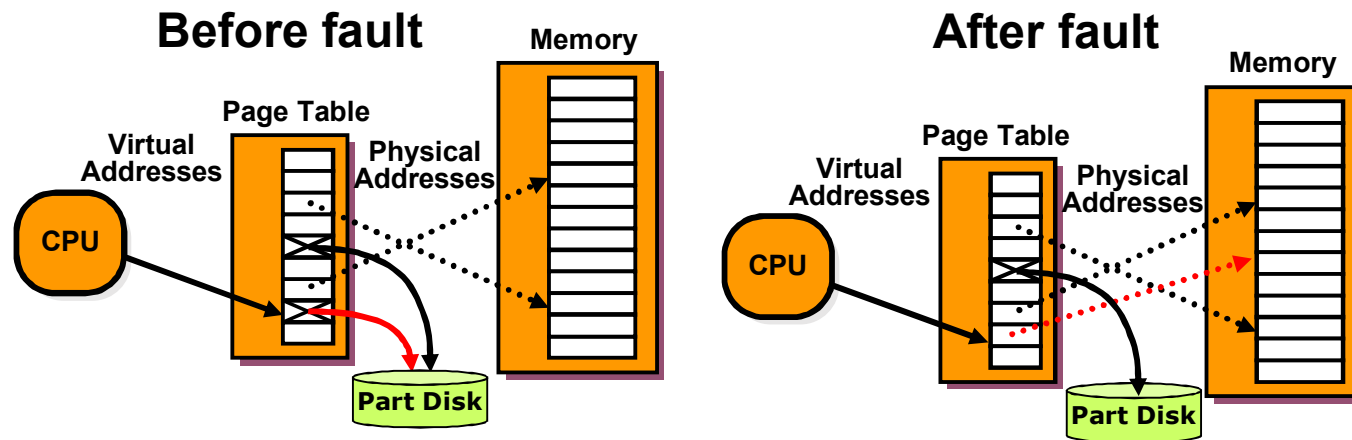
- **CPU地址：虚拟地址**
 - 需要虚实变换，硬件通过OS维护的页表将虚拟地址转换为物理地址
 - 工作站，服务器，现代PC



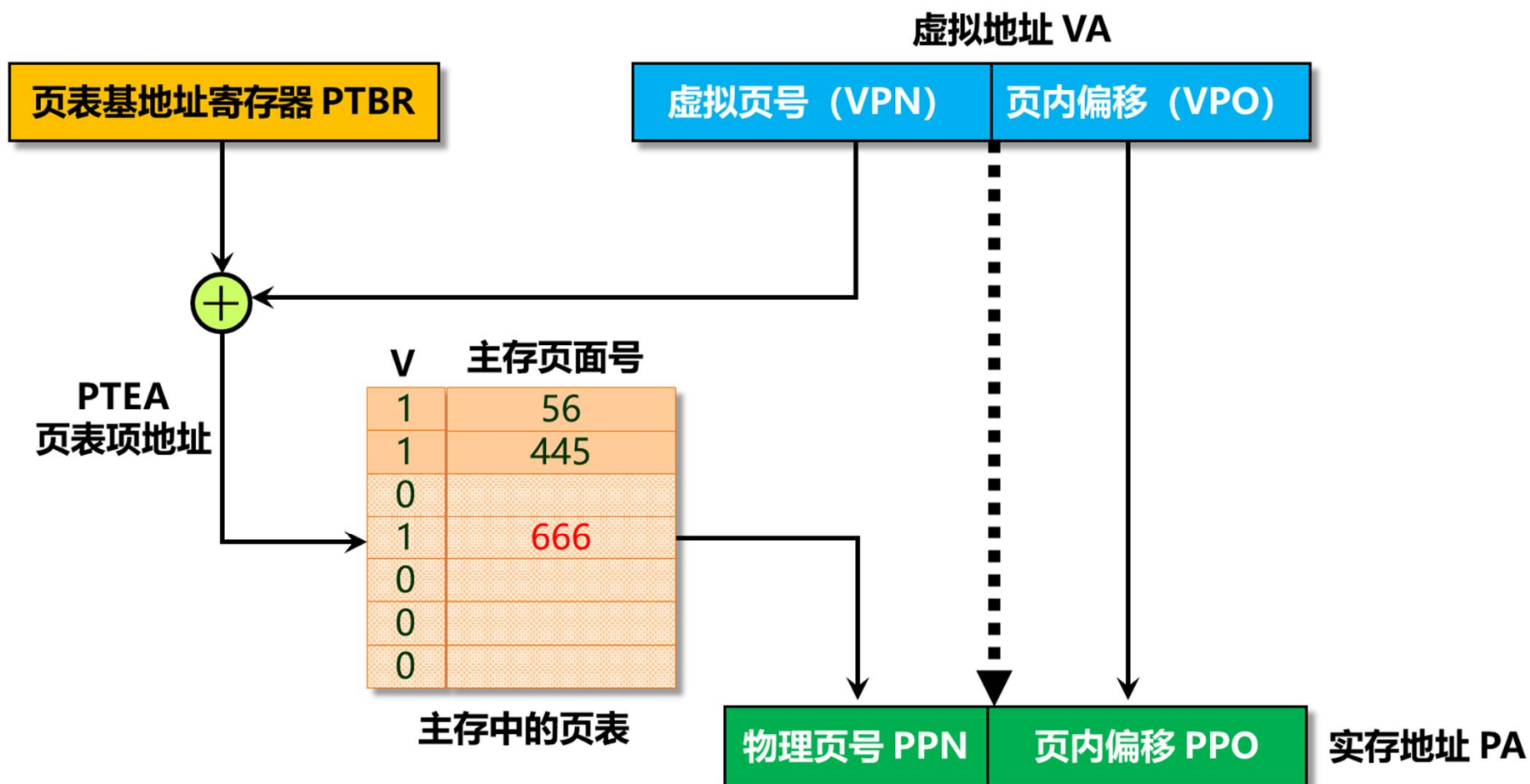
缺页 Page Faults

4.4

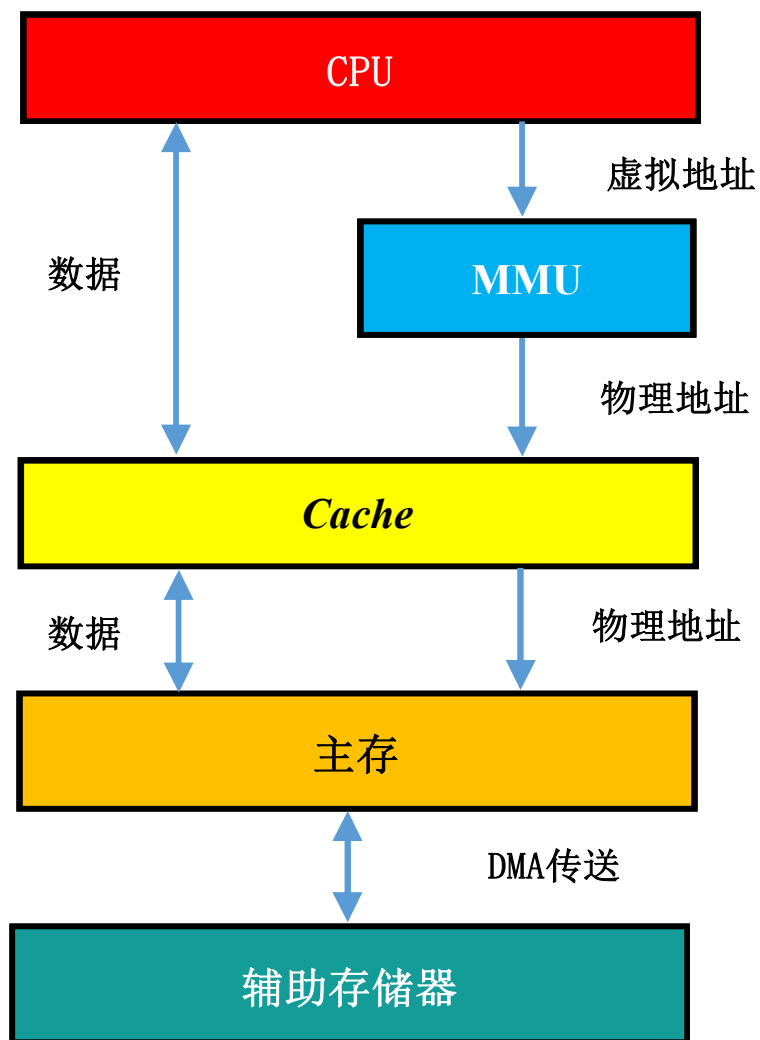
- 查询页表发现：虚拟地址不在内存中
 - 操作系统负责将数据从磁盘迁移到内存中
 - 当前进程挂起
 - 操作系统负责所有的替换策略
 - 唤醒挂起进程



页式虚拟存储器结构

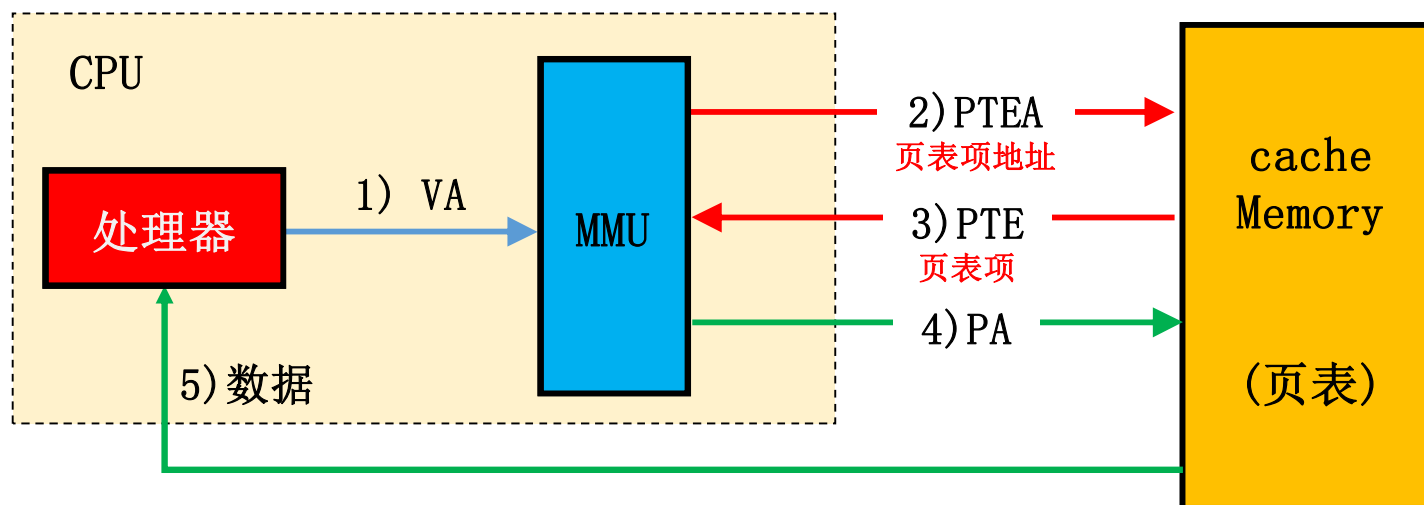


层次性结构

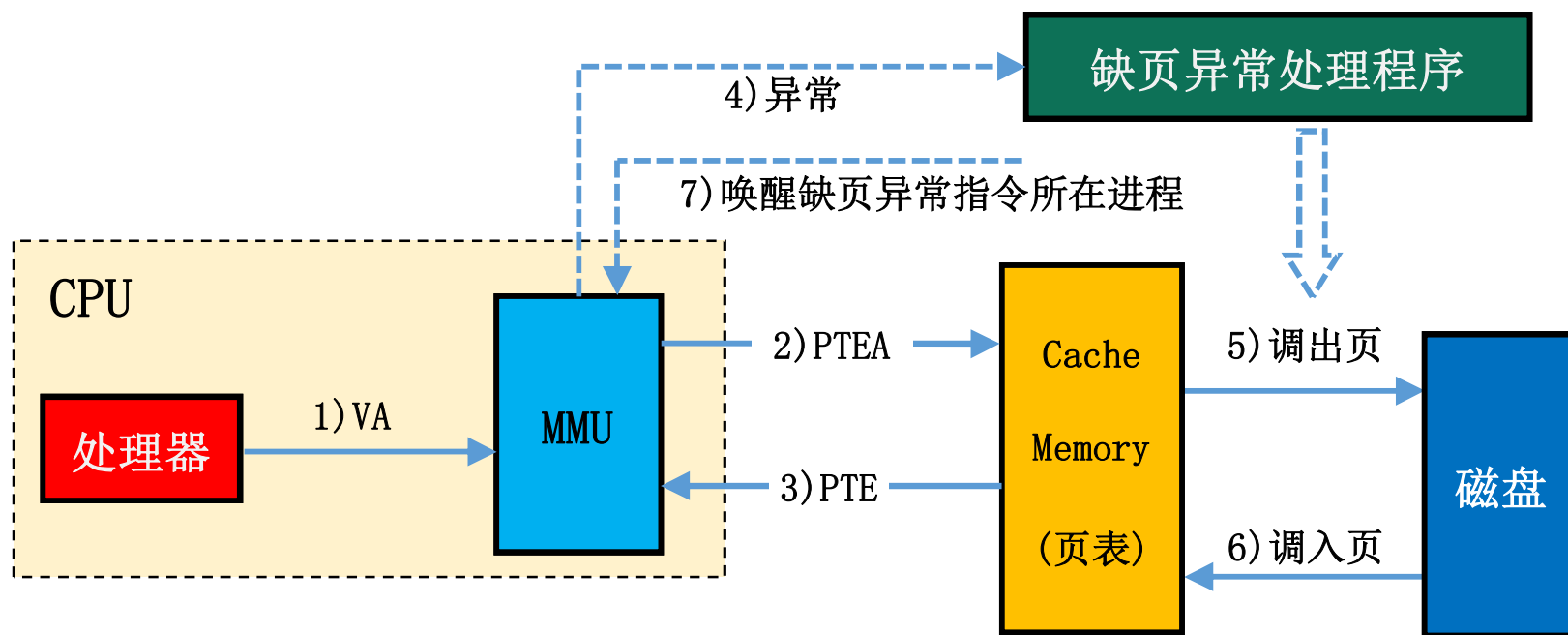


为什么Cache用物理地址访问？

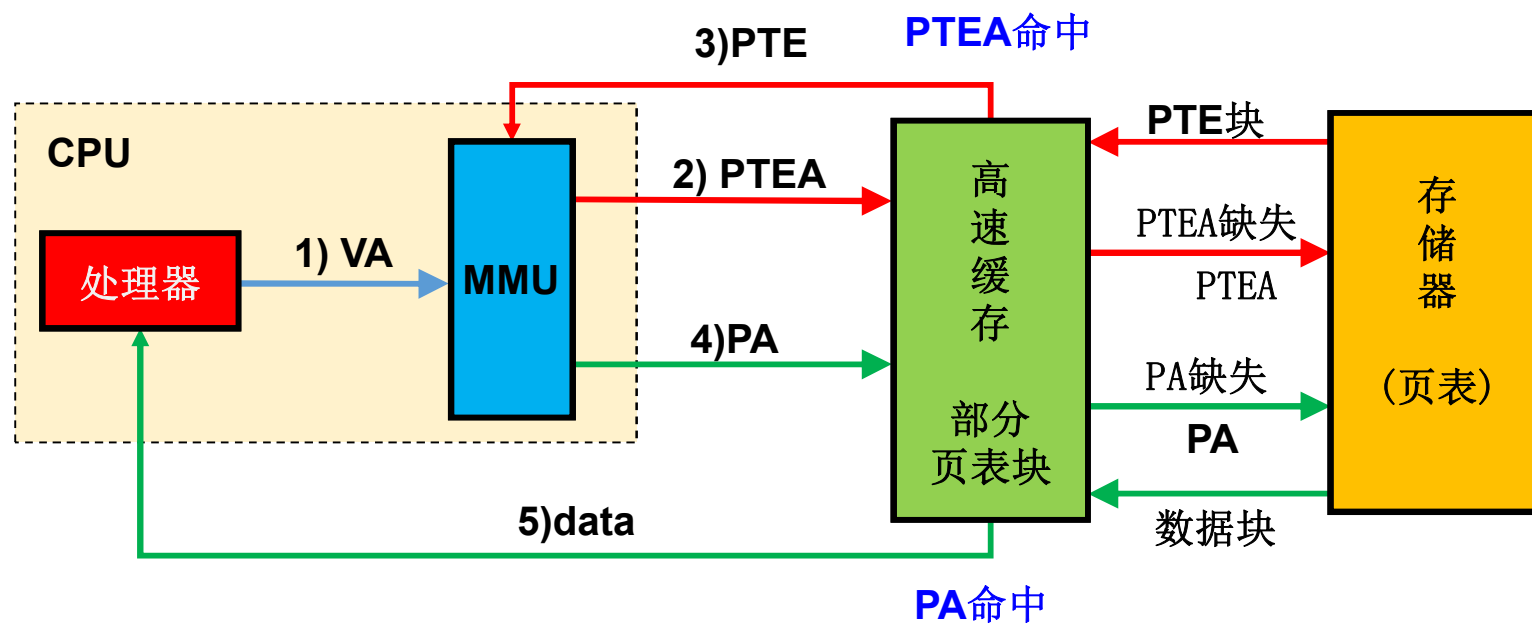
虚拟地址→物理地址（页命中）



虚拟地址→物理地址（缺页）



带Cache的虚存-主存层次



快表提高地址转换速度

- 地址转换速度慢

- 访问页表，访问数据，需2次访存，速度慢
- 为缩小页表大小，OS普遍采用多级页表结构，速度更慢

- 加速方法：引入一个体积小的快表TLB

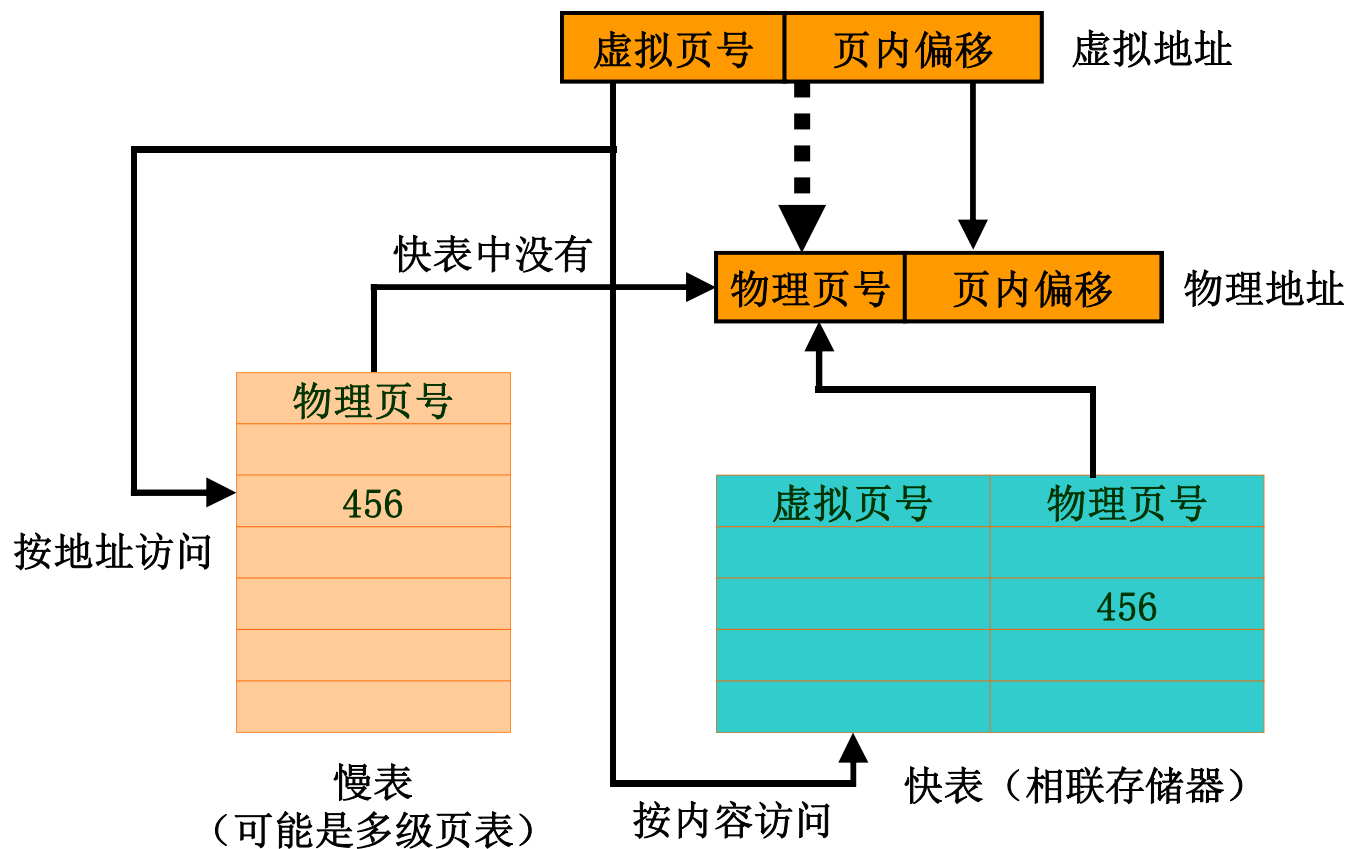
- 缓存页表中经常被访问的表项
- (Valid, VPN, PPN)

多路并发比较机制

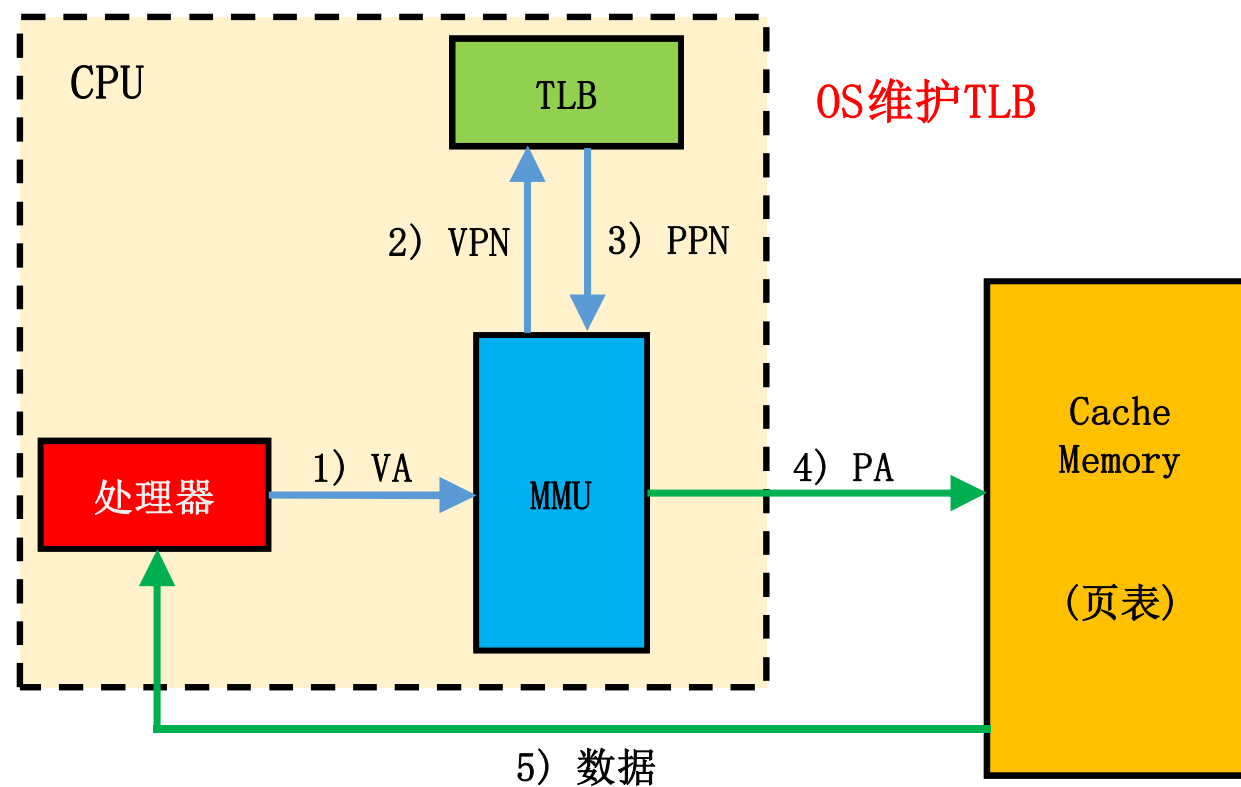
- 快表引入相联存储器机制，提高查找速度

- 采用随机替换算法

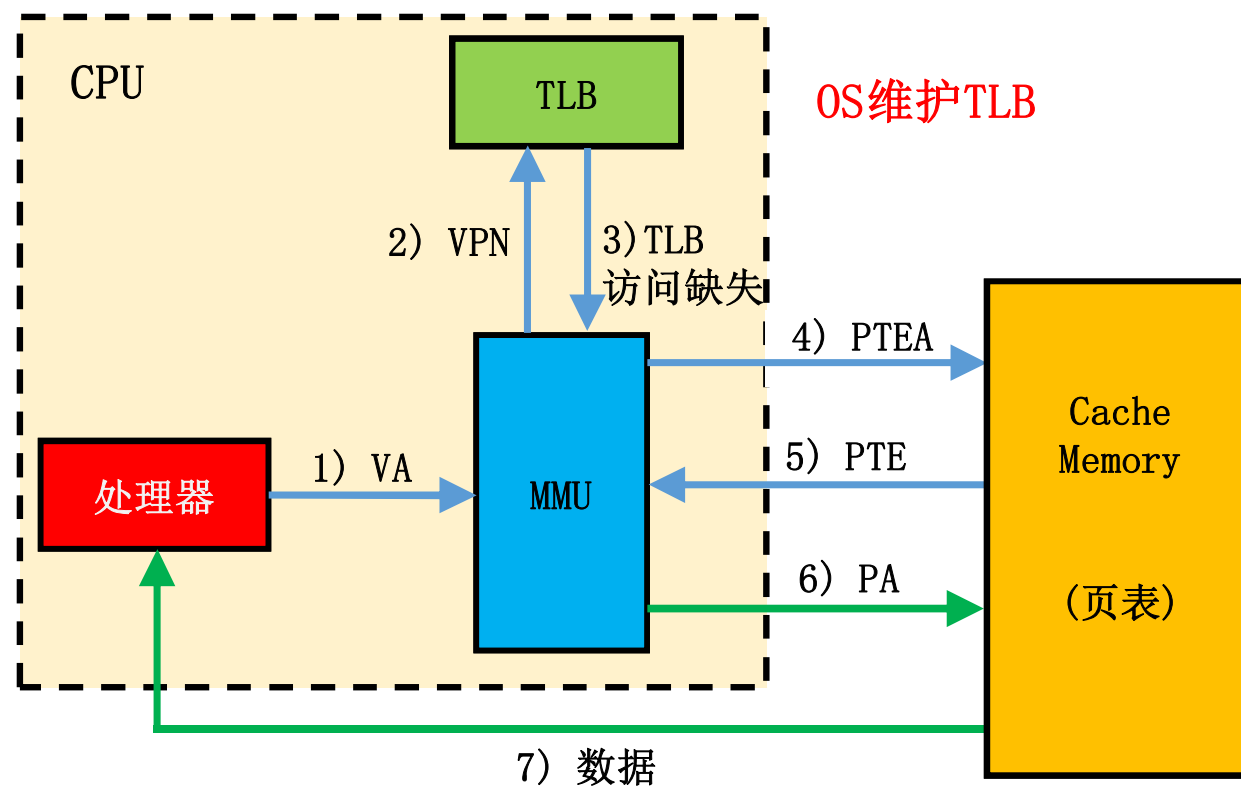
经快慢表实现内部地址转换



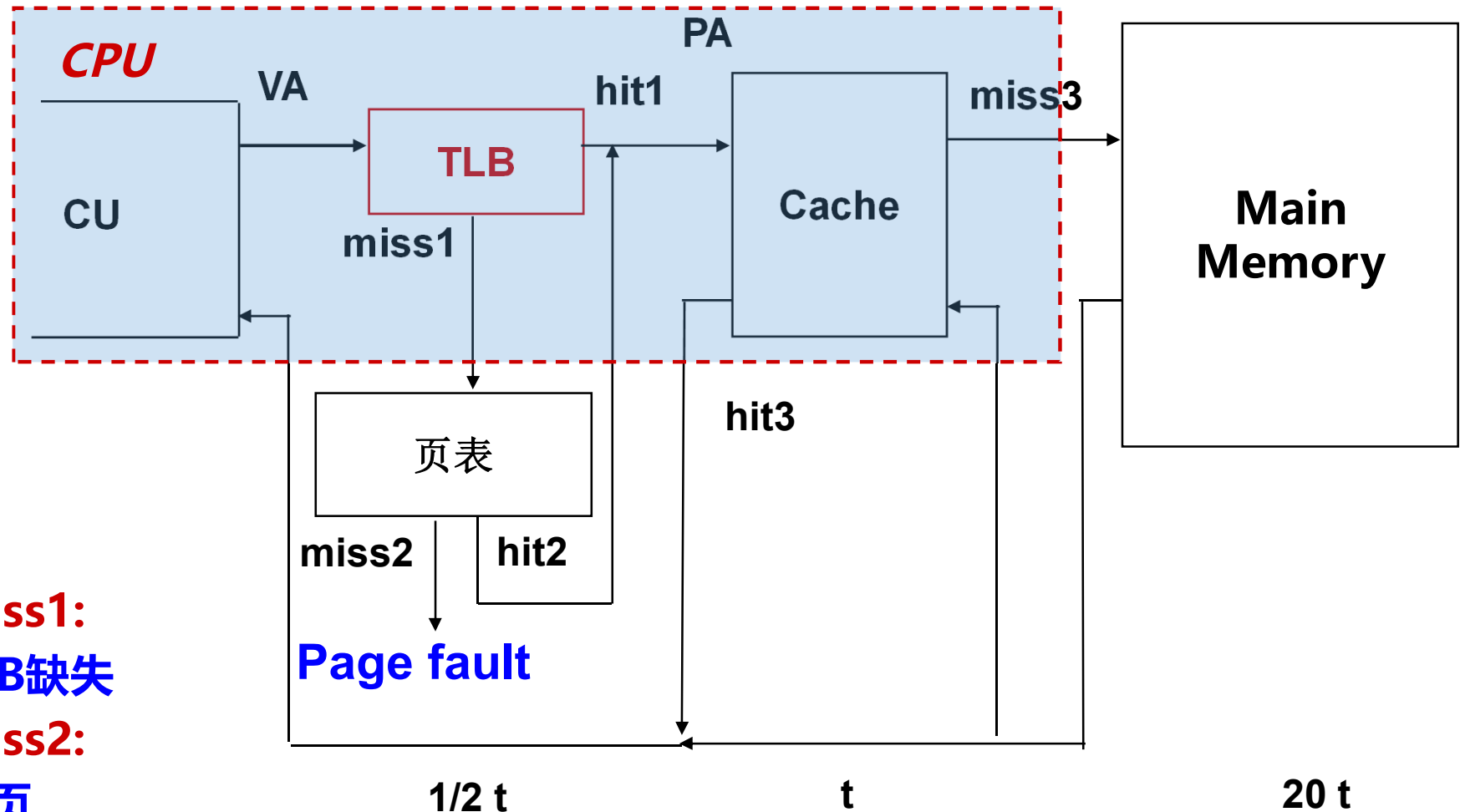
TLB 命中



TLB 缺失



Translation Look-Aside Buffers

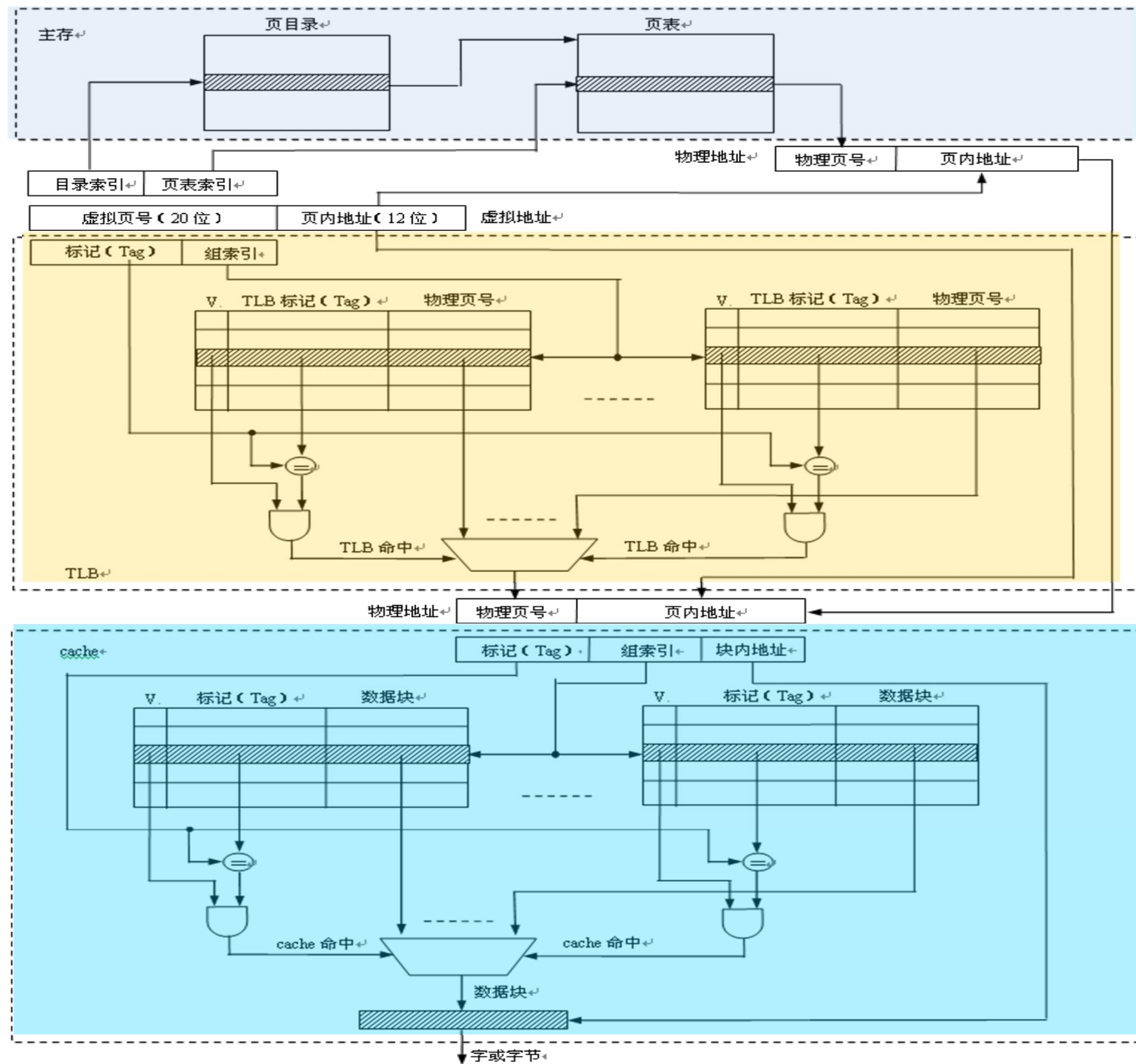
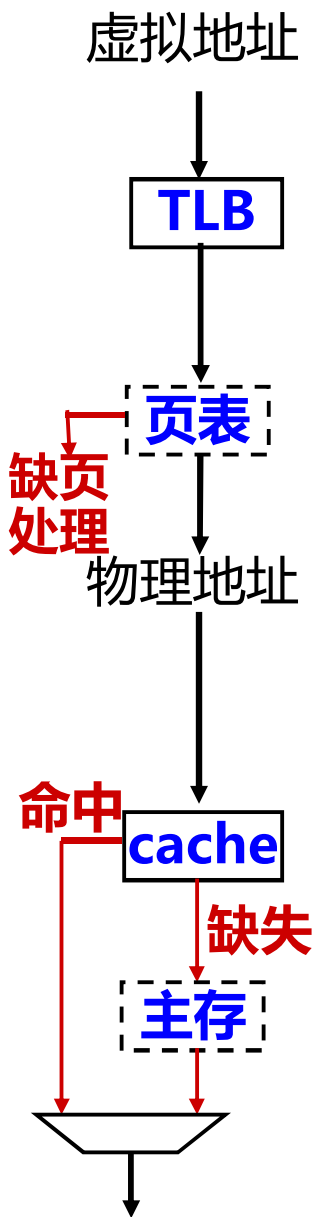


Miss1:
TLB缺失

Miss2:
缺页

Miss3:
PA 在主存中，但不在Cache中

TLB冲刷指令和Cache冲刷指令
都是操作系统使用的特权指令



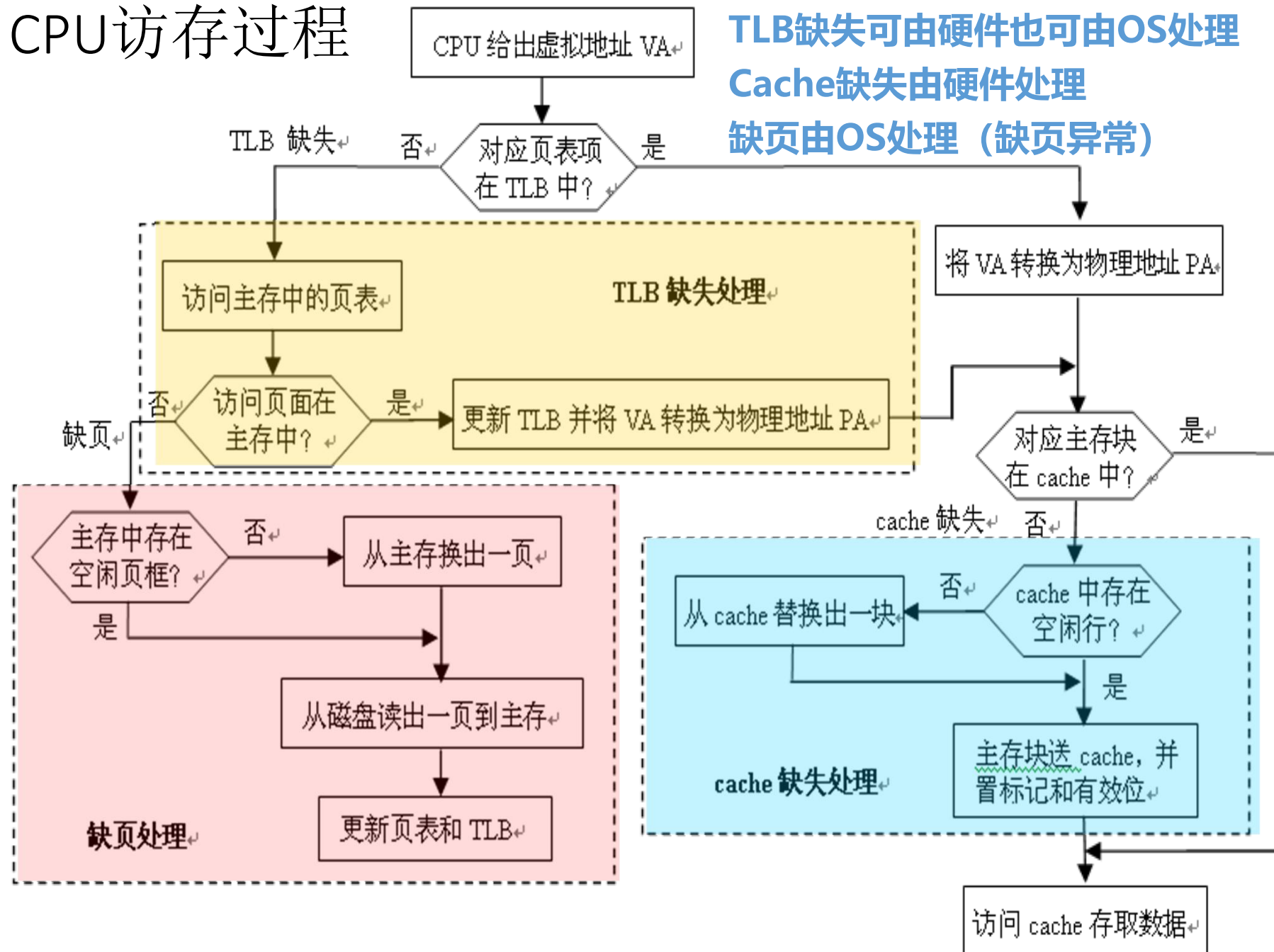
CPU访存过程

CPU 给出虚拟地址 VA

TLB缺失可由硬件也可由OS处理

Cache缺失由硬件处理

缺页由OS处理（缺页异常）



举例：三种不同缺失的组合

TLB	Page table	Cache	Possible? If so, under what circumstance?
hit	hit	miss	可能，TLB命中则页表一定命中，但实际上不会查页表
miss	hit	hit	可能，TLB缺失但页表命中，信息在主存，就可能在Cache
miss	hit	miss	可能，TLB缺失但页表命中，信息在主存，但可能不在Cache
miss	miss	miss	可能，TLB缺失页表缺失，信息不在主存，一定也不在Cache
hit	miss	miss	不可能，页表缺失，信息不在主存，TLB中一定没有该页表项
hit	miss	hit	同上
miss	miss	hit	不可能，页表缺失，信息不在主存，Cache中一定也无该信息

最好的情况是hit、hit、hit，此时，访问主存几次？ 不需要访问主存！

以上组合中，最好的情况是？ hit、hit、miss和miss、hit、hit 访存1次

以上组合中，最坏的情况是？ miss、miss、miss 需访问磁盘、并访存至少2次

介于最坏和最好之间的是？ miss、hit、miss 不需访问磁盘、但访存至少2次

第4章 存储系统

4.1 概述

4.2 主存储器

4.3 高速缓冲存储器

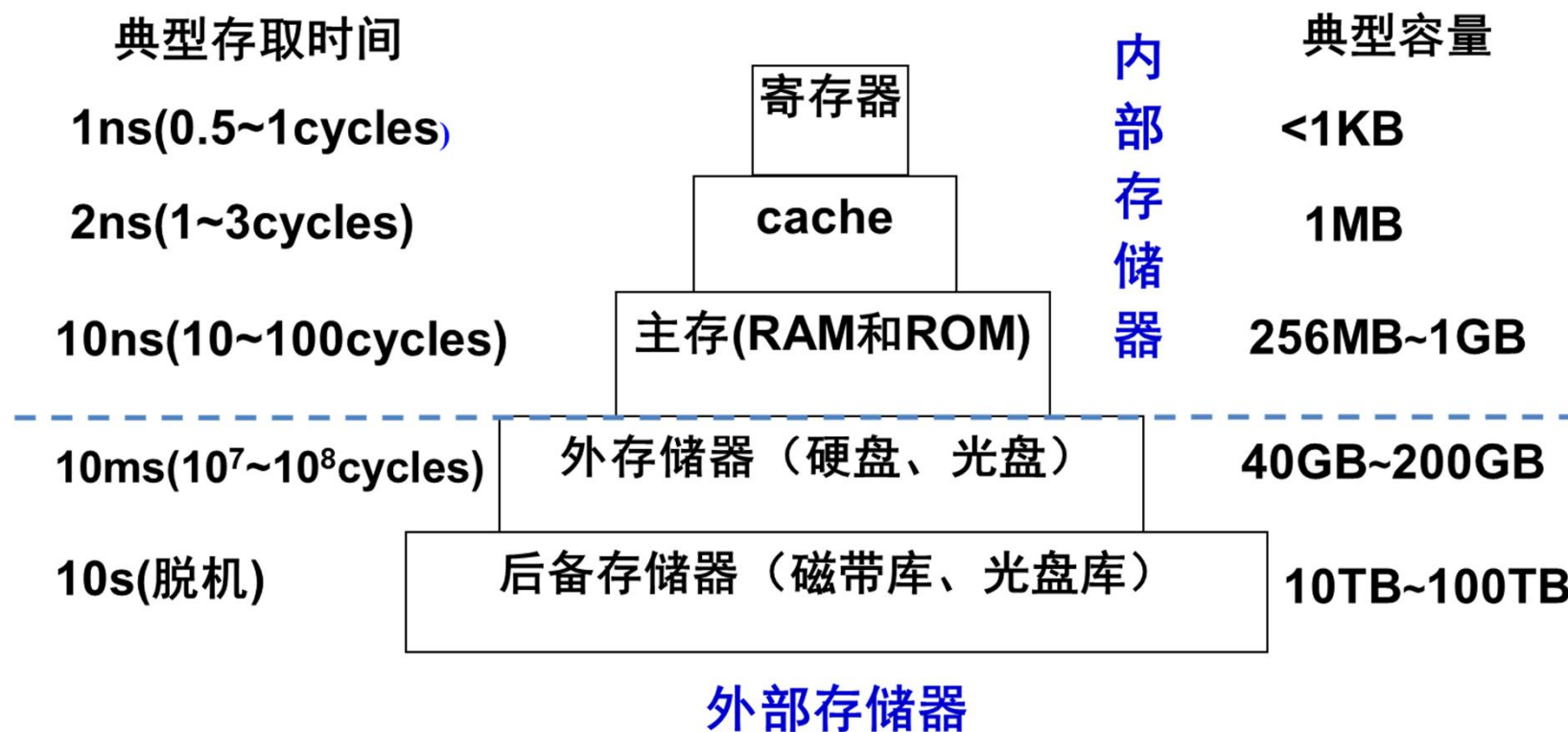
4.4 虚拟存储器

4.5 辅助存储器

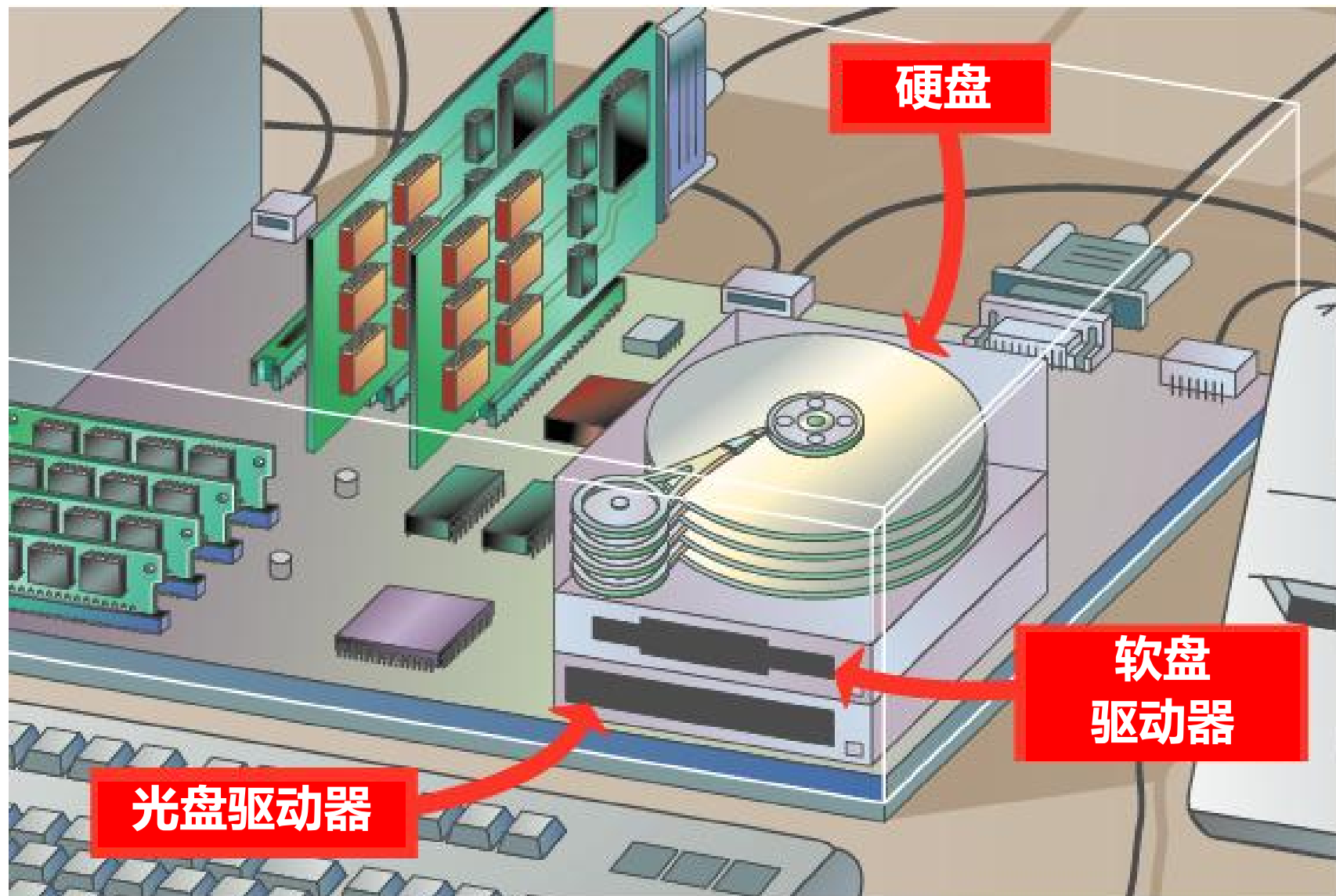
4.5 辅助存储器

一、概述

1. 特点 不直接与 CPU 交换信息



列出的时间和容量会随时间变化，但数量级相对关系不变。

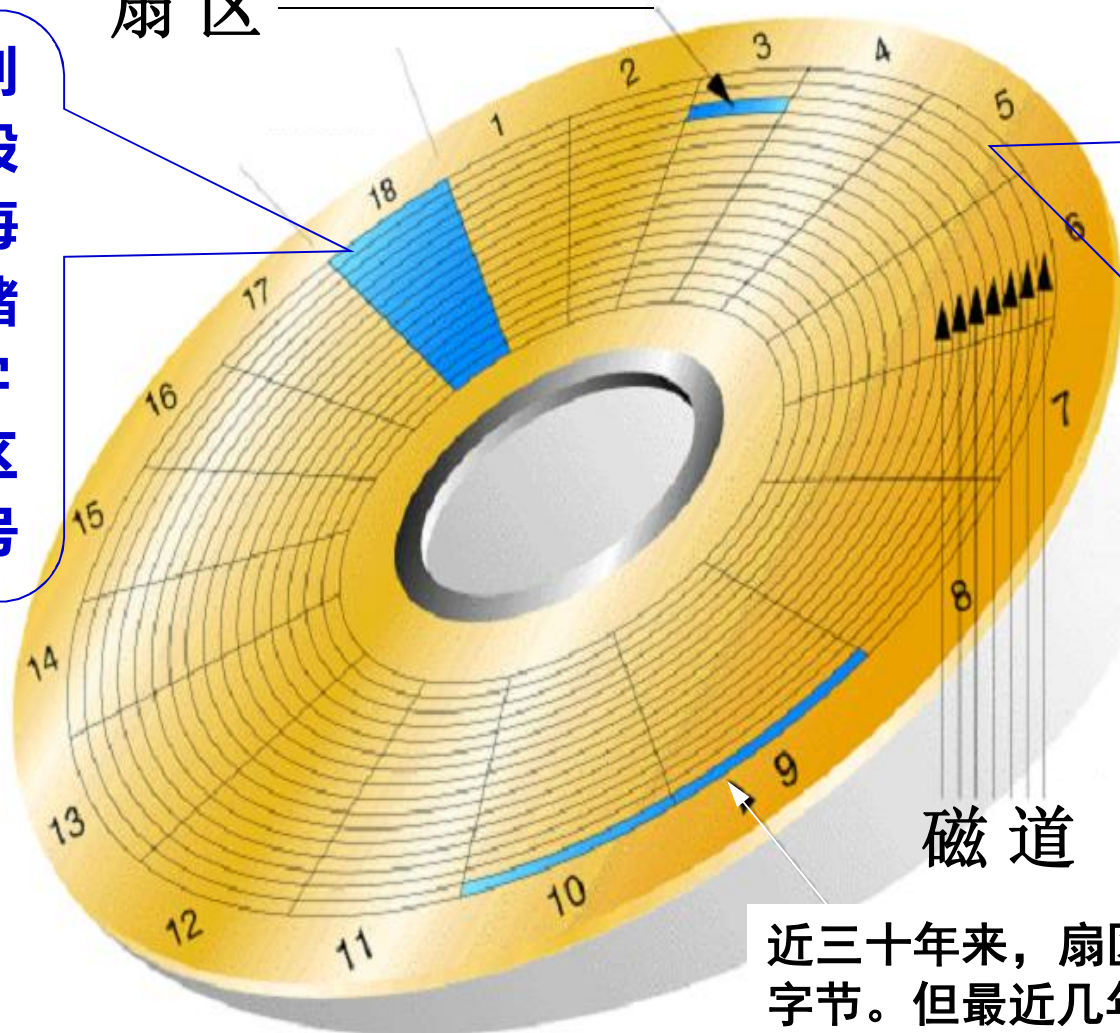


磁盘的磁道和扇区

4.5

每个磁道被划分为若干区段（扇区），每个扇区的存储容量为512字节。每个扇区都有一个编号

扇区



磁盘表面被分为许多同心圆，每个同心圆称为一个磁道。每个磁道都有一个编号，最外面的是0磁道。

磁道

近三十年来，扇区大小一直是512字节。但最近几年正迁移到更大、更高效的4096字节扇区，通常称为4K扇区。国际硬盘设备与材料协会（IDEMA）将之称为高级格式化。



4.5 辅助存储器

2. 磁表面存储器的技术指标

(1) 记录密度 道密度 D_t 位密度 D_b

(2) 存储容量 $C = n \times k \times s = \text{面数} \times \text{磁道数} \times \text{道容量}$

(3) 平均寻址时间 寻道时间 + 等待时间

辅存的速度 $\left\{ \begin{array}{l} \text{寻址时间} \\ \text{磁头读写时间} \end{array} \right.$

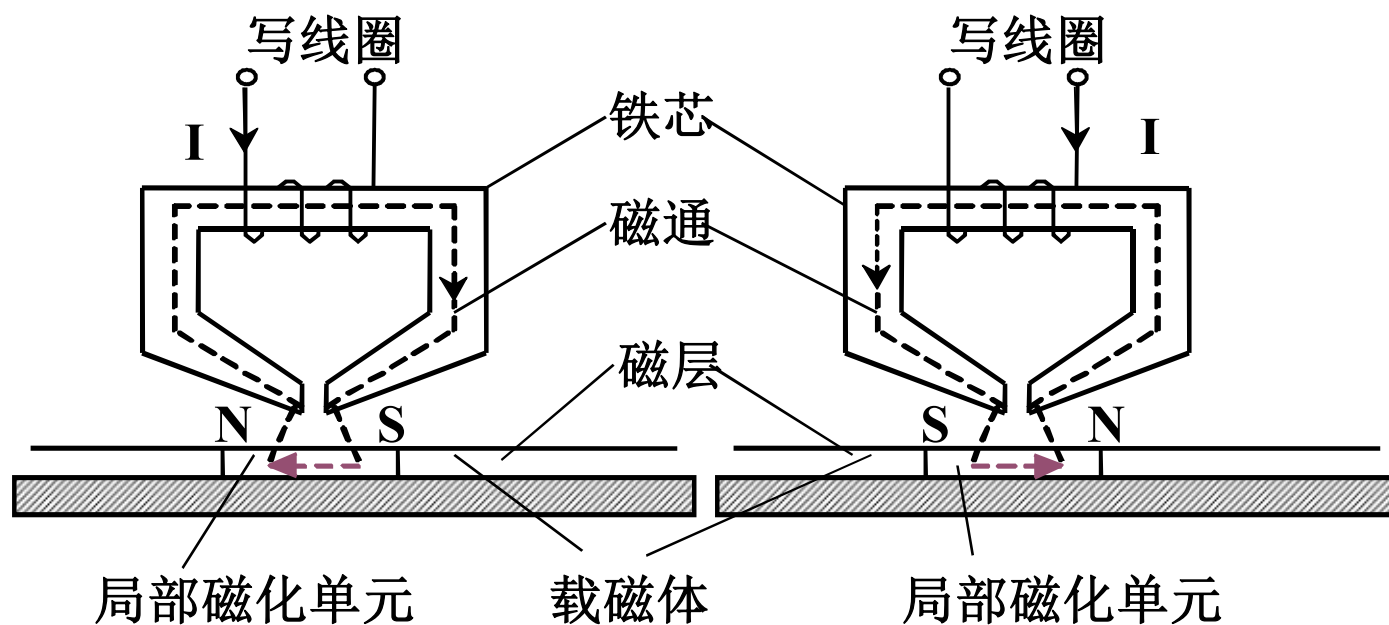
(4) 数据传输率 $D_r = D_b \times V = \text{记录密度} \times \text{线速度}$

(5) 误码率 出错信息位数与读出信息的总位数之比

二、磁记录原理和记录方式

1. 磁记录原理

写



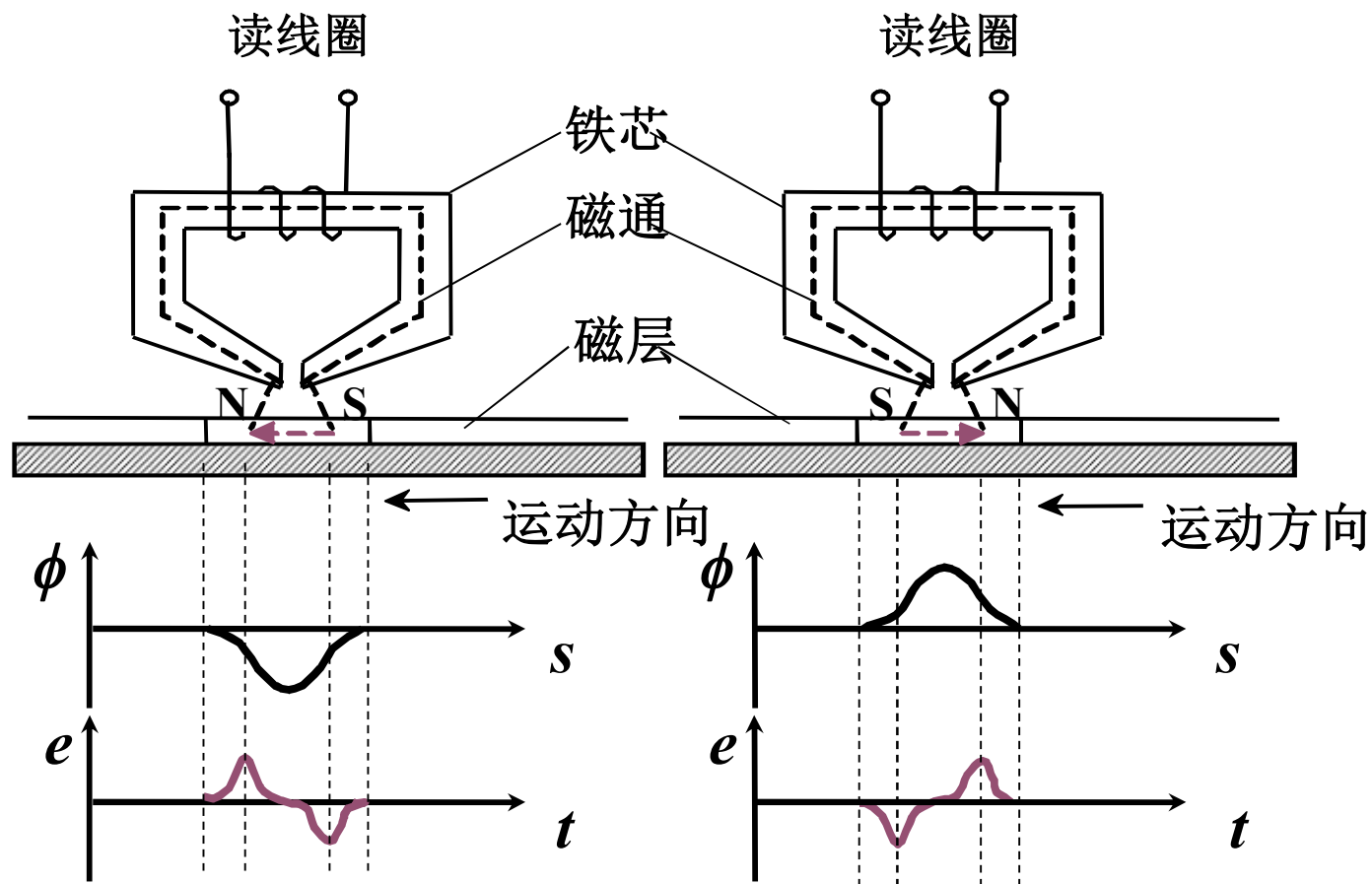
写入“0”

写入“1”

磁记录原理

4.4

读

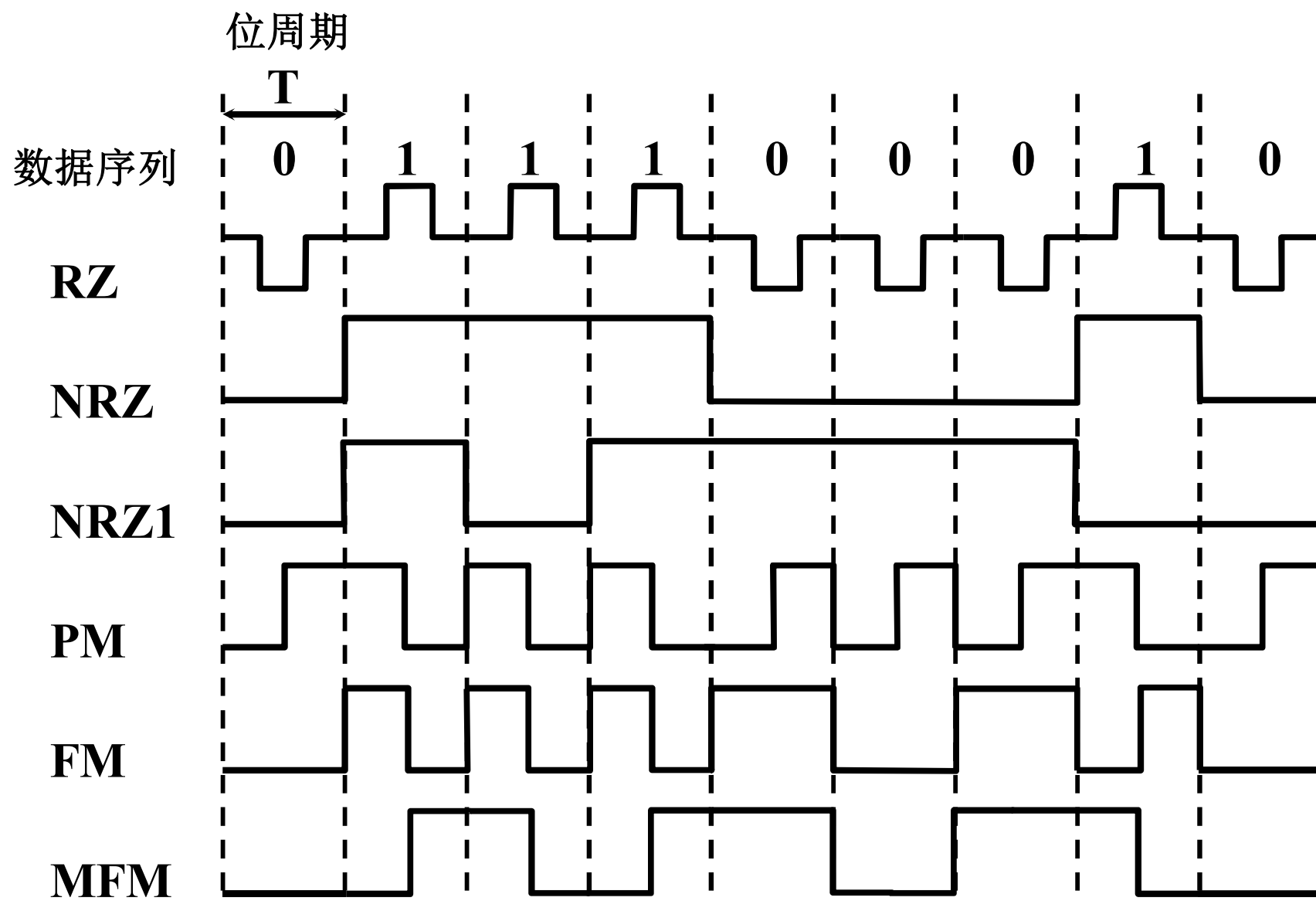


读出“0”

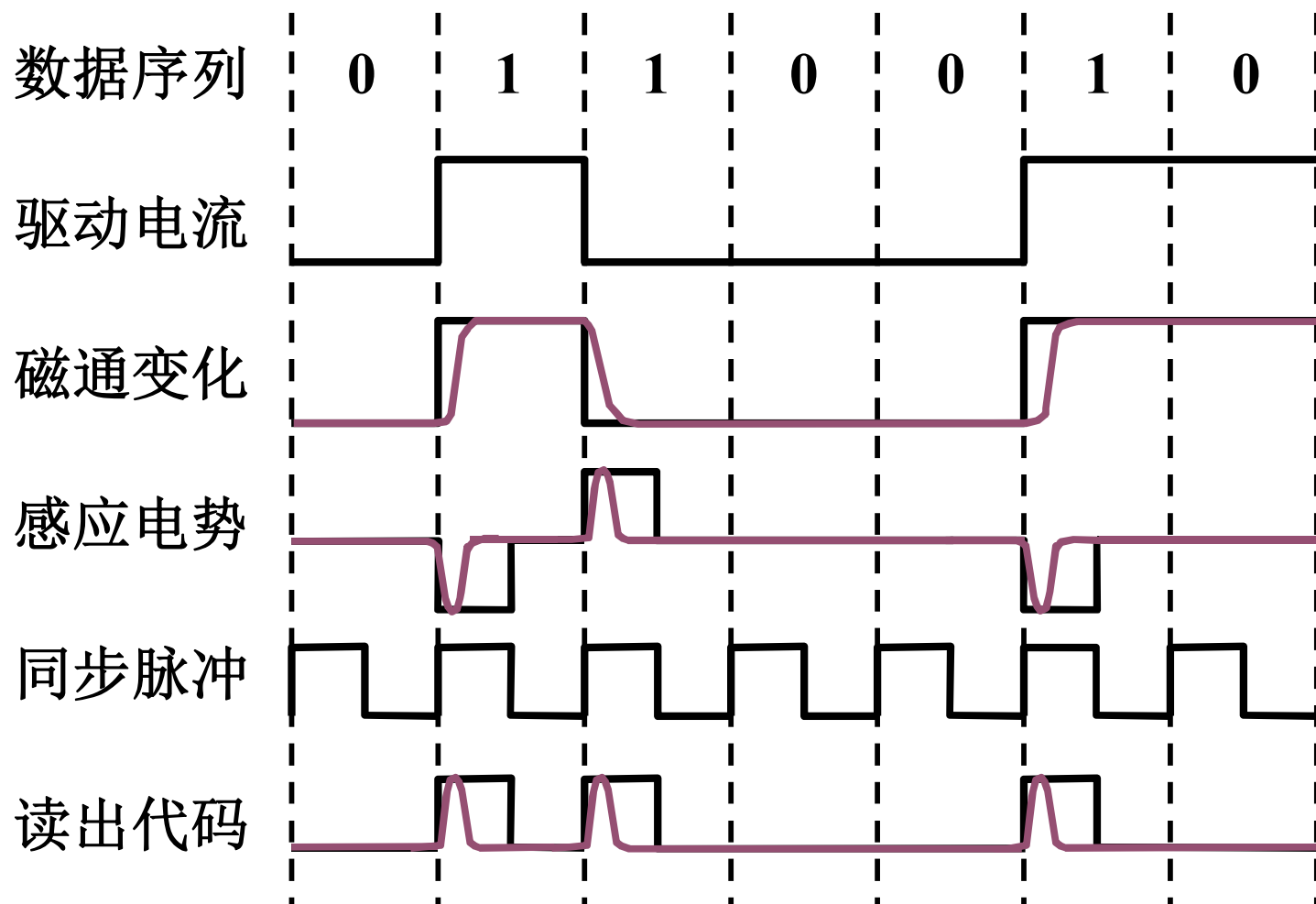
读出“1”

2. 磁表面存储器的记录方式

4.4



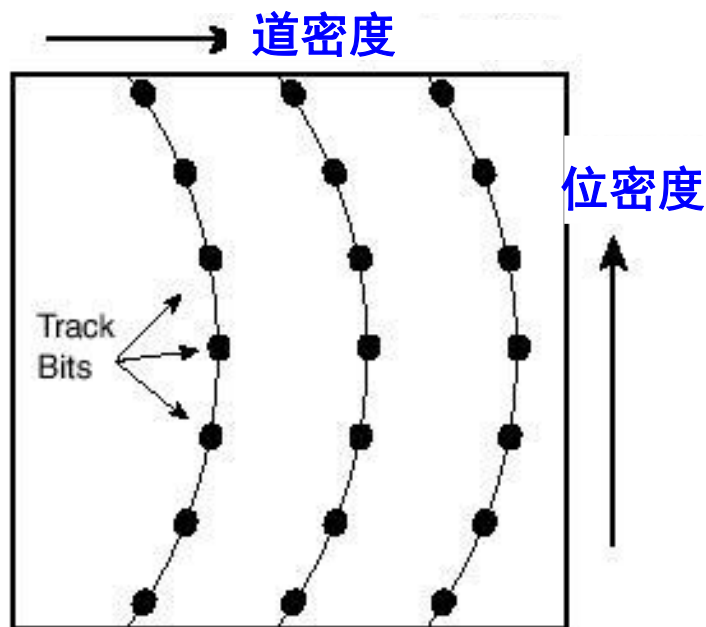
例 NRZ1 的读出代码波形



如何增大磁盘片的容量？

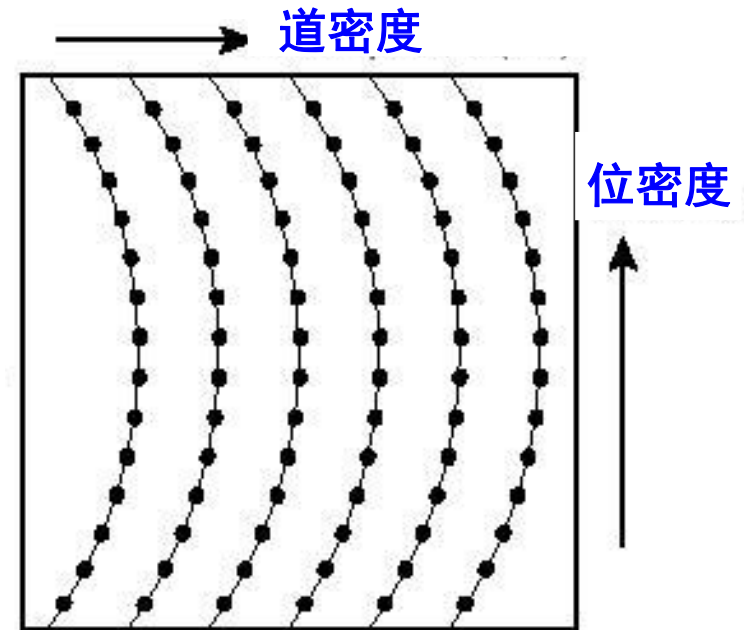
4.5

- 提高盘片上的信息记录密度！
 - 增加磁道数目——提高磁道密度
 - 增加扇区数目——提高位密度，并采用可变扇区数



低密度存储示意图

早期磁盘所有磁道上的扇区数相同，所以位数相同，内道上的位密度比外道位密度高

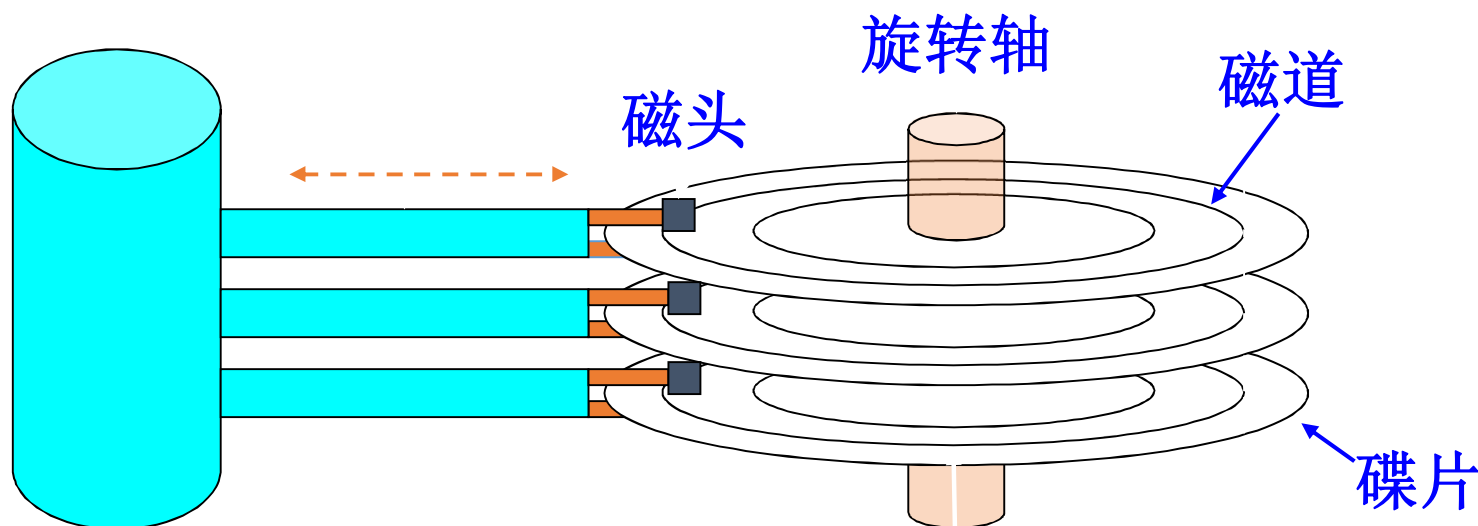


高密度存储示意图

现代磁盘磁道上的位密度相同，所以，外道上的扇区数比内道上扇区数多，使整个磁盘的容量提高

平均存取时间

4.5



硬盘的操作流程如下（CHS模式）：

所有磁头同步寻道（由柱面号控制）→ 选择磁头（由磁头号控制）→
被选中磁头等待扇区到达磁头下方（由扇区号控制）→ 读写该扇区中数据

- 磁盘上的信息以扇区为单位进行读写，平均存取时间为：

$T = \text{平均寻道时间} + \text{平均旋转等待时间} + \text{数据传输时间（忽略不计）}$

- **平均寻道时间**——磁头寻找到指定磁道所需平均时间 (大约5ms)
- **平均旋转等待时间**——指定扇区旋转到磁头下方所需平均时间 (大约4~6ms) (转速：4200 / 5400 / 7200 / 10000rpm)
- **数据传输时间**——(大约0.01ms / 扇区)

磁盘响应时间计算举例

- 假定每个扇区512字节，磁盘转速为5400 RPM，声称寻道时间（最大寻道时间的一半）为12 ms, 数据传输率为4 MB/s, 磁盘控制器开销为1 ms, 不考虑排队时间，则磁盘响应时间为多少？

$$\begin{aligned}
 \text{Disk Response Time} &= \text{Seek time} + \text{Rotational Latency} + \text{Transfer time} \\
 &\quad + \text{Controller Time} + \text{Queuing Delay} \\
 &= 12 \text{ ms} + 0.5 / 5400 \text{ RPM} + 0.5 \text{ KB} / 4 \text{ MB/s} + 1 \text{ ms} + 0 \\
 &= 12 \text{ ms} + 0.5 / 90 \text{ RPS} + 0.125 / 1024 \text{ s} + 1 \text{ ms} + 0 \\
 &= 12 \text{ ms} + 5.5 \text{ ms} + 0.1 \text{ ms} + 1 \text{ ms} + 0 \text{ ms} \\
 &= 18.6 \text{ ms}
 \end{aligned}$$

如果实际的寻道时间只有1/3的话，则总时间变为10.6ms，这样旋转等待时间就占了约50%！

$$12/3 + 5.5 + 0.1 + 1 = 10.6 \text{ ms}$$

所以，磁盘转速非常重要！

为什么实际的寻道时间可能只有1/3？

访问局部性使得每次磁盘访问大多在局部几个磁道，实际寻道时间变少！

三、硬盘存储器

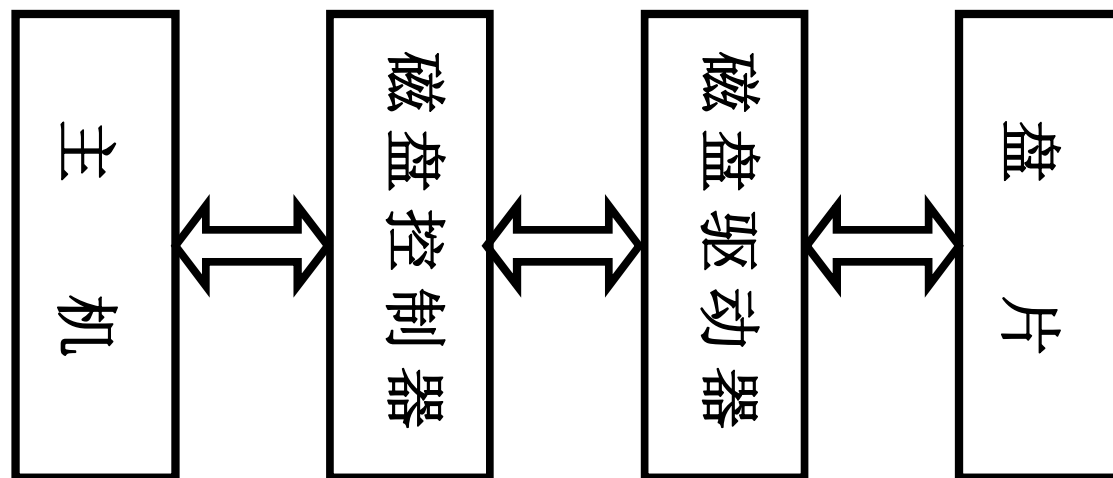
4.4

1. 硬盘存储器的类型

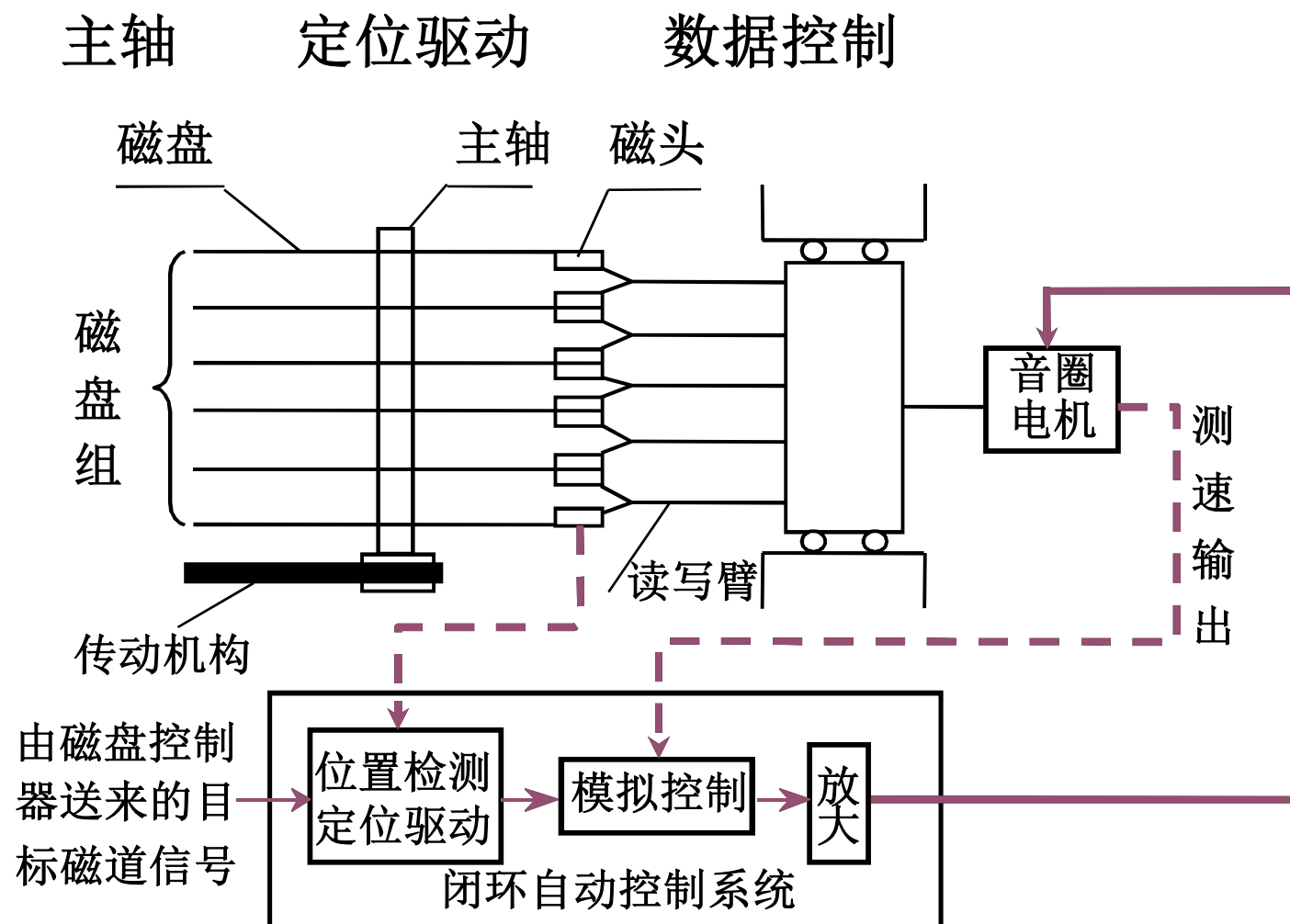
(1) 固定磁头和移动磁头

(2) 可换盘和固定盘

2. 硬盘存储器结构



(1) 磁盘驱动器



(2) 磁盘控制器

- 接受主机发来的命令，转换成磁盘驱动器的控制命令
- 实现主机和驱动器之间的数据格式转换
- 控制磁盘驱动器读写

磁盘控制器 是

主机与磁盘驱动器之间的 接口 { 对主机 通过总线
对硬盘 (设备)

(3) 盘片

由硬质铝合金材料制成

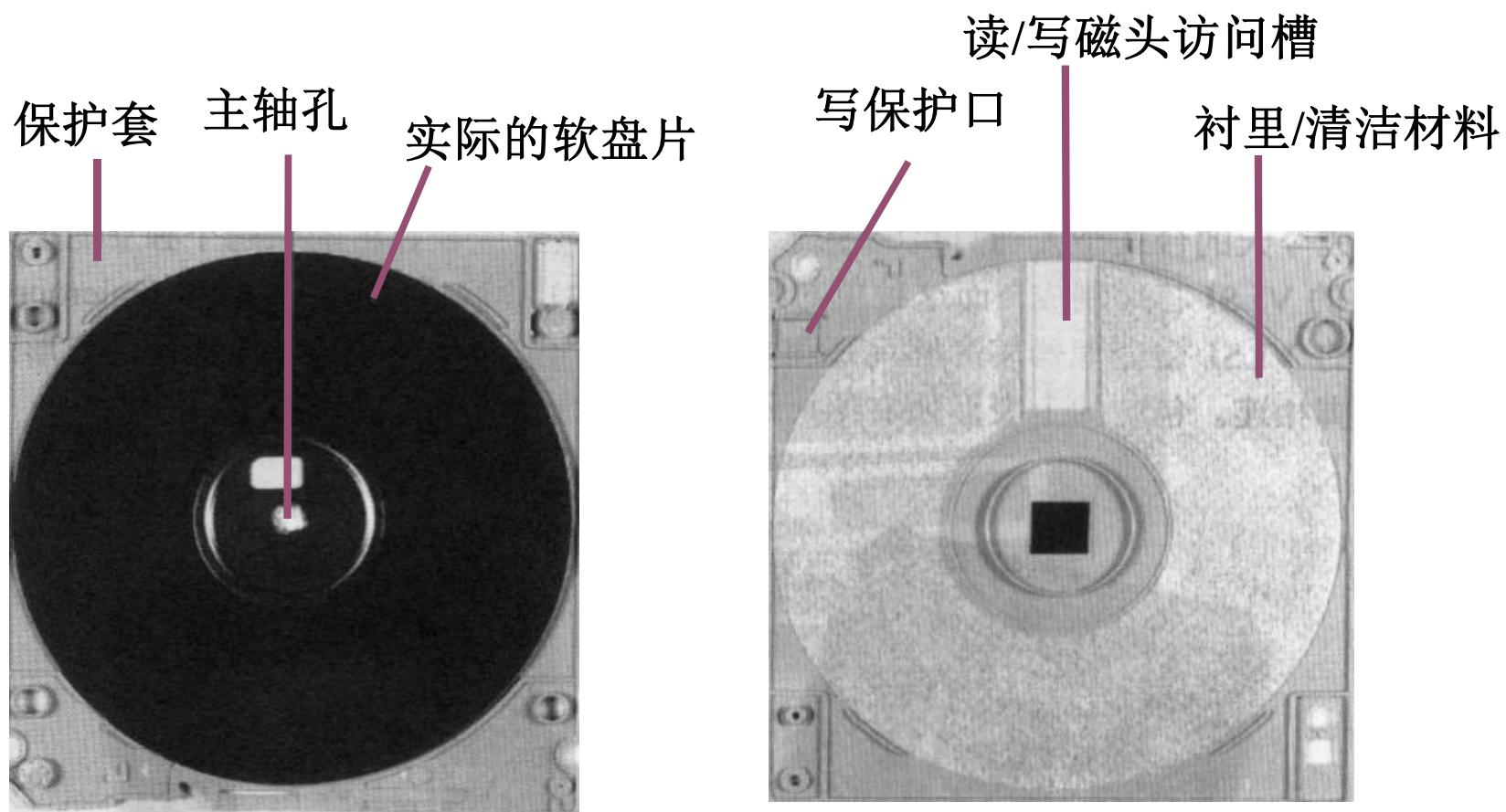
四、软磁盘存储器

1. 概述

	硬盘	软盘
速度	高	低
磁头	固定、活动	活动
	浮动	接触盘片
盘片	固定盘、盘组 大部分不可换	可换盘片
价格	高	低
环境	苛刻	

2. 软盘片

由聚酯薄膜制成



五、光盘

1. 概述

采用光存储技术

利用激光写入和读出

{ 第一代光存储技术
第二代光存储技术

采用非磁性介质

不可擦写

采用磁性介质

可擦写

2. 光盘的存储原理

只读型和只写一次型

热作用（物理或化学变化）

可擦写光盘

热效应(CD/DVD-RW)

热磁效应(MO)

第4章 作业

(T4. 3, T4. 5, T4. 15, T4. 16, T4. 20, T4. 28, T4. 32, T4. 39)

4.3 存储器的层次结构主要体现在什么地方,为什么要分这些层次,计算机如何管理这些层次?

4.5 什么是存储器的带宽?若存储器的数据总线宽度为 32 位,存取周期为 200 ns,则存储器的带宽是多少?

4.15 设 CPU 共有 16 根地址线,8 根数据线,并用 \overline{MREQ} (低电平有效) 作访存控制信号, R/\overline{W} 作读/写命令信号(高电平为读,低电平为写)。现有这些存储芯片:ROM(2 K×8 位,4 K×4 位,8 K×8 位),RAM(1 K×4 位,2 K×8 位,4 K×8 位)及 74138 译码器和其他门电路(门电路自定)。

试从上述规格中选用合适的芯片,画出 CPU 和存储芯片的连接图。要求如下:

(1) 最小 4 K 地址为系统程序区,4096~16383 地址范围为用户程序区。

(2) 指出选用的存储芯片类型及数量。

(3) 详细画出片选逻辑。

4.16 CPU 假设同上题,现有 8 片 8 K×8 位的 RAM 芯片与 CPU 相连。

(1) 用 74138 译码器画出 CPU 与存储芯片的连接图。

(2) 写出每片 RAM 的地址范围。

(3) 如果运行时发现不论往哪片 RAM 写入数据,以 A000H 为起始地址的存储芯片都有与其相同的数据,分析故障原因。

(4) 根据(1)的连接图,若出现地址线 A_{15} 与 CPU 断线,并搭接到高电平上,将出现什么后果?

4.20 欲传送的二进制代码为 1001101,用奇校验来确定其对应的汉明码,若在第 6 位出错,说明纠错过程。

第4章 作业

(T4. 3, T4. 5, T4. 15, T4. 16, T4. 20, T4. 28, T4. 32, T4. 39)

4.28 设主存容量为 256 K 字, Cache 容量为 2 K 字, 块长为 4。

- (1) 设计 Cache 地址格式, Cache 中可装入多少块数据?
- (2) 在直接映射方式下, 设计主存地址格式。
- (3) 在四路组相联映射方式下, 设计主存地址格式。
- (4) 在全相联映射方式下, 设计主存地址格式。
- (5) 若存储字长为 32 位, 存储器按字节寻址, 写出上述三种映射方式下主存的地址格式。

4.32 设某机主存容量为 4 MB, Cache 容量为 16 KB, 每字块有 8 个字, 每字 32 位, 设计一个四路组相联映射(即 Cache 每组内共有 4 个字块)的 Cache 组织。

- (1) 画出主存地址字段中各段的位数。
- (2) 设 Cache 的初态为空, CPU 依次从主存第 0, 1, 2, ..., 89 号单元读出 90 个字(主存一次读出一个字), 并重复按此次序读 8 次, 问命中率是多少?
- (3) 若 Cache 的速度是主存的 6 倍, 试问有 Cache 和无 Cache 相比, 速度约提高多少倍?

4.39 某磁盘存储器转速为 3 000 r/min, 共有 4 个记录盘面, 每毫米 5 道, 每道记录信息 12 288 字节, 最小磁道直径为 230 mm, 共有 275 道, 求:

- (1) 磁盘存储器的存储容量。
- (2) 最高位密度(最小磁道的位密度)和最低位密度。
- (3) 磁盘数据传输率。
- (4) 平均等待时间。