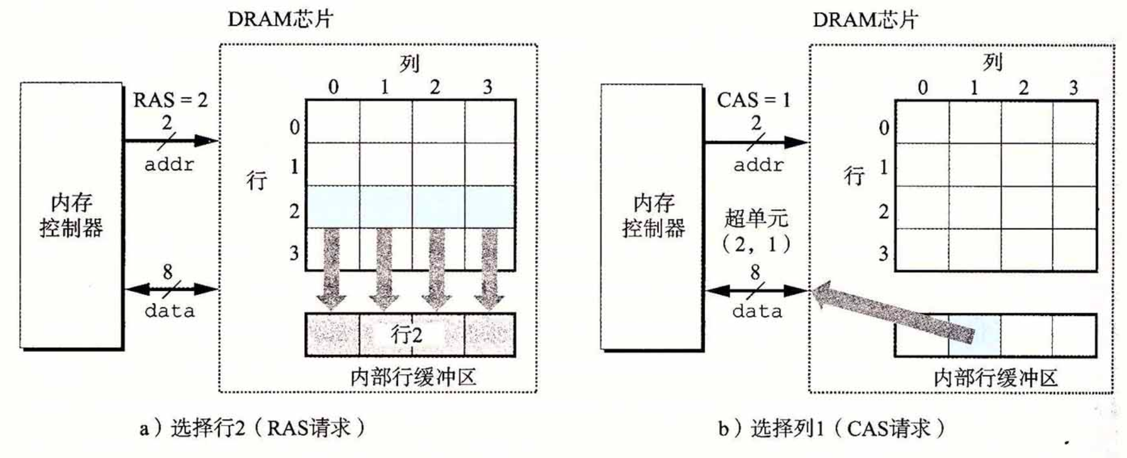
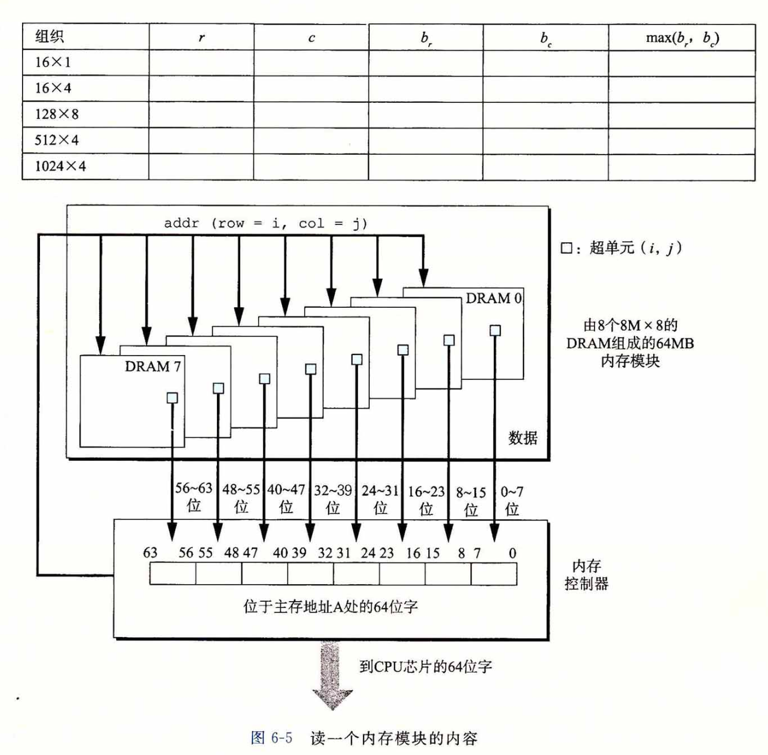
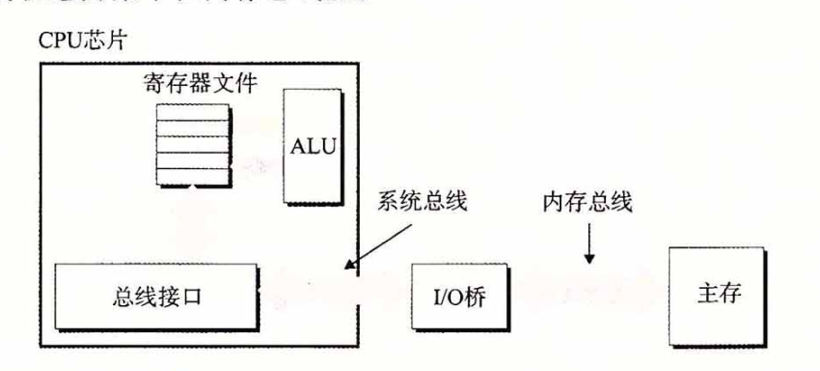
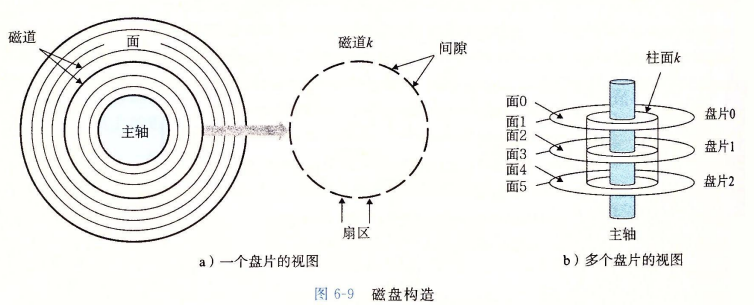
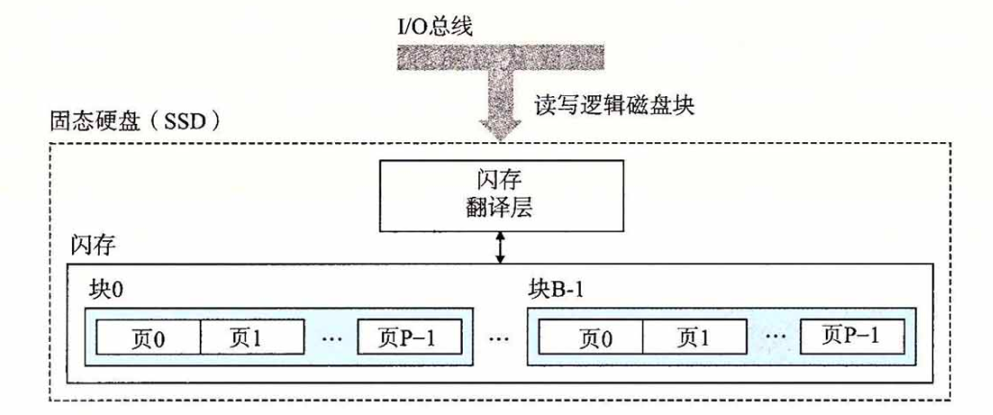
# 第6章 存储器层次结构

* **6.1存储技术**
  + **6.1.1随机访问存储器（RAM）**
    - **SRAM**
      * 作为高速缓存，可以在CPU芯片上或芯片外
      * 每个位存储在一个双稳态的结构单元，每个单元用一个六晶体管电路实现
      * 抗干扰能力强
    - **DRAM**
      * 作为主存以及图形系统的帧缓冲区
      * 每个位存储为对一个电容的充电，每个单元用一个电容和一个访问晶体管实现
      * 对干扰非常敏感（数码照相机和摄像机传感器）
      * 10～100ms内失去电荷，内存系统需要周期性读出然后重写/通过纠错码发现并纠正任何单个的错误位
    - **传统DRAM**
      * 单元（位）被分为d个超单元，每个超单元由w个单元组成；超单元被组织为矩阵（降低地址引脚数量）
      * 数据引脚和地址引脚，内存控制器先后发送RAS和CAS请求，二者共享相同的地址引脚（增加了访问时间）
      * 每个芯片被连接到内存控制器，内存控制器可以一次传送w位（一个超单元）（传入/传出）
    - **增强DRAM**
      * 快页模式DRAM（FPM DRAM）
        1. 复用行缓冲区的内容而非丢弃
        2. (RAS, CAS)\*3 -> RAS, CAS, CAS, CAS
      * 扩展数据输出DRAM（EDO DRAM）
        1. FPM DRAM的增强形式
        2. 允许各个CAS信号在时间上靠得更紧
      * 同步DRAM（SDRAM）
        1. 控制信号与驱动内存控制器的外部时钟信号相同
        2. 比异步DRAM（前面三种）更快
      * 双倍速率数据同步DRAM（DDR SDRAM）
        1. SDRAM的增强形式
        2. 使用两个时钟沿作为控制信号，使速度翻倍
      * 视频RAM（VRAM）
        1. 用于图形系统帧缓冲区，与FPM DRAM类似
        2. 输出通过依次对内部缓冲区的整个内容进行移位得到
        3. 允许对内存并行读写
    - **内存模块**
      * DRAM封装在内存模块中，内存模块插到主板的扩展槽上
      * 将多个内存模块连接到内存控制器可以聚合成主存



* + - **访问主存**

I/O桥接器执行系统总线信号与内存总线信号之间的翻译，包括内存控制器

* + - **非易失性存储器**
      * 只读存储器（ROM）
        1. 有的既可以读也可以写
      * 可编程ROM（PROM）
        1. 只能被编程一次
        2. 每个存储器单元有一种熔丝，只能用高电流熔断一次
      * 可擦写可编程ROM（EPROM）
        1. UV照过透明石英窗口，单元被清除为0
        2. 编程通过一种写入1的特殊设备完成
        3. 1000次
      * 电子可擦写可编程ROM（EEPROM）
        1. 不需要独立的编程设备
        2. 105次
      * 闪存
        1. 基于EEPROM
        2. 固态硬盘（SSD）
      * 固件
        1. 存储在ROM中的程序
        2. 通电后运行
  + **6.1.2磁盘存储**
    - 磁盘构造
    - 磁盘容量
      * 记录密度
      * 磁道密度
      * 面密度
        1. 多区记录技术
    - 磁盘操作
      * 寻道时间
        1. 读写头冲撞
        2. 3～9ms，最大20ms
      * 旋转时间
        1. 最大旋转时间的一半
        2. 与寻道时间大致相等
      * 传送时间
        1. 一个扇区需要旋转多久
        2. 很短
    - 逻辑磁盘块
      * 逻辑块序列+磁盘控制器
    - 连接I/O设备
      * 系统总线和内存总线与CPU相关，I/O总线与CPU无关
      * I/O总线较慢，但可连接多种I/O设备
        1. USB
        2. 图形卡/适配器
        3. 主机总线适配器（SCSI贵，支持多个磁盘，SATA只支持一个）
    - 访问磁盘
      * 内存映射I/O
        1. 指令1发送一个命令字，告诉磁盘发起一个读；同时发送其他参数
        2. 指令2指明应该读的逻辑块号
        3. 指明存储磁盘扇区内容的主存地址
      * CPU发出指令后，磁盘执行读，CPU做其他事
      * 磁盘控制器接收到CPU命令后
        1. 进行地址翻译（逻辑块号->扇区地址）
        2. 读扇区内容
        3. 将内容直接传送到主存（直接内存访问，DMA）
        4. 向CPU发送中断信号
  + **6.1.3固态硬盘**
    - 读比写快
    - 只有在一页所属的块整个被擦除（全设置为1）后才能写这一页；块被擦除后每一个页就可以直接写一次了（有磨损）
  + **6.1.4 存储技术趋势**
    - DRAM、磁盘与CPU之间的性能差距在增大->因此引入SRAM
* **6.2局部性（略）**
* **6.3存储器层次结构**
  + **6.3.1存储器层次结构中的缓存**
    - 缓存块
      * 每个块都有唯一的地址或名字
      * 一般是固定大小的，也可以是可变大小的
      * 相邻层次间块大小一般是相同 的，但其他层次对之间有不同的块大小
    - 缓存不命中
      * 冷不命中：缓存整个是空的
      * 冲突不命中：限制性的放置策略
      * 容量不命中：工作集太大
* **6.4高速缓存存储器**
  + **6.4.1通用的高速缓存存储器组织结构**
  + **6.4.2直接映射高速缓存**
  + **6.4.3组相连高速缓存**
  + **6.4.4全相连高速缓存**
  + **6.4.5有关写的问题**
    - 写命中
      * 直写
        1. 每次写都会引起总线流量
      * 写回
        1. 推迟更新
        2. 必须为每个缓存行维护一个额外的修改位
    - 写不命中
      * 写分配（+写回）
      * 非写分配（+直写）
  + **6.4.6一个真实的高速缓存层次结构的解剖**
  + **6.4.7高速缓存参数的性能影响**
    - 不命中率/命中率/命中时间/不命中处罚
    - 高速缓存大小的影响
      * 大，可能会提高命中率
      * 大，可能会增加命中时间
    - 块大小的影响
      * 大，能更好利用程序中的空间局部性，可能提高命中率
      * 给定缓存大小，块越大行数越少，会损害时间局部性为主的程序的命中率
      * 大，不命中处罚可能增加
    - 相联度的影响
      * 大，降低出现抖动的可能性
      * 大，增加命中时间
    - 写策略的影响
* **6.5编写高速缓存友好的代码**
* **6.6综合：高速缓存对程序性能的影响**
  + **存储器山**