# 存储系统

马士兵教育研究院

# 目录

1. 存储器的层次结构
2. 半导体存储器

- 存储器的层次结构
- 局部性原理

# 1.存储器的层次结构

## 存储器的层次结构

- 高速缓冲存储器

  Cache Memory（$L_1$~$L_3$缓存）

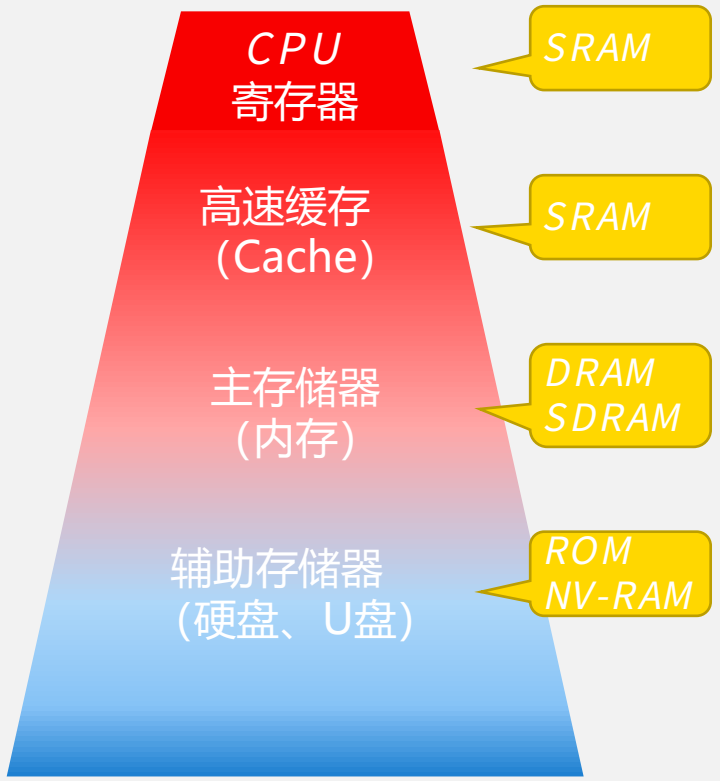
- 主存储器

  内存条

- 辅助存储器

  固定磁盘（硬盘、磁盘、闪存）

  可移动存储介质（U盘、光盘、磁带）



# 1.存储器的层次结构

## 存储器的层次结构

- 高速缓冲存储器

  Cache Memory（$L_1$~$L_3$缓存）

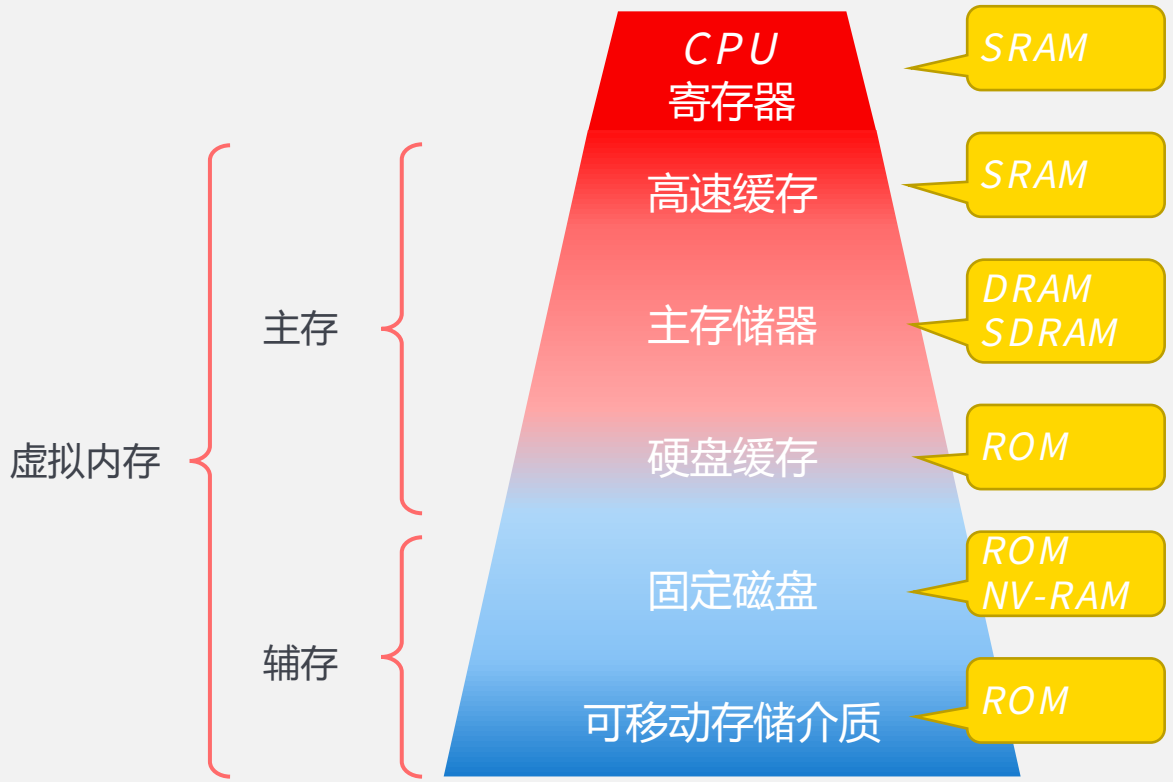
- 主存储器

  内存条

- 辅助存储器

  固定磁盘（硬盘、磁盘、闪存）

  可移动存储介质（U盘、光盘、磁带）



# 1.存储器的层次结构

## 局部性原理

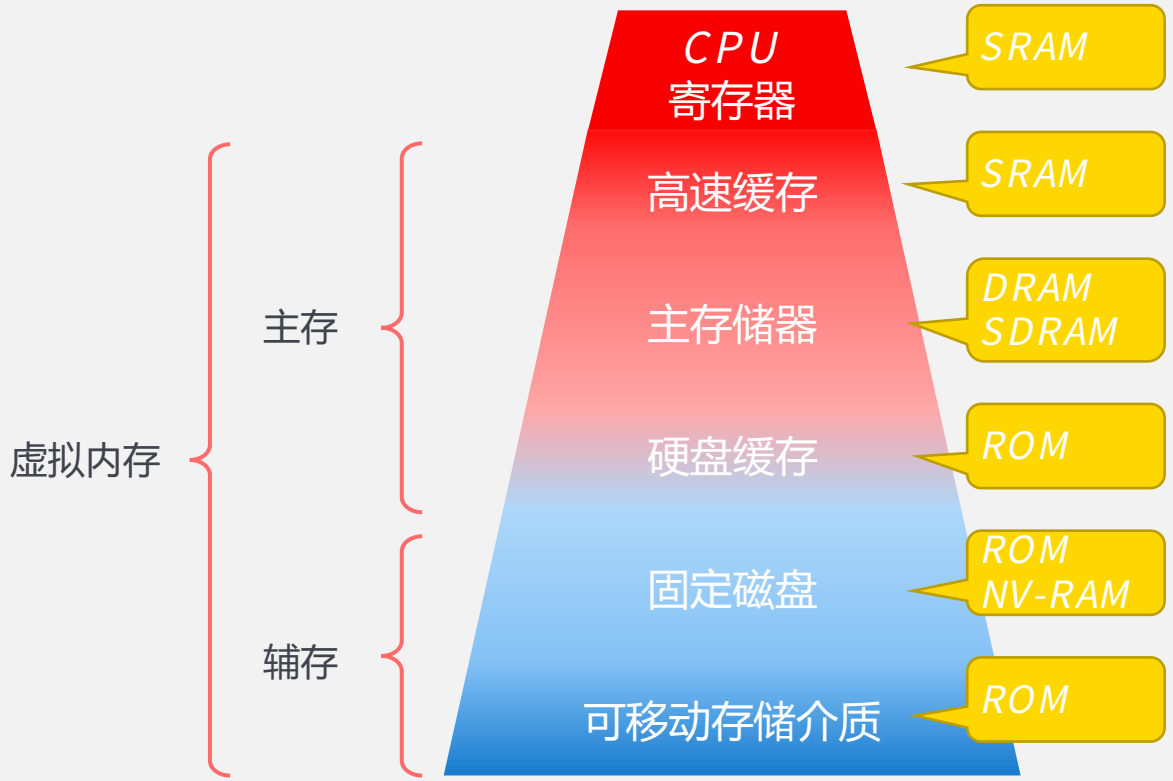
- 虚拟内存

  从逻辑上扩充内存容量的存储器系统

- 局部性原理

  时间局部性

  空间局部性



# 目录


1. 存储器的层次结构
2. 半导体存储器
   - 主存储器模型
   - 半导体存储器
     - RAM
     - ROM
   - Cache

# 2.半导体存储器

## 主存储器模型

- CPU与主存储器的连接

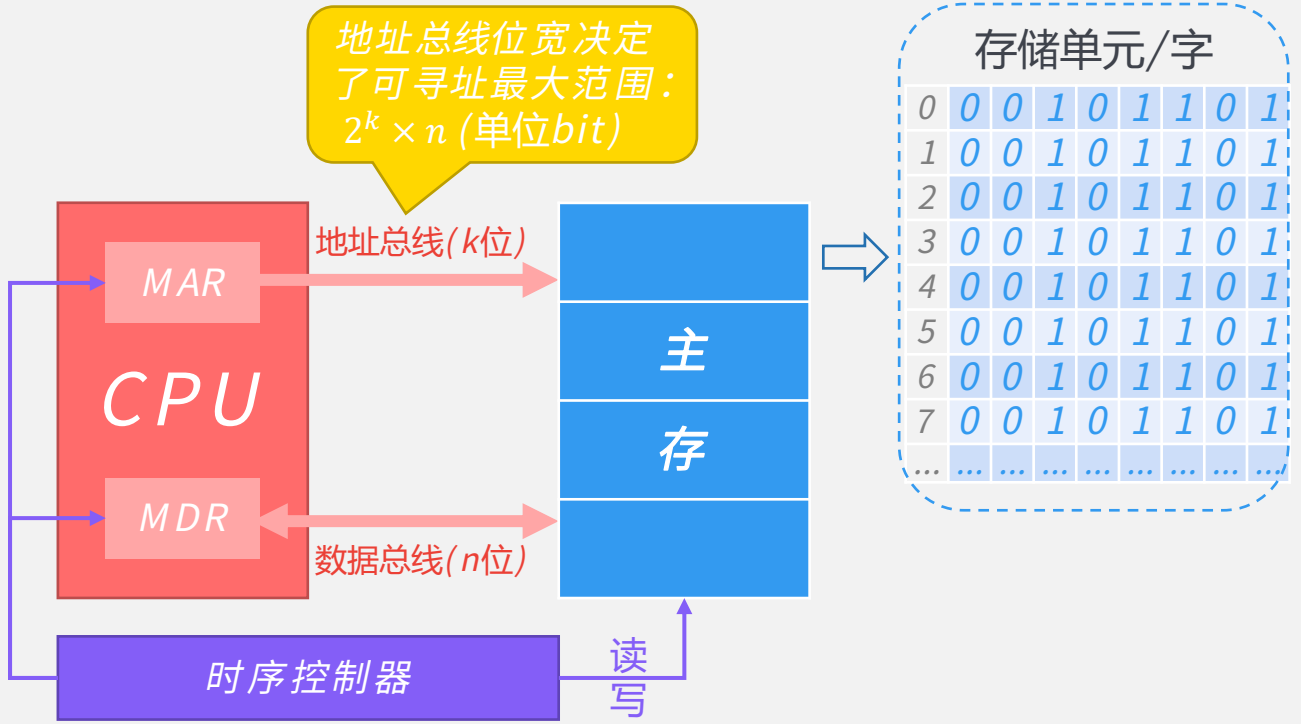
数据总线：

存取周期：两次连续存取的最小时间间隔

存储器总线宽度=数据总线位数=存储字长

数据传输率=存储器总线宽度/存取周期

单位时间内存取二进制信息的位数




地址总线：

存储容量：半导体存储芯片所能存储的二进制信息位数（单位bit）

可寻址最大范围：$2^k \times n$（单位bit，1Byte=8bit）

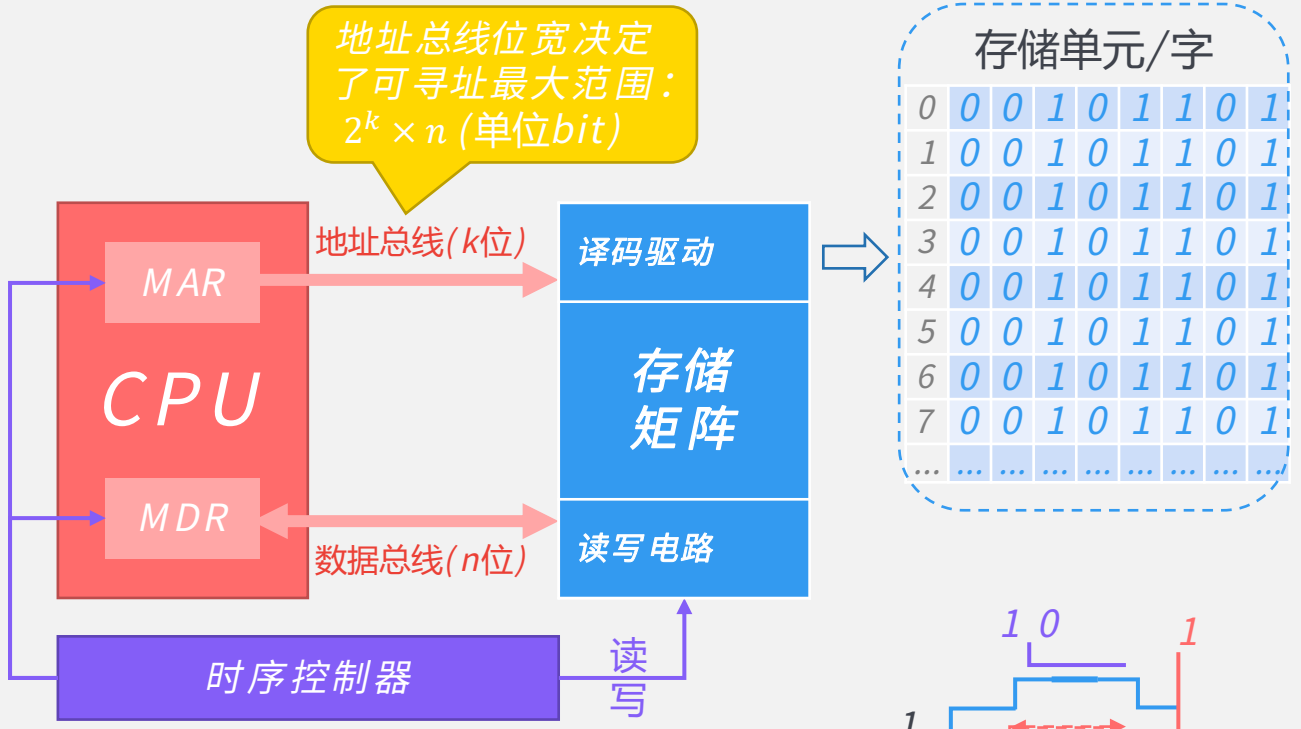
$1\mathrm{P} = 1\mathrm{T} \times 2^{10} = 1\mathrm{G} \times 2^{20} = 1\mathrm{M} \times 2^{30} = 1\mathrm{K} \times 2^{40} = 1\mathrm{B} \times 2^{50}$

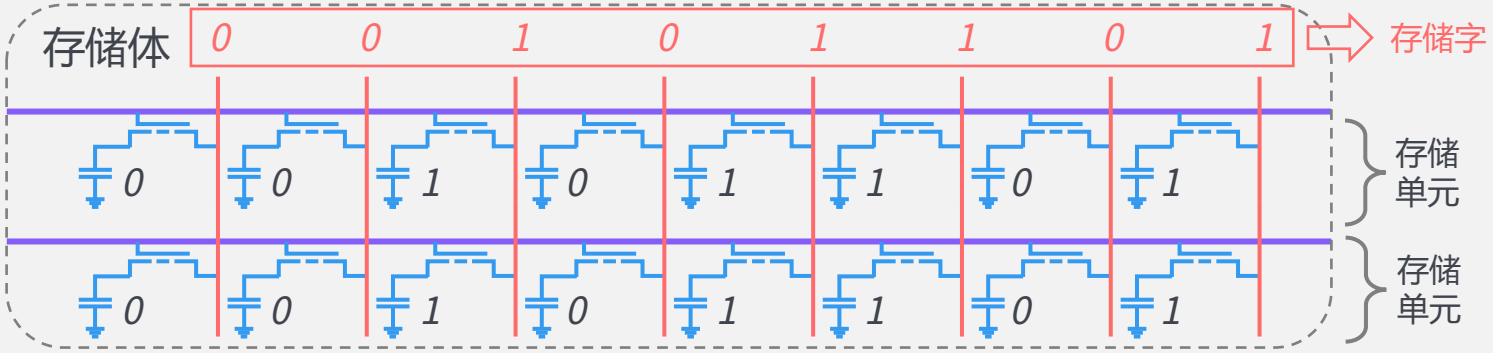
控制总线

# 2.半导体存储器

## 主存储器模型

- CPU与主存储器的连接
  - 数据总线
  - 地址总线
  - 控制总线







存储元

# 2.半导体存储器

## 半导体存储器

- RAM(Random Access Memory)

  随机存取存储器，可随机读写，断电易失

  静态（static）RAM：

  以触发器存储二进制位（6个MOS管）

  动态（dynamic）RAM：

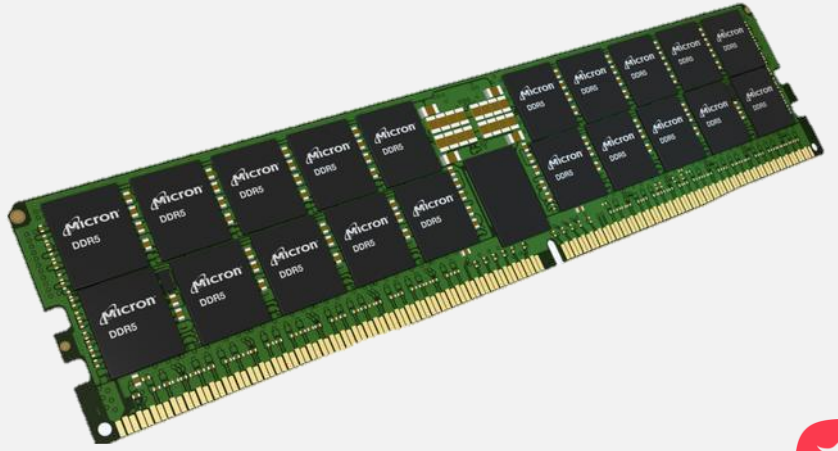
  一个/三个MOS管和一个电容存储二进制位

  刷新操作，给电容补电

  非易失性（non-volatile）RAM：

  采用CMOS管构成的低功耗SRAM存储单元

  使用锂电池作为后备电源



# 2.半导体存储器

## 半导体存储器

- SRAM与DRAM对比

静态（static）RAM：

以触发器存储二进制位（6个MOS管）

动态（dynamic）RAM：

一个/三个MOS管和一个电容存储二进制位

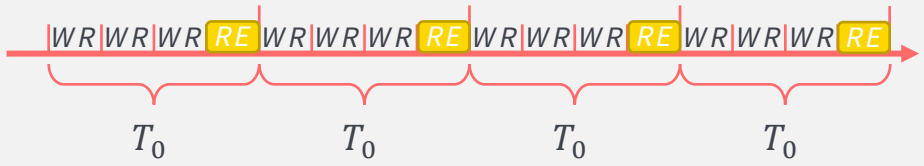
刷新操作，给电容补电

集中刷新：2ms内安排时间全部刷新

分散刷新：每次读完就刷新

异步刷新：每间隔一段时间刷新一行

| | SRAM | DRAM |
|---|---|---|
| 应用场景 | 寄存器、Cache | 主存 |
| 实现方式 | 触发器（6个MOS管） | 电容+一个MOS管 |
| 读写操作 | 读：查看触发器状态<br>写：改变触发器状态 | 读：连接电容，检测电流变化<br>写：给电容充/放电 |
| 破坏性读出 | 否 | 是 |
| 刷新操作 | 否 | 是 |
| 送入地址 | 行地址、列地址同时送入 | 行地址、列地址分两次送入 |
| 速度 | 快 | 慢 |
| 集成度 | 低 | 高 |
| 发热量 | 大 | 小 |
| 存储成本 | 高 | 低 |



# 2.半导体存储器

## 半导体存储器

- ROM(Read-Only Memory)

  只读存储器，可随机读，~~不可写入，~~非易失性

  ~~掩膜式只读存储器（MROM）~~

  ~~一次性可编程只读存储器（PROM）~~

  可擦除可编程~~只读~~存储器（EPROM）

    紫外线擦除（UV-EPROM）

      修改次数有限，写入耗时

    电擦除（EEPROM）

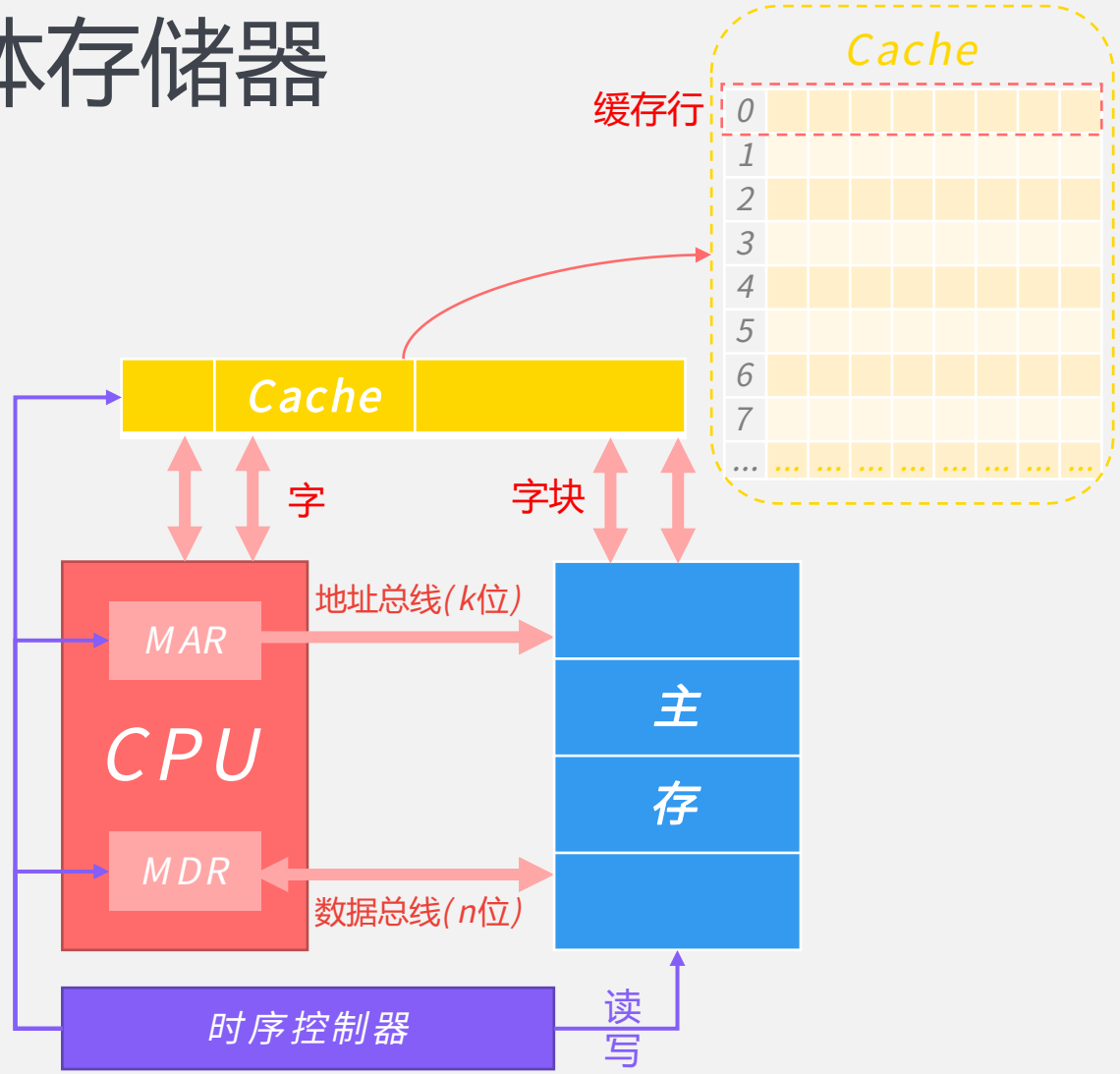
      闪速存储器（Flash Memory）

      固态硬盘（Solid State Drives）


# 2.半导体存储器

## 高速缓冲存储器Cache

- 基本概念
  - 介于CPU和主存之间，缓解二者速度矛盾
  - 局部性原理：时间/空间局部性
  - 访问顺序：CPU->Cache->主存
    - Cache中存放主存 *部分数据* 的拷贝（副本）
    - 缓存行/缓存块：与主存传送数据的基本单位（Byte）
  - 缓存命中率：85%+
    - Cache大小
    - Cache组织形式
    - 程序特性



# 2.半导体存储器

## 高速缓冲存储器Cache

- Cache的组织结构

**Cache存储体**

存放从主存调入的数据

由多个缓存行/字块构成：64B

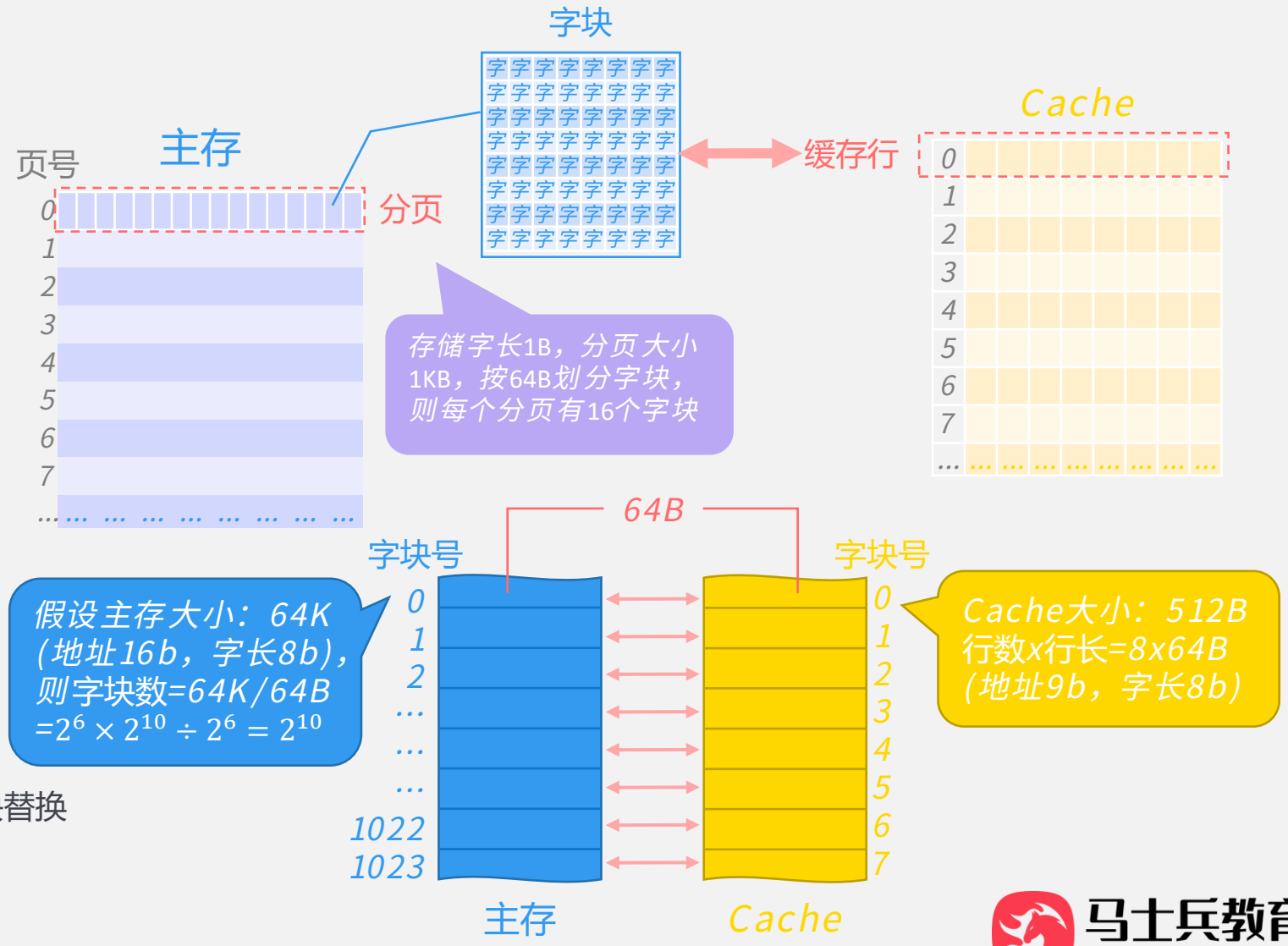
**地址转换机构/部件**

实现主存地址到缓存地址的转换

目录表，记录地址映像关系

替换机构/部件

按照一定策略（算法）进行数据块替换



# 2.半导体存储器

## 高速缓冲存储器Cache

- Cache与主存的映像

  全相连映像（映射）：空位随意放

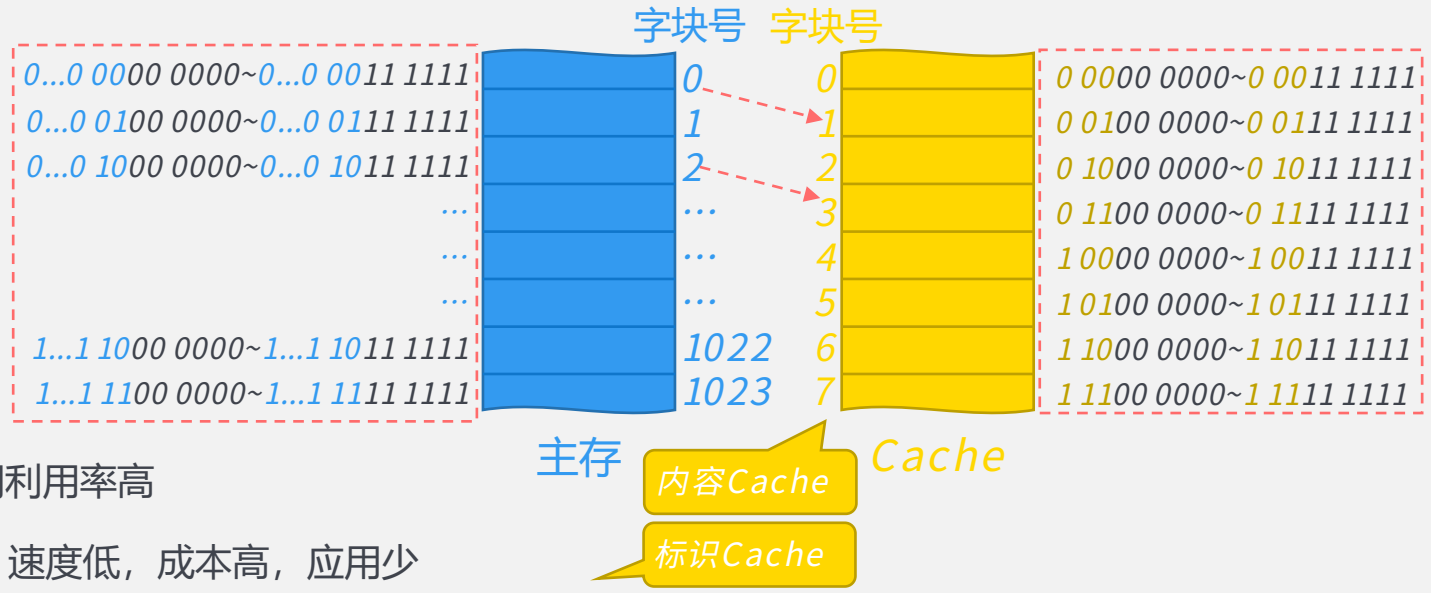
  任意主存字块放到Cache任意位置

  优点：命中率比较高，Cache存储空间利用率高

  缺点：每次访问都要与全部内容比较，速度低，成本高，应用少

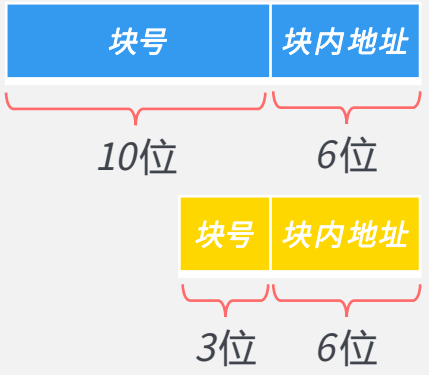
  直接映像

  组相连映像




将主存地址右移(>>>)6位得到块号，逐条与目录表主存块地址比较，然后获取缓存块地址

| | 主存块地址 | 缓存块地址 | 有效位 |
|---|---|---|---|
| 0 | | | 0 |
| 1 | 0 | 1 | 1 |
| 2 | | | 0 |
| 3 | 2 | 3 | 1 |
| 4 | | | 0 |
| 5 | | | 0 |
| 6 | | | 0 |
| 7 | | | 0 |



# 2.半导体存储器

## 高速缓冲存储器Cache

- Cache与主存的映像

全相连映像（映射）：空位随意放

直接映像：主存分区，对号入座

主存容量是缓存的整数倍

主存字块只能映像到Cache特定的块中

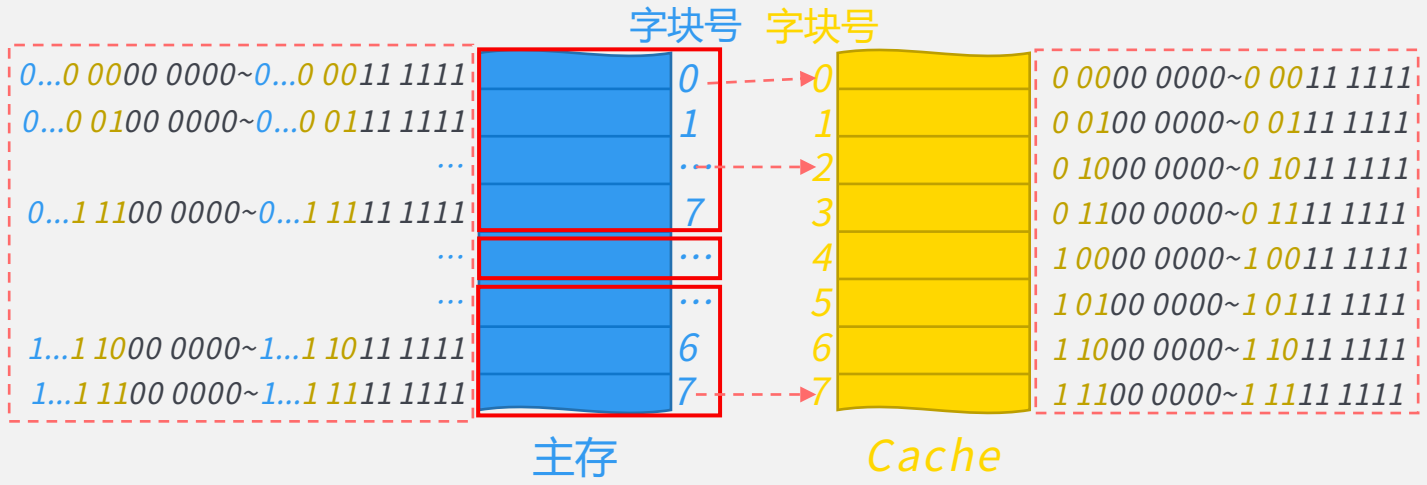
将主存空间按Cache大小分区，则每区块数=Cache块数

某区某块只能存入Cache的相同块号中

优点：数据访问时，只需检查区号是否相等即可

缺点：替换操作频繁，命中率比较低

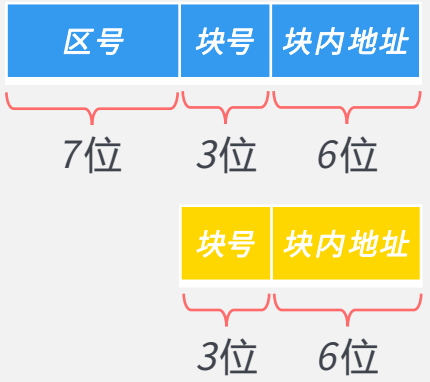
组相连映像




区数：128=64K/512

| | 主存块地址 | 主存区地址 | 有效位 |
|---|---|---|---|
| 0 | 0 | 0 | 1 |
| 1 | | | 0 |
| 2 | 2 | 0 | 1 |
| 3 | | | 0 |
| 4 | | | 0 |
| 5 | | | 0 |
| 6 | | | 0 |
| 7 | 7 | 127 | 1 |

将主存地址右移(>>>)6位得到x：将x右移3位得到区号g，将x左移13位得到块号b；以b为目录表索引获取条目，比对区号



# 2.半导体存储器

## 高速缓冲存储器Cache

- Cache与主存的映像

全相连映像（映射）：空位随意放

直接映像：主存分区，对号入座

组相连映像：先分区再分组，组内随意放

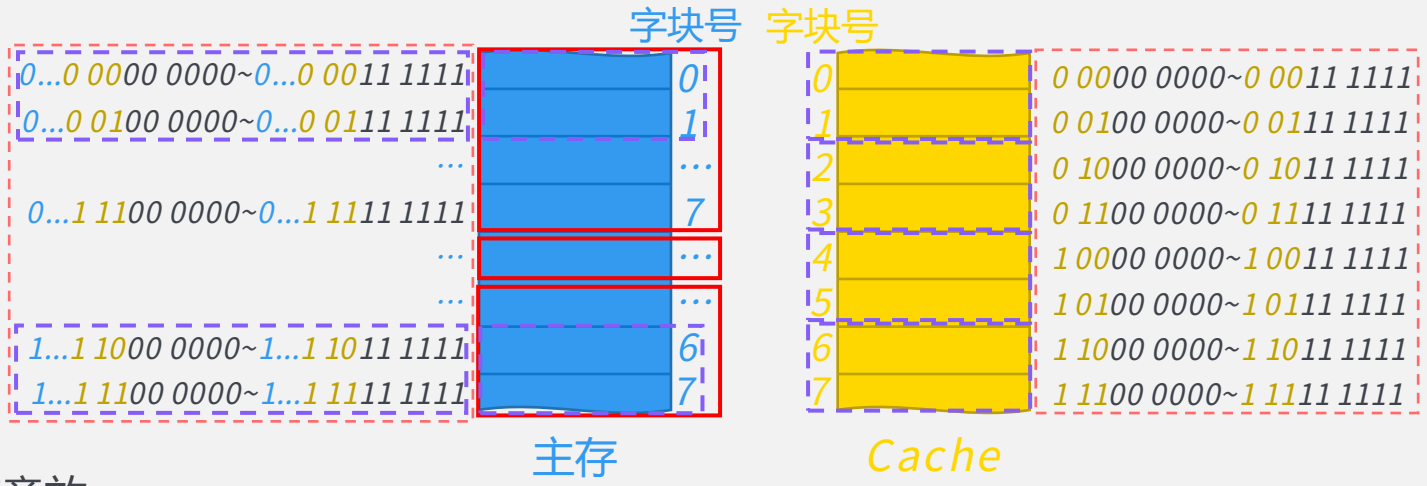
主存容量是缓存的整数倍

将主存空间按Cache大小分区，则每区块数=Cache块数

主存每区和Cache按同样大小划分成组，则每区组数=Cache组数

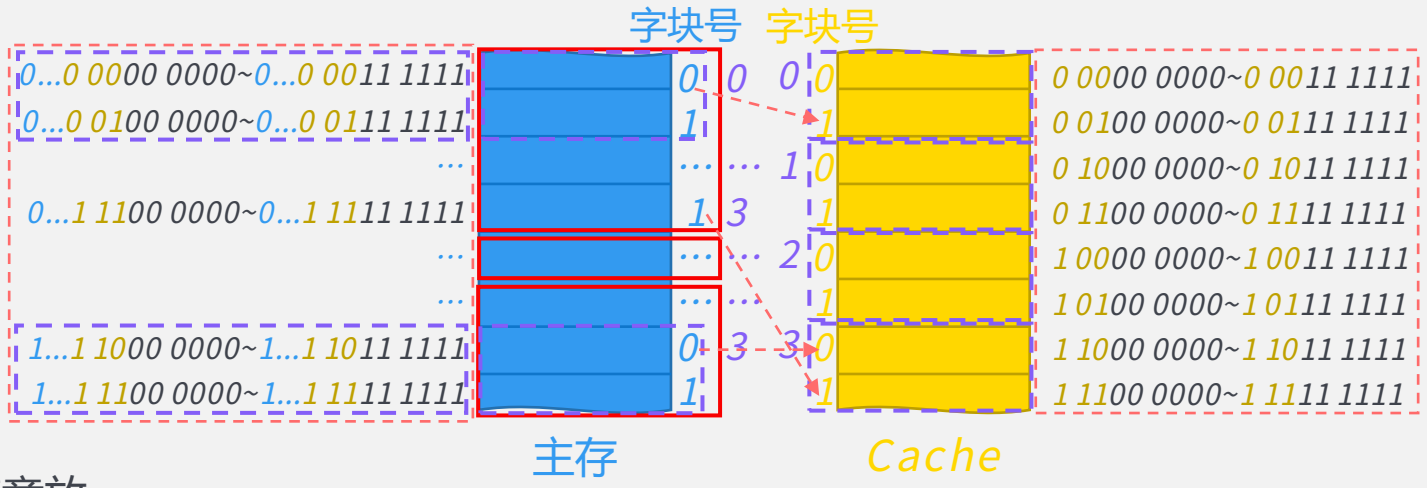
某区某组某块只能存入Cache的相同组号中，组内随意

组间直接映像，组内全相连映像



# 2.半导体存储器

## 高速缓冲存储器Cache




- Cache与主存的映像

  全相连映像（映射）：空位随意放

  直接映像：主存分区，对号入座

  组相连映像：先分区再分组，组内随意放

  主存容量是缓存的整数倍

  将主存空间按Cache大小分区，则每区块数=Cache块数

  主存每区和Cache按同样大小划分成组，则每区组数=Cache组数

  某区某组某块只能存入Cache的相同组号中，组内随意

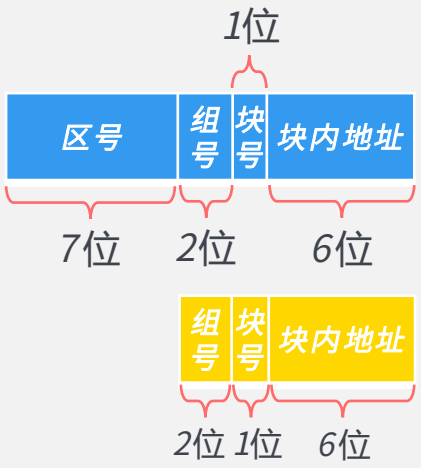
  组间直接映像，组内全相连映像

  优点：块冲突概率低，块利用率大幅提高，块失效率降低

  缺点：实现难度和造价要比直接映像方式高

区数：128=64K/512

| | 缓存块地址 | 主存块地址 | 主存组地址 | 主存区地址 | 有效位 |
|---|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 | 1 |
| 1 | | | | | 0 |
| 2 | 1 | 1 | 3 | 0 | 1 |
| 3 | | | | | 0 |
| 4 | | | | | 0 |
| 5 | | | | | 0 |
| 6 | | | | | 0 |
| 7 | 0 | 0 | 3 | 127 | 1 |



# 2.半导体存储器

## 高速缓冲存储器Cache

- Cache的组织结构

  Cache存储体

  地址转换机构/部件

  **替换机构/部件**

  随机算法（Rand）：

  随机确定被替换的块。实现简单，未遵循局部性原理

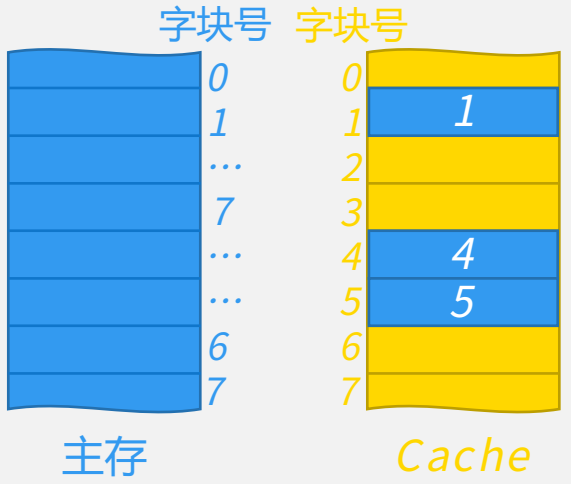
  先进先出策略（FIFO，First In First Out）：

  始终选择最早调入Cache的块替换。实现简单，未遵循局部性原理

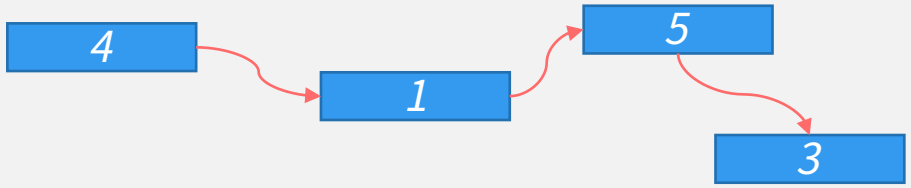
  最近最少使用（LRU，Least Recently Used）

  随时记录每个块最近一次使用次数（times），命中清零，未命中加1

  淘汰次数最大的块




| | times |
|---|---|
| 0 | 0 |
| 1 | 2 |
| 2 | 0 |
| 3 | 3 |
| 4 | 7 |
| 5 | 14 |
| 6 | 0 |
| 7 | 12 |



马士兵教育

www.mashibing.com

扫码加马老师微信