大实验报告要求

- 实验目标概述
- 主要模块设计
  - 硬件：Datapath、Controller、Memory、I/O
  - 软件：对系统软件的修改、应用软件
- 主要模块实现
- 实验成果展示
- 实验心得和体会
- 提交材料
  - 实验报告
  - 视频材料
  - 完整源代码及相关说明

清華大學
Tsinghua University

第三单元 第二讲

静态存储器及高速缓冲存储器

刘卫东

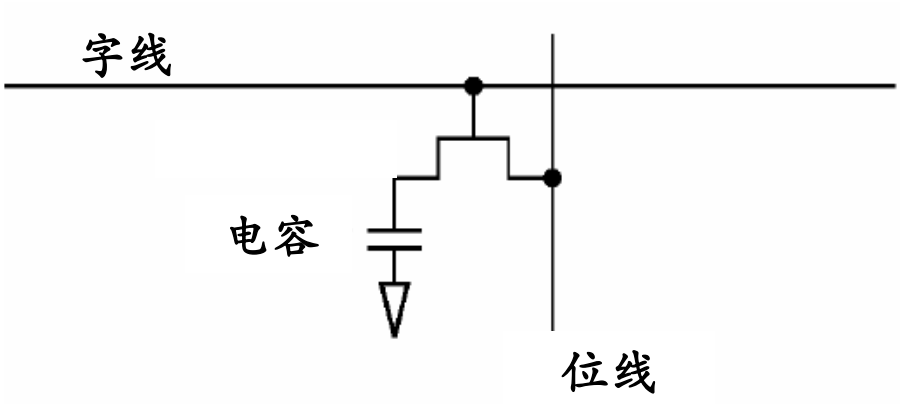
计算机科学与技术系

内容提要

- 动态存储器存储原理
- 静态存储器存储原理
- 高速缓冲存储器（Cache）概述
- Cache的地址映射
  - 直接映射
  - 全相联
  - 多路组相连

动态存储器存储原理

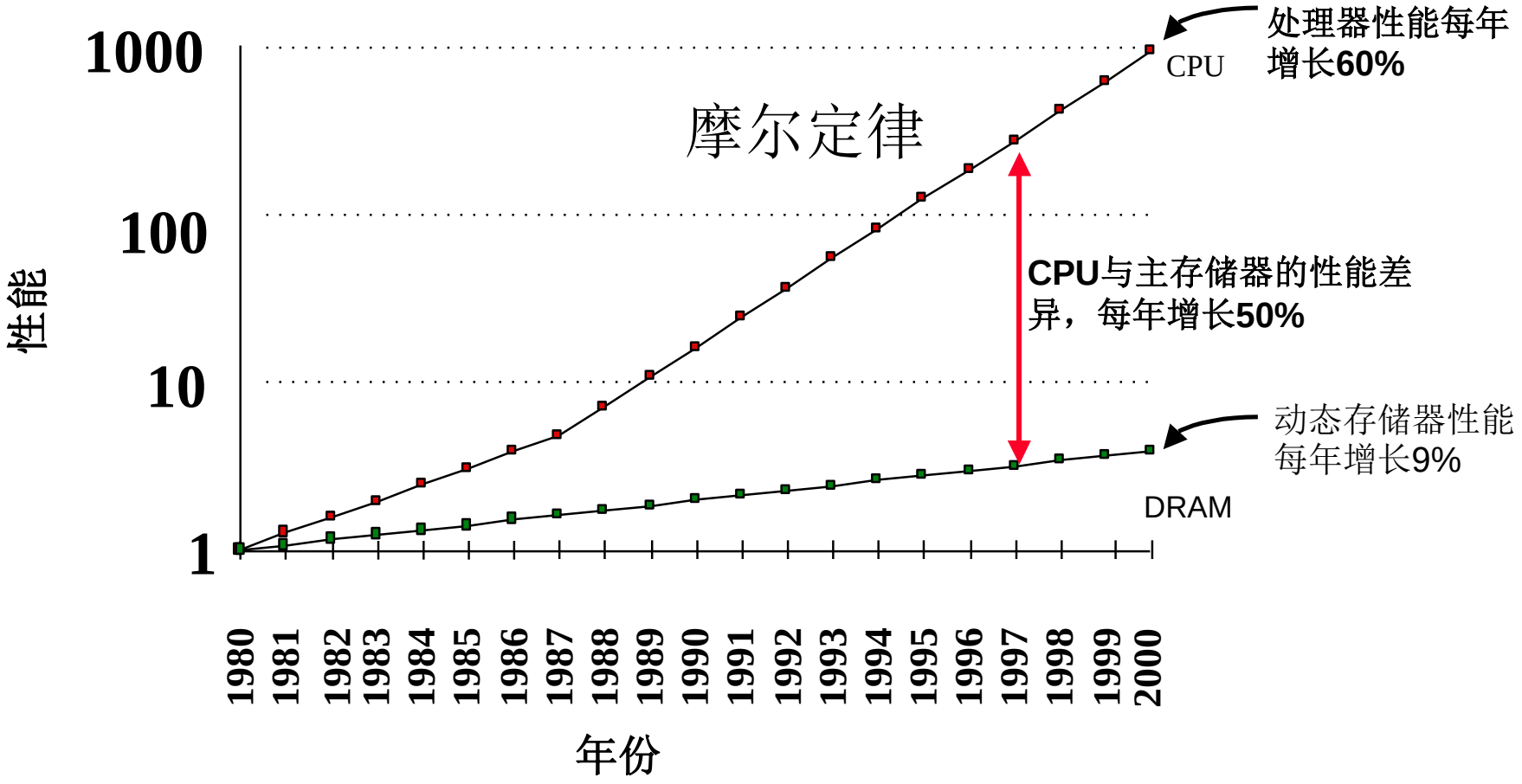
- 写
  - 往位线上送数据
  - 选择字线
- 读
  - 将位线上置高电平
  - 选中字线
  - 感知电容是否放电并放大
  - 写回
- 刷新
  - 定期的批量读操作



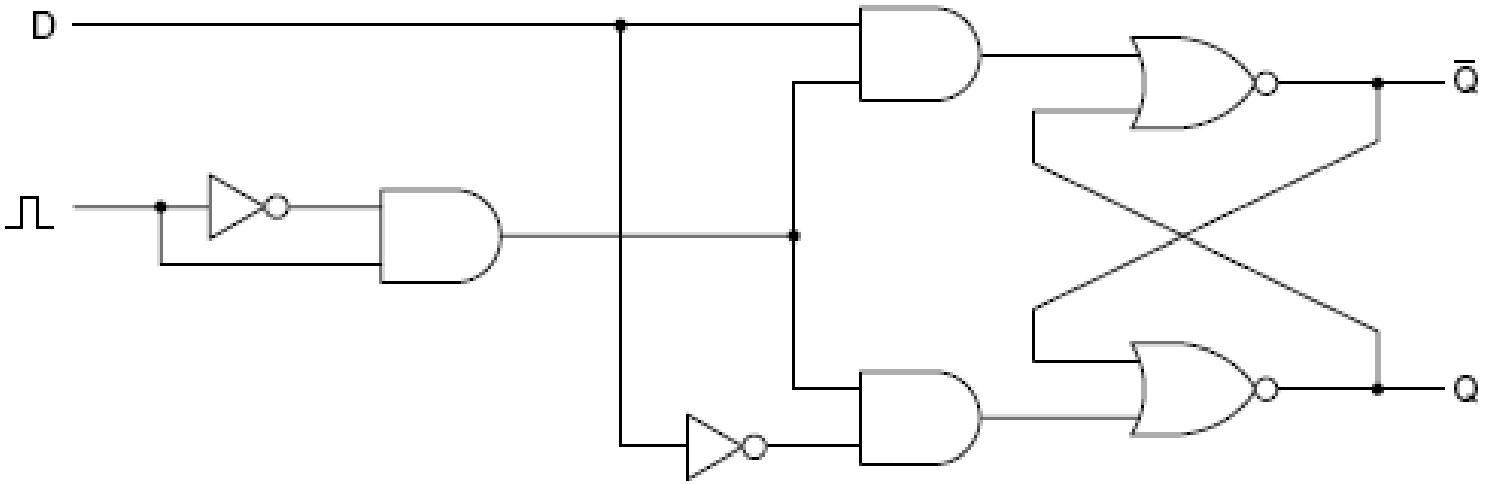
动态存储器特点

- 存储容量高
  - 单位存储单元面积小
- 访问速度慢
  - 电容充放电
  - 刷新
- 能耗低
- 成本低

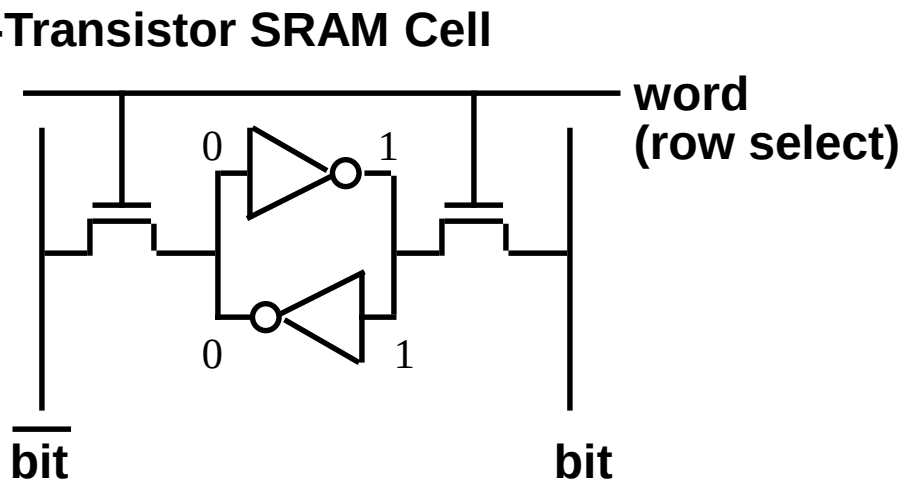
Moore定律


1000
100
10
1
性能
摩尔定律
处理器性能每年
增长60%
CPU
CPU与主存储器的性能差
异，每年增长50%
动态存储器性能
每年增长9%
DRAM
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
年份

D触发器

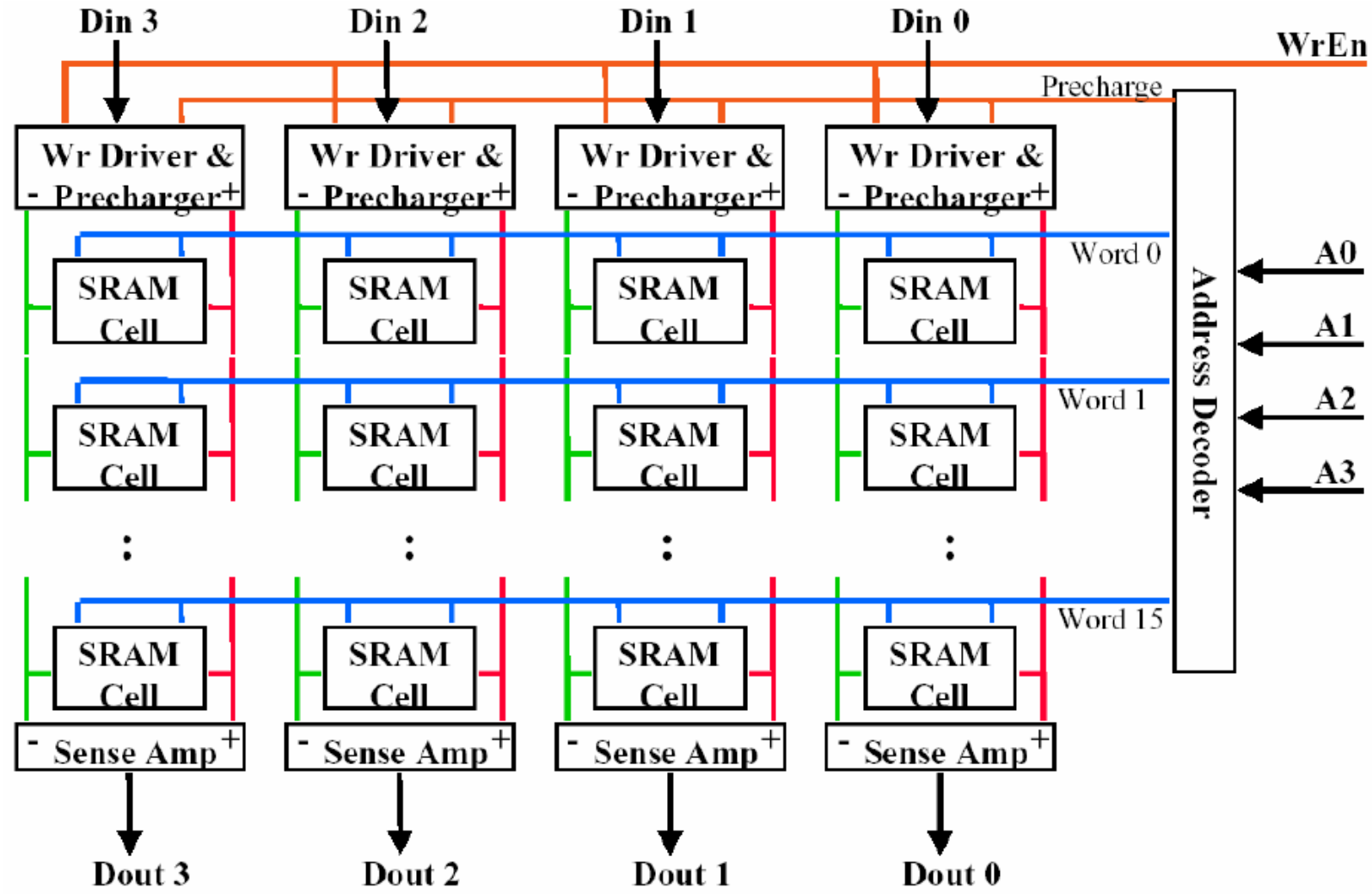


静态存储器存储单元

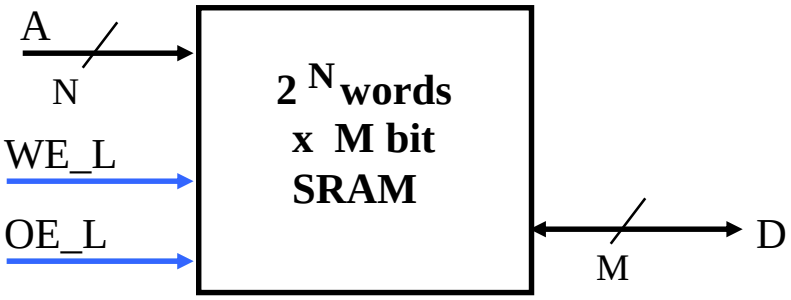


- 写1:
  1. 在位线上设置使(bit=1, $\overline{bit}$=0)
  2. 使字线选通
- 读:
  1. 使 bit 和 $\overline{bit}$ 都充为高电平Vdd
  2. 使字线选通
  3. 根据触发器的状态,将使其中一条位线电平为低
  4. 放大器感知 bit 和 $\overline{bit}$的变化,读出存储的值

静态存储器典型组织方式

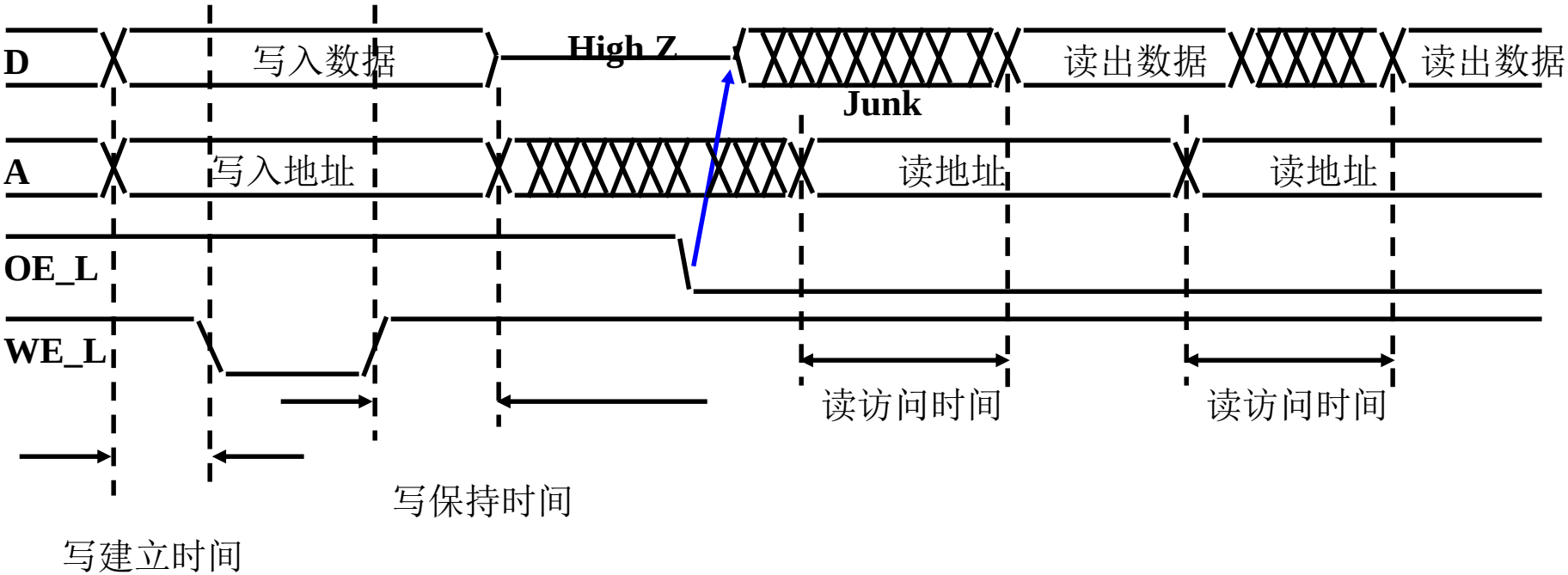


SRAM典型时序



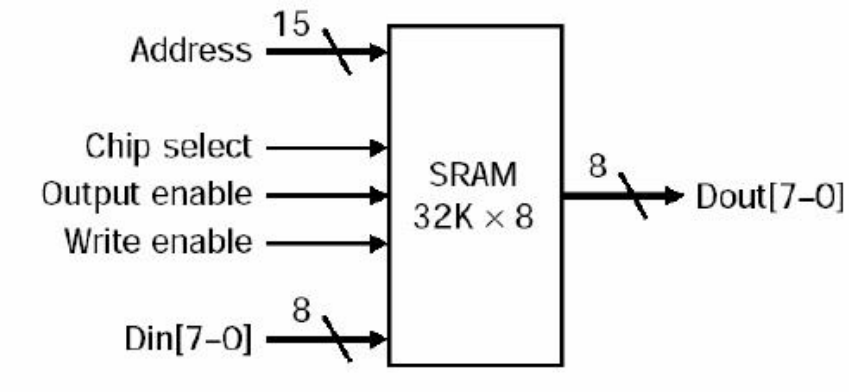
写时序:

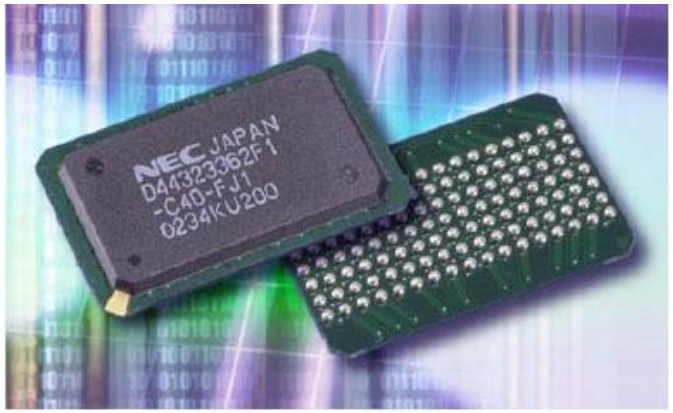
读时序:



静态存储器

- 速度快
- 存储密度低，单位面积存储容量小
- 数据入/出共用管脚
- 能耗高
- 价格高





与动态存储器比较

清華大學 Tsinghua University

	SRAM	DRAM
存储信息	触发器	电容
破坏性读出	非	是
需要刷新	不要	需要
送行列地址	同时送	分两次送
访问速度	快	慢
集成度	低	高
发热量	大	小
存储成本	高	低

程序运行的局部性原理

清華大學 Tsinghua University

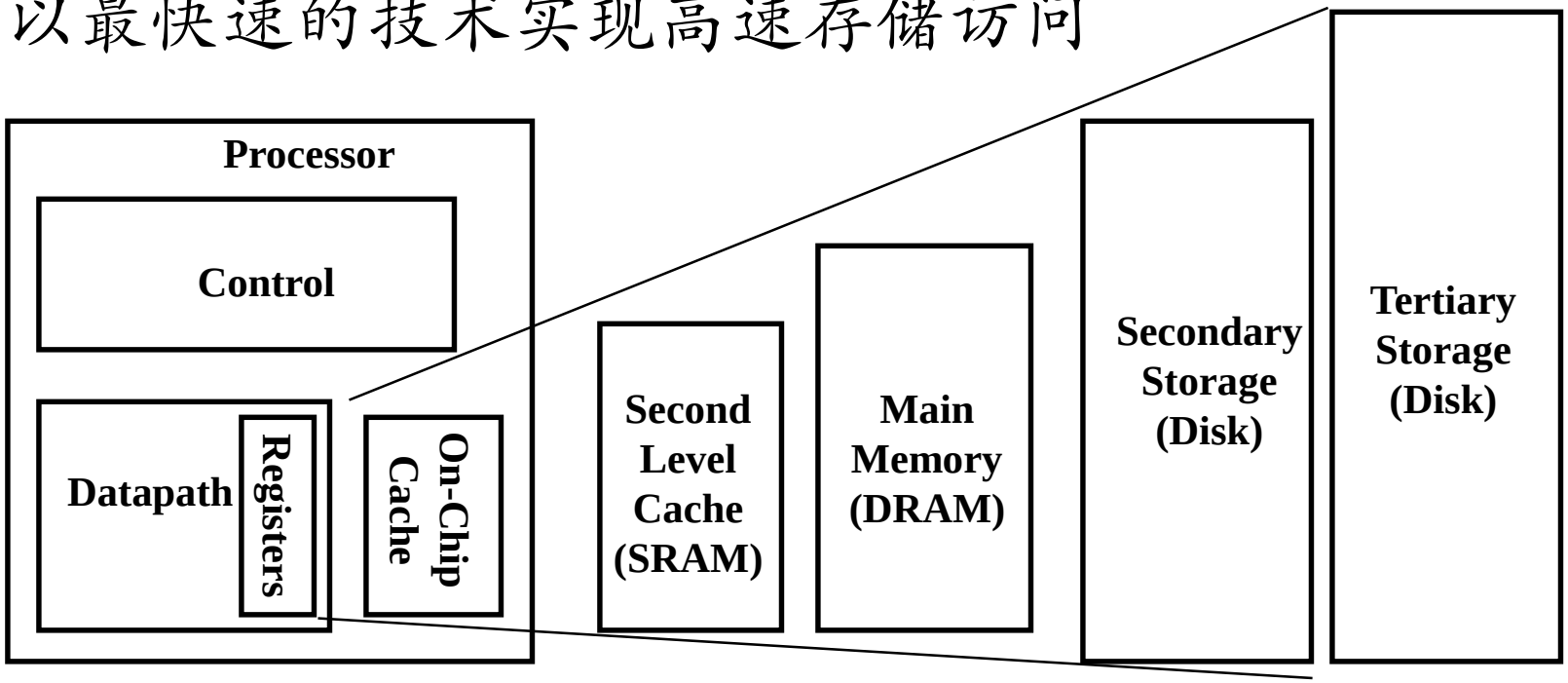
程序运行时的局部性原理表现在：

在一小段时间内，	最近被访问过的程序和数据很可能再次被访问
在空间上	这些被访问的程序和数据往往集中在一小片存储区
在访问顺序上，	指令顺序执行比转移执行的可能性大(大约 5:1)

合理地把程序和数据分配在不同存储介质中

层次存储器系统

- 利用程序的局部性原理:
  - 以最低廉的价格提供尽可能大的存储空间
  - 以最快速的技术实现高速存储访问



程序的局部性原理

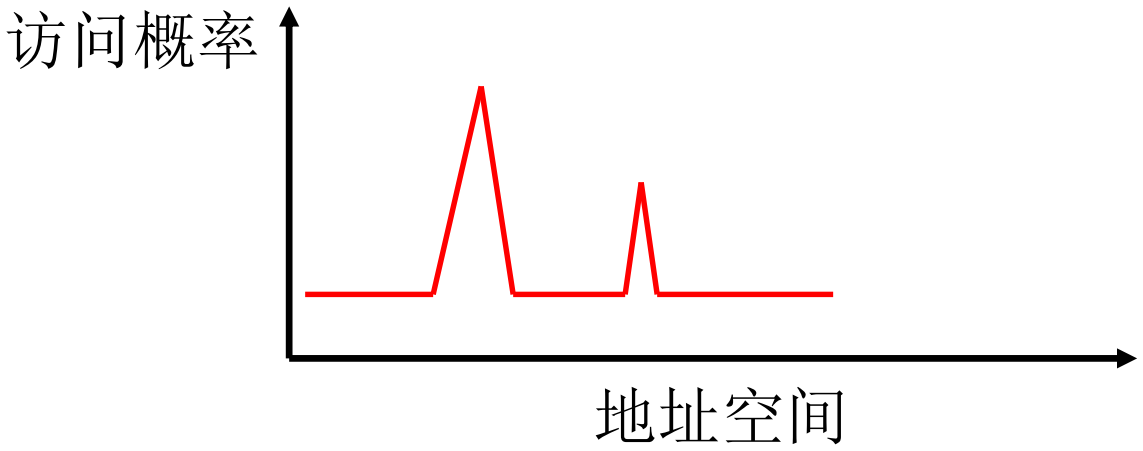
```
for (i=0; i<1000; i++) {
    for (j=0; j<1000; j++ ) {
      a[i] = b[i] + c[i];
       }
   }
If err { …… }
 else for (i=0; i<1000; i++) {
    for (j=0; j<1000; j++ ) {
      e[i] = d[i] * a[i];
       }
   }
……
```

数据流访问的局部性

指令访问的局部性

不同的程序段可能访问不同的内存空间

程序的局部性原理



- 程序在一定的时间段内通常只访问较小的地址空间
- 两种局部性：
  - 时间局部性
  - 空间局部性

层次存储器系统

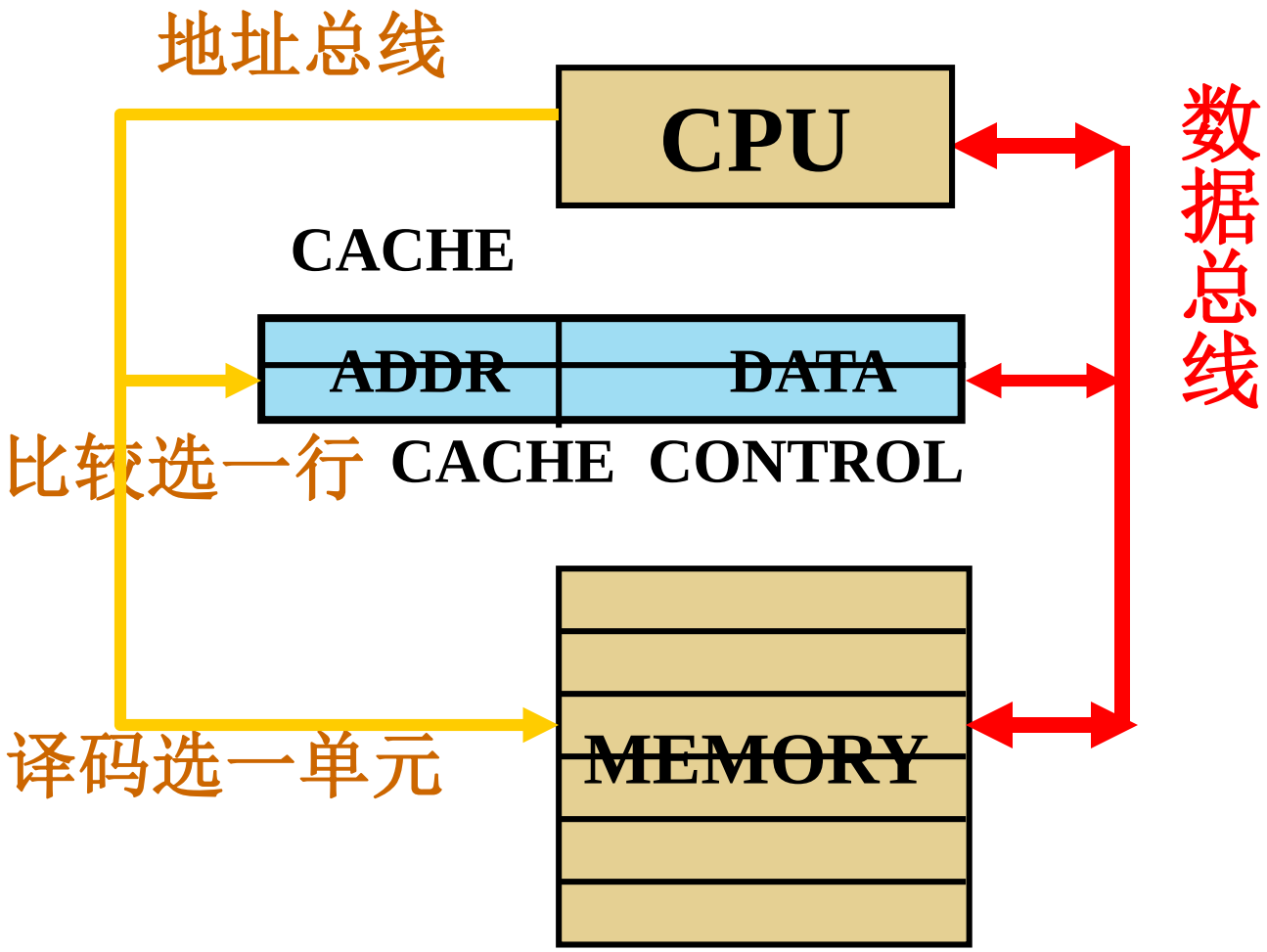
清華大學 Tsinghua University

- 使用高速缓冲存储器Cache来提高CPU对存储器的平均访问速度。
- 时间局部性：最近被访问的信息很可能还要被访问。
  - 将最近被访问的信息项装入到Cache中。
- 空间局部性：最近被访问的信息临近的信息也可能被访问。
  - 将最近被访问的信息项临近的信息一起装入到Cache中。

高速缓冲存储器Cache

- 定义
  - 设置于主存和CPU之间的存储器，用高速的静态存储器实现，缓存了CPU频繁访问的信息。
- 特点
  - 高速：与CPU的运行速度基本匹配
  - 透明：完全硬件管理，对程序员透明

CACHE的基本运行原理



读过程为例

要解决的问题

清華大學 Tsinghua University

1. 地址和Cache行之间的映射关系：

   如何根据主存地址得到Cache中的数据？

2. 数据之间一致性：

   Cache中的内容是否已经是主存对应地址的内容？

3. 数据交换的粒度：

   Cache中的内容与主存内容以多大的粒度交换？

4. Cache内容装入和替换策略

   如何提高Cache的命中率？

Cache参数

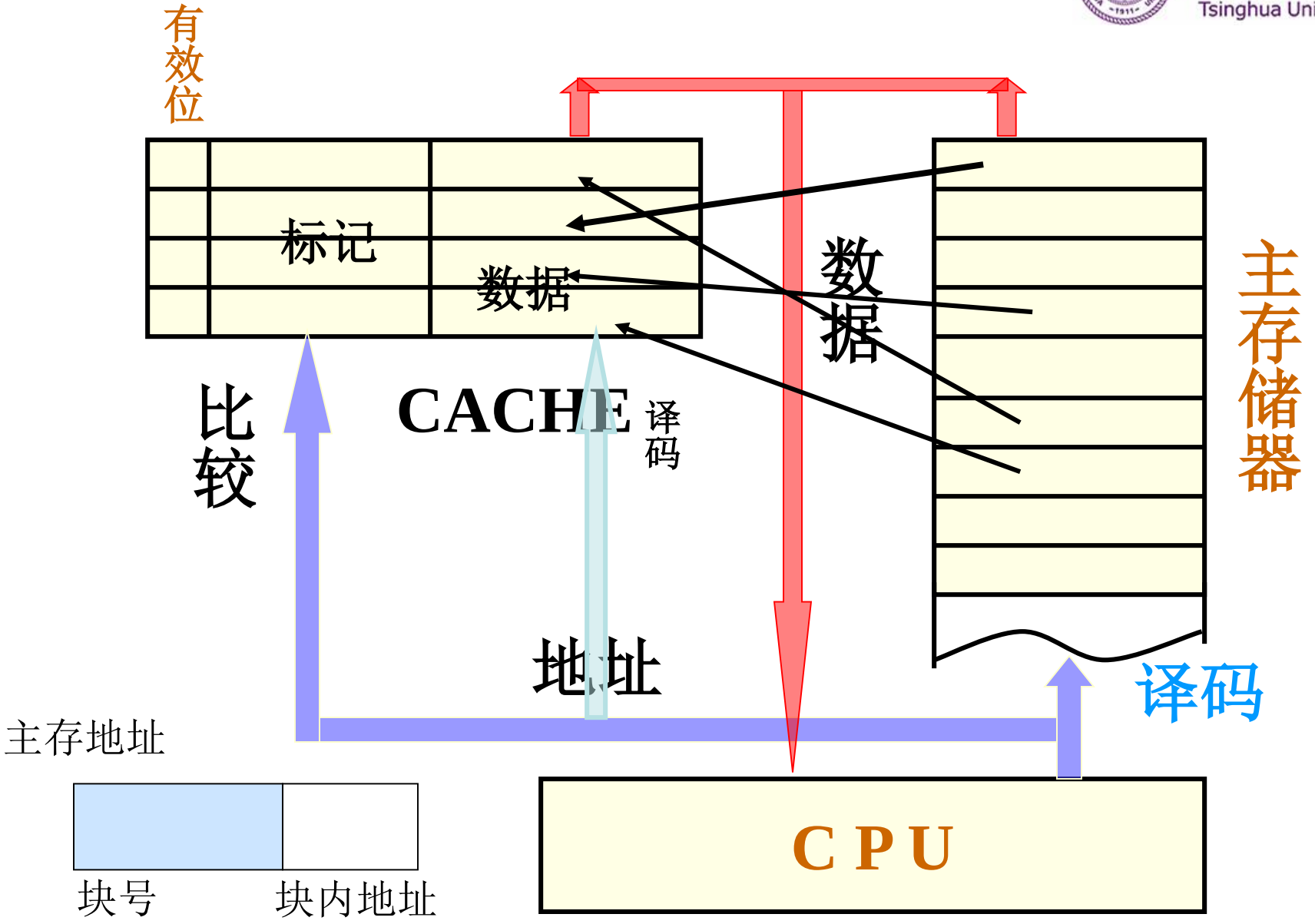
清華大學 Tsinghua University

- 块（Line）：数据交换的最小单位
- 命中（Hit）：在较高层次中发现要访问的内容
  - 命中率（Hit Rate）：命中次数/访问次数
  - 命中时间：访问在较高层次中数据的时间
- 失效（Miss）：需要在较低层次中访问块
  - 失效率（Miss Rate）：1-命中率
  - 失效损失（Miss Penalty）：替换较高层次数据块的时间+将该块交付给处理器的时间
- 命中时间<<失效损失
- 平均访问时间=HR*命中时间+(1-HR)*失效损失

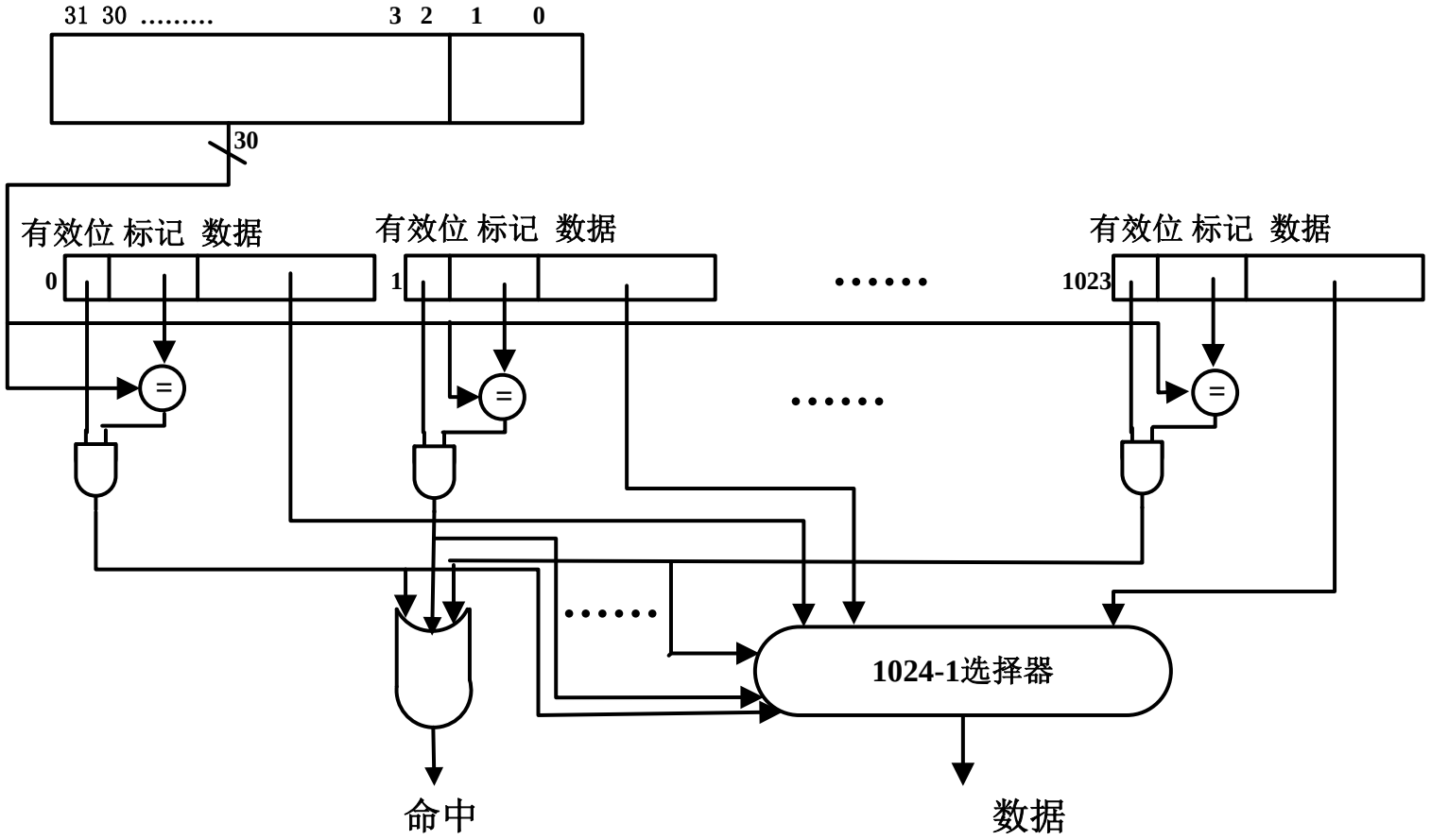
参数典型数值

- 块大小：4~128 Bytes
- 命中时间：1~4周期
- 失效损失：
  - 访问时间：6~10个周期
  - 传输时间：2~22个周期
- 命中率：80%~99%
- Cache容量：1KB~256KB

全相联方式



全相联映射硬件实现举例



主存：4GB，Cache：4KB，块大小：4B，全相联

标记位数？

全相联方式的地址映射关系

清華大學 Tsinghua University

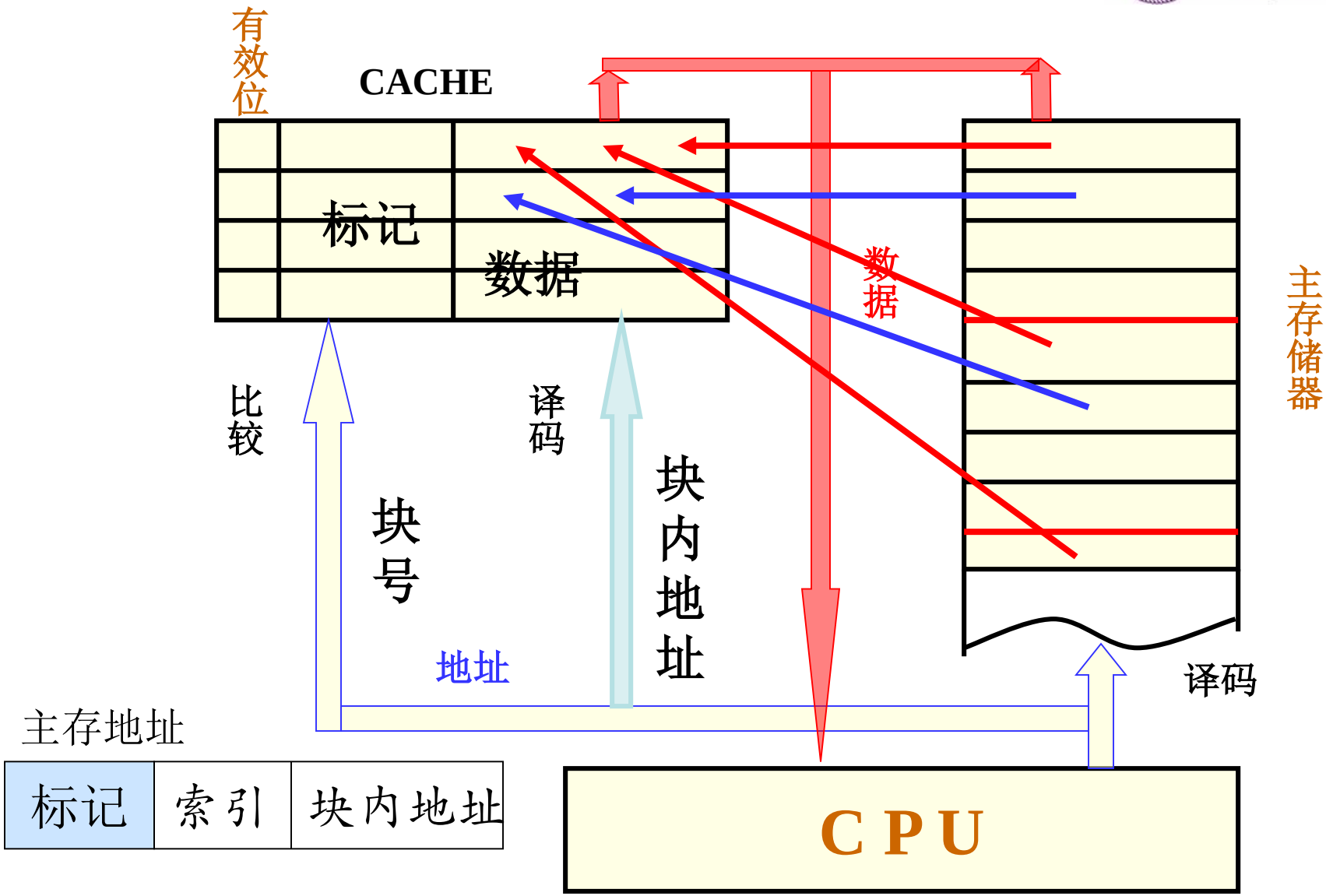
特点

1. 主存的字块可以和Cache的任何字块对应，利用率高，方式灵活。

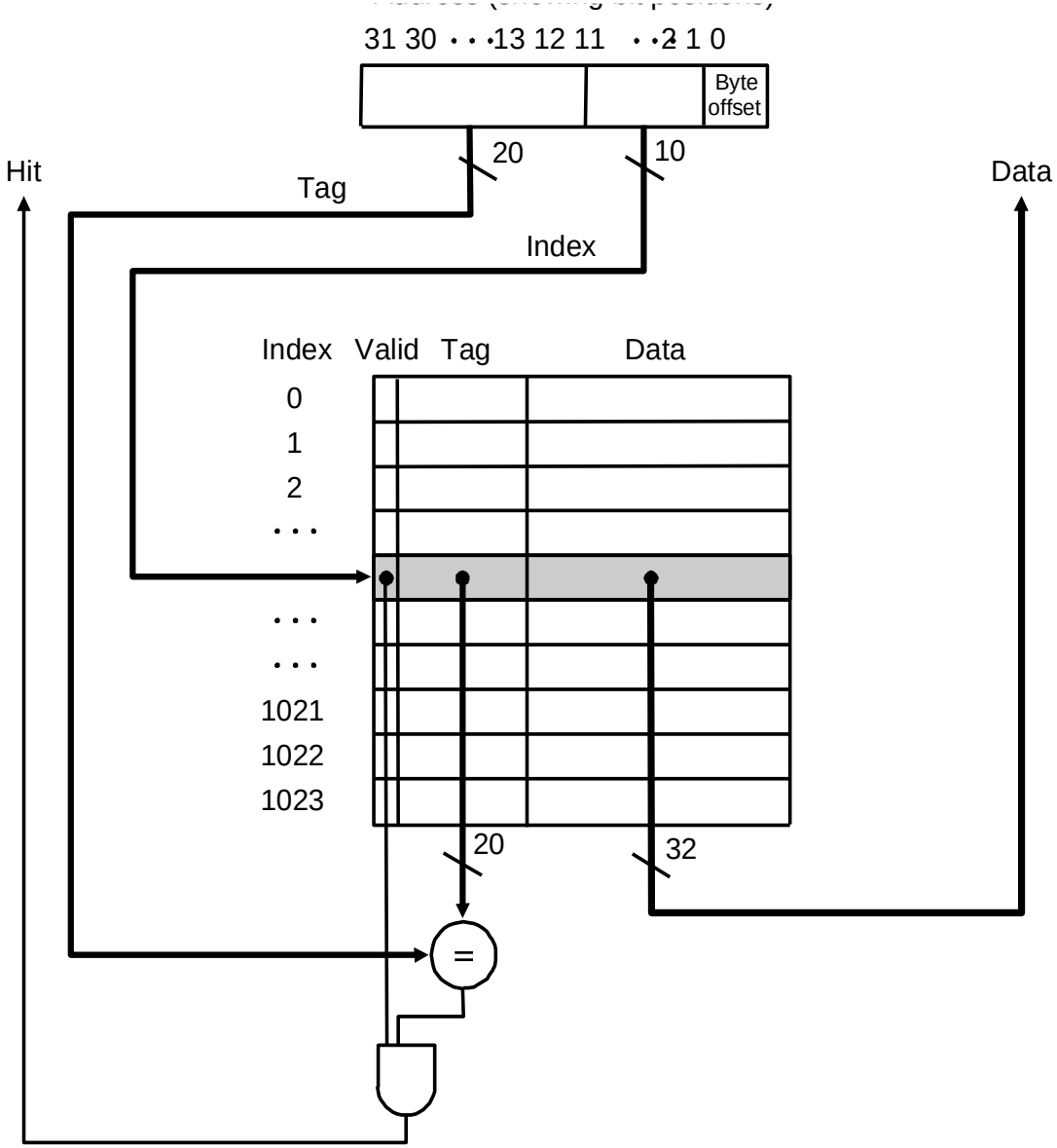
2. 标志位较长，比较电路的成本太高。如：n位的主存地址，块内地址为b位，Cache有m块，则需要有m个比较电路，标志位需n-b位。

使用成本太高

直接映射方式



直接映射 Cache:硬件实现



主存：4GB

Cache：4KB

块大小：4B

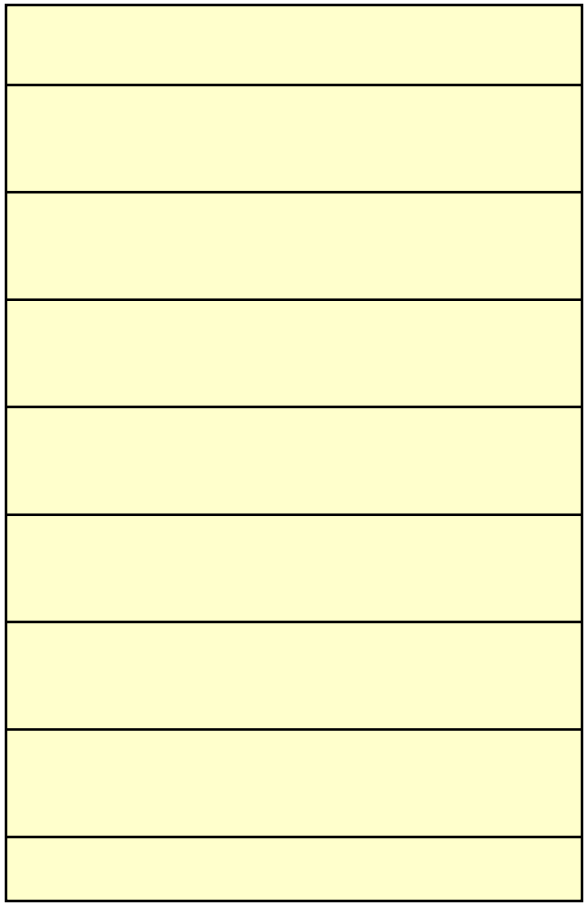
直接映射

标记位数?

索引位数?

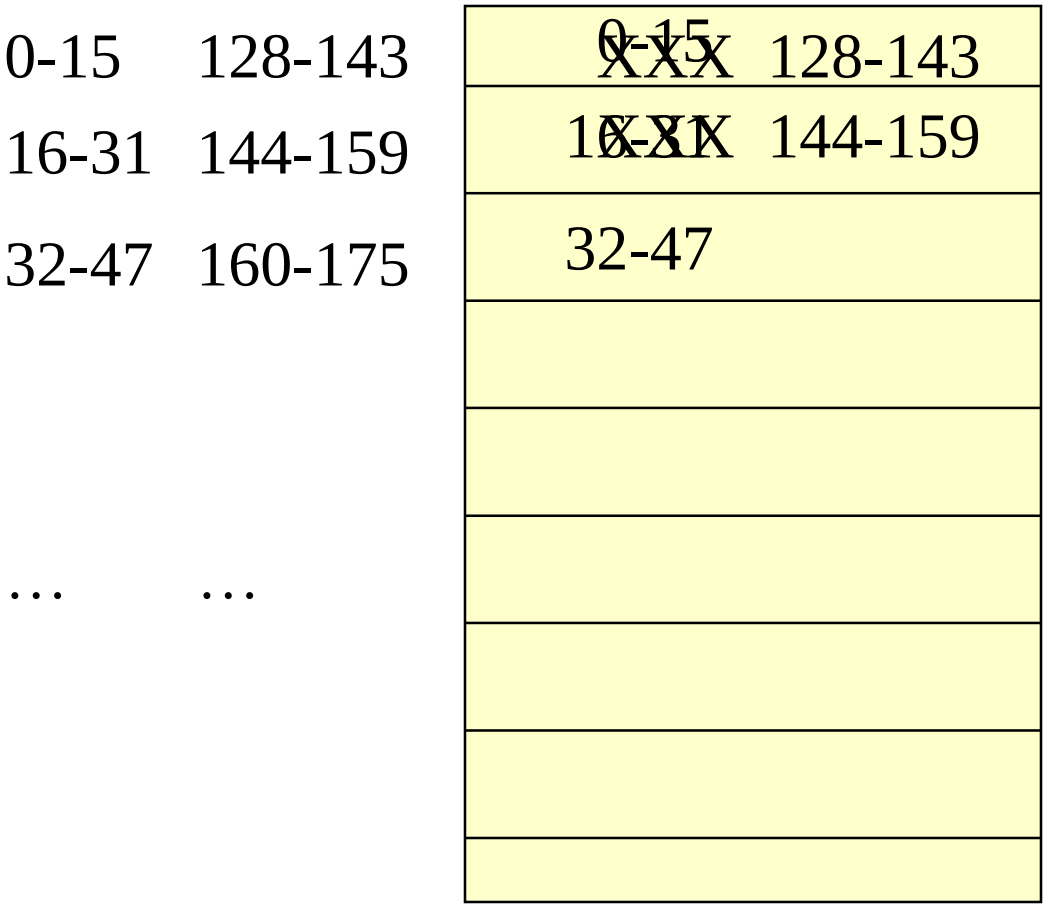
Cache 举例

- 8 块 cache
- 每块16 字节
- “直接映射”：内存中的每个单元在Cache中只会有一个唯一的位置和它对应。



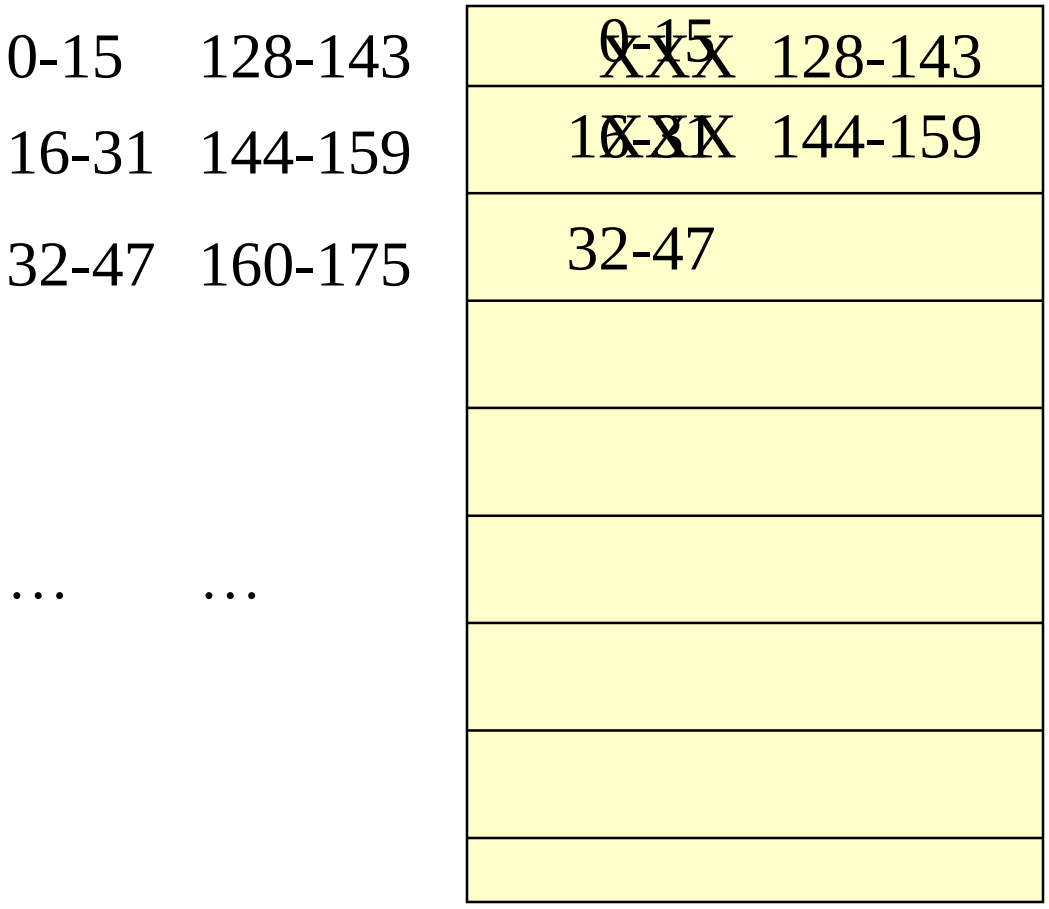
0-15	128-143
16-31	144-159
32-47	160-175
…	…

直接映射Cache 举例



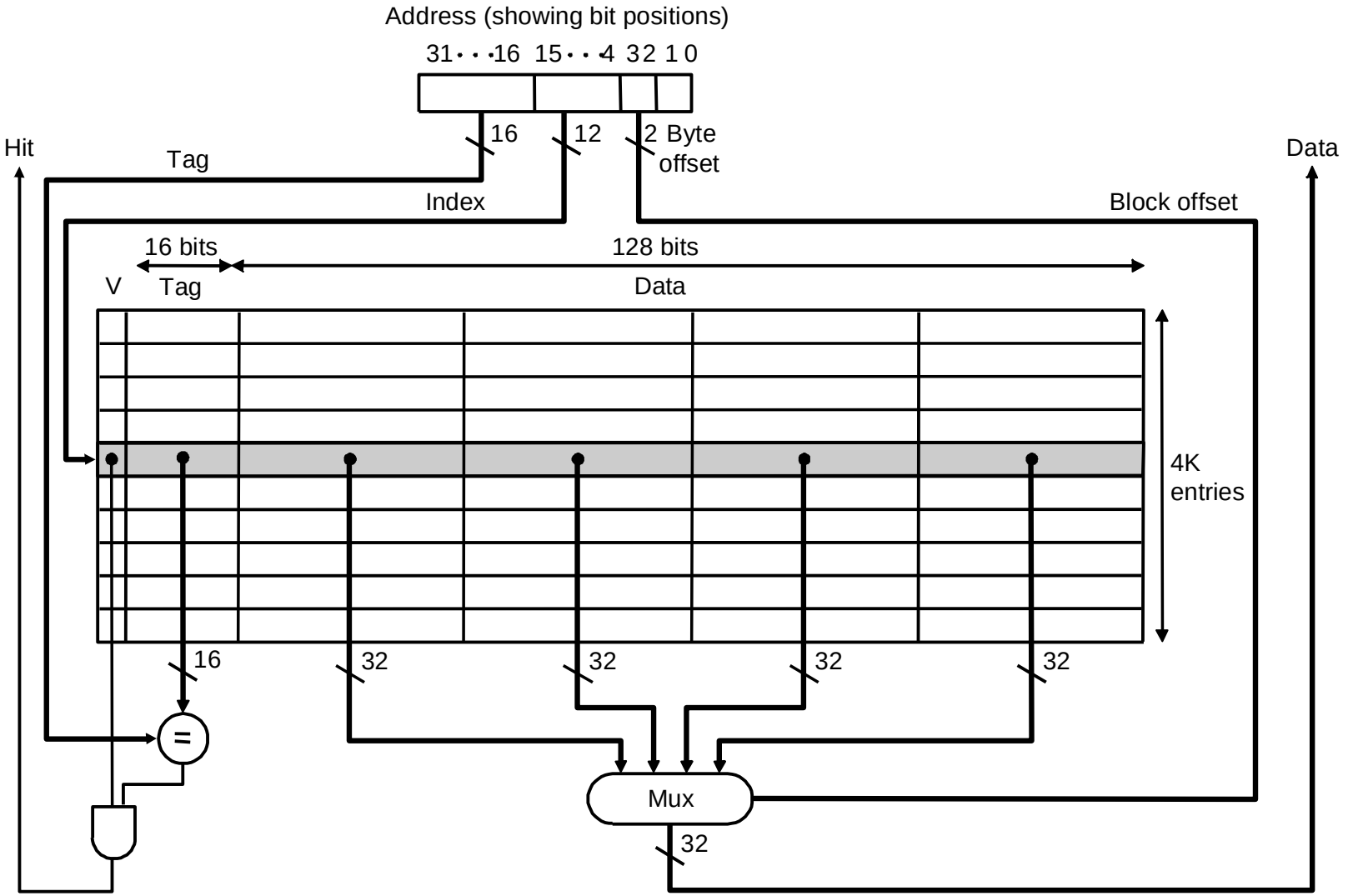
- 假定有如下访问操作:
  - Read location 0
  - Read location 16
  - Read location 32
  - Read location 4
  - Read location 8
  - Read location 36
  - Read location 32
  - Read location 128
  - Read location 148
- cache中命中和缺失各有多少次?

Cache 举例：续



- Cache中命中和缺失次数?
  - Read location 0: Miss
  - Read location 16: Miss
  - Read location 32: Miss
  - Read location 4: Hit
  - Read location 8: Hit
  - Read location 36: Hit
  - Read location 32: Hit
  - Read location 128: Miss
  - Read location 148: Miss
- 命中率 = 4/9 = 45%
- 注意:失效的原因
  - 启动失效
  - 冲突失效

直接映射 Cache: 硬件实现



直接映射方式的地址映射

清華大學 Tsinghua University

特点

1. 主存的字块只可以和固定的Cache字块对应，方式直接，利用率低。

2. 标志位较短，比较电路的成本低。如果主存空间有2^m块，Cache中字块有2^c块，则标志位只要有m-c位。且仅需要比较一次。

利用率低，命中率低，效率较低

小结

- 静态存储器
  - 存储速度快
  - 集成度低，容量小
  - 成本高
- Cache
  - 在CPU和主存储器之间设置
  - 提高访问存储器的速度
  - Cache和主存地址映射方式
    - 全相联
    - 直接映射

阅读与思考

- 阅读
  - 教材5.2节
  - 参考书
- 思考
  - 设置高速缓冲存储器的目的?如何提高性能?
  - 对高速缓冲存储器的地址映射方式进行比较.