# 第七章 多处理机和多计算机

## 7.1 多处理机概念

**1.多处理机的概念**

**两个或两个**以上处理机（包括PU和CU）通过高速互连网络连接起来，在统一的操作系统管理下，实现指令以（任务级、作业级）并行。多处理机系统**属于MIMD**计算机。

**2.多处理机系统的分类方法**

（1）按照处理机之间的连接程度：

耦合度：反映多处理机之间物理连接的紧密程度和交叉作用的能力的强弱

1. 紧耦合多处理机：
   1. **直接耦合系统**
   2. 通过公共硬件 如存储器 I/O 系统等 通信
   3. 典型： SMP 多核处理器等
   4. 一般将紧耦合多处理机称为 **多处理机**
2. 松散耦合多处理机:
   1. **间接耦合系统**
   2. 通过通道 、 通信线路或网络 、 消息传递系统通信
   3. 典型：机群 计算机网络等
   4. 一般将松耦合多处理机称为 **多计算机**

（2）按照是否共享主存储器：共享存储器和分布存储器多处理机

（3）按照处理机是否相同分为三类：

1. 同构型：基于资源重复
2. 异构型：基于时间重叠
3. 分布式：基于资源共享



**3.多处理机系统的特点**

**（1）结构灵活: 满足多种并行计算的不同需求**

并行处理机：专用，PE很多（几k个），固定有限的通信

多处理机：通用，几十个，高速灵活

**（2）程序并行性: 实现作业 、 任务之间的并行**

并行处理机的并行性存在于指令内部，识别容易

多处理机的并行性存在于指令外部，在多个任务之间，识别较难

**（3）并行任务派生:** **可并行执行任务的识别 、 派生与分配 。**

并行处理机把同种操作集中在一起，由指令直接启动各PE同时工作

多处理机用专门指令来表示并发关系，一个任务开始执行时能够派生除与它并行执行的另一些任务。若任务数多于处理机数，多余任务进入排队器等待

**（4）进程同步 : 解决数据相关和控制依赖问题 。**

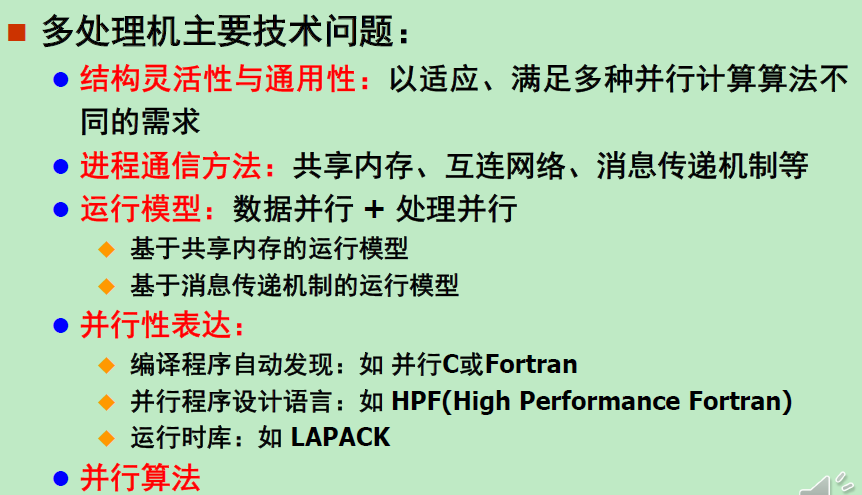
并行处理机只有一个CU，自然是同步的

多处理机执行不同的指令，要采取特殊的同步措施来保持程序要求的正确顺序

**（5）资源分配和进程调度: 动态分配资源和调度任务 以获得更好性能和更高效率**

并行处理机的PE是固定的，采用屏蔽手段改变实际参与操作的PE数量

多处理机执行并发任务，需要处理机的数目不固定，各个处理机进出的时机也不同，所需共享资源的品种和数量随时变化。提出资源分配和进程调度问题对整个系统的效率有很大影响。



## 7.2多处理机结构

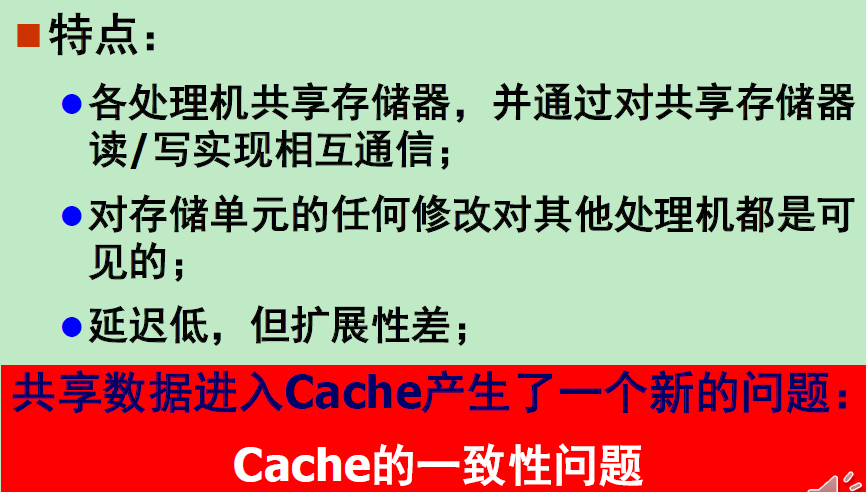
从**存储器的分布和使用**上看分为：**共享存储器结构**和**分布式存储器结构**

### 7.2.1 共享存储器结构

**（1）共享存储器结构：**各处理机通过互连网络 **共享存储器和 I/O 设备** ，并通

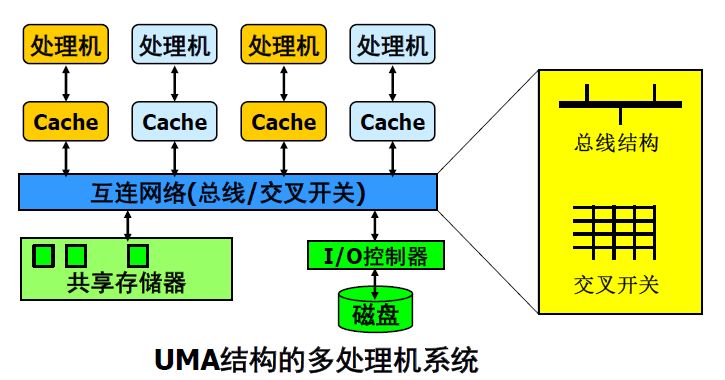
过共享存储器相互联系.

* 特点：



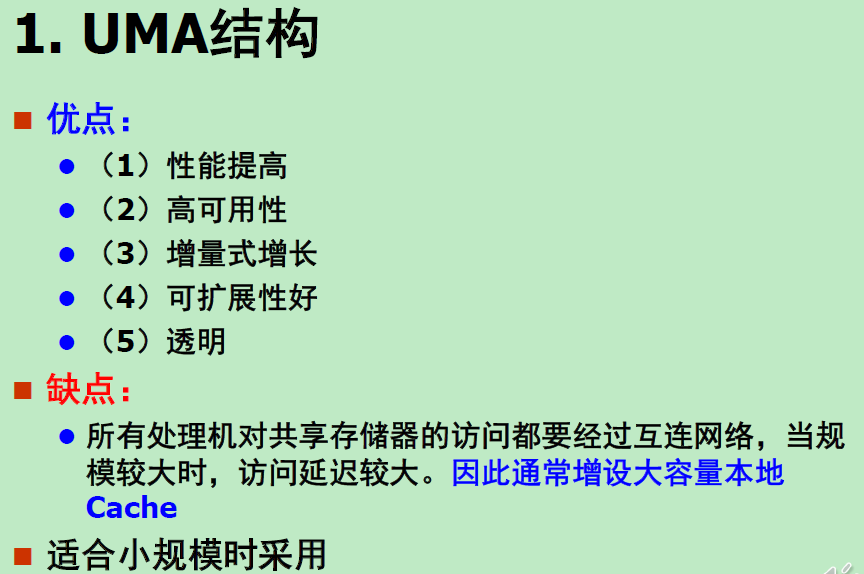
* 分类：
  + **1. 物理共享：**UMA（Uniform Memory Access，**均衡存储器访问结构**/**集中式共享存储器**结构）
    - **特点：各处理机对存储器的访问时间、访问功能相同**
    - 这种结构的处理机称为 对称多处理机

UMA结构：



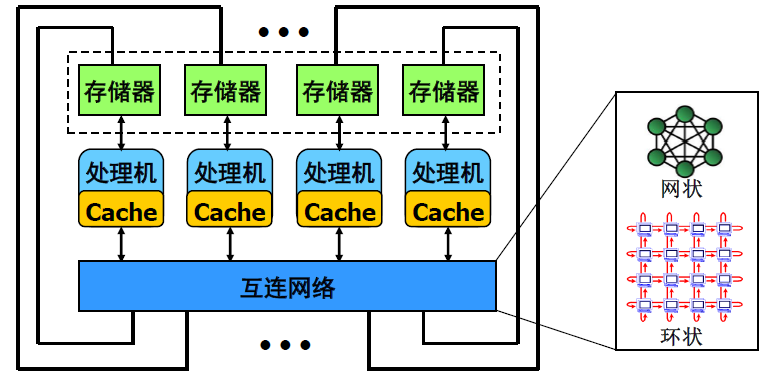
互连网络可以是总线、交叉开关或多级交换网络。

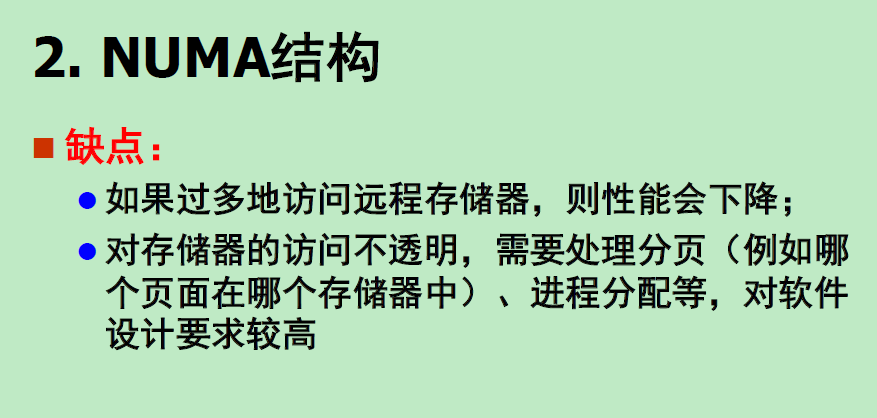
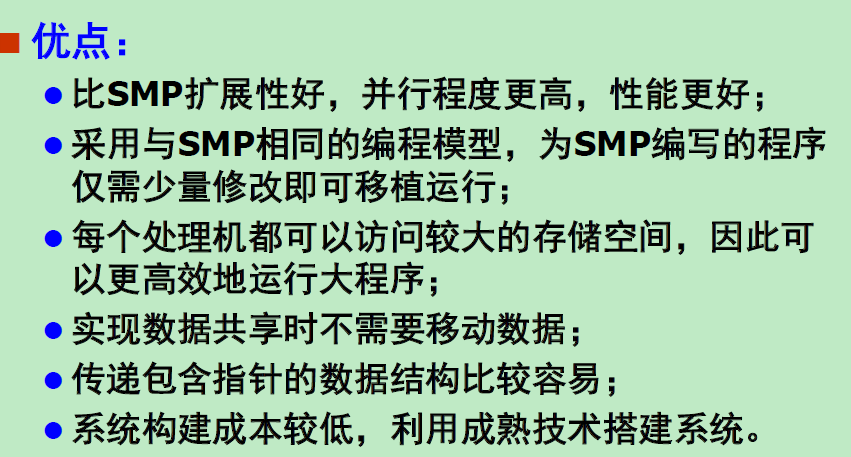
大多数的对称多处理机采用总线连接

。

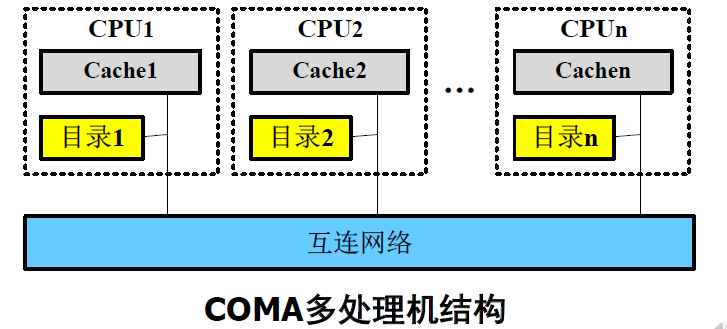
* + **2. 逻辑共享：**NUMA（Non UMA，**非均衡存储器访问结构/分布式共享存储器结构**），ccNUMA（Cache-coherent NUMA），COMA（Cache Only Memory Architecture）
    - **特点：**
      * **分布于各个处理机的存储器被 统一编址， 可由所有处理机共享；**
      * **根据存储器位置的不同，各处理机对存储器的 访问时间不相等。 处理机访问本地存储器的速度较快，通过互连网络访问其他处理机上的远地存储器相对较慢。**

NUMA结构：





COMA结构：

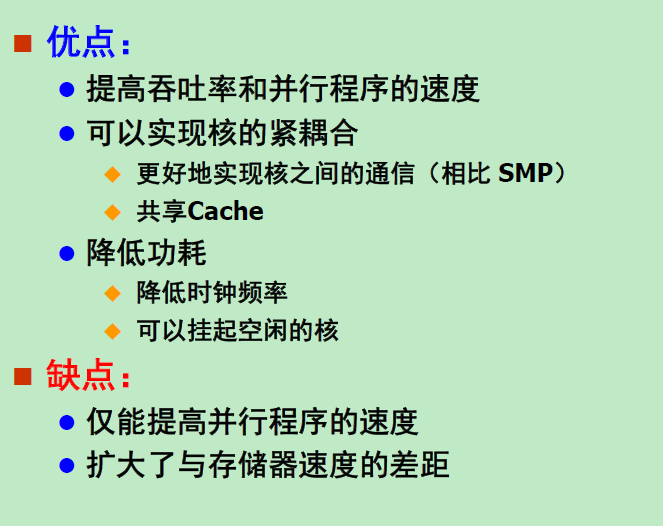


## 7.2.2 分布式存储器结构

**（2）分布式存储器结构（NoRMA，No-Remote Memory Access）：非远程存储访问模型**

MPP（大规模并行处理机）、COW（Cluster of Workstations）/NOW（Network of Workstations）

## 7.3 多核处理机

* 多核处理器：
  + 一枚处理器中集成了两个或多个独立处理单元（称为核）的处理器
  + 每个核由一个独立处理器的所有组件所组成，可以 独立 运行程序指令 （多指令 可以访问存储器的不同部分 （多数据）。
* 
* 多核处理器结构：

同构多核，异构多核

1. **多核处理器结构设计主要考虑的因素：**

**（1）同构还是异构**：

CMP的构成分成同构和异构两类，

* 同构CMP在一块芯片中集成多个相同的处理器核，同一个任务可以分配给任意一个核处理，简化了任务分配。
* 异构CMP中包含不同结构的处理器核，有事务处理型和计算型的。用不同类型的处理器核处理不同的任务，是异构体系结构处理器的优势所在。

**（2）核的数量：**

设置多少个核

**（3）存储器或者Cache的设置及访问：**

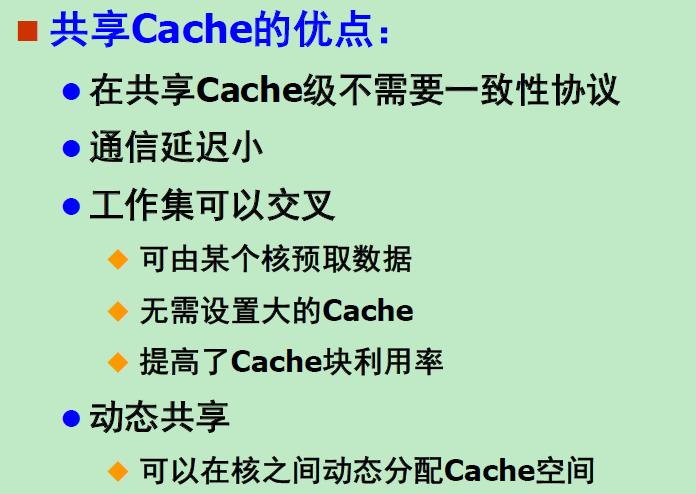
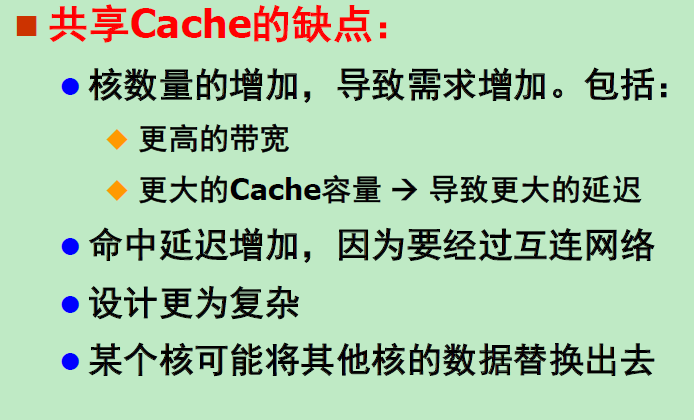
* + 分布式存储器结构
  + 共享存储器结构，共享访问的 Cache 多大
  + Cache 层次

**（4）核间通信技术：**

各CPU核执行的程序之间有时需要进行数据共享与同步，因此某硬件系统必须支持核间通信，高效的通信机制是CMP处理器高性能的保障。

* + 通过基于总线共享的 Cache
  + 通过片上互连网络

共享cache

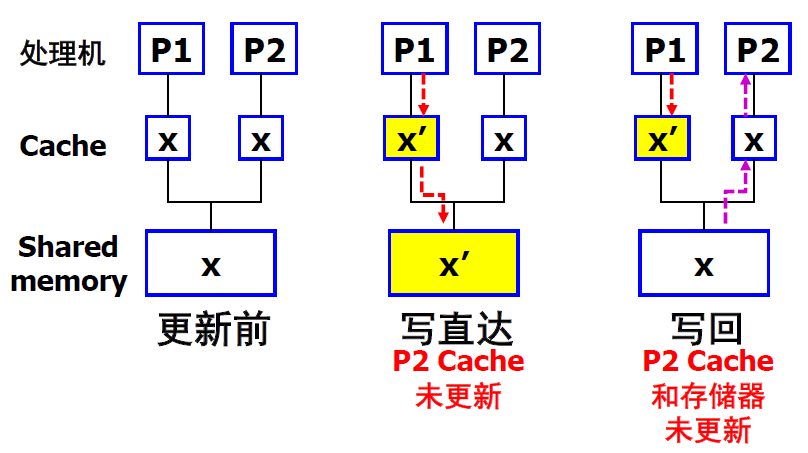
 

## 7.4 多处理机的多cache一致性

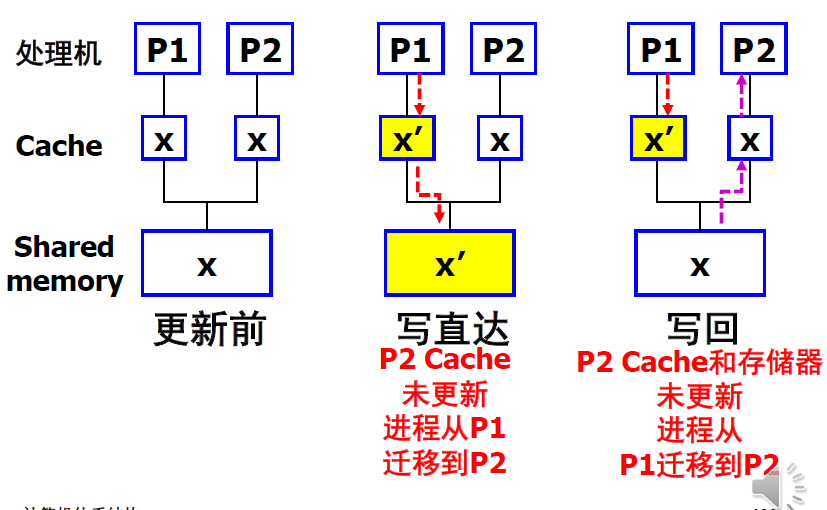
**多处理机的Cache一致性：**

1. 为解决多个Cache之间内容不一致问题。原因有三：**共享可写数据/进程迁移/ I/O传输**

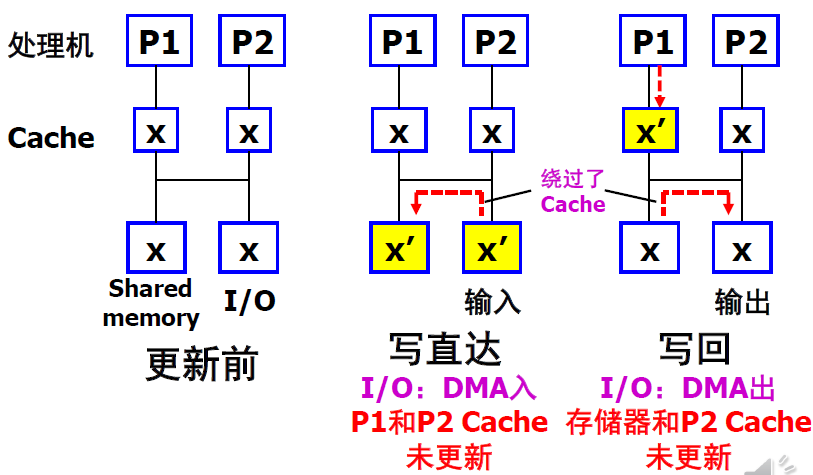
（1）共享可写数据



（2）进程迁移



（3）I/O传输



1. **多处理机的Cache一致性解决方法**

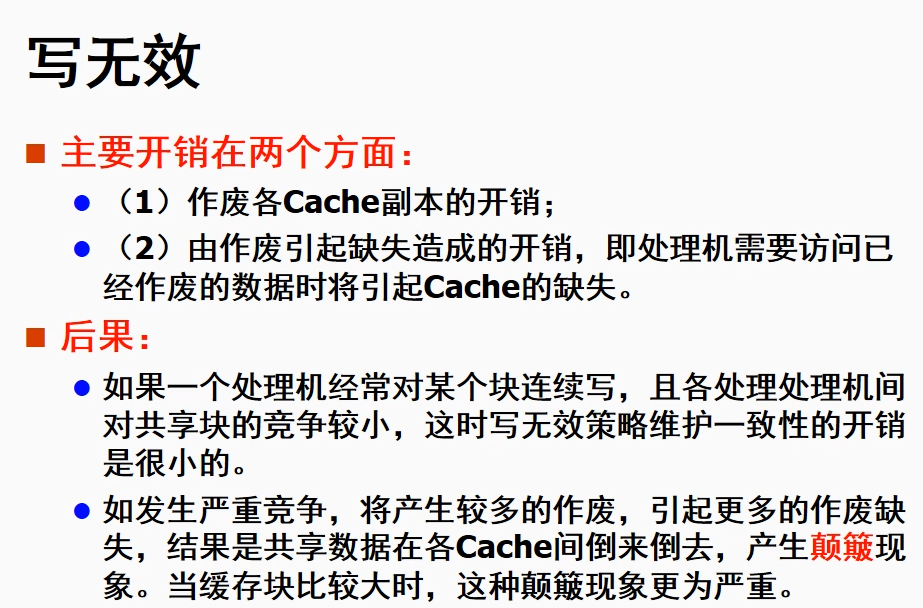
主要有两类：

* 以硬件为基础的做法：

Cache一致性协议：

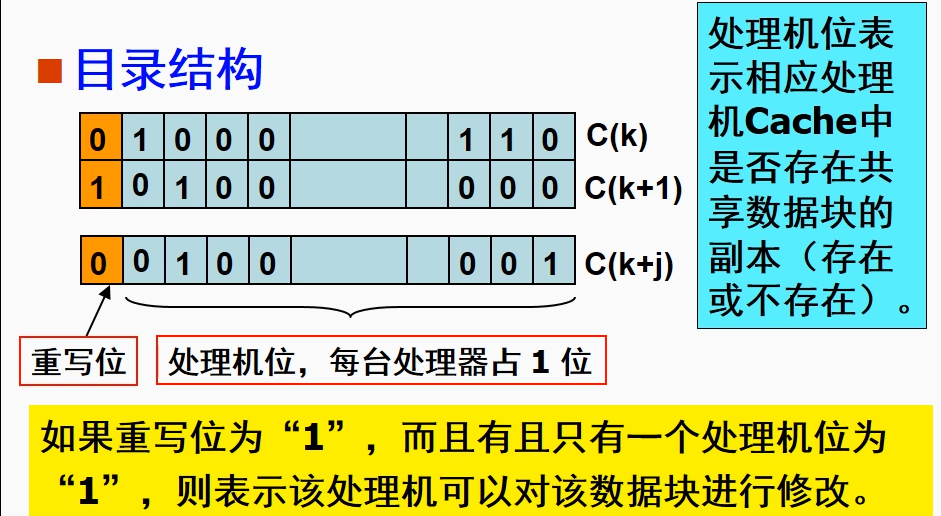
**（1）监听协议：**拥有数据副本的 Cache 各自记录数据块的状态，无集中的状态。适用于基于总线的共享存储器结构的多处理机。**适用于具有广播能力的 总线结构多机系统，但只适用于小规模的多处理机系统。**

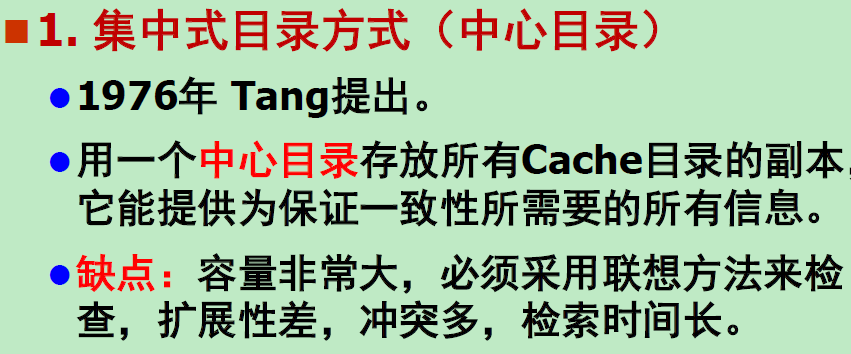
写无效：写共享数据时，使其他Cache中的共享数据的副本无效，独占数据，然后写入或更新。

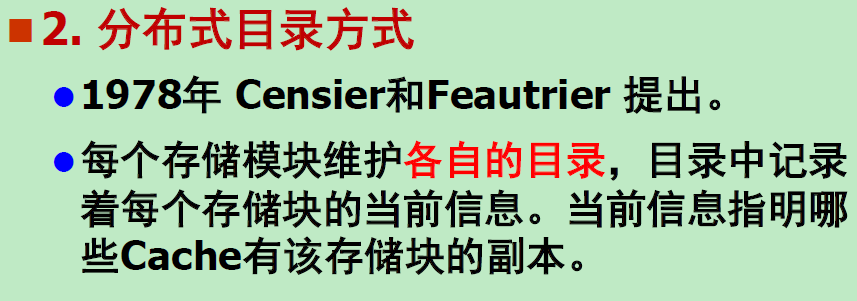
 

写更新：通过总线将更新后的数据分发给所有其他处理机，更新所有Cache中的副本

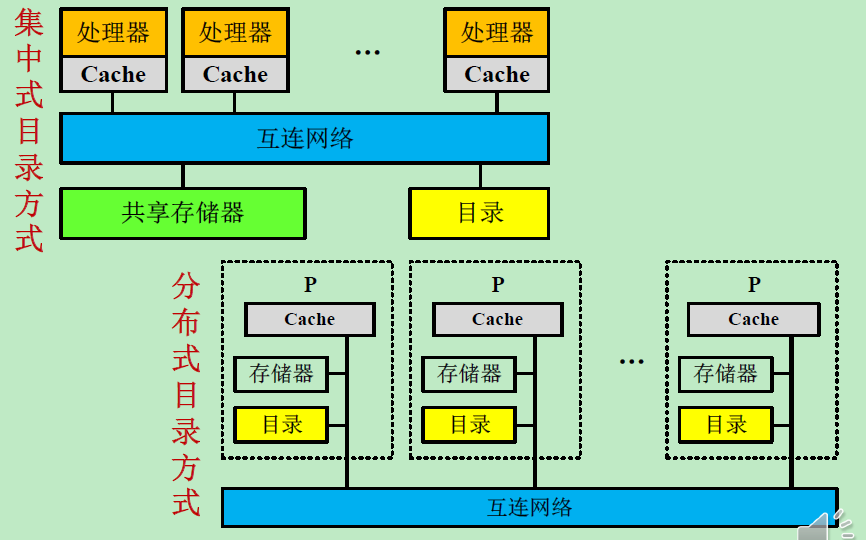
**（2）基于目录的协议：将每个数据块的共享状态记录保存在某个地点-目录。**



* 目录存放：分为集中式（集中存放在本地）和分布式（分布存放在不同处理机中）
  + 集中式（集中存放在本地）
  + 分布式（分布存放在不同处理机中）



* 目录表项：记录数据块被处理机更新的情况



* 目录形式：
  + 全映射（全局存储器中每个块都有存放）、
  + 有限（每个目录项有固定数目的指针）、
  + 链式（将目录分布到各个Cache）

·以软件为基础的做法。

系统加速比



## 7.6 程序并行性

* 并行处理面临的两个重要挑战：

（1）程序中有限的并行性（采用并行性更好的算法）

（2）相对较高的通信开销（靠体系结构支持和编程技术）

* 解决问题的方法
  + 解决问题的方法：
  + 并行性不足： 采用并行性更好的算法
  + 远程访问延迟的降低： 靠体系结构支持和编程 技术
  + 例如：减少远程访问频率的措施：
    - 缓存共享数据 (HW)
    - 合理分布数据，尽可能多地访问本地数据
  + 目前： 使用硬件方法，通过使用 Cache 减少延迟
  + 并行性开发途径：语言、编译、算法、 OS 等方面
* **并行算法分类**
  + 根据运算的**基本对象的不同** 分为：
    - 数值并行算法（数值计算）
    - 非数值并行算法（符号计算）
  + 根据**进程之间的依赖关系**分为
    - 同步并行算法（步调一致）
    - 异步并行算法（步调、进展互不相同）
    - 纯并行算法（各部分之间没有关系）。
  + 根据并行计算任务的大小分为
    - 粗粒度并行
    - 细粒度并行
    - 中粒度并行
  + *并行的粒度越小，就有可能开发更多的并行性，提高并行度，但是通信次数和通信量就增加很多。*
  + 并行算法分为
    - 多机并行
    - 多线程并行

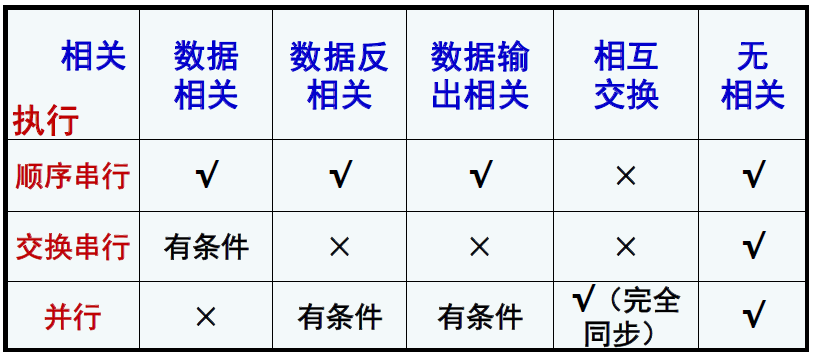
### 7.6.2 程序段之间相关性分析

程序段中各类数据的相关 是限制程序并行的重要因素。

多处理机上的相关:



1. 数据相关:“先写后读 ”，可以顺序串行 但 不能并行
2. 数据反相关: “先读后写 ”，可以顺序串行，不能交换串行，在特殊情况下可以并行。数据输出相关。“写 写” ”，可以顺序串行，不能交换串行，在特殊情况下可以并行。
3. 控制相关：条件语句
4. 相互交换：同时具有“先写后读”和“先读后写”相关，以交换数据为目的

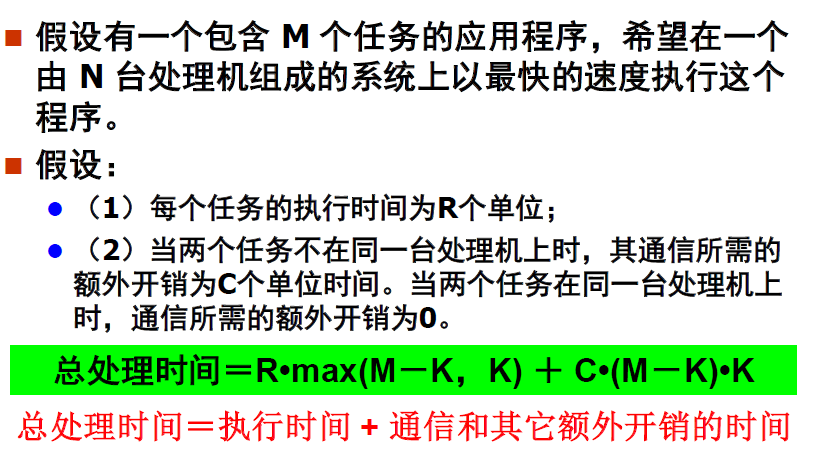


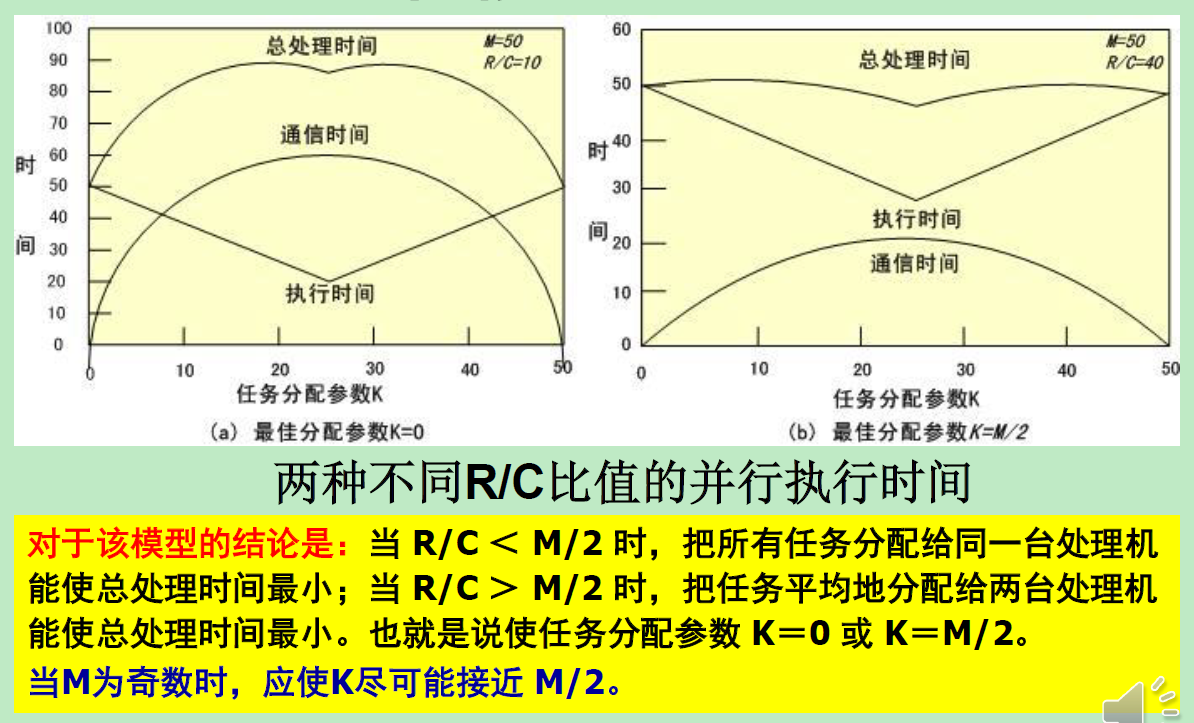
## 7.7 多处理机性能

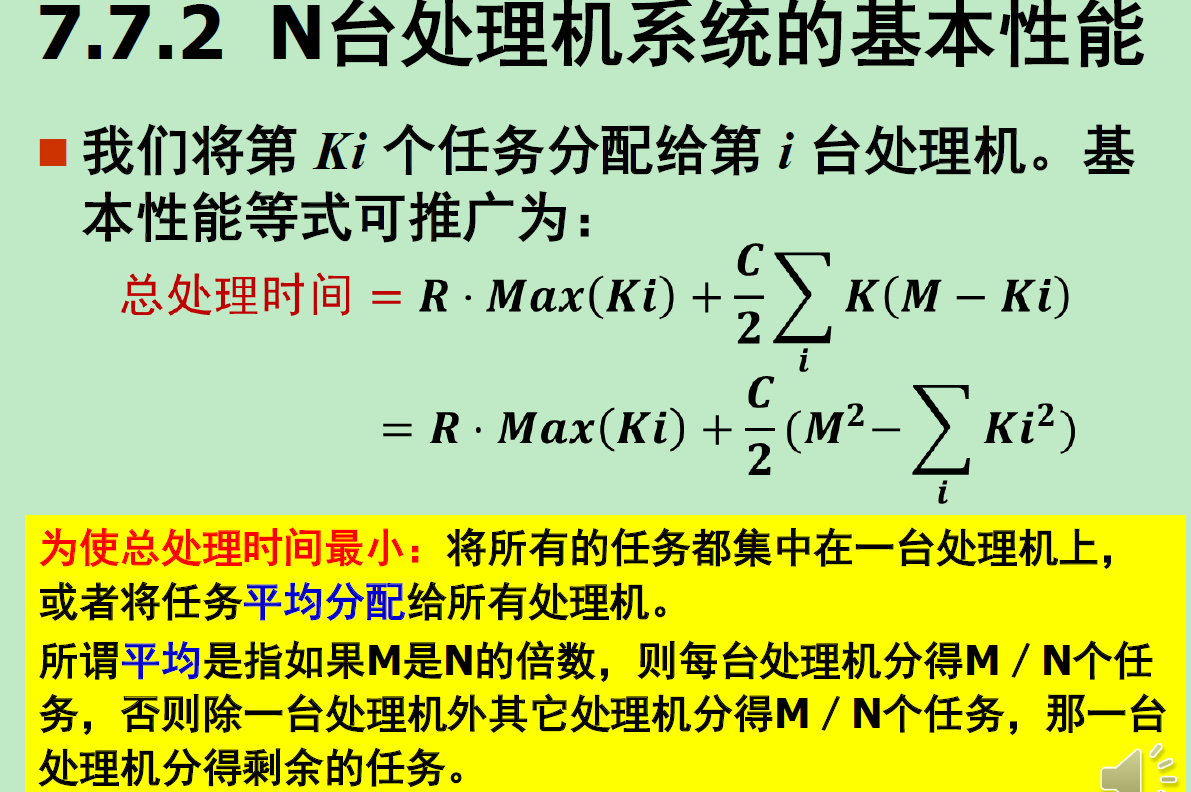
1. 任务粒度的大小会显著影响多处理机的性能和效率
2. 性能和效率在很大程度上依赖R/C比值，R:程序执行时间，C:通信开销。
3. R/C比值是衡量任务粒度大小的尺度，R/C越大，粒度越大，通信开销小；

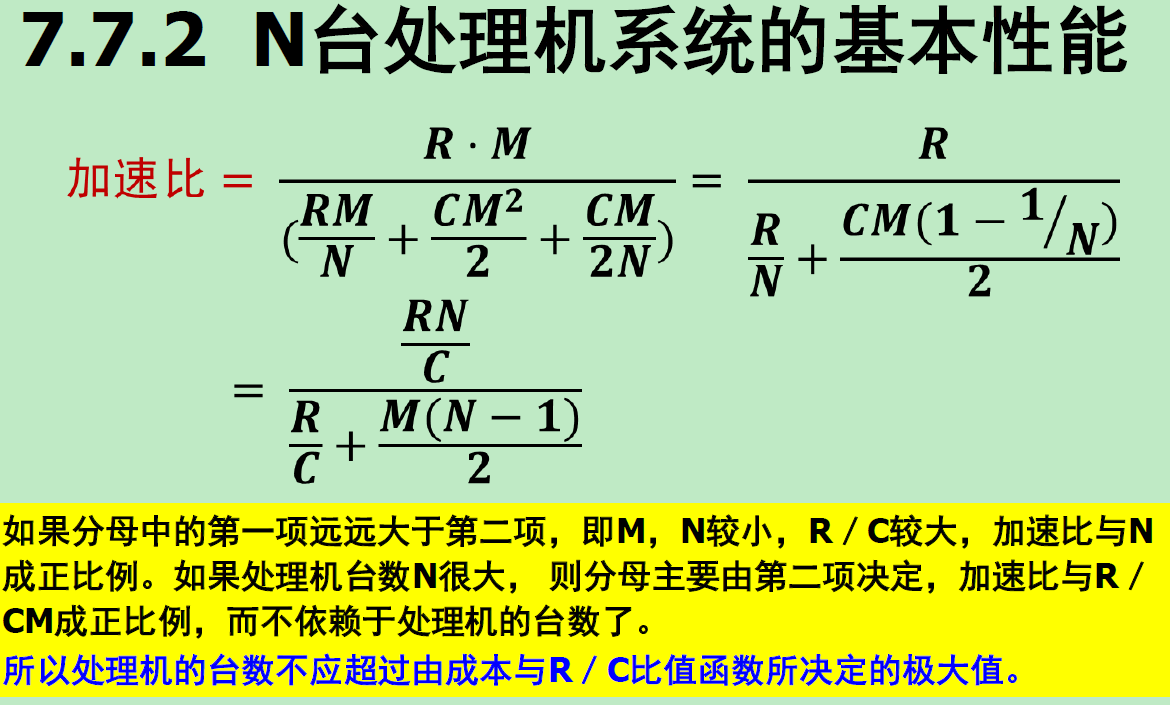
R/C越小，粒度越小，通信开销大。

14.总处理时间：









## 7.8多处理机操作系统

多台处理机协同工作完成所要求任务的操作系统

1. 多处理机操作系统的类型（按照结构来划分）：
2. 主从型：由一台主处理机进行系统的 **集中控制** ，负责记录、控制其它从处理机的状态，并分配任务给从处理机。
   1. 优：简单；
   2. 缺：对主处理机可靠性要求高，出现不可恢复错误时易崩溃，需要重启，系统灵活性差
3. 独立监督式：操作系统将控制功能分散给多台处理机，共同完成对整个系统的控制工作。每个处理机均有各自的管理程序**(**操作系统的内核**)**
   1. 优：每个处理机相对独立，出故障不会影响其他，访问公用表格冲突少，系统效率高；
   2. 缺：复杂、进程调度复杂性和开销大、各处理机负荷平衡困难
4. 浮动监督式：系统中 每次只有一台处理机 作为执行全面管理功能的“主处理机”，“主处理机”可以根据需要浮动，即从一台 切换 到另一台处理机。这样，即使执行管理功能的主处理机故障，系统也能照样运行下去；
   1. 优：系统可靠性强，没有单主处理崩溃瓶颈，处理机负载均衡；
   2. 缺：系统实现复杂）