



# 第四单元 输入/输出系统和设备

## 第二讲 总线

刘卫东

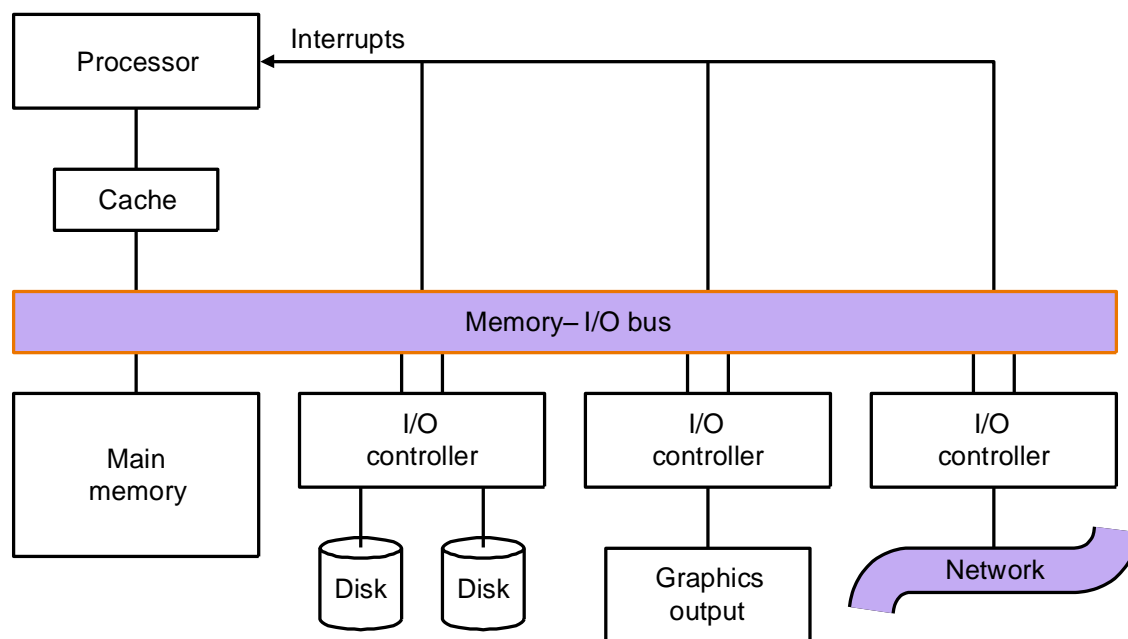
计算机科学与技术系

# 主要教学内容



- ❖ 总线概念
- ❖ 总线分类
- ❖ 总线组成
- ❖ 设计总线的关键问题
  - ❖ 总线仲裁
  - ❖ 通信方式
- ❖ 总线举例

# 处理器和其他组成部件的接口



- ✚ 输入/输出系统设计受到多方面因素的影响（可扩展性，可恢复性等）
- ✚ 性能：
  - 访问延迟
  - 吞吐量
  - 设备和系统的连接关系
  - 层次存储系统
  - 操作系统
- ✚ 用户和应用也各不相同



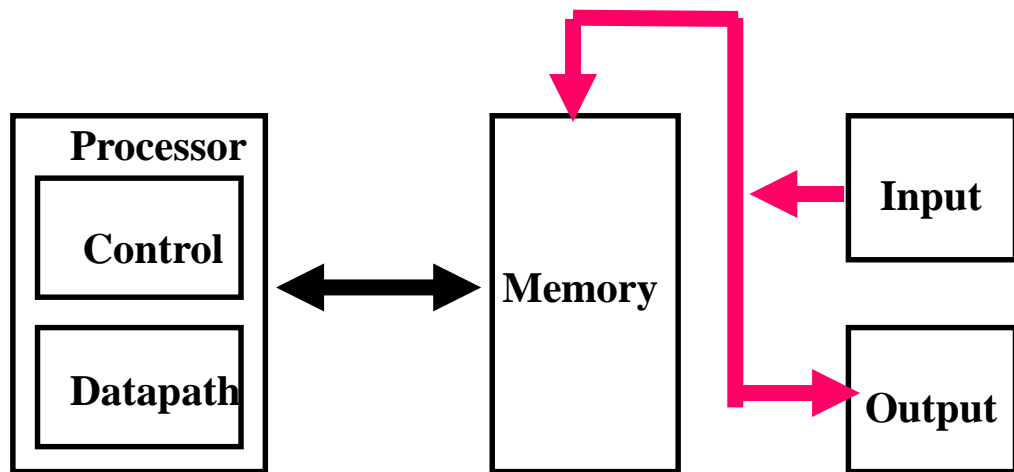
# 什么是 bus?



- 公共汽车：一种公共交通工具
- 一组导线
- 共同点：多个使用者共享通道
  - 不再是通信双方进行点到点的连接
  - 总线上的信号可以被所有连接在总线上的设备接收到

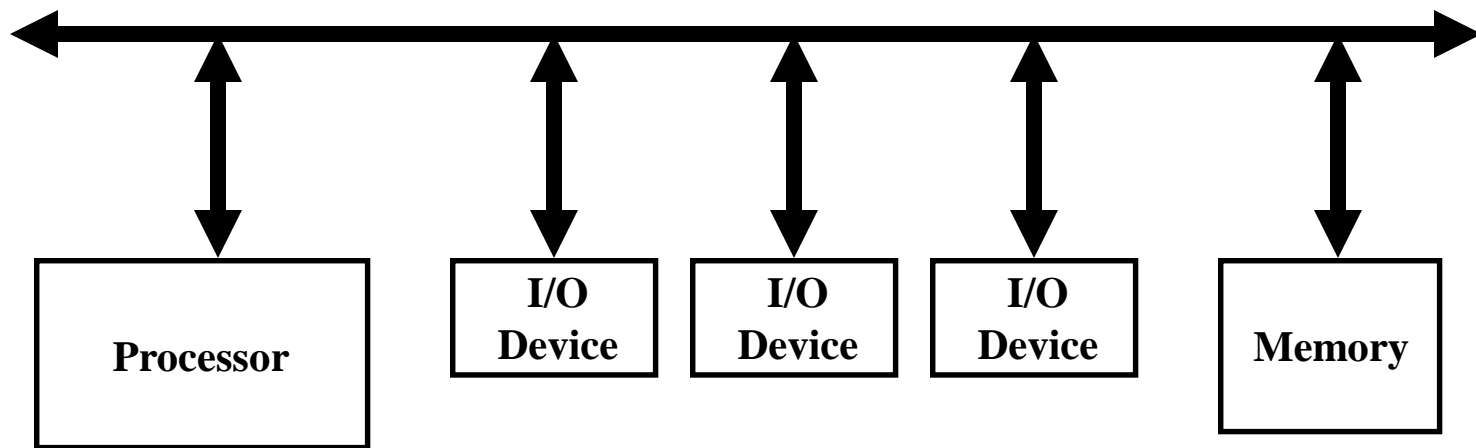
# 计算机总线

- 共享的信息通道
- 用于连接计算机多个子系统（部件）



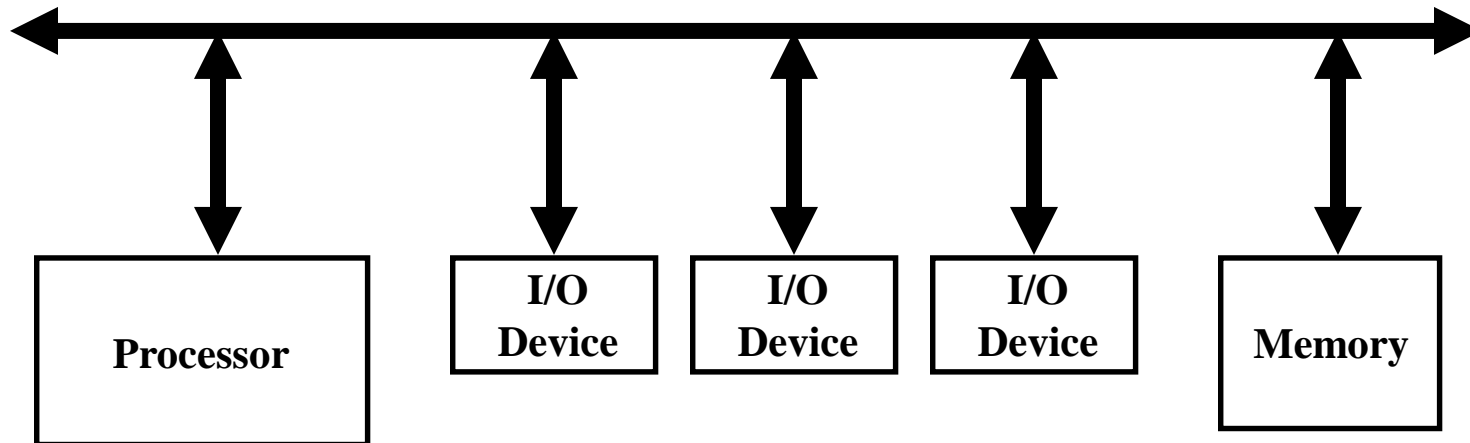
- 总线也是连接复杂巨系统的一种基本工具
  - 功能抽象

# 使用总线的好处



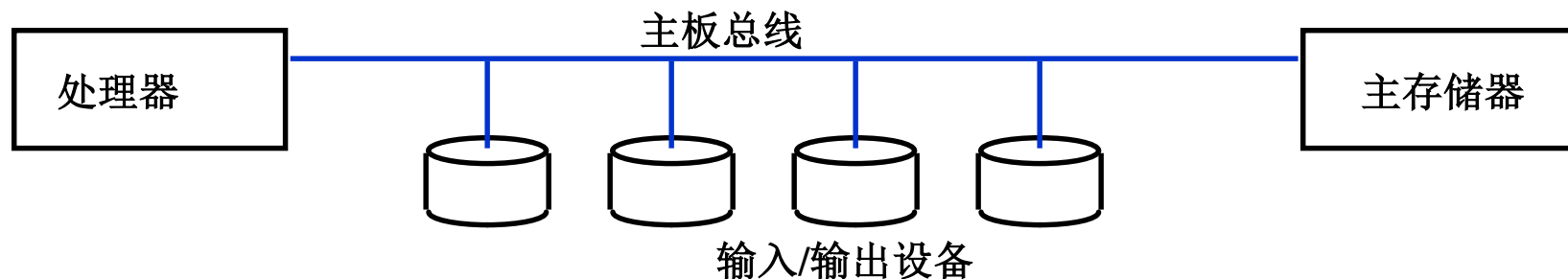
- ✿ 解决外部设备“多”的问题
  - ✦ 容易增添新的设备
  - ✦ 使用相同总线标准的外部设备可在不同计算机间兼容
- ✿ 降低成本
  - ✦ 总线可供多个设备共享
- ✿ 简化设计
  - ✦ 将数据传输的功能抽象到总线上

# 总线的不足



- ✿ 容易成为信息通道的瓶颈
  - ❑ 总线故障
  - ❑ 总线带宽限制了整条总线的吞吐量
- ✿ 总线的最高速度主要由下列因素决定：
  - ❑ 总线长度
  - ❑ 总线负载的设备数
  - ❑ 负载设备的特性
    - ◆ 延迟是否差异较大？
    - ◆ 数据传输率差异较大？

# 单总线计算机：主板总线



## ✿ 使用一条总线：

- ✿ 处理器和主存储器之间通信

- ✿ 主存储器和输入/输出设备之间通信

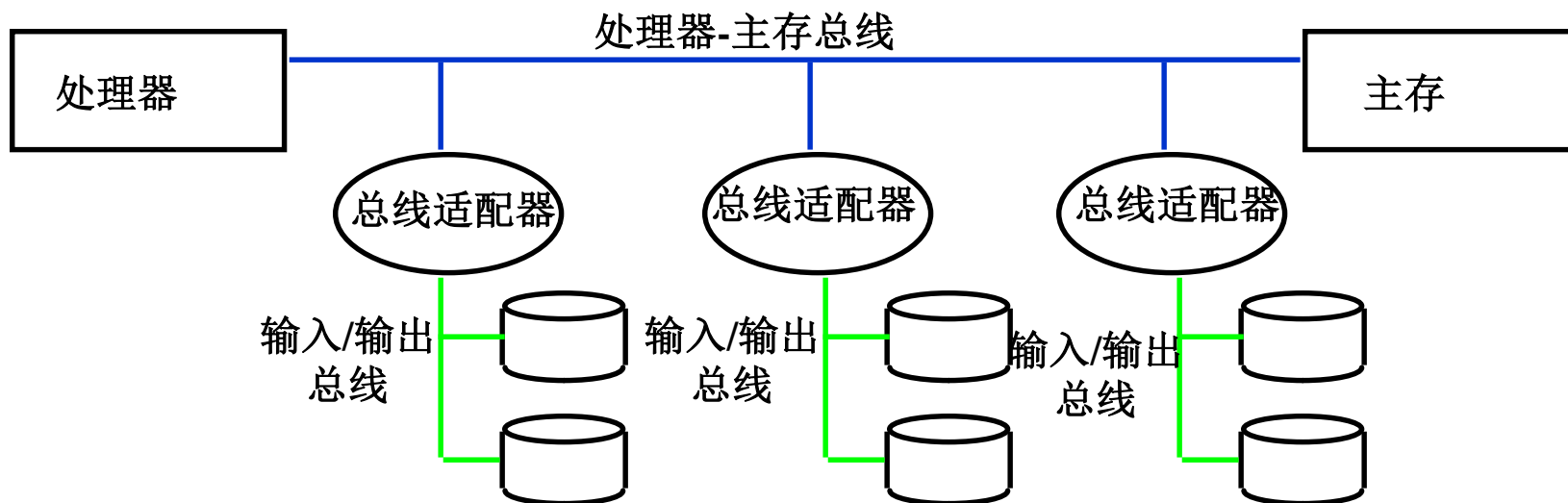
- ✿ 优点：简单、成本低

- ✿ 缺点：速度慢，总线将成为系统瓶颈

- ✿ 应用：PDP-1、IBM PC – ISA EISA

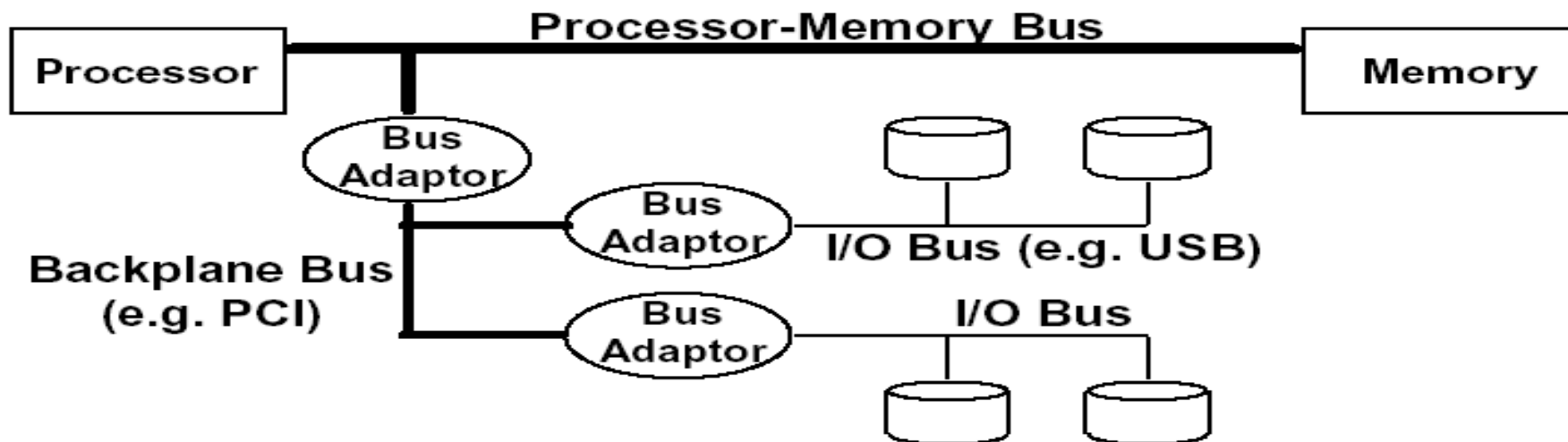


# 双总线系统



- ✿ 输入/输出总线通过适配器和处理器-主存总线相连：
  - ✦ 处理器-主存总线：主要用于处理器和主存储器之间的通信
  - ✦ 输入/输出总线：为输入/输出设备提供信息
- ✿ 应用举例：
  - ✦ Apple Macintosh II
    - ◆ NuBus：处理器、主存和选定的少量I/O设备
    - ◆ SCSI总线：其余I/O设备

# 三总线系统



## ❖ 主板总线连接到处理器-主存总线

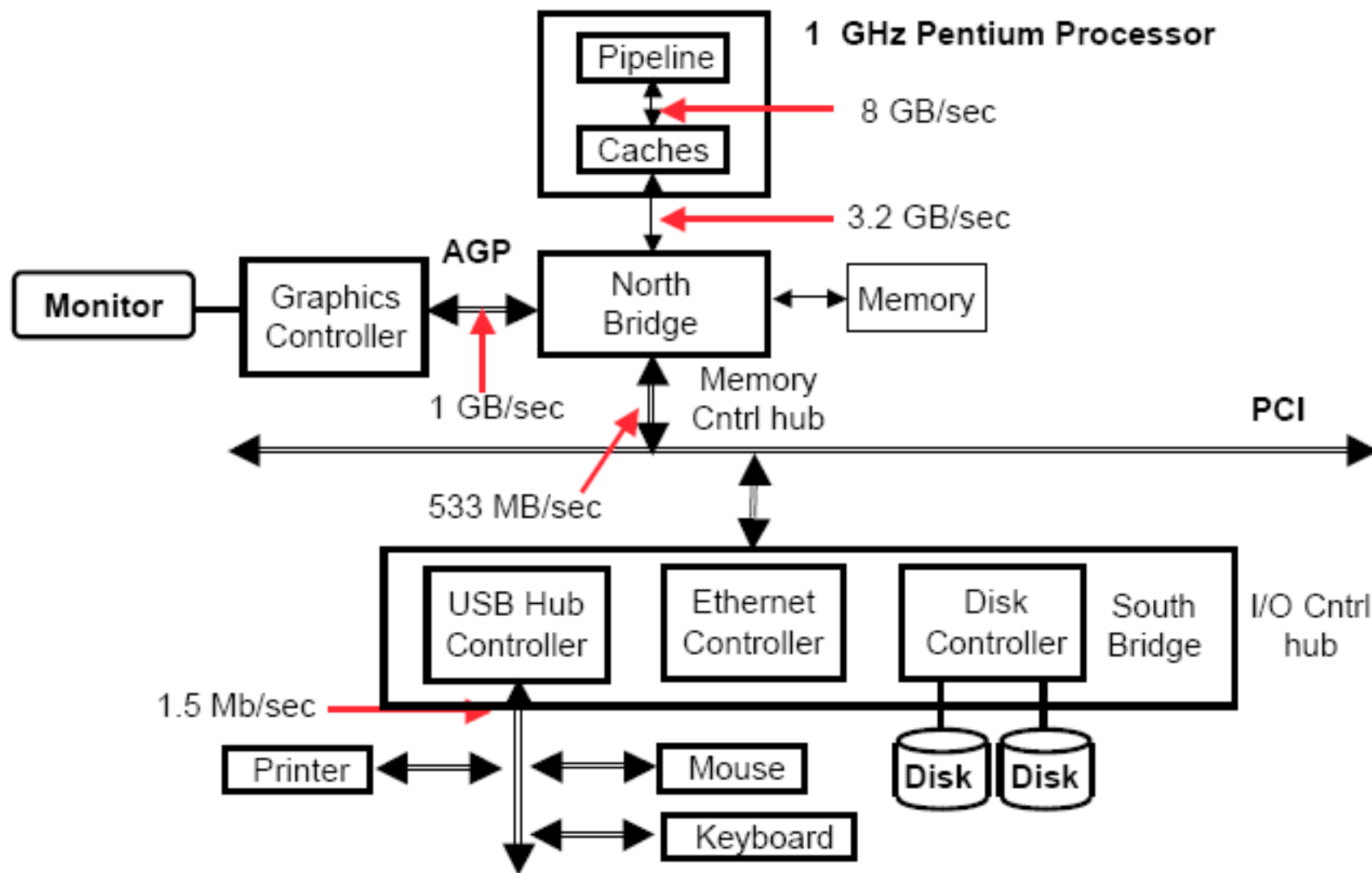
- ❑ 处理器-主存总线主要用于处理器和主存之间数据交换
- ❑ I/O总线连接到主板总线

## ❖ 优点

- ❑ 大大减少处理器-主存总线负载

## ❖ 例：现代PC采用的结构

# Pentium 计算机的总线



## ❁ 处理器-主存总线（专用）

- ❑ 传输距离短、速度快，主存储器专用

  - ◆ 保证主存储器-处理器之间的高带宽

- ❑ 直接和处理器连接

- ❑ 优化处理使之适应Cache块传送

## ❁ 输入/输出总线（行业标准）

- ❑ 通常距离较长，速度较慢

- ❑ 需要适应多种输入/输出设备

- ❑ 和处理器-主存总线通过桥连接（或通过主板总线）

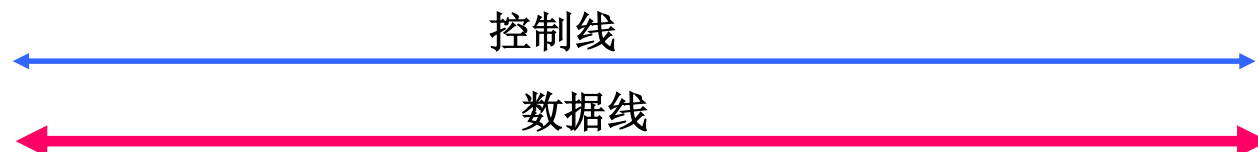
## ❁ 主板总线（行业标准或专门设计）

- ❑ 主板：连接各部件器件的底盘

- ❑ 应允许处理器、主存储器和输入/输出设备互连

- ❑ 应有价格优势：所有组件连接在一条总线上

# 总线的一般组成



## ❊ 控制线:

- ❑ 标记总线事务的开始和结束
- ❑ 指明数据线上传输信息的类型

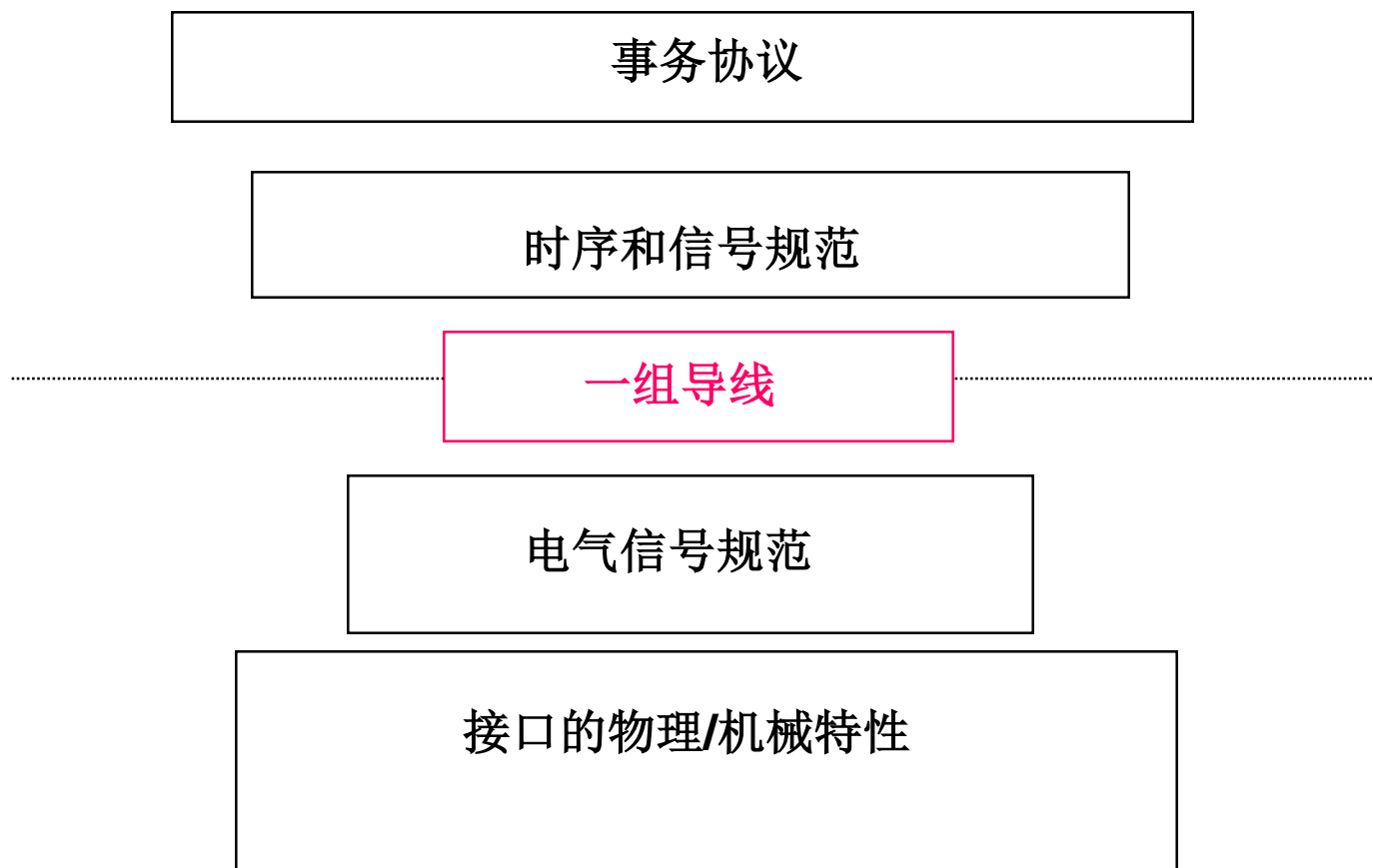
## ❊ 数据线 在源设备和目标设备间传送信息

- ❑ 数据和地址
- ❑ 复杂的命令

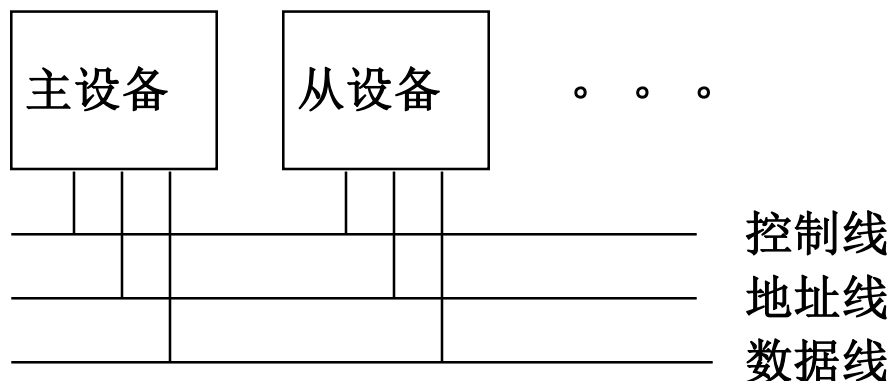
- ❖ 设备用于人机交互
- ❖ 总线定义了交互的通信协议/标准:
  - ❖ PCI
  - ❖ EISA
  - ❖ SCSI
  - ❖ USB
  - ❖ Bluetooth
  - ❖ ...

- ❖ 标准十分重要:
  - ❖ 不同公司设计的外部设备，应该能在同一计算机上安装使用。
  - ❖ 不同公司的计算机也应该可以使用某一外部设备。
  - ❖ 外部设备的通讯速度差异很大
  - ❖ 标准是抽象的设计
  - ❖ 标准可以影响性价比，可靠性等

# 总线结构



# 总线相关概念



**总线主设备：**有能力控制总线，发起总线事务

**总线从设备：**响应主设备请求

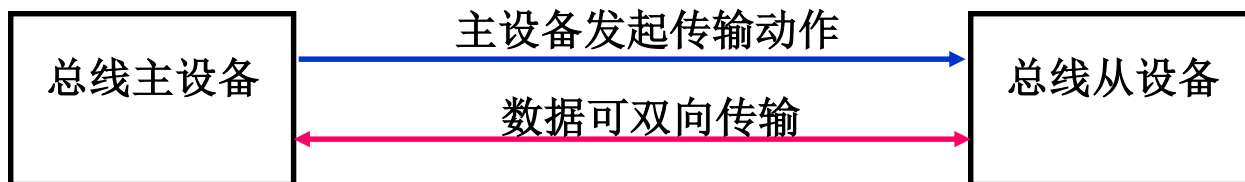
**总线通信协议：**定义总线传输中的事件顺序和时序要求

**异步总线传输：**控制信号（请求，应答）作为总控信号

**同步总线传输：**使用共同的时钟信号



# 主设备和从设备



## ❊ 总线事务包括两个部分：

- ❑ 发起命令（和地址）
- ❑ 传输数据

## ❊ 主设备是总线事务的发起者：

- ❑ 发出命令（和地址）

## ❊ 从设备是总线事务的响应者：

- ❑ 若主设备发出的是读命令，则将数据发送到主设备
- ❑ 否则，接收主设备发来的写入数据

# 仲裁：获得总线使用权



## ❁ 总线设计中重要问题之一：

- ❁ 如何为需要使用总线的设备安排总线？

## ❁ 可通过主—从设备的安排来避免冲突：

- ❁ 只允许总线主设备发起总线事务，控制所有总线请求
- ❁ 从设备响应主设备的读写请求

## ❁ 最简单的设计：

- ❁ 处理器作为唯一的总线主设备
- ❁ 所有总线请求均由处理器控制
- ❁ 主要缺点：处理器被卷入到每一个总线事务中

## ✿ 总线仲裁的基本要求：

- ✿ 某总线主设备使用总线前应发出总线请求
- ✿ 只有得到授权后，主设备才能使用总线
- ✿ 使用完毕后，主设备应通知仲裁器

## ✿ 总线仲裁器在以下两方面取得平衡：

- ✿ 高效性：优先级高的设备应该得到优先服务
- ✿ 公平性：但低优先级的设备也应有得到服务的机会

## ✿ 总线仲裁方式：

### ✿ 集中仲裁和分布仲裁

◆ 集中仲裁：例如，交通警察在路口指挥交通

◆ 分布仲裁：路口没有交通警察，所有车辆先停下，确认其他方向没有来车后通行

### ✿ 按优先级仲裁或轮循仲裁

◆ 优先级仲裁：例如，救护车在道路上有高优先级

# 总线仲裁方式



## 集中仲裁

### 菊花链仲裁

- ◆所有设备共用一个总线请求信号

### 集中平行仲裁

- ◆通过集中的仲裁器进行

## 分布仲裁

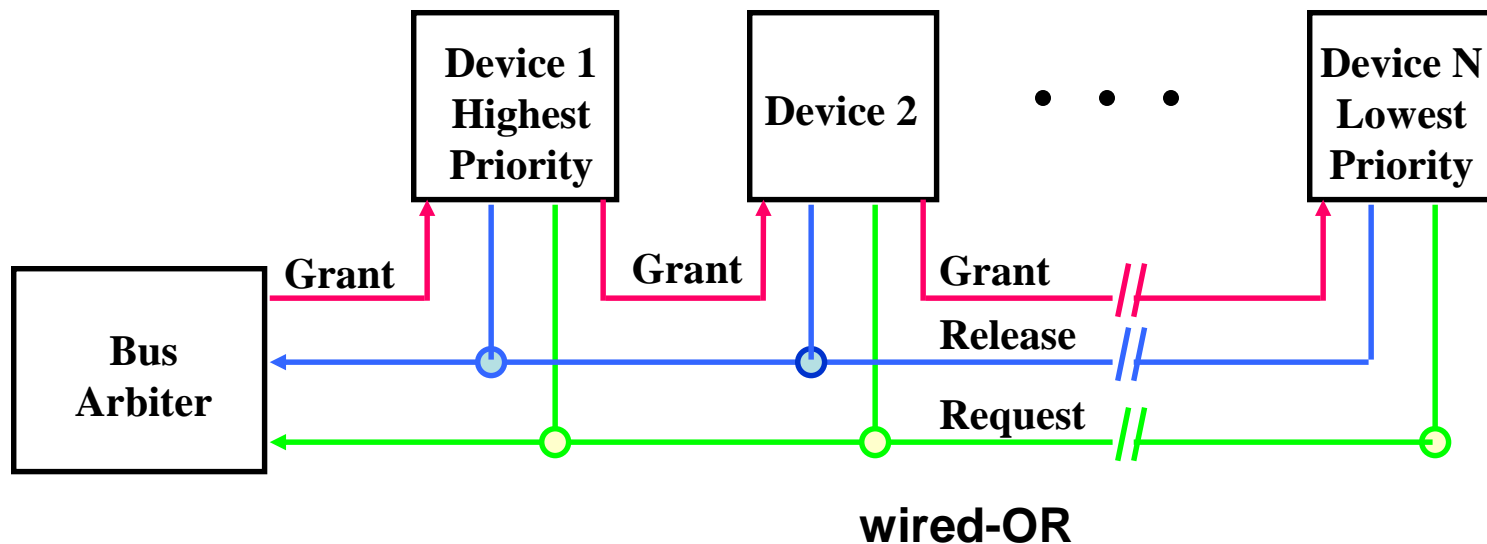
### 通过自我选择进行分布式仲裁

- ◆每个要使用总线的设备将自己的标识放在总线上

### 碰撞检测

- ◆以太网

# 菊链仲裁



优点：简单

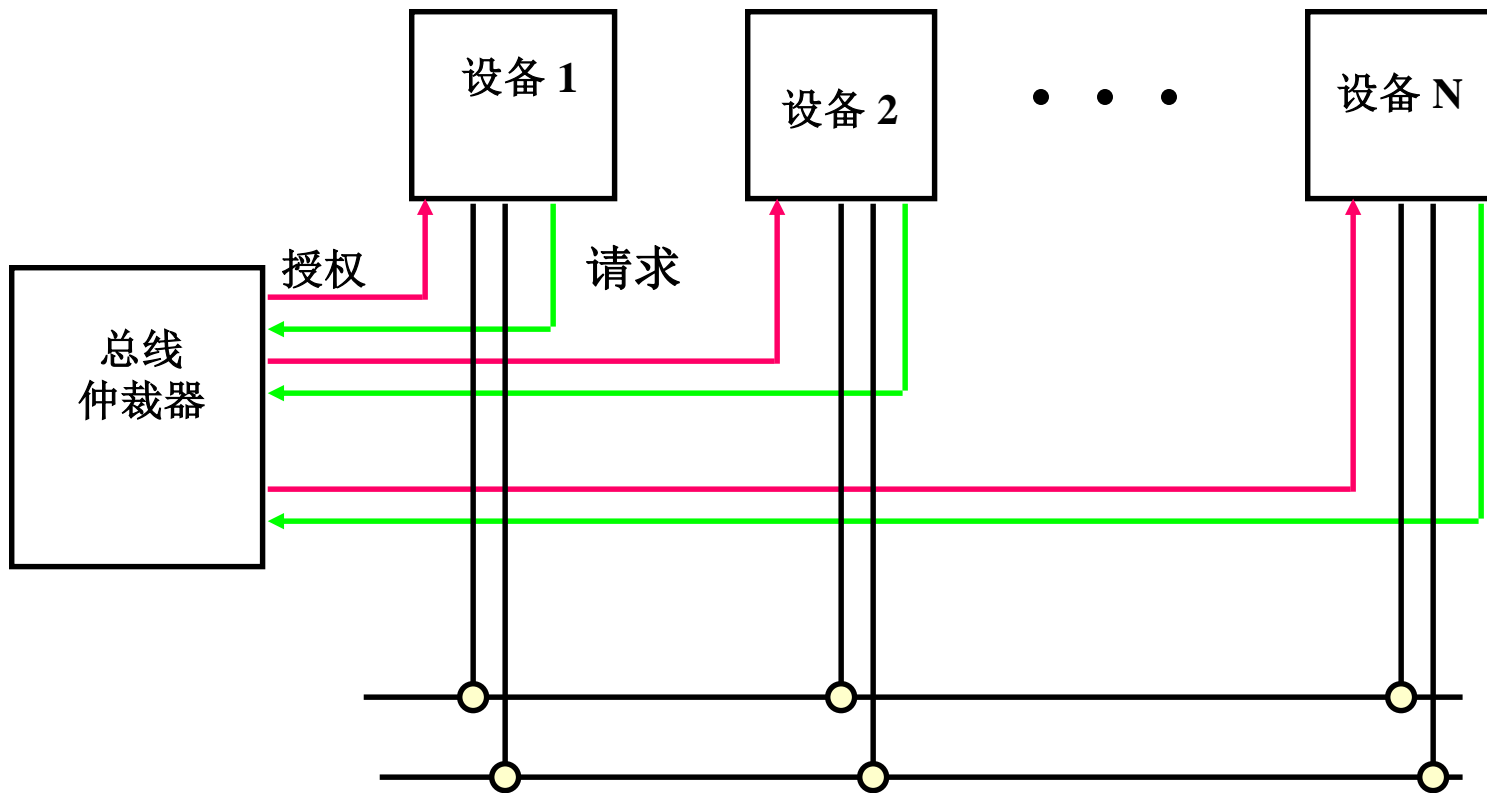
缺点：

- 无法保证公平性

低优先级设备可能得不到总线使用权

- 总线授权信号的逐级传递限制了申请总线的速度

# 集中平行仲裁



- ⊙ 用于几乎所有处理器-主存总线和一些高速输入/输出总线

# 同步和异步总线



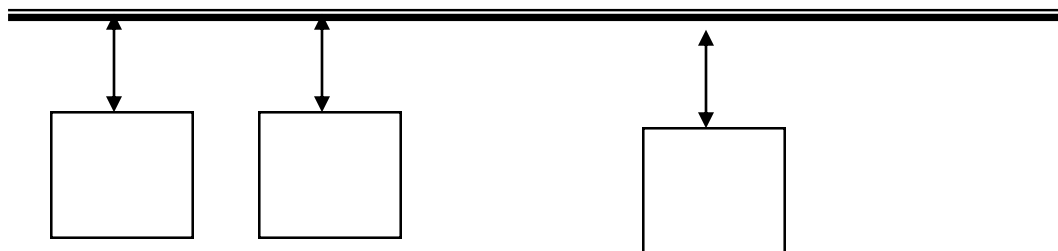
## 同步总线：

- ❖ 控制线中包含有一根时钟信号线
- ❖ 传输协议根据时钟信号制定：
  - ◆ 例如：主设备提出总线请求后5个时钟周期，可以获得能否使用总线的信号。
- ❖ 优点：逻辑简单、高速
- ❖ 缺点：
  - ◆ 总线上所有设备必须按时钟频率工作
  - ◆ 为防止时钟信号扭曲，高速工作时，总线距离必须足够短

## 异步总线：

- ❖ 不使用统一的时钟
- ❖ 可适应设备的不同速度
- ❖ 不用担心时钟信号扭曲，距离可较长
- ❖ 使用握手协议

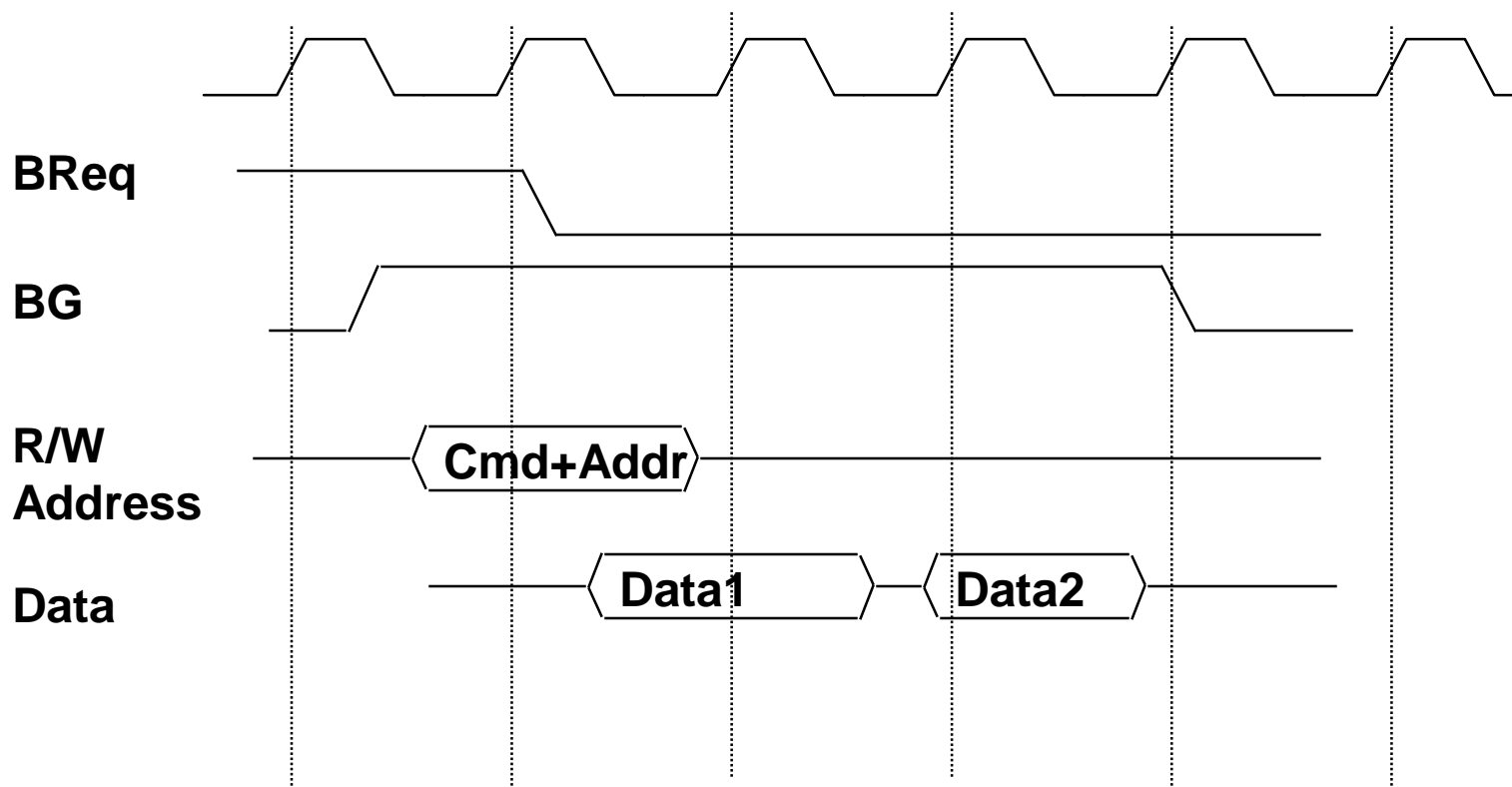
# 最简单的总线模式



- ⊕ 所有设备同步工作
- ⊕ 所有设备以同样的速度工作
- ⊕ => 简单的协议
  - ▣ 只需管理源和目标



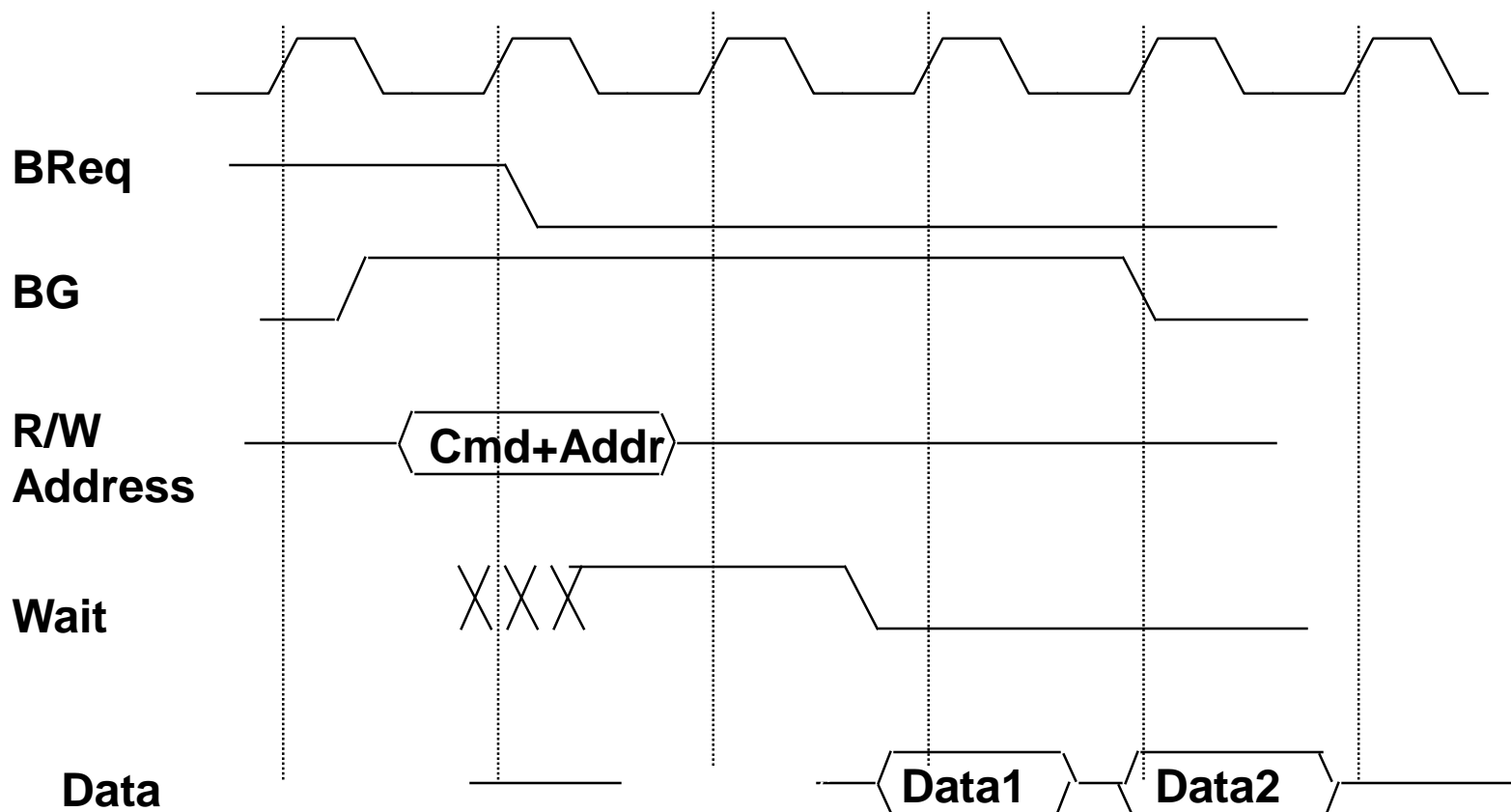
# 简单的同步协议



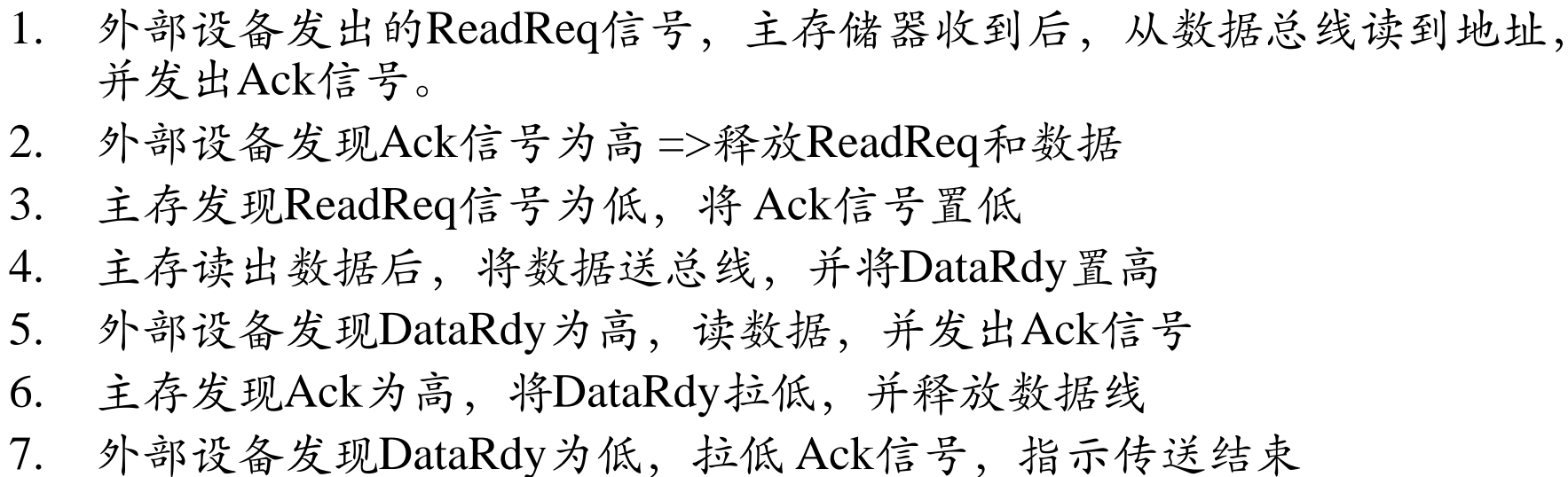
🔴 就是处理器-主存储器总线也比它复杂

- 🔲 主存（从设备）需要响应时间
- 🔲 需要控制数据速度

# 典型的同步协议



- 从设备指示何时开始传送数据
- 实际传送开始后，按总线时钟传送数据

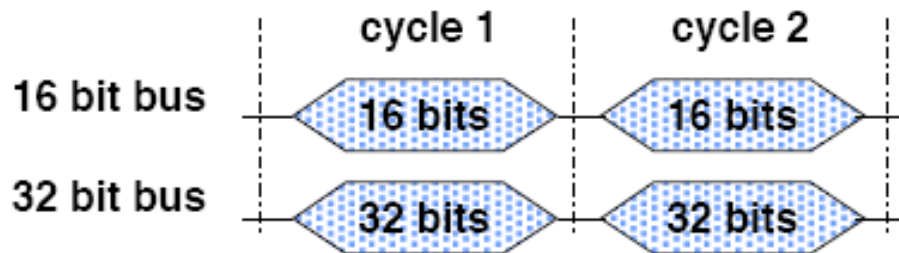


# 增加总线带宽



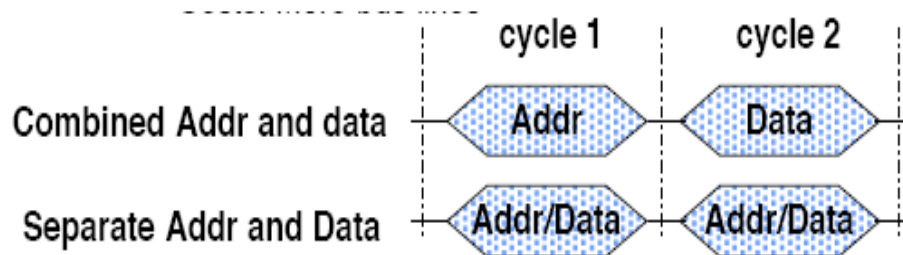
## 增加总线的宽度

- 可增加每个周期传送数据的量
- 提高了成本



## 分别设置数据总线和地址总线

- 可同时传送数据和地址
- 提高了成本

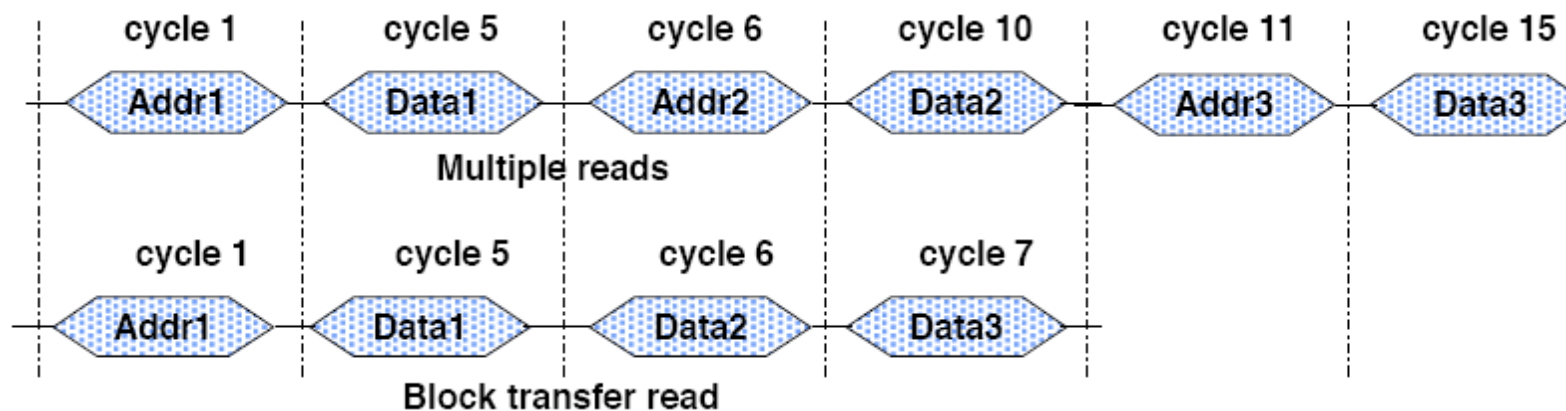


# 增加总线带宽



## 采用成组传送方式

- 一个总线事务传送多个数据
- 每次只需要在开始的时候传送一个地址
- 直到数据传送完毕才释放总线
- 代价
  - 复杂度提高
  - 延长后续总线请求的等待时间



# 多主设备总线提高事务数量



## ⊕ 仲裁重叠

- ⊞ 在当前事务时，为下一总线事务进行仲裁

## ⊕ 总线占用

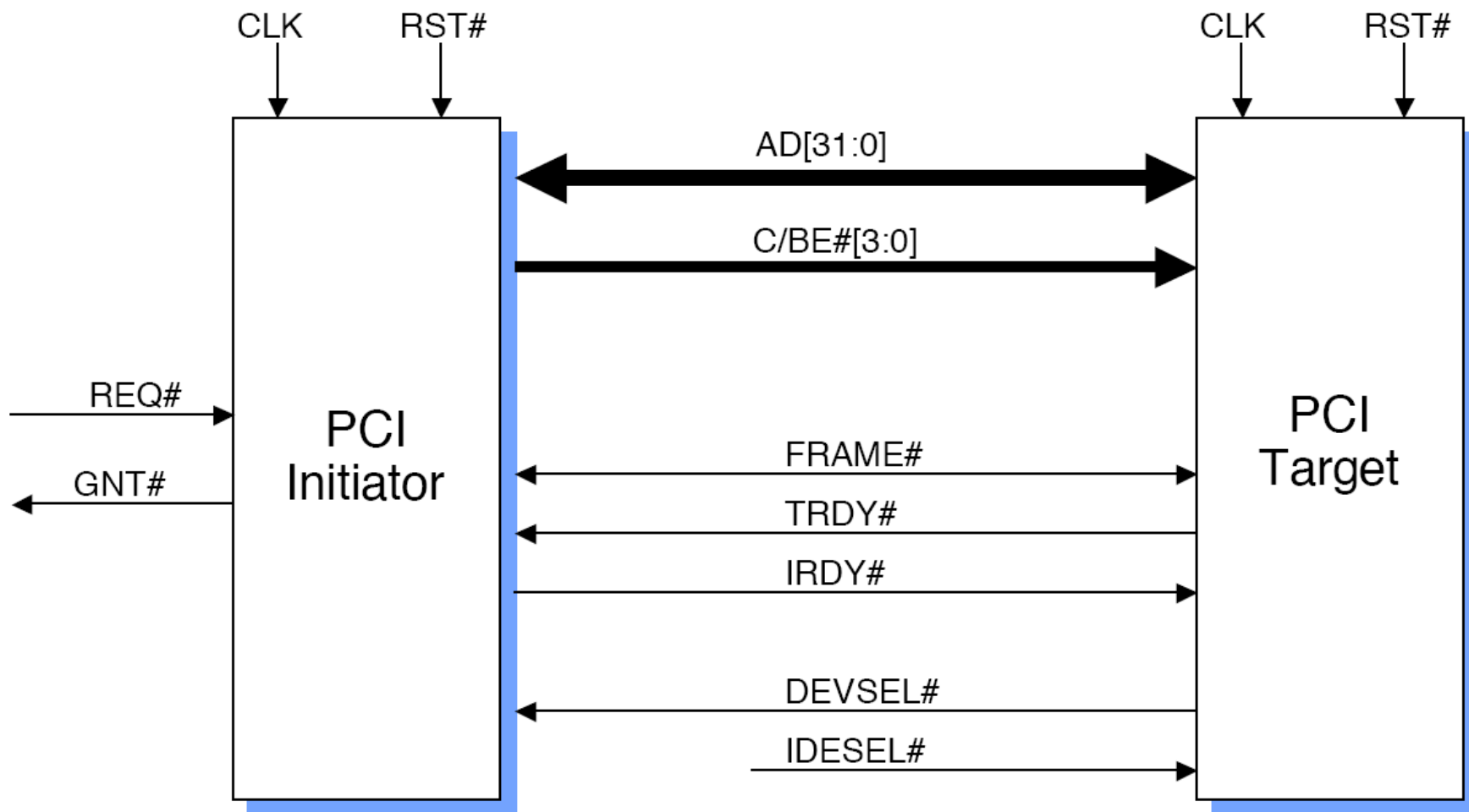
- ⊞ 在没有其他主设备请求总线的情况下，某主设备一直占用总线，完成多个总线事务

## ⊕ 地址、数据传送重叠

## ⊕ 在现代内存总线上，应用了上述全部技术

- ⊗ 外部组件互连总线
- ⊗ 时钟频率：33MHz或66MHz（CLK）
- ⊗ 集中仲裁方式（REQ#、GNT#）
  - ⊠ 和上一事务重叠
- ⊗ 32位地址和数据线互用（AD）
  - ⊠ V2.1 为64位
- ⊗ 总线协议
  - ⊠ 总线周期：内存读、内存写、内存成组读等（C/BE#）
  - ⊠ 地址握手和保持（FRAME#、IRDY#）
  - ⊠ 数据宽度（C/BE#）
  - ⊠ 通过IRDY#和TRDY#握手信号传输变长的数据块
- ⊗ 最大带宽达133MB（33MHz）或528MB（66MHz）

# 32位PCI总线的信号







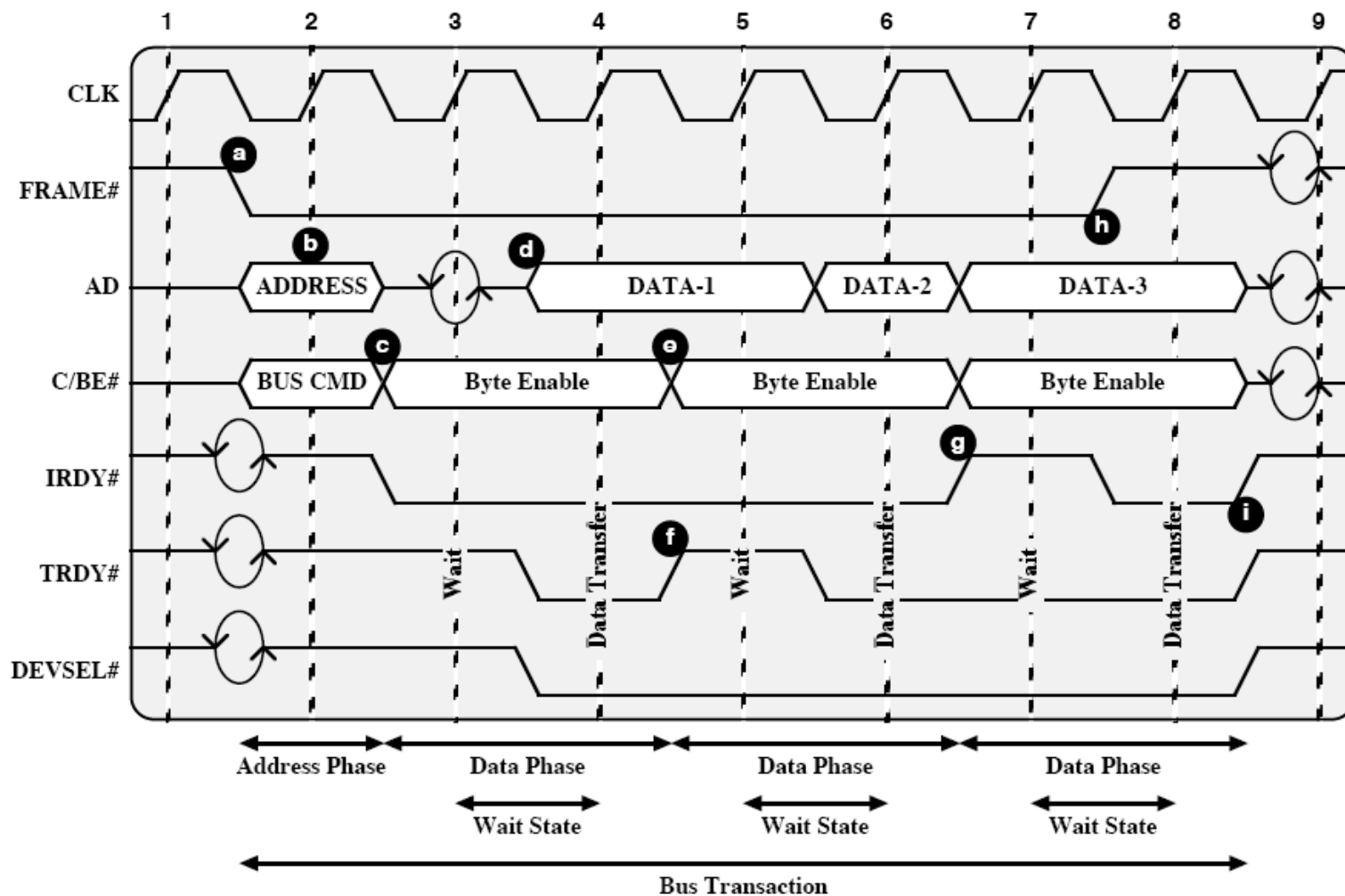
# PCI 总线的读/写事务

- ⊕ 所有信号在时钟正边沿采样
- ⊕ 集中平行仲裁
  - ⊠ 和上一事务重叠
- ⊕ 所有事务可无限制成组传送
- ⊕ 地址段起始于 FRAME#信号有效
- ⊕ 第一时钟周期主设备发出cmd和address
- ⊕ 数据传送
  - ⊠ 当主设备准备好传输数据，主设备发出IRDY# 信号
  - ⊠ 从设备准备好传输数据，发出TRDY#信号
  - ⊠ 上述两个信号均有效时的时钟上升沿开始传送数据
- ⊕ 主设备准备结束数据传送时，将FRAME#信号失效

# PCI 总线读事务



## PCI Read



- ❖ 总线主设备得到授权后，将FRAME#置为有效，开始读事务。并通过AD发送要读的地址，C/BE#发送读命令
- ❖ 从设备从AD上识别是否被选中
- ❖ 主设备释放对AD的控制，同时，在C/BE#上给出AD上哪些位是有用的（1~4Bytes）。并置IRDY#为有效，表示已准备好，可以接收数据。
- ❖ 被选中的从设备置DEVSEL信号，表示已收到命令并可响应。将读出的数据送AD，并置TRDY#通知主设备接收。

- 主设备可在周期4读到第一个数据。并根据需要决定是否要改变C/BE#的值。
- 如果从设备的速度不高，则需要插入等待周期。
- 主设备通过FRAME信号通知从设备结束数据传输，并将IRDY置高。
- 从设备相应地将TRDY和DEVSEL信号置高，总线返回到空闲状态

# PCI 写事务

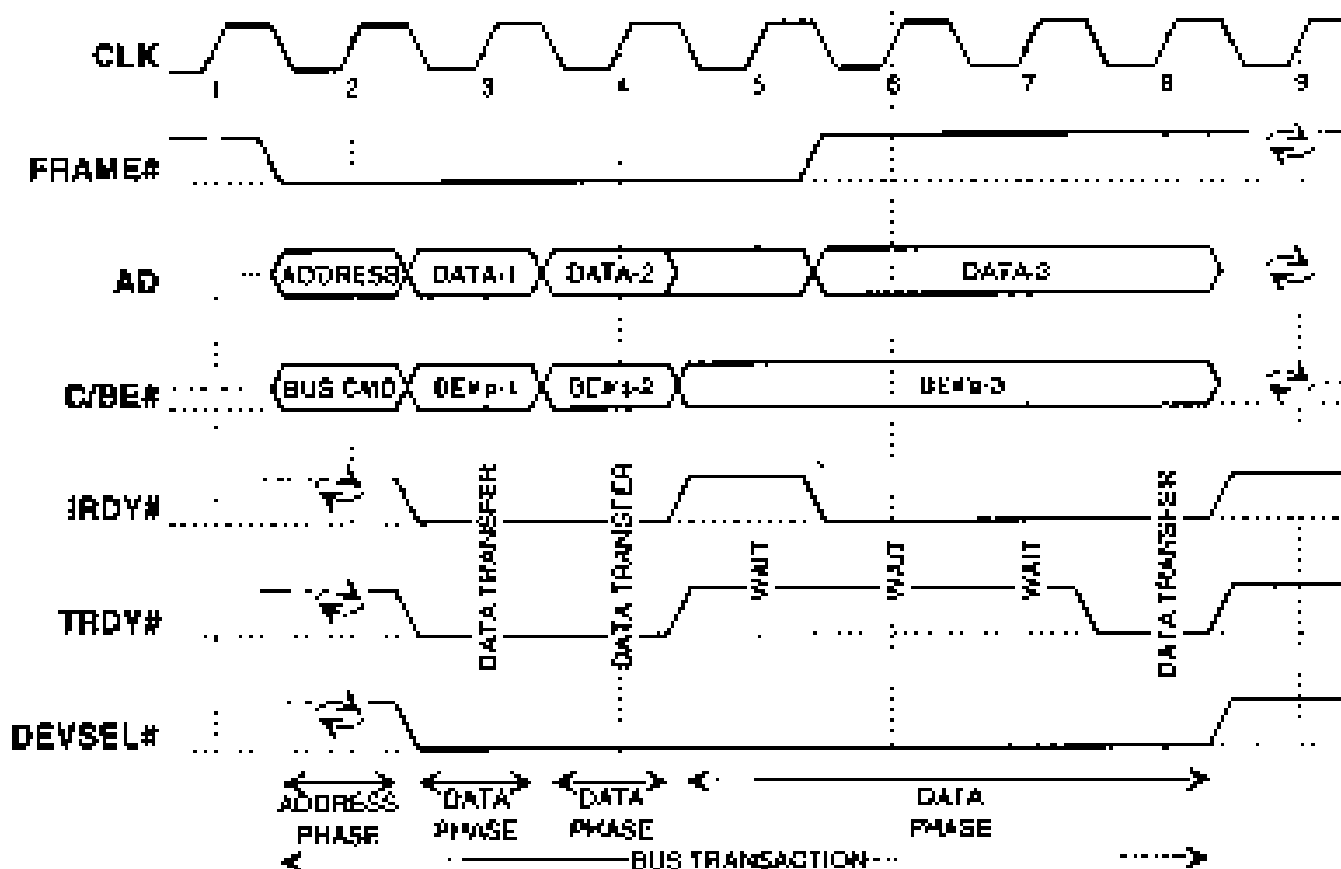


Figure 3-2: Basic Write Operation



# PCI 优化

## 尽量使总线有效传输

- 可采用类似RISC技术，仲裁和数据传输并行进行

## 总线占用

- 为上一主设备保留总线授权，直到有其他主设备申请使用总线
- 得到授权的主设备可在不仲裁的情况下直接开始下一传送过程

## 仲裁时长

- 主设备和从设备尽力延长传输流（使用xRDY）
- 从设备使用 STOP (abort or retry)信号终止连接
- 主设备通过FRAME信号终止连接
- 仲裁器通过 GNT信号终止连接

## 延迟(挂起, 时段分离)事务

- 对慢速设备，在请求后暂时释放总线



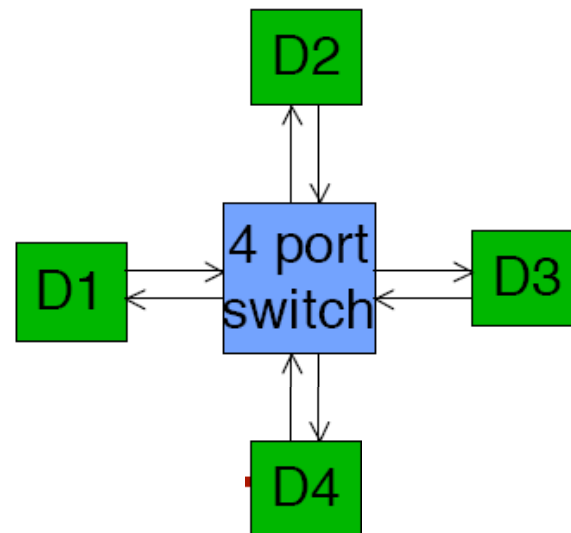
# PCI 的其他问题

- ❖ 中断：
  - ❖ 用于支持控制I/O设备
- ❖ Cache一致性：
  - ❖ 用于支持I/O和多处理器
- ❖ 加锁：
  - ❖ 支持分时操作, I/O和多处理器
- ❖ 可配置地址空间

# 总线发展趋势



- 逻辑总线，物理交换
- 许多总线已采用新的点到点标准
  - 3GIO
  - PCI Express
  - Serial ATA







# 总线参数选择

## 选择

总线宽度  
数据宽度  
大小  
主设备  
时钟  
协议

## 提高性能

将地址和数据线分开  
越宽越快 (32位)  
多字可减少总线开销  
多主设备 (仲裁)  
同步  
并行

## 降低成本

互用地址和数据线  
越窄越廉价 (8位) 传送  
单字传送简单  
单主设备  
异步  
串行

## 总线

- 多个部件之间进行数据传送的共享通道
- 计算机总线
  - 处理器内部总线
  - 系统总线
  - I/O总线

## 总线设计

- 总线仲裁
- 数据传输模式
- 提高总线性能

## ❖ 阅读

- ❖ 参考书相关内容

## ❖ 思考

- ❖ 计算机总线作用？总线仲裁应考虑哪些方面？
- ❖ 总线数据传输模式有哪些？各有什么特点？

## ❖ 实践

- ❖ 书面作业