

计算机组织结构

## 2 计算机的顶层视图

任桐炜

2021年9月9日



南京大學  
NANJING UNIVERSITY

# 教材对应章节

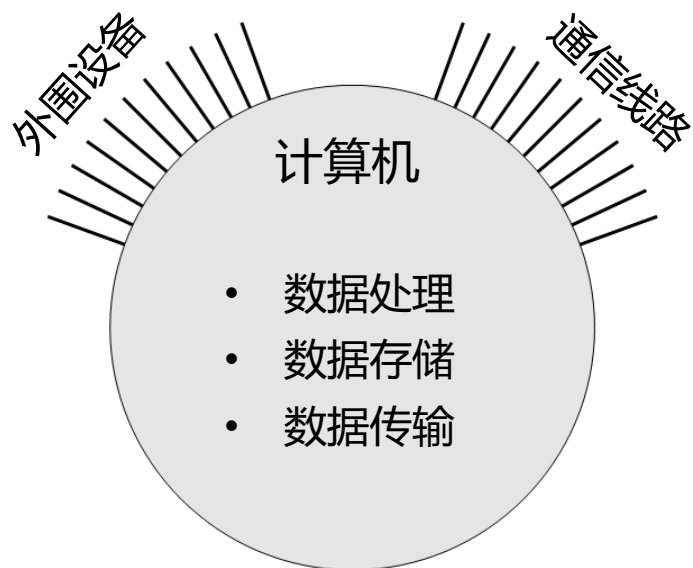


## 第1章 计算机系统概述

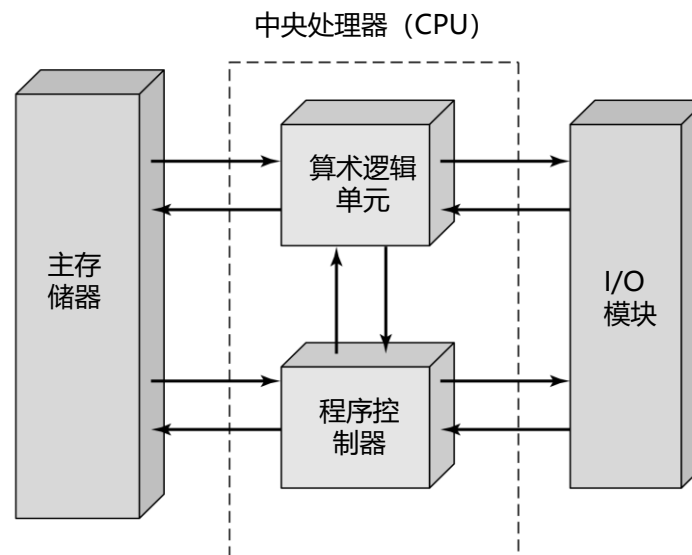


## 第3章 计算机功能和互连的顶层视图

# 计算机的不同视图

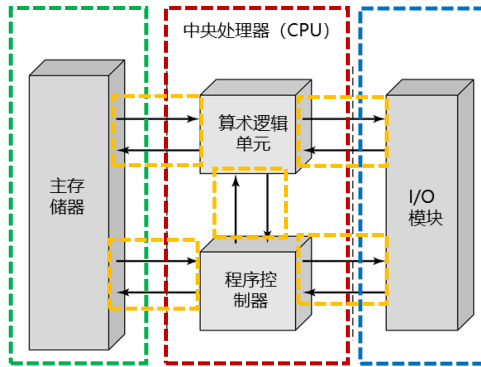


基本功能

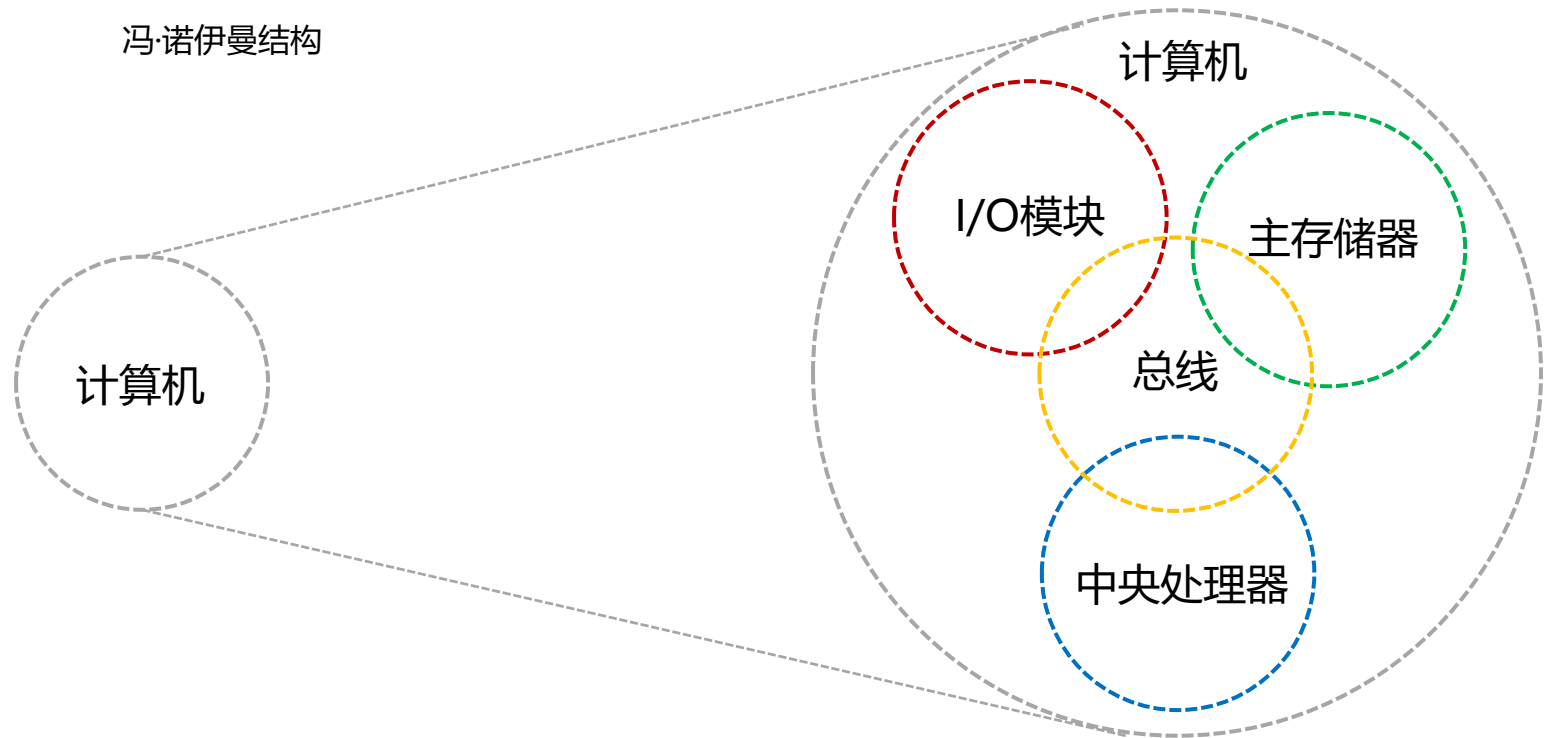


冯·诺伊曼结构

# 计算机顶层结构

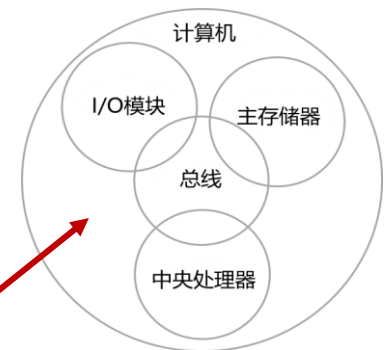
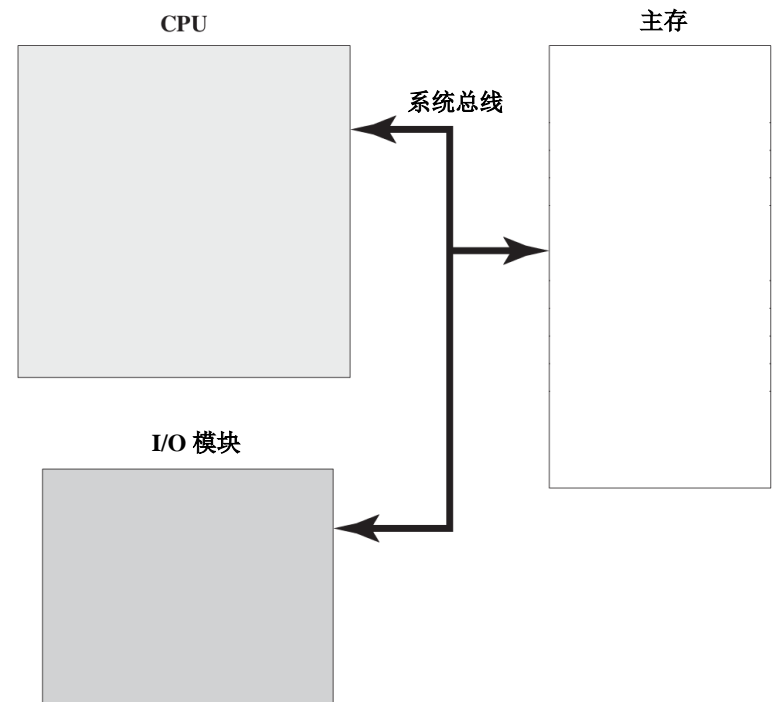


冯·诺伊曼结构



# 计算机的工作原理

- 指令和数据存储在单个读写存储器中
- 主存中的内容按位置访问, 无需考虑其中包含的类型
- CPU从一条指令到下一条指令以顺序方式执行 (除非明确修改)
- I/O模块与 CPU、主存交换计算机系统外部的数据



不成比例扩展效应  
(Incommensurate Scaling)

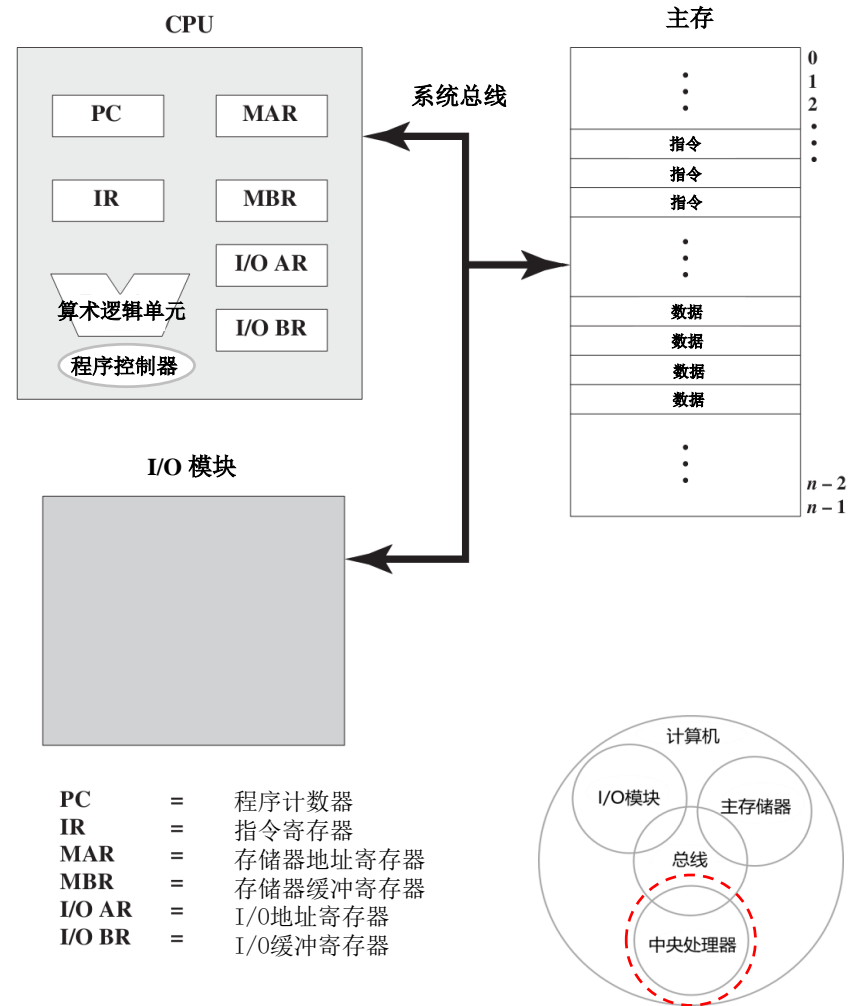


# 回顾: CPU



# 计算机组件: CPU

- CPU从一条指令到下一条指令以顺序方式执行（除非明确修改）
- 指令和数据存储在单个读写存储器中
- 主存中的内容按位置访问，无需考虑其中包含的类型



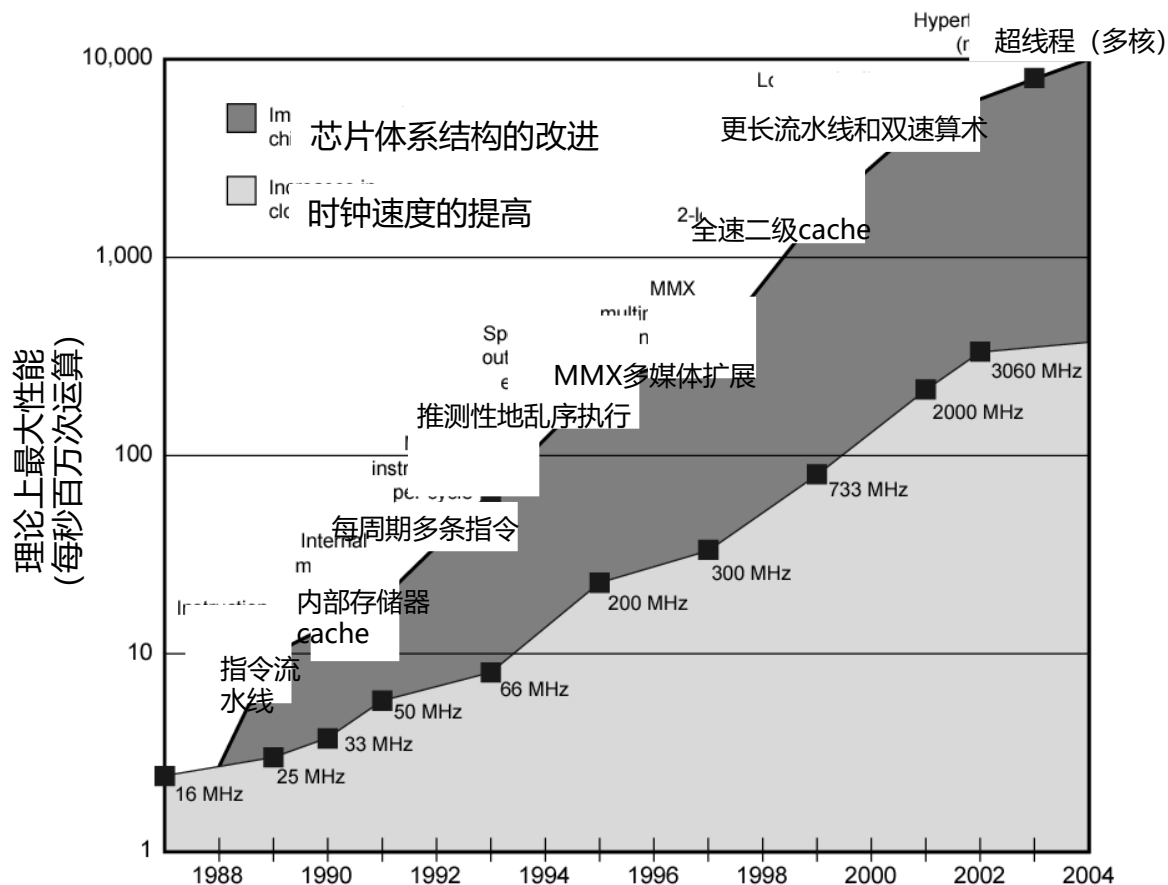
# 问题1：CPU的频率不能无限提高

- 理论限制
  - mos管开关、脉冲通过门电路需要时间
  - 为了信号同步，每个脉冲信号需要持续一定的时间
  - .....
- 制造限制
  - 芯片面积越来越大，导致连线延迟越来越大，需要保证信号在设计指定时钟周期内从芯片的一角到达另一角
  - 频率越高（即mos管的开关频率也越高）会导致开关损耗也越高，cpu会费电和散热高





# 解决1：改进CPU芯片结构

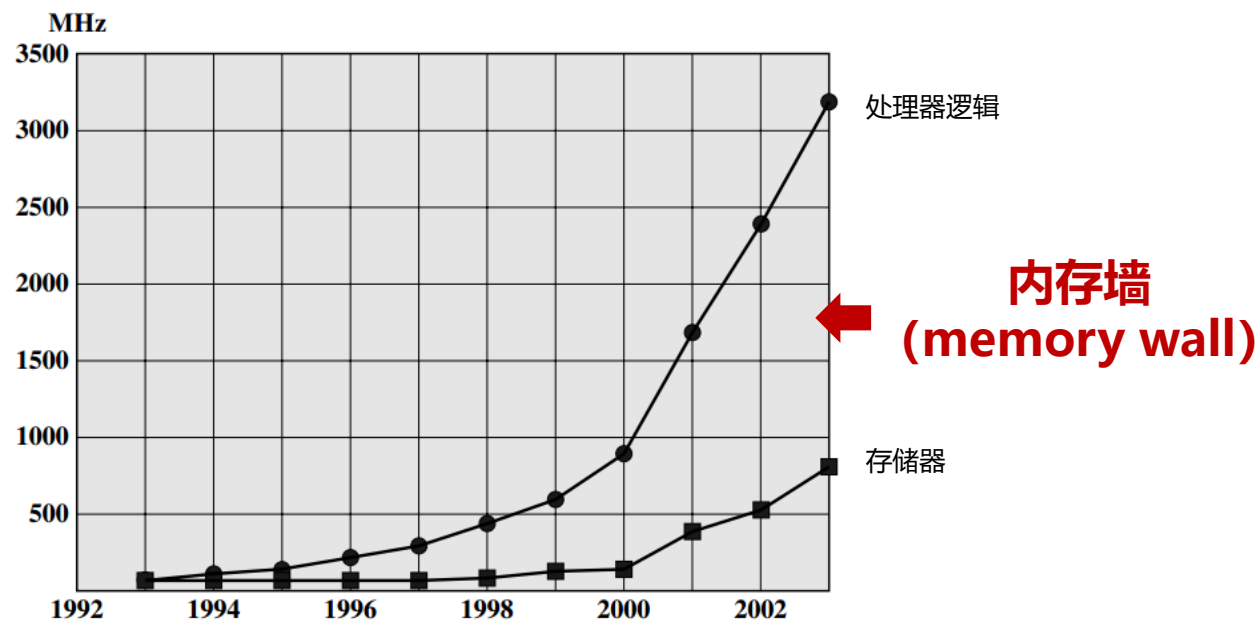


- ✓ 第14讲：CPU结构和功能
- ✓ 第15讲：控制器



# 问题2：内存墙的存在

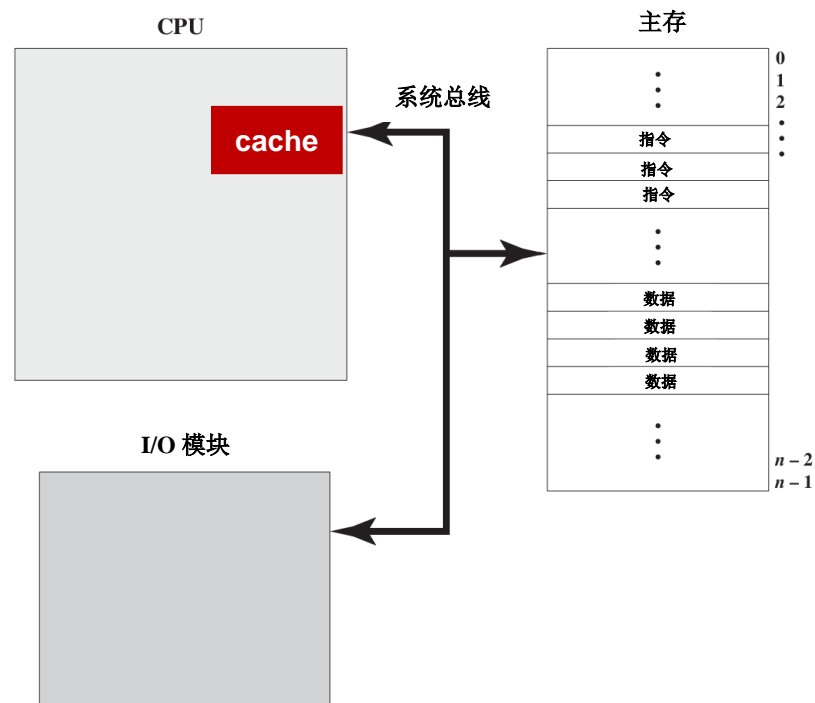
- 问题
  - 主存和CPU之间传输数据的速度跟不上CPU的速度



# 解决2：采用高速缓存（Cache）

- 解决方法

- 添加一级或多级缓存以减少存储器访问频率并提高数据传输速率
- 增大总线的数据宽度，来增加每次所能取出的位数
- ...

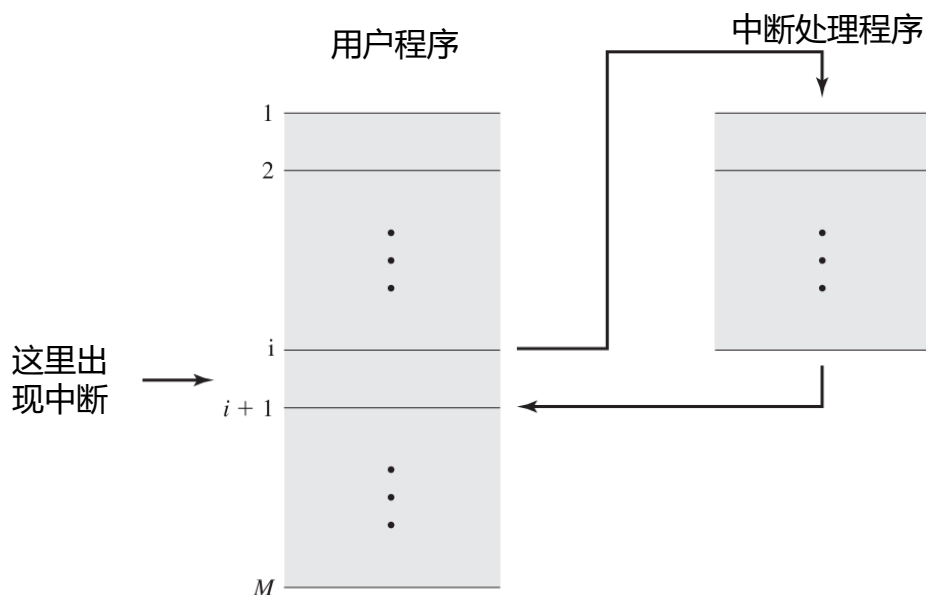


✓ 第08讲：高速缓冲存储器（Cache）



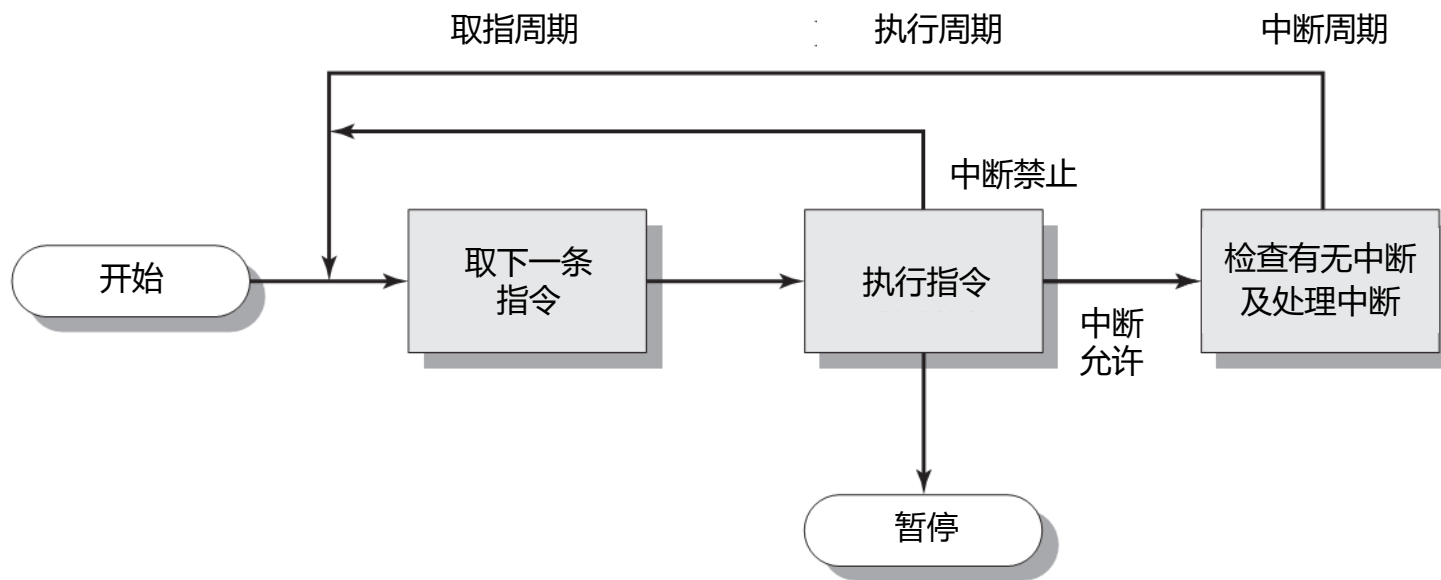
# 问题3：CPU等待I/O传输数据

- 问题
  - CPU 在等待 I/O 设备时保持空闲
- 解决方法
  - **中断**：其他模块（例如 I/O）可以中断正常处理顺序的机制



# 解决3：采用中断机制

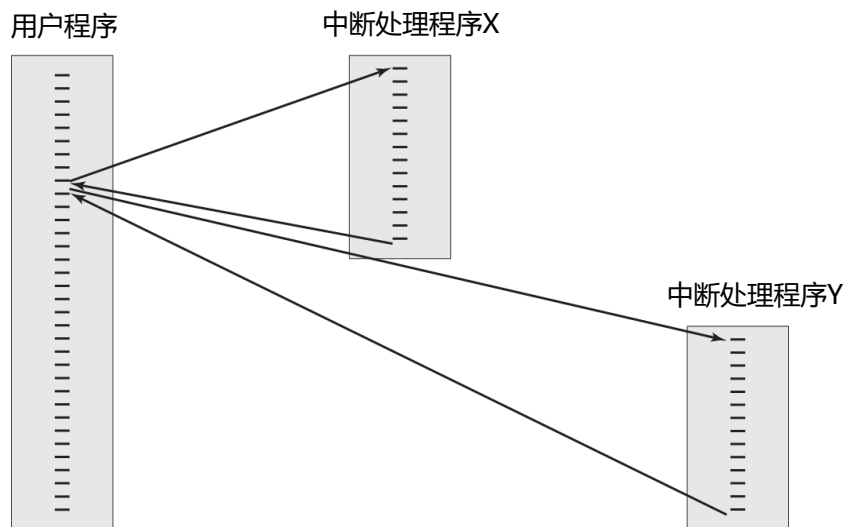
- 中断检测
  - 将中断周期加入指令周期



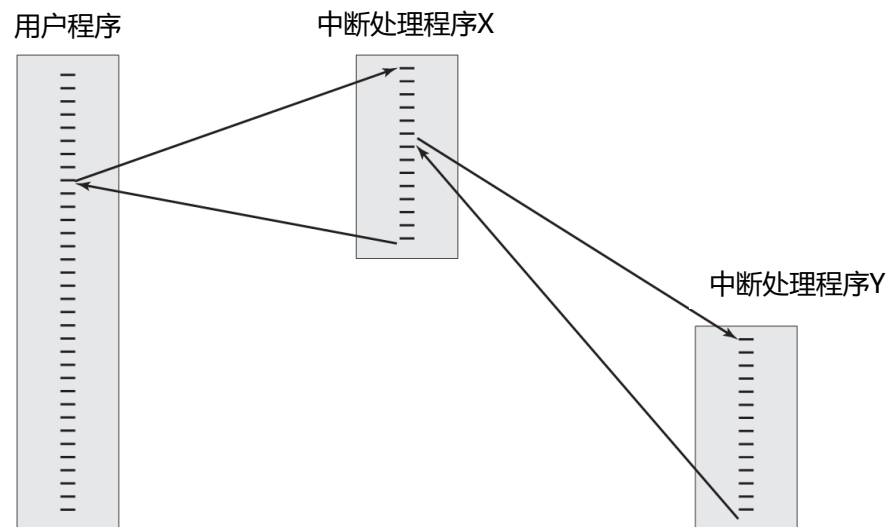
✓ 第13讲：指令集



# 多重中断



顺序中断处理



嵌套中断处理

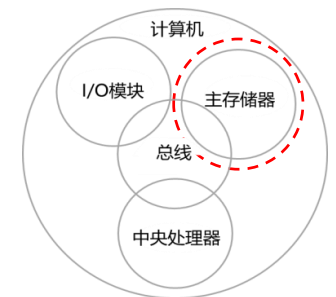
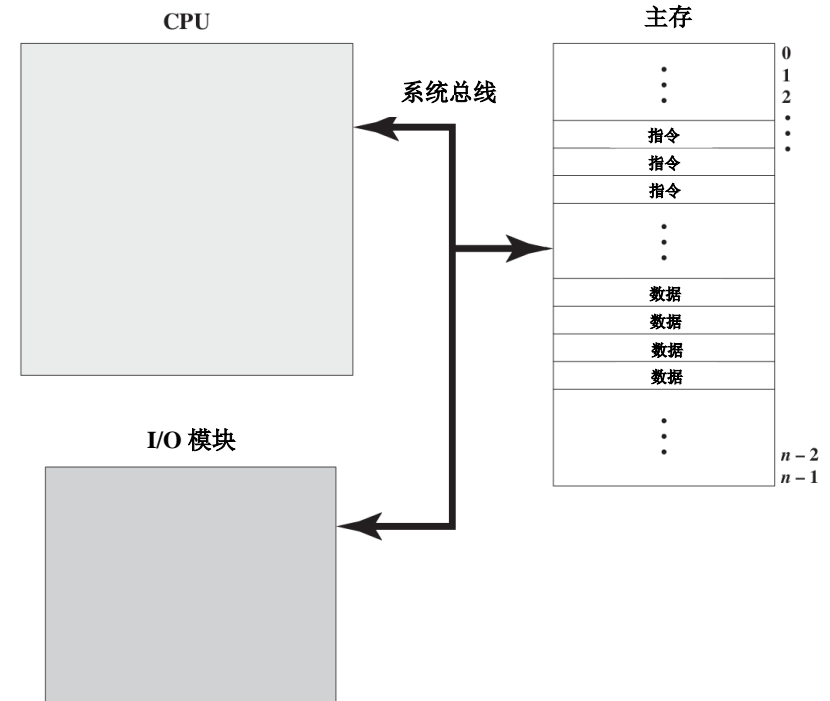


# 回顾: 存储器



# 计算机组件: 存储器

- 指令和数据存储在单个读写存储器中
- 主存中的内容按位置访问, 无需考虑其中包含的类型



✓ 第13讲: 指令集





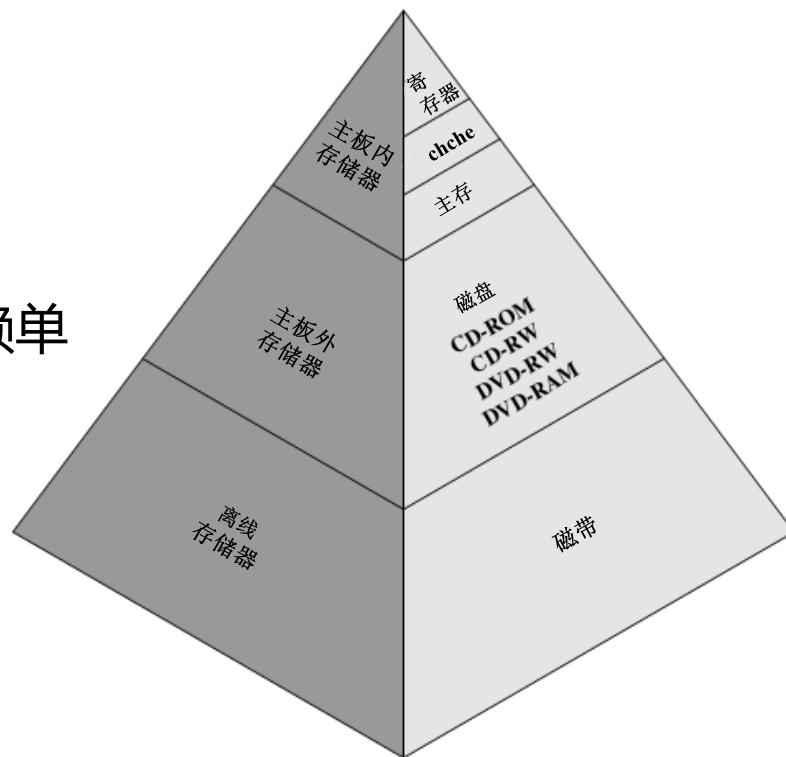
# 问题4：兼顾存储容量、速度和成本

- 约束
  - 容量：越大越好
  - 速度：跟上处理器
  - 成本：相对于其他组件合理
- 约束之间的关系
  - 更短的访问时间，更高的每比特成本



# 解决4：层次式存储结构

- 需求
  - 大容量数据存储
  - 高速性能
- 解决方案
  - 使用存储器层次结构而不是依赖单个存储器组件



- ✓ 第07讲：内部存储器
- ✓ 第09讲：外部存储器
- ✓ 第10讲：数据校验码
- ✓ 第11讲：磁盘冗余阵列 (RAID)
- ✓ 第12讲：虚拟存储器

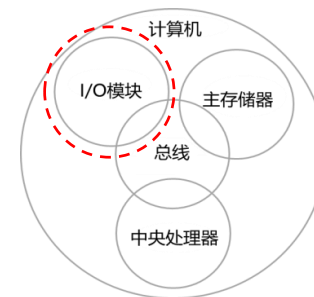
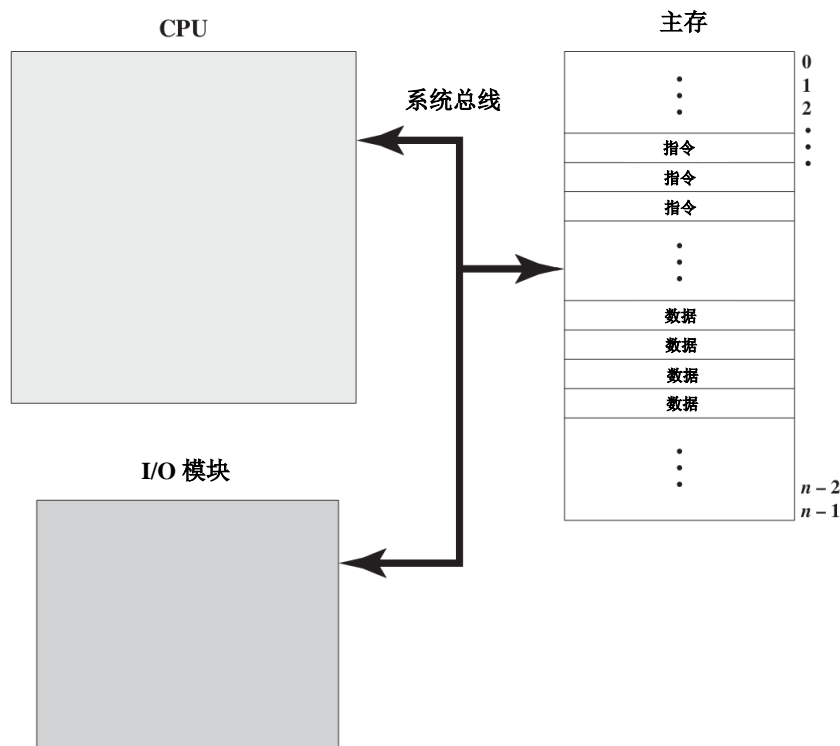


# 回顾: I/O



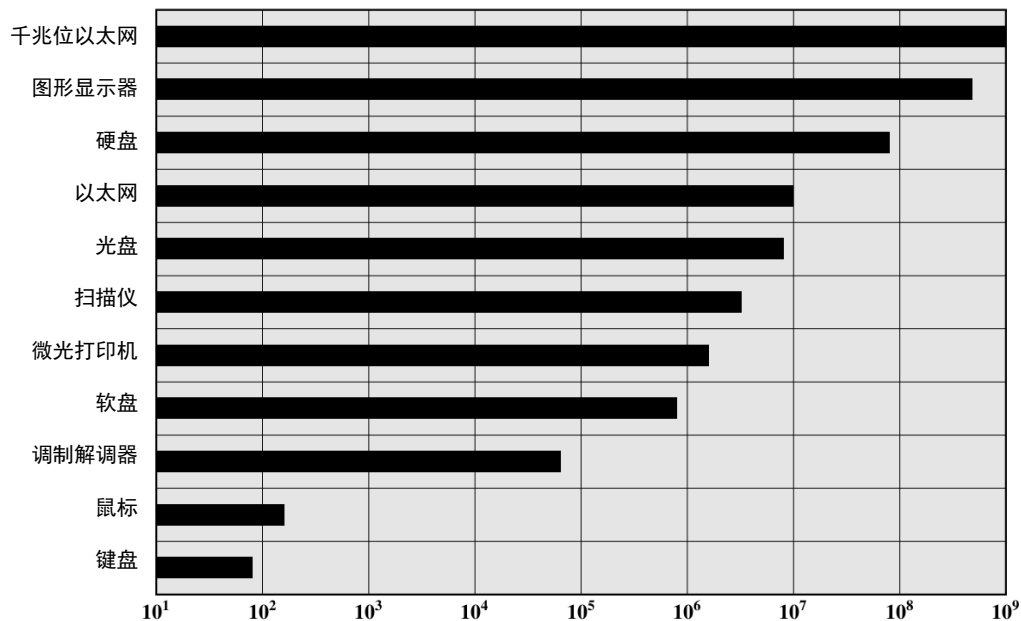
# 计算机组件: I/O模块

- 与 CPU 和内存交换从外部来源收集的数据



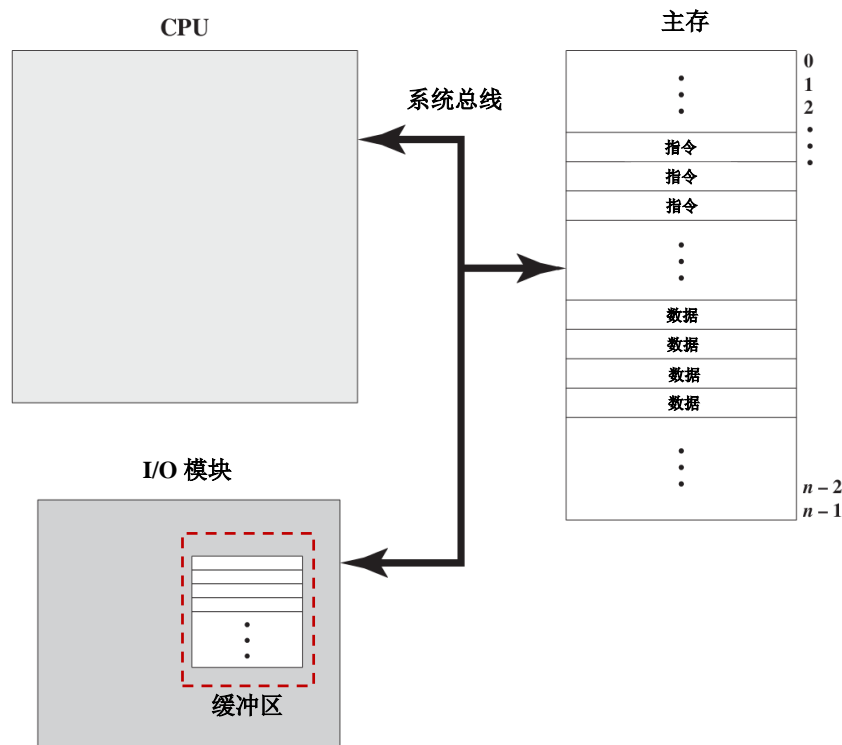
# 问题5：I/O设备传输速率差异大

- 问题
  - I/O性能跟不上CPU速度的提升



# 解决5：采用缓冲区和改进I/O操作技术

- 解决方法
  - 设立缓冲区
  - 新的接口技术
  - 不同的I/O操作技术
  - ...

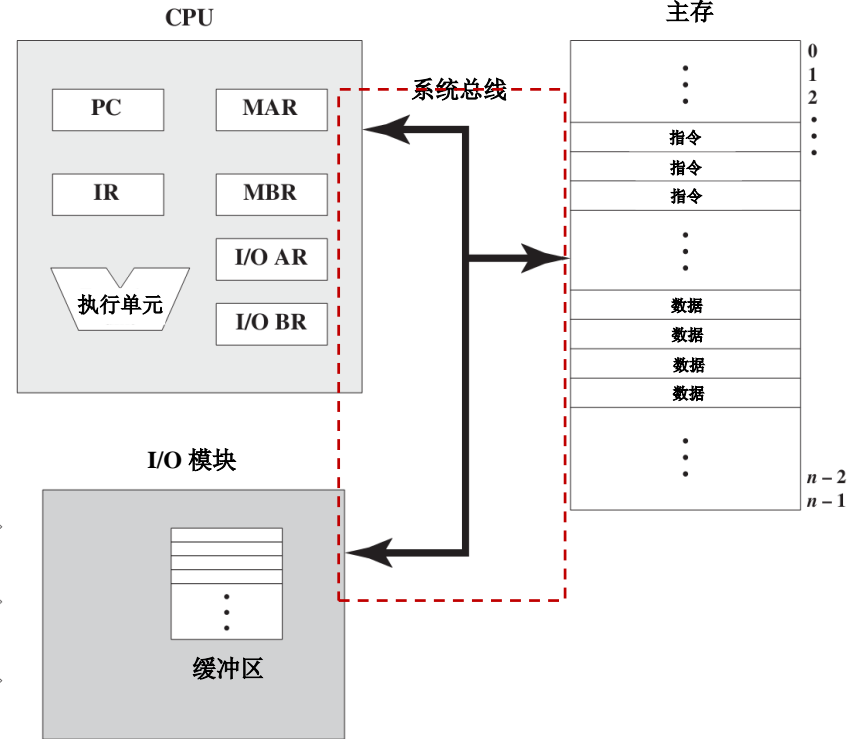
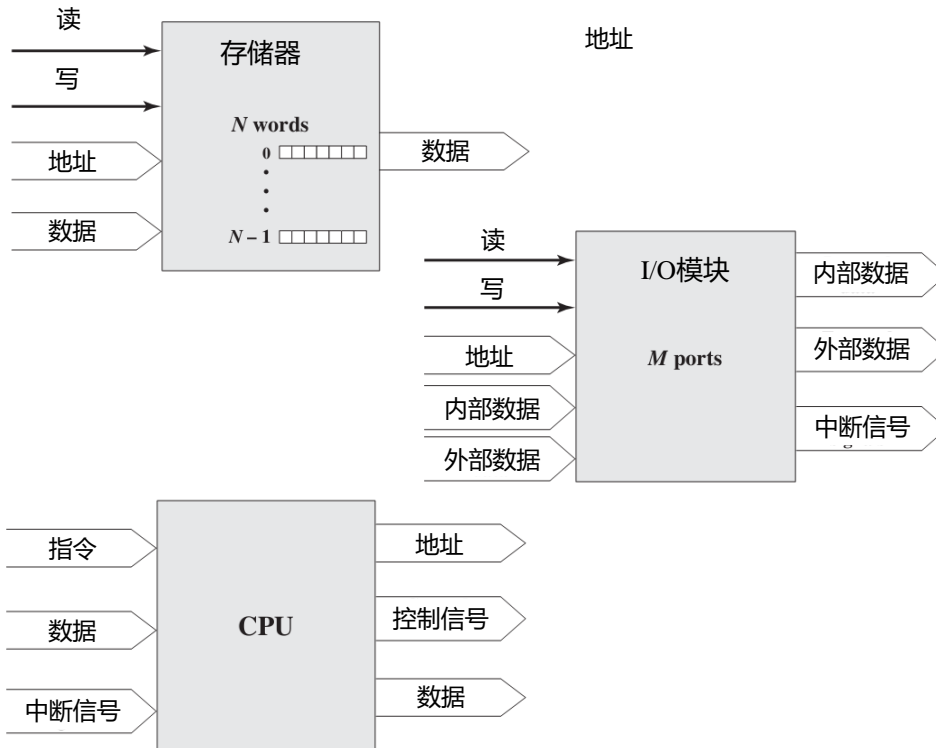


# 回顾: 总线

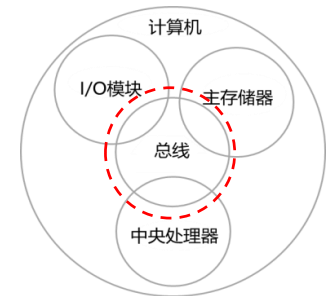


# 计算机组件: 总线

- 总线是连接两个或多个设备的通信通路



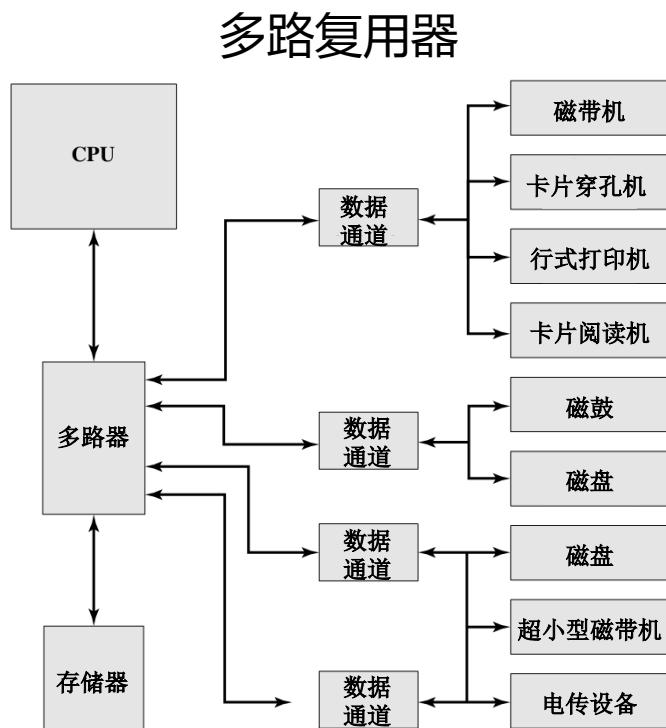
PC	=	程序计数器
IR	=	指令寄存器
MAR	=	存储器地址寄存器
MBR	=	存储器缓冲寄存器
I/O AR	=	I/O地址寄存器
I/O BR	=	I/O缓冲寄存器





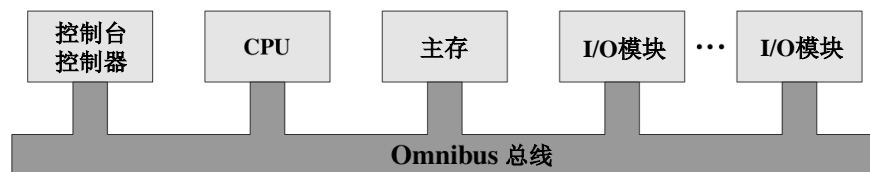
# 问题6：计算机部件互连复杂

- 互连方案



IBM 7094

## 总线

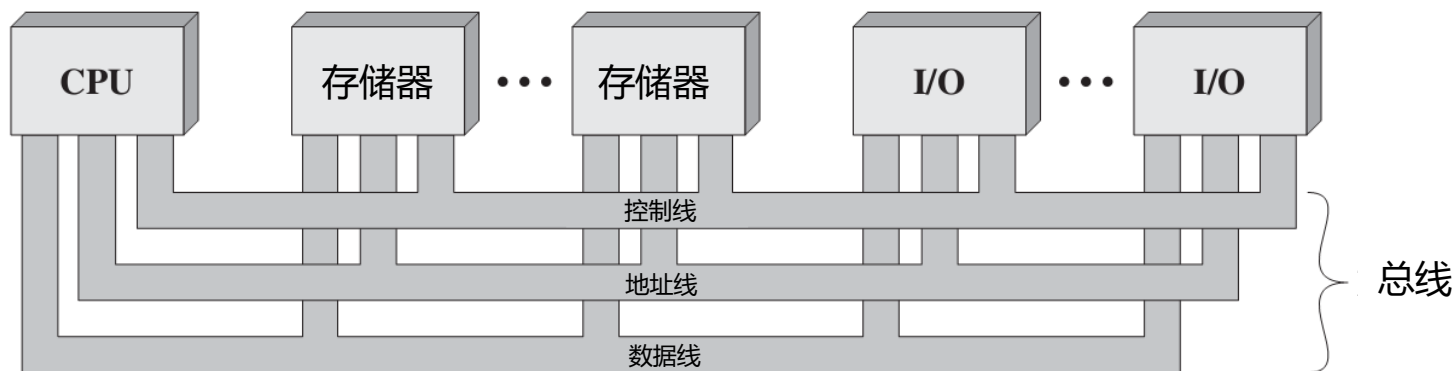


PDP-8

**共享传输介质**  
**简化互连布局和处理器控制**

# 解决6：采用总线

- 数据传输类型
  - 控制线：控制数据线和地址线的访问和使用
  - 地址线：指定数据总线 and 地址I/O端口上数据的来源或去向
  - 数据线：在系统模块之间传送数据



# 总结

- 计算机的顶层视图
  - 基本功能，冯·诺伊曼结构
- 计算机体系结构遇到的问题及解决方案
  - CPU的频率不能无限提高 → 改进CPU芯片结构
  - 内存墙的存在 → 采用高速缓存（Cache）
  - CPU等待I/O传输数据 → 采用中断机制
  - 兼顾存储容量、速度和成本 → 层次式存储结构
  - I/O设备传输速率差异大 → 采用缓冲区和改进I/O操作技术
  - 计算机部件互连复杂 → 采用总线



# 谢谢

rentw@nju.edu.cn



南京大學  
NANJING UNIVERSITY