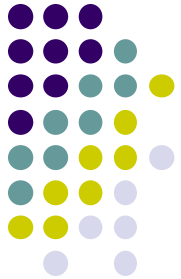


# 计算机组成与系统结构

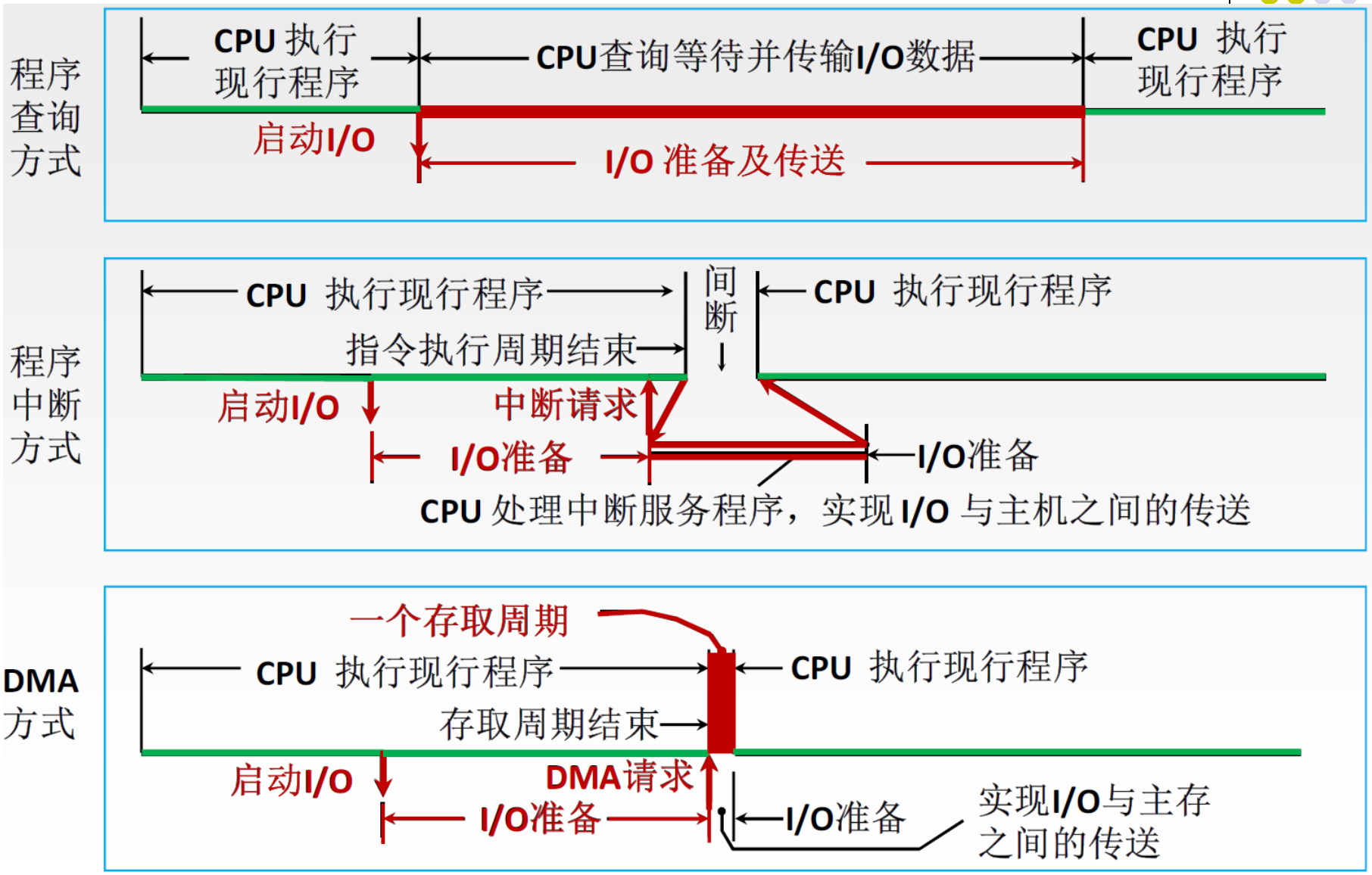
# 第八章 输入输出系统

吕昕晨

lvxinchen@bupt.edu.cn

网络空间安全学院

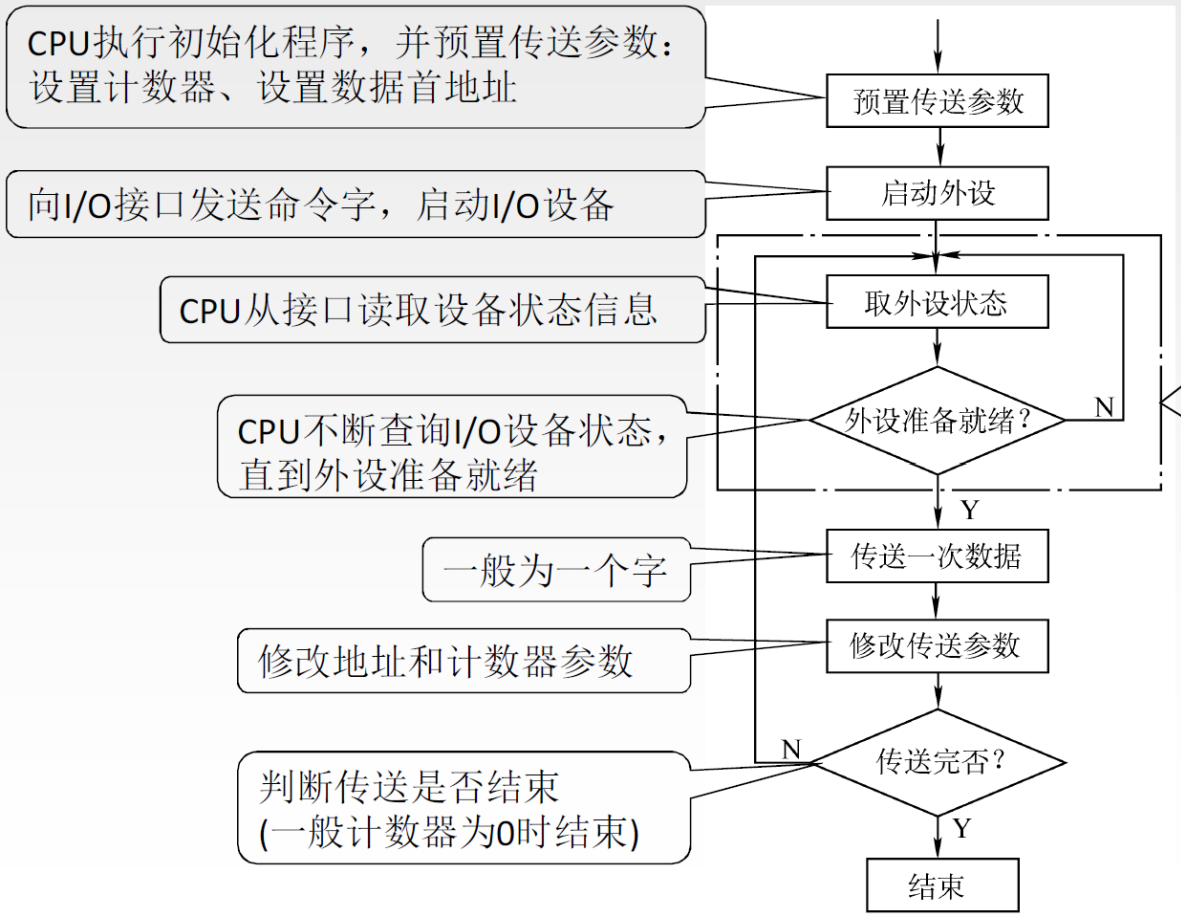
# 输入输出方式总结


程序查询方式
CPU 执行现行程序
CPU查询等待并传输I/O数据
CPU 执行现行程序
启动I/O
I/O 准备及传送
程序中断方式
CPU 执行现行程序
指令执行周期结束
间断
CPU 执行现行程序
启动I/O
中断请求
I/O准备
I/O准备
CPU 处理中断服务程序，实现 I/O 与主机之间的传送
DMA方式
一个存取周期
CPU 执行现行程序
CPU 执行现行程序
存取周期结束
启动I/O
DMA请求
I/O准备
I/O准备
实现I/O与主存之间的传送

# 第八章 输入输出系统

- 程序查询方式
- 程序中断方式
- DMA方式

# 程序查询流程

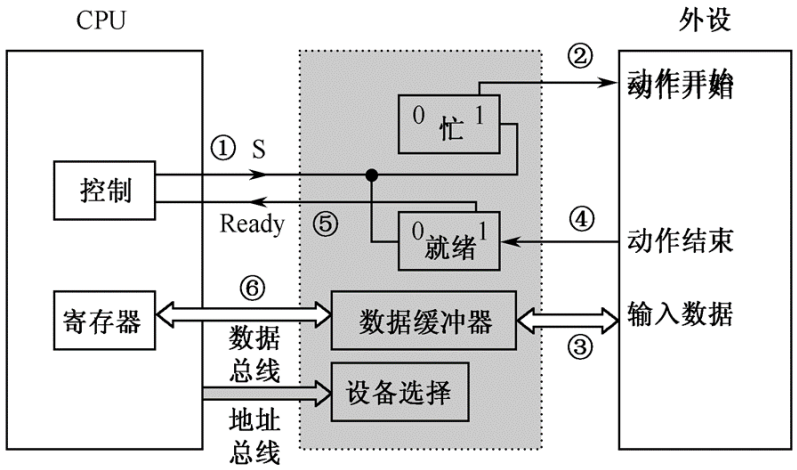



CPU一旦启动I/O，必须停止现行程序的运行，并在现行程序中插入一段程序。
主要特点：CPU有“踏步”等待现象，CPU与I/O串行工作。

优点：接口设计简单、设备量少。
缺点：CPU在信息传送过程中要花费很多时间用于查询和等待，而且在一段时间内只能和一台外设交换信息，效率大大降低。

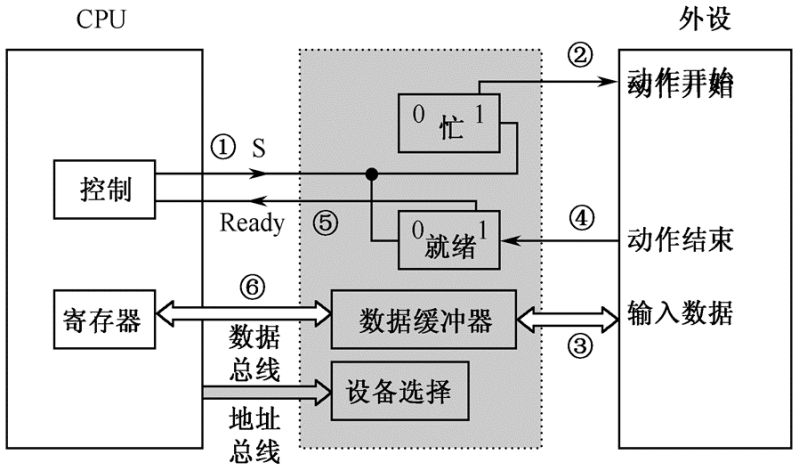
# 输入输出指令

- 输入输出指令一般功能：
  - 置1或置0
    - I/O接口的某些控制触发器，用于控制设备进行某些动作
    - 如启动、关闭设备等
  - 测试设备的某些状态
    - 如“忙”、“准备就绪”等，以便决定下一步的操作
  - 传送数据
    - 输入/输出数据



# 程序查询接口

- 设备选择电路
  - 每个设备接口电路都包含一个设备选择电路，用它判别地址总线上呼叫的设备是不是本设备
- 数据缓冲寄存器
  - 实现CPU与外设之间数据输入输出操作的缓冲，实现速率匹配
- 设备状态标志
  - 是接口中的标志触发器，如“忙”、“准备就绪”、“错误”等，用来标志设备的工作状态，以便接口对外设动作进行监视



# 多设备程序查询流程

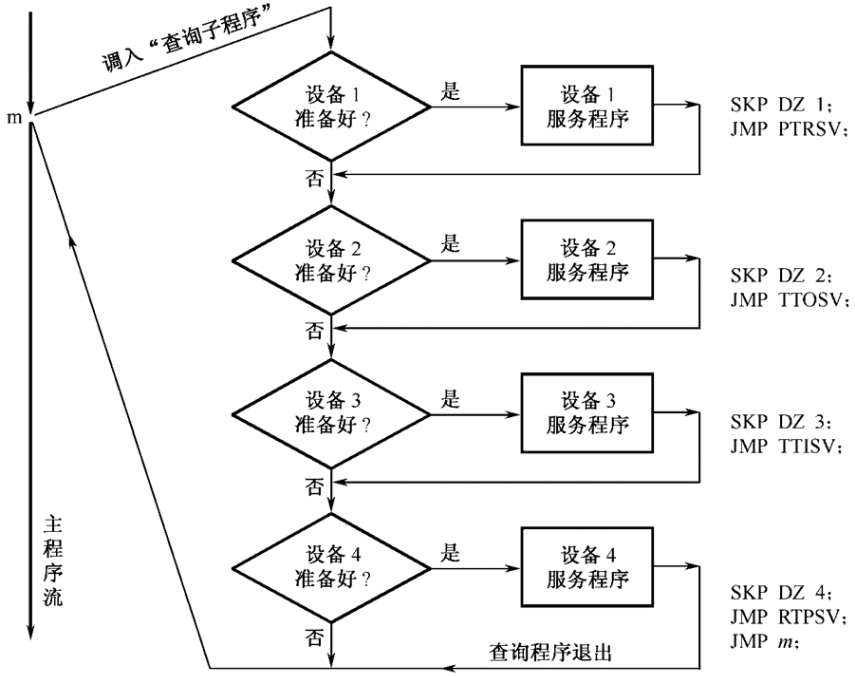
（1）先向I/O设备发出命令字，请求进行数据传送。

（2） 从I/O接口读入状态字。

（3） 检查状态字中的标志，看看数据交换是否可以进行。

（4） 假如这个设备没有准备就绪，则第（2）、第（3）步重复进行，直到这个设备发出准备就绪信号“Ready”为止。

（5） 数据传输：CPU从I/O接口的数据缓冲寄存器输入数据，或从CPU输出至接口的数据缓冲寄存器。



# 程序查询例题

程序查询系统中，假设不考虑处理时间，每一个查询操作需要100个时钟周期，CPU频率为50MHz。现有鼠标和硬盘两个设备，CPU每秒需对鼠标30次查询，硬盘以32位字长为单位传输数据，每32位被CPU查询一次，传输速率为$2*2^{20}$B/s。

求CPU对这两个设备查询所花费时间及占CPU比例

# 程序查询例题

1）鼠标：

每秒耗时：30*100*20(ns)=60000ns

时间比率：60000ns/1s=0.006%

2）硬盘

硬盘查询次数：$2*2^{20}/4=2^{19}$次/秒

查询硬盘耗时：$2^{19}*2000ns=1.05s$

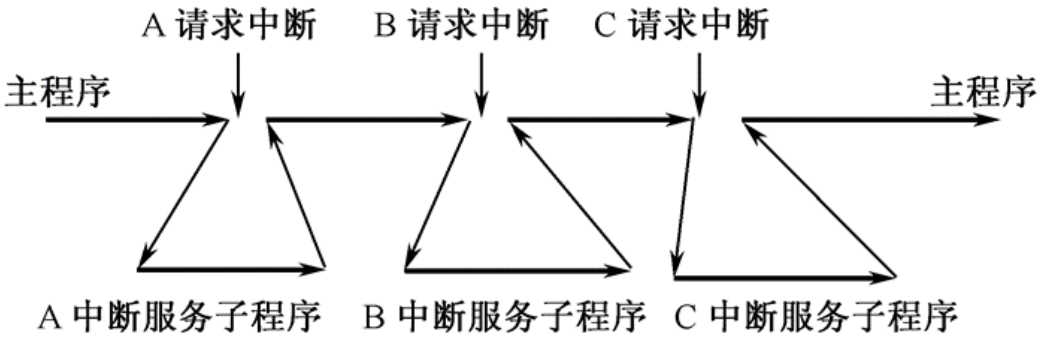
时间比率：1.05/1=105%

# 第八章 输入输出系统

- 程序查询方式
- 程序中断方式
  - 中断基本概念与I/O接口
  - 单级中断、多级中断、中断控制器
- DMA方式

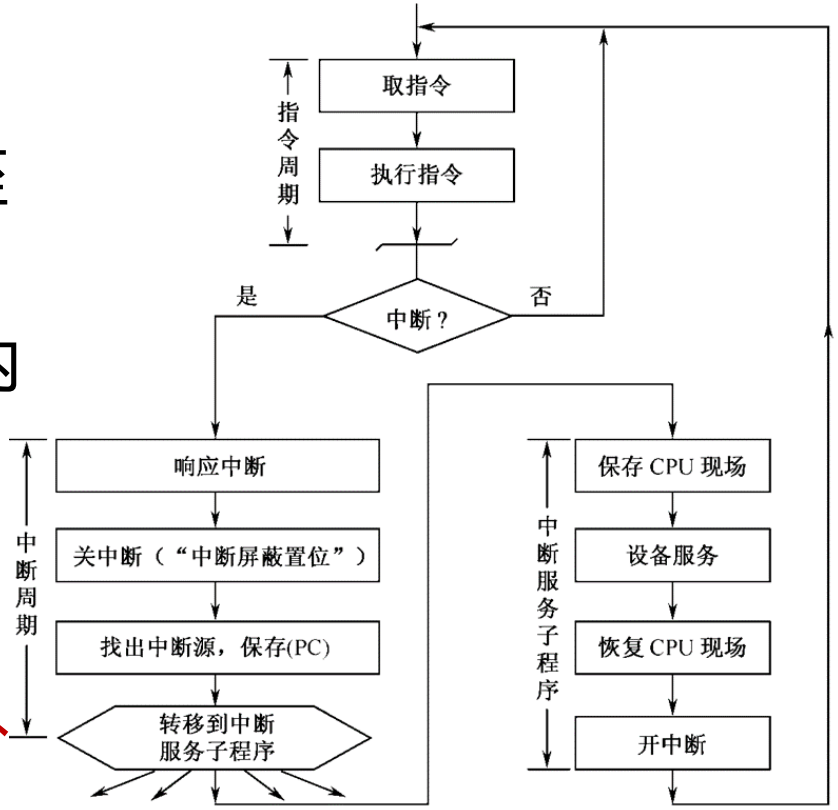
# 中断的基本概念

- 中断（Interrupt）是指CPU暂时中止现行程序，转去处理随机发生的紧急事件，处理完后自动返回原程序的功能和技术。
- 中断系统是计算机实现中断功能的软硬件总称。一般在CPU中设置中断机构，在外设接口中设置中断控制器，在软件上设置相应的中断服务程序。



# 中断处理基本流程

- 中断处理过程注意几个问题：
  - 响应中断时机：CPU只有在当前机器指令执行完毕后，转至公操作，处理中断
  - 断点保护问题（PC，寄存器内容和状态的保存）
  - 中断屏蔽：开中断和关中断
  - 中断是软硬件结合实现的
  - 中断分为内中断（异常）和外中断

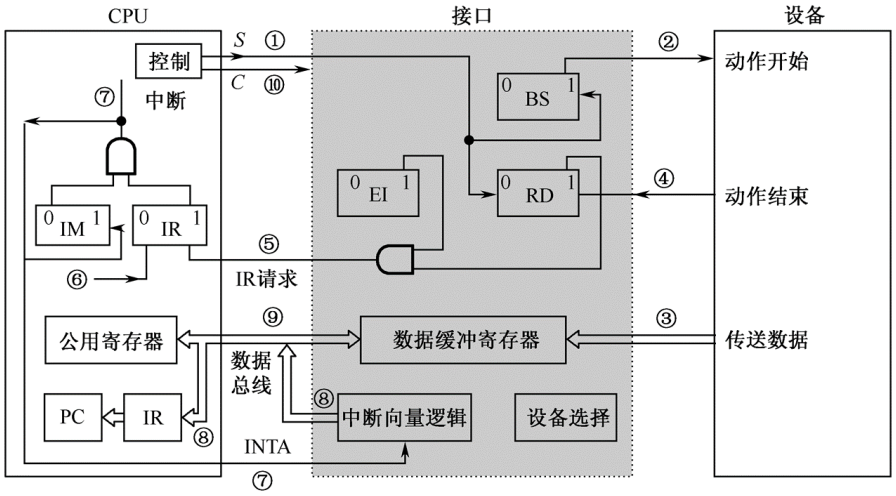


# 中断典型应用

- 实现CPU与外界进行信息交换的握手联络
  - 中断可以实现CPU与外设的并行工作
  - 对于慢速I/O设备，使用中断方式可以有效提高CPU的效率
- 故障处理
  - 用于处理常见的硬件故障，如掉电、校验错、运算出错等；处理常见的软件故障，如溢出、地址越界、非法指令等。
- 实时处理
  - 中断可以保证在事件出现的实际时间内及时地进行处理
- 程序调度
  - 中断是操作系统进行多任务调度的手段
- 软中断（程序自愿中断）
  - 软中断不是随机发生的，而是与子程序调用功能相似，但其调用接口简单，不依赖于程序入口地址，便于软件的升级维护和调用

# 中断基本I/O接口

- 接口方面
  - 设备选择器：判别总线上送出的地址（或称呼叫的设备）是否为本设备
  - BS外设接口忙（BuSy）标志
  - RD外设准备就绪（ReaDy）标志
  - EI（Enable Interrupt中断允许触发器）
- CPU方面
  - IR（Interrupt Request）中断请求触发器
  - IM（Interrupt Mask）中断屏蔽触发器

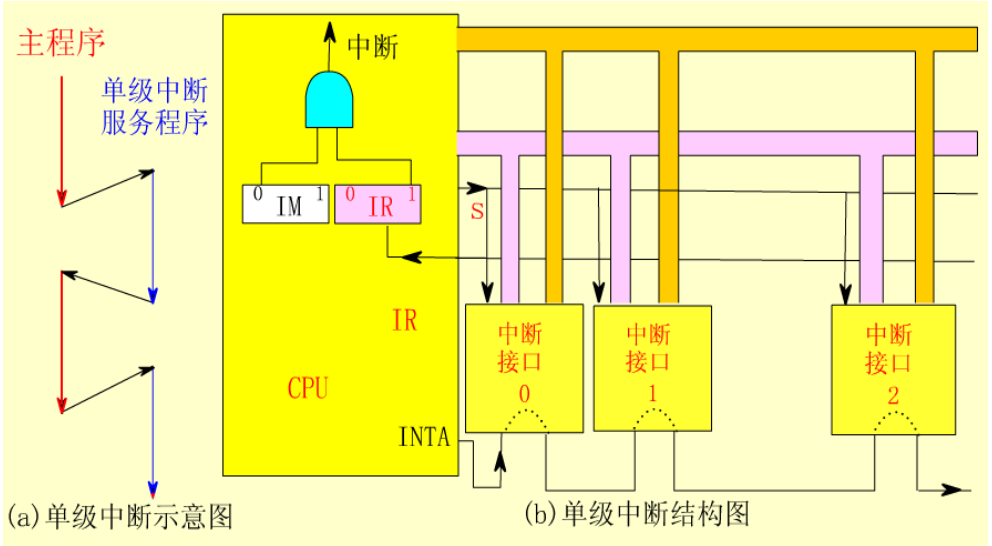


# 第八章 输入输出系统

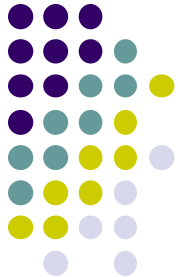
- 程序查询方式
- 程序中断方式
  - 中断基本概念与I/O接口
  - 单级中断、多级中断、中断控制器
- DMA方式

# 单级中断

- 单级中断的概念
  - 所有中断源属于同一级（不允许嵌套），离CPU越近，优先级越高
- 中断源的识别
  - 采用串行排队链法（对比总线仲裁）
    - IR为中断请求信号
    - INTA为中断相应信号
    - IS1~3为中断选中信号



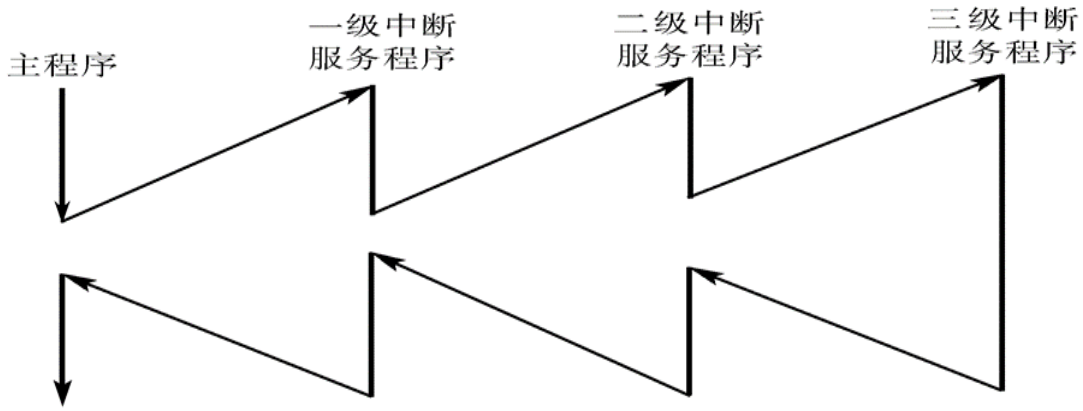
# 中断向量号



- 中断向量
  - 每个中断源分别有一个中断服务程序，而每个中断服务程序又有自己的向量地址
  - 当CPU响应中断时，由硬件直接产生一个与该中断源对应的向量地址
  - 由向量地址指出每个中断源设备的中断服务程序入口，这种方法通常称为向量中断

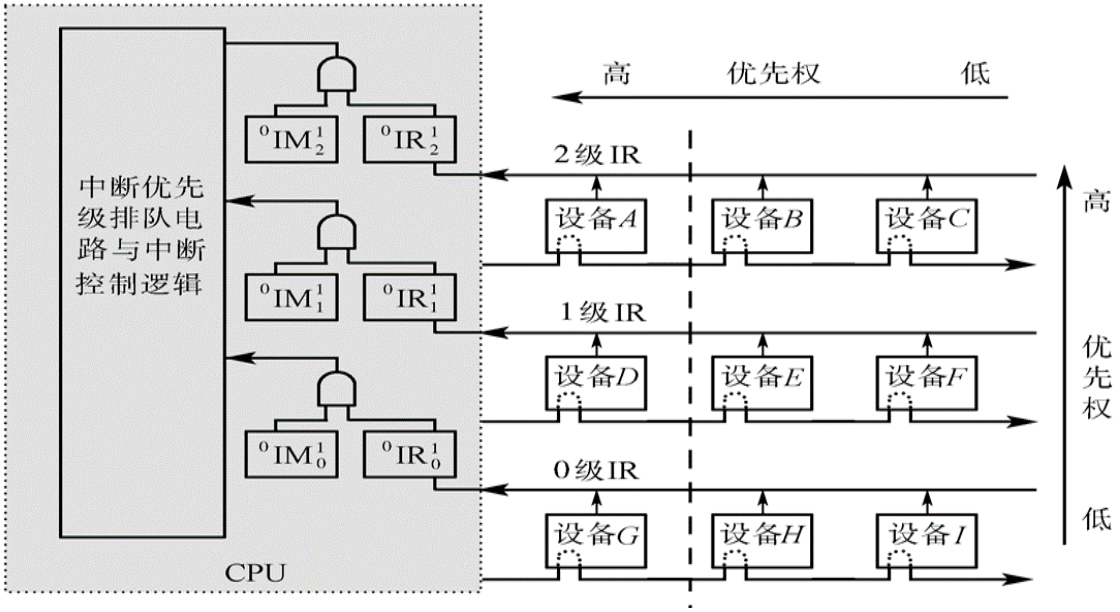
# 多级中断

- 多级中断概念
  - 根据各中断事件的轻重缓急程度不同而分成若干级别，每一中断级分配给一个优先权
  - 一般说来，优先权高的中断级可以打断优先权低的中断服务程序，以程序嵌套方式进行工作
  - 可分为一维多级中断和二维多级中断



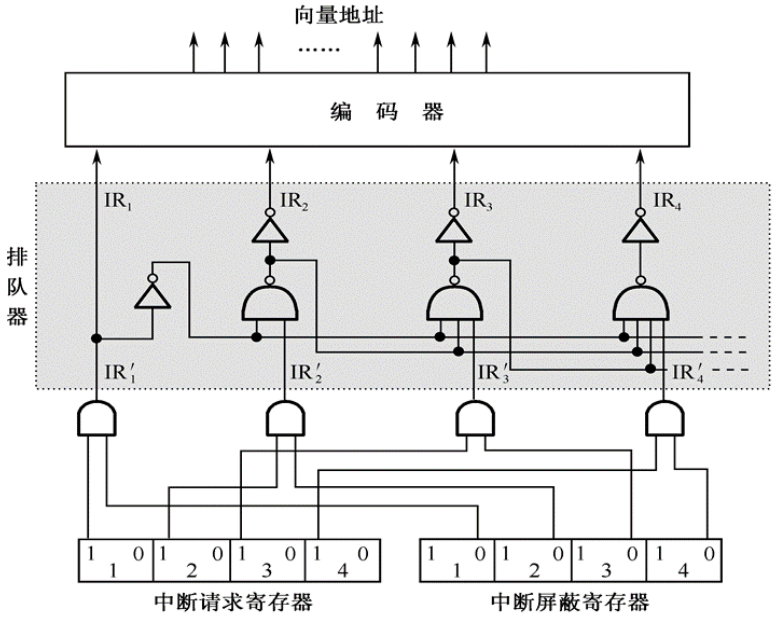
# 多级中断结构

- 一个系统有n级中断，则CPU中有n个IR，n个IM，某级中断被响应后，则关闭本级和低于本级的IM，开放更高级的IM
- 多级中断可以嵌套，但同一级的中断不允许嵌套
- 中断响应时，确定哪一级中断和中断源采用硬件实现。采用了独立请求方式和链式查询方式相结合的方式。
- 使用多级堆栈保存现场



# 多级中断识别

- 中断请求放入中断请求寄存器
- 优先级排序（排队器）
  - 请求源1：优先级最高
  - 请求源4：优先级最低
- 中断屏蔽寄存器
  - 决定是否响应对应请求
- 编码器：根据中断源产生中断向量号
- 例如
  - 中断请求寄存器(IR)：1111
  - 中断屏蔽寄存器(IM)：0010
  - 排队器输出：1000
  - 编码器产生中断源1对应中断向量号



# 多级中断例题

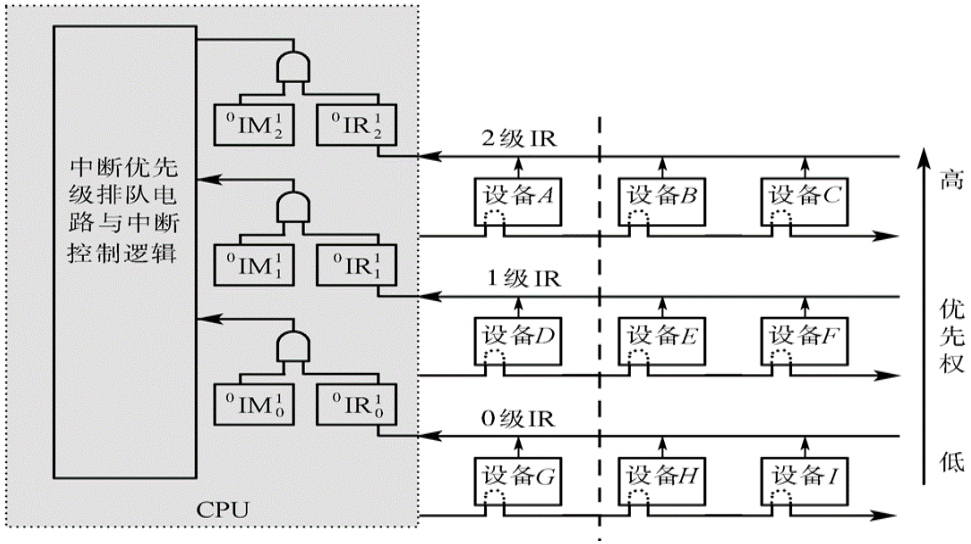
- 二维中断系统如图所示，回答如下问题：

  1）在中断情况下，CPU和设备优先级排序情况

  2）CPU现执行设备B的中断服务程序，IM2-IM0的状态是？如果执行设备D的中断服务程序，IM2-IM0状态是?

  3）IM2-IM0能否实现对具体单个设备进行屏蔽，若想实现，应采用什么方法

  4）若设备C提出中断请求，CPU立即响应，应如何调整



# 多级中断例题

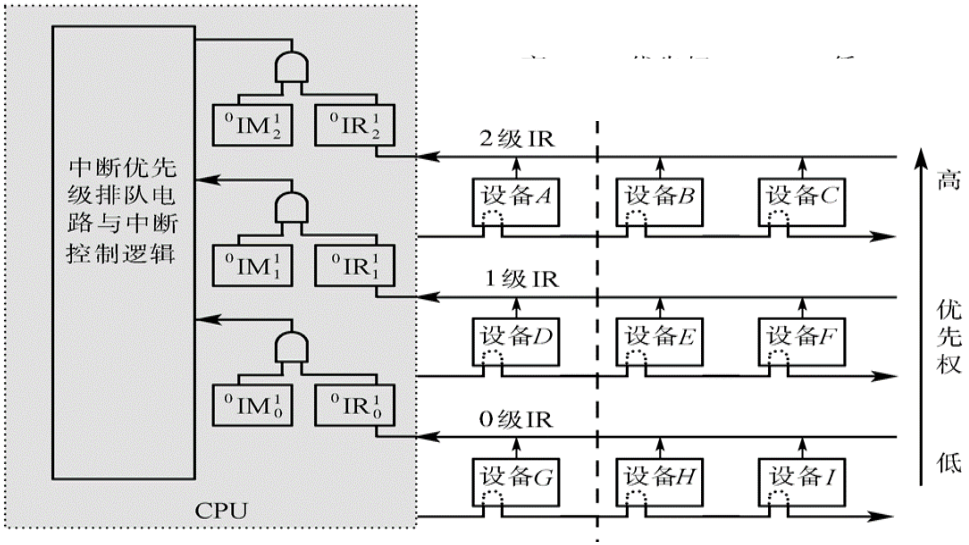
1）在中断情况下，CPU和设备优先级排序情况

CPU优先级最低，设备优先级为A-B-C-D-E-F-G-H-I

2）CPU现执行设备B的中断服务程序，IM2-IM0的状态是？如果执行设备D的中断服务程序，IM2-IM0状态是?

设备B（最高优先级），IM0=1、IM1=1、IM2=1

设备D（次高优先级），IM0=1、IM1=1、IM2=0



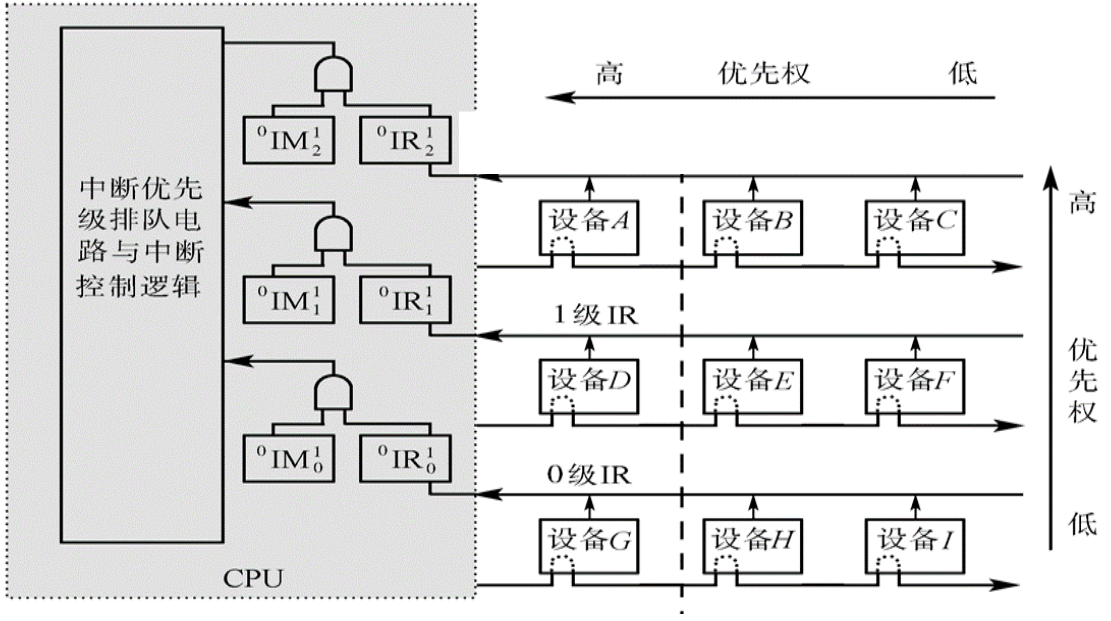
# 多级中断例题

3）IM2-IM0能否实现对具体单个设备进行屏蔽，若想实现，应采用什么方法

不可以，可通过程序设置各设备的接口EI（中断允许）标志

4）若设备C提出中断请求，CPU立即响应，应如何调整

需增加第三级IR，仅将设备C至于第三级IR上，IM3优先级最高



# 第八章 输入输出系统

- 程序查询方式
- 程序中断方式
- DMA方式
  - DMA基本概念与传送方式
  - DMA数据传送流程

# DMA方式的特点

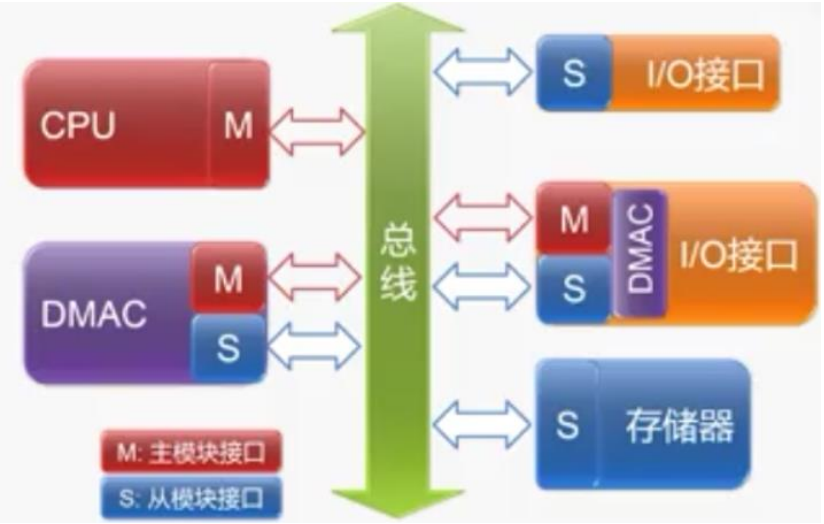
- DMA方式的主要特点
  - DMA方式以响应随机请求的方式，实现主存与I/O设备间的快速数据传送
  - DMA方式不影响CPU的程序执行状态，只要不存在访存冲突，CPU就可以继续执行自己的程序
  - DMA只能处理简单的数据传送，不能在传送数据的同时进行判断和计算
- 基本流程
  - 外围设备DMA请求
  - CPU响应请求，设置DMA方式，DMA接管总线
  - DMA完成数据传输，向CPU汇报DMA传输结束

# 直接内存访问（DMA）

- DMA控制器（DMAC）
  - 数据传送过程不需要CPU干预（CPU继续进行数据计算）
  - 进行外设与存储器间直接数据传送
  - 分类
    - 独立DMA芯片：Intel 8237
    - DMA集成在SoC内部，STM32系列
    - DMA集成在I/O接口内部，固态硬盘等高速外设

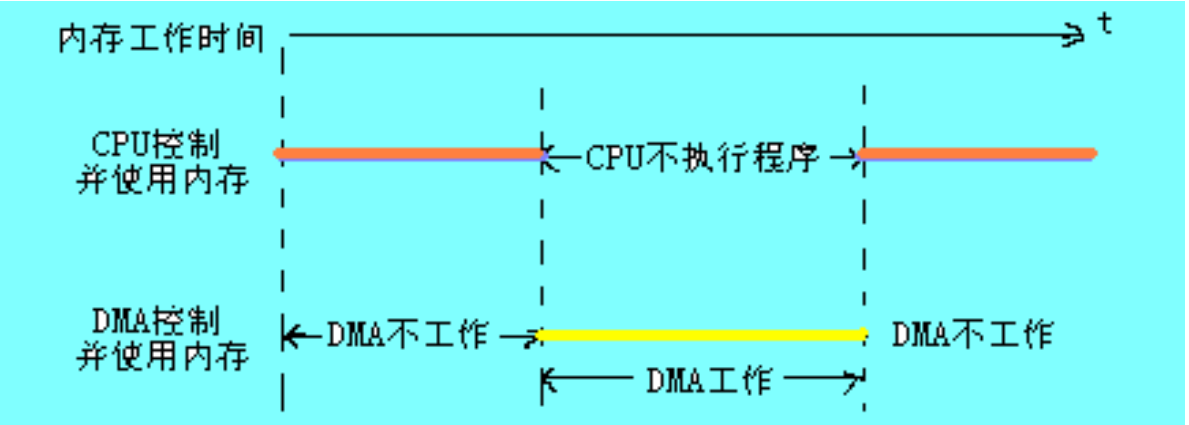
# DMA与CPU的竞争

- DMAC和CPU均具有主模块功能（Master-Mode），复用总线，且可能同时对存储器进行读写
- 问题：争用总线与存储器的使用/读写权限
- 传送分配方式
  - 停止CPU访问内存
  - 周期挪用
  - DMA与CPU交替访内



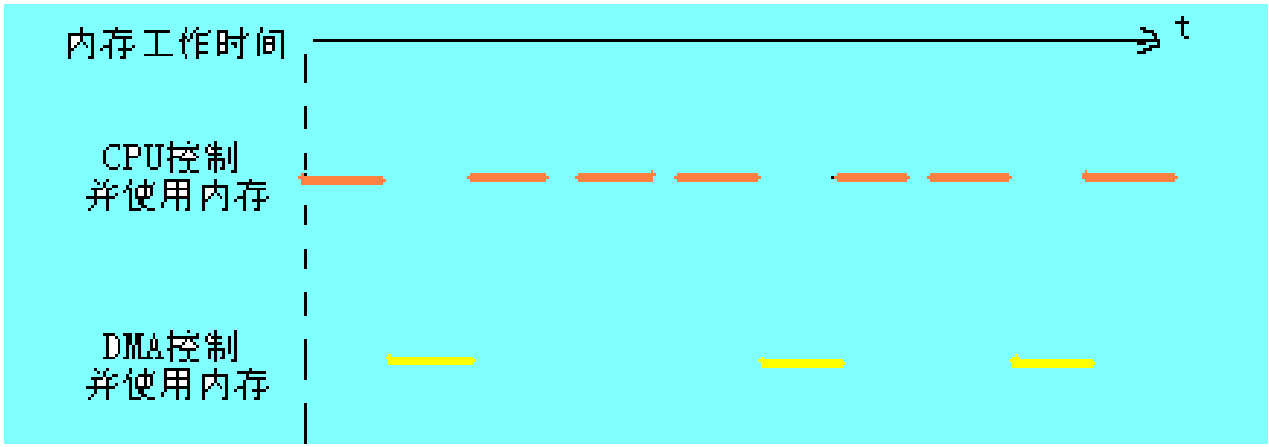
# 方法1：停止CPU访问内存

- 主机响应DMA请求后，让出总线，直到一组数据传送完毕后，DMA控制器才把总线控制权交还给CPU
- 优点
  - 控制简单，适用于数据传输率很高的设备进行成组传送
- 缺点
  - 在DMA控制器访内阶段，一部分内存工作周期是空闲的



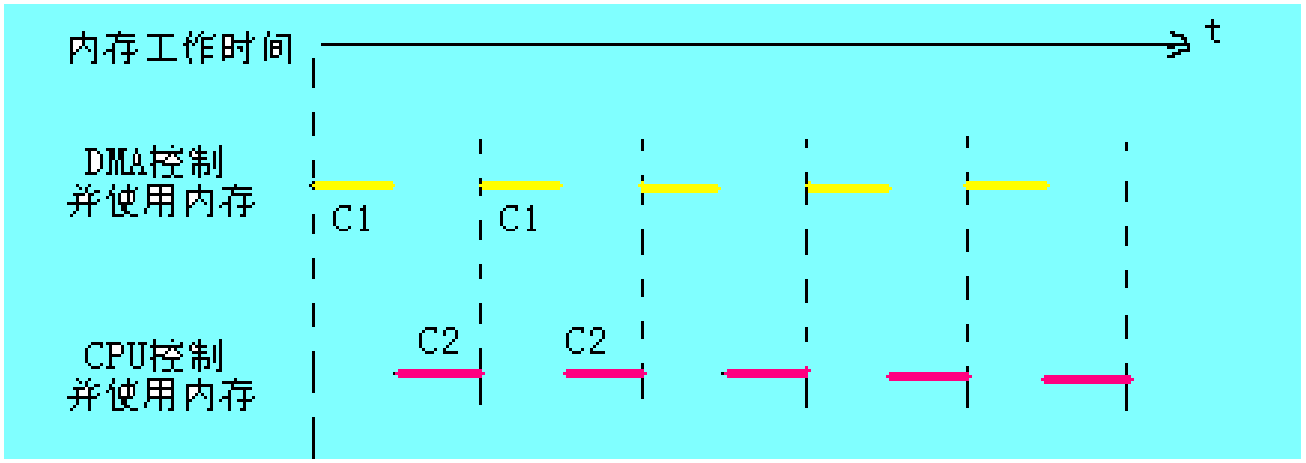
# 方法2：周期挪用方式

- DMA控制器与主存储器之间传送一个数据，占用（窃取）一个CPU周期，即CPU暂停工作一个周期，然后继续执行程序
  - CPU不需要访内，例如CPU正在执行乘法指令。由于乘法指令执行时间较长，此时I/O访内与CPU访内没有冲突
  - I/O设备要求访内时CPU也要求访内，这就产生了访内冲突，在这种情况下I/O设备访内优先



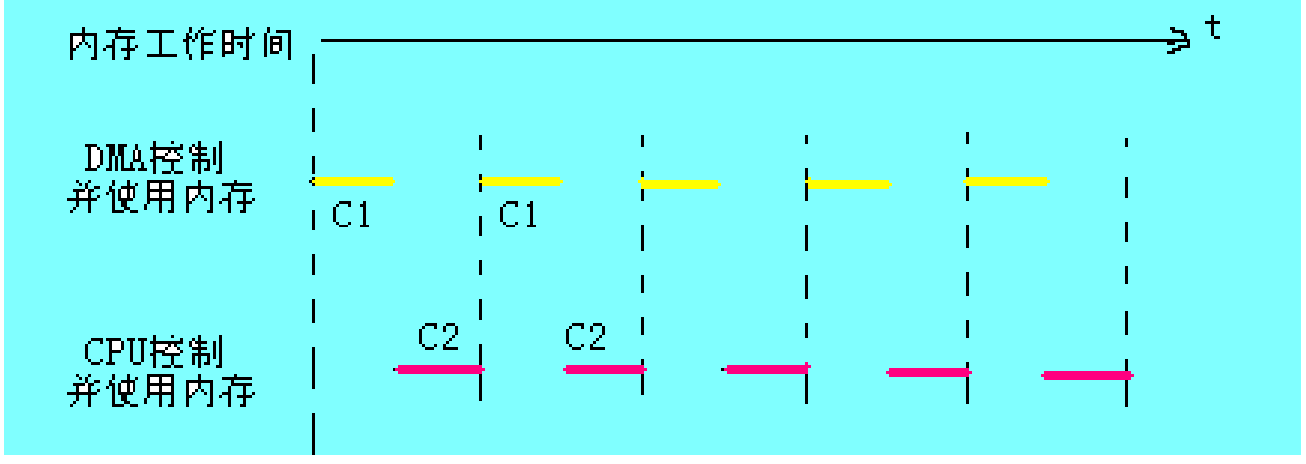
# 方法3：DMA与CPU交替访内

- 如果CPU的工作周期比内存存取周期长很多，此时采用交替访内的方法可以使DMA传送和CPU同时发挥最高的效率
  - 假设CPU工作周期为 1.2μs，内存存取周期小于0.6μs
  - 那么一个CPU周期可分为C1和C2两个分周期，其中C1供DMA控制器访内，C2专供CPU访内
- 总线控制权的转移速度快，DMA效率高



# 方法3：DMA与CPU交替访内

- 这种方式不需要总线使用权的申请、建立和归还过程，总线使用权是通过C1和C2分时进行的
- 这种传送方式又称为“透明的DMA”方式
  - 这种DMA传送对CPU来说，如同透明的玻璃一般，没有任何感觉或影响
  - 在透明的DMA方式下工作，CPU既不停止主程序的运行，也不进入等待状态，是一种高效率的工作方式

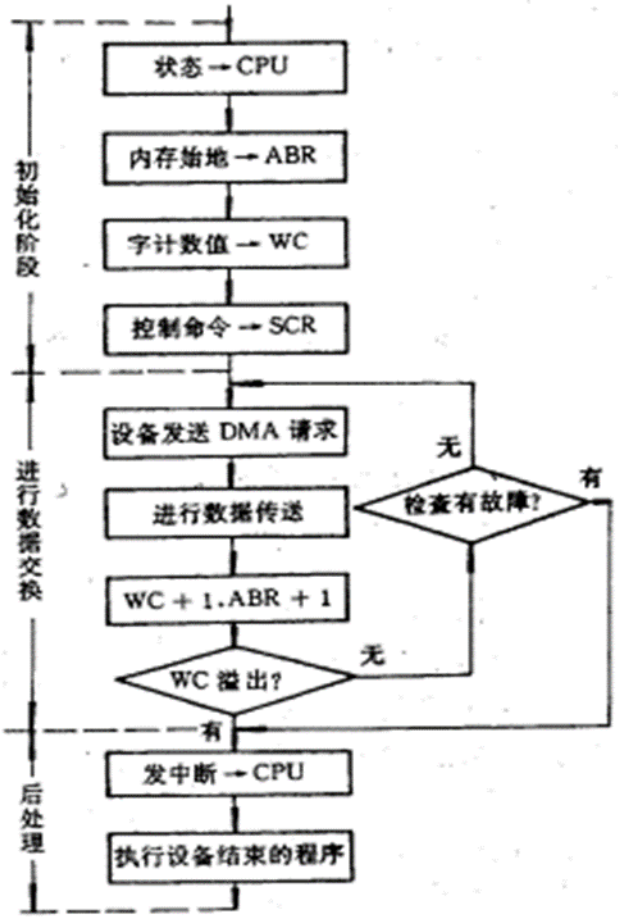


# 第八章 输入输出系统

- 程序查询方式
- 程序中断方式
- DMA方式
  - DMA基本概念与传送方式
  - DMA数据传送流程

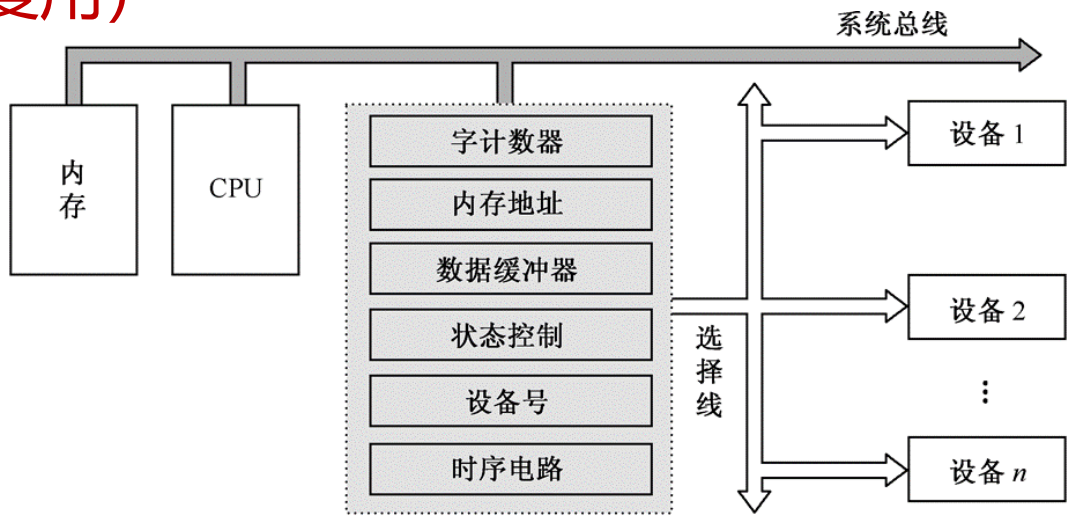
# DMA数据传输流程

- DMA的数据块传送过程分为三个阶段
  - 预处理：CPU向DMA控制器的设备地址寄存器中送入设备号并启动设备，向内存地址计数器中送入起始地址，向字计数器中送入交换的数据字个数
  - 正式传送：当外设准备好发送数据或接受数据时，它发出DMA请求，由DMA控制器向CPU发出总线使用权的请求（HOLD）
  - 后处理：一旦DMA的中断请求得到响应，CPU停止主程序的执行，转去做一些DMA的结束处理工作



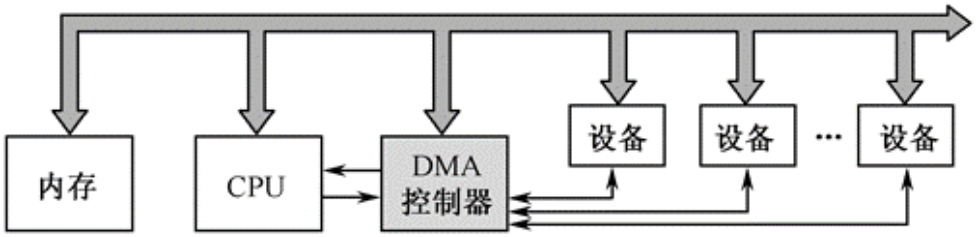
# 选择型DMA控制器

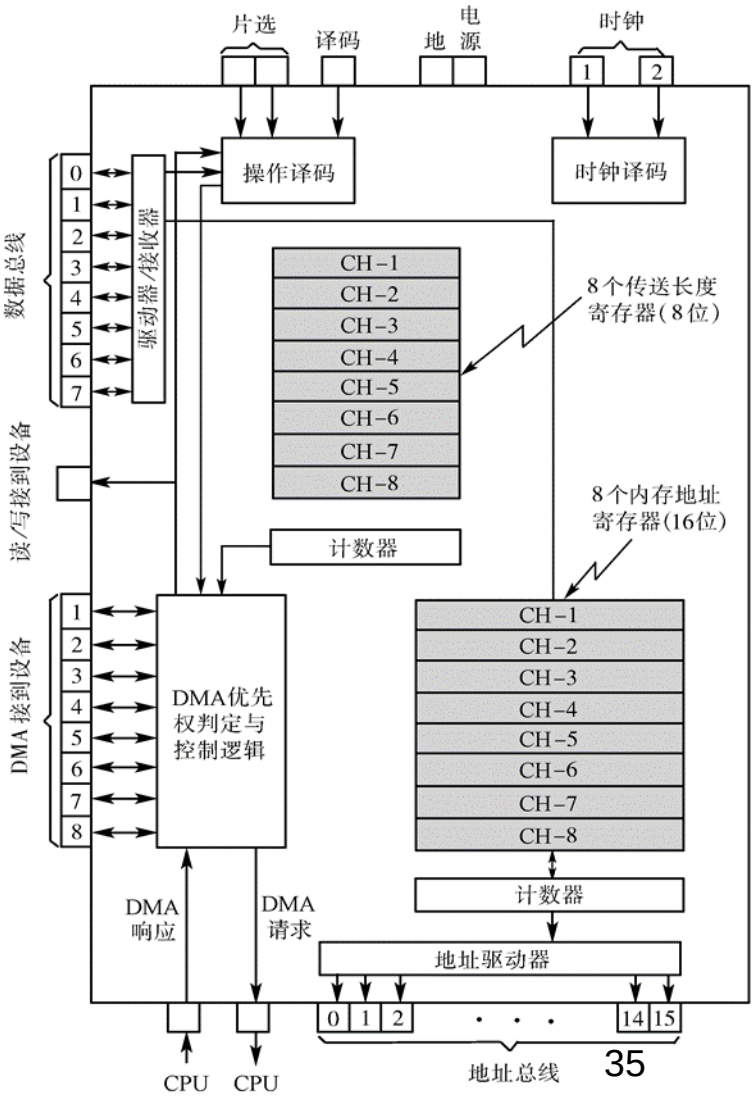
- 选择型DMA控制器在物理上可以连接多个设备，而在逻辑上只允许连接一个设备（同一时间服务一个设备）
- 只增加少量硬件达到为多个外围设备服务的目的，适合数据传输率很高以至接近内存存取速度的设备
- 在很快地传送完一个数据块后，控制器又可为其他设备服务（时分复用）



# 多路型DMA控制器

- 多路型DMA不仅在物理上可以连接多个外围设备，而且在逻辑上也允许这些外围设备同时工作，各设备以字节交叉方式通过DMA控制器进行数据传送
- 多路型DMA控制器适合于同时为多个慢速外围设备服务



# CPU占用例题

某CPU主频为500MHz，平均CPI为5，外设数据传输速率0.5MB/s。采用中断与主机数据传送，以32位为传输单位，对应中断服务程序包含18条指令，其他开销为2条指令。

1）中断方式下，CPU用于I/O时间占CPU时间百分比为？

2）若采用DMA方式，数据传输率为5MB/s，每次DMA数据块5000B，DMA访存与CPU无冲突，DMA开销500个时钟周期，则CPU-DMA方式找时间占比为？

# CPU占用例题

某CPU主频为500MHz，平均CPI为5，外设数据传输速率0.5MB/s。采用中断与主机数据传送，以32位为传输单位，对应中断服务程序包含18条指令，其他开销为2条指令。

1）中断方式下，CPU用于I/O时间占CPU时间百分比为？

一次传输CPU周期（18+2）*5=100周期

每秒次数：0.5MB/4=0.125M

CPU时间占比：12.5M/500M=2.5%

# CPU占用例题

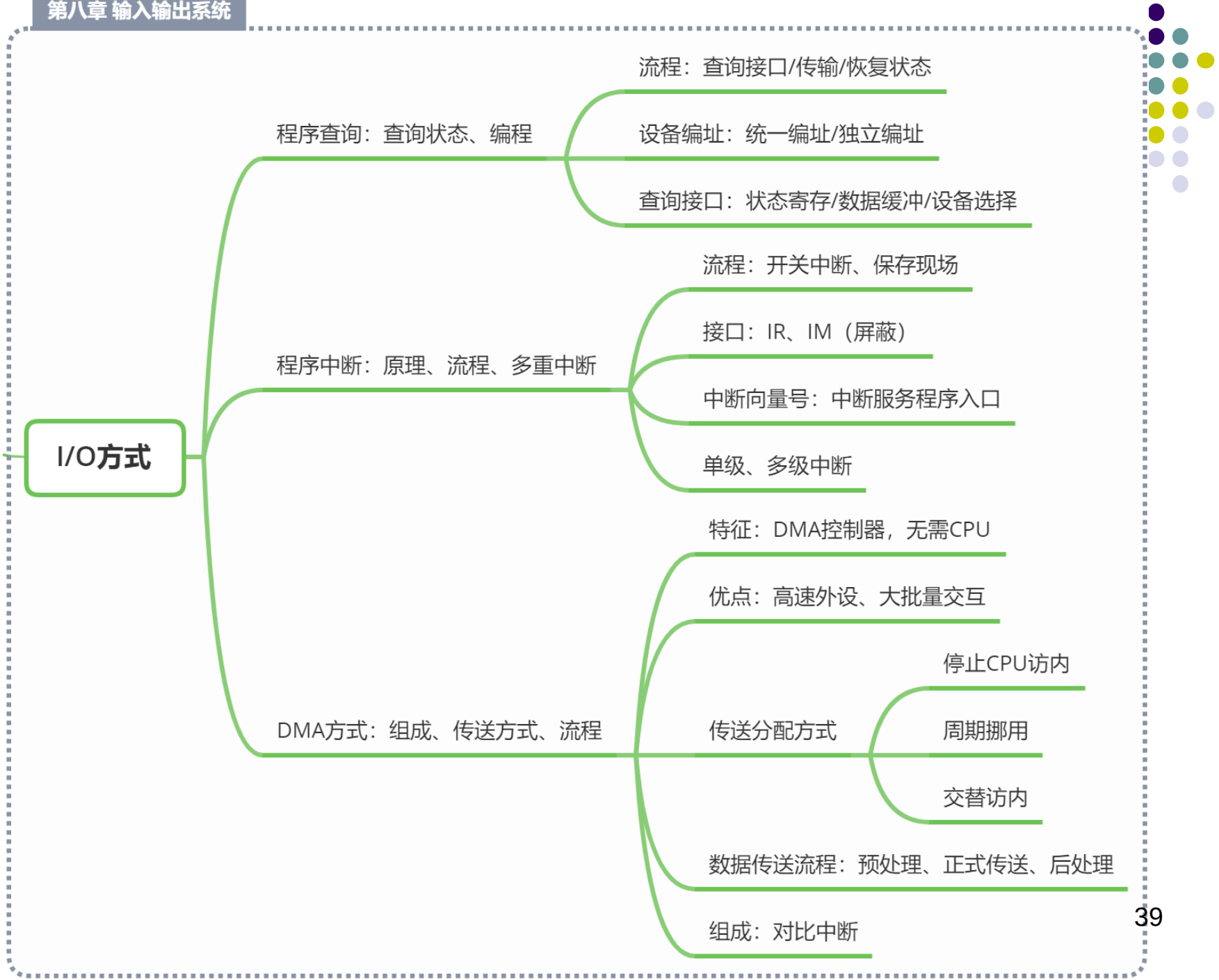
某CPU主频为500MHz。

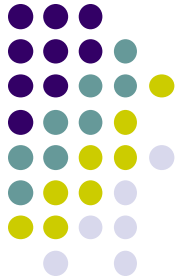
2）若采用DMA方式，数据传输率为5MB/s，每次DMA数据块5000B，DMA访存与CPU无冲突，DMA开销500个时钟周期，则CPU-DMA方式的时间占比为？

每秒DMA次数：5M/5000=1k

DMA开销时钟周期：1k*500=500k

占比：500k/500MHz=0.1%


I/O方式
程序查询：查询状态、编程
流程：查询接口/传输/恢复状态
设备编址：统一编址/独立编址
查询接口：状态寄存/数据缓冲/设备选择
程序中断：原理、流程、多重中断
流程：开关中断、保存现场
接口：IR、IM（屏蔽）
中断向量号：中断服务程序入口
单级、多级中断
DMA方式：组成、传送方式、流程
特征：DMA控制器，无需CPU
优点：高速外设、大批量交互
传送分配方式
停止CPU访内
周期挪用
交替访内
数据传送流程：预处理、正式传送、后处理
组成：对比中断



# 计算机组成与系统结构

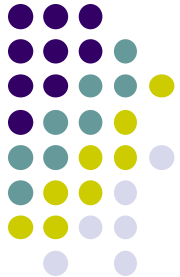
# 第九章 并行体系机构

吕昕晨

lvxinchen@bupt.edu.cn

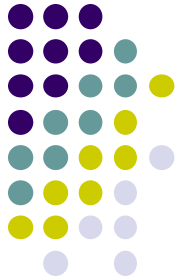
网络空间安全学院

# 体系结构中的并行性



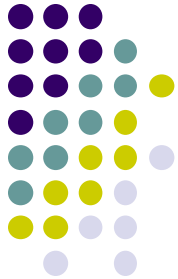
- 所谓并行性，是指计算机系统具有可以同时进行运算或操作的特性，它包括同时性与并发性两种含义。
  - 同时性--两个或两个以上的事件在同一时刻发生。
  - 并发性--两个或两个以上的事件在同一时间间隔发生（分时交替执行、时分复用）

# 并行性等级——处理数据



- 计算机系统中的并行性有不同的等级。
- **从处理数据的角度看**，并行性等级从低到高可分为：
  - **字串位串** 同时只对一个字的一位进行处理。这是最基本的串行处理方式，不存在并行性。
  - **字串位并** 同时对一个字的全部位进行处理，不同字之间是串行的。这里已开始出现并行性。
  - **字并位串** 同时对许多字的同一位进行处理。这种方式有较高的并行性。
  - **全并行** 同时对许多字的全部位进行处理。这是最高一级的并行。

# 并行性等级——执行程序



- **并行性等级--从执行程序的角度分**
  - **指令内部并行**

    一条指令执行时各微操作之间的并行。
  - **指令级并行**

    并行执行两条或多条指令。
  - **任务级或过程级并行**

    并行执行两个以上过程或任务（程序段）。
  - **作业或程序级并行**

    并行执行两个以上作业或程序。

# 并行性途径

**提高并行性的技术途径：**

**时间重叠：即时间并行**

- 多个处理过程在时间上相互错开，**轮流重叠**地使用同一套硬件设备的各个部分

**资源重复：即空间并行**

- 通过**重复设置硬件**资源，大幅度提高计算机系统的性能

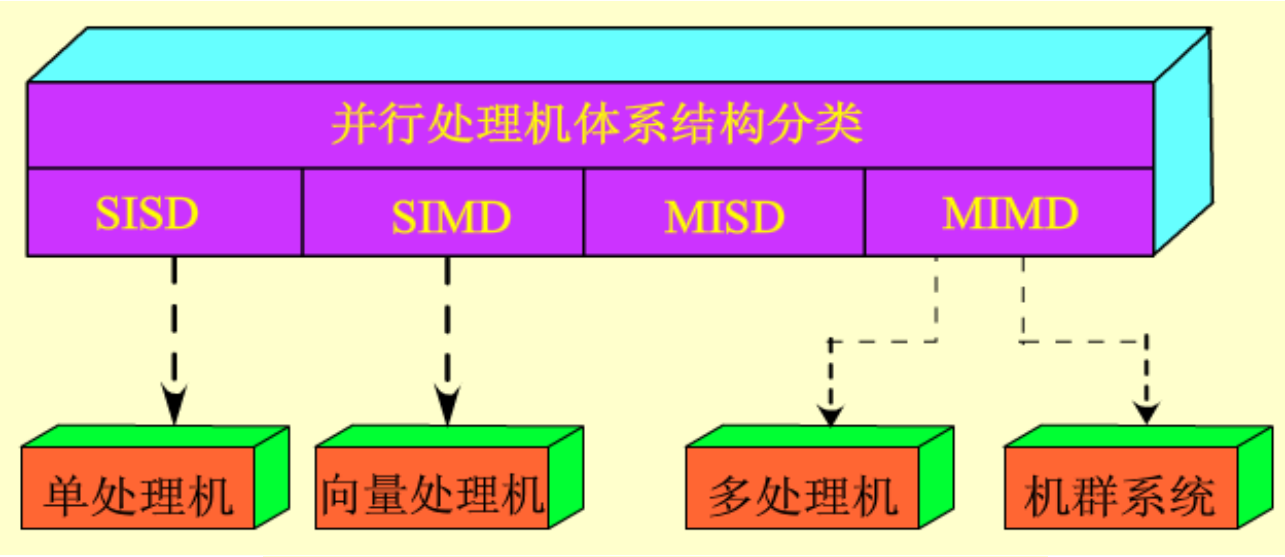
**时间重叠+资源重复---主流技术**

**资源共享**

- 用软件方法实现多个任务按一定时间顺序轮流使用同一套硬件设备

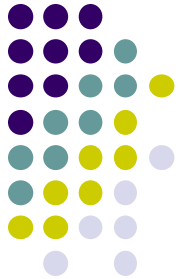
# 并行处理机结构类型分类

- 指令流和数据流的不同组织方式：（Flynn分类法）
  - 单指令流单数据流（SISD)，其代表机型是单处理机。
  - 单指令流多数据流（SIMD)，其代表机型是向量处理机。
  - 多指令流单数据流（MISD)，这种结构从来没有实现过。
  - 多指令流多数据流（MIMD)，其代表机型是多处理机和机群系统。




并行处理机体系结构类型

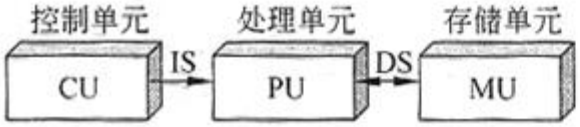
# 本章重要知识点总结（1）



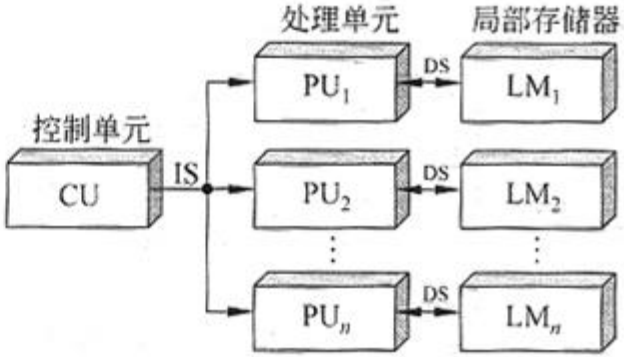
| 名称 | 概念 |
| --- | --- |
| 并行性 | 并行性是指计算机系统具有可以同时进行运算或者操作的特性，它包括同时性与并发性两种含义；同时性是指两个或两个以上的事件在同一时刻发生；并发性是指两个或两个以上的事件在同一时间间隔发生（如分时交替执行、重叠执行等）。 |
| VLIW（超长指令字）处理机 | 由编译程序在编译时找出指令间潜在的并行性，进行适当调度安排，把多个能并行执行的操作组合在一起，成为一条具有多个操作段的超长指令。由这条超长指令去控制VLIW处理机中多个互相独立工作的功能部件，每个操作段控制一个功能部件，相当于同时执行多条指令。 |
| 超线程技术 | 同时调度多个线程执行，即多条指令流，按一定的策略往超标量流水线中交替/混合发射指令。流水线处理机可以同时处理来自不同线程的多条指令，可有效避免单指令流中的相关问题。 |

# 本章重要知识点总结（2）

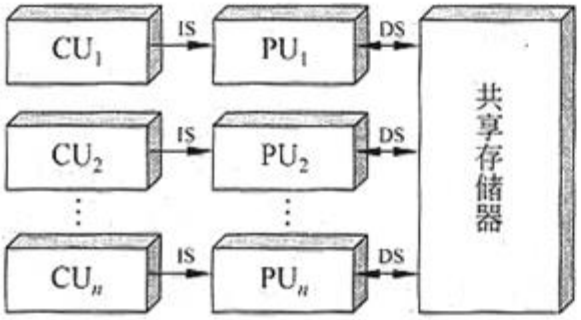
| 名称 | 代表机型 |
| --- | --- |
| 单指令流单数据流（SISD) | 单处理机 |
| 单指令流多数据流（SIMD) | 向量处理机。 |
| 多指令流单数据流（MISD) | 无，未实现过 |
| 多指令流多数据流（MIMD) | 多处理机和机群系统。 |



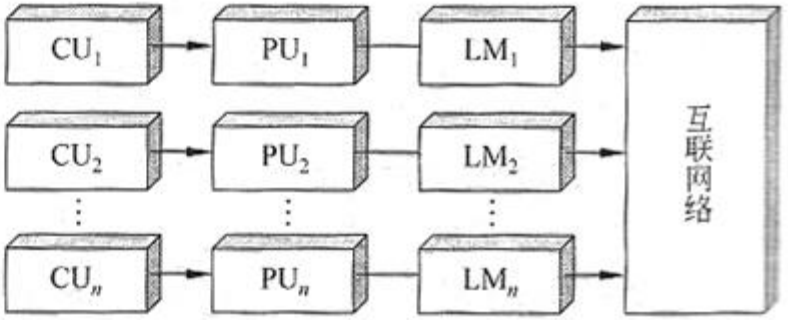
(a) SISD



(b) SIMD(分布式存储器)



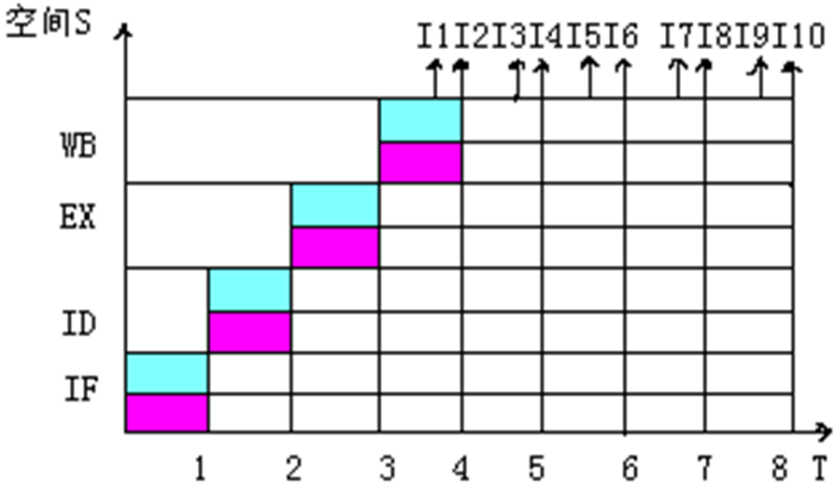
(c) MIMD(共享存储器)



(d) MIMD(分布式存储器)

# 本章习题

- 下列论述中，不正确的是
  - A. 超线程技术在一颗处理机芯片内设计多个逻辑上的处理机内核
  - B. 多线程技术能够屏蔽线程存储器的访问延迟，增加系统吞吐率
  - C. 多指令流单数据流（MISD）从未实现过
  - D. 超标量技术是同时多线程技术在英特尔处理机产品的具体实现




答案：D

# 本章习题

- 计算机系统的并行性是指
  - A. 只有一个事件发生
  - B. 两个以上事件不在同一时刻发生
  - C. 两个以上事件在同一时刻发生
  - D. 两个以上事件在同一时刻发生或在同一时间间隔发生

- 所谓并行性，是指计算机系统具有可以同时进行运算或操作的特性，它包括同时性与并发性两种含义。
  - 同时性--两个或两个以上的事件在同一时刻发生。
  - 并发性--两个或两个以上的事件在同一时间间隔发生（分时交替执行、时分复用）

答案：D

# 本章习题

- 从处理数据角度看，不存在并行性的是
  - A. 字串位串
  - B. 字串位并
  - C. 字并位串
  - D. 字并位并

- 计算机系统中的并行性有不同的等级。
- **从处理数据的角度看**，并行性等级从低到高可分为：
  - **字串位串** 同时只对一个字的一位进行处理。这是最基本的串行处理方式，不存在并行性。
  - **字串位并** 同时对一个字的全部位进行处理，不同字之间是串行的。这里已开始出现并行性。
  - **字并位串** 同时对许多字的同一位进行处理。这种方式有较高的并行性。
  - **全并行** 同时对许多字的全部位进行处理。这是最高一级的并行。

答案：A