

程序查询方式

每时每刻需要不断地查询IO设备是否准备就绪；

当IO设备较多时，CPU需要按照各个IO设备在系统中的优选级别进行逐级查询；

CPU一旦启动IO，必须停止现行程序的运行，并在现行程序中插入一段程序；

流程图

特点

CPU有踏步等待的现象；CPU与IO串行工作

优点

- 接口设计简单；
- 设备量小；

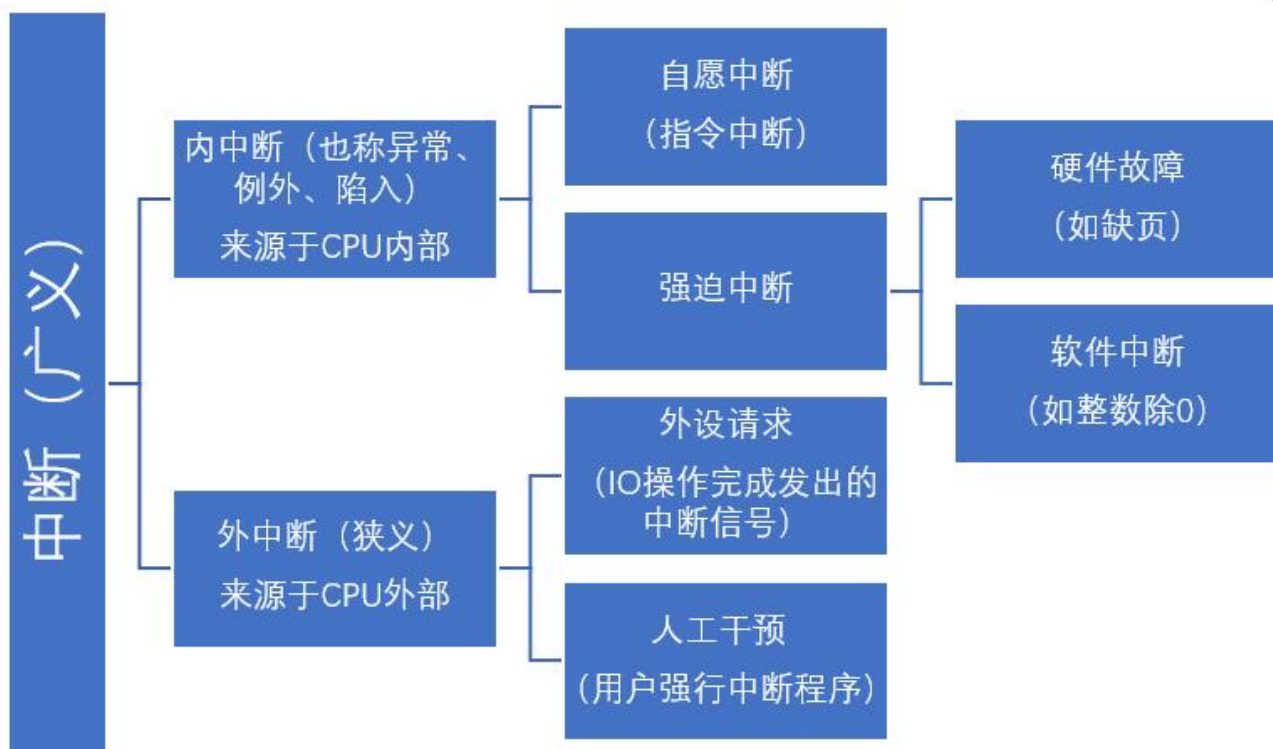
缺点

- 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；

程序中断方式

计算机执行现行程序的过程中，出现某些急需处理的异常情况或者特殊请求，CPU暂时中止现行程序，转而处理异常或者特殊请求，处理完毕后CPU又自动返回现行程序断点处；

分类



中断请求过程

中断响应过程

响应中断的条件

- 中断源有中断请求
- CPU允许中断以及开中断
- 一条指令执行完毕，进行查询有无中断的阶段
- 新的中断请求比当前运行的程序优先级更高

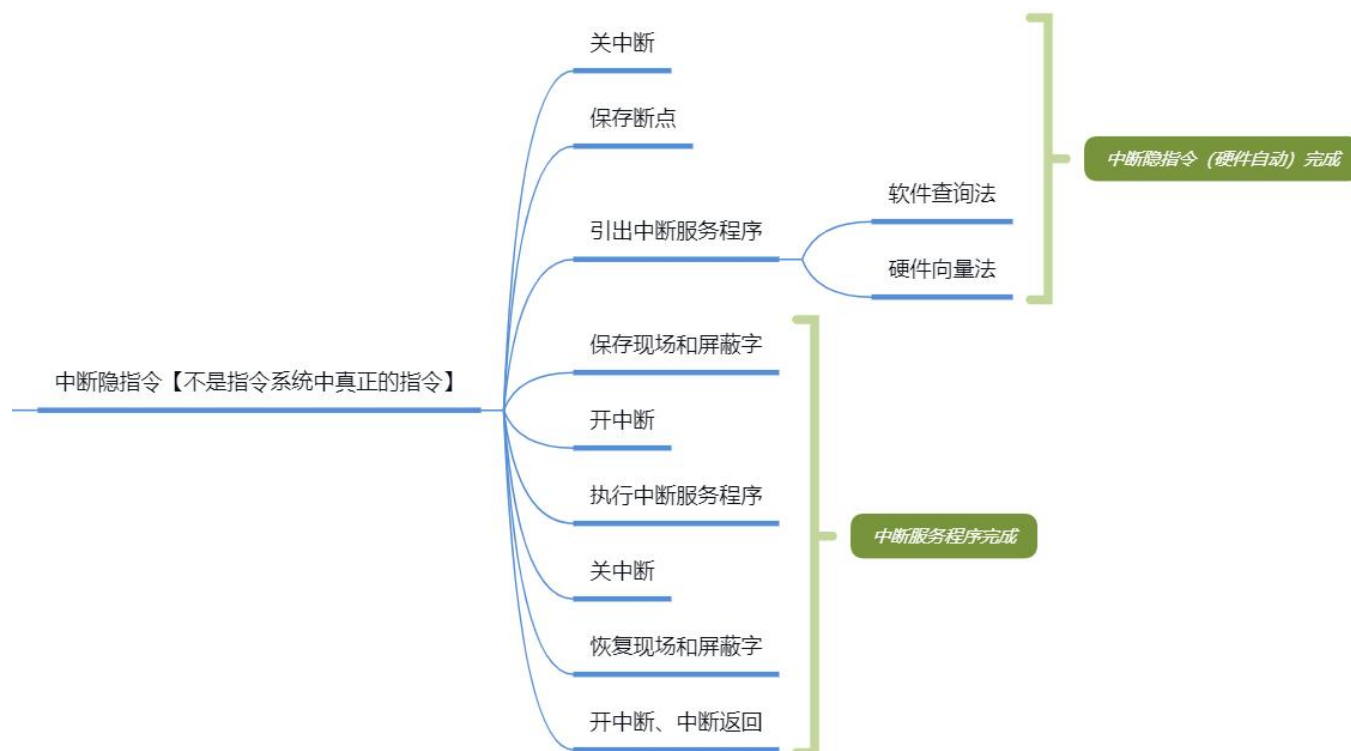
中断判优

- 软件；查询程序
- 硬件；硬件排队器

优先级

- 硬件 > 软件
- 非屏蔽 > 屏蔽
- DMA > IO
- 高速 > 低速
- I > O
- 实时 > 普通

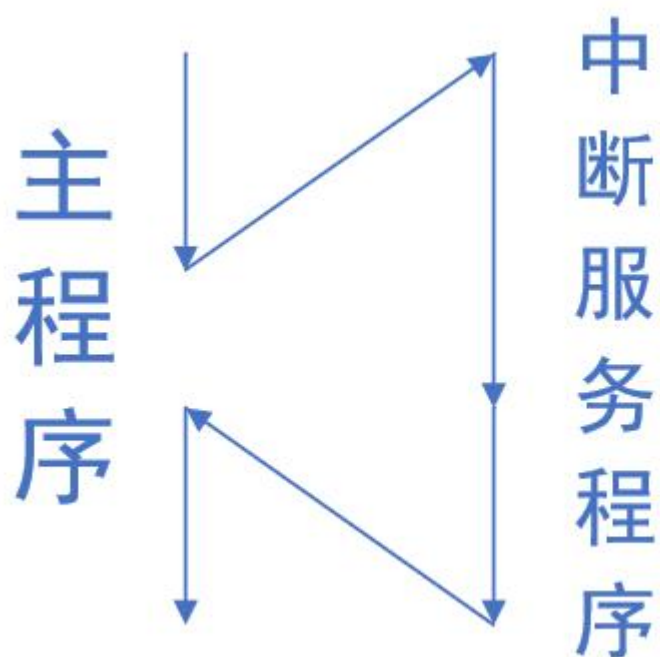
中断处理过程



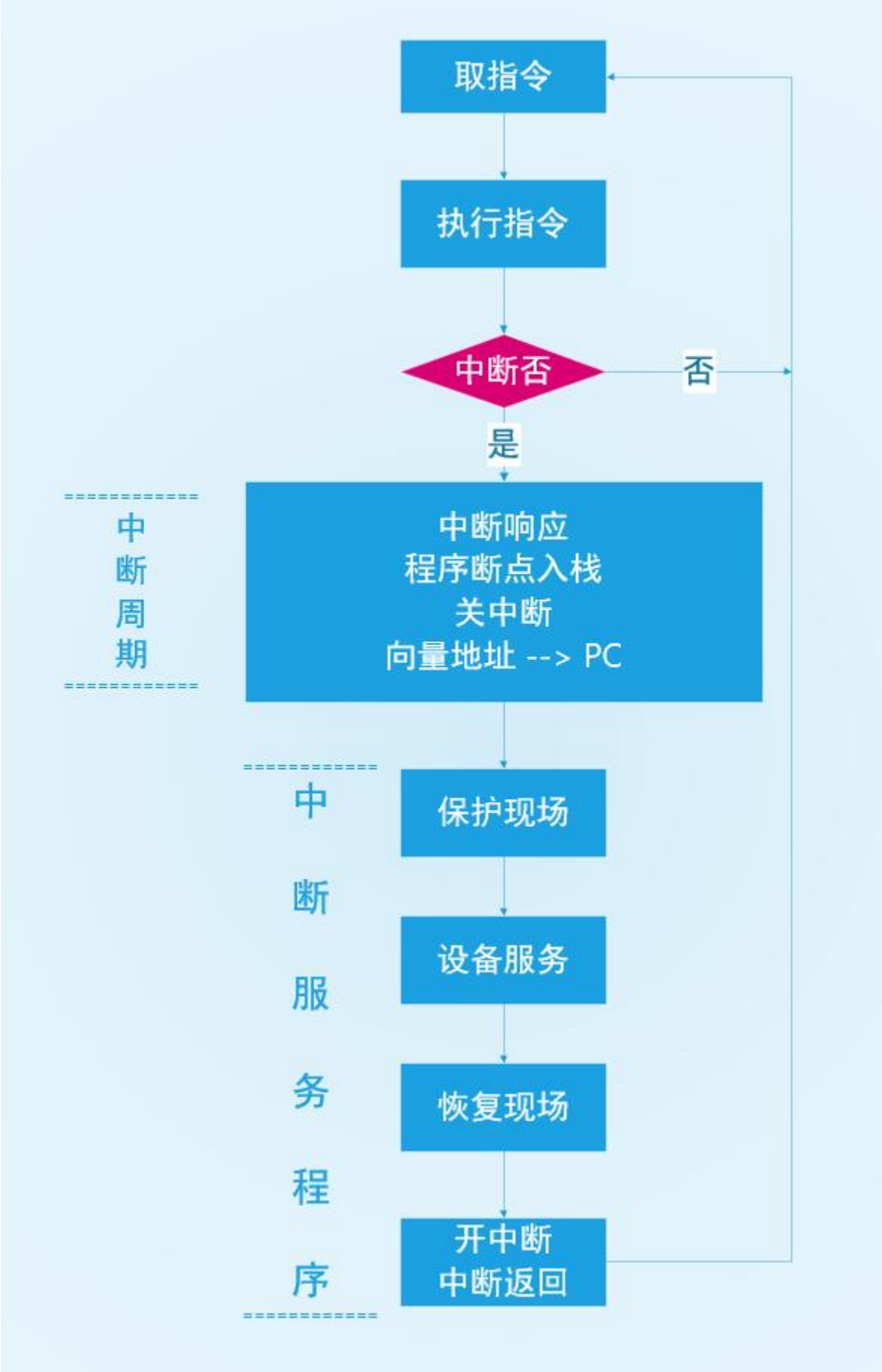
中断服务程序

1. 保护现场；
 - 保存程序断点
 - 保存通用寄存器和状态寄存器的内容
2. 中断服务；
 - 对于不同的中断请求源，其中断服务内容是不同的
3. 恢复现场；
 - 将原程序中断时的“现场”恢复到原来的寄存器中（即恢复原样）
4. 中断返回；
 - 最后一条指令通常为中断返回指令，使其返回到原程序的断点处

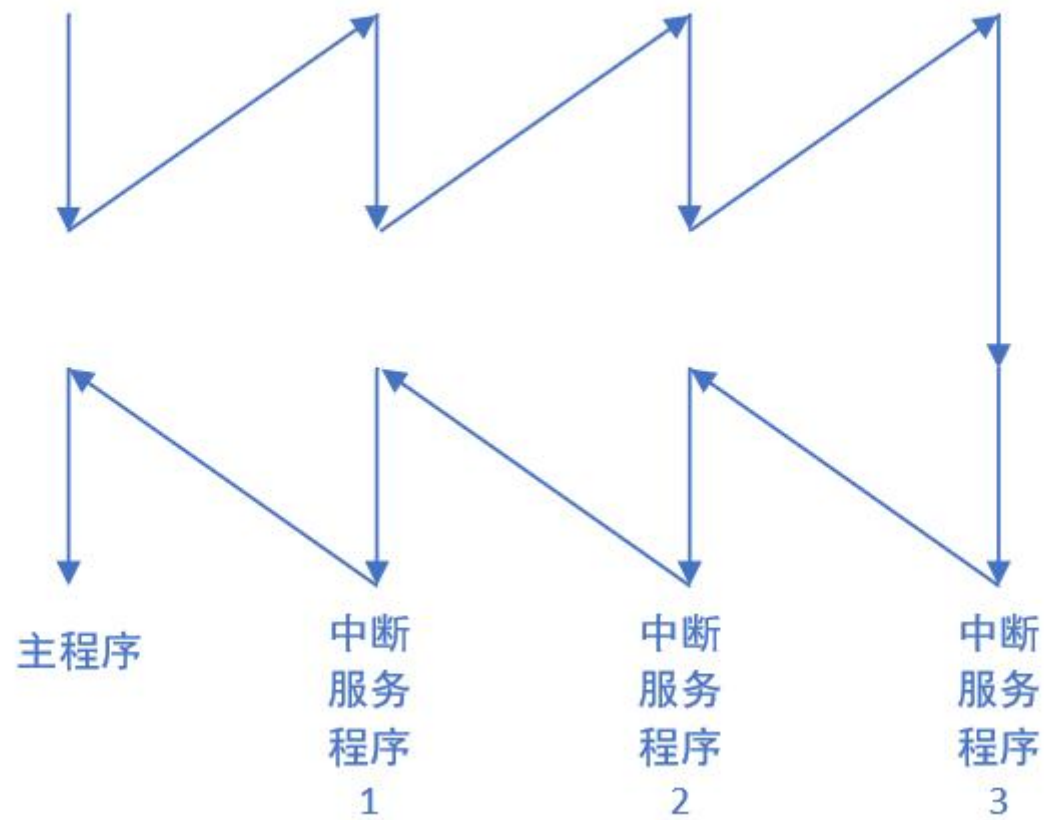
单重中断



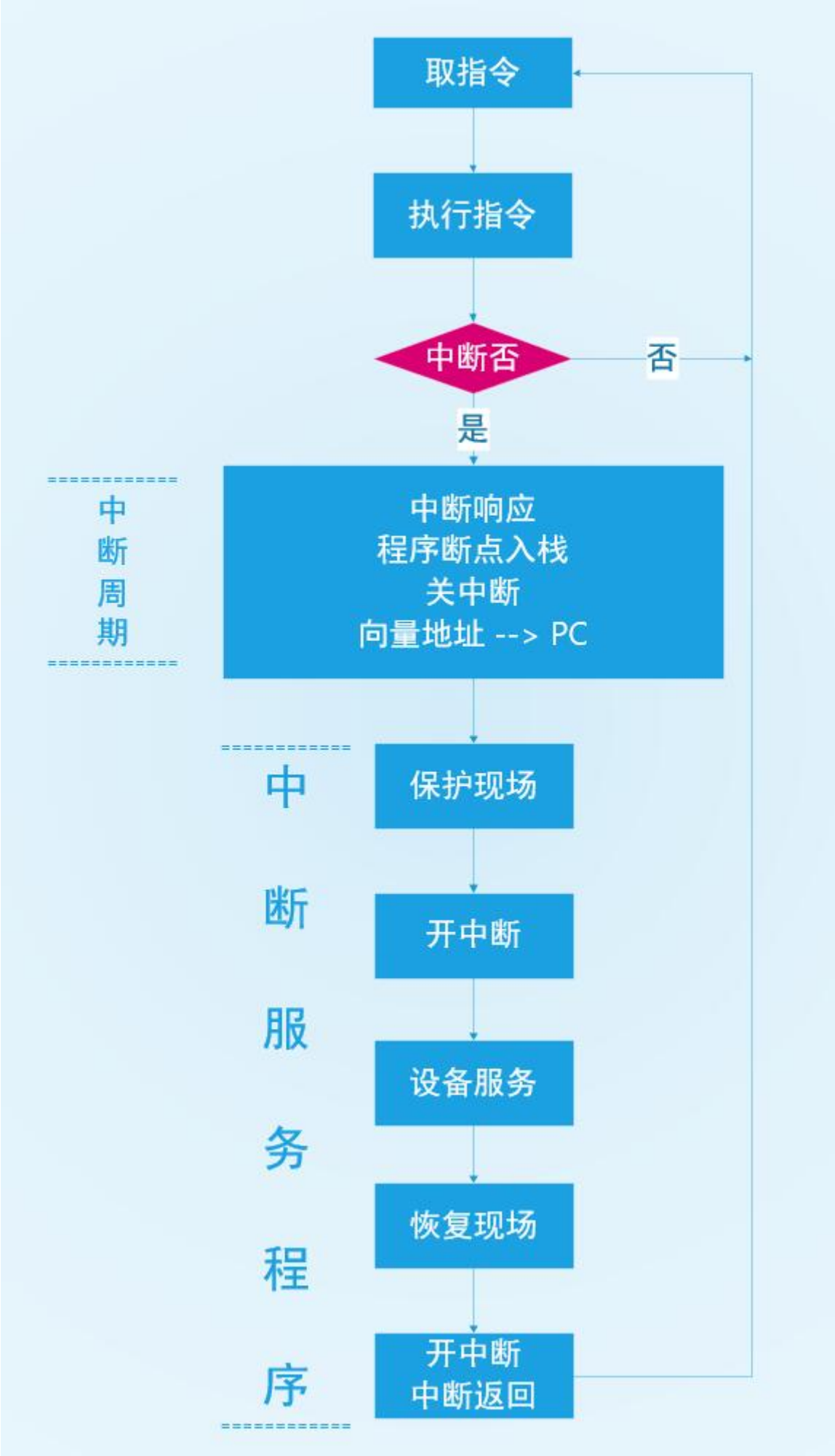
如果CPU在执行中断服务程序的过程中，又出现了新的更高优先级的中断请求，而CPU对新的中断请求不予响应，这种中断称为单重中断；



多重中断



如果CPU暂停现行的中断服务程序，转而去执行新的中断请求，这种中断为多重中断，又称中断嵌套



单重中断与多重中断的区别

多重中断与单重中断的区别在于多重中断在保护现场后需要开中断，以便响应更高优先级的中断请求，而在恢复现场前又需要关中断，恢复现场的过程不能被中断

中断屏蔽

中断屏蔽技术主要用于多重中断

CPU要满足中断必须满足以下条件

- 在中断服务程序中提前设置开中断指令
- 优先级别高的中断源有权中断优先级别低的中断源

每个中断源都有一个屏蔽触发器，1表示屏蔽该中断源的请求，0表示可以正常申请

- 屏蔽字触发器；所有屏蔽触发器组成在一起构成的
- 屏蔽字；屏蔽字寄存器内容

DMA

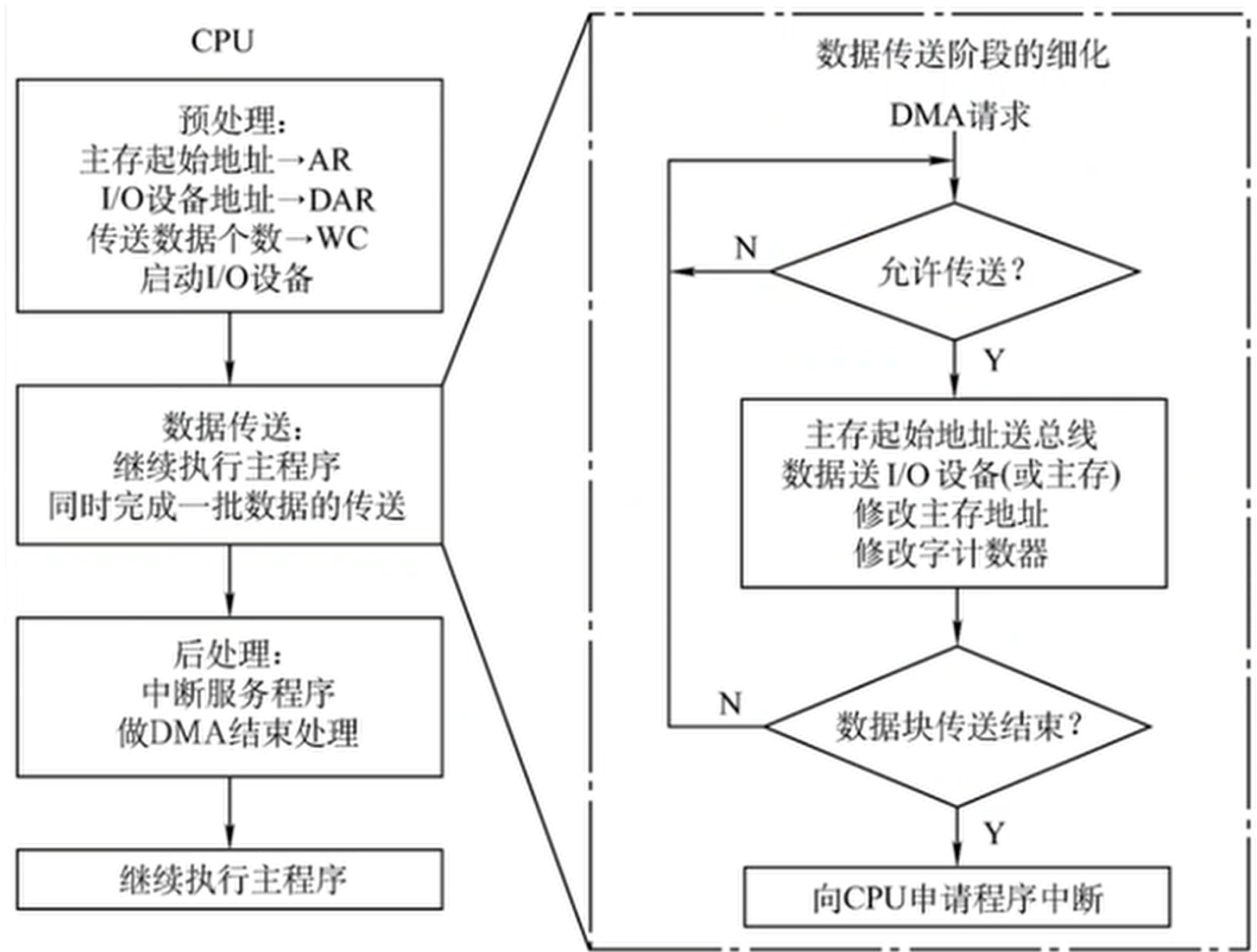
主存和DMA接口之间有一条直接数据通路

由于DMA方式传送数据不需要经过CPU，因此不必中断现行程序，IO与主机并行工作，程序和传送并行工作

DMA方式具有以下特点：

- 它使主存与CPU的固定关系脱钩，主存既可以被CPU访问，又可以被外设访问
- 在数据块传送时，主存的地址确定、传送数据的计数等都由硬件电路直接实现
- 主存中要开辟专用缓冲区，及时供给和接收外设的数据
- DMA传送速度快，CPU和外设并行工作，提高了系统效率
- DMA在传送开始前要通过程序进行预处理，结束后要通过中断方式进行后处理

DMA传送过程



DMA传送方式

- 停止CPU访问
- DMA与CPU交替访问
- 周期挪用（周期窃取）

DMA与中断的区别

	中断	DMA
数据传送	程序控制 程序的切换 → 保存和恢复现场	硬件控制 CPU只需进行预处理和后处理
中断请求	传送数据	后处理
响应	指令执行周期结束后响应中断	每个机器周期结束均可，总线空闲时即可响应DMA请求
场景	CPU控制，低速设备	DMA控制器控制，高速设备
优先级	优先级低于DMA	优先级高于中断
异常处理	能处理异常事件	仅传送数据