

第8章 输入输出系统

The title is centered and surrounded by six light purple circles. Three circles are positioned above the text, and three are below it. The circles are of varying sizes and are arranged in a way that they partially overlap the text and each other.

OUTLINES

- 8.1 外围设备的的速度分级与信息交换方式
——了解
- 8.2 程序查询方式 ——了解
- 8.3 程序中断方式——掌握
- 8.4 DMA方式——掌握
- 8.5 通道方式 ——了解
- 8.6 通用I/O标准接口——了解

考研大纲要求

(一) I/O系统基本概念

(二) 外部设备

(三) I/O接口(I/O控制器)

1. I/O接口的功能和基本结构

2. I/O端口及其编址

(四) I/O方式

1. 程序查询方式

2. 程序中断方式

3. DMA方式

DMA控制器的组成; DMA传送过程。

4. 通道方式

第7章

8.1 外围设备的速度分级与信息交换方式

- 8.1.1 外围设备的速度分级
- 8.1.2 信息交换方式

8.1.1 外围设备的速度分级

如何使CPU与外设同步工作

● 根据外设的工作速度，**CPU**与外设的定时方式有以下3种：

① 速度极慢或简单的外围设备

如：机械开关、发光二极管

◆ **CPU**直接接收或发送数据。

② 慢速或中速的外围设备

如：键盘、显示器

询问信号
应答信号

◆ 采用异步定时方式，或称为应答式数据交换；

◆ **CPU**与外设之间通过两个相互的联络信号来决定开始数据传送的时间。

③ 高速的外围设备

如：主存、辅存

◆ 采用同步定时方式；

由时钟脉冲控制

◆ **CPU**以等间隔的速率执行输入 / 输出指令。

8.1.2 外设信息交换方式

动画演示：8-1.swf

- 程序查询方式

- 早期计算机中使用的一种方式，效率低。

- 程序中中断方式

- 适用于随机出现的服务。

- 直接内存访问（**DMA**）方式

- 适用于内存和高速外围设备之间大批数据交换的场合。

- 通道方式

- 增加一个具有特殊功能的处理器——通道，将**CPU**的输入输出权力下放。

- 外围处理机方式(**PPU**)

- 是通道方式的进一步发展。

程序查询方式

- 工作过程

- CPU 传送数据之前先检查外设的状态，若没有准备好，则继续查询等待，直至外设就绪即进行数据传送。

- 硬件要求

- 只需接口电路的状态、数据口，不需增加其他控制电路。

- 特点

- CPU 主动查询，程序控制数据传送过程，简单易行；
- 每次查询之后只能传送一个字或一个字节的数据，数据传输率不高，CPU 时间浪费较多。

- 适用场合

- CPU 不太忙且对传送速度要求不高的系统。

程序中中断方式

●工作过程

- CPU有传送要求时，启动外设后可处理其他事件，当外设就绪后，通过中断的方式和CPU完成数据传送工作。

●硬件要求

- 需要附加的中断控制电路。

●特点

- 一定程度上实现了CPU和外设的并行工作；
- 中断操作过程增加了软硬件的开销，且每次数据传送只能传送一个字或一个字节的数据，数据传送效率低。

●适用场合

- CPU与慢速外设之间的数据传送。

直接内存访问（DMA）方式

存储器与外设之间
数据交换时使用

● 工作过程

- CPU分配总线使用权之后，在硬件DMA控制器（DMAC）的控制下完成存储器与高速外设之间的大量数据的传送。

● 硬件要求

- 需要相应的DMA控制器及数据通路，电路结构复杂，硬件开销大。

● 特点

- 数据传送不需要经过CPU，直接由硬件控制；
- 外设与存储器之间的数据传送量大。

● 适用场合

- 包含有高速外设的系统中。

通道方式

● 工作过程

- CPU将IO控制的权利下放给通道，由通道统一管理所有的输入输出操作。

● 硬件要求

- 需要IO通道（也称IO处理器，IOP）；
- 通道是一个具有特殊功能的处理器。

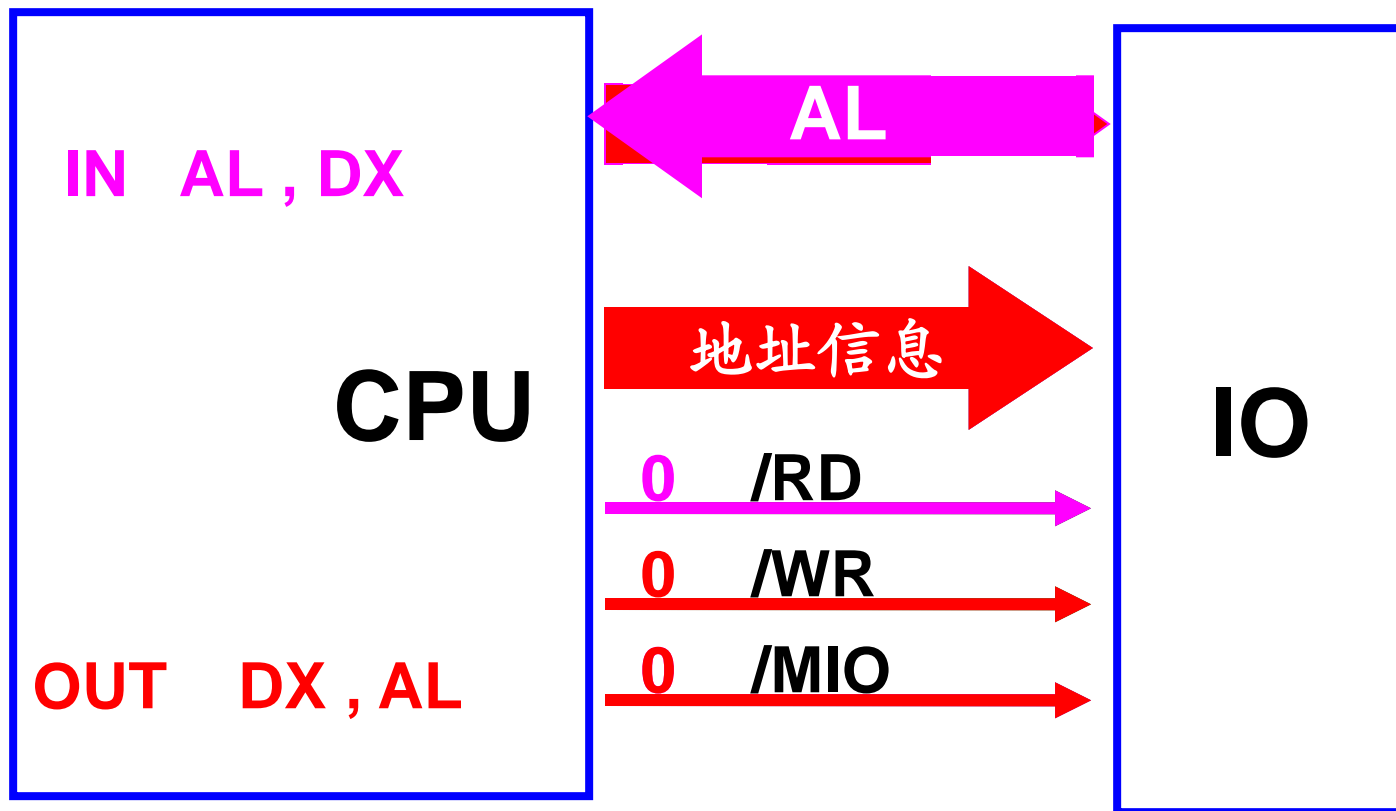
● 特点

- 硬件代价较高；
- IO传送的效率高，并且减轻了CPU的工作负担。

● 适用场合

- 高性能要求的系统中。

8.2 程序查询方式



数据的输入
输出完全由
程序控制!

1、设备编址——统一编址方式

●统一编址方式

以20位主存地址、
16位I/O地址为例

○将I/O系统与主存系统作为一个整体进行编址；

●优点：

○访问I/O端口可使用访存指令，操作类型多样，使用灵活；

○I/O端口有较大的编址空间；

●缺点：

○占用主存空间，使实际主存容量减小；

○I/O访问的指令字长较长，执行速度慢。

00000H

主
存
储
器

FFFFFFH

F0000H

I/O
设备

FFFFFFH

1、设备编址——独立编址方式

- 独立编址方式

- 将I/O系统与主存系统分别编址;

- 特点:

- I/O端口地址不占用主存空间;

- 使用专用的I/O指令，指令字长短，
执行速度快;

- 与主存空间区分容易。

以20位主存地址、
16位I/O地址为例

00000H

FFFFFFH

主
存
储
器

0000H

FFFFH

I/O
设
备

2、输入/输出指令

●IO指令

○ 输入指令: **IN AL/AX, DX/PORT**

◆ 从指定端口读入一个字节/字数据到累加器;

○ 输出指令: **OUT DX/PORT, AL/AX**

◆ 将累加器中的一个字节/字数据送到指定端口输出;

●IO指令一般的功能:

① 对I/O接口的控制触发器置1或0, 控制其进行某些操作;

◆ CPU向IO接口发命令字;

② 测试设备的某些状态;

◆ CPU读取IO接口的状态字;

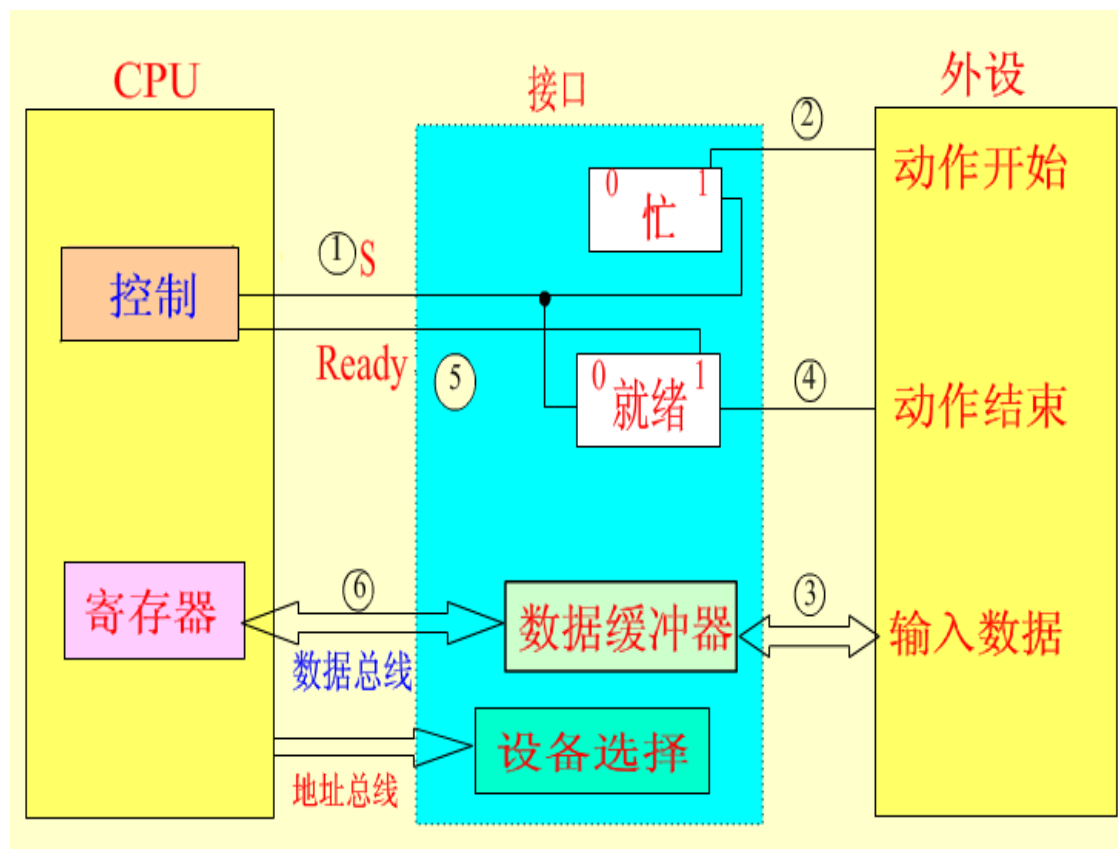
③ 输入或输出数据;

◆ CPU与IO接口之间的数据传送;

} 以数据形式传送
控制、状态信息

3、程序查询方式的接口

- CPU通过地址信号选中某设备接口；
- CPU通过向该接口发送命令字的方式，启动外设；
- 外设开始工作后，设置当前“忙”状态；
- CPU与外设通过接口内部的数据缓冲器传送数据；



动画演示：
8-2. swf

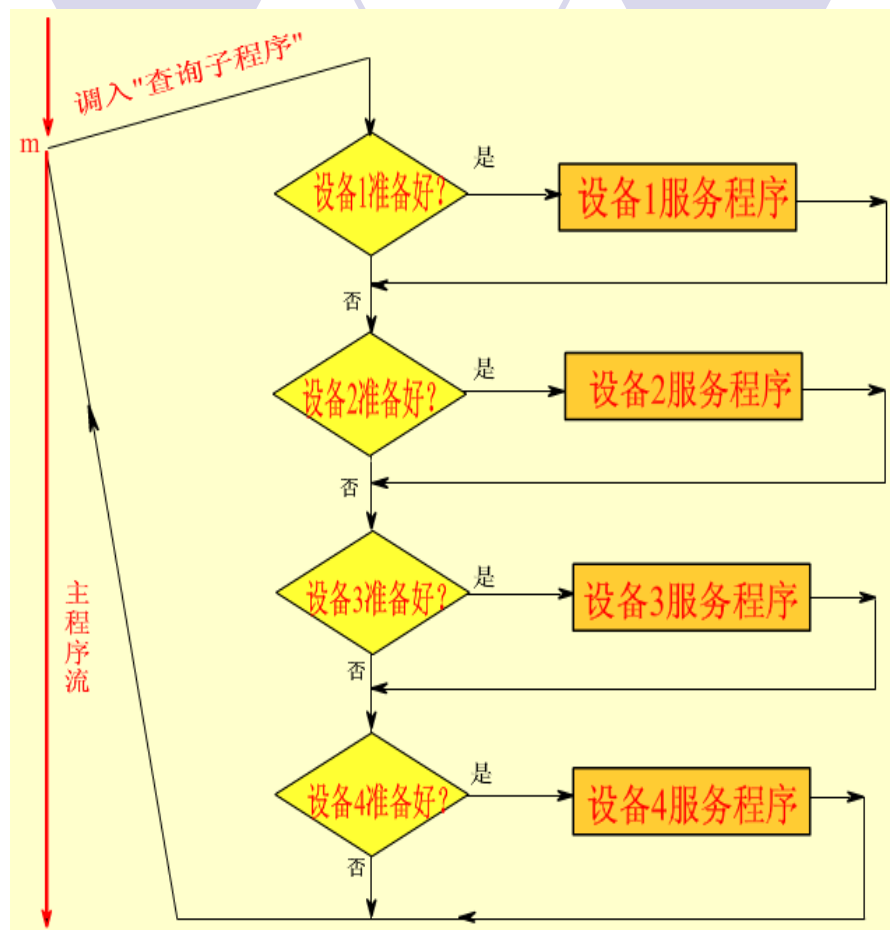
4、程序查询输入/输出方式

●程序执行动作如下：

- ① CPU请求数据传送；
- ② CPU从I/O接口读入状态字；
- ③ 检查状态字中的标志；
- ④ 未就绪，则重复②、③步，直到设备就绪为止。
- ⑤ CPU输入或输出数据，同时复位接口中的状态标志。

●特点

- 可通过改变查询顺序修改设备的优先权；
- CPU工作效率低；



动画演示：
8-3. swf

8.3 程序中中断方式

- 8.3.1 中断的基本概念
- 8.3.2 程序中中断方式的基本IO接口
- 8.3.3 单级中断
- 8.3.4 多级中断
- 8.3.5 中断控制器
- 8.3.6 PC系列机中断机制

8.3.1 中断的基本概念

●定义:

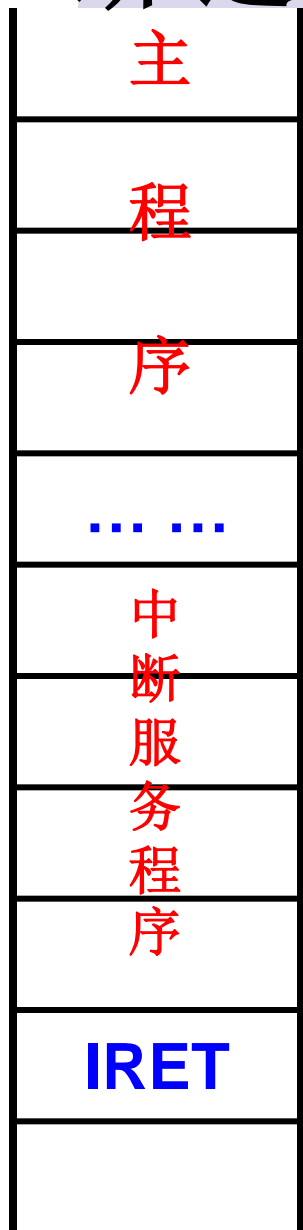
- 中断是指CPU正常运行程序时，由系统内/外部非预期事件或程序中预先安排好的指令性事件引起的，CPU暂停当前程序的执行，转去为该事件服务的程序中执行，服务完毕后，再返回原程序继续执行的过程。

动画演示：
[8-4. swf](#)

●注意:

- 中断是一个CPU执行程序的变化过程;
- 所有能引起中断的事件均称为中断源;
- 处理中断事件的中断服务程序是预先设置好的;
- 结束中断返回原程序时，要以原状态返回暂停处继续执行。

中断过程示意



← IP

主程序

断点

中断请求

中断响应

中断返回

中断服务程序

IRET

有关中断

- 中断处理过程是由**硬件和软件结合**来完成的。
- 为什么要使用中断？
 - 解决速度问题，使**CPU**和**I/O**并行工作；
 - 对意外情况(如磁盘损坏、运算溢出等)能够及时处理。
 - 在实时控制领域中，及时响应外来信号的请求。
- 中断系统需要解决的问题
 - 各中断源如何向**CPU**发出中断请求；
 - 有多个中断源时，如何确定优先级；
 - **CPU**在什么条件、什么时候、以什么方式响应中断；
 - 如何保护/恢复现场、如何转入中断服务程序的入口地址；
 - 如何处理多重中断。

CPU的中断处理流程

1. 中断请求

- CPU在结束一个指令周期后，检测中断请求信号；

第一个问题：
CPU公操作时检测保存中断请求信号的寄存器

2. 中断响应

- 关中断；由硬件自动完成
- 保护断点现场；由软硬件共同完成
- 判断中断源，获取中断向量；
- 根据中断向量转入中断服务程序执行。

第三个问题：
避免再次中断影响当前中断响应，屏蔽中断源；
可实现中断优先权管理

3. 中断服务

- 保护CPU现场；对CPU内部寄存器的保护
- 执行中断服务程序；
- 开中断；由软件完成
- 恢复CPU现场；

第二个问题：
为保证CPU能够正常返回主程序

4. 中断返回

- 恢复断点现场，返回主程序继续执行；

第四个问题：
中断处理过程由软硬件结合完成

中断向量

●中断向量:

- 中断服务程序的入口地址，包括段地址和段内偏移地址，共4个字节的内容；
- CPU响应中断时，将中断源对应的中断向量送入CS、IP寄存器中，以跟踪中断服务程序的执行。

●中断向量表:

- 集中存放系统中所有中断向量的存储区。
- 8086 PC机中，将存储器物理地址为0~3FFH的1024个单元作为中断向量表，每个向量占用4个字节，可容纳256个中断向量。

中断向量表

类型号 0	偏移地址	0000 : 0000
		0000 : 0001
	段地址	0000 : 0002
		0000 : 0003
类型号 8
	偏移地址	0000 : 0020
		0000 : 0021
	段地址	0000 : 0022
		0000 : 0023

类型号 255	偏移地址	0000 : 03FC
		0000 : 03FD
	段地址	0000 : 03FE
		0000 : 03FF

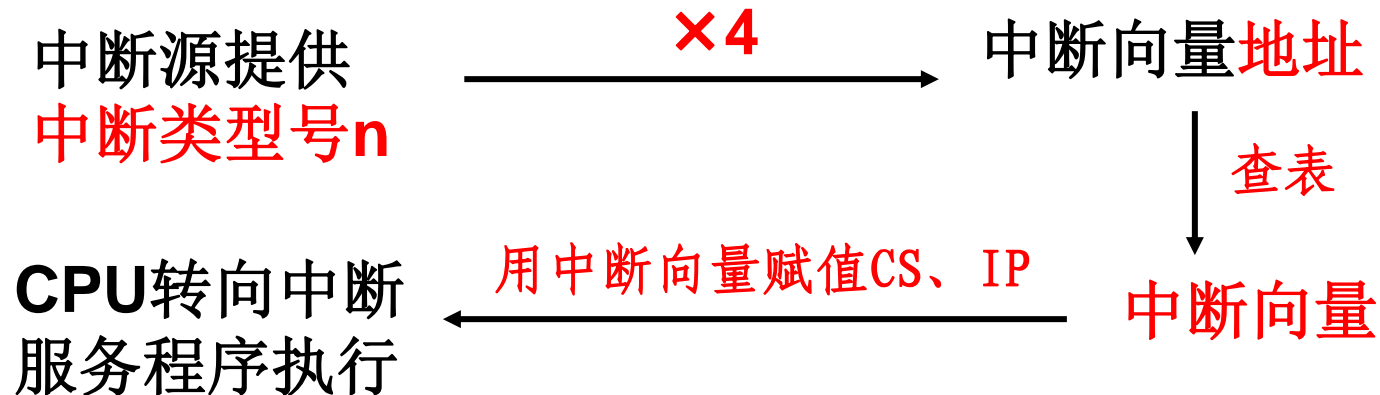
中断类型号

每个中断向量
在中断向量表
中的序号，值
为0-255。

中断类型号与中断向量

中断类型号 (n) $\times 4$ = 中断向量在表中的偏移地址
如: $n=8$, 则应从向量表20H~23H中取出中断向量

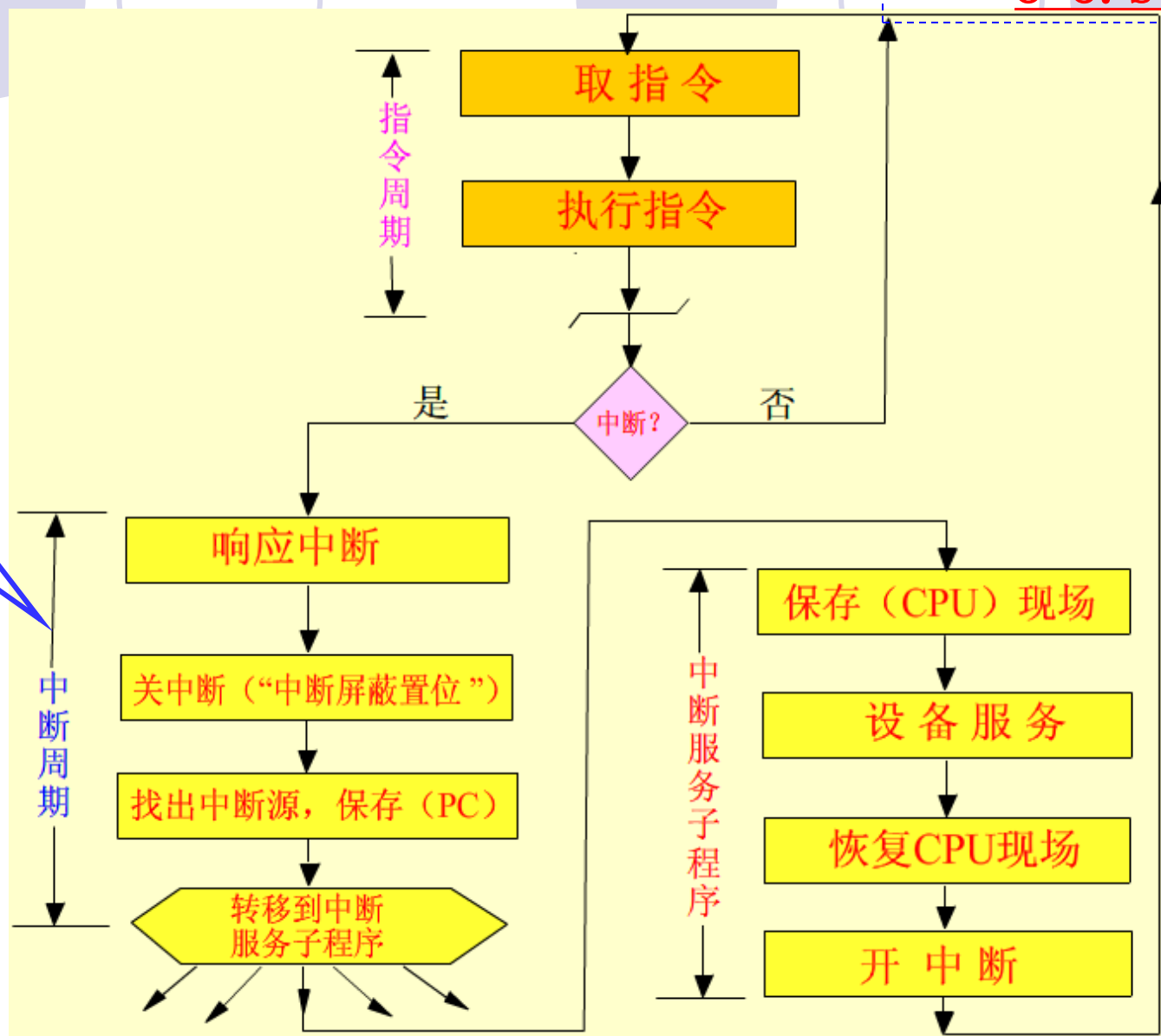
CPU使用向量中断的过程:



中断处理流程

动画演示：
8-5. swf

由中
断隐
令执
行



中断处理过程中的问题

● 第一个问题

- CPU只有在一条指令执行完毕转入公操作时时，才能处理外设的中断请求。

● 第二个问题

中断返回时再恢复断点状态

- 为了保证中断服务程序执行完毕后，能够正确地返回到原断点位置，则必须保存PC和当前CPU的状态到堆栈中。

● 第三个问题

对中断源的优先权管理

- CPU在中断响应期间，还可能新的中断请求。为了不致于造成混乱，中断管理部件中必须有中断屏蔽触发器，在中断响应期间屏蔽其他中断源。

● 第四个问题

- 中断处理过程是由硬件和软件结合来完成的；
- 中断周期由硬件实现，中断处理程序由机器指令序列实现。

8.3.2 程序中断方式的基本I/O接口

●接口内部组成

- 数据缓冲寄存器;
- 就绪触发器RD、忙状态触发器BS、允许中断触发器EI;
- 中断向量产生逻辑;

●CPU的相应部件

- 中断请求触发器IR、中断屏蔽触发器IM;

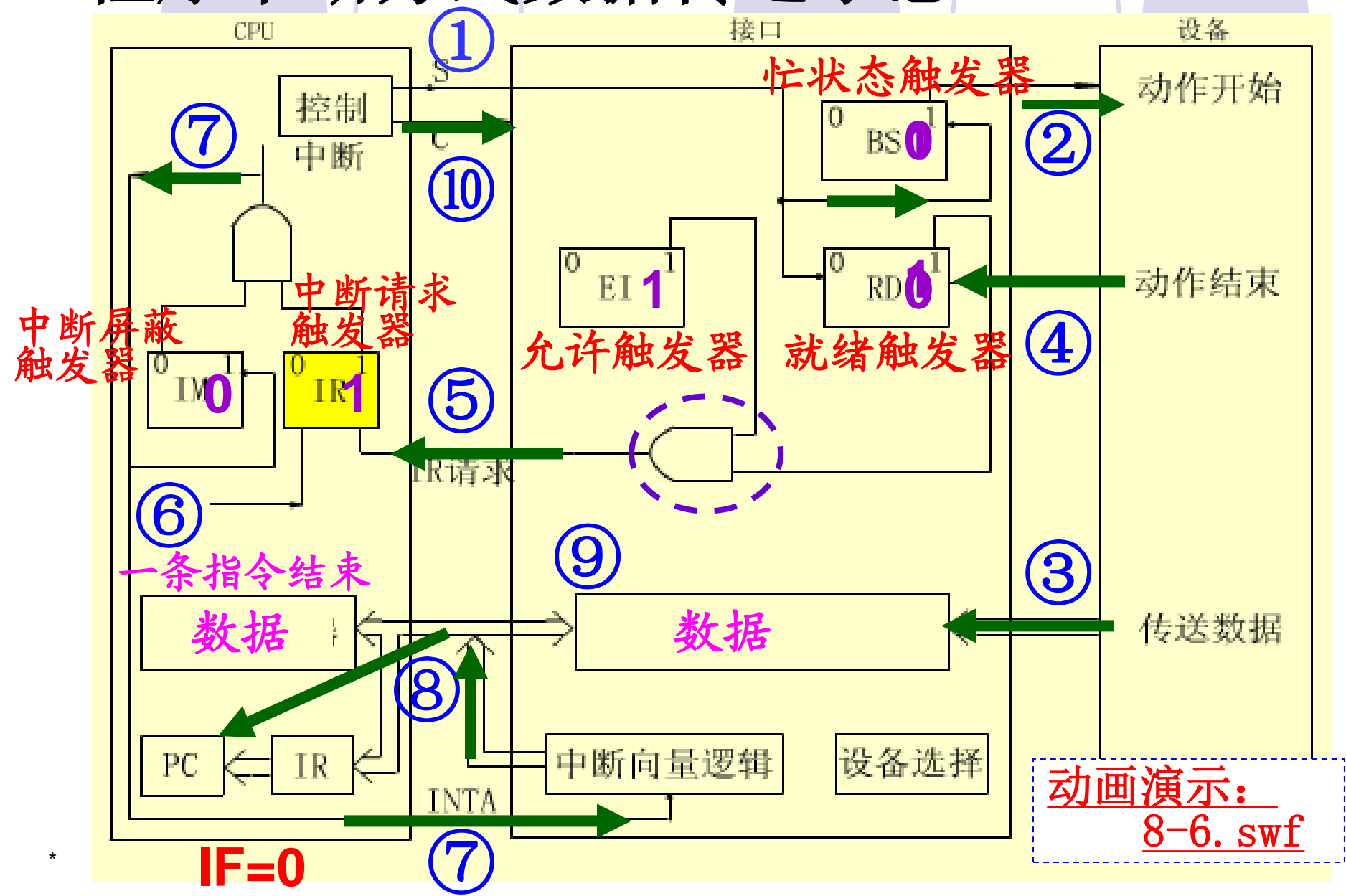
●程序中断的数据传送过程:

CPU启动外设

→外设准备就绪后, 向CPU请求中断

→CPU接受请求, 完成数据传送

程序中中断方式数据传送示意



中断处理的策略

- 根据计算机系统中断处理的策略的不同，可分为：

- 单级中断系统

- 所有的中断源都属于同一个级别，不允许有中断嵌套；

- 多级中断系统

- 中断源分为不同的级别，可以发生中断嵌套，高优先权的中断源请求可以打断低优先权的中断服务；

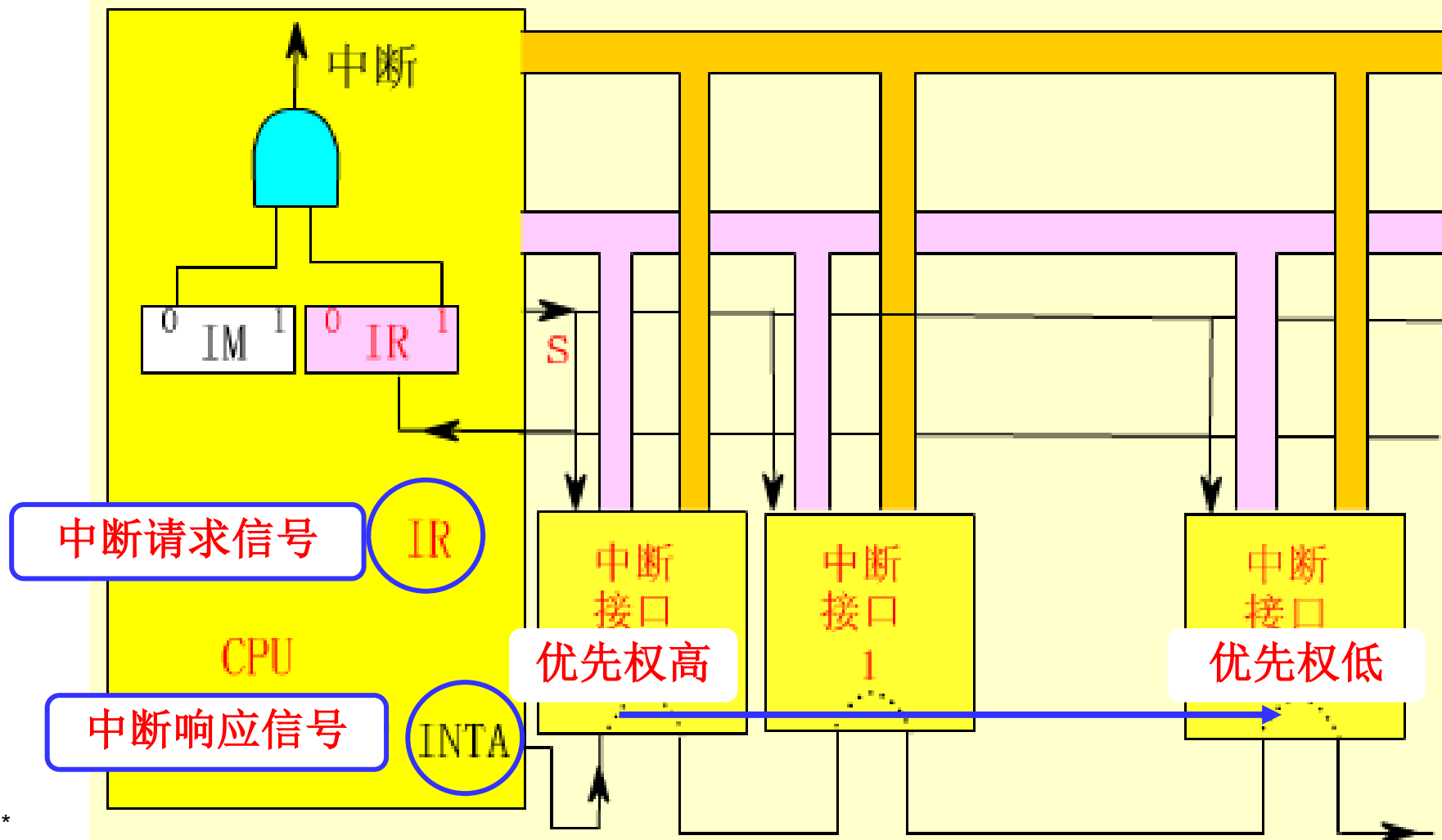
- 实现方法

- 进入中断服务时的中断屏蔽设置；

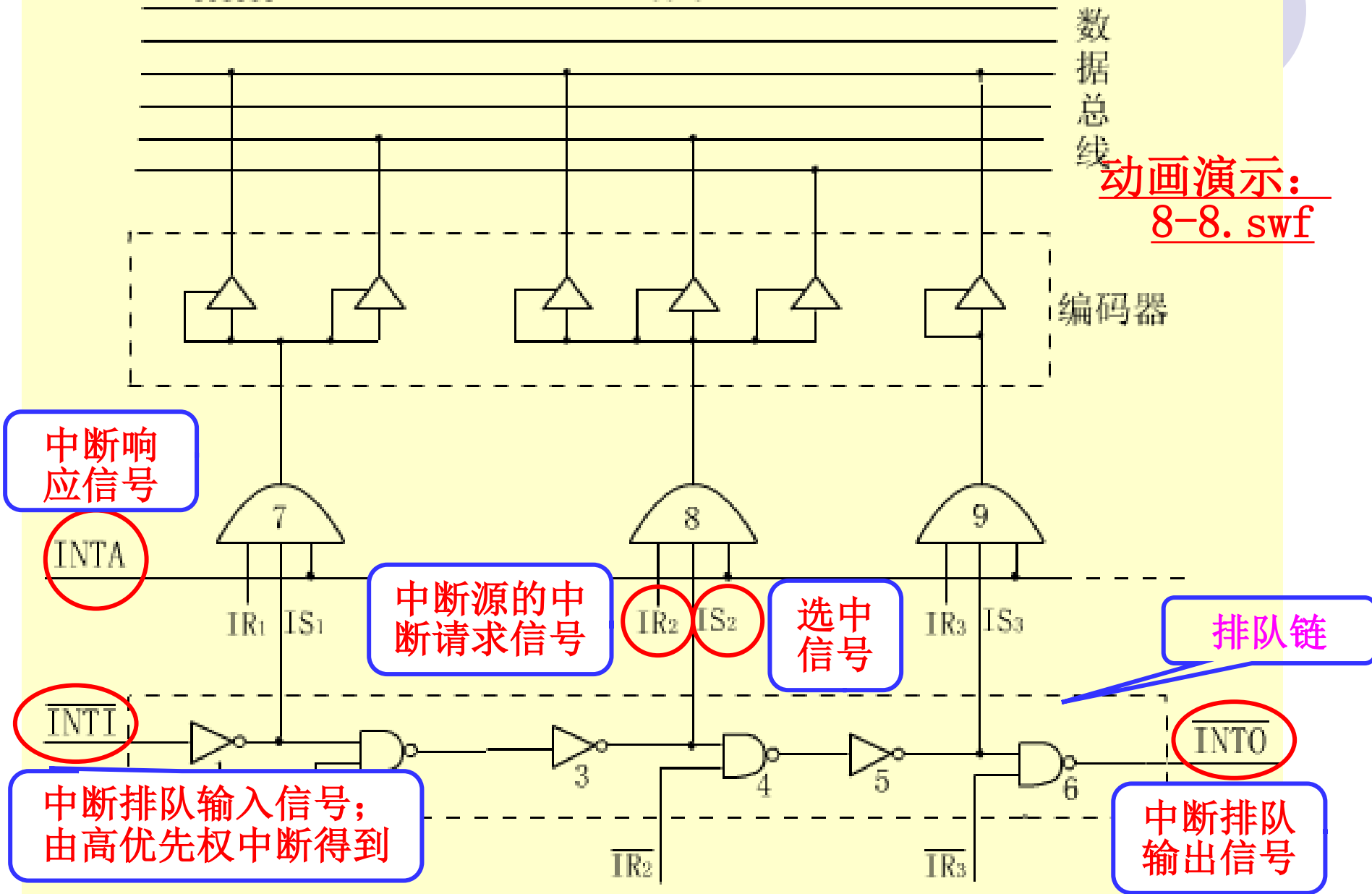
8.3.3 单级中断

动画演示：
8-7.swf

- 单级中断中，所有中断源属于同一个级别，不允许中断嵌套。



2、单级中断源的识别——串行排队链法



中断向量的产生

1. 向量中断

- 由硬件直接产生一个与该中断源对应的**向量地址**;

 - ◆ 向量地址——中断源的中断服务程序入口地址;

- 要求：在硬件设计时考虑所有中断源的向量地址。

2. 位移量中断

- 由硬件直接产生一个**位移量**，该位移量加上CPU中某寄存器中的基地址，得到中断处理程序的入口地址。

3. 向量地址转移

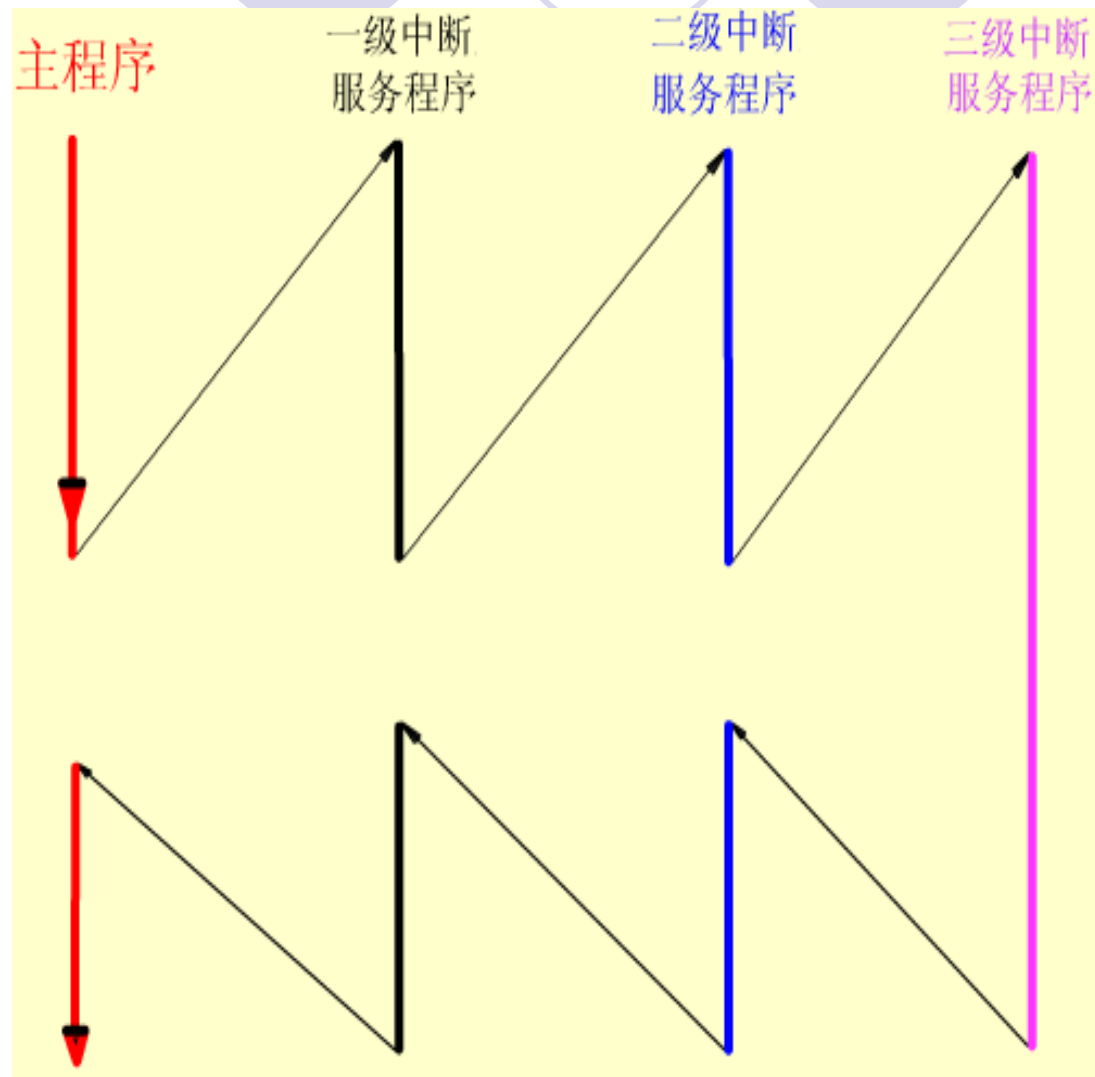
- 由硬件直接产生一个该中断源对应的**固定地址码**，该地址码中存放可转入中断服务程序的入口地址。

8.3.4 多级中断

- 当CPU正在执行某个中断服务程序时，另一个中断源又提出了新的中断请求，而CPU再次响应这一新的请求；

- 暂停当前的中断服务程序，转去执行新的中断服务程序。

- 多级中断中需要进行优先权的控制和判断。



一维多级中断结构

- 一维多级中断结构

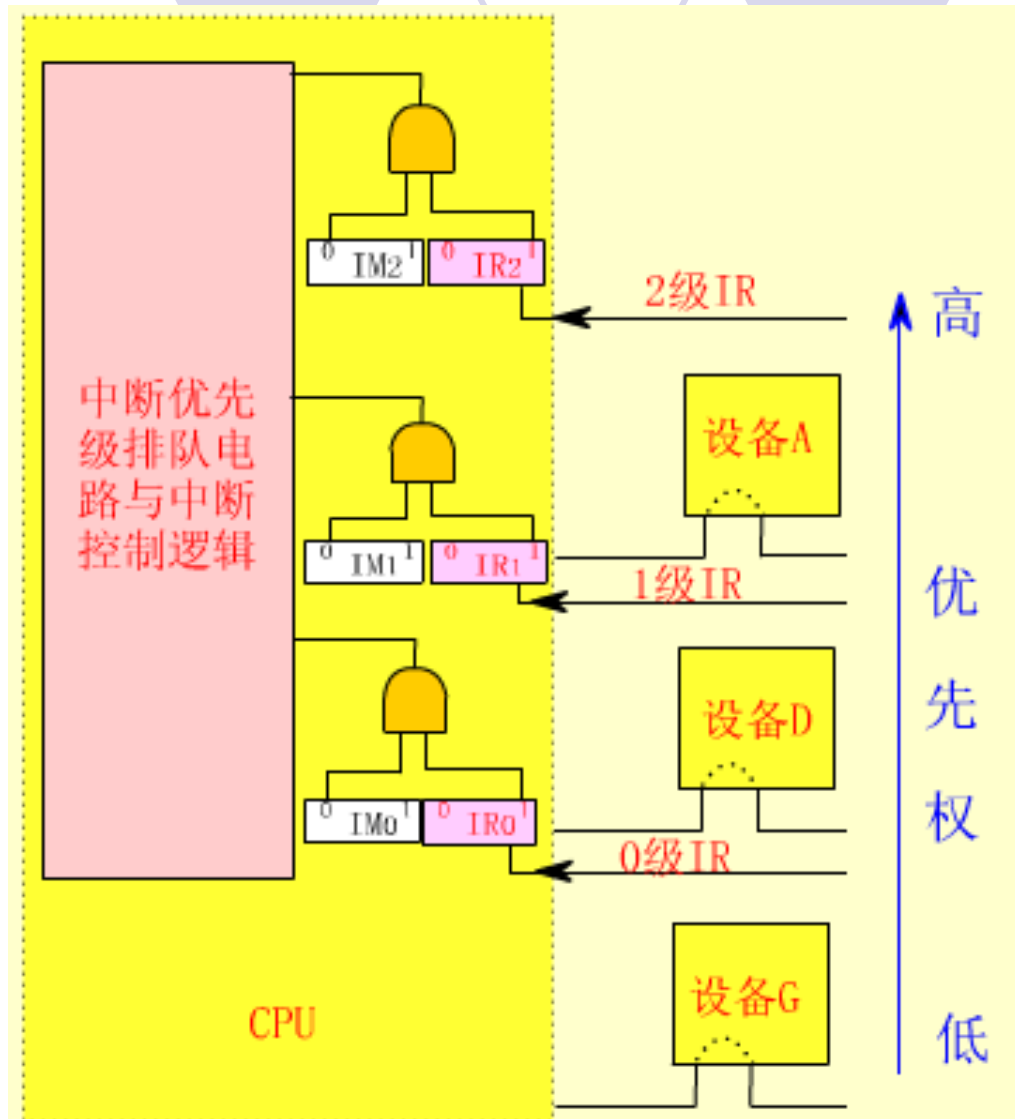
- 每级仅有一个中断源；

- 中断屏蔽触发器IM

- 其状态决定对应级别的中断源是否能够被响应；

- 中断优先权控制

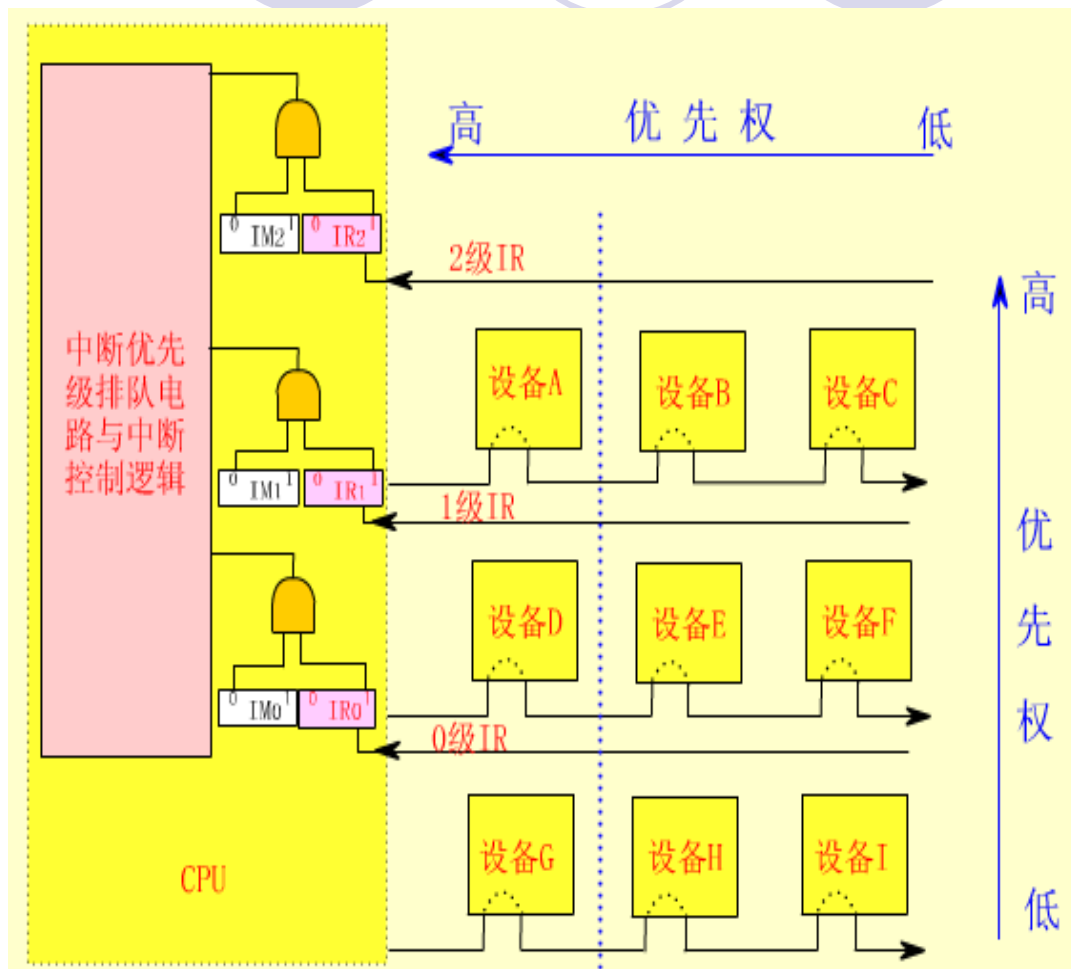
- 当某中断源请求被响应时，将同时设置低级中断源的IM=1，而高级中断源的IM=0，可以发生中断嵌套。



二维多级中断结构

动画演示:
8-9. swf

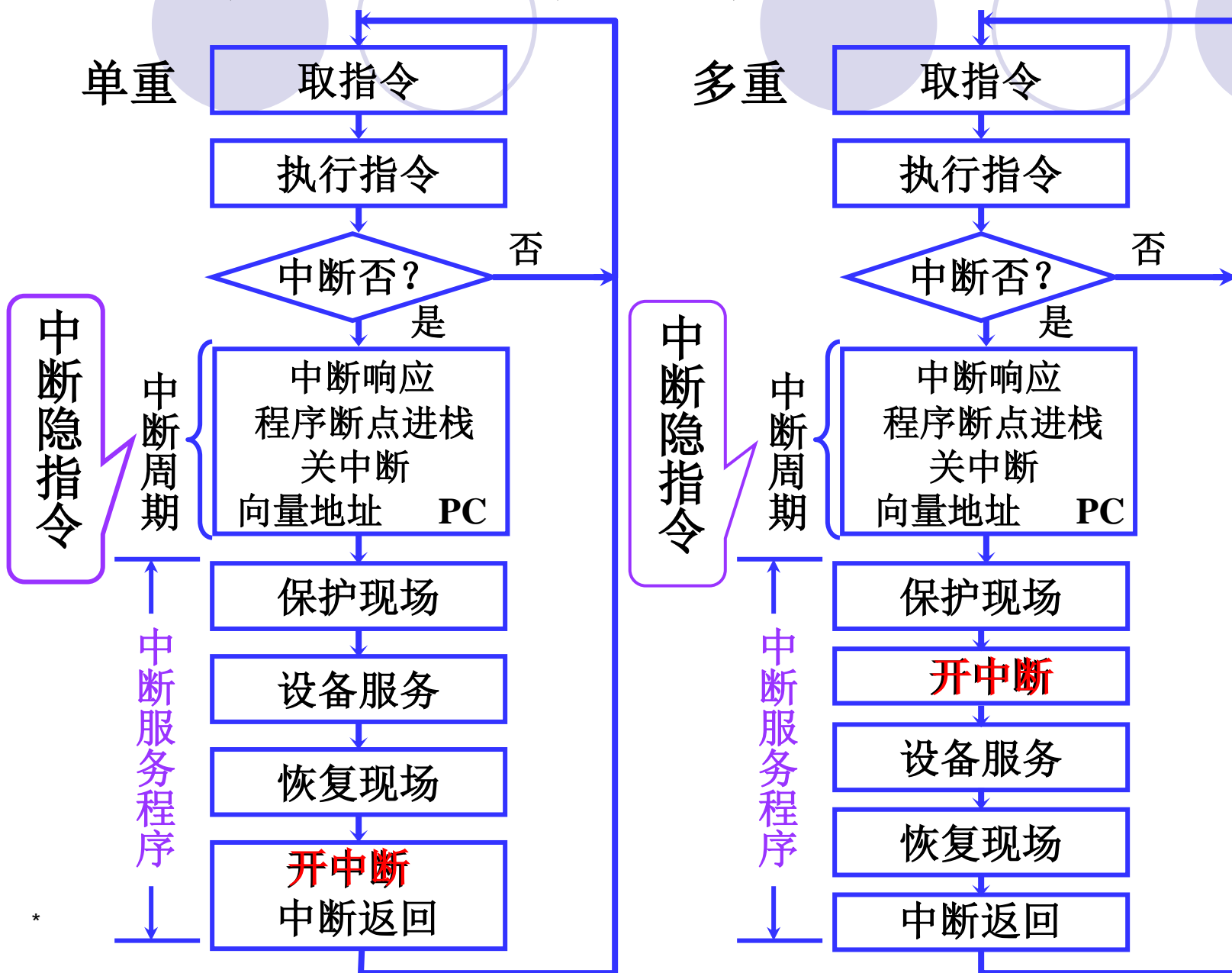
- 二维多级中断结构
 - 每级包含多个中断源;
 - 同级的中断源的服务是不能嵌套的;
- 若设备E、F、H同时请求中断，则先响应设备E;
- 此时，若设备D请求中断，则不响应;
- 若设备B请求中断，则打断设备E的中断服务，嵌套响应设备B;



CPU内部的中断请求/屏蔽寄存器

- 对于每一个中断源，CPU内部会有一个中断请求触发器和一个中断屏蔽触发器，用于管理当前该中断源的状态；
 - 中断请求触发器：1——有请求；0——无请求；
 - 中断屏蔽触发器：1——被屏蔽；0——未被屏蔽；
- 若系统中共有n级中断，则有两个n位中断请求寄存器和中断屏蔽寄存器；
 - CPU响应中断时，系统需要进行优先权控制，实现正常的中断嵌套；
 - ◆ 置“1”（关闭）本级和低级的中断屏蔽触发器；
 - ◆ 清“0”（开放）更高级的中断屏蔽触发器；

单重中断和多重中断的服务程序流程



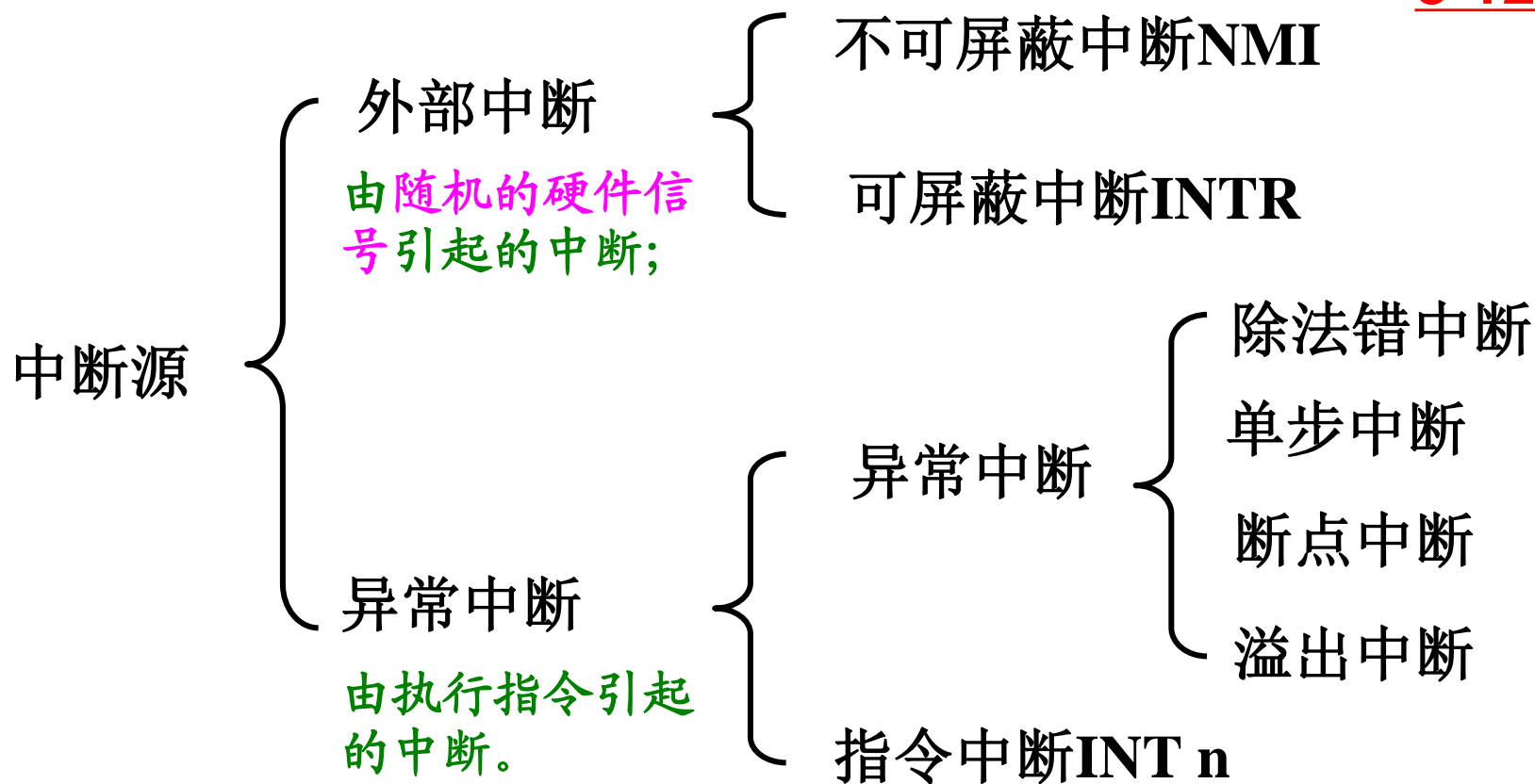
中断屏蔽技术举例

例：某机有5个中断源L0~L4，按中断响应优先级从高到低为L0→L1→L2→L3→L4，现要求将中断处理次序改为L1→L3→L4→L0→L2，请写出各中断源的屏蔽字。

中断源	屏蔽字 (L0L1L2L3L4)
L0	11111
L1	00100
L2	10111
L3	10101
L4	

8.3.6 PC系列机中断机制

动画演示:
8-12.swf



不可屏蔽中断NMI

- 由**CPU**的**NMI**引脚引入该中断请求，**CPU**得到该中断请求**立即响应**，不需要**CPU**发响应信号；
- 中断类型号固定为**2**；
- 中断优先权高于可屏蔽中断**INTR**，一般用于系统对紧急情况的处理，用户不能使用；
- 常见的**NMI**中断：
 - 系统**RAM**奇偶校验错、**I/O**通道校验错、协处理器出错等。

可屏蔽中断INTR

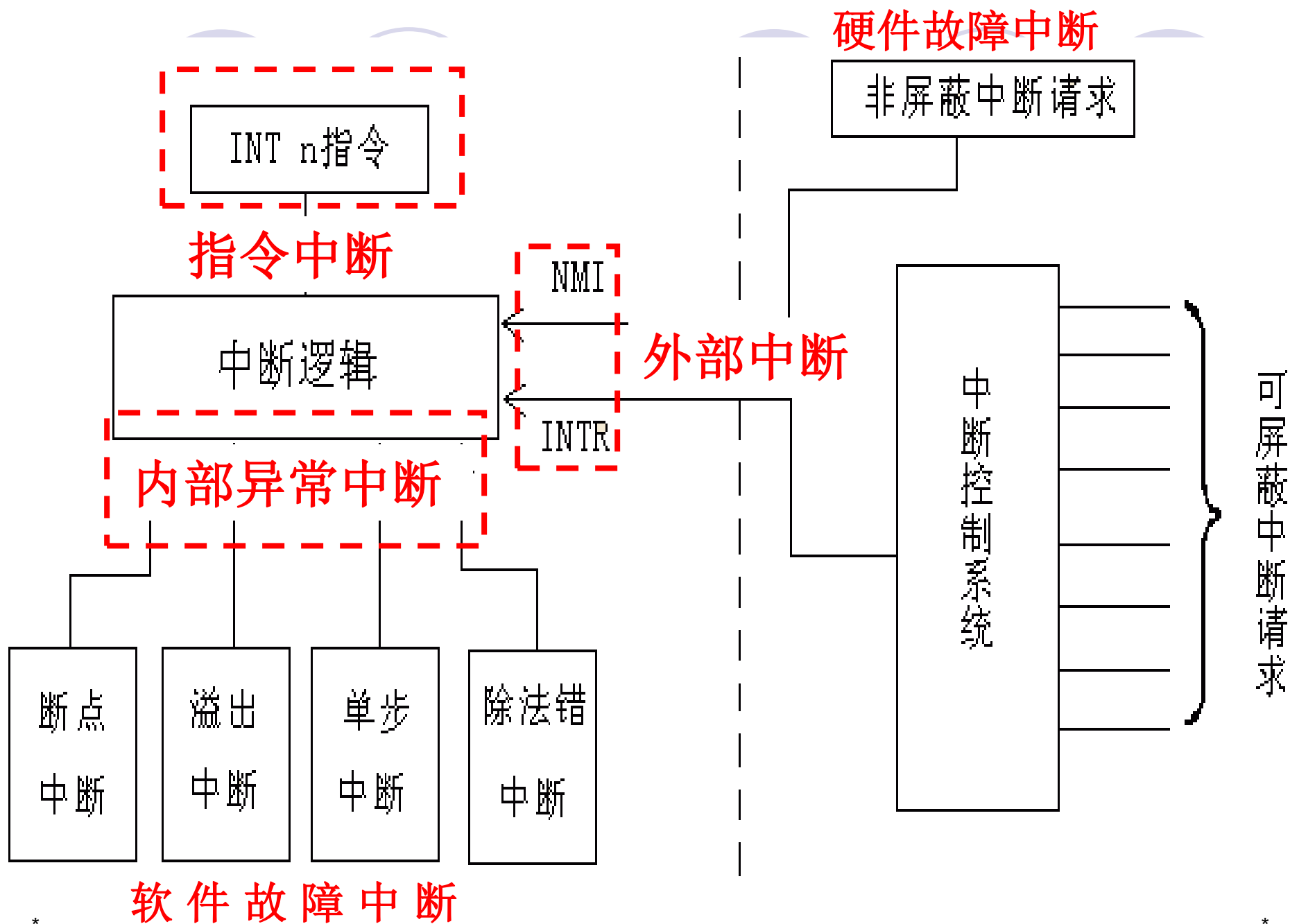
- 由CPU的INTR引脚引入该中断请求；
 - 该中断请求只有在CPU的中断允许标志位IF=1时，CPU才会通过发回响应信号的方式响应；
 - CPU对INTR中断的响应控制可由STI（允许响应INTR中断）、CLI（禁止响应INTR中断）两条指令完成。
- 此类中断的中断类型号由中断源提供；
- 常见的INTR中断：
 - 外部设备的中断请求，如I/O设备的数据传送请求，用户可自行设计；

指令中断INT n

- 由**CPU**执行在程序中预先安排好的中断指令引起的；
 - 该中断是可预期的、且不可屏蔽的；
- 处理该中断时，**CPU**不需要发响应信号；
- 指令的操作数字段（**n**）即为中断类型号；
 - 中断类型号的范围为 $5 \leq n \leq 255$ ；
- 这类中断包括**BIOS**中断、**DOS**中断以及一些未定义的自由中断（可由系统扩充或根据应用需要自定义）。

异常中断

- 由**CPU**内部的突发事件引起的中断，是不可屏蔽中断；
- 其中断类型号固定；
 - 除法错中断——类型号为0；
 - ◆ 执行除法指令时，除数为0，或除得的商大于规定位数；
 - 单步中断——类型号为1；
 - 断点中断——类型号为3；
 - 溢出中断——类型号为4；
- 该类中断的实质
 - 由**CPU**运行过程中的突发事件引起系统在适当位置自动添加一条中断指令，然后由**CPU**执行该指令引起的中断；
- 同指令中断一样，处理该中断时，**CPU**不需要发响应信号。



各类中断的比较

	中断产生的特点	中断类型号的来源	CPU是否应答	可屏蔽性
NMI中断	随机性 突发性	系统固定为2	不需要	不可屏蔽
INTR中断	随机性 突发性	由中断源提供	需要	可屏蔽（IF）
指令中断	可预期	由指令提供	不需要	不可屏蔽
内部中断	随机性 突发性	系统固定为 0、1、3、4	不需要	不可屏蔽 (除单步中断TF)

8.4 DMA方式

8.4.1 DMA的基本概念

- **DMA（直接内存访问）方式** 磁盘与主存之间的数据传送
 - 完全由硬件控制执行I/O交换的工作方式；
 - **DMAC完全接管系统总线的控制权**，利用总线控制数据在主存和外设之间的直接传送，所交换的数据不经过**CPU**。
- **DMA特点**
 - 速度快，但硬件复杂度高。
- **DMA方式中，DMAC的工作：**
 - 接受外设的**DMA**请求，并提交给**CPU**；
 - 接管总线的控制权，控制主存和外设之间的数据传送；
 - 完成数据传送后，交还总线使用权。

8.4.2 DMA传送方式

- **DMA传送时，CPU不能使用总线，会影响CPU的工作效率；**

- 当**CPU**需要访存，或者访问外设时，程序会阻塞；

- 可使**DMAC**与**CPU**分时复用总线；

- **常用的DMA传送方式**

- **停止CPU访问方式**

- ◆ 早期使用的一种方式；

- **周期挪用方式（周期窃取方式）**

- ◆ 适用于外设读写周期大于主存的存取周期的系统中；

- **DMA与CPU交替访问方式（透明DMA方式）**

- ◆ 适用于**CPU**周期大于主存存取周期的系统中；

每次挪用的一个总线周期，对于DMAC来说需要2-5个周期，用于总线控制权的申请、建立和归还。

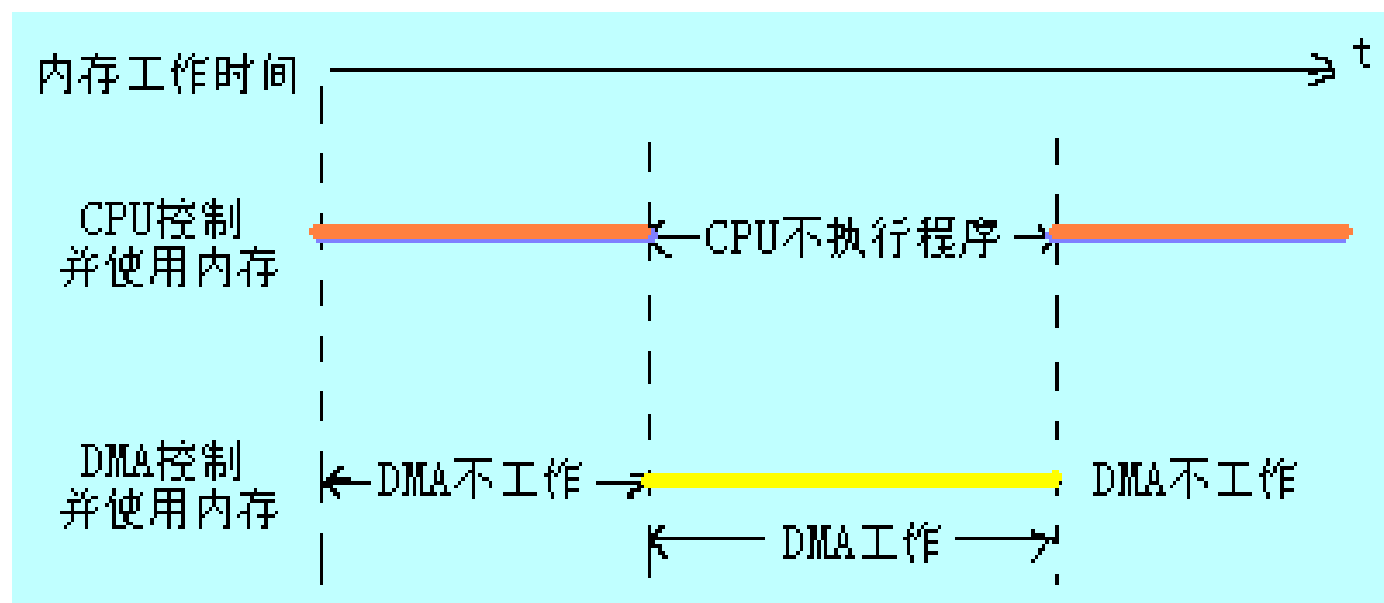
停止CPU访问方式

● 工作过程

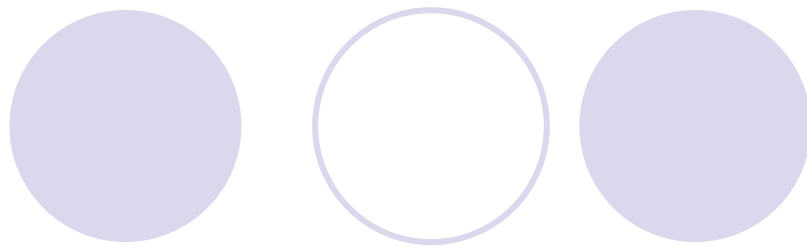
- DMA传送期间，由DMAC长期占用总线，CPU不能访存，只有DMA传送结束时才将总线归还给CPU使用；

● 特点

- 控制简单，适于高速外设的数据成组传送；
- 内存使用效率不高。(一般外设的读写周期远大于主存的存取周期)



周期挪用方式

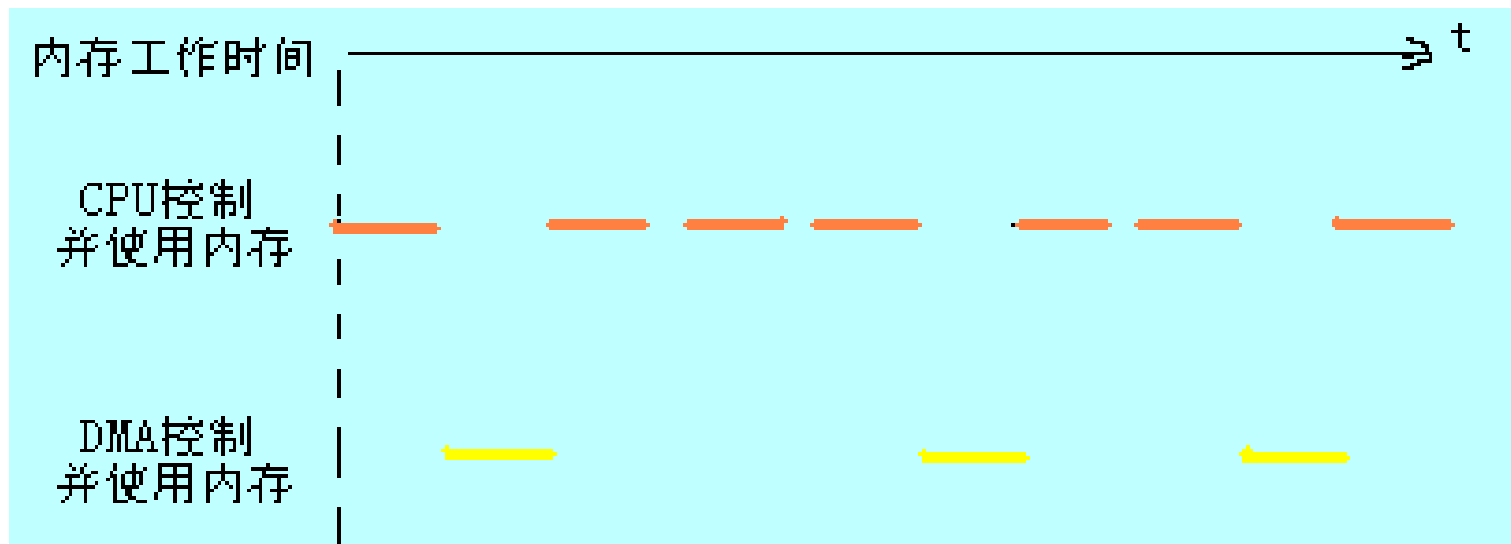


● 工作过程

- 每次**DMA**传送，都是通过外设挪用一个或几个主存周期，分别完成数据的成块传送。

● 特点：

- 主存使用效率较高，常用；
- ◆ **CPU**和外设访存冲突时，外设优先访问。



DMA与CPU交替访存方式

- 工作过程

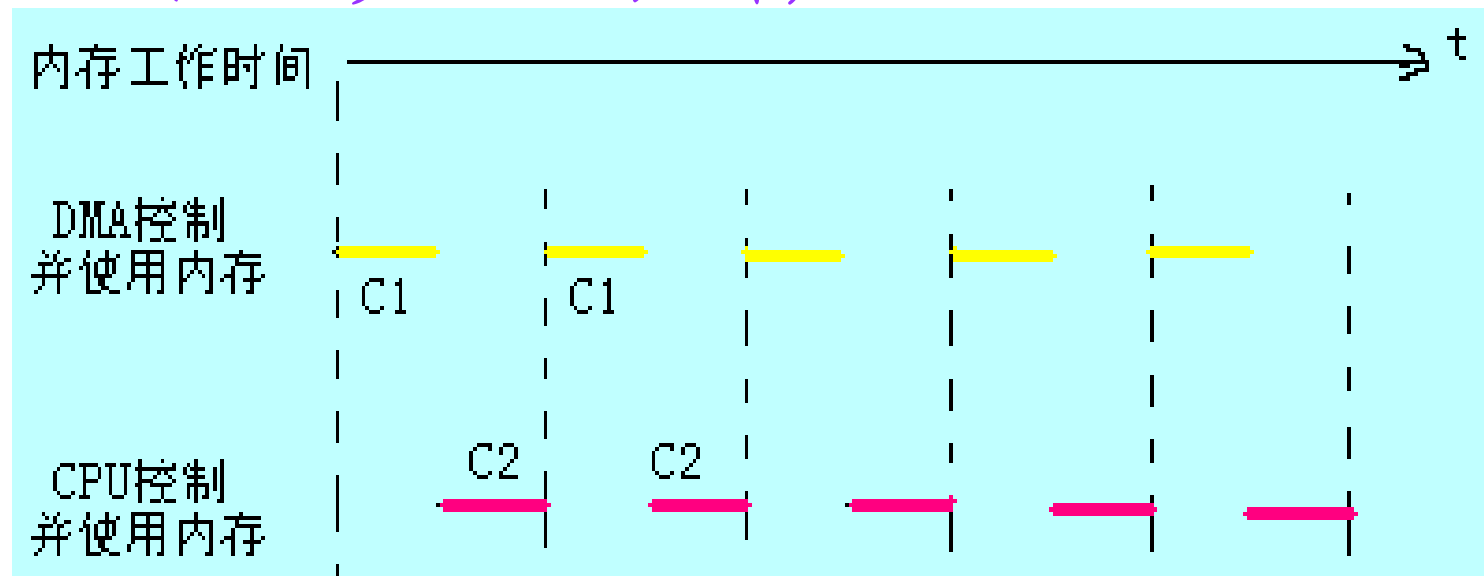
- 将CPU的工作周期一分为二，分别用于DMA和CPU访存；

- 特点：

- 对主存的访问时间不会发生冲突，也不需要总线控制权的申请建立和归还过程；

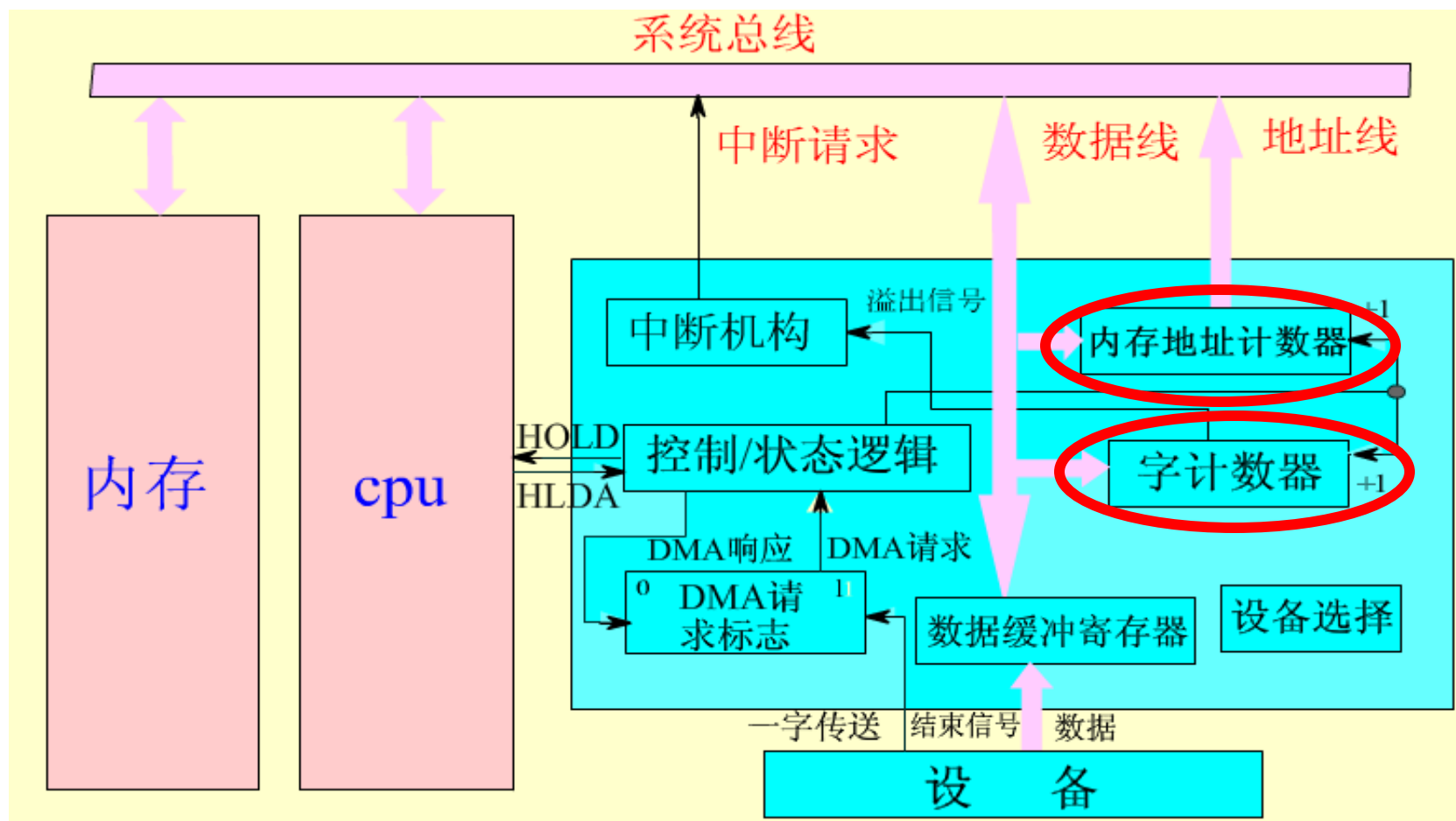
- 透明DMA方式

- DMA传送不影响CPU的工作；



8.4.3 DMA控制器

1、DMA控制器的基本组成



2、DMA数据传送过程

●DMA请求

○DMAC向CPU请求总线使用权;

●DMA响应

○CPU给DMAC分配总线使用权;

●DMA传送

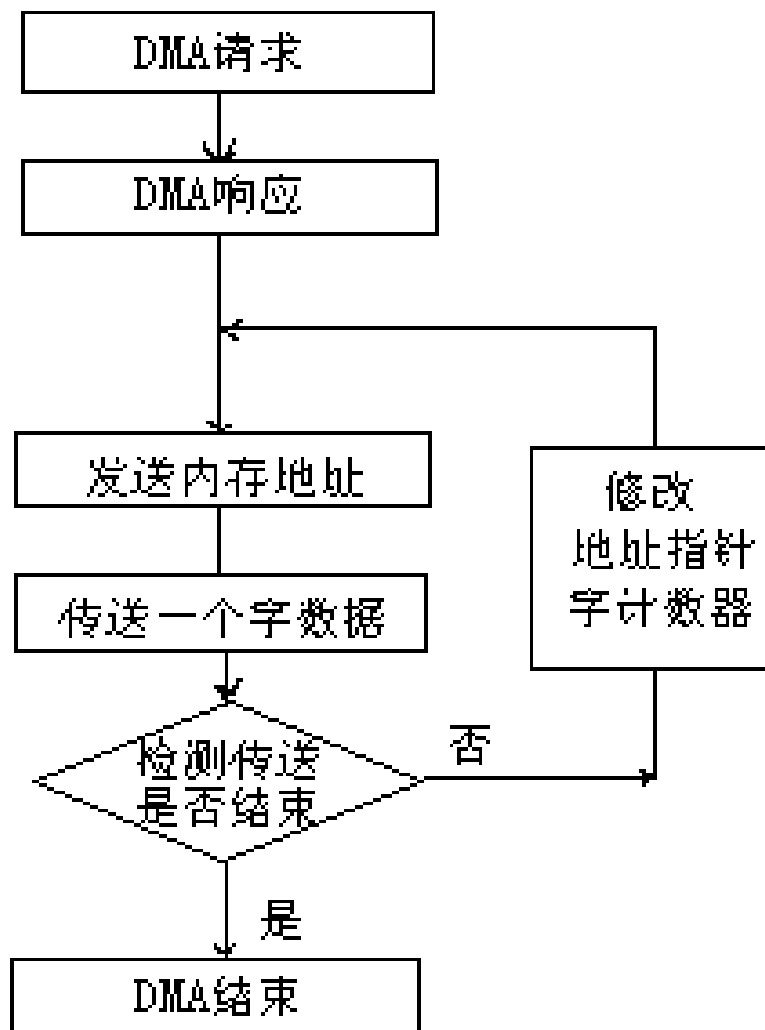
○DMAC寻址外设和存储器;

○DMAC控制外设与主存之间的数据传送,并检测是否传送结束;

●DMA结束

○DMAC归还总线使用权。

之前要有对**DMAC**芯片的编程操作——**传送前的预处理**



DMA方式和中断控制方式的区别

- 数据传送的实现方式

- 中断——程序传送； DMA——硬件实现；

- CPU响应请求的时间

- 中断——一个指令周期结束； DMA——一个总线周期结束；

- 请求的目的

- 中断——CPU的服务； DMA——总线的使用权；

- 是否需要保护现场

- 中断——需要； DMA——不需要（CPU不参与数据传送）；

- DMA的优先权高于中断的优先权。

8.5 通道方式

8.5.1 通道的功能

8.5.2 通道的类型

8.5.3 通道结构的发展

8.5.1 通道的功能

1. 通道的功能

- 执行通道指令、组织外围设备和内存进行数据传输；
- 按I/O指令要求启动外围设备，向CPU报告中断。

2. CPU对通道的管理

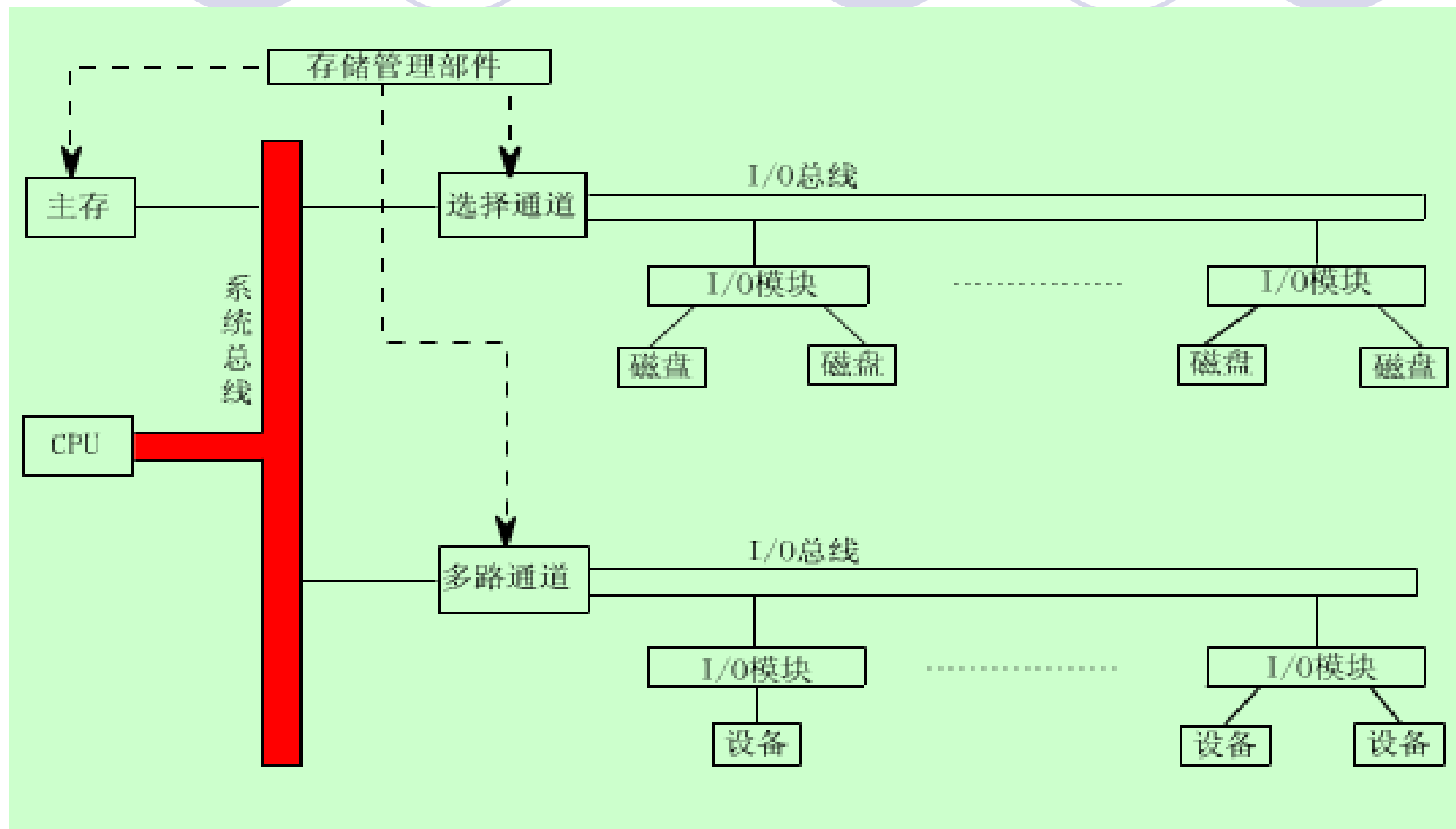
- CPU通过执行I/O指令控制通道的数据传送；
- CPU处理来自通道的中断，实现对通道的管理；
 - ◆ 一种是数据传送结束中断；一种是故障中断。

3. 通道对设备控制器的管理

- 通道使用通道指令控制I/O模块进行数据传送操作；
- 通道接收通道状态字了解外设的状态。

通道结构

动画演示:
[8-20.swf](#)



通道指令

1. 通道指令包含的内容

- （一）操作码：它规定指令所执行的操作，如读、写等。
- （二）内存地址：标明数据传送时内存的首址。
- （三）计数：表示传送数据的字节数。
- （四）通道程序结束位**R0**，表示通道程序是否结束。
- （五）记录结束标志**R1**，表示所处理的记录是否结束。

8.5.2 通道的类型

● 选择通道 高速通道

- 物理上可以连接多个设备，但这些设备不能同时工作；
 - ◆ 某一段时间内通道只能选择一个设备进行工作。

● 多路通道 多路转换通路

- 同一时间能处理多个I/O设备的数据传输。

○ 多路通道的类型

◆ 数组多路通道 连接高速设备

利用设备寻址的空隙为其他设备服务；

◆ 字节多路通道 连接低速设备

利用设备数据传送中字节之间的空隙为其他设备服务；

8.5.3 通道结构的发展

- 通道结构的I/O处理器

- 输入输出处理器(IOP)

- IOP可以和CPU并行工作，提供高速的DMA处理能力，实现数据的高速传送；

- IOP不独立于CPU工作，它是主机的一个部件；

- 外围处理机（PPU）

- PPU独立于主机工作的。

8.6 通用I/O标准接口

- 并行I/O标准接口**SCSI**（小型计算机系统接口）
 - 设计思想来源于**IBM**大型机系统的I/O通道结构；
 - 目的是使**CPU**摆脱对各种设备的繁杂控制。
 - 它是一个高速智能接口，可以混接各种磁盘、光盘等设备。
- 高速串行I/O标准接口**IEEE 1394**
 - 苹果公司开发的高速串行I/O标准接口；
 - 具有数据传送速度快、实时性好、连接方便等特点。
 - **1394**接口在家电等消费类设备的连接应用方面有很好的前景。

09年考研真题

22. 下列选项中，能引起外部中断的事件是（A）

- A. 键盘输入
- B. 除数为0
- C. 浮点运算下溢
- D. 访存缺页

09年考研真题

43. (8分) 某计算机主频为**500MHz**，**CPI**为**5**（即执行每条指令平均需要**5**个时钟周期）。假设某外设的数据传速率为**0.5MB/S**，采用中断方式与主机进行数据传送，以**32**位为传输单位，对应的中断服务程序包含**18**条指令，中断服务的其他开销相当于**2**条指令的执行时间。

(1) 在中断方式下，**CPU**用于该外设**I/O**的时间占整个**CPU**时间的百分比是多少？

(2) 当该外设的数据传输率达到**5MB/S**时，改用**DMA**方式传送数据。假定每次**DMA**传送块大小为**5000B**，且**DMA**预处理和后处理的总开销为**500**个时钟周期，则**CPU**用于该外设**I/O**的时间占整个**CPU**时间的百分比是多少？

(1) 在中断方式下，CPU用于该外设I/O的时间占整个CPU时间的百分比是多少？

● CPU每次中断所需的时钟周期数为：

○ $(18+2) * 5 = 100$

● 外设数据传输率0.5MB/s，每次中断传送32位数据；

○ 外设每秒申请中断的次数为：

$$0.5\text{MB}/4\text{B}=125000$$

● 1秒钟内用于中断的时钟周期数为：

○ $100 * 125000 = 12.5\text{M}$

● CPU用于外设I/O的时间占整个CPU时间的百分比：

○ $12.5\text{M} / 500\text{M} = 2.5\%$

(2) 当该外设的数据传输率达到**5MB/S**时，改用**DMA**方式传送数据。假定每次**DMA**传送块大小为**5000B**，且**DMA**预处理和后处理的总开销为**500**个时钟周期，则**CPU**用于该外设**I/O**的时间占整个**CPU**时间的百分比是多少？

● 1秒钟内需产生的**DMA**次数为：

○ $5\text{MB}/5000\text{B}=1000$ 次

● 1秒钟内**CPU**用于**DMA**处理的总时钟周期数为：

○ $1000*500=0.5\text{M}$ 个时钟周期

● **CPU**用于外设的时间占整个**CPU**时间的百分比为：

○ $0.5\text{M}/500\text{M}=0.1\%$

10年考研真题

21、单级中断系统中，中断服务程序执行顺序是（A）

I、保护现场 II、开中断 III、关中断 IV、保存断点

V、中断事件处理 VI、恢复现场 VII、中断返回

A、I->V->VI->II->VII

B、III->I->V->VII

C、III->IV->V->VI->VII

D、IV->I->V->VI->VII

本章作业



P269

1、2、4、12、13、14、15、16（课本）

8（作业本）