

第五章 输入输出系统

授课教师：

电子邮箱：



第五章 输入输出系统

5.1 I/O 系统简介

5.2 中断处理程序和设备驱动程序

5.3 与设备无关的I/O软件

5.4 磁盘系统及磁盘调度



5.1 I/O 系统简介

- **设备管理的对象：** 主要是I/O设备。
- **设备管理的基本任务：** 完成用户提出的I/O请求，提高I/O速率以及改善I/O设备的利用率。
- **设备管理的主要功能有：** 缓冲区管理、设备分配、设备处理、虚拟设备及实现设备独立性等。



5.1 I/O 系统简介

I/O 系统的基本功能

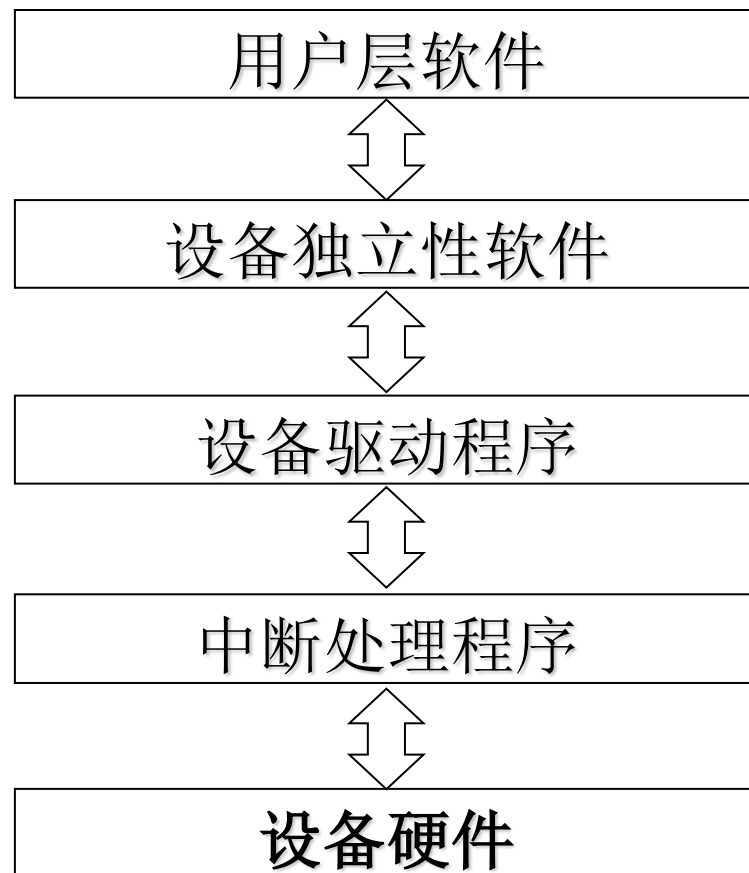
- 1) 设备分配
- 2) 设备映射
- 3) 设备驱动
- 4) I/O缓冲区的管理



5.1 I/O 系统简介

通用设备管理分层模型

► 为了使得复杂的 I/O 软件具有清晰的结构、更好的可移植性和易适应性，I/O 系统目前普遍采用层次式的结构。通常划分为四层。



I/O系统的层次结构



5.1 I/O 系统简介

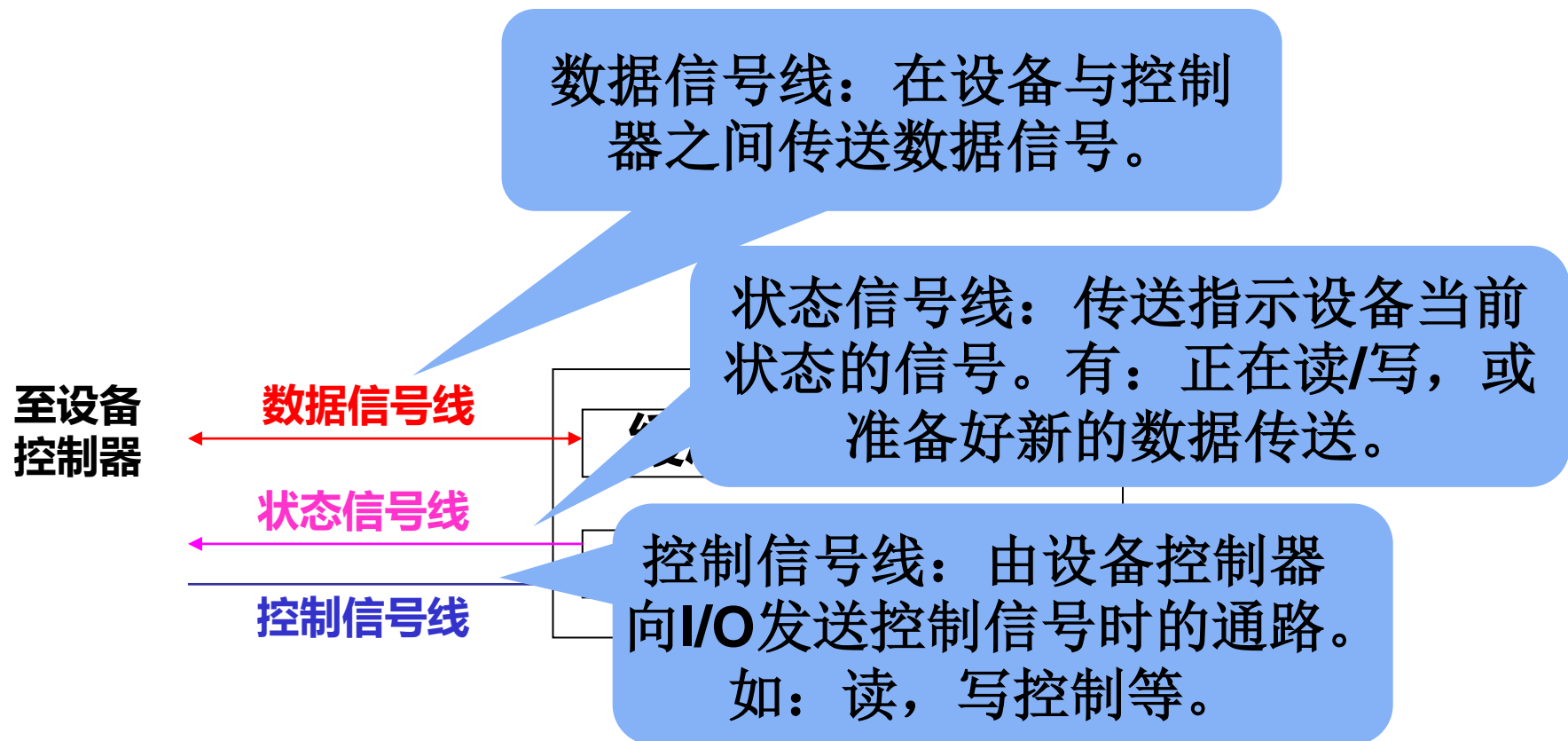
I/O设备和设备控制器等硬件

- 在I/O系统中，除了需要直接用于I/O和存储信息的设备外，还需要有相应的设备控制器和高速总线。
- 在有的大、中型计算机系统中，还配置了I/O通道或I/O处理机。



5.1 I/O 系统简介

通常，设备并不是直接与CPU进行通信，而是与设备控制器通信，因此，在设备与设备控制器之间应有一接口





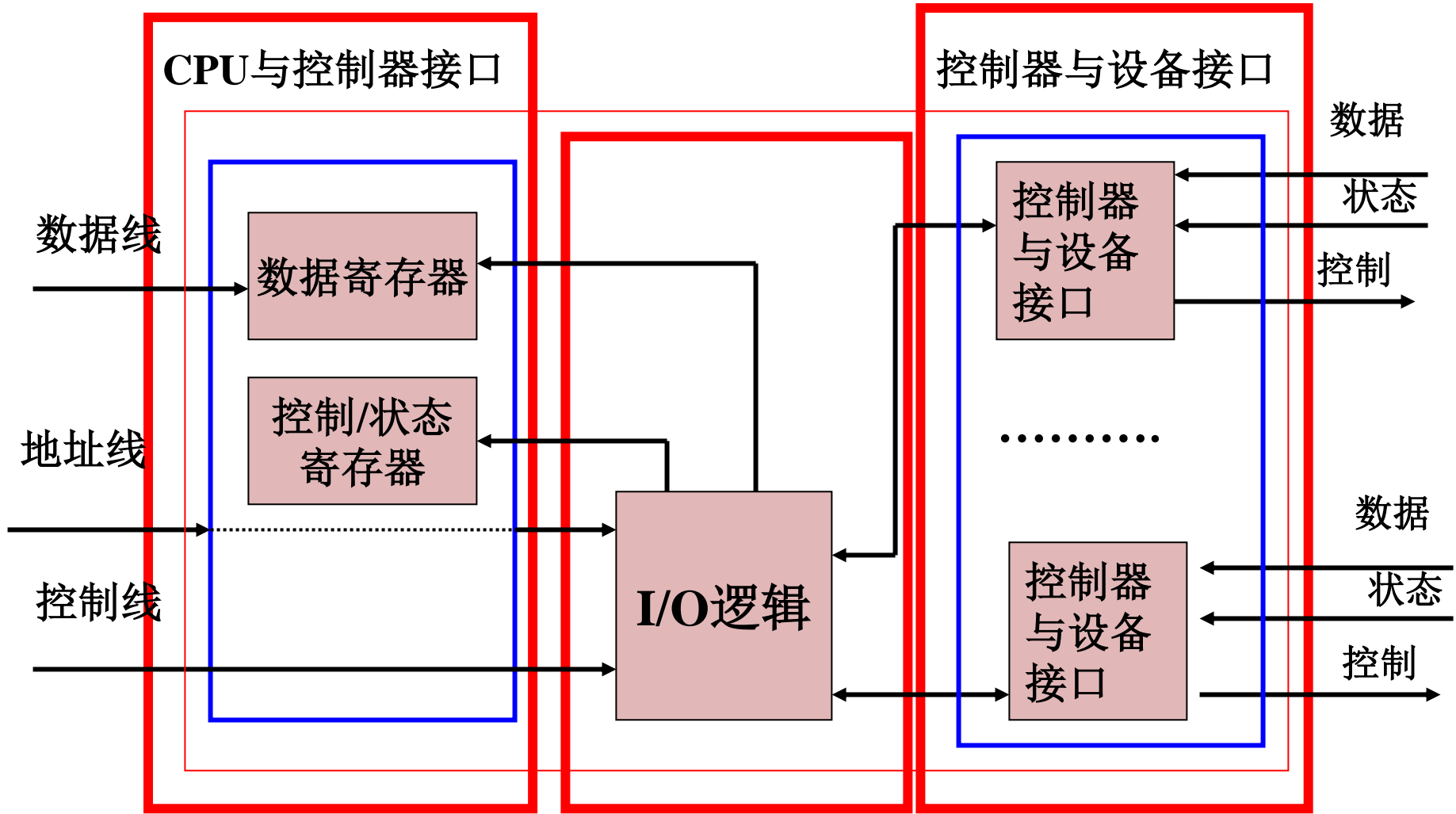
5.1 I/O 系统简介

设备控制器

- 设备控制器是CPU与I/O设备之间的接口，它接收从CPU发来的命令，并去控制I/O设备工作，以使处理机从繁杂的设备控制事务中解脱出来。
- 设备控制器主要职责是控制一个或多个I/O设备，以实现I/O设备和计算机之间的数据交换。
- 若控制器可连接多个设备时，则应含有多设备地址，并使每一个设备地址对应一个设备。



5.1 I/O 系统简介



设备控制器的组成



5.1 I/O 系统简介

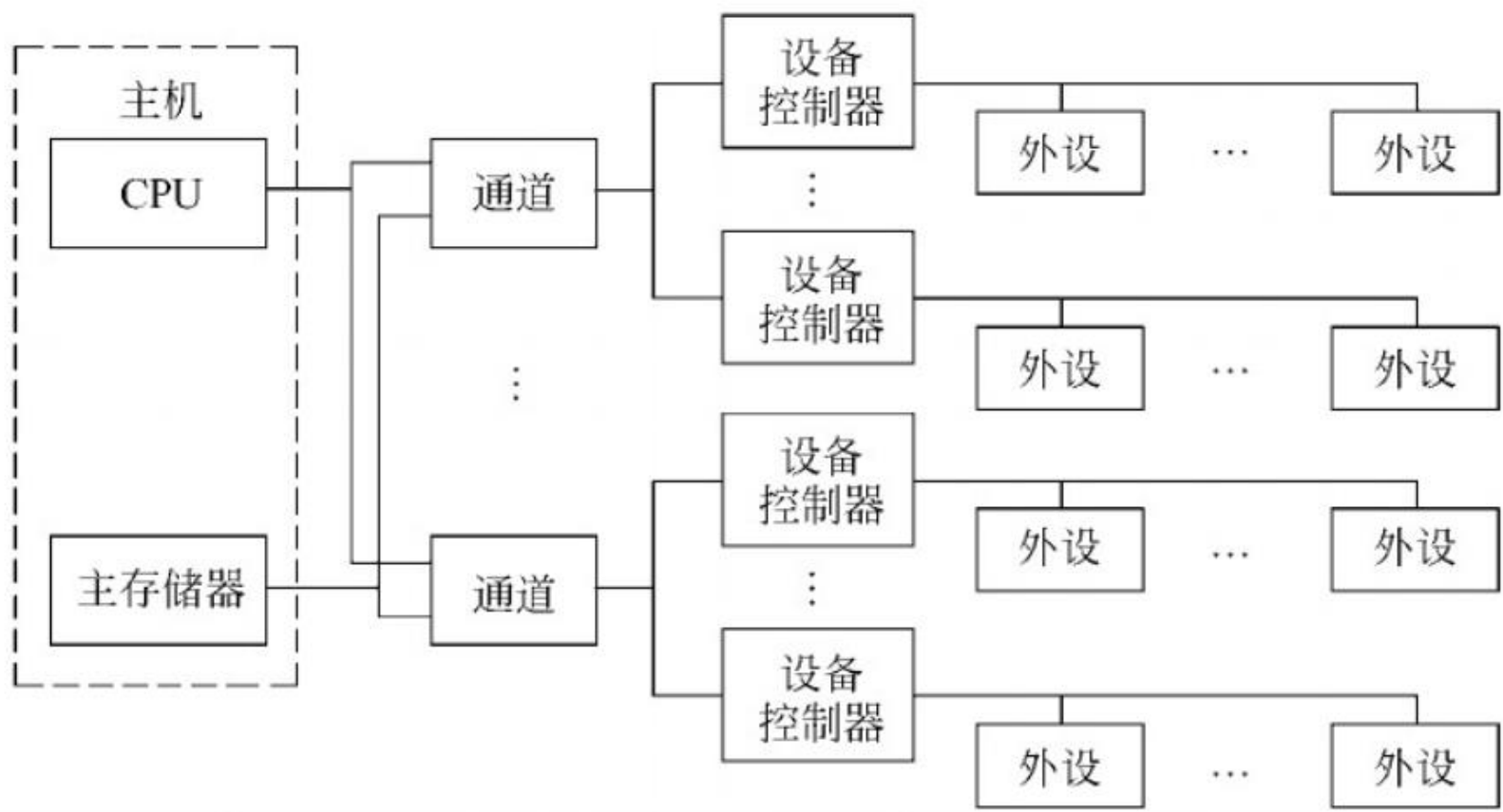
I/O 通道

- I/O通道设备的引入 目的是使一些原来由CPU处理的I/O任务转由通道来承担，从而把CPU从繁杂的I/O任务中解脱出来。
- 采用通道有以下特点：
 - ① DMA（直接存储器存取）方式显著地减少了CPU的干预。
 - ② 只需向I/O通道发送一条I/O指令，即可完成一组相关的读（或写）操作及有关控制。
 - ③ 可实现CPU、通道和I/O设备三者的并行操作，从而更有效地提高整个系统的资源利用率。



5.1 I/O 系统简介

通道是一种特殊的执行I/O指令的处理机





课堂练习

1. 虚拟设备是指被多个用户或进程交替使用的设备，宏观上好象多个用户同时在使用。()
2. 通道技术根本上是从软件上解决操作系统对输入输出操作的控制问题。()
3. 逻辑设备是物理设备属性的表示，用来指定某一具体设备。()
4. 从设备的资源属性分类，可把设备分为独占设备、共享设备和虚拟设备。()
5. 用户在使用I/O设备时，通常采用物理设备名，指明具体的设备。()



第五章 输入输出系统

5.1 I/O 系统简介

5.2 中断处理程序和设备驱动程序

5.3 与设备无关的I/O软件

5.4 磁盘系统及磁盘调度



5.2 中断处理程序和设备驱动程序

1. 中断简介

- 1) 中断和陷入—CPU外部事件和内部事件导致
- 2) 中断向量表 and 中断优先级
- 3) 对多中断源的处理方式
 - 屏蔽（禁止）中断
 - 嵌套中断



1. 中断简介

■ 基本概念

- **中断源**：引起中断发生的**事件**
- **中断请求**：中断源向CPU发出的请求中断处理**信号**
- **中断响应**：CPU收到中断请求后转到相应的事件处理程序的**过程**
- **关中断/开中断**：CPU内部的PSW的中断允许位被清除/被设置，不允许/允许CPU响应中断。用于保证某段程序执行的**原子性**
- **中断屏蔽**：在中断请求产生后，系统有选择地**封锁**一部分中断而允许另一部分仍能得到响应。有些具有最高优先级的中断不允许被屏蔽。



2. 中断处理程序

中断处理层的**主要工作**有：

- 进行进程上下文的切换，
- 对处理中断信号源进行测试，
- 读取设备状态和修改进程状态等



2. 中断处理程序

- 对于为每一类设备设置一个I/O进程的设备处理方式，其中断处理程序的处理过程分成以下几个步骤
- 1. 测定是否有未响应的中断信号
 - 程序完成当前指令后测试是否有未响应的中断信号。
 - 如果没有，继续执行下一条指令。
 - 如果有，则停止原有进程的执行，准备转去执行中断处理程序，为把处理机的控制权转交给中断处理程序做准备。

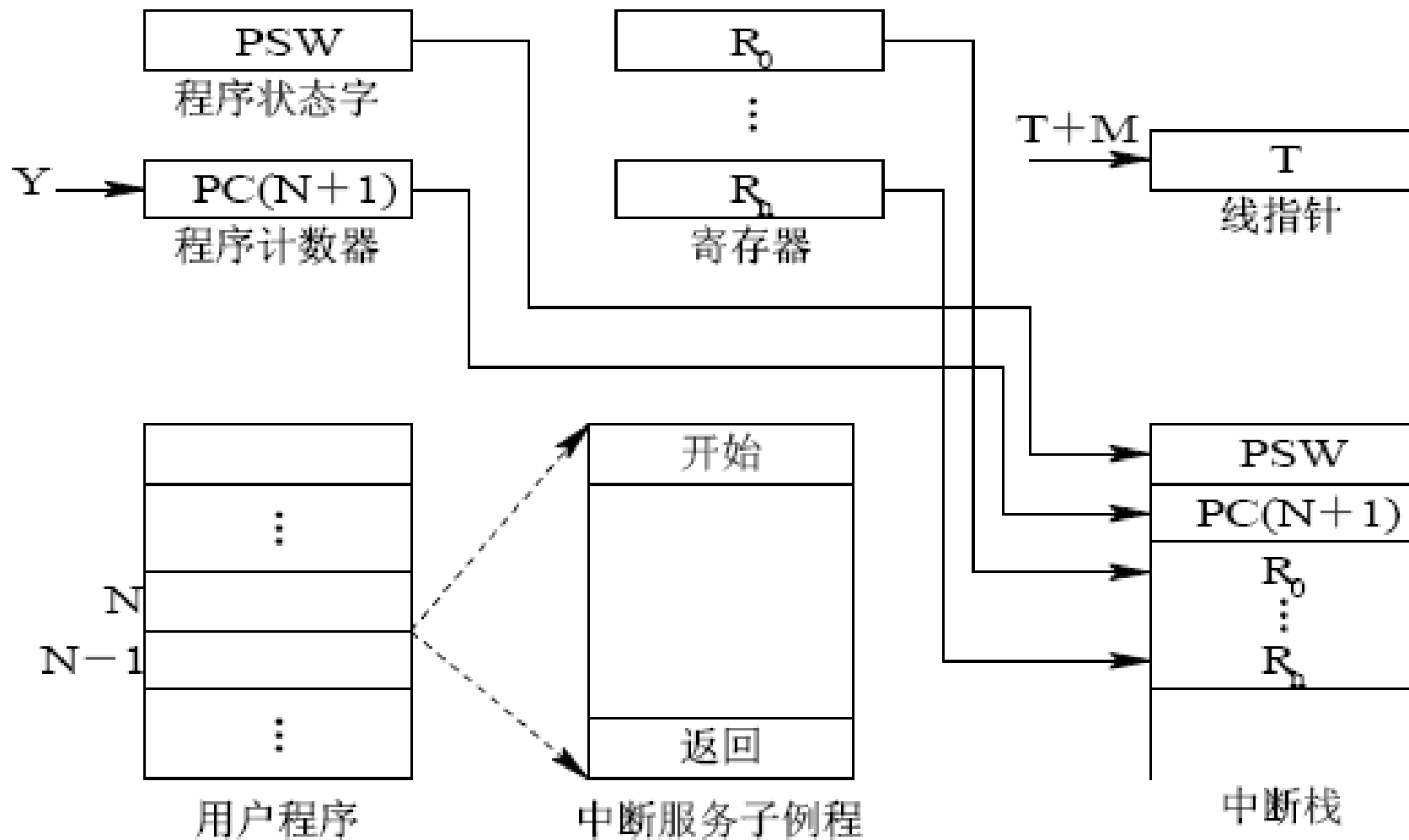


保护被中断进程的CPU 环境

- 通常由**硬件自动**将处理机状态字PSW 和程序计数器(PC)中的内容，保存在中断保留区(栈)中
- 然后把被中断进程的**CPU现场信息**(即包括所有的CPU寄存器，如通用寄存器、段寄存器等内容)都**压入中断栈中**



中断线程保护示意图





转入相应的设备处理程序

- 由处理机对各个中断源进行测试，以确定引起本次中断的 I/O 设备，并发送一应答信号给发出中断请求的进程，使之消除该中断请求信号
- 然后将相应的设备中断处理程序的入口地址装入到程序计数器中，使处理机转向中断处理程序。



中断处理

- 该程序首先从设备控制器中读出设备状态，以判别本次中断是正常完成中断，还是异常结束中断。
- 若是前者，中断程序便进行结束处理；若还有命令，可再向控制器发送新的命令，进行新一轮的数据传送。
- 若是异常结束中断，则根据发生异常的原因做相应的处理。

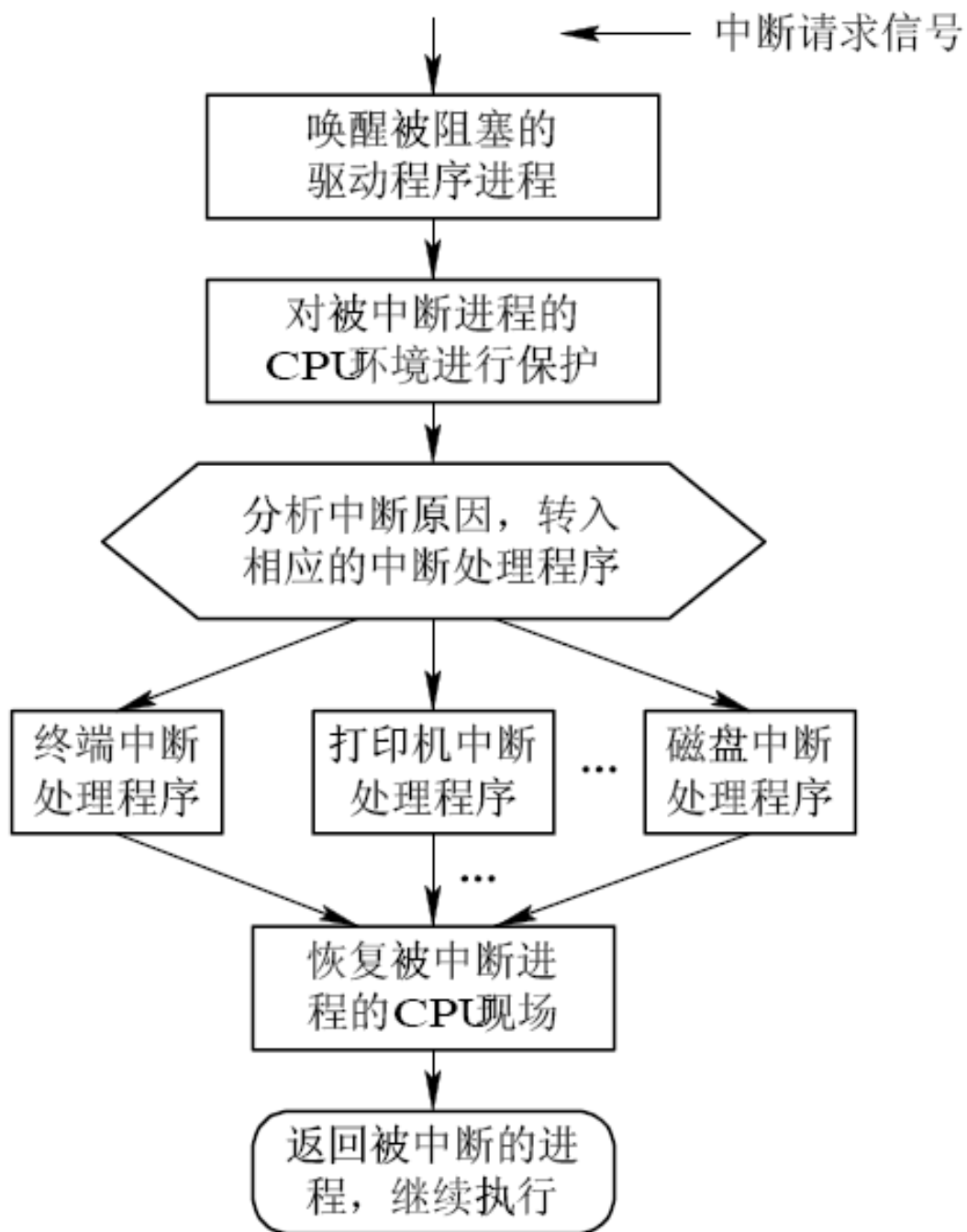


恢复被中断进程的现场

- 当中断处理完成以后，便可将保存在中断栈中的被中断进程的现场信息取出，并装入到相应的寄存器中，其中包括该程序下一次要执行的指令的地址 $N+1$ 、处理机状态字PSW，以及各通用寄存器和段寄存器的内容。
- 这样，当处理机再执行本程序时，便从 $N+1$ 处开始，最终返回到被中断的程序。



中断处理流程





设备驱动程序

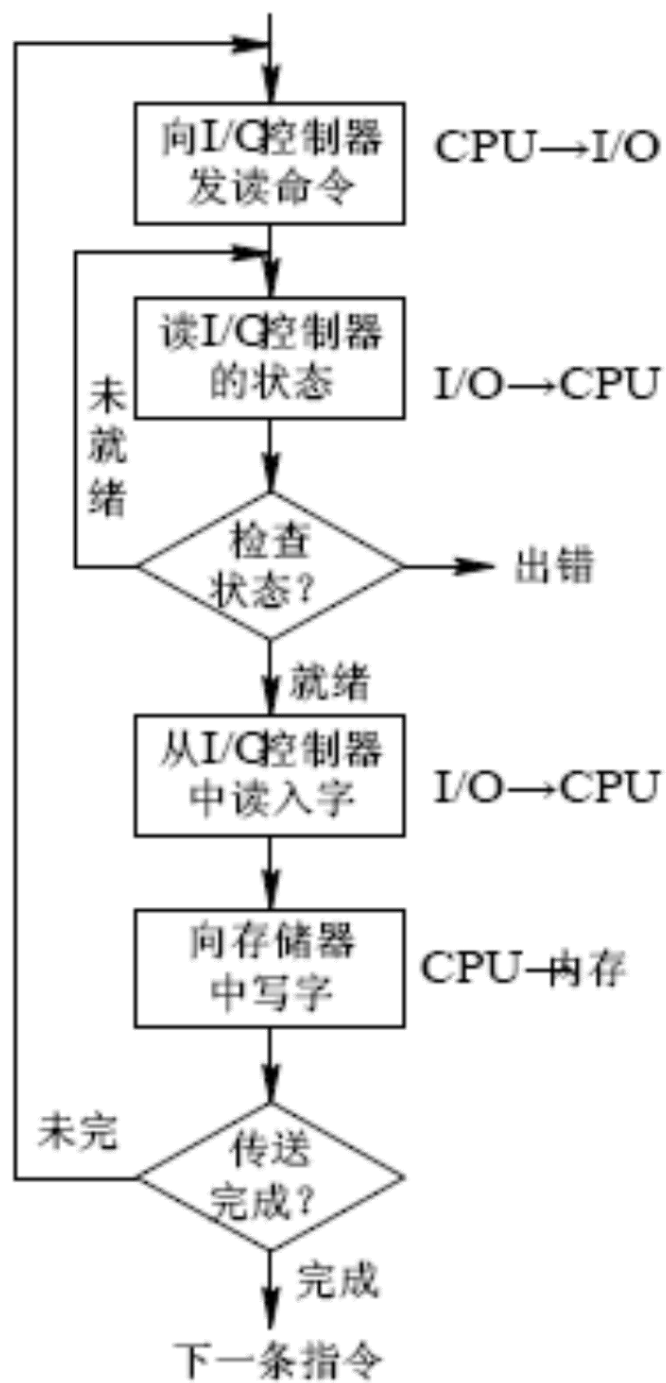
- 设备驱动程序概述
- 设备驱动程序的处理过程
- 对I/O设备的控制方式
 - 使用轮询的可编程I/O方式
 - 使用中断的可编程I/O方式
 - 直接存储器访问方式
 - I/O通道控制方式



对I/O设备的控制方式

- 1. 使用轮询的可编程I/O方式
 - 程序I/O (Programmed I/O) 方式，或称为**忙--等待**方式。处理机向控制器发出一条I/O指令启动输入设备输入数据时，同时把busy置为1，再不断循环测试busy。
 - Busy=0，完成输入，处理机读取数据，送入指定单元，完成一次I/O。
 - 对状态寄存器中的忙/闲标志busy的检查实现控制。

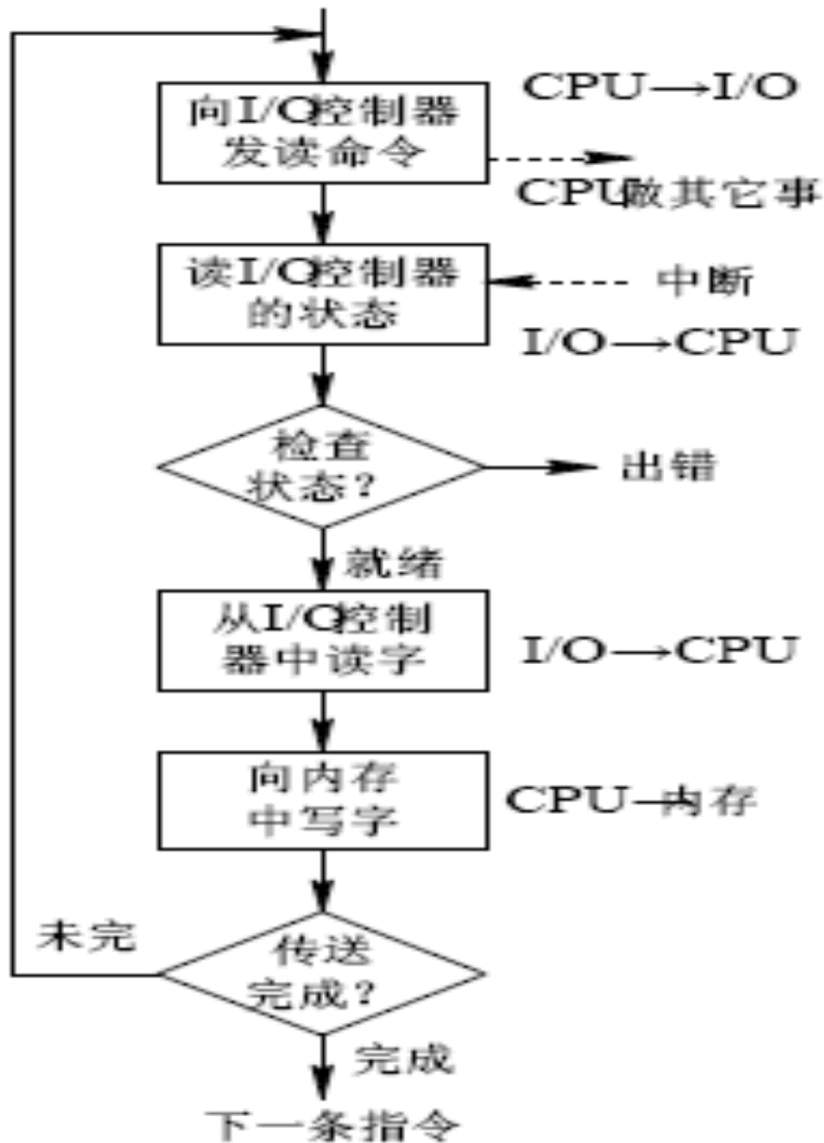
■ 程序/IO方式





使用中断的可编程I/O方式

- 中断驱动方式可以成百倍地提高CPU的利用率。
- CPU与I/O 设备并行工作。



中断驱动方式



直接存储器访问方式

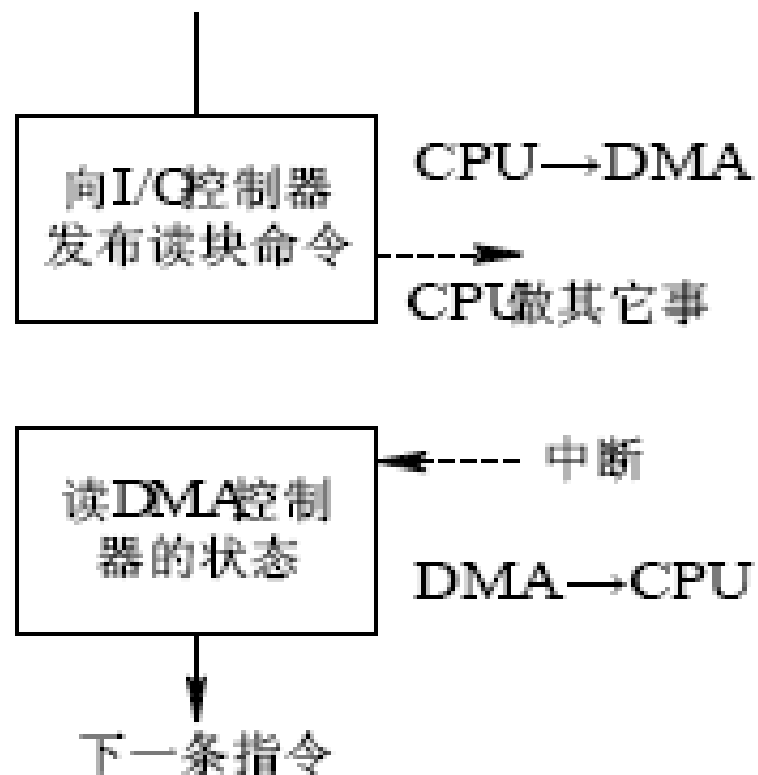
1) DMA控制方式的引入

- 为了进一步减少CPU对I/O的干预而引入了直接存储器访问方式
- DMA控制方式的特点：
 - ①数据传输的基本单位是数据块，即在CPU与I/O设备之间，每次传送至少一个数据块；
 - ②所传送的数据是从设备直接送入内存的，或者相反；
 - ③仅在传送一个或多个数据块的开始和结束时，才需CPU干预，整块数据的传送是在控制器的控制下完成的。



DMA控制方式的引入

- DMA方式较之中断驱动方式，又是成百倍地减少了CPU对I/O的干预，进一步提高了CPU与I/O设备的并行操作程度。



(c) DMA方式



DMA控制器的组成

① 主机与DMA控制器的接口。

② DMA控制器与块设备的

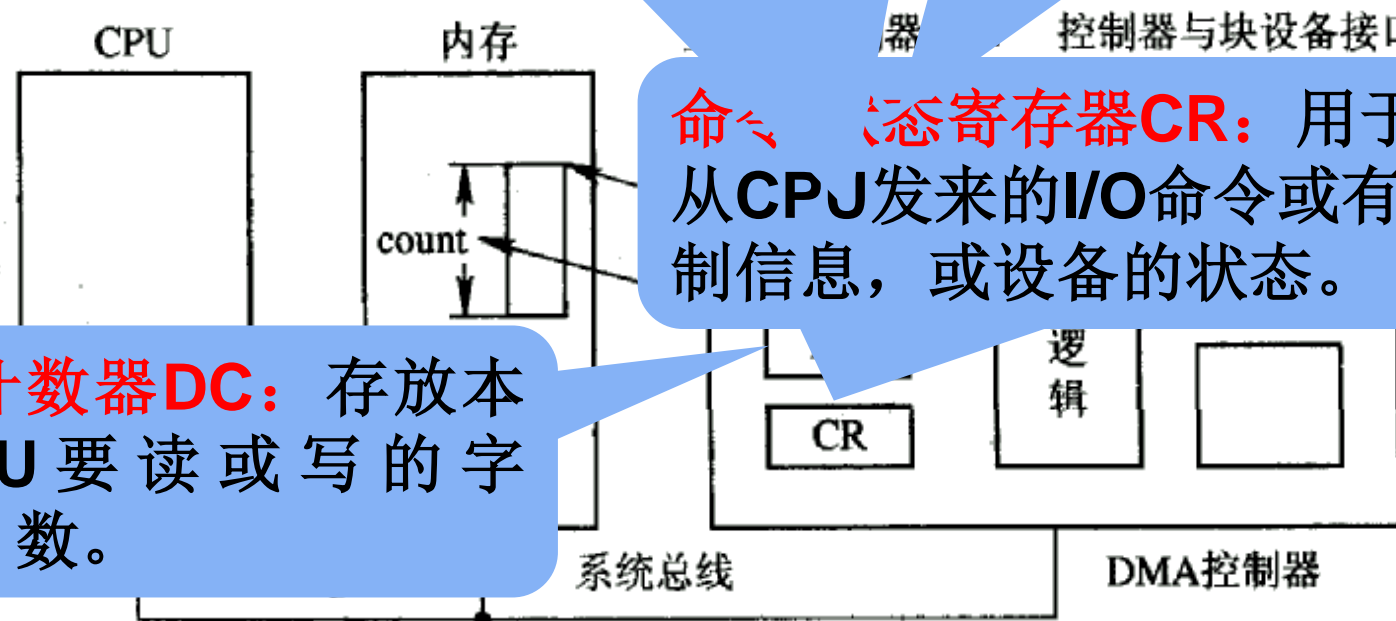
③ I/O控制逻辑。

内存地址寄存器AR：存放把数据从目标地址；有到设备的内存

数据寄存器DR：用于暂存从设备到内存，或从内存到设备的数据。

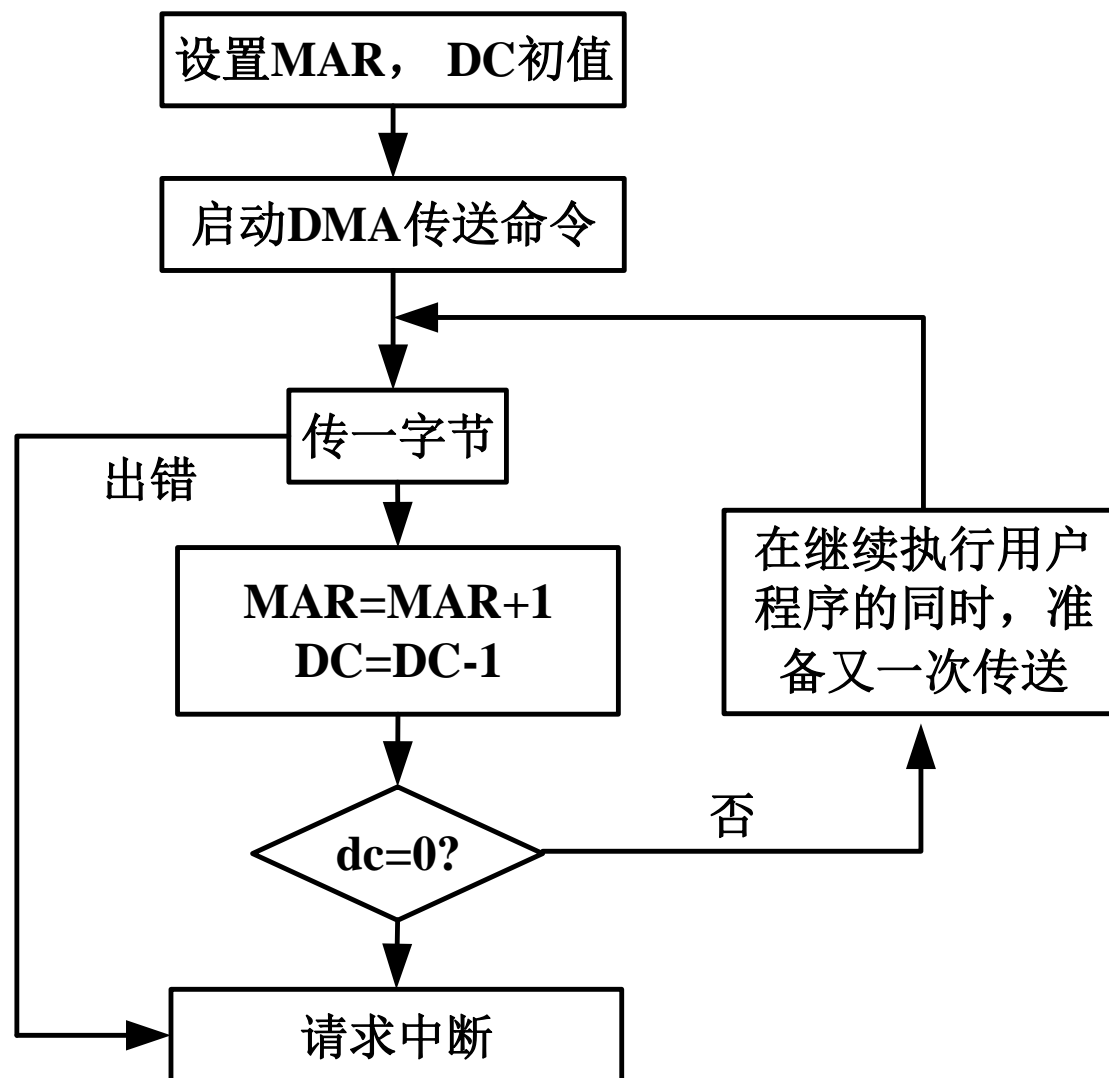
命令寄存器CR：用于接收从CPU发来的I/O命令或有关控制信息，或设备的状态。

数据计数器DC：存放本次CPU要读或写的字（节）数。





DMA工作过程



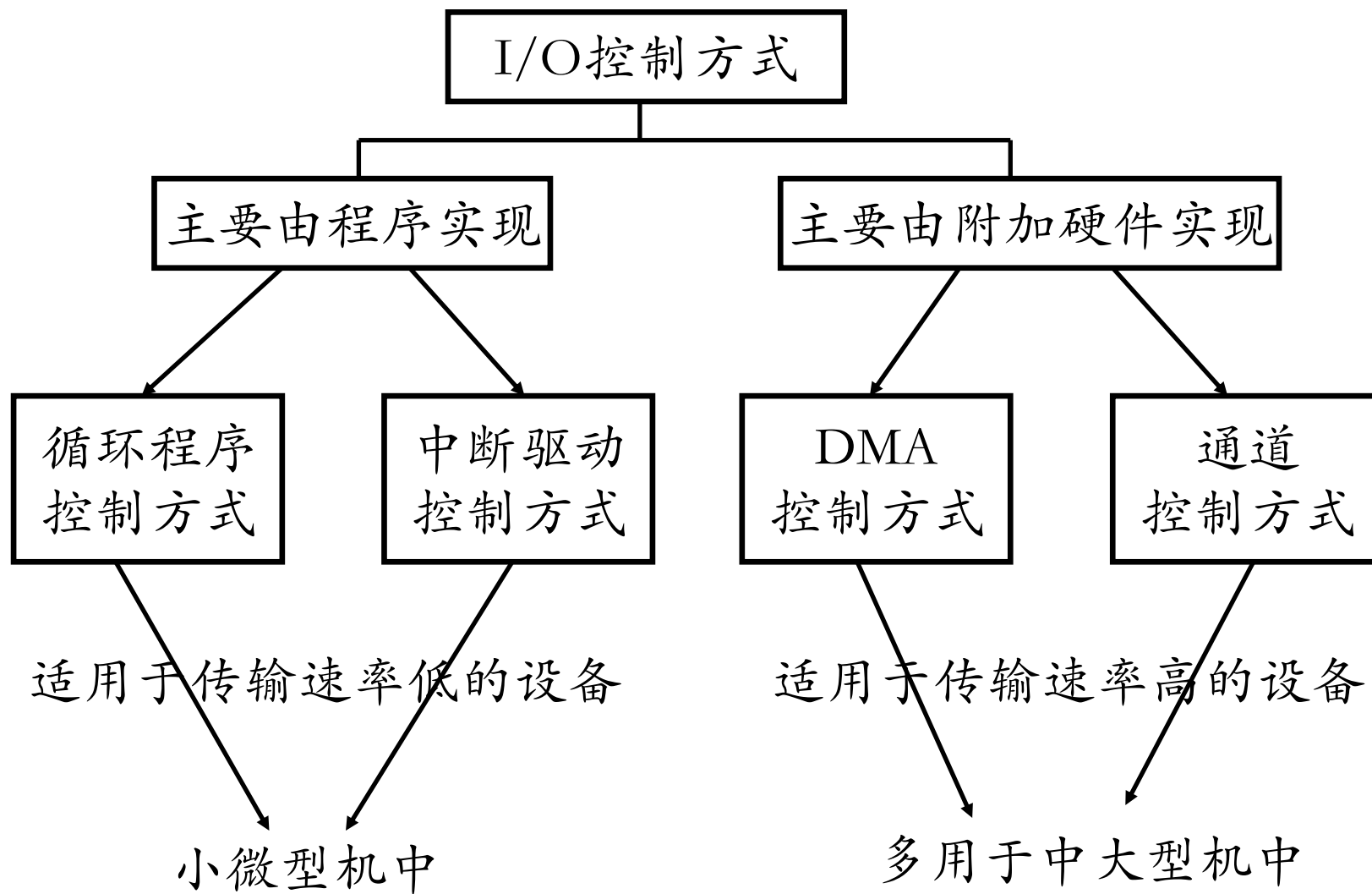


1) I/O通道控制方式的引入

- I/O通道方式是DMA方式的发展，它可进一步减少CPU的干预，即把对**一个**数据块的读（或写）为单位的干预，减少为对**一组**数据块的读（或写）及有关的管理为单位的干预。
- 可实现CPU、通道和I/O设备三者的并行操作，从而更有效地提高整个系统的资源利用率。



总结：I/O控制方式





课堂练习

1. 什么是中断？系统内部中断与外部中断有什么区别？
2. 缺页中断与一般中断的主要区别是什么？
3. 什么是设备的独立性？
4. I/O控制方式中的中断控制方式和DMA方式有什么不同？



第五章 输入输出系统

5.1 I/O 系统简介

5.2 中断处理程序和设备驱动程序

5.3 与设备无关的I/O软件

5.4 磁盘系统及磁盘调度



5.3 与设备无关的I/O软件

设备独立性

- 应用程序独立于具体使用的物理设备。
- 为了实现设备独立性而引入了**逻辑设备**和**物理设备**这两个概念。在应用程序中，使用逻辑设备名称来请求使用某类设备；而系统在实际执行时，还必须使用物理设备名称。
- 因此，系统须具有将逻辑设备名称转换为某物理设备名称的功能



5.3 与设备无关的I/O软件

- 实现设备独立性后，可带来**两方面**的好处。

1) 设备分配时的灵活性

- 进程能以逻辑设备名称来请求某类设备时，系统可立即将该类设备中的任一台分配给进程，仅当所有此类设备已全部分配完毕时，进程才会阻塞。

2) 易于实现I/O 重定向

- 所谓I/O 重定向，是指用于I/O 操作的设备可以更换(即重定向)，而不必改变应用程序。



5.3 与设备无关的I/O软件

设备独立性软件

- 驱动程序是一个与硬件(或设备)紧密相关的软件
- 为了实现设备独立性，必须再在驱动程序之上设置一层软件，称为设备独立性软件



5.3 与设备无关的I/O软件

- 设备独立性软件的主要功能可分为以下**两个**方面：

(1) 执行所有设备的公有操作。

- 这些公有操作包括：

- ① 对独立设备的分配与回收；
- ② 将逻辑设备名映射为物理设备名，进一步可以找到相应物理设备的驱动程序；
- ③ 对设备进行保护，禁止用户直接访问设备；
- ④ 缓冲管理
- ⑤ 差错控制
- ⑥ 提供独立于设备的逻辑块



5.3 与设备无关的I/O软件

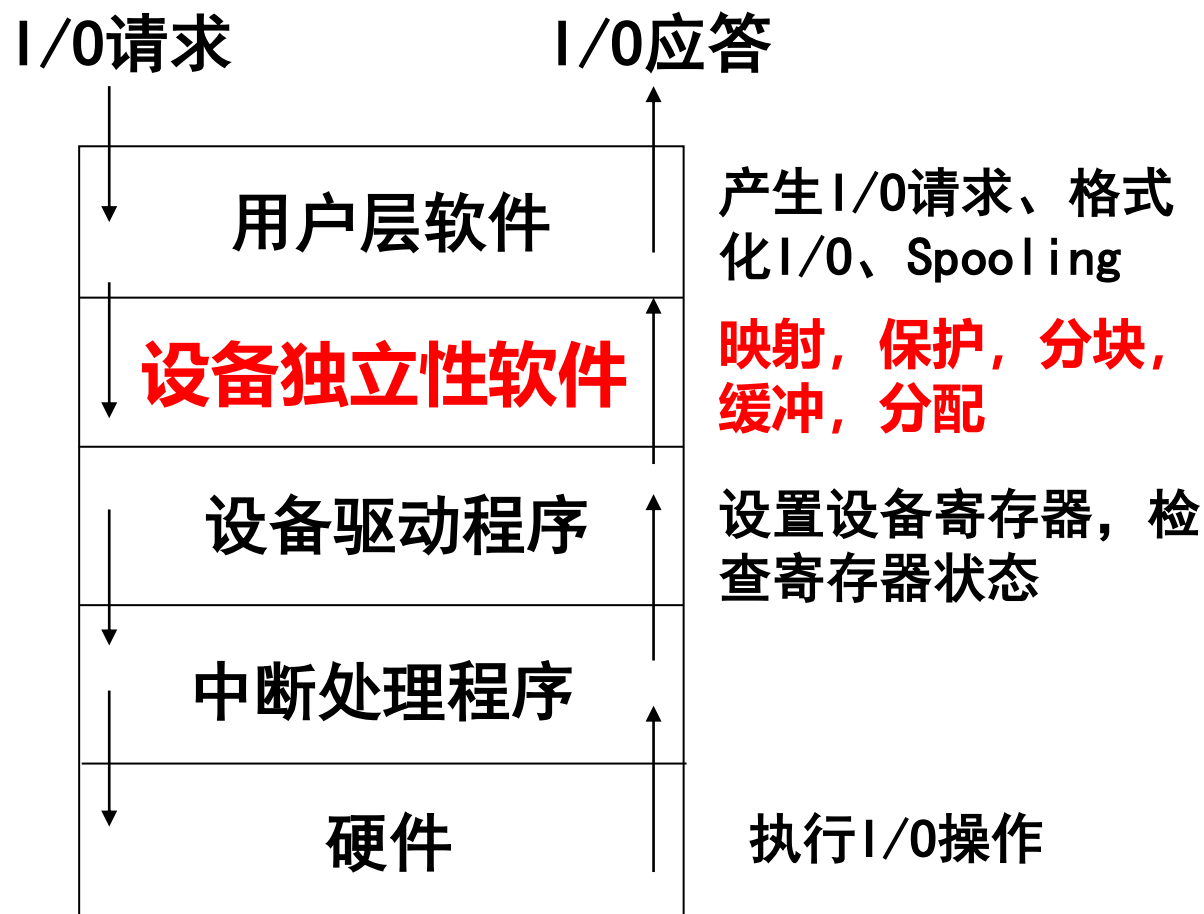
(2) 向用户层(或文件层)软件提供统一接口

- 无论何种设备，它们向用户所提供的接口应该是相同的。
- 例如，对各种设备的读操作，在应用程序中都使用read；而对各种设备的写操作，也都使用write。



5.3 与设备无关的I/O软件

- ◆ 大部分I/O软件都在操作系统内部，仍有一小部分在用户层
- ◆ 与用户程序链接在一起的库函数
- ◆ 完全运行在内核之外的一些程序（如Spooling系统）





第五章 输入输出系统

5.1 I/O 系统简介

5.2 中断处理程序和设备驱动程序

5.3 与设备无关的I/O软件

5.4 磁盘系统及磁盘调度



5.4 磁盘系统及磁盘调度

- 现代计算机系统中，都配置了磁盘存储器，并以它为主来存放文件。
- 对文件的操作，都将涉及到对磁盘的访问，磁盘I/O速度的高低和磁盘系统的可靠性，都将直接影响到系统性能。因此，设法改善磁盘系统的性能，已成为现代操作系统的重要任务之一。



5.4 磁盘系统及磁盘调度

提高磁盘I/O速度的主要途径：

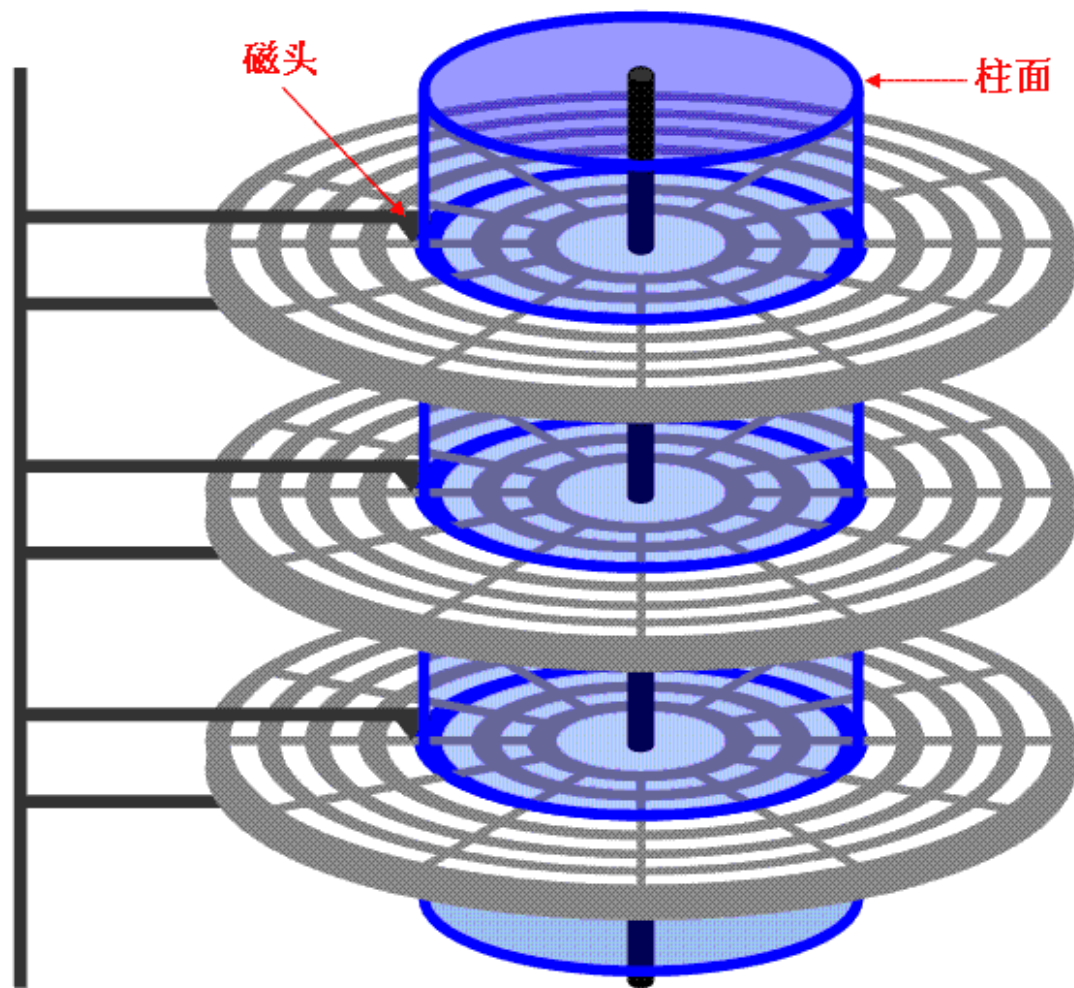
- (1) 选择性能好的磁盘**
- (2) 采用好的磁盘调度算法**
- (3) 设置磁盘高速缓存 (Disk Cache)**
- (4) 其它方法**
- (5) 采用高度可靠、快速的容量磁盘系统——磁盘冗余阵列**



5.4 磁盘系统及磁盘调度

数据的组织和格式

- 存储面 (surface)
- 磁道 (track)
- 柱面
- 扇区 (sectors)





5.4 磁盘系统及磁盘调度

- 磁盘访问时间分成以下三部分：

1) 寻道时间 T_s ：

这是指把磁臂（磁头）移动到指定磁道上所经历的时间。

$$T_s = m \times n + s$$

s ：启动磁臂的时间

n ：磁头移动 n 条磁道

m ：移动每一条磁道所花费的时间



5.4 磁盘系统及磁盘调度

3. 磁盘访问时间分成以下三部分：

2) 旋转延迟时间 T_τ ：

- 这是指定扇区移动到磁头下面所经历的时间。
- 例如：
 - 软盘旋转速度为 300 r/min或600 r/min，这样，平均 T_τ 为50 ~ 100 ms。
 - 硬盘旋转速度为15 000 r/min，每转需时4 ms，平均旋转延迟时间 T_τ 为2 ms；



5.4 磁盘系统及磁盘调度

- 3. 磁盘访问时间分成以下三部分:

3) 传输时间 T_t

- 这是指把数据从磁盘读出或向磁盘写入数据所经历的时间。
- T_t 的大小与每次所读/写的字节数 b 和旋转速度有关:

$$T_t = \frac{b}{rN}$$

- r 为磁盘每秒钟的转数; N 为一条磁道上的字节数



5.4 磁盘系统及磁盘调度

磁盘调度

磁盘是可供多个进程共享的设备，当有多个进程都要求访问磁盘时，应采用一种最佳调度算法，以使各进程对磁盘的平均访问时间最小。目前常用的磁盘调度算法有：

1. 先来先服务FCFS
2. 最短寻道时间优先SSTF
3. 扫描 (SCAN) 算法
4. 循环扫描 (CSCAN) 算法



1. 先来先服务FCFS

请求序列：55, 58, 39, 18, 90, 160, 150, 38, 184

- ★根据进程请求访问磁盘的先后次序进行调度。此算法的优点是公平、简单，且每个进程的请求都能依次地得到处理，不会出现某一进程的请求长期得不到满足的情况。

(从 100 号磁道开始)	
被访问的下一个磁道号	移动距离 (磁道数)
平均寻道长度:	



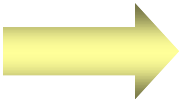
2. 最短寻道时间优先SSTF

- 该算法选择这样的进程，其要求访问的磁道，与当前磁头所在的磁道距离最近，以使每次的寻道时间最短，
- SSTF算法虽然能获得较好的寻道性能，但却可能导致某个进程发生“饥饿”现象。



2. 最短寻道时间优先SSTF

55
58
39
18
90
160
150
38
184



(从 100 号磁道开始)	
被访问的下一个磁道号	移动距离 (磁道数)
平均寻道长度:	



3. 扫描 (SCAN) 算法

- SSTF算法虽然能获得较好的寻道性能，但却可能导致某个进程发生“饥饿”(Starvation)现象。
- 该算法优先考虑的是磁头当前的移动方向。例如，磁头自里向外移动，并同时自里向外地访问，直至再无更外的磁道需要访问时，才将磁臂换向自外向里移动。（又常称之为电梯调度算法）



3. 扫描 (SCAN) 算法

55
58
39
18
90
160
150
38
184



(从 100#磁道开始, 向磁道号增加方向访问)	
被访问的下一个磁道号	移动距离 (磁道数)
平均寻道长度:	



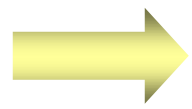
4. 循环扫描 (CSCAN) 算法

- CSCAN算法规定磁头单向移动，例如，只是自里向外移动，当磁头移到最外的磁道并访问后，磁头立即返回到最里的欲访问磁道，亦即将最小磁道号紧接着最大磁道号构成循环，进行循环扫描。



4. 循环扫描 (CSCAN) 算法

55
58
39
18
90
160
150
38
184



(从 100#磁道开始, 向磁道号增加方向访问)	
被访问的下一个磁道号	移动距离 (磁道数)
平均寻道长度:	



本章重点

1. I/O部分：I/O控制器、通道、**I/O控制方式（查询I/O方式、中断I/O控制方式、DMA方式）**、设备驱动程序。
2. 数据结构：设备控制表、控制器表、通道表、系统设备表。
3. 设备分配及分配算法：硬盘硬件结构、**硬盘分配算法（FIFO、SSTF、SCAN、CSCAN）**。
4. 缓冲技术：设备分配的独立性和分配时的安全性。



课后作业

■教材第236页

练习题4、10、13、16、18、21、24、

31