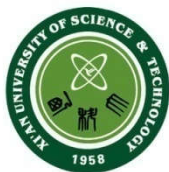


西安科技大学
XI' AN UNIVERSITY OF SCIENCE TECHNOLOGY

第七章 I/O管理

刘晓建

2017年12月11日



西安科技大学

XI' AN UNIVERSITY OF SCIENCE TECHNOLOGY

I/O系统所含设备种类繁多，差异又非常大，致使I/O系统成为操作系统中最繁杂且与硬件最紧密相关的部分。对I/O设备进行高度抽象，建立一个I/O设备的虚拟界面使编程人员能够容易地检索和存储数据，这是I/O系统设计所面临的主要问题。

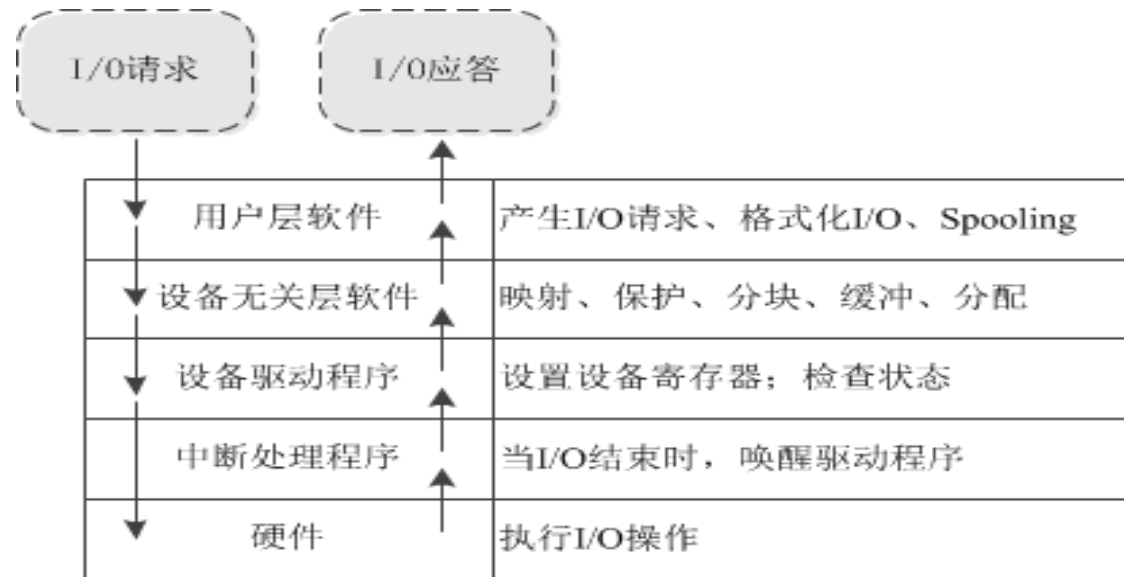
本章将介绍这样几个内容：

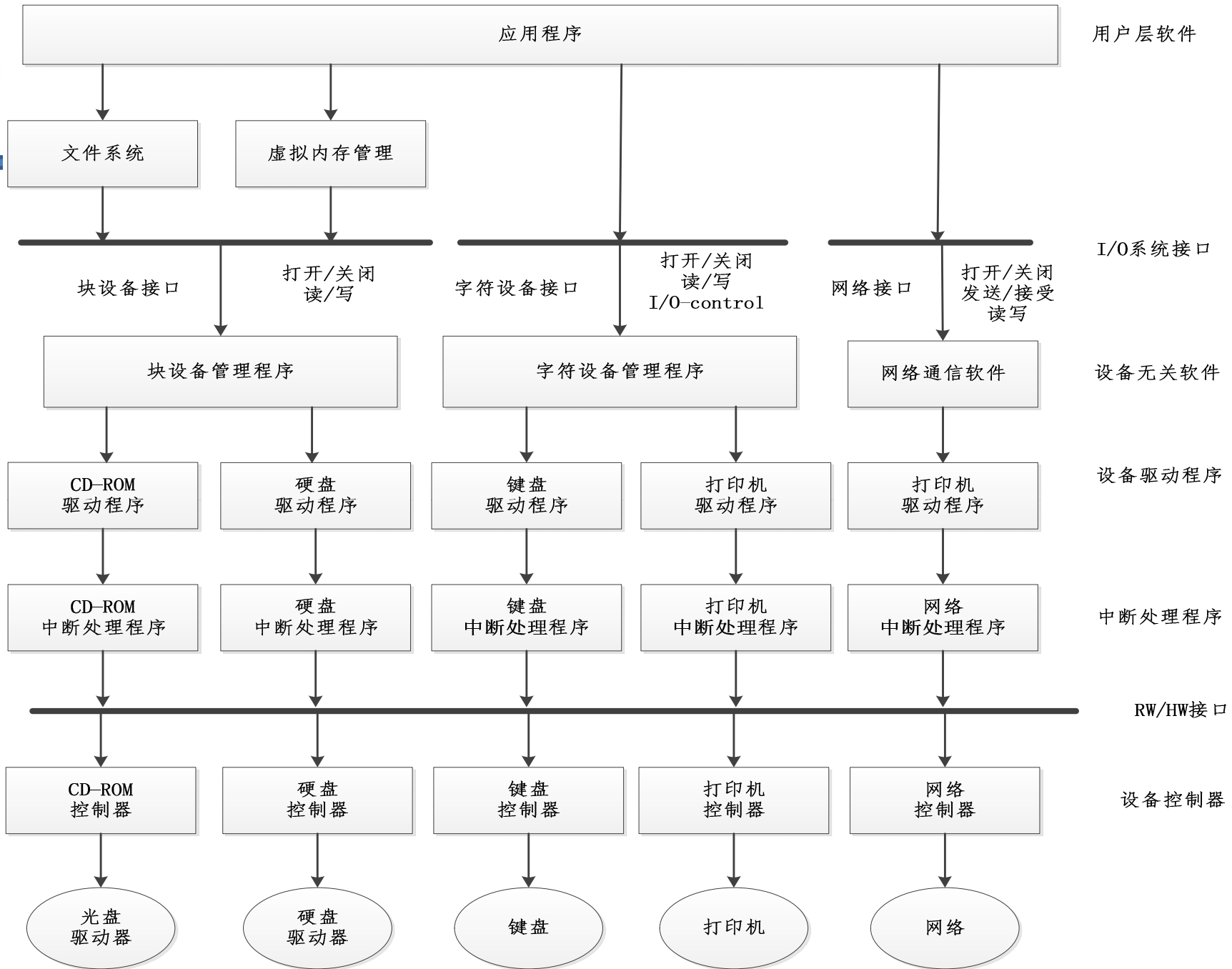
- I/O系统硬件组织
- I/O系统接口
- I/O设备控制器和通道
- I/O设备的控制方式
- 缓冲处理技术
- 磁盘驱动调度

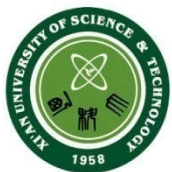


7.1.2 I/O系统层次结构和模型

I/O系统涉及的内容很广泛，向下与硬件密切相关，向上又与文件系统、虚拟内存系统和用户直接交互。为了使复杂的I/O系统具有清晰的架构以及良好的可移植性和易用性，目前普遍采用层次结构的I/O系统。







§ 7.2 I/O系统硬件结构和组织

表7-1 按使用特性对设备分类

外部设备			
存储设备	输入/输出设备	终端设备	脱机设备
磁带 硬盘 软盘 光盘 U盘 磁鼓	键盘 打印机 显示器 图形输入/输出设备 图像输入/输出设备 绘图仪 音频输入/输出设备 网络通信设备 其他	通用终端 专用终端 智能终端 虚拟终端	纸带穿孔机

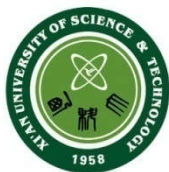


7.2.3 I/O设备控制器

- 设备控制器通常和硬件设备连为一体，主要功能是控制一个或多个I/O设备，以实现I/O设备和计算机之间的数据交换。
- I/O控制器是处理机与I/O设备之间的接口，接收从处理机发来的命令，控制I/O设备工作，使处理机从繁杂的设备控制事务中解脱出来。
- 设备控制器是一个**可编址**的设备，当它仅控制一个设备时，它只有一个唯一的设备地址；若控制器可连接多个设备，则应含有多个设备地址，每一个设备地址对应一个设备。可以把
- 设备控制器分成两类：字符设备控制器和块设备控制器。

设备控制器的基本功能

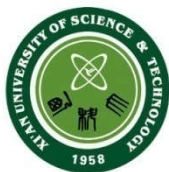
- 接收和识别命令
- 数据交换
- 标识和报告设备的状态
- 地址识别——识别设备地址和控制器中的寄存器地址
- 数据缓冲区和差错控制



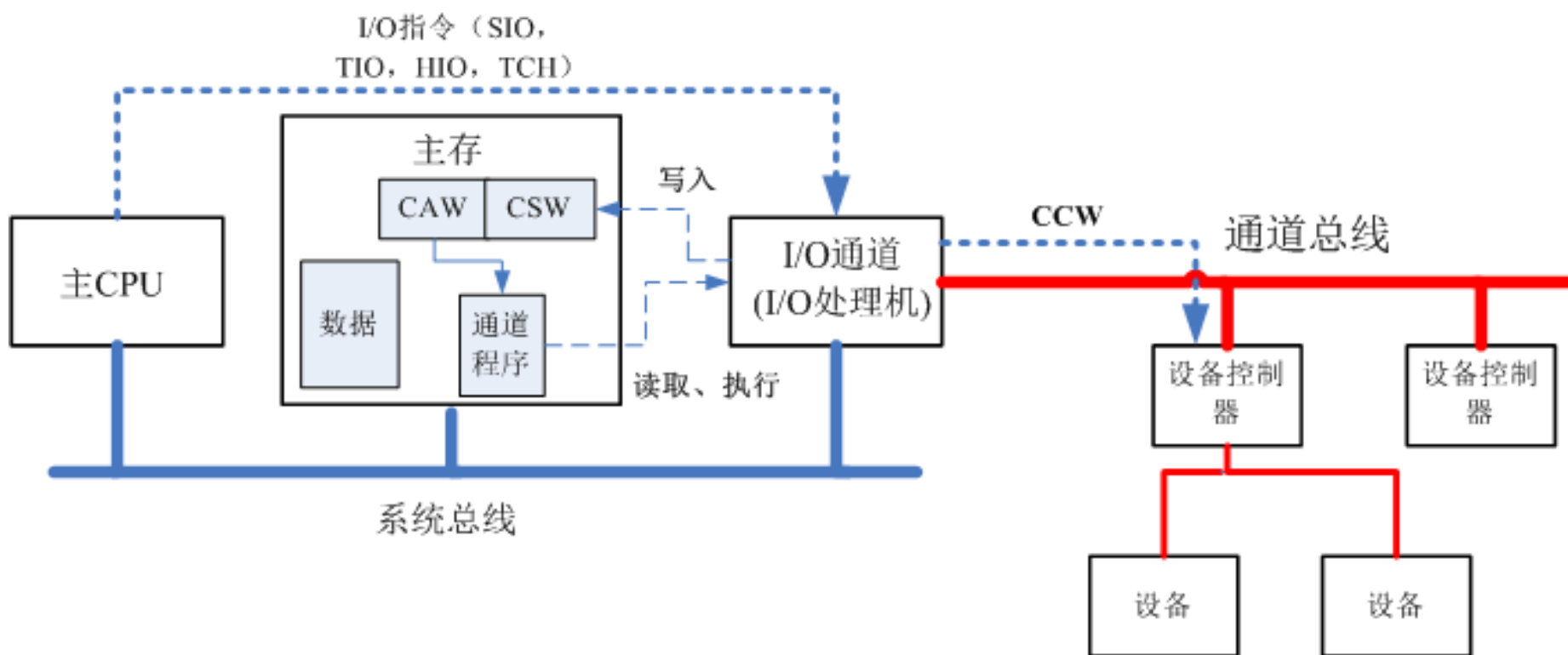
7.2.4 I/O 通道

虽然在处理器和I/O设备之间增加了设备控制器后，已大大减少了处理器对I/O的直接干预，但当主机所配置的外设很多时，处理器的负担仍然很重。为此在处理器与设备控制器之间又增设了**专门负责I/O的处理机**，即**I/O通道** (I/O Channel) 。

把一些原来由处理器处理的I/O任务转而由通道来承担，从而把处理器从繁杂的I/O任务中解脱出来。设置了通道之后，处理器只需向通道发送一条**I/O命令**，通道收到后，便从内存中取出本次要执行的**通道程序**，然后执行它，仅当通道完成了指定的I/O任务后，才向处理器发中断信号。



CPU->通道: I/O指令
通道->设备: CCW



通道结构和通道相关概念



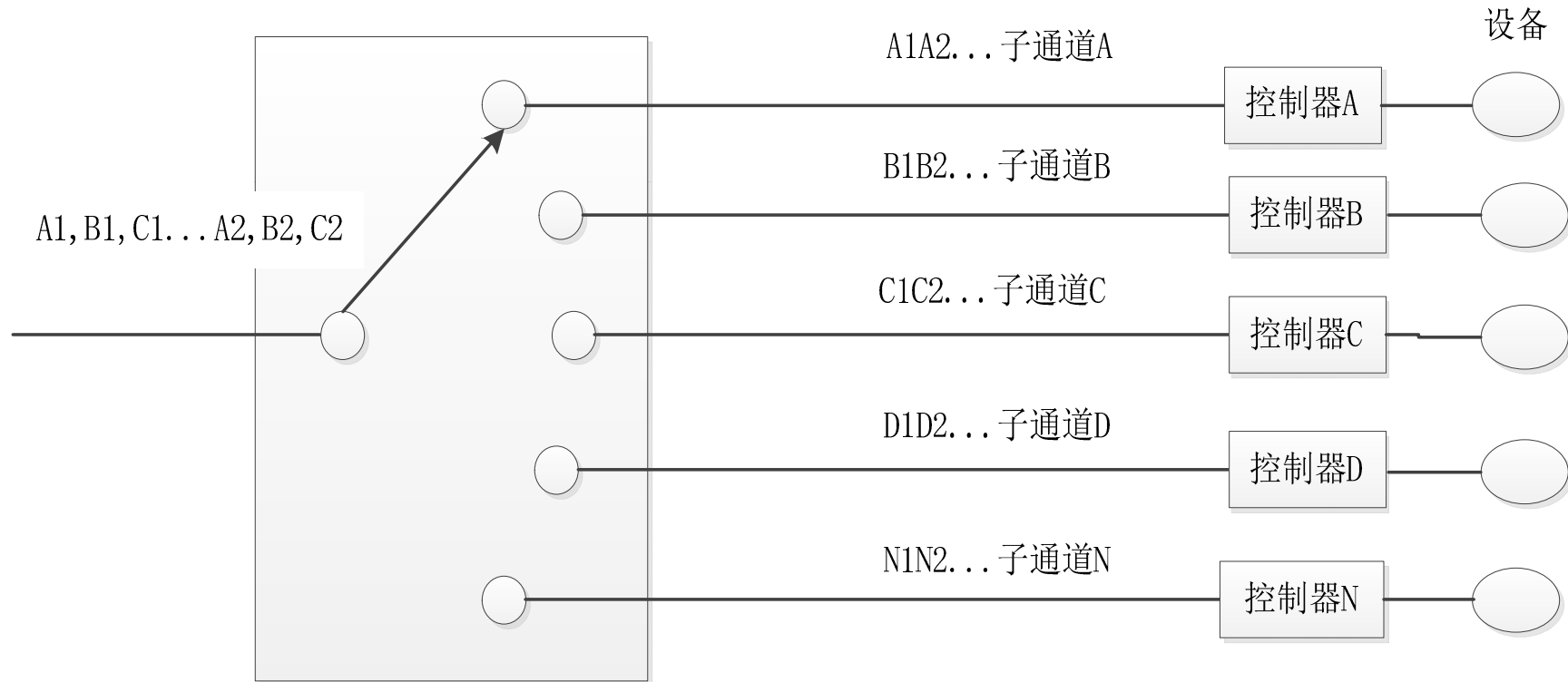
I/O通道实际上是一种特殊的处理机，它具有执行I/O指令的能力，并通过编制和执行通道程序（I/O程序）来控制I/O操作。I/O通道与一般处理机不同，主要表现在如下两个方面：

- 一是其指令类型单一，这是由于通道硬件比较简单，其所能执行的命令主要局限于与I/O操作有关的指令；
- 二是通道没有自己的内存，通道所执行的通道程序存放在主机的内存中（主机把这些通道程序作为数据块对待），即通道处理机与主处理器共享内存。



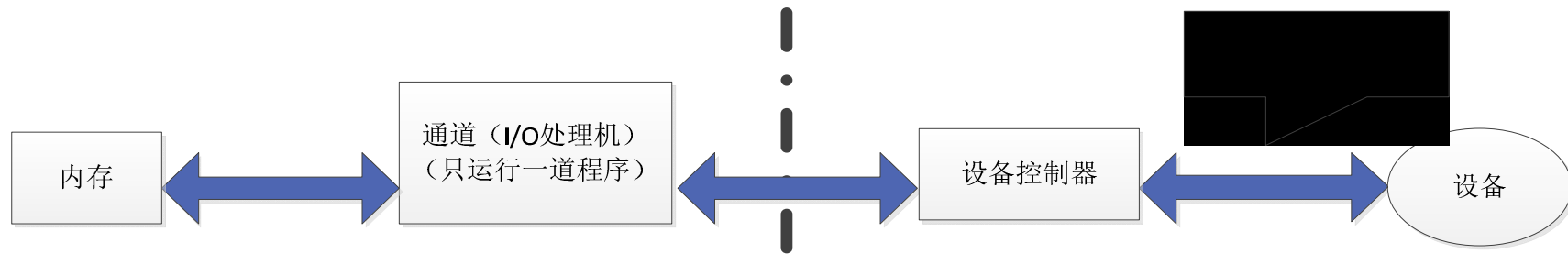
通道类型

字节多路通道 (Byte Multiplexor Channel)

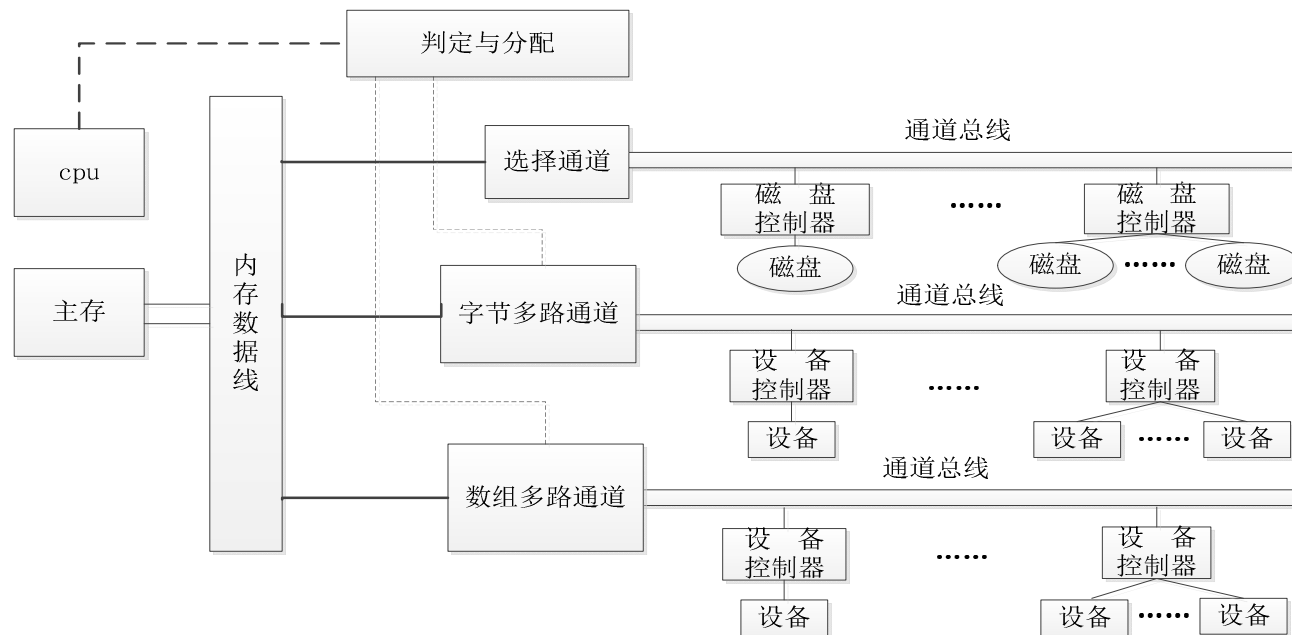




数组选择通道 (Block selector channel)



数组多路通道 (Block multiplexor channel)



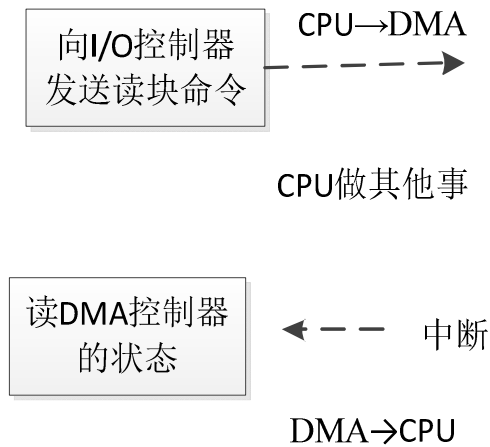


7.2.5 I/O 设备的控制方式

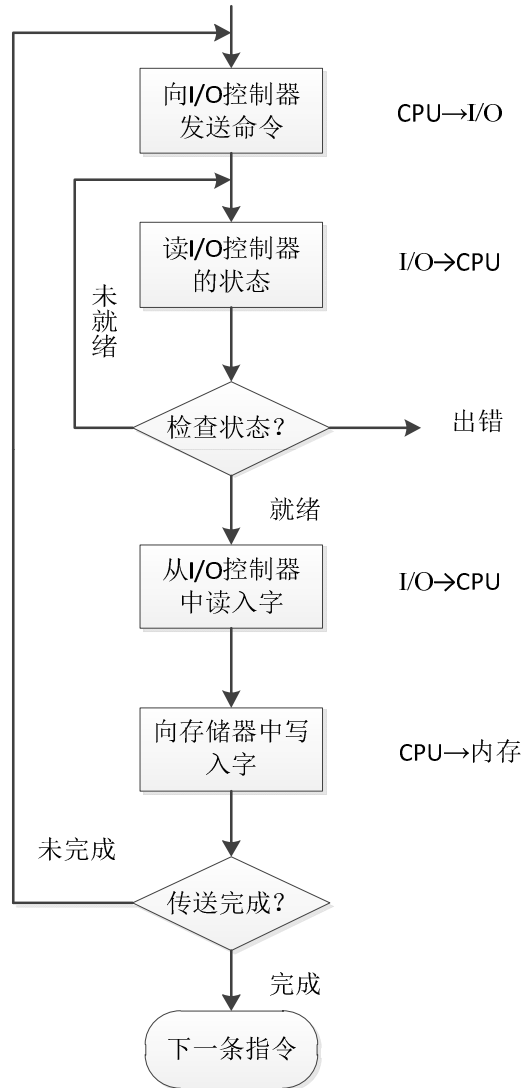
可编程I/O

中断驱动I/O

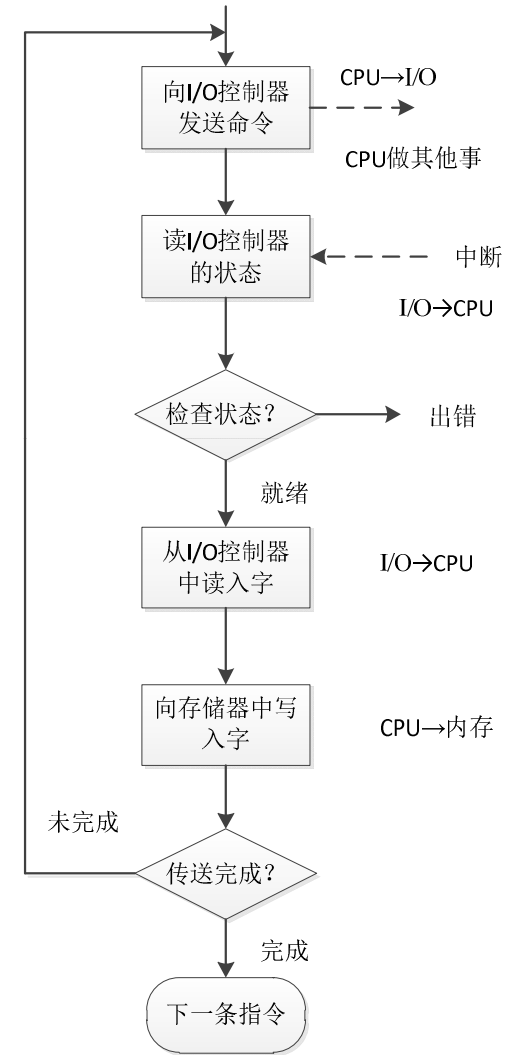
DMA



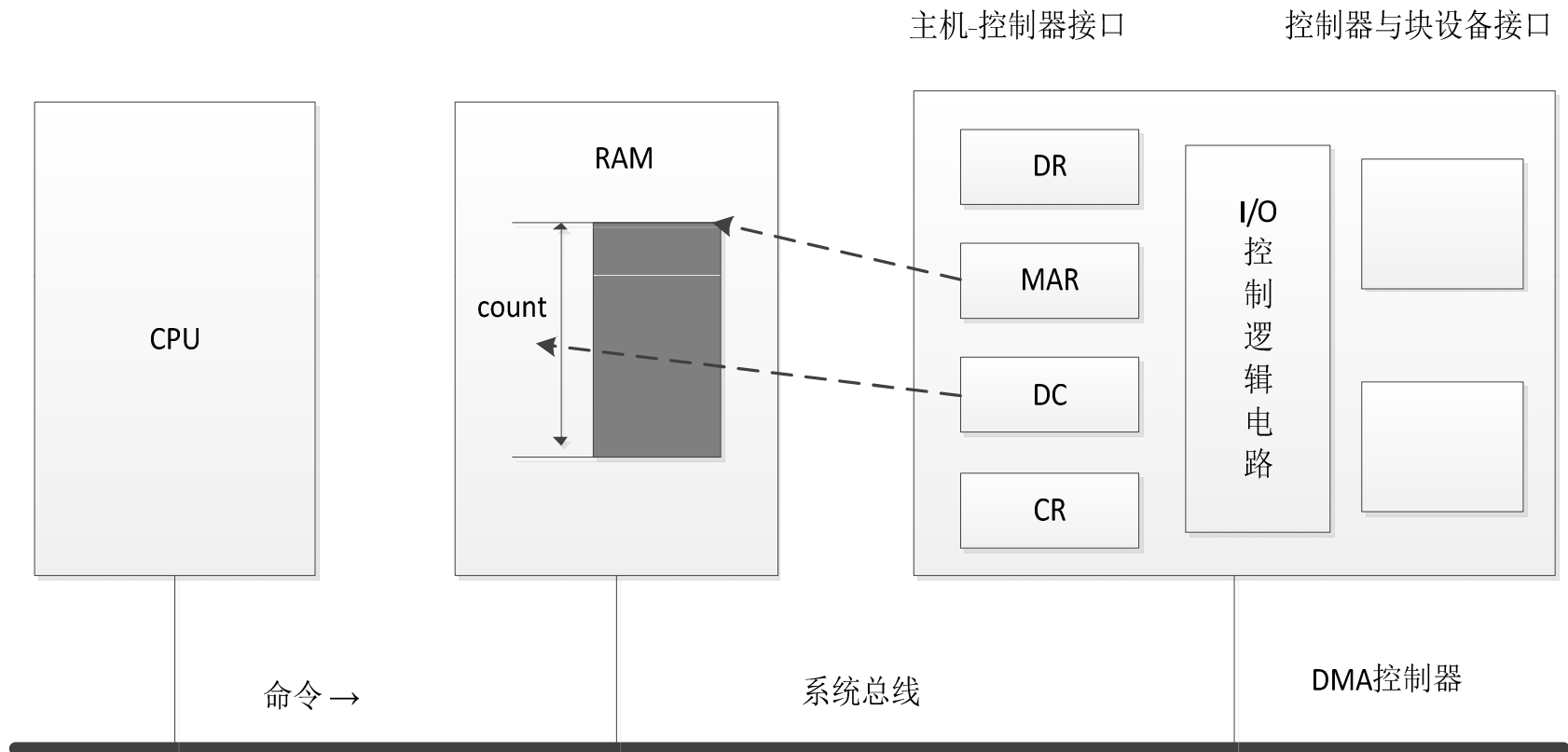
(c)

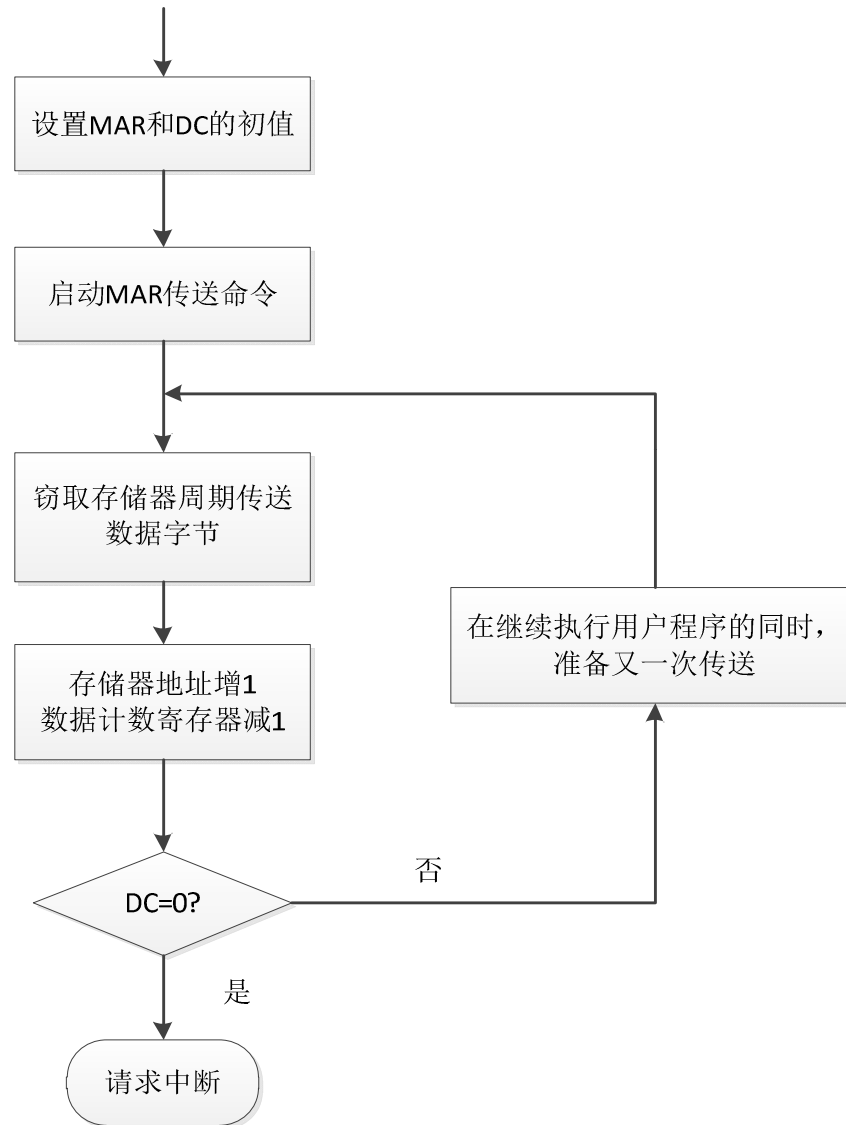


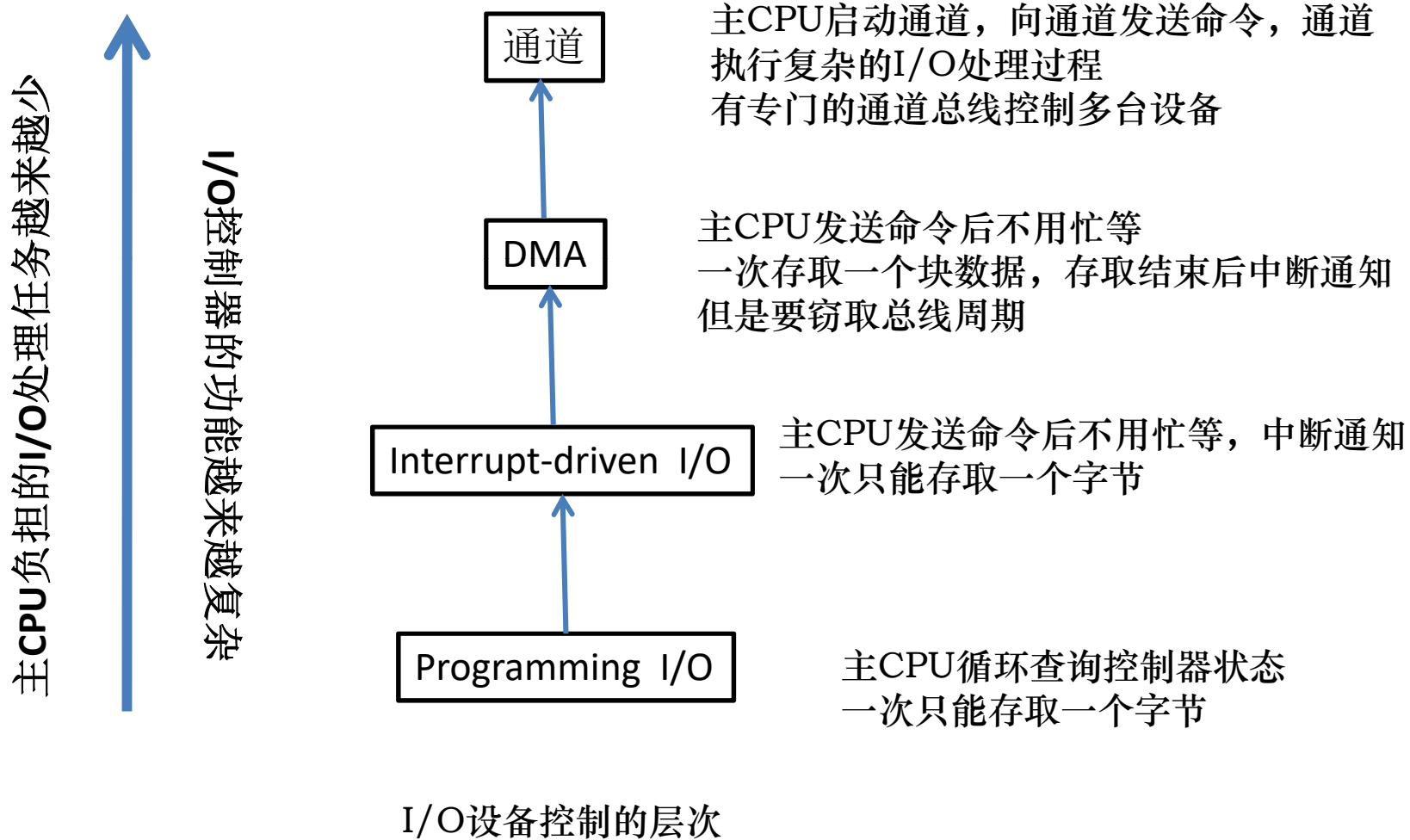
(a)

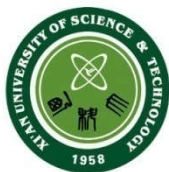


(b)









§ 7.3 I/O系统软件组织

设计目标:

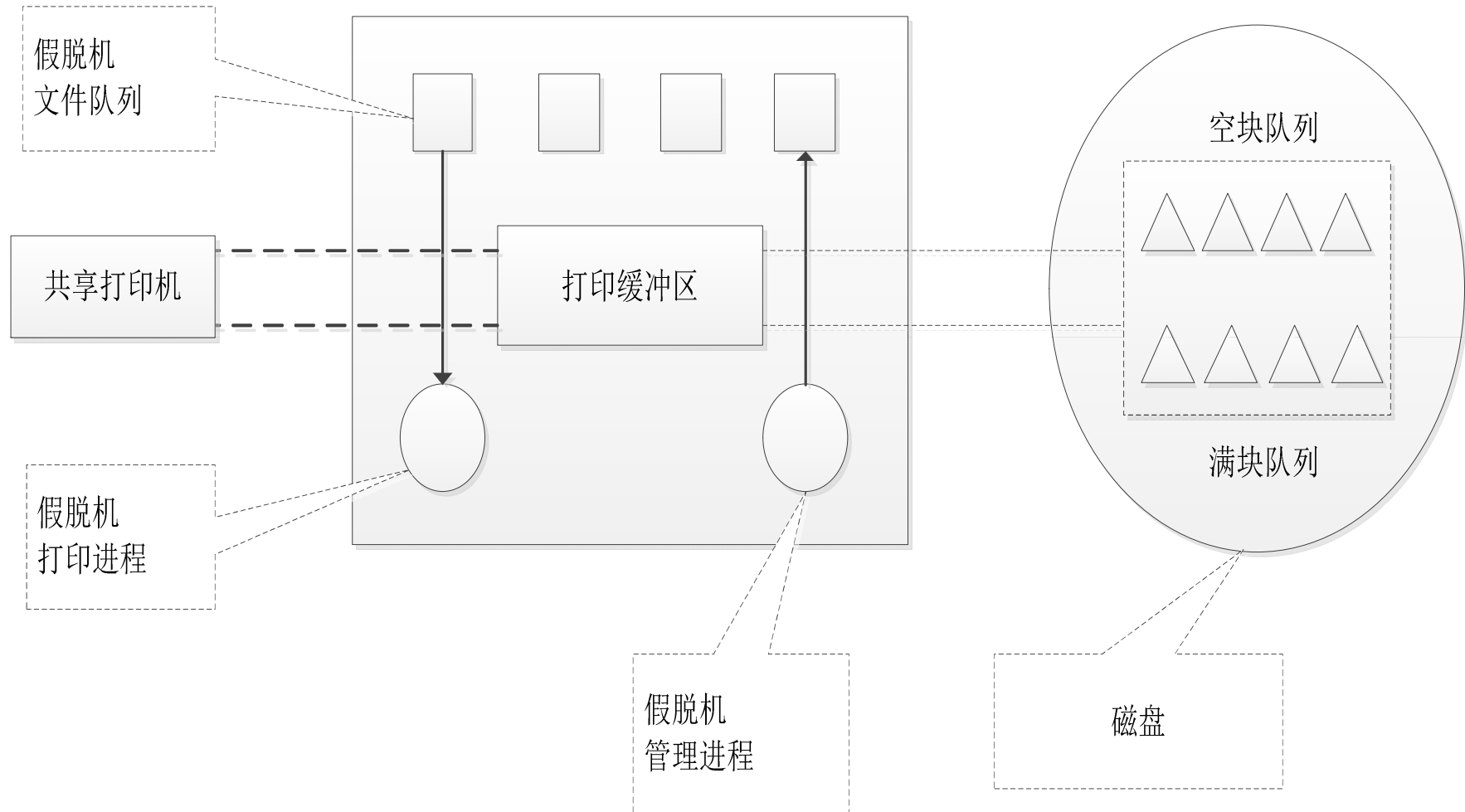
- 设备无关性
- 差错控制
- 同步/异步传输
- 处理独占设备和共享设备的I/O操作



西安科技大学

XI' AN UNIVERSITY OF SCIENCE TECHNOLOGY

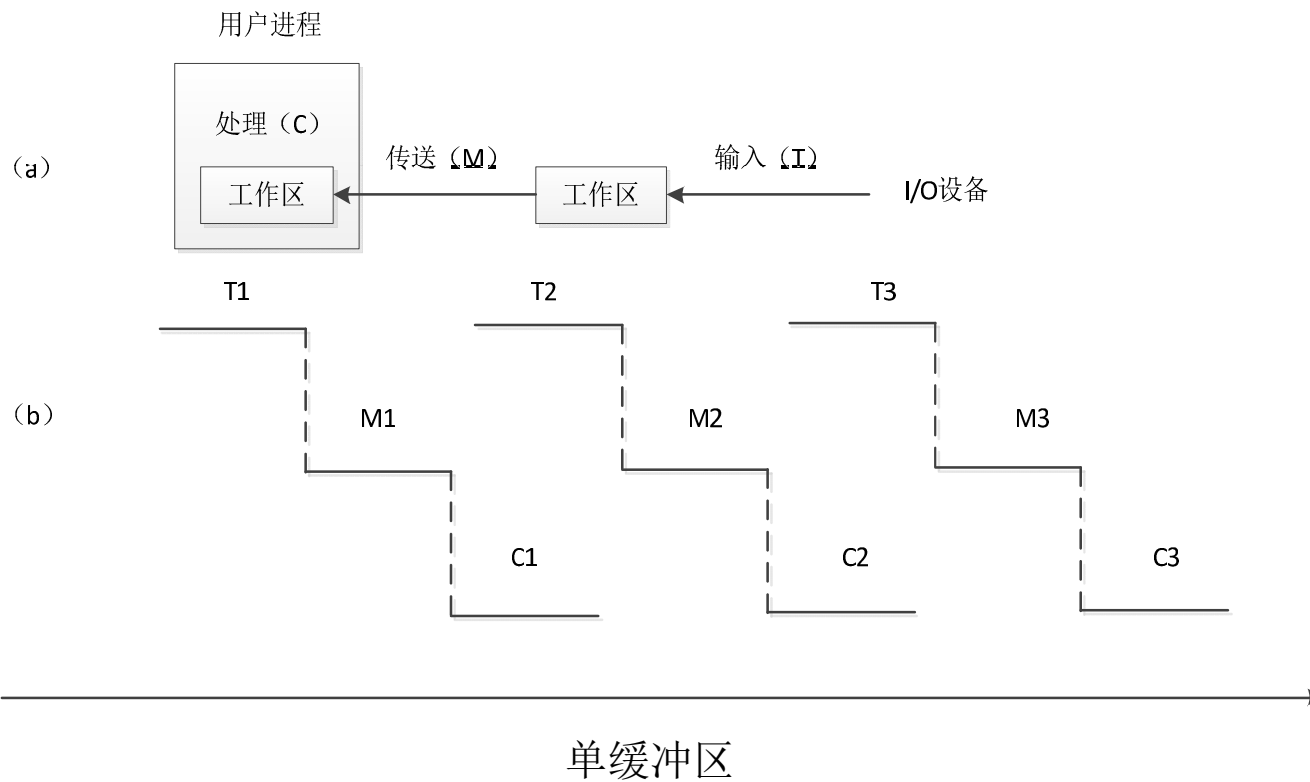
假脱机 (Spooling) 系统，它是多道程序设计中处理独占设备的一种方法。例如，打印机是经常用到的输出设备，属于独占设备。如果一个进程申请并分配到打印机，在它释放打印机前，其它进程无法再使用这台打印机。然而，利用假脱机技术可以将**它改造为一台供多个用户共享的打印设备**，从而提高设备的利用率，也方便了用户使用。

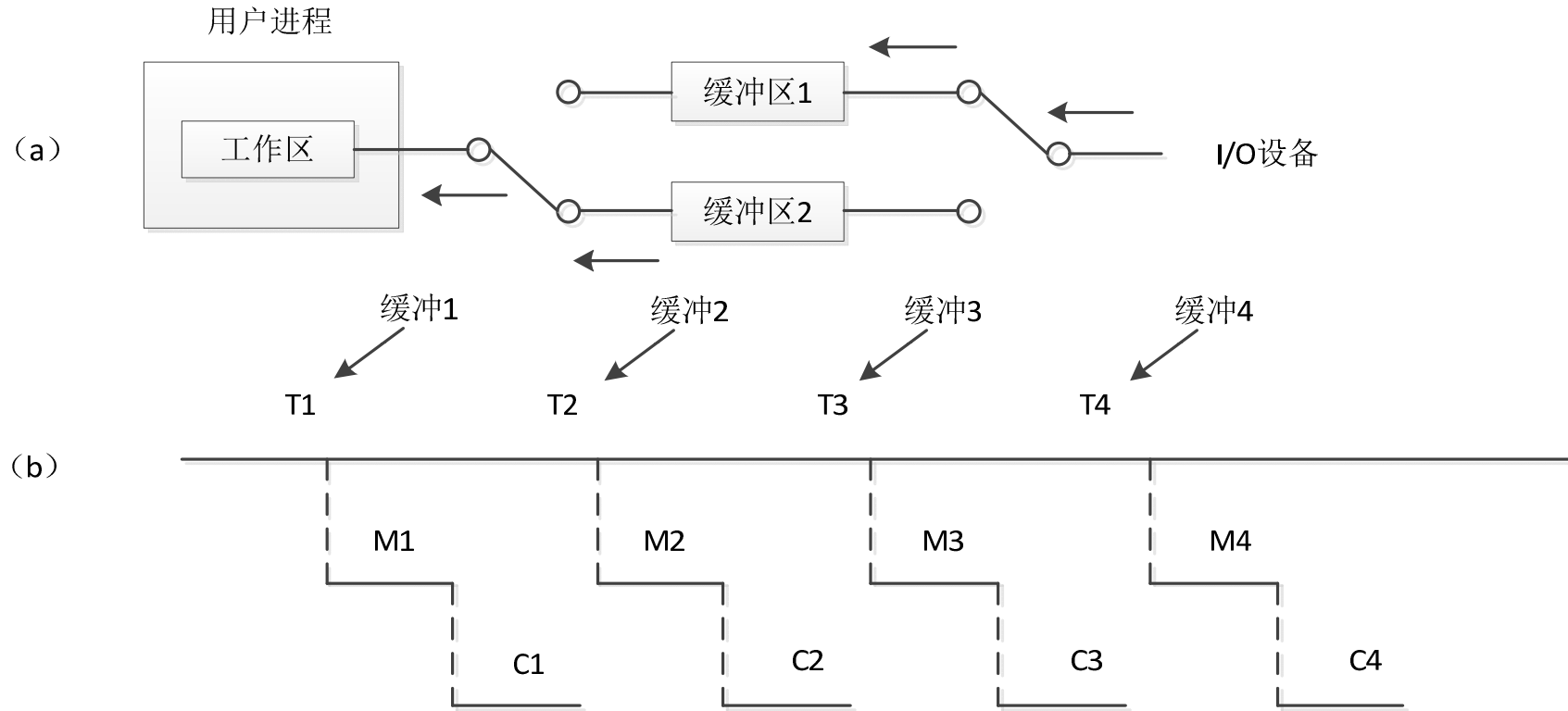




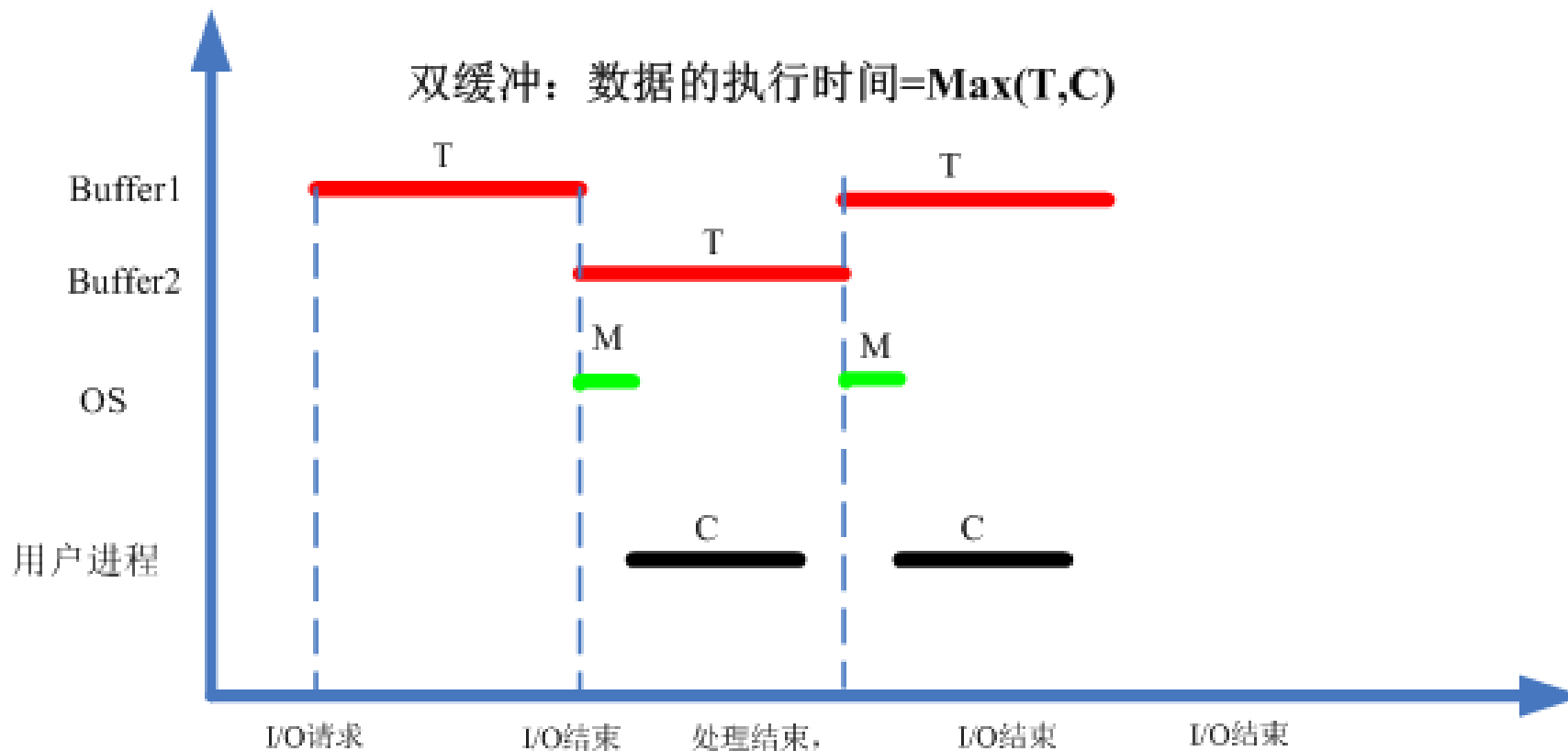
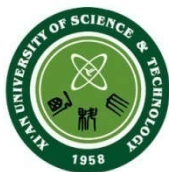
§ 7.4 缓冲处理技术

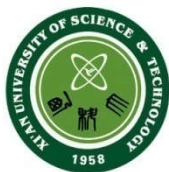
缓冲区是一个由操作系统内核进行管理，而且位于内核存储区域。





双缓冲区



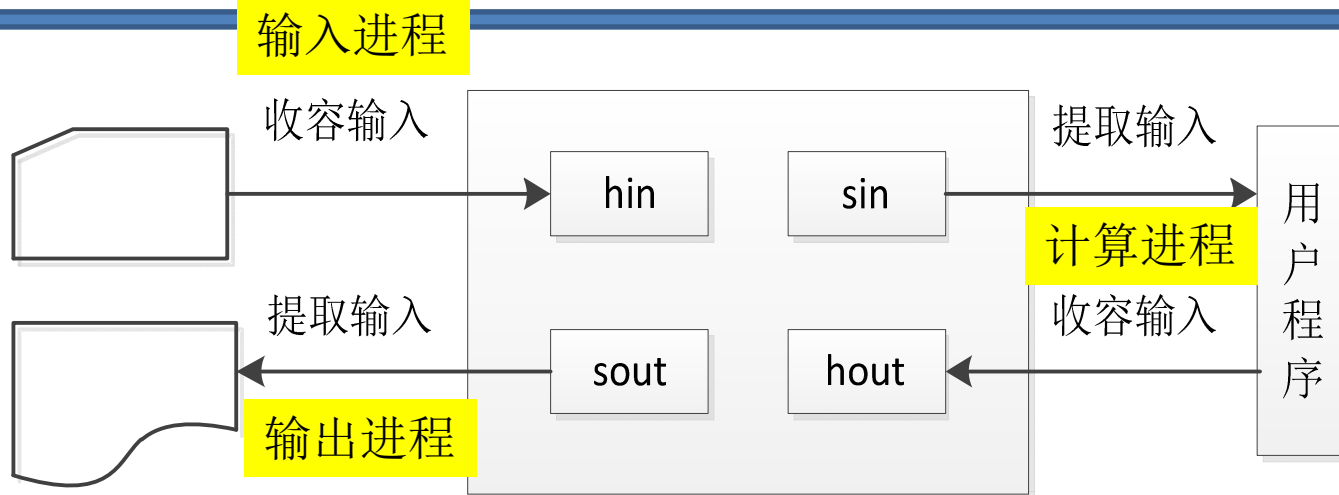


7.4.4 缓冲池

为了提高缓冲区的利用率，目前广泛使用既可用于输入又可用于输出的公用**缓冲池**，在池中设置了多个可供若干个进程共享的缓冲区。缓冲池与缓冲区的区别在于：缓冲区仅仅是一组内存块组成的链表，而**缓冲池是一个管理缓冲区的机构，其中不仅包括多个缓冲区队列，而且包括一组管理缓冲区的数据结构和对缓冲区进行管理的操作。**

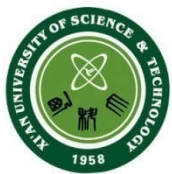
缓冲池包括三个缓冲区队列：

- 空白缓冲区队列emq。由空缓冲区所链成的队列。
- 输入缓冲区队列inq。由装满输入数据的缓冲区所链成的队列。
- 输出缓冲区队列outq。由装满输出数据的缓冲区所链成的队列。



- 收容输入。输入进程::Getbuf(emq); take it as hin; put data to hin;Putbuf(inq,hin)
- 提取输入。计算进程::Getbuf(inq); take it as sin; read data from sin;Putbuf(emq,sin)
- 收容输出。计算进程::Getbuf(emq);take it as hout; put data to hout;Putbuf(outq,hout)
- 提取输出。输出进程::Getbuf(outq); take it as sout; read data from sout;Putbuf(emq,sout)

由于Getbuf和Putbuf有可能对同一队列操作，因此要注意并发问题。



§ 7.5 磁盘驱动调度

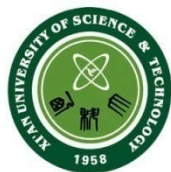
磁盘的访问时间划分成以下三部分：

• 寻道时间 $T_s = m \times n + s$

• 旋转时间 T_r

• 传输时间 $T_t = \frac{b}{rN}$

$$T_a = T_s + \frac{1}{2r} + \frac{b}{rN}$$



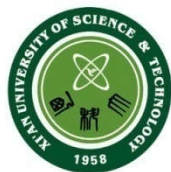
7.5.2 早期的磁盘调度算法

下一柱面号	移动距离 (柱面数)
55	45
58	3
39	19
18	21
90	72
160	70
150	10
38	112
184	146
平均寻道长度 55.3	

图 7-31 FCFS 调度算法

下一柱面号	移动距离 (柱面数)
90	10
58	32
55	3
39	16
38	1
18	20
150	132
160	10
184	24
平均寻道长度 27.5	

图 7-32 SSTF 调度算法



向柱面号增加的方向移动	
下一柱面号	移动距离 (柱面数)
150	50
160	10
184	24
90	94
58	32
55	3
39	16
38	1
18	20
平均寻道长度 27.8	

图 7-33 SCAN 调度算法

向柱面号增加的方向移动	
下一柱面号	移动距离 (柱面数)
150	50
160	10
184	24
18	166
38	20
39	1
55	16
58	3
90	32
平均寻道长度 35.8	

图 7-34 CSCAN 调度算法



§ 7.7 I/O进程控制

