

第六章 设备管理

- **要求：**掌握I/O控制的原理、控制方式、设备的使用方法，输入/输出的层次结构；掌握缓冲技术，存储设备、常用磁盘调度算法；了解磁盘阵列
- **重点与难点：**
- 三种不同的I/O控制方式，三种不同的设备使用方法
- 层次结构，设备驱动程序，缓冲技术
- 磁盘调度的方法，盘阵选择

主要内容

- **1 I/O硬件**
 - **1.1 常用设备分类**
 - **1.2 设备控制器**
 - **1.3 I/O控制方式**
- **2 设备I/O子系统**
 - **2.1 设备的使用方法**
 - **2.2 I/O层次结构**
 - **2.3 缓冲技术**
 - **2.4 设备分配**
 - **2.5设备驱动程序**
- **3 存储设备**
 - **3.1 常见存储外设**
 - **3.2 磁盘调度**
 - **3.3 磁盘阵列**

6.1 输入/输出硬件概念

1. 常见I/O设备的分类

- 人-机交互设备 (字符设备, 发送接收以字符方式);
- 存储类型设备 (块设备, 读写以数据块方式);
- 网络通信的设备

不同的分类决定不同的管理方法:

- 数据传输单位不同
- 控制的复杂性: 设备无关性
- 设备的使用目的不同

(1) 按信息交换的单位分类 (UNIX或Linux操作系统)

- 字符设备 (character device)
 - 以字符为单位进行输入输出设备
 - 在UNIX系统下，用ls -l /dev 列目录时，屏幕显示的第一个字符为c。
- 块设备 (block device)
 - 以字符块为单位进行输入输出的设备，在不同的操作系统或同一操作系统的不同版本中，块的大小是不一样的
 - 在UNIX操作系统中，用ls -l /dev列目录时，第一字符为b。

(2) 按资源特点分类

- **独享设备 (monopolize device,independent device)**
- **共享设备 (sharing device)**
- **虚拟设备 (virtual device)**

（3）按设备硬件物理特性分类

- **顺序存取设备（sequential access device）**
 - 存取时间与物理上当前位置有关
 - 例如：磁带
- **直接存取设备（direct access device）**
 - 存取时间与物理上的当前位置关系不大
 - 例如：磁盘

(4)按设备使用分类

- 物理设备（**physical device**）
 - 一般由操作系统分类命名
 - 如dos中的CON、PRN、LPT、COM
- 逻辑设备（**logical device**）
 - 用户可以重新命名的设备
- 伪设备（**pseudo device**）
 - 泛指具有特定用途的逻辑设备，如模拟的空设备，可以在内存或外存上临时开辟一个区域，充当一个“设备”用
 - 在dos中，NUL就是一个逻辑空设备。如COPY NUL filename 相当于将filename的内容清除

Null 设备

忽略发送给它的一切数据。向它写数据时，它始终报告写操作是成功的；向它读取数据时，该设备总是返回没有数据。如果你想确保一个命令执行，但又不想把文字输到屏幕上，如下：

```
[alex@example.com ~]$  
cat shopping.txt > /dev/null
```


2. 设备控制器（I/O部件）

I/O设备通常包含一个机械部件和一个电子部件。电子部件被称做I/O部件或设备控制器（当控制多设备时：又叫总线控制器，通道控制器）。

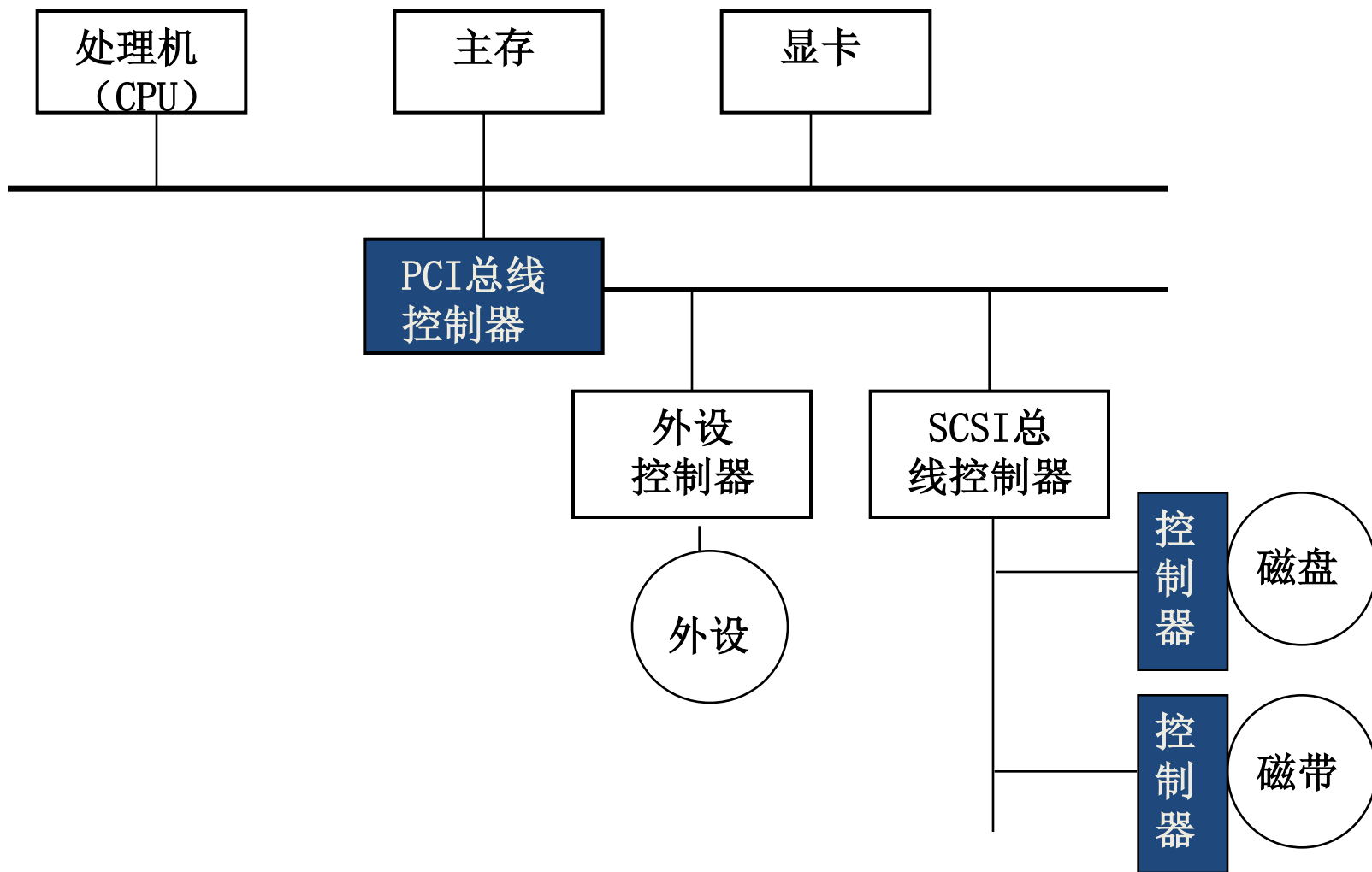
操作系统一般只与控制器打交道，而非设备本身

智能I/O控制器

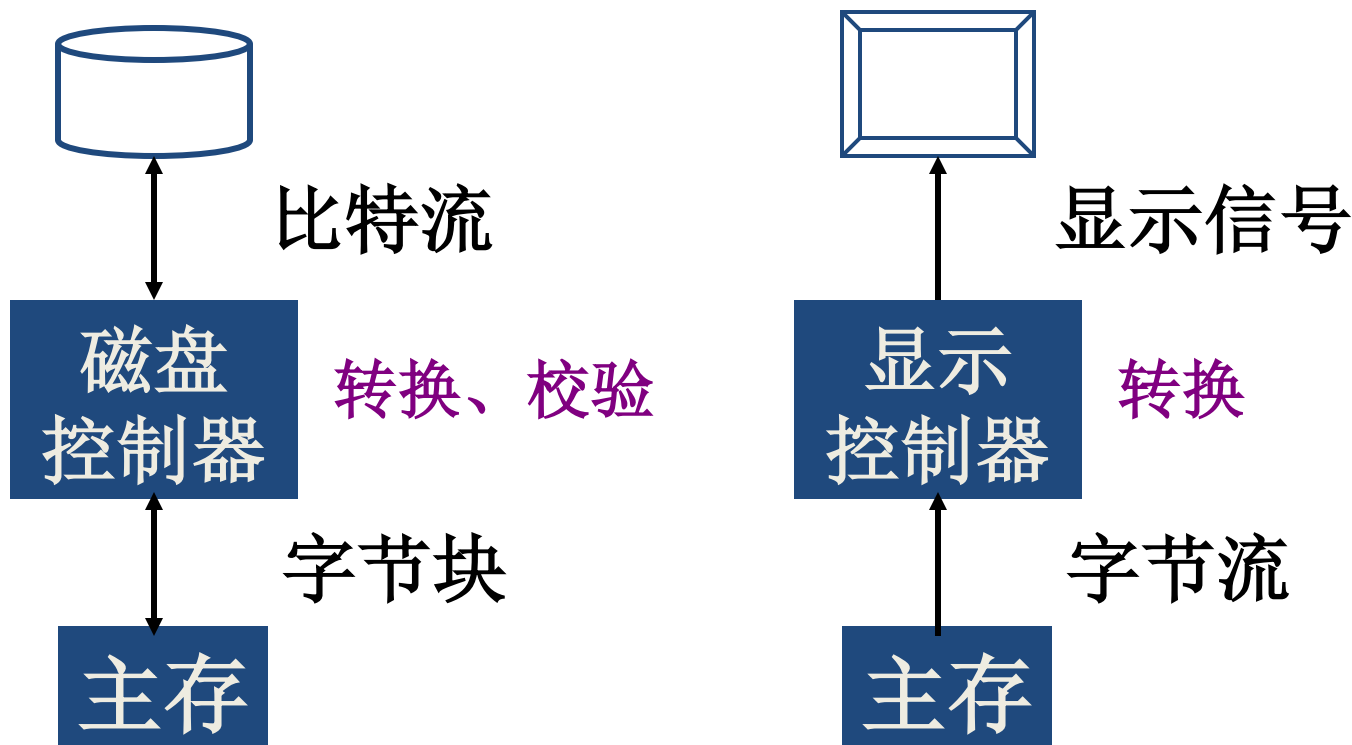
MODEL **2500**



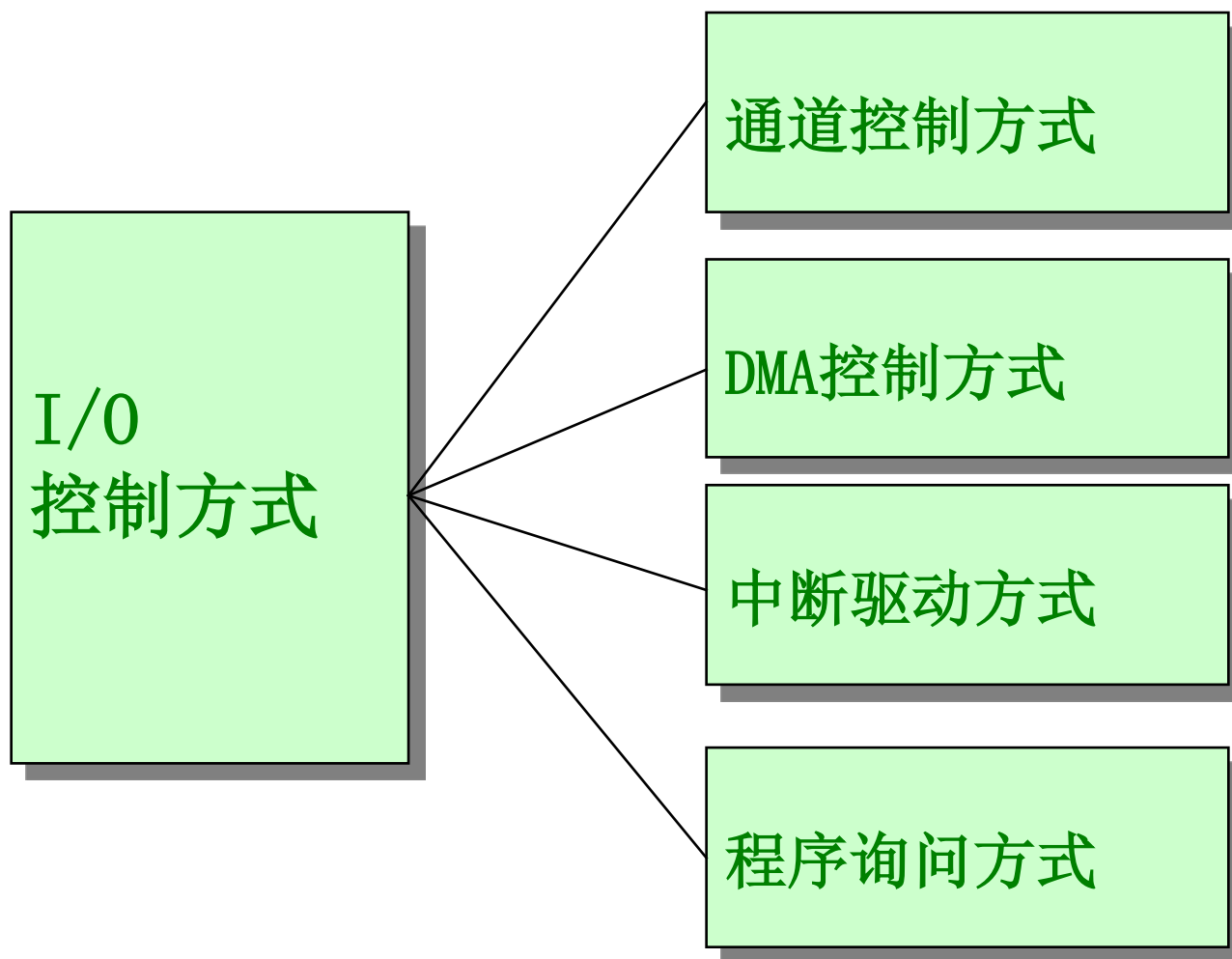
连接CPU、主存、设备控制器和I/O设备模型



控制器的任务是在外部设备与CPU（内存）之间完成比特流（外部信号）和字节流（块）之间的转换
组成：



3.输入/输出控制方式

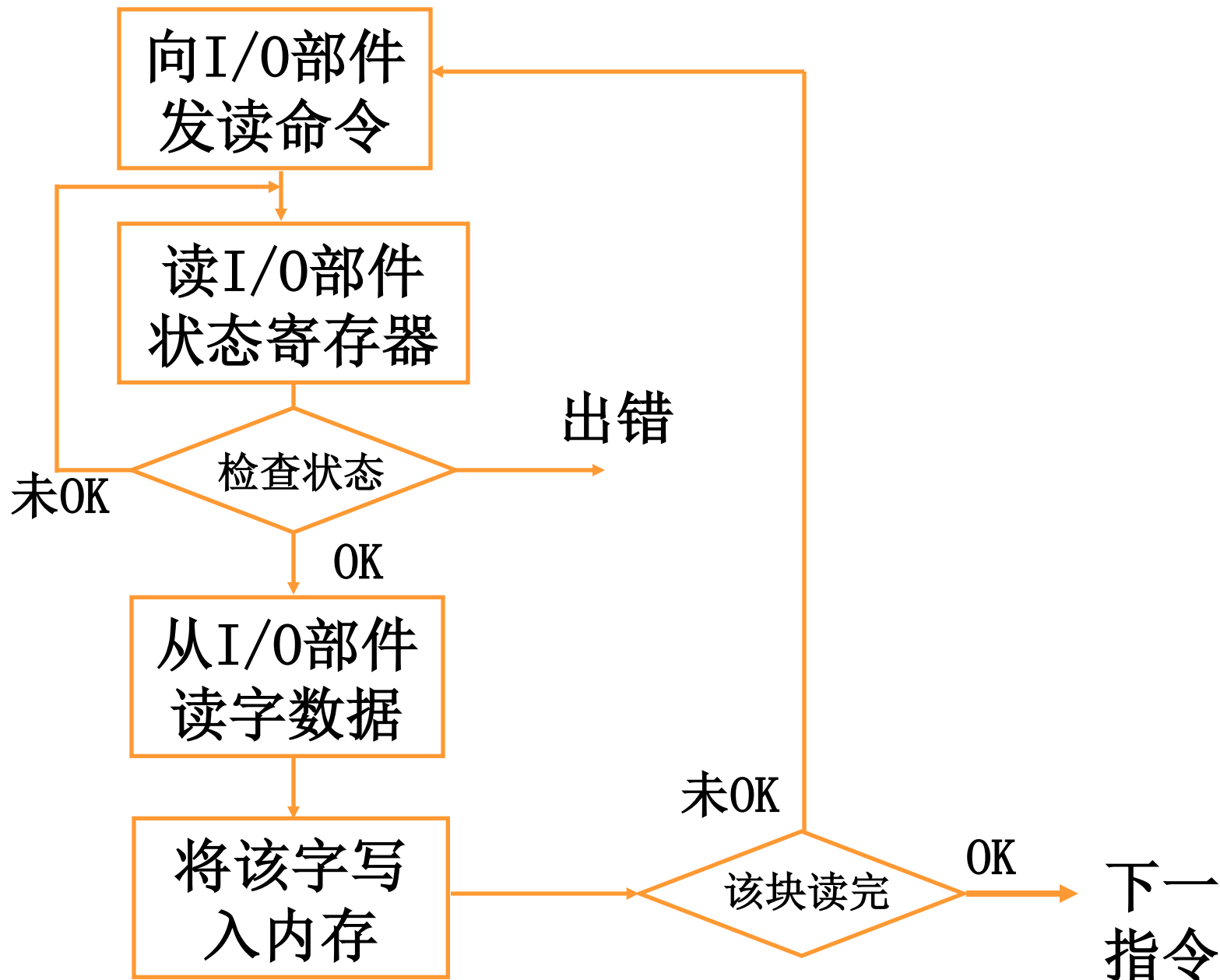


(1) 程序直接控制方式

CPU直接控制I/O操作的全过程，包括测试设备状态、发送读写命令、传输数据

处理机指令集：包括控制类、测试类、读写类I/O指令

控制步骤：I/O部件接收到相应的命令后，置于I/O状态寄存器的相应位置上。**随着操作的执行更改状态位**，但并不通知CPU。I/O数据通过CPU寄存器转发。



(2) 中断控制方式

CPU向I/O部件发出命令后，转去做其他有用的工作。当I/O部件准备好数据后，利用中断通知CPU，再由CPU完成数据传输。

CPU不必反复测试寄存器状态，节约了时间。但总体看来，中断控制方式仍然消耗大量的CPU时间，因为每个字的数据传输都必须经过CPU寄存器转发。

中断

向I/O部件
发读命令

读I/O部件
状态寄存器

检查状态

出错处理

OK

从I/O部件
读字数据

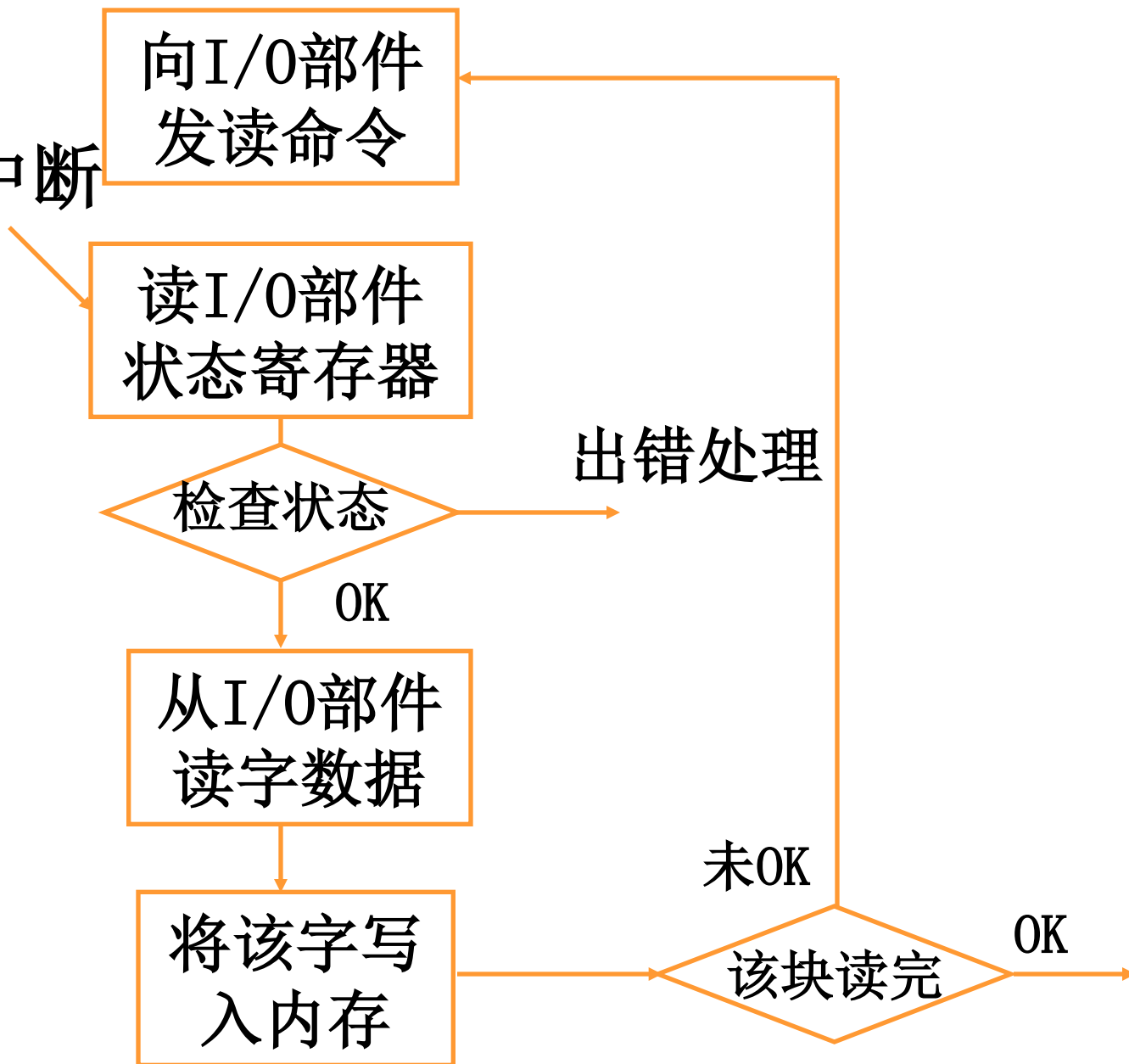
将该字写
入内存

未OK

该块读完

OK

下一
指令



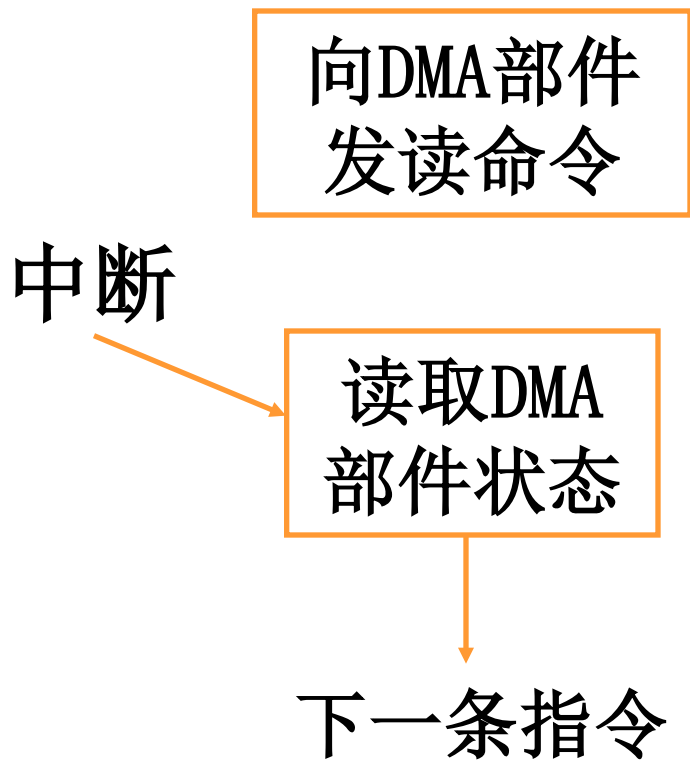
(3) DMA方式

前两种方式的缺陷：

- I/O的传输速率受CPU测试或中断响应的速度限制；
- CPU为管理I/O耗费大量时间。

DMA（直接内存存取）负责完成整个I/O操作，无需再经CPU寄存器转发，并在全部传输结束后向CPU发中断信号

CPU向DMA部件发送I/O命令后，即可进行其他工作。给DMA的命令中应包括：操作类别、I/O设备的地址、读写数据在内存中的首地址、字数。



DMA的功能可以以独立的DMA部件在系统I/O总线上完成，也可整合到I/O部件中完成。

读写内存时，DMA部件需要控制总线，CPU可能在涉及存储访问时因此而忙等待。

(4) 通道方式

采用通道后的I/O操作过程

- CPU在执行主程序时遇到I/O请求，它启动指定通道上选址的外围设备，一旦启动成功，通道开始控制外围设备进行操作。
- CPU就可执行其他任务并与通道并行工作，直到I/O操作完成。通道发出操作结束中断时，CPU才停止当前工作，转向处理I/O操作结束事件。

通道方式

通道三种类型

- 字节多路通道。
- 选择通道。
- 数组多路通道。

输入/输出控制方式的发展过程

CPU直接控制外部设备



引入I/O部件，CPU直接控制I/O部件



引入中断驱动方式



引入DMA



I/O通道或I/O处理机

6.2 设备输入/输出子系统

6.2.1 设备的使用方法

一、设备相关系统调用简介

1.申请设备

该系统调用中有参数说明**要申请的设备名称**，操作系统处理该系统调用时，会按照设备特性（是独占还是分时共享式使用）及设备的占用情况来**分配设备**，返回申请是否成功标志。

2.将数据写入设备

3.从设备读取数据

4.释放设备。这是申请设备的逆操作。

说明：

- 上述的系统调用主要用于对人-机交互类慢速外设的使用。
- 对于存储类外设，用户程序一般通过对文件的访问，由文件管理模块读写存储外设间接使用它们，系统也提供直接使用存储类外设的接口。
- 对于网络通信外设，用户级程序也不直接使用它们，用户通过SOCKET通信系统调用接口调用TCP/IP层程序，由IP层程序选择调用网络通信设备驱动程序。

在UNIX中，也可以用如下的系统调用将数据直接写入软盘中：

`fd=open (“/dev/fd0”, O__RDRW);` 申请软盘，
`/dev/fd0`代表软盘。

`lseek (fd, 1004, 0);` 将软盘当前I/O位置定位到1024字节位置。

`Write (fd, buffer, 36);` 将用户缓冲区buffer中的36个字节写入软盘1024~1059字节。

.....

`close (fd) ;` 释放软盘。

显然，这样的使用方式**绕过了文件管理，而直接读 / 写软盘空间**。当然你必须清楚软盘的什么位置存放了什么信息，才能做到正确的读写。

二、独占式共享使用设备

独占式共享使用设备是以一次设备使用过程（包含多次I/O操作）为单位使用设备。

在申请设备时，如果设备空闲，就将其独占，不再允许其他进程申请使用，一直等到该设备被释放，才允许被其他进程申请使用。

未来展望

- 要使计算机能看、能说、能听、并与外部设备密切相关
- 和目前流行的小闹钟大小相当，甚至小到在身上、衣服上到处可安装
- 电脑象电话机一样等待你，计算机联网工作就象打电话、看电视一样方便
- 显示器大到能挂在墙上或随意放大缩小，小到能放在提包里或衣袋里
- 每台计算机要能连接更多的复杂外部设备，并具有自动的“时空感应”控制功能



立体显示和平面显示不同，最重要的就是需要再现一个立体模型的三维数据，包括形状，位置等等信息

挂在墙上不用油墨的“报纸”



新闻中只有头条标题和正文是由E-Ink电子油墨面板显示，图片则由普通LCD显示屏负责。

LG的涂鸦液晶

- **LG L1530TM是一款极具特色的LCD显示器，它支持手写输入，可以让用户直接在显示屏上工作，从而为用户带来了全新手绘涂鸦的应用体验。**



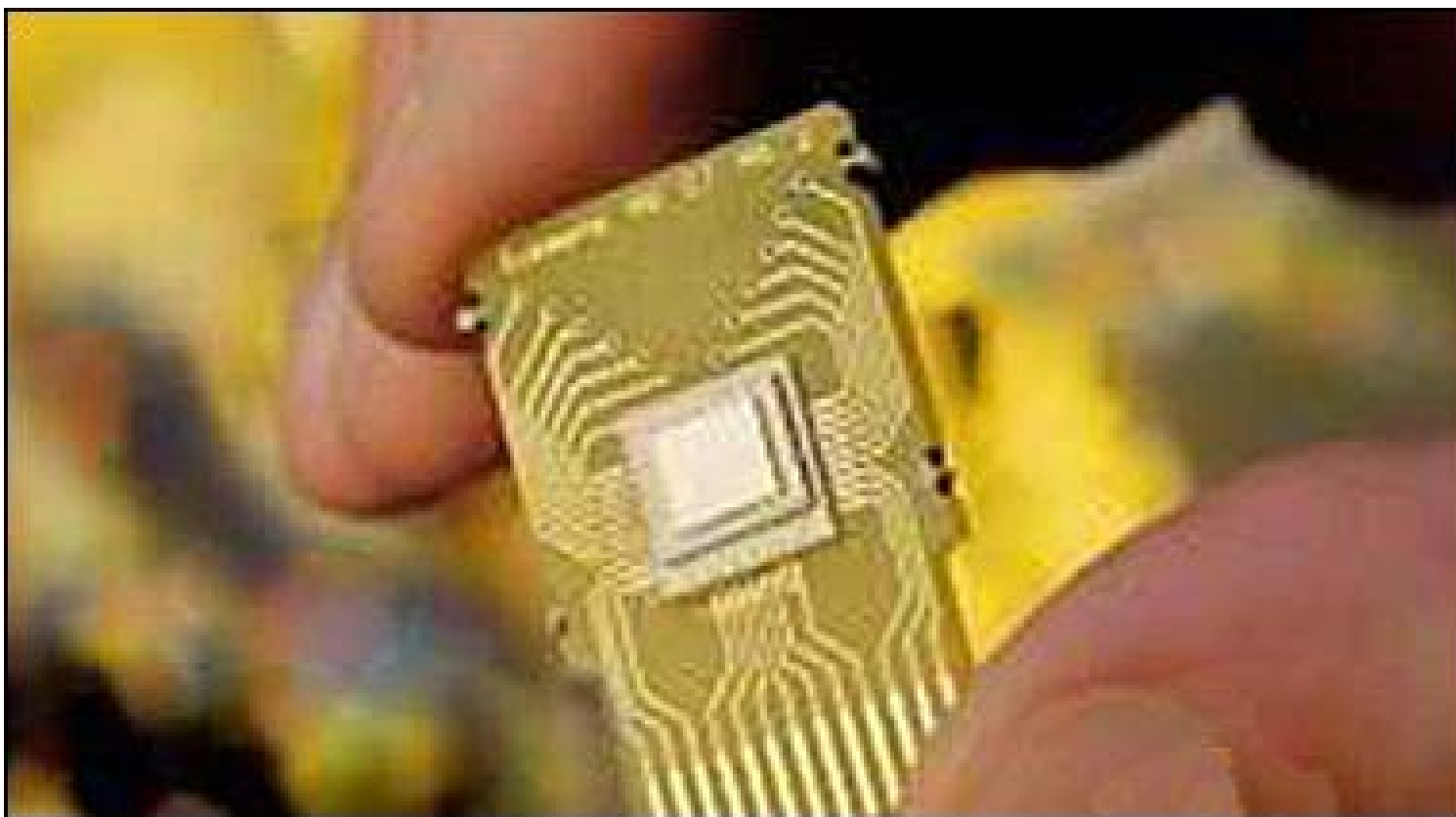
电子纸图书终端



显示器



主机板



纳米技术

燃料电池



**卡西欧宣布开发笔记本燃料电池，
并决定于2007年上市**

键盘



可以用脚来操作的鼠标

