第6章　输入/输出和中断技术

微机系统的经典结构

目前各种微机系统采用的基本上是计算机的经典结构—冯•诺依曼结构。结构特点是：

- 硬件上由运算器、控制器、存储器、输入设备和输出设备五大部分组成；
- 数据和程序以二进制代码的形式不加区别地存放在存储器中，存放位置由地址指定，地址码也为二进制形式；
- 控制器根据存放在存储器中的指令序列，即程序来工作，并由一个程序计数器控制指令的执行。控制器具有判断能力，能根据计算结果选择不同的动作流程。

微机系统的经典结构

微机系统在硬件上由运算器、控制器、存储器、输入和输出设备几大部分组成。而各部分间又是通过数据、地址和控制三条总线相连，故这种系统结构也称为三总线结构。

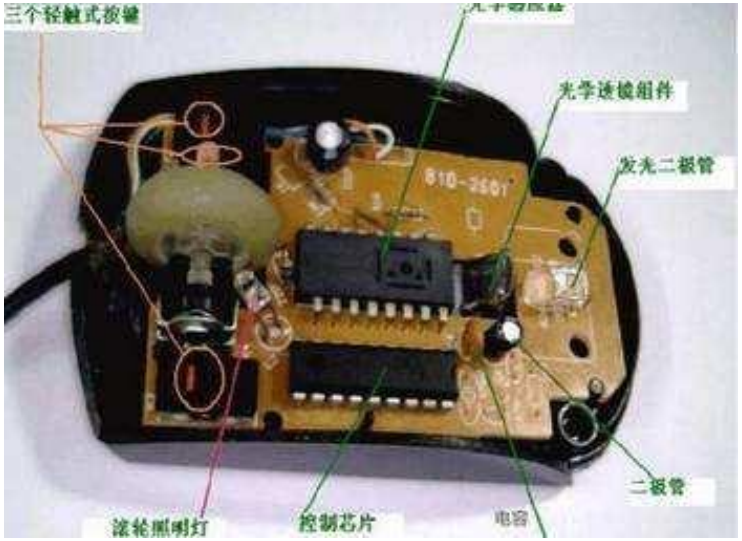
微处理器

存储器

I/O接口

总　线

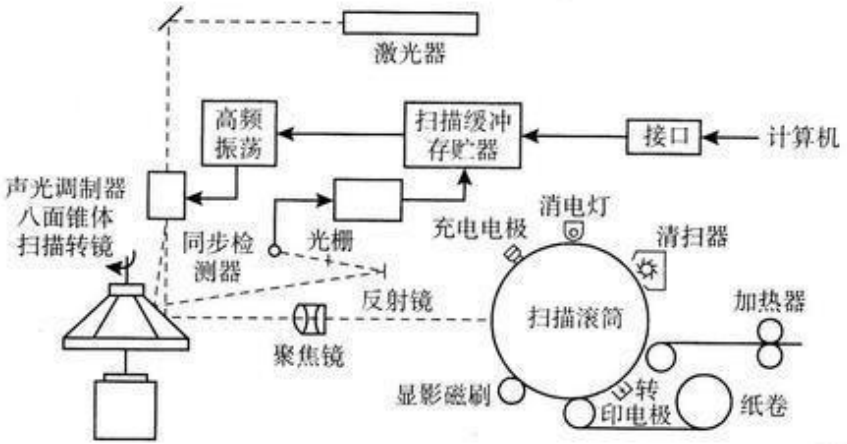
常见外接设备



常见外接设备


Esc
F1
F2
F3
F4
F5
F6
F7
F8
F9
F10
F11
F12
PraSt
Enter
Enter
Backspace
Delete
End
Enter
Tab
Q
W
E
R
T
Y
U
I
O
P
Enter
Enter
Enter
Enter
Enter
Enter
CapsLK
A
S
D
F
G
H
J
K
L
Enter
Shitf
Z
X
C
V
B
N
M
Shitf
Ctrl
Alt
Alt
Ctrl
Enter
0

常见外接设备


激光器
高频振荡
扫描缓冲存贮器
接口
计算机
声光调制器
八面锥体
扫描转镜
同步检测器
光栅
充电电极
消电灯
清扫器
反射镜
扫描滚筒
加热器
聚焦镜
显影磁刷
转印电极
纸卷

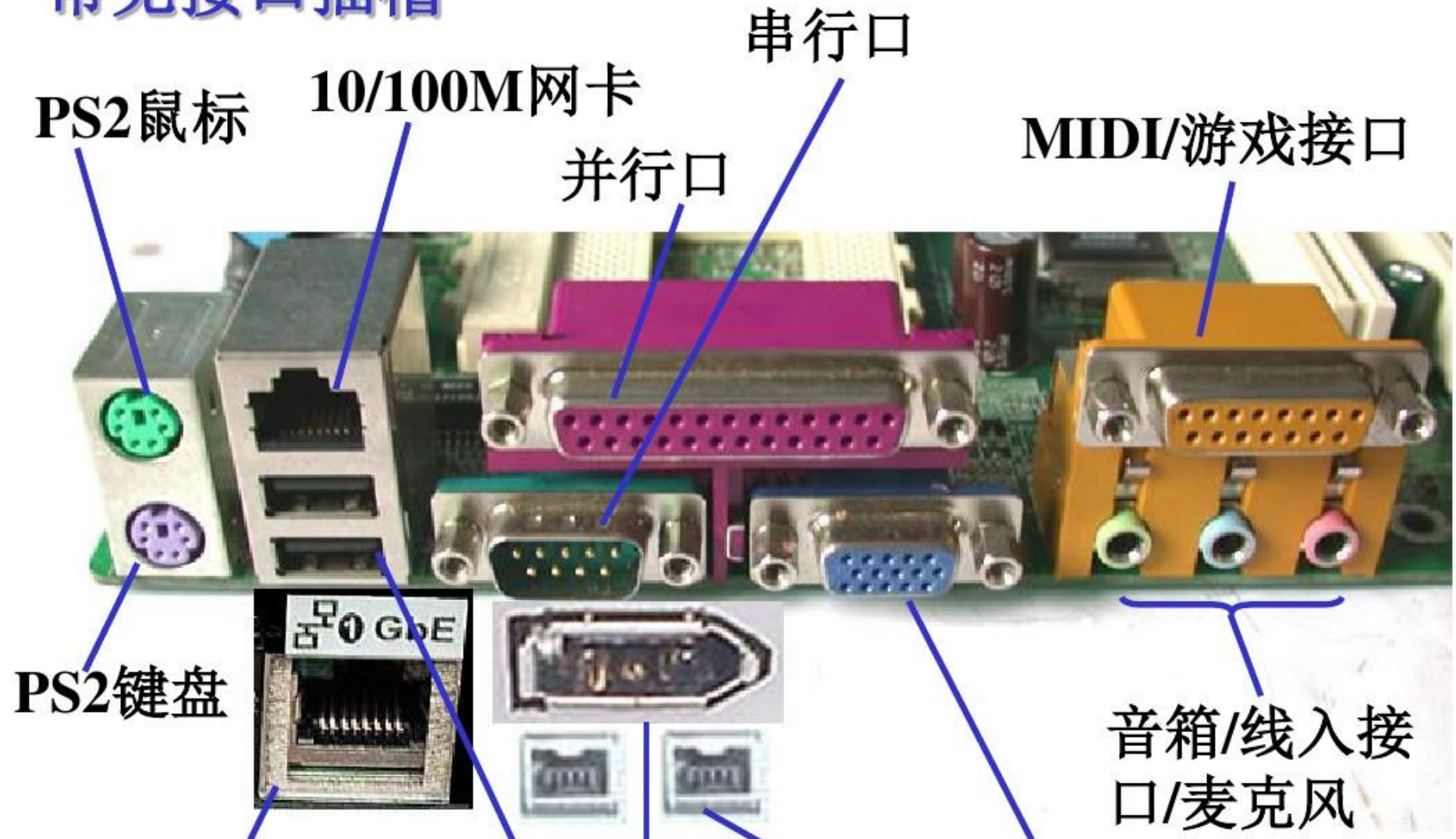
6.1 输入/输出系统概述

- 输入/输出是微机系统与外部设备进行信息交换的过程。处理器和主存储器之外的部分统称为输入/输出系统

- 6.1.1 I/O系统的特点
- 1．复杂性
  - 输入/输出设备的复杂性
  - 处理器本身和操作系统所产生的一系列随机事件也要调用输入/输出系统进行处理，如中断等。
  - 2. 异步性
  - 不同的外部设备有各自不同的定时和控制逻辑，且大都与CPU时序不一致，它们与CPU的工作通常都是异步进行的

- 3. 实时性
  - 处理器对每一个连接到它的外设或处理器本身，在需要或出现异常时，都要能够给予及时的处理，以防止错过服务时机使数据丢失或产生错误。
- 4. 与设备无关性
  - 由于输入/输出设备在信号电平、信号行驶、信号格式及时许等方面的差异，使得它们与CPU之间不能够直接地连接，而必须通过一个中间环节，这就是输入/输出接口。

常见接口插槽


串行口
10/100M网卡
PS2鼠标
并行口
MIDI/游戏接口
PS2键盘
音箱/线入接口/麦克风
千兆网
USB
1394
1394a
显示器接口

6.1.2 I/O接口的基本功能

- I/O接口就是将外设连接到系统总线上的一组逻辑电路的总称
- 1. I/O接口要解决的问题
  - 速度匹配问题
  - 信号电平和驱动能力问题
  - 信号形式匹配问题
  - 信息格式问题
  - 时序匹配问题
- 2. I/O接口的功能
  - I/O地址译码与设备选择
  - 信息的输入/输出
  - 命令、数据和状态的缓冲与锁存
  - 信息转换

6.1.3 I/O端口的编址方式

- 微机系统采用总线结构形式，即通过一组总线来连接组成系统的各个功能部件（包括CPU、内存、I/O端口），CPU、内存、I/O端口之间的信息交换都是通过总线来进行的，如何区分不同的内存单元和I/O端口，是输入/输出寻址方式所要讨论的问题。
- CPU与I/O接口进行通信实际上是通过I/O接口内部的一组寄存器实现的，这些寄存器通常称为I/O端口（I/O port）
- 端口包括3个类型：
  - 数据端口：CPU通过数据端口从外设读入数据（或向外设输出数据）
  - 状态端口：读入设备的当前状态
  - 命令（或控制）端口：向外设发出控制命令
- I/O地址：8086/8088CPU最多能管理64K个端口，每个端口分配一个地址
- 基地址：当一个外设有多个端口时，通常为其分配一个连续的地址块，其中最小的那个地址称为外设的基地址。

- **1．I/O端口与内存单元统一编址**
  - 又称为存储器映射编址方式，即把每个I/O端口都当做一个存储单元对待，端口与存储单元在同一个地址空间中进行编址。
  - 优点：可以用访问内存的方法来访问I/O端口；理论上所有用于内存的指令都可以用于外设
  - 缺点：外设占用了一部分内存地址空间，减少了内存可用的地址范围，对内存容量有潜在的影响。

- **2．I/O端口独立编址**
- **内存地址空间和外设地址空间是相互独立的。CPU在寻址内存和外设时，使用不同的控制信号来区分当前是对内存操作还是对I/O端口操作。**
- **特点：**
  - **（1）I/O端口的地址空间与内存地址空间完全独立**
  - **（2）I/O端口与内存使用不同的控制信号**
  - **（3）指令系统中设置了专门用于访问外设的I/O指令。**
- **例如：对存储器进行编址：0H~FFFFH**

  对外设进行编址：0H~FFFFH

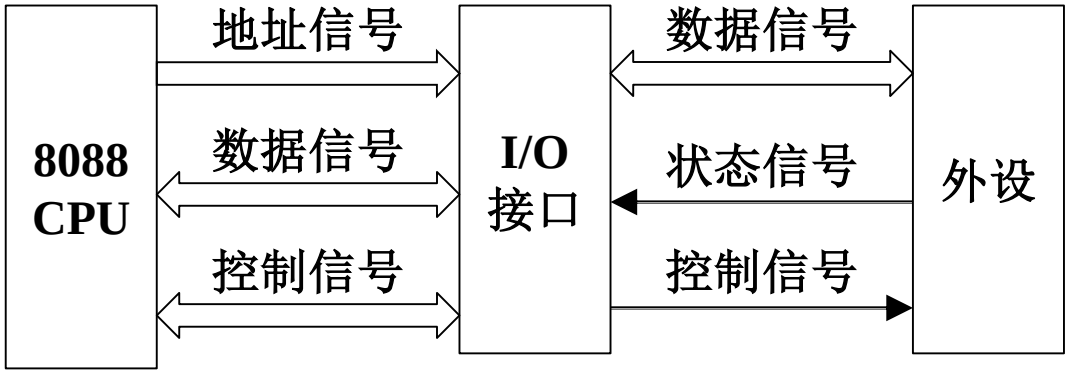
6.1.4 I/O端口地址的译码

- 译码：CPU首先需要将要访问端口的地址放到地址总线上，然后才能对其进行读写，将总线上的地址信号转换为某个端口的“使能”信号，这个操作就称为端口地址的译码
- 几点注意：
  - （1）8088CPU能够寻址的内存空间为1MB，故地址总线的全部20根信号线都要使用，高位用于确定芯片的地址范围，而低位用于片内寻址
  - （2）当CPU工作在最大模式下，对存储器的读写要求控制信号MEMR或MEMW有效；如果是对I/O端口读写，则要求控制信号IOR或IOW有效。
  - （3）地址总线上呈现的信号是内存的地址还是I/O端口的地址取决于8088CPU的IO/M引脚的状态。
- I/O地址译码的方式主要分为两种：用基本逻辑门电路构成译码器、用专门的译码器。

6.2 简单接口电路

6.2.1 接口电路的基本构成

- 输入接口：负责把信息从外部设备送入CPU的接口（端口）。要求输入接口必须要具有对数据的控制能力。若外设本身具有数据保持能力，通常可以仅用一个三态门缓冲器作为输入接口
- 输出接口（端口）：将信息从CPU输出到外部设备的接口（端口）。要求输出接口必须要具有数据的锁存能力。简单的输出接口一般由锁存器构成



6.2.2 三态门接口

- 三态门芯片：74LS244，由8个三态门构成。两个控制端E_1和E_2，各控制4个三态门。

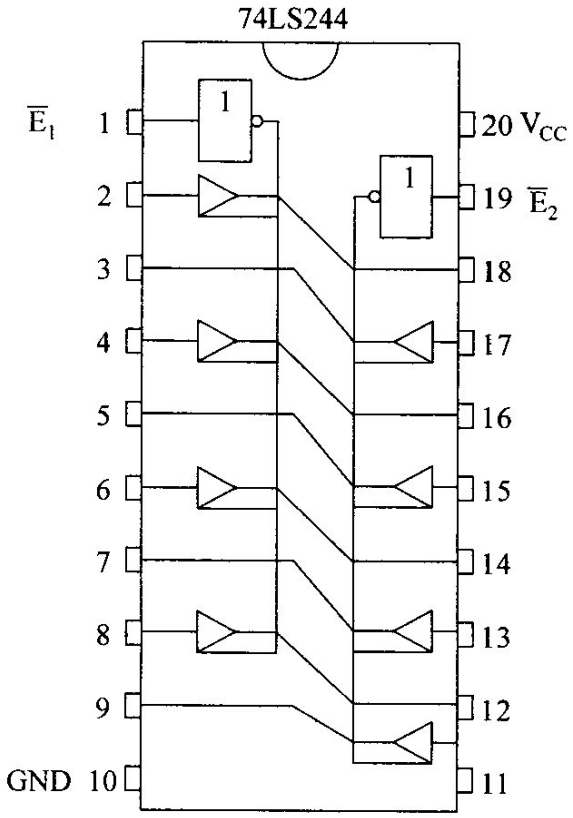


图 6-3　74LS244 芯片引脚图

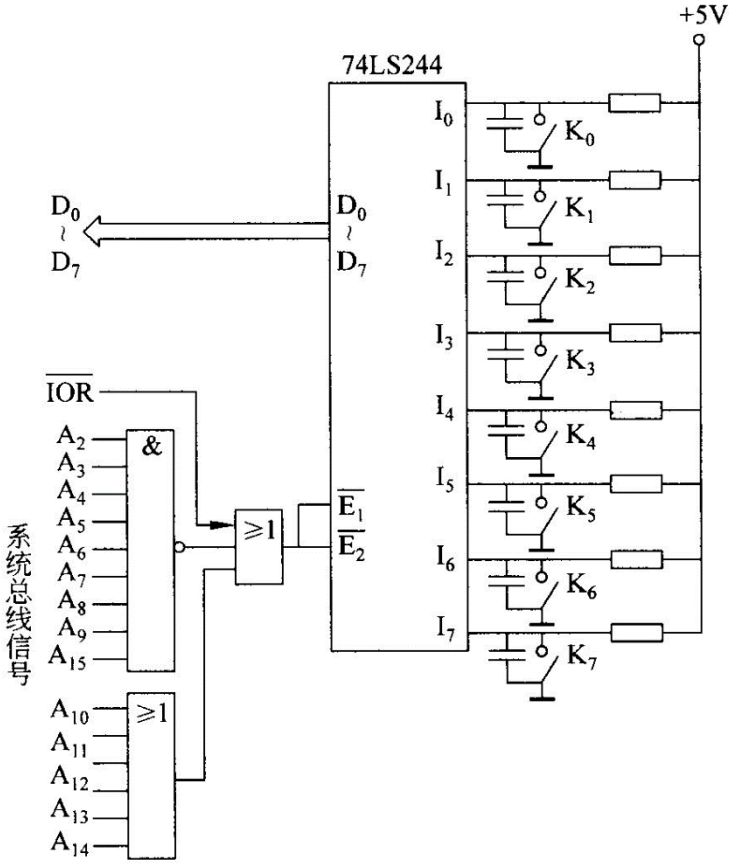


图 6-4　三态门作输入接口

- 例6-1，编写程序判断图6-4中的开关状态，如果所有的开关都闭合，则程序转向标号为NEXT1的程序段执行，否则转向标号为NEXT2的程序段执行。

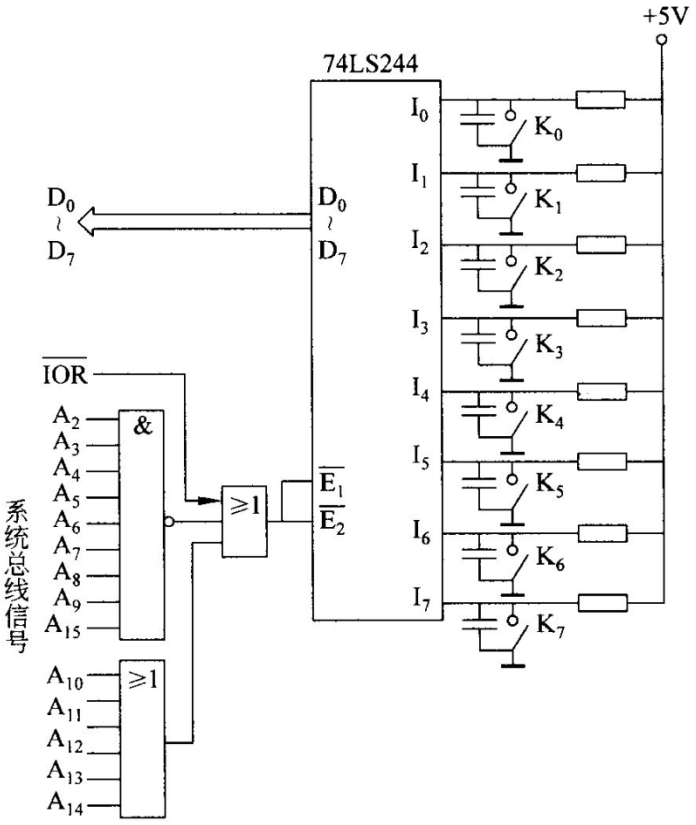
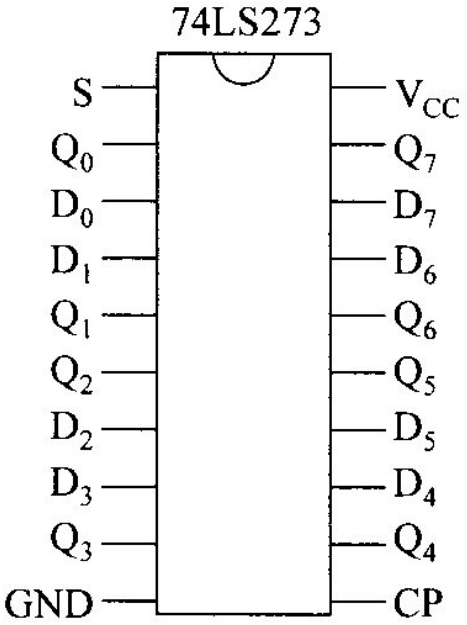


图 6-4　三态门作输入接口

```
MOV     DX, 83FCH
IN      AL,  DX
AND     AL,  0FFH
JZ      NEXT1
JMP     NEXT2
```

6.2.3 锁存器接口

- 三态门不具备数据的保存（或称锁存）能力，要求信号源能够将信号保持足够长的时间直到被CPU读取，所以一般只用作输入接口，而不能直接用作数据输出接口。
- 输出接口通常采用具有信息存储能力的双稳态触发器来实现。最简单的输出接口可用D触发器构成，如74LS273。



S	CP	D_i	Q_i
0	X	X	0
1	↑	1	1
1	↑	0	0

图 6-5　74LS273 引线图和真值表

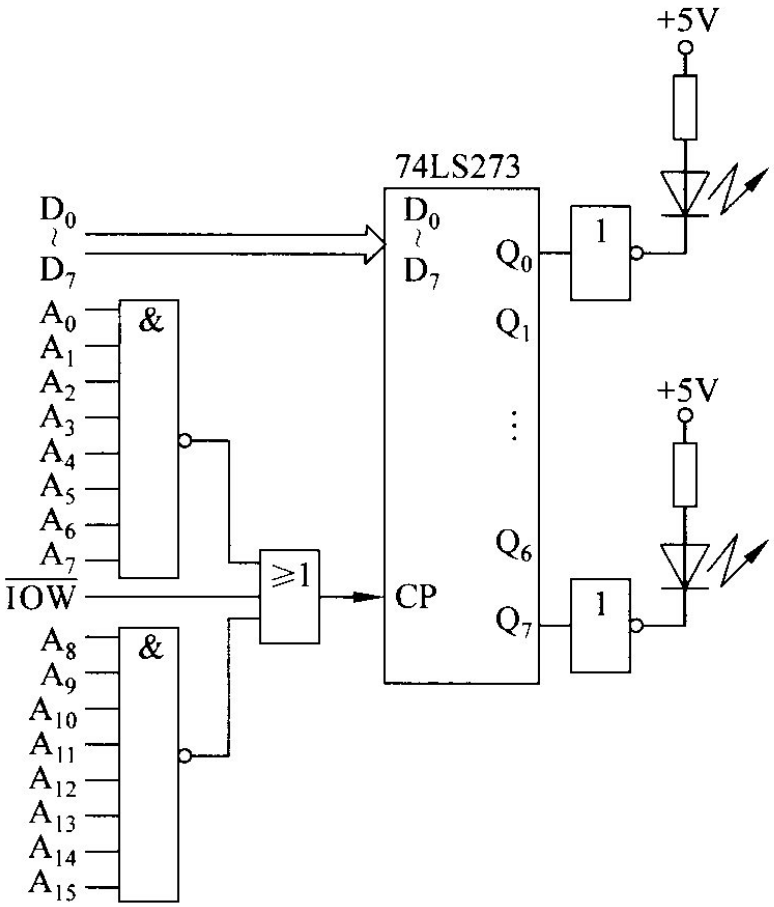
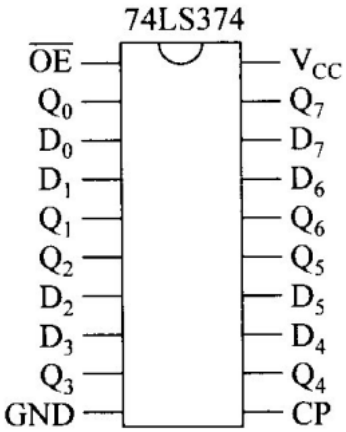


图 6-6　74LS273 作输出接口的应用

```
MOV     DX, 0FFFFH
MOV     AL, 01000001B
OUT     DX, AL
```



D_i	CP	$\overline{OE}$	Q_i
1	↑	0	1
0	↑	0	0
X	X	1	高阻

图 6-7　74LS374 引线图和真值表

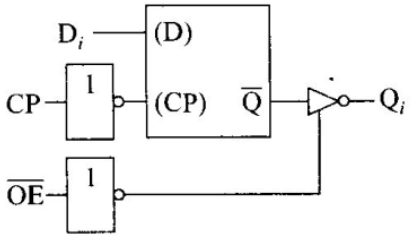
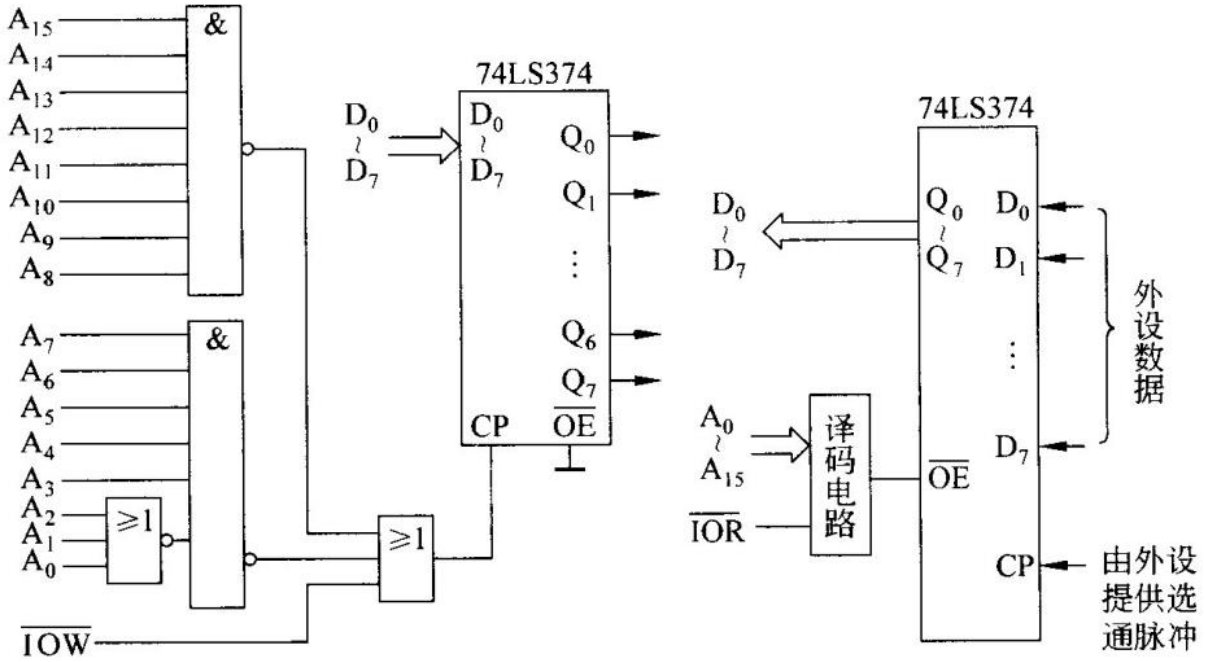


图 6-8　74LS374 内部结构



(a) 74LS374用作输出接口　　(b) 74LS374用作输入接口

图 6-9　74LS374 用作输入和输出接口

6.2.4 简单接口的应用举例

- 1. LED数码管
- 2. 应用与连接

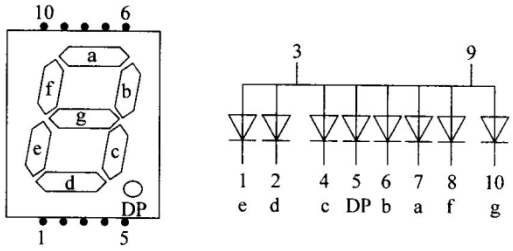


图 6-10　共阳极 LED 数码管示意图

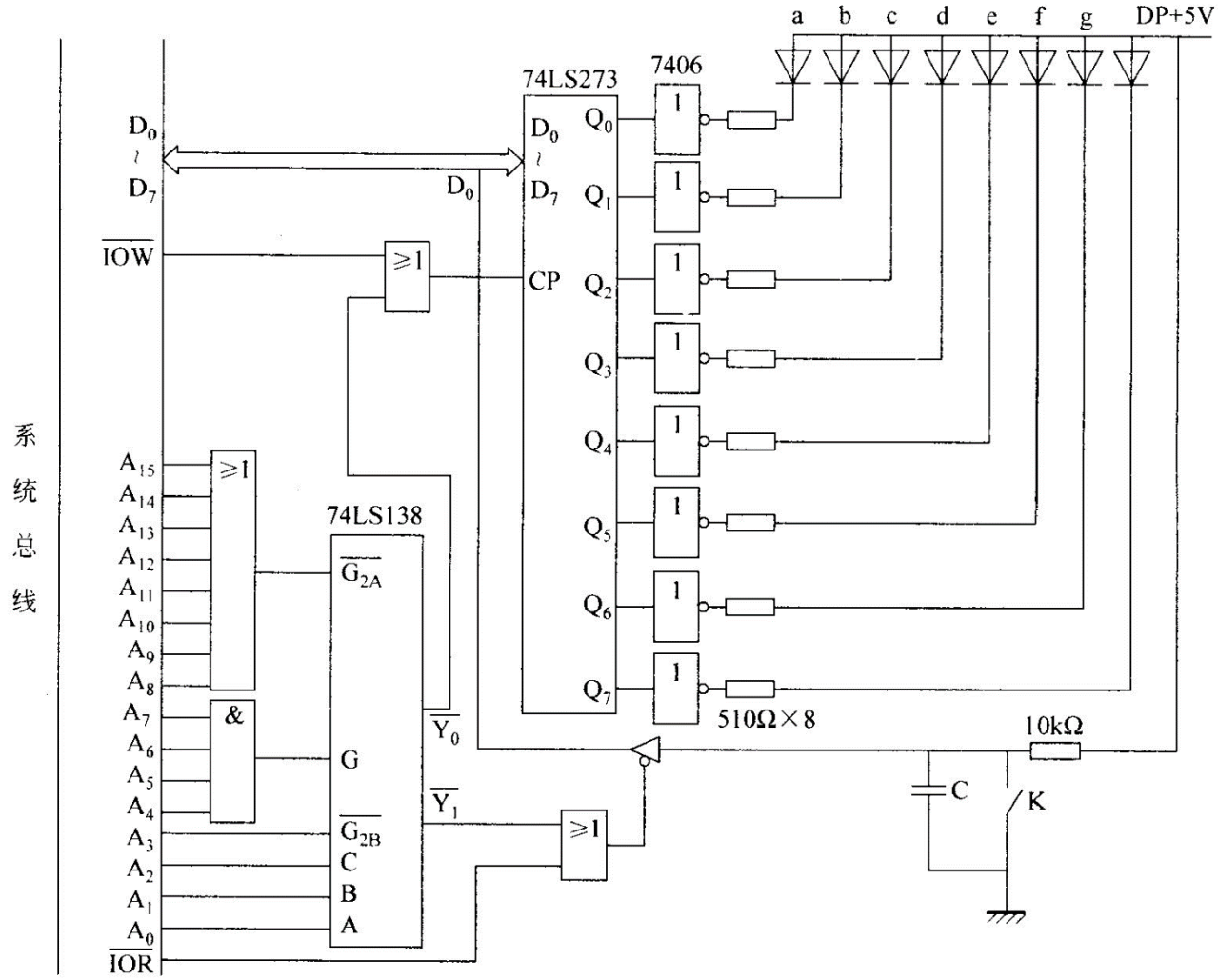


图 6-11　简单接口的应用

6.3 基本输入/输出方式

- 无条件传送
- 查询式传送

} 程序控制方式

- 中断方式传送
- 直接存储器存取（DMA）

- 其中无条件方式和查询方式又称为程序控制方式，指用输入/输出指令，来控制信息传输的方式，是一种软件控制方式。

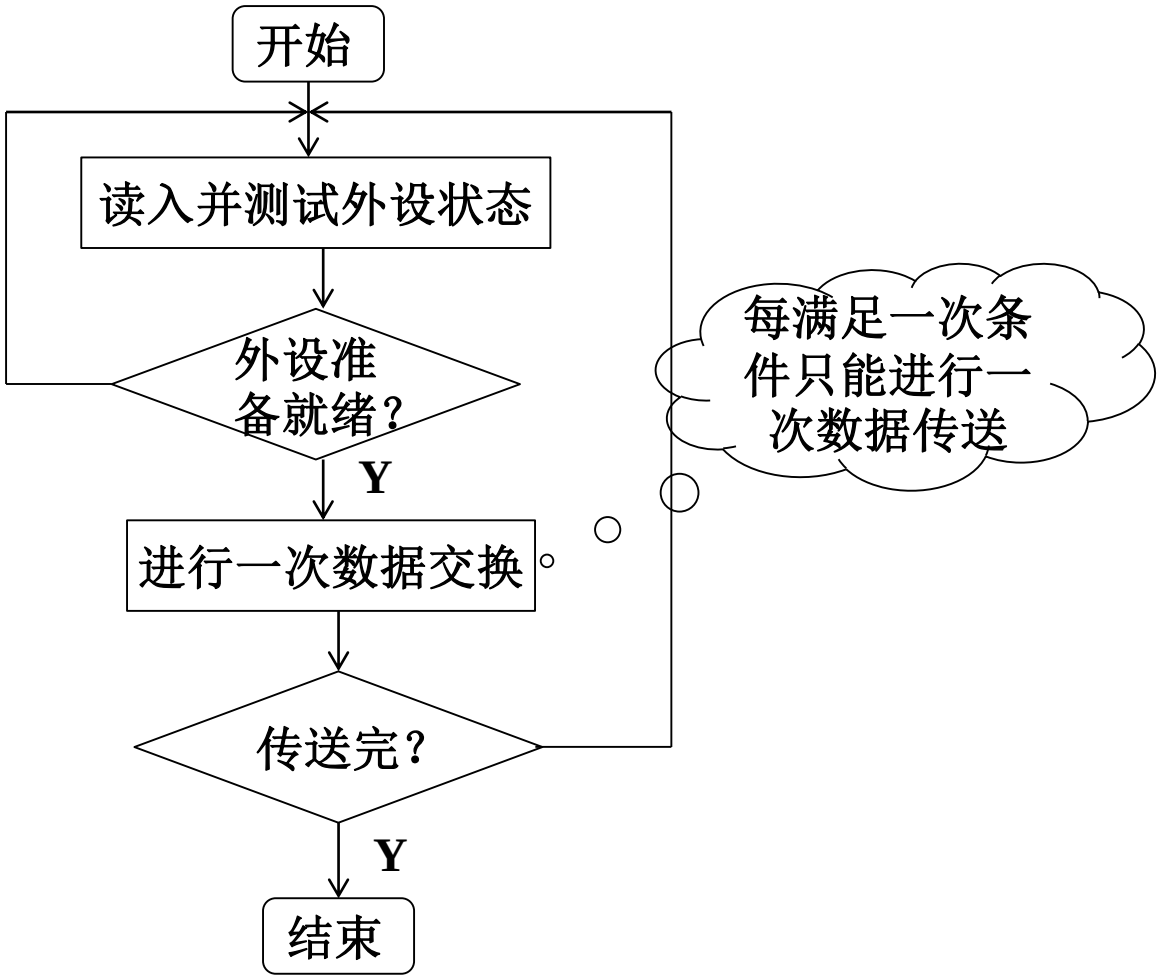
6.3.1 无条件传送方式

- 利用程控方式与外设交换信息时，如果输入/输出的时刻，都可以保证外设总是处于“准备好”状态，则可以直接利用输入/输出指令进行信息的输入/输出操作。
- 要求外设总是处于准备好状态
- 优点：软件及接口硬件简单
- 缺点：只适用于简单外设，适应范围较窄
- 例子
  - 读取开关的状态
  - 当开关闭合时，输出编码使发光二极管亮

6.3.2 查询方式

- 通过程序查询相应设备的状态，若状态不符合，则CPU不能进行输入/输出操作，需要等待；只有当状态信号符合要求时，CPU才能进行相应的输入/输出操作。
- 一般外设均可以提供一些反映其状态的信号，如对输入设备来说，它能够提供“准备好”ready信号，ready=1表示输入数据已准备好。输出设备则提供“忙”busy信号，busy=1表示当前时刻不能接收CPU来的数据，只有当busy=0时，才表明它可以接收来自于CPU的输出数据。
- 仅当条件满足时才能进行数据传送
- 每满足一次条件只能进行一次数据传送
- 适用场合：
  - 外设并不总是准备好
  - 对传送速率和效率要求不高
- 工作条件：
  - 外设应提供设备状态信息
  - 接口应具备状态端口

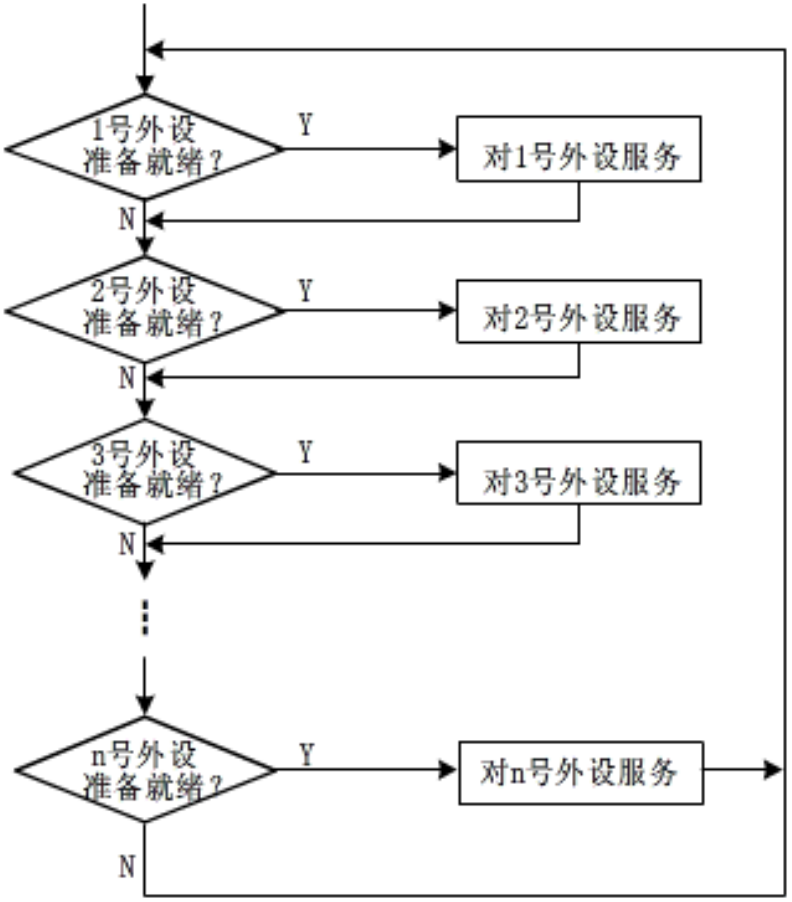
- 查询方式流程图
- 对ready的状态查询，是通过读状态端口的相应位来实现的，输出的情况亦大致相同，这种传送控制方式的最大优点是，能够保证输入/输出数据的正确性。



- 举例：
  - 将48000H地址中的顺序100个单元的数据发送到外设中，利用查询的方式，数据输出和查询状态共用一个地址00FFH。
  - 程序如下：

```
START:  MOV AX,  4000H
        MOV  DS,   AX
        MOV  SI,  8000H
        MOV  CX,  100
GOON:   MOV  DX,   00FFH
WAIT:   IN  AL,  DX
        AND  AL,  01H
        JZ  WAIT
        MOV  AL,   [SI]
        OUT  DX,  AL
        INC  SI
        LOOP GOON
        RET
```

- 优点：
  - 软硬件比较简单
- 缺点：
  - CPU效率低，数据传送的实时性差，速度较慢
- 使用条件
  - （1）连接到系统的外设是简单的、慢速的，且对实时性要求不高
  - （2）连接到同一系统的外设，其工作速度是相近的



6.3.3 中断方式

- 程序查询传送方式明显的缺点是CPU利用率低，不能满足实时输入输出的需要，中断方式可以较好地解决这一问题。
- 所谓中断是指程序运行中出现了某种紧急事件，CPU必须中止现正在执行的程序，转去处理该紧急事件（执行一段中断服务程序），并在处理完后返回原运行的程序的过程。完整的中断处理过程包括中断请求、中断判优、中断响应、中断处理和中断返回
- 特点：
  - 外设在需要时向CPU提出请求，CPU再去为它服务。服务结束后或在外设不需要时，CPU可执行自己的程序。
- 优点：
  - CPU效率高，实时性好，速度快
- 缺点
  - 程序编制相对较为复杂

6.3.4 直接存储器存取方式

- 又称为DMA方式，DMA方式是一种由专门的硬件电路执行I/O交换的传送方式，它让外设接口与内存直接进行告诉的数据交换，而不必经过CPU，从而实现对存储器的直接存取，并获得总线控制权，来实现内存与外设或者内存的不同区域之间大量数据的快速传送。这种专门的硬件叫DMA控制器，简称DMAC。
- 特点
  - 外设直接与存储器进行数据交换，CPU不再担当数据传输的中介者
  - 总线由DMA控制其（DMAC）进行控制（CPU要放弃总线控制权），内存/外设的地址和读写控制信号均由DMAC提供。

- **1. DMA控制器的功能**
  - **（1）收到接口发出的DMA请求后，DMA控制器要向CPU发出总线请求信号HOLD，请求CPU放弃总线的控制。**
  - **（2）当CPU响应请求并发出响应信号HLDA后，这时DMA控制器要接管总线的控制权，实现对总线的控制。**
  - **（3）能向地址总线发出内存地址信息，找到相应单元并能够自动修改其地址计数器。**
  - **（4）能向存储器或外设发出读/写命令。**
  - **（5）能决定传送的字节数，并判断DMA传送是否结束。**
  - **（6）在DMA过程结束后，能向CPU发出DMA结束信号，将总线控制权交还给CPU。**

- **2. DMA控制器的工作过程**
  - **（1）外设向DMA控制器发出“DMA传送请求”信号DRQ**
  - **（2）DMA控制器收到请求后，向CPU发出“总线请求”信号HOLD**
  - **（3）CPU在完成当前总线周期后悔立即发出HLDA信号，对HOLD信号进行响应**
  - **（4）DMA控制器收到HLDA信号后，就开始控制总线，并向外设发出DMA响应信号DACK。**
  - **（5）DMA控制器送出地址信号和相应的控制信号，实现外设与内存或内存与内存之间的直接数据传送**
  - **（6）DMA控制器自动修改地址和字节计数器，并据此判断是否需要重复传送操作。**