

微机原理与接口技术实验

(第三版)

信息工程系实验室

袁鹏

2009. 3. 5

目录

一	显示程序实验	3
二	数据传送实验	5
三	数码转换程序实验	6
四	运算类程序实验	8
五	分支程序设计实验	9
六	循环程序设计实验	11
七	基本 I/O 接口电路设计实验	12
八	地址译码电路设计实验	15
九	点阵 LED 显示设计实验	17
十	8259 中断控制器应用实验	19
十一	8255 并口控制器应用实验	27
十二	A/D 转换实验	32
十三	D/A 转换实验	35
十四	电子发声设计实验	38
十五	键盘扫描及显示设计实验	43

一 显示程序实验

1、实验目的

- (1) 掌握在PC机上以十六进制形式显示数据的方法。
- (2) 掌握部分DOS功能调用使用方法。
- (3) 熟悉 Tddebug 调试环境和 Turbo Debugger 的使用。

2、实验内容及说明

一般来说，有很多程序需要显示输出提示运行的状况和结果，有的还需要将数据区中的内容显示在屏幕上。本实验要求将指定数据区的数据以十六进制数形式显示在屏幕上，并通过DOS功能调用完成一些提示信息的显示。实验中可使用 DOS功能调用（INT 21H）。

(1) 显示单个字符输出

入口： AH=02H

调用参数： DL=输出字符

(2) 显示字符串

入口： AH=09H

调用参数： DS:DX=串地址，' '\$' 为结束字符

(3) 键盘输入并回显

入口： AH=01H

返回参数： AL=输出字符

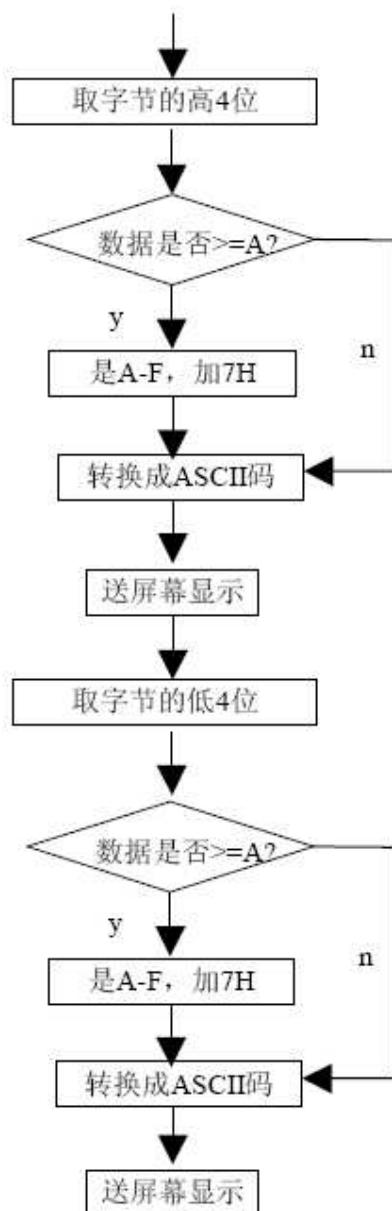
(4) 返回DOS系统

入口： AH=4CH

调用参数： AL=返回码

3、实验步骤

- (1) 运行Tddebug软件，选择Edit菜单，根据实验内容的描述编写实验程序
- (2) 使用Compile菜单中的Compile和Link对实验程序进行汇编、连接。
- (3) 使用Rmrn菜单中的Run，运行程序，观察运行结果。
- (4) 使用Rmrn菜单中的Debug，调试程序，观察调试过程中，数据传输指令执行后，各寄存器及数据区的内容。
- (5) 更改数据区中的数据，考察程序的正确性。



程序流程图

二 数据传送实验

1、实验目的

- (1) 熟悉Tddebug调试环境和Turbo Debugger的使用。
- (2) 掌握与数据有关的不同寻址方式。

2、实验内容及说明

本实验要求将数据段中的一个字符串传送到附加段中，并输出附加段中的目标字符串到屏幕上。

3、实验步骤

- (1) 运行Tddebug软件，选择Edit菜单编写实验程序。
- (2) 使用Compile菜单中的Compile和Link对实验程序进行汇编、连接。
- (3) 使用Rmrun菜单中的Run，运行程序，观察运行结果。
- (4) 使用Rmrun菜单中的Debug，调试程序，观察调试过程中，数据传输指令执行后，各寄存器及数据区的内容。
- (5) 更改数据区中的数据，考察程序的正确性。

三 数码转换程序实验

1、实验目的

掌握不同进制数及编码相互转换的程序设计方法。

2、实验内容及说明

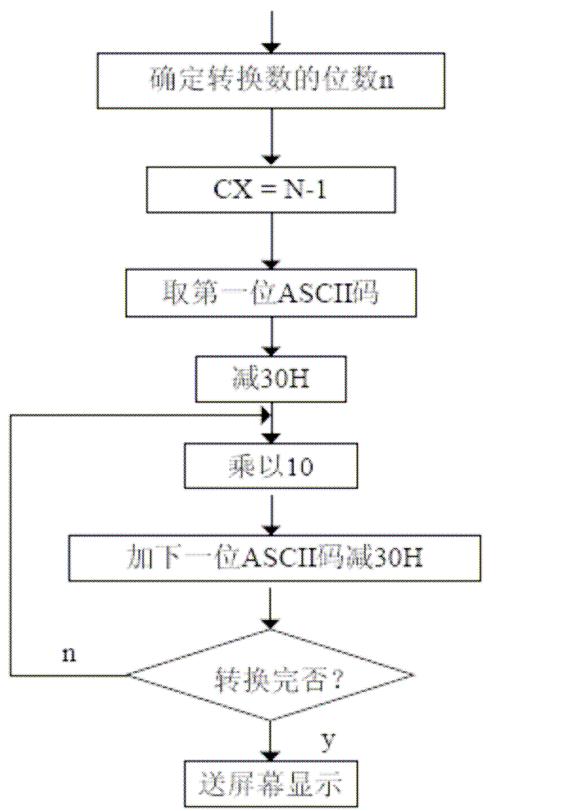
计算机输入设备输入的信息一般是由 ASCII 码或 BCD 码表示的数据或字符，CPU 一般均用二进制数进行计算或其他信息处理，处理结果的输出又必须依照外设的要求变为 ASCII 码、BCD 码或七段显示码等。因此，在应用软件中，各类数制的转换和代码的转换是必不可少的。

1. 将ASCII码表示的十进制数转换为二进制数

十进制数可以表示为： $D_n \times 10^n + D_{n-1} \times 10^{n-1} + \dots + D_0 \times 10^0 = D_i \times 10^i$ 其中 D_i 代表十进制数 1、2、3…9、0。

上式可以转换为： $\sum D_i \times 10^i = ((\dots(D_n \times 10 + D_{n-1}) \times 10) + D_{n-2}) \times 10 + \dots + D_1 \times 10 + D_0$

由上式可归纳十进制数转换为二进制的方法：从十进制数的最高位 D_n 开始作乘10加次位的操作，依次类推，则可求出二进制数结果。本实验要求将缓冲区中的一个五位十进制数00012的ASCII码转换成二进制数，并将转换结果按位显示在屏幕上。转换过程的参考流程如图所示



3、实验步骤

- (1) 运行Tddebug软件，选择Edit菜单，根据实验内容描述的数码转换内容编写实验程序。
- (2) 利用Compile菜单中的Compile和Link对实验程序进行汇编、连接。
- (3) 利用Rmrn菜单中的Run，运行程序，观察运行结果。
- (4) 使用Rmrn菜单中的Debug，调试程序，观察调试过程中，数据传输指令执行后，各寄存器及数据区的内容。
- (5) 更改数据区中的数据，考察程序的正确性。

四 运算类程序实验

1、实验目的

- (1) 掌握运算类指令编程及调试方法。
- (2) 掌握运算类指令对各状态标志位的影响及测试方法。

2、实验内容及说明

1. 二进制双精度加法运算

本实验要求计算 $X+Y=Z$ ，将结果 Z 输出到屏幕，其中 $X=001565A0H$ ， $Y=0021B79EH$ 。实验利用累加器AX，先求低十六位和，并存入低址存储单元，后求高16位和，再存入高址存储单元。由于低位和可能向高位有进位，因而高位字相加语句需用ADC指令，则低位相加有进位时，CF=1，高位字相加时，同时加上CF中的1。在80386以上微机中可以直接使用32位寄存器和32位加法指令完成本实验的功能。

3、实验步骤

- (1) 运行Tddebug软件，选择Edit菜单编写实验程序。
- (2) 使用Compile菜单中的Compile和Link对实验程序进行汇编、连接。
- (3) 使用Rmrnrun菜单中的Run，运行程序，观察运行结果。
- (4) 使用Rmrnrun菜单中的Debug，调试程序，观察调试过程中，数据传输指令执行后，各寄存器及数据区的内容。
- (5) 更改数据区中的数据，考察程序的正确性。

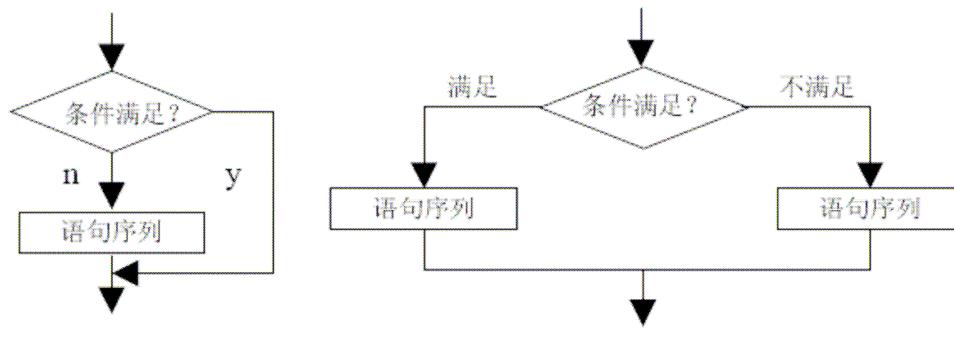
五 分支程序设计实验

1、实验目的

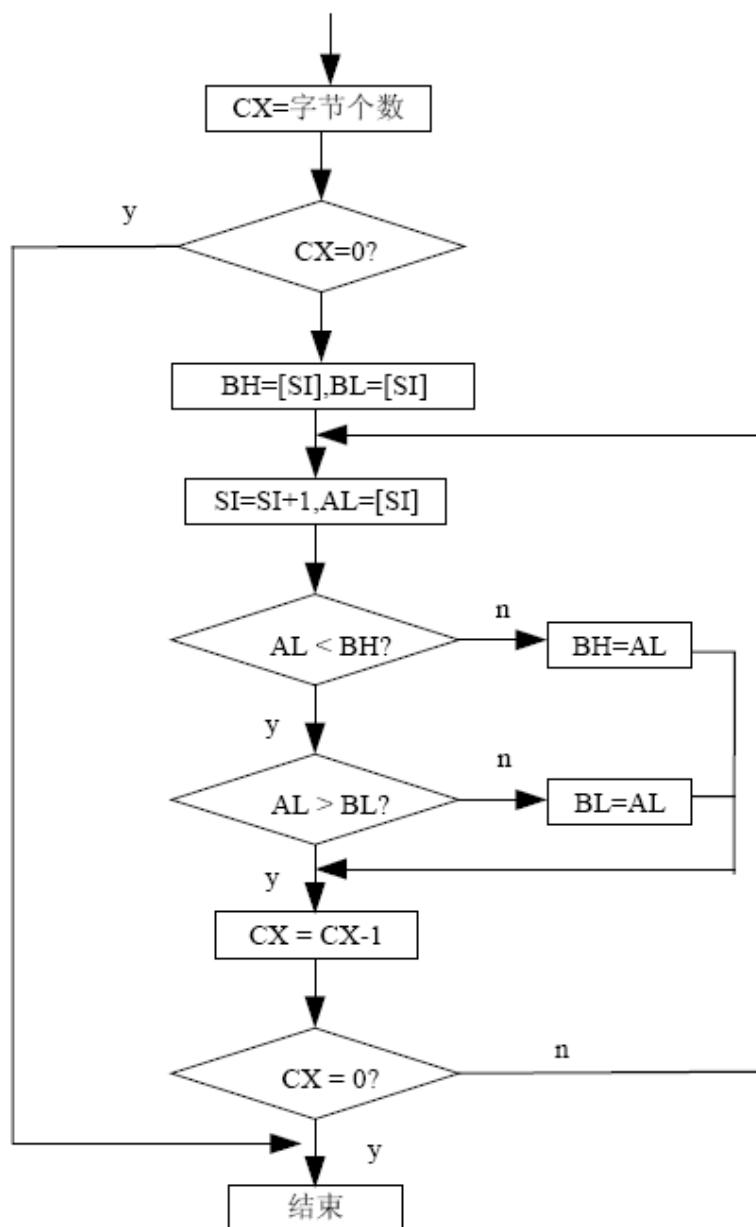
掌握分支程序的设计方法。

2、实验内容及说明

程序有顺序、循环、分支和子程序四种结构形式，分支结构的示意图如图所示。本实验要求通过求无符号字节序列中的最大值和最小值来反映分支程序的结构形式。



实验可以使用BH, BL作为暂存现行的最大值和最小值，且在程序的初始，将BH和BL初始化为首字节的内容，然后进入循环操作。在循环操作中，依次从字节序列中逐个取出一个字节的内容与BH, BL进行比较，若取出的字节内容比BH的内容大或比BL中的内容小，则修改之。当循环结束操作时，将BH, BL分别送屏幕显示。参考实验流程如图



3、实验步骤

- (1) 运行Tddebug软件，选择Edit菜单编写实验程序。
- (2) 使用Compile菜单中的Compile和Link对实验程序进行汇编、连接。
- (3) 使用Rmrn菜单中的Run，运行程序，观察运行结果。
- (4) 更改数据区中的数据，考察程序的正确性。

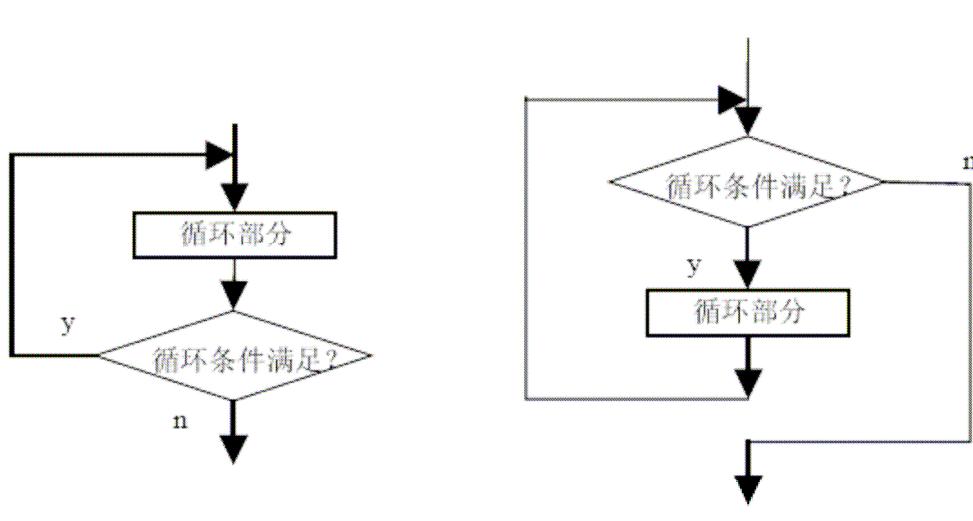
六 循环程序设计实验

1、实验目的

掌握循环程序的设计方法。

2、实验内容及说明

本实验要求通过求某数据区内负数的个数来表现循环程序的结构形式。要求实验程序在数据区中存放一组数据，为统计负数的个数，逐个判断区内的数据，然后将所有数据中凡是符号位为1的数据的个数累加起来，即得到区内所包含负数的个数。循环程序的结构示意如图所示。



3、实验步骤

- (1) 运行Tddebug软件，选择Edit菜单编写实验程序。提供10字节的数据：12H, 88H, 82H, 89H, 33H, 90H, 01H, 10H, 0BDH, 01H。
- (2) 使用Compile菜单中的Compile和Link对实验程序进行汇编、连接。
- (3) 使用Rmrun菜单中的Run，运行程序，观察运行结果。
- (4) 更改数据区中的数据，考察程序的正确性。

七 基本 I/O 接口电路设计实验

1、实验目的

- (1) 掌握基本I/O接口电路的设计方法。
- (2) 熟练汇编语言I/O端口操作指令的使用。

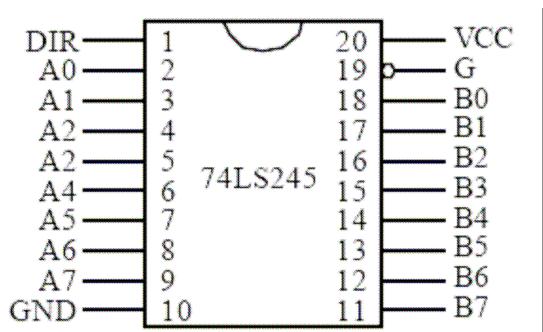
2、实验内容及说明

利用三态缓冲器74LS245、锁存器74LS374设计微机总线和外部设备的数据通道，实现微机对外部输入数据的读取和对输出数据的输出。用开关及LED显示单元的开关和数据灯作为输入和输出显示设备，将读到开关的数据显示在数据灯上。

3、实验原理

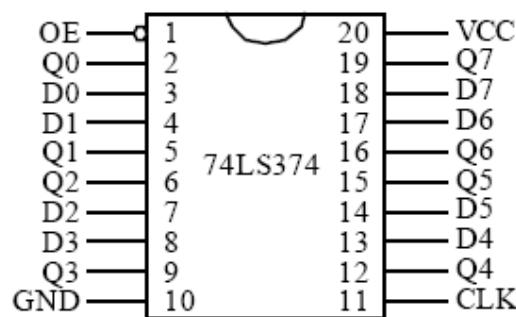
1. 输入接口设计

输入接口一般用三态缓冲器实现，外部设备输入数据通过三态缓冲器，通过数据总线传送给微机系统。74LS245是一种8通道双向的三态缓冲器，其管脚结构如图所示。DIR引脚控制缓冲器数据方向，DIR为1表示数据由A[7:0]至B[7:0]，DIR为0表示数据由B[7:0]至A[7:0]。G引脚为缓冲器的片选信号，低电平有效。



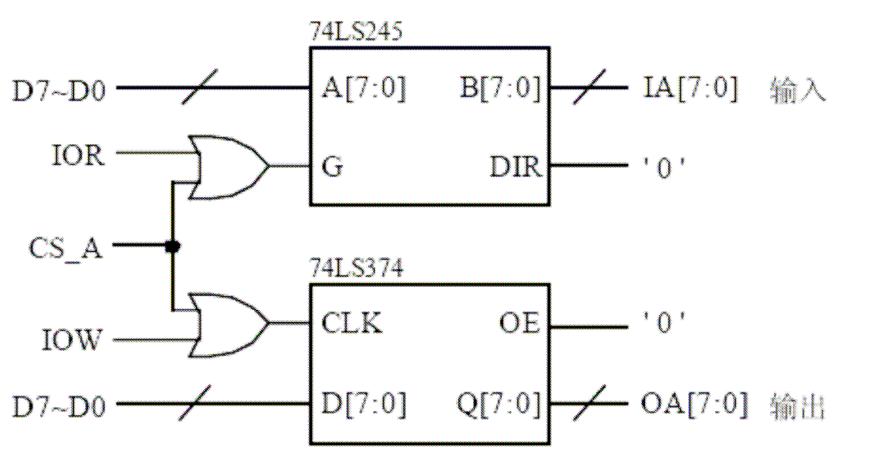
2. 输出接口设计

输出接口一般用锁存器实现，从总线送出的数据可以暂存在锁存器中。74LS374是一种8通道上沿触发锁存器。其管脚结构如图所示。D[7:0]为输入数据线，Q[7:0]为输出数据线。CLK引脚为锁存控制信号，上升沿有效。当上升沿到来时，输出数据线锁存输入数据线上的数据。OE引脚为锁存器的片选信号，低电平有效。



3. 输入输出接口设计

用74LS245和74LS374可以组成一个输入输出接口电路，既实现数据的输入又实现数据的输出，输入输出可以占用同一个端口。是输入还是输出用总线读写信号来区分。总线读信号IOR和片选信号CS相“或”来控制输入接口74LS245的使能信号G。总线写信号IOW和片选信号CS相“或”来控制输出接口74LS374的锁存信号CLK。实验系统中基本输入输出单元就实现了两组这种的电路，任意A组的电路连接如图所示。



4、实验说明及步骤

本实验实现的是将开关K[7:0]的数据通过输入数据通道读入CPU的寄存器，然后再通过输出数据通道将该数据输出到数据灯显示，该程序循环运行，直到按动键

盘上任意按键再退出程序。实验程序流程如图3-1-4所示。参考实验接线如图3-1-5所示。

实验步骤如下。

- (1) 确认从PC机引出的两根扁平电缆已经连接在实验平台上。
- (2) 参考图3-1-5所示连接实验线路。
- (3) 首先运行CHECK程序，查看并记录与片选信号对应的I/O端口始地址。
- (4) 参考实验流程图编写程序，注意使用正确的端口地址，然后编译链接。
- (5) 运行程序，拨动开关，观看数据灯

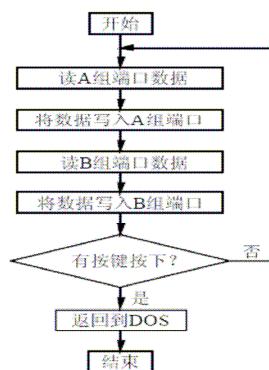


图 3-1-4 基本 I/O 接口设计实验参考程序流程图

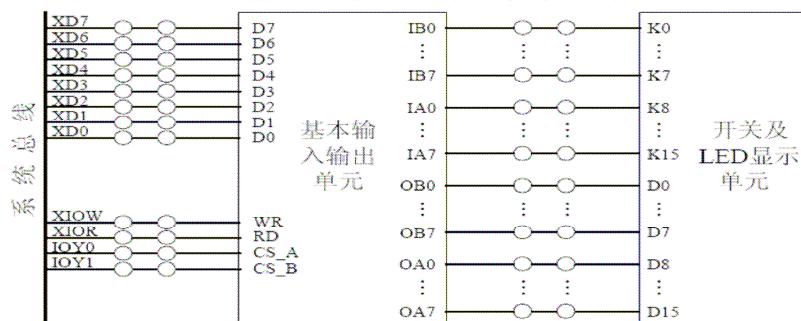


图 3-1-5 基本 I/O 接口设计实验参考接线图

八 地址译码电路设计实验

1、实验目的

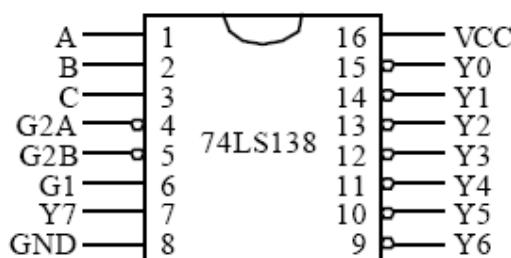
- (1) 学习3-8译码器在接口电路中的应用。
- (2) 掌握地址译码电路的一般设计方法。

2、实验内容及说明

用74LS138译码器设计地址译码电路，并用其输出作为基本输入输出单元的片选信号，使用设计的端口地址编写程序，实现数据的输入输出。

3、实验原理

微机接口电路中，常采用74LS138译码器来实现I/O端口或存储器的地址译码。74LS138有3个输入引脚、3个控制引脚及8个输出引脚，其管脚信号如图所示。当3个控制信号有效时，相应于输入信号A、B、C状态的那个输出端为低电平，该信号即可作为片选信号。



32位扩展系统总线上有一个M/I/O信号，该信号为低电平时指示当前操作为I/O操作，为高电平时指示当前操作为存储器操作，它和译码器不同的连接可以用来区分是I/O端口译码还是存储器端口译码。32位总线地址是由A2开始，所以地址是以4字节边界对齐的。

实验系统的I/O地址空间共有256字节，偏移地址一般从00H~FFH。起始地址由PC机系统分配，可以用CHECK程序读出。所以设计地址译码电路，主要是针对

低8位地址线译码，得到偏移在00H~FFH之间的端口。本实验要求不使用总线上的片选信号，自行设计端口偏移地址分别为COH~DFH和E0H~FFH的译码电路，然后用译码输出作为基本输入输出单元的片选。编写程序，完成I/O数据操作。实验参考线路如图3-2-2所示。

4、实验说明及步骤

- (1) 确认从PC机引出的两根扁平电缆已经连接在实验平台上。
- (2) 按图3-2-2所示连接实验线路。
- (3) 首先运行CHECK程序，查看I/O端口始地址。
- (4) 利用设计好的端口地址编写程序，然后编译链接。
- (5) 运行程序，拨动开关，观看数据灯显示是否正确。

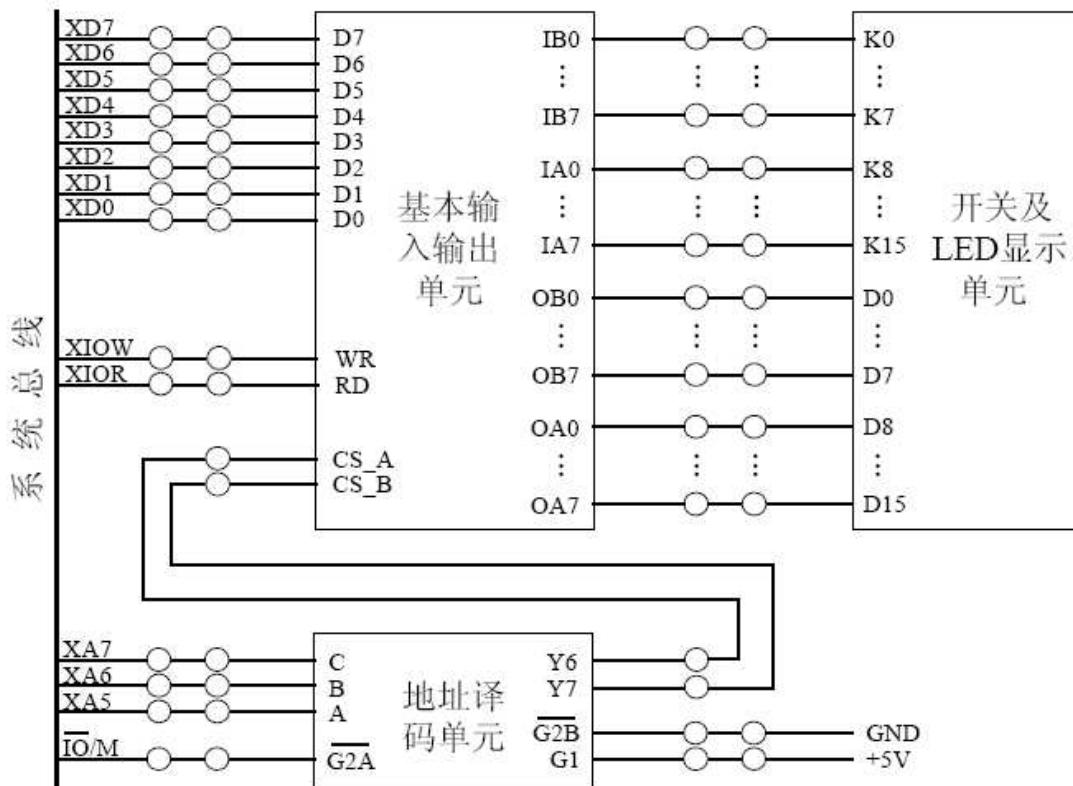


图 3-2-2 地址译码设计实验参考接线图

九 点阵 LED 显示设计实验

1、实验目的

学习点阵LED的操作方法。

2、实验内容及说明

使用基本输入输出单元中的两路输出 $OA[7:0]$ 和 $OB[7:0]$ 分别控制点阵 LED 的行控制 $R[7:0]$ 和列控制 $L[7:0]$ 。编写程序，使点阵 LED 的每一行和每一列依次循环显示。

3、实验说明及步骤

8×8 点阵 LED 相当于 8×8 个发光管组成的阵列，其中每一行共用一个阳极（行控制），每一列共用一个阴极（列控制）。行控制和列控制满足正确的电平就可使相应行列的发光管点亮。实验平台上点阵 LED 的管脚及相应的行、列控制位如图 1 所示。

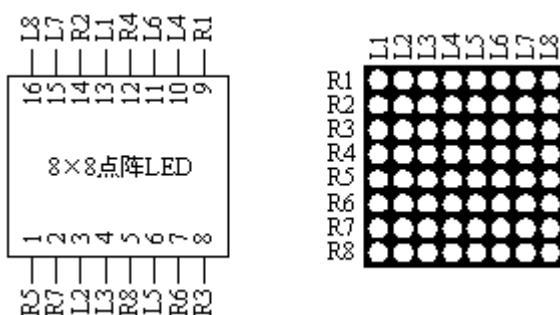


图 1

本实验利用基本输入输出单元的两路输出分别控制点阵 LED 的行和列，编写一个简单的程序，使每一行和每一列依次循环显示。参考程序流程如图 2 所示。实验参考接线如图 3 所示。

实验步骤如下：

- (1) 确认从 PC 机引出的两根扁平电缆已经连接在实验平台上。
- (2) 首先运行 CHECK 程序，查看 I/O 空间始地址。
- (3) 利用查出的地址编写程序，然后编译链接。
- (4) 参考图 3 所示连接实验线路。
- (5) 运行程序，观察 LED 显示是否正确。

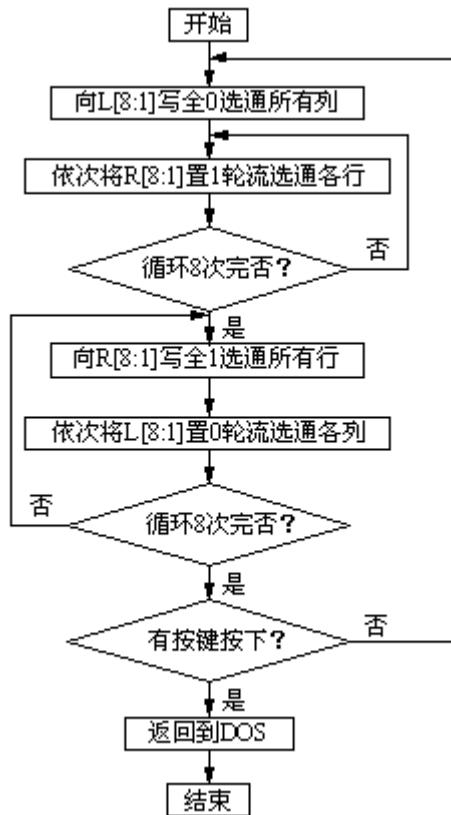


图 2

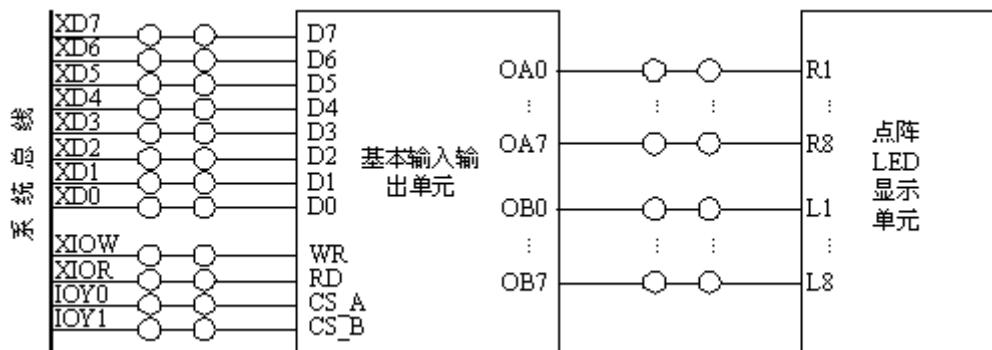


图 3

十 8259 中断控制器应用实验

1、实验目的

- (1) 学习中断控制器8259的工作原理。
- (2) 掌握可编程控制器8259的应用编程方法。

2、实验内容及说明

- (1) 利用PC机给实验系统分配的中断线，设计一个单中断应用。使用单次脉冲模拟中断产生。编写中断处理程序，在显示器屏幕上显示一个字符。
- (2) 利用实验平台上8259控制器对扩展系统总线上的中断线INTR进行扩展。编写程序对8259控制器的IR0和IR1中断请求进行处理。

3、实验原理

1. 8259控制器的介绍

中断控制器8259A是Intel公司专为控制优先级中断而设计开发的芯片。它将中断源优先级排队、辨别中断源以及提供中断矢量的电路集于一片中，因此无需附加任何电路，只需对8259A进行编程，就可以管理8级中断，并选择优先模式和中断请求方式，即中断结构可以由用户编程来设定。同时，在不需增加其他电路的情况下，通过多片8259A的级连，能构成多达64级的矢量中断系统。它的管理功能包括：1) 记录各级中断源请求，2) 判别优先级，确定是否响应和响应哪一级中断，3) 响应中断时，向CPU传送中断类型号。8259A的内部结构和引脚如图3-5-1所示。

8259A的命令共有7个，一类是初始化命令字，另一类是操作命令。8259A的编程就是根据应用需要将初始化命令字ICW1-ICW4和操作命令字OCW1- OCW3分别写入初始化命令寄存器组和操作命令寄存器组。ICW1-ICW4各命令字格式如图3-5-2所示，OCW1-OCW3各命令字格式如图3-5-3所示，其中OCW1用于设置中断屏

蔽操作字，OCW2用于设置优先级循环方式和中断结束方式的操作命令字，OCW3用于设置和撤销特殊屏蔽方式、设置中断查询方式以及设置对8259内部寄存器的读出命令。

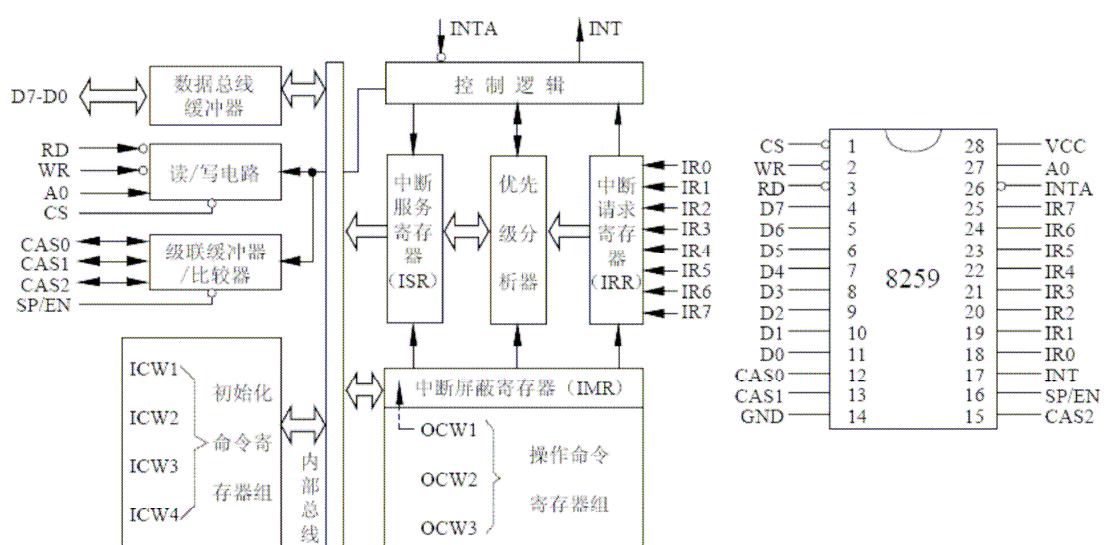


图 3-5-1 8259 内部结构和引脚图

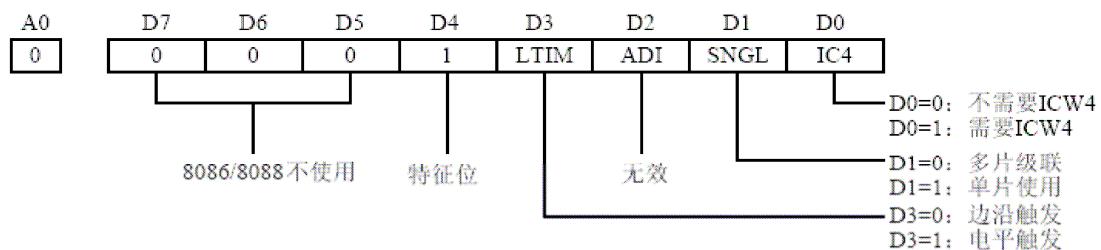


图 3-5-2 (a) ICW1 格式



图 3-5-2 (b) ICW2 格式

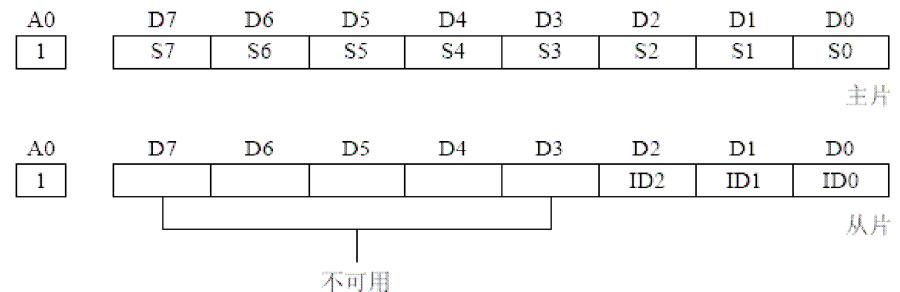


图 3-5-2 (c) ICW3 格式

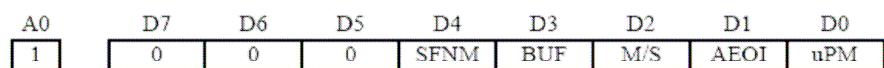


图 3-5-2 (d) ICW4 格式

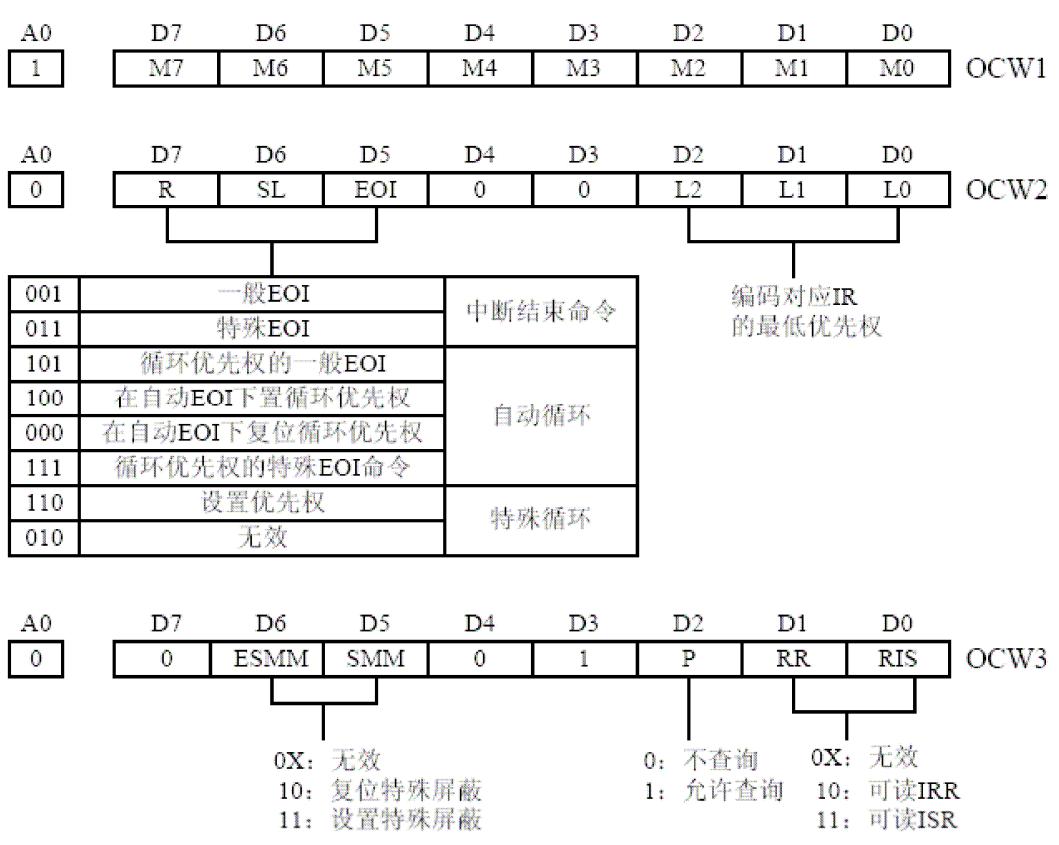


图 3-5-3 OCW 命令字格式

2. 8259寄存器及命令的控制访问

在硬件系统中，8259仅占用两个外设接口地址，在片选有效的情况下，利用 A_0 来寻址不同的寄存器和命令字。对寄存器和命令的访问控制如表3-5-1所示。

表 3-5-1 8259 寄存器及命令的访问控制

A0	D4	D3	读信号	写信号	片选	操作
0			0	1	0	读出 ISR.IRR 的内容
1			0	1	0	读出 IMR 的内容
0	0	0	1	0	0	写入 OCW2
0	0	1	1	0	0	写入 OCW3
0	1	×	1	0	0	写入 ICW1
1	×	×	1	0	0	写入 OCW1, ICW2, ICW3, ICW4

3. PC微机系统中的8259

在80x86系列PC微机系统中，系统中包含了两片8259A中断控制器，经级连可以管理15级硬件中断，但其中部分中断号已经被系统硬件占用，具体使用情况如表3-5-2示。两片8259A的端口地址为：主片8259使用020H和021H两个端口；从片

使用0A0H和0A1H两个端口。系统初始化两片8259的中断请求信号均采用上升沿触发，采用全嵌套方式，优先级的排列次序为0级最高，依次为1级、8级～15级，然后是3级～7级。

在扩展系统总线上的INTR对应的中断线就是PC机保留中断其中的一个。对INTR中断的初始化PC机已经完成，在使用时主要是将其中断屏蔽打开，修改中断向量。

表 3-5-2 PC 微机系统中的硬件中断

中断号	功能	中断向量号	中断向量地址
主 8259A IRQ0	日时钟/计数器 0	08H	0020H～0023H
主 8259A IRQ1	键盘	09H	0024H～0027H
主 8259A IRQ2	接从片 8259A	0AH	0028H～002BH
主 8259A IRQ3	串行口 2	0BH	002CH～002FH
主 8259A IRQ4	串行口 1	0CH	0030H～0033H
主 8259A IRQ5	并行口 2	0DH	0034H～0037H
主 8259A IRQ6	软盘	0EH	0038H～003BH
主 8259A IRQ7	并行口 1	0FH	003CH～003FH
从 8259A IRQ8	实时钟	70H	01C0H～01C3H
从 8259A IRQ9	保留	71H	01C4H～01C7H
从 8259A IRQ10	保留	72H	01C8H～01CBH
从 8259A IRQ11	保留	73H	01CCH～01CFH
从 8259A IRQ12	保留	74H	01D0H～01D3H
从 8259A IRQ13	协处理器中断	75H	01D4H～01D7H
从 8259A IRQ14	硬盘控制器	76H	01D8H～01DBH
从 8259A IRQ15	保留	77H	01DCH～01DFH

4、实验说明及步骤

1. 单中断应用实验

本实验要求使用总线上INTR中断请求线完成一次单中断应用实验。中断处理程序完成在屏幕上的显示字符“9”。实验前先运行CHECK程序，得到INTR所对应的中断号、相应的初始化命令字寄存器ICW及操作命令字寄存器OCW的地址、开屏蔽的命令字、中断矢量地址和PCI卡中断控制寄存器INTCSR的地址。得到这些信

息后就可以开始设计实验了。参考程序流程如图3-5-5所示。

实验步骤如下。

- (1) 确认从PC机引出的两根扁平电缆已经连接在实验平台上。
- (2) 首先运行CHECK程序，查看INTR中断号及相关信息。
- (3) 参考实验流程图编写程序，然后编译链接。
- (4) 将单次脉冲单元的KK1+连接到系统总线上的INTR。
- (5) 运行程序，按动KK1+按键，观察中断是否产生。

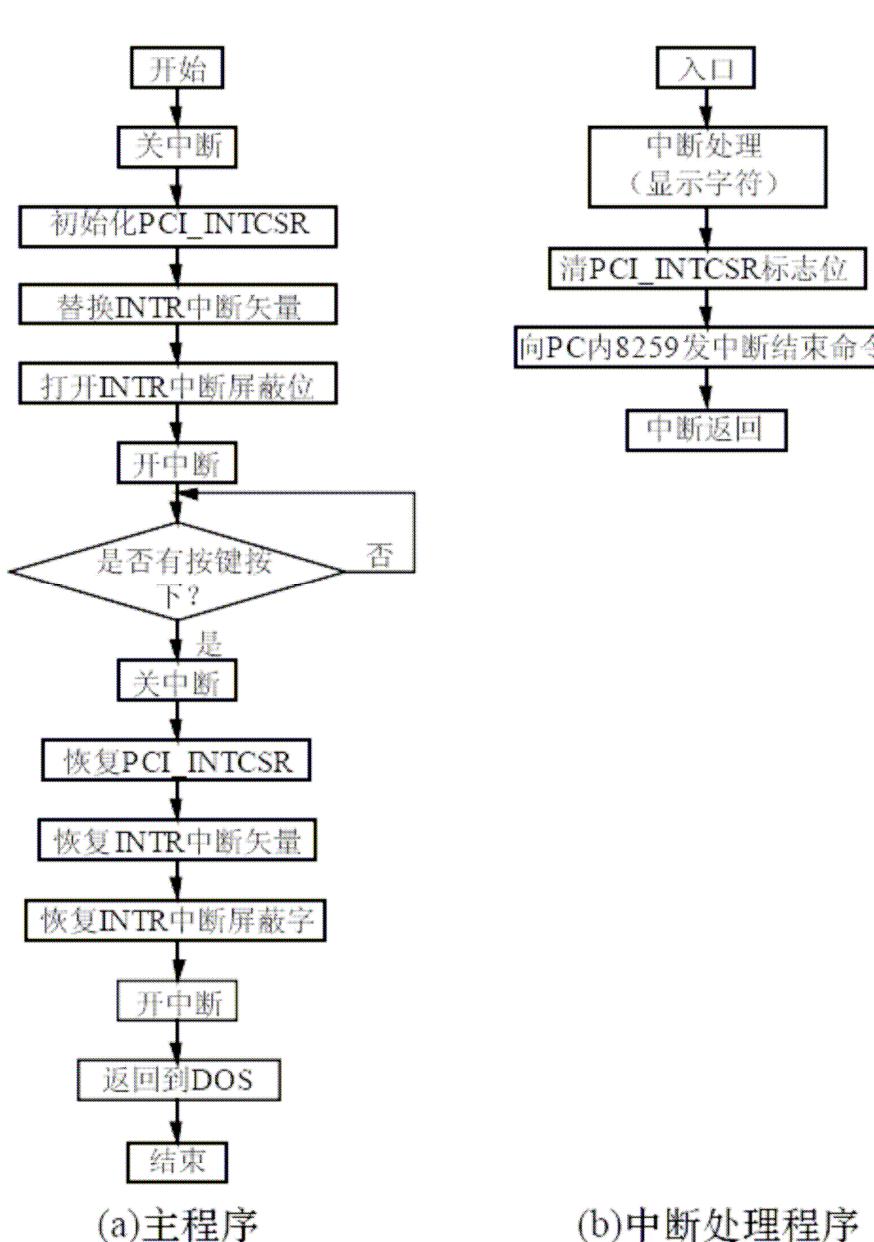


图 3-5-5 8259 中断应用实验(1)参考程序流程图

2. 扩展多中断源实验

利用实验平台上的8259控制器，可以对总线上的INTR进行扩展。将8259的INT连接到INTR，8259的8路中断请求线IR0～IR7就成了单一INTR中断请求线的扩充。这8路中断源共用INTR的中断矢量，共用INTR的中断服务线程。在INTR的中断服务线程中通过对8259 OCW3的查询，以确定是IR0～IR7中哪个产生中断，然后转到相应的服务线程进行处理。将8259的OCW3中P位置1即可执行查询，查询字格式如图3-5-6所示。



图 3-5-6 8259 OCW3 查询命令字格式

本实验要求实现8259控制器IR0、IR1两路中断。用KK1+和KK2+模拟两个中断源，在IR0对应的服务程序中显示字符“0”，在IR1对应的服务程序中显示字符“1”。实验程序参考流程如图3-5-7所示。参考实验电路如图3-5-8所示。

实验步骤如下：

- (1) 确认从PC机引出的两根扁平电缆已经连接在实验平台上。
- (2) 首先运行CHECK程序，查看INTR中断号及相关信息。
- (3) 参考实验流程图编写程序，然后编译链接。
- (4) 参考实验接线图连接实验电路。
- (5) 运行程序，按动KK1+、KK2+按键，观察中断响应是否正常。

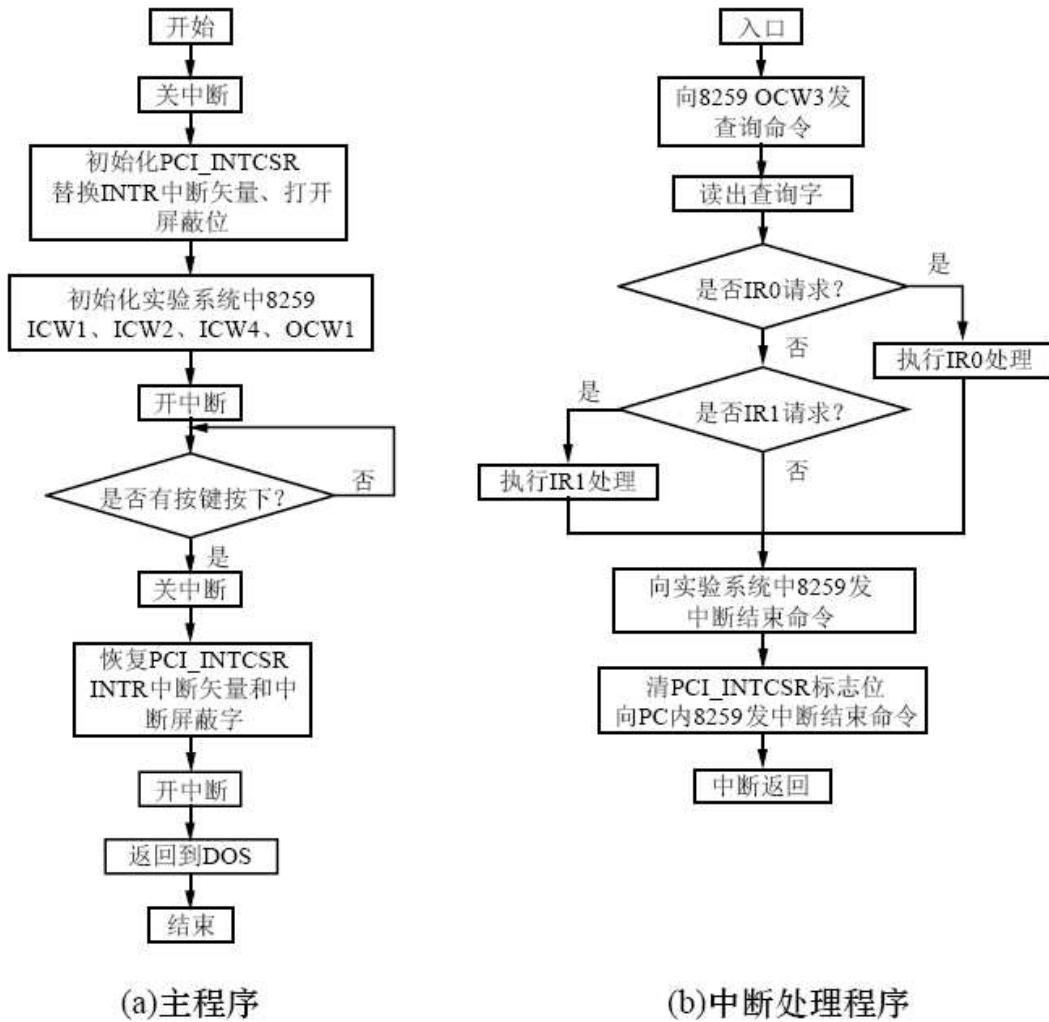


图 3-5-7 8259 中断应用实验(2)参考程序流程图

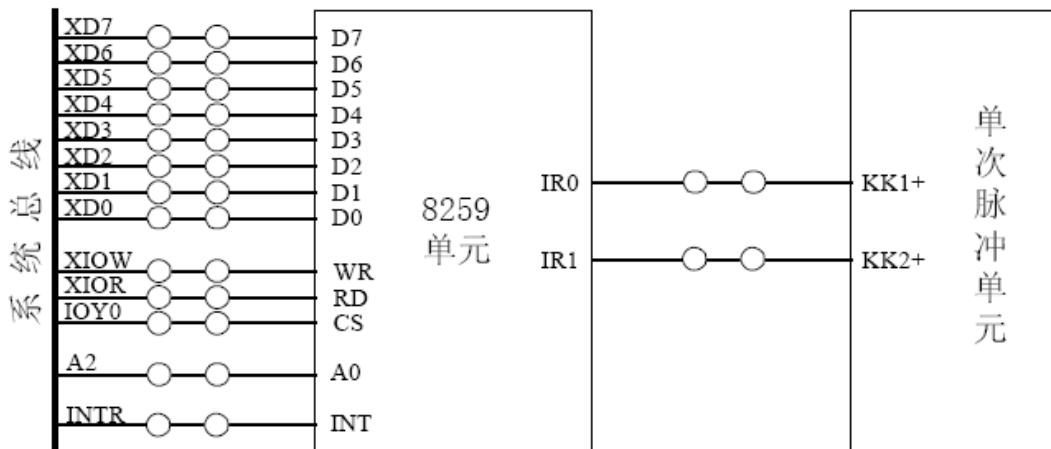


图 3-5-8 8259 中断应用实验(2)参考接线图

十一 8255 并口控制器应用实验

1、实验目的

- (1) 掌握8255的工作方式及应用编程。
- (2) 掌握8255的典型应用电路接法。

2、实验内容及说明

- (1) 基本输入输出实验。编写程序，使8255的A口为输出，B口为输入，完成拨动开关到数据灯的数据传输。要求只要开关拨动，数据灯的显示就改变。
- (2) 流水灯显示实验。编写程序，使8255的A口和B口均为输出，实现16位数据灯的相对循环显示。

3、实验原理

并行接口是以数据的字节为单位与I/O设备或被控制对象之间传递信息。CPU和接口之间的数据传送总是并行的，即可以同时传递8位、16位、32位等。8255可编程外围接口芯片是Intel公司生产的通用并行I/O接口芯片，它具有A、B、C三个并行接口，用+5V单电源供电，能在以下三种方式下工作：方式0—基本输入/出方式、方式1—选通输入/出方式、方式2—双向选通工作方式。8255的内部结构及引脚如图3-7-1所示，8255工作方式控制字和C口按位置位/复位控制字格式如图3-7-2所示。

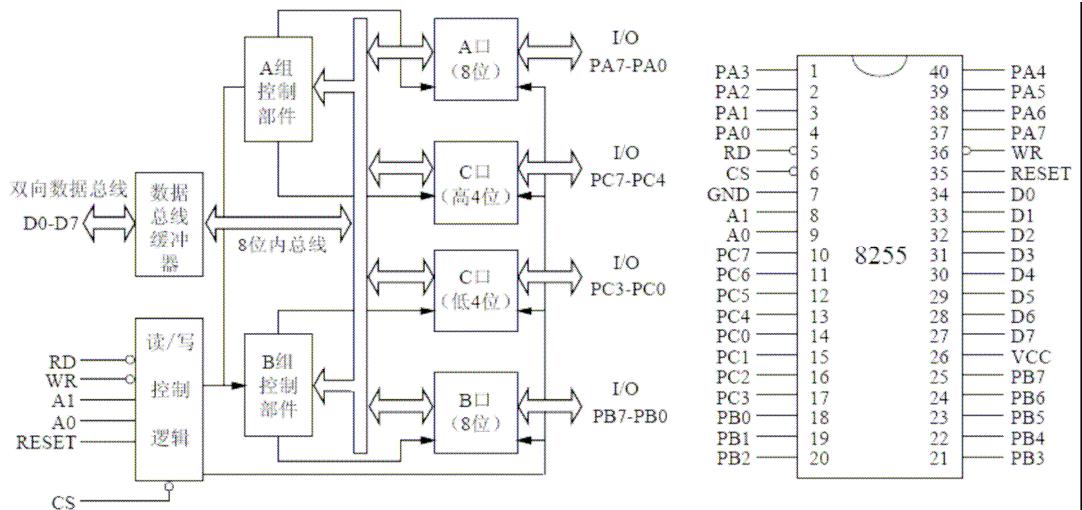


图 3-7-1 8255 的内部结构及引脚

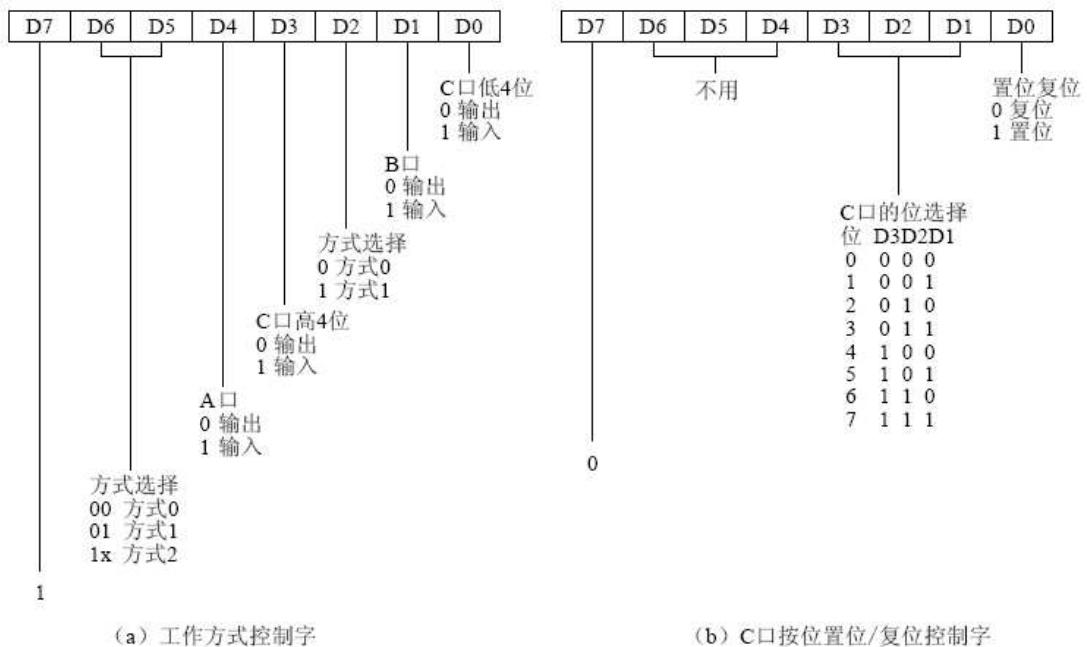


图 3-7-2 8255 控制字格式

4、实验说明及步骤

1. 基本输入输出实验

本实验使8255端口A工作在方式0并作为输出口，端口B工作在方式0并作为输入口。用一组开关信号接入端口B，端口A输出线接至一组数据灯上，然后通过对8255芯片编程来实现输入输出功能。参考程序流程如图3-7-3所示。

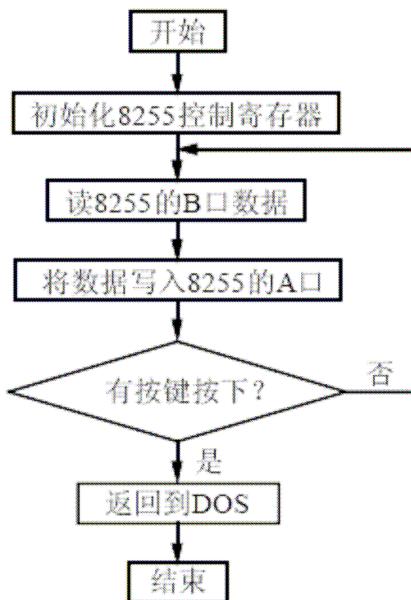


图 3-7-3 8255 并口应用实验(1)参考程序流程图

实验步骤如下。

- (1) 确认从PC机引出的两根扁平电缆已经连接在实验平台上。
- (2) 参考图3-7-4所示连接实验线路。
- (3) 首先运行CHECK程序，查看I/O空间始地址。
- (4) 利用查出的地址编写程序，然后编译链接。
- (5) 运行程序，拨动开关，看数据灯显示是否正确。

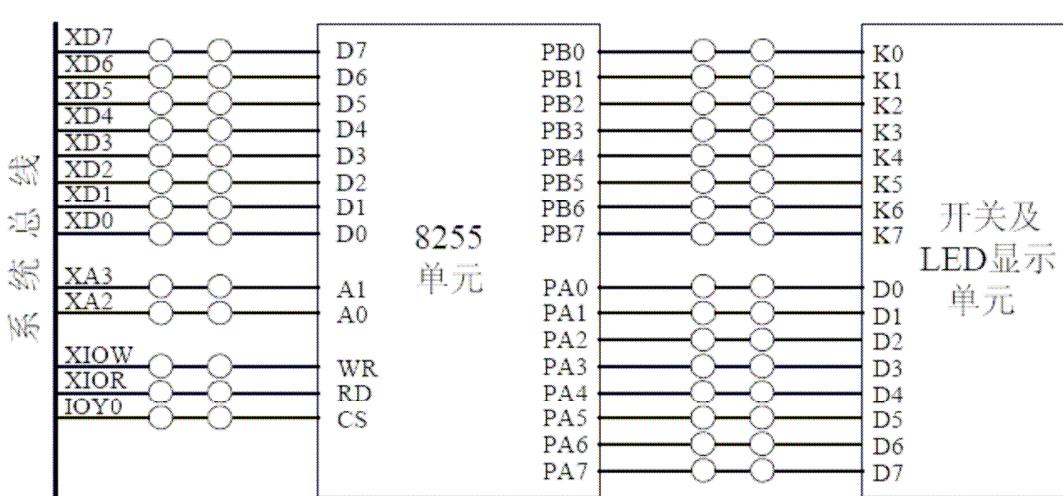


图 3-7-4 8255 并口应用实验(1)参考接线图

2. 流水灯显示实验

首先分别向A口和B口写入80H和01H，然后分别将该数右移和左移一位，再送

到端口上，这样循环下去，从而实现流水灯的显示。参考实验程序流程如图3-7-5所示。

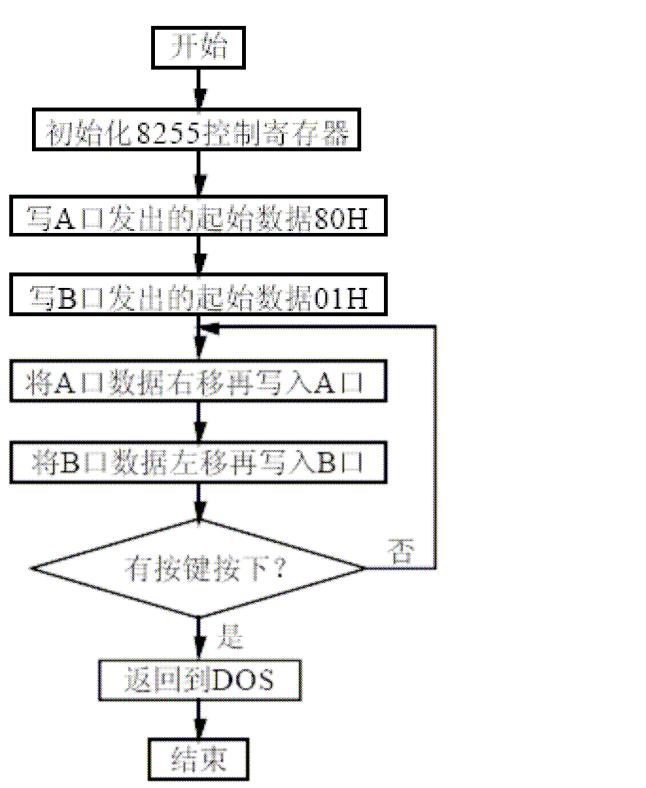


图 3-7-5 8255 并口应用实验(2)参考程序流程图

实验步骤如下：

- (1) 确认从PC机引出的两根扁平电缆已经连接在实验平台上。
- (2) 首先运行CHECK程序，查看I/O空间始地址。
- (3) 利用查出的地址编写程序，然后编译链接。
- (4) 参考图3-7-6所示连接实验线路。
- (5) 运行程序，看数据灯显示是否正确。

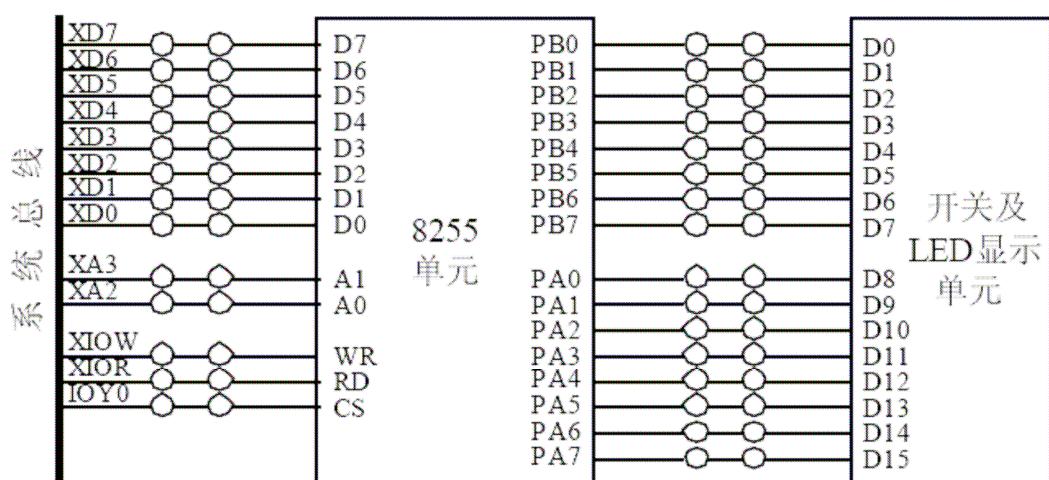


图 3-7-6 8255 并口应用实验(1)参考接线图

十二 A/D 转换实验

1、实验目的

- (1) 学习掌握模/数信号转换基本原理。
- (2) 掌握ADC0809芯片的使用方法。

2、实验内容及说明

编写实验程序，用ADC0809完成模拟信号到数字信号的转换。输入模拟信号由A/D转换单元可调电位器提供的0~5V，输出数字量显示在显示器屏幕上。显示形式为：AD0809：IN0 XX。

3、实验原理

ADC0809包括一个8位的逐次逼近型的ADC部分，并提供一个8通道的模拟多路开关和联合寻址逻辑。用它可直接输入8个单端的模拟信号，分时进行A/D转换，在多点巡回检测、过程控制等应用领域中使用非常广泛。ADC0809的主要技术指标为：

- 分辨率：8位 • 单电源：+5V
- 总的不可调误差：±1LSB • 转换时间：取决于时钟频率
- 模拟输入范围：单极性 0~5V • 时钟频率范围：10KHz~1280KHz

ADC0809的外部管脚如图3-10-1所示，地址信号与选中通道的关系如表3-10-1所示。

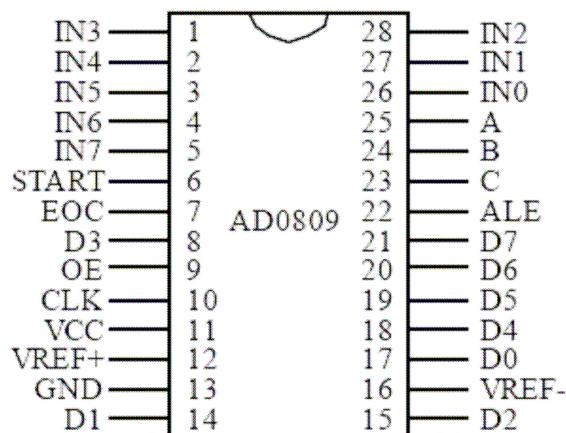


图 3-10-1 ADC0809 管脚图

4、实验说明及步骤

- (1) 确认从PC机引出的两根扁平电缆已经连接在实验平台上。
- (2) 首先运行CHECK程序，查看I/O空间始地址。
- (3) 利用查出的地址编写程序，然后编译链接。
- (4) 参考图3-10-2所示连接实验线路。
- (5) 运行程序，调节电位器，观察屏幕上显示的数字量输出。

表 3-10-1 地址信号与选中通道的关系

地 址			选中通道
C	B	A	
0	0	0	IN0
0	0	1	IN1
0	1	0	IN2
0	1	1	IN3
1	0	0	IN4
1	0	1	IN5
1	1	0	IN6
1	1	1	IN7

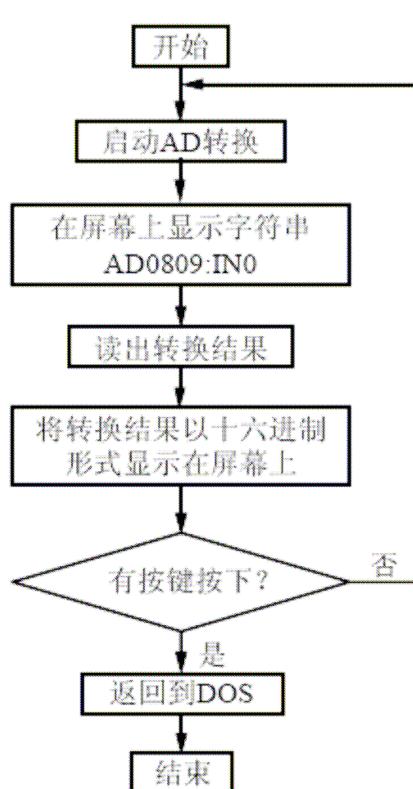


图 3-10-2 A/D 转换实验参考程序流程图

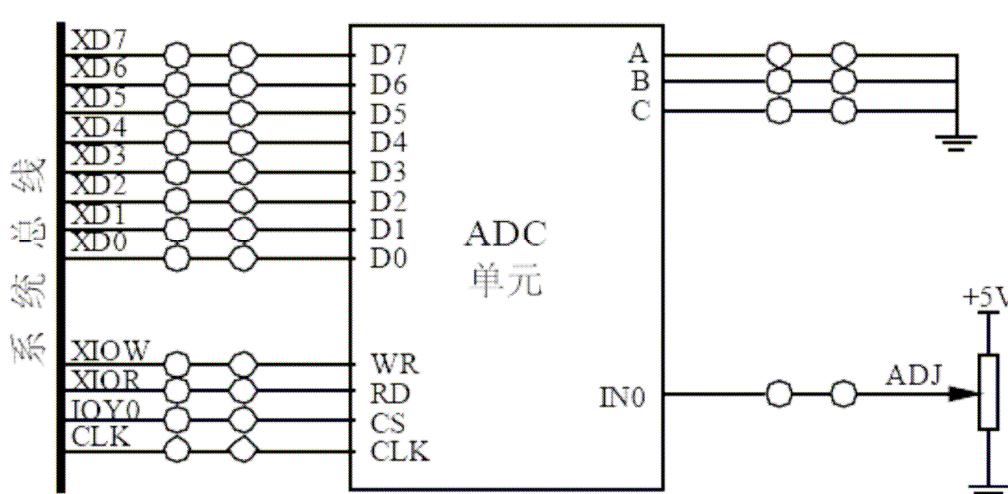


图 3-10-2 A/D 转换实验参考接线图

十三 D/A 转换实验

1、实验目的

- (1) 学习掌握D/A转换原理及接口设计方法。
- (2) 掌握DAC0832芯片的使用方法。

2、实验内容及说明

设计实验线路并编写程序，实现数字信号到模拟信号的转换，输入数字量由程序给出。
要求产生方波和三角波，并用示波器观察输出模拟信号的波形。

3、实验原理

D/A转换器是一种将数字量转换成模拟量的器件，其特点是：接收、保持和转换的数字信息，不存在随温度、时间漂移的问题，其电路抗干扰性较好。大多数的D/A转换器接口设计主要围绕D/A集成芯片的使用及配置响应的外围电路。DAC0832是8位芯片，采用CMOS工艺和R-2RT形电阻解码网络，转换结果为一对差动电流Iout1和Iout2输出。DAC0832引脚如图3-11-1所示。主要性能参数如表3-11-1示，

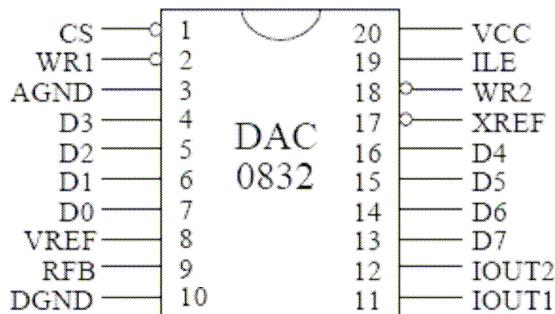


图 3-11-1 DAC0832 的引脚图

表 3-11-1 DAC0832 性能参数

性能参数	参数值
分辨率	8 位
单电源	+5V- +15V
参考电压	+10V- -10V
转换时间	1Us
满刻度误差	±1LSB
数据输入电平	与 TTL 电平兼容

4、实验步骤

- (1) 确认从PC机引出的两根扁平电缆已经连接在实验平台上。
- (2) 首先运行CHECK程序，查看I/O空间始地址。
- (3) 利用查出的地址编写程序，然后编译链接。
- (4) 参考图3-11-2所示连接实验线路。
- (5) 运行程序，用示波器观察输出模拟信号波形是否正确。

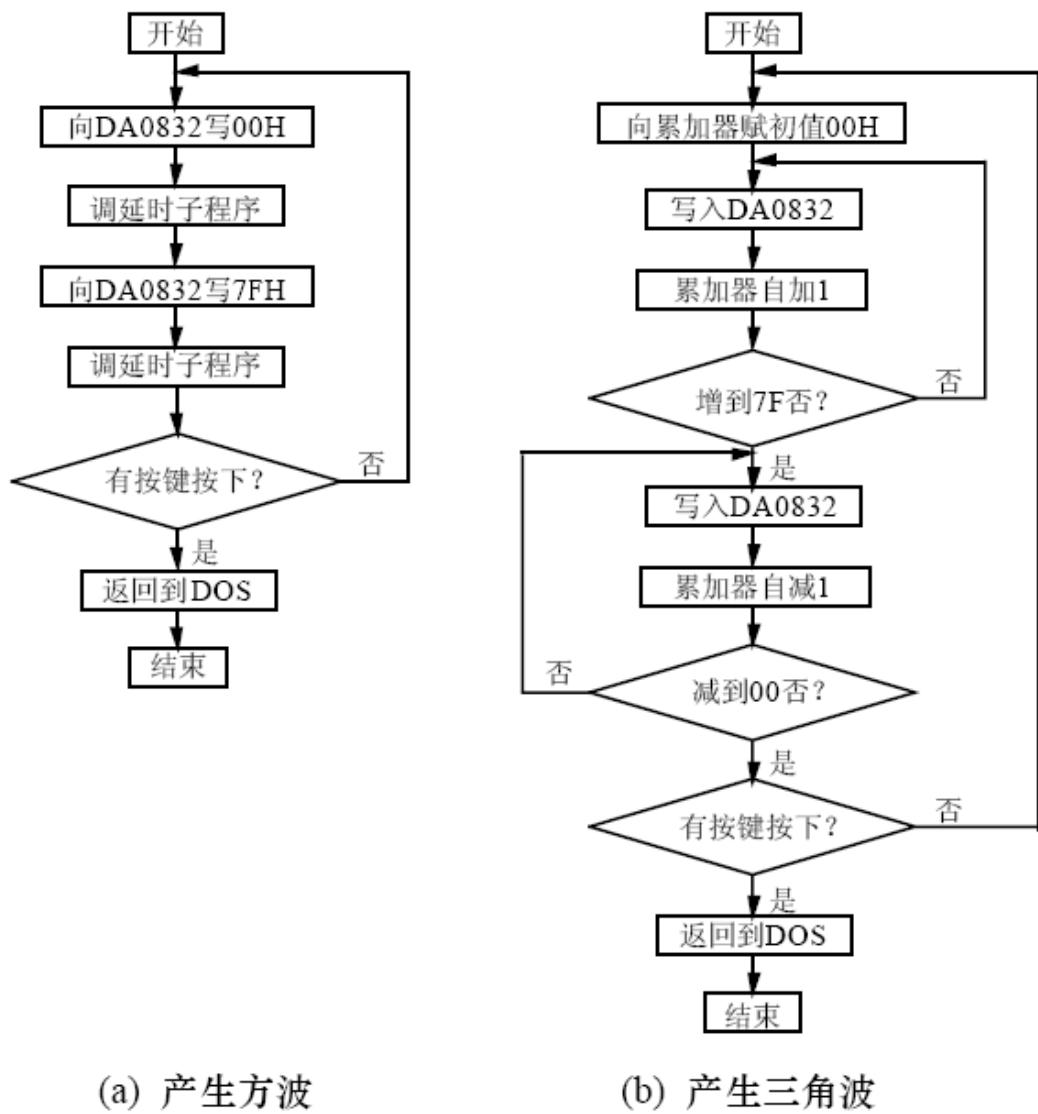


图 3-11-2 D/A 转换实验参考程序流程图

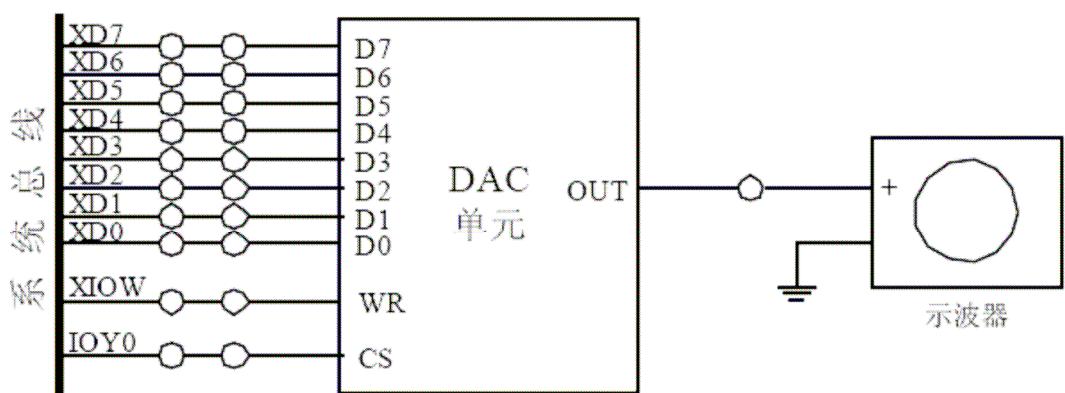


图 3-11-2 D/A 转换实验参考接线图

十四 电子发声设计实验

1、实验目的

学习用8254定时/计数器使扬声器发声的编程方法。

2、实验内容

根据实验提供的音乐频率表和时间表，编写程序控制8254，使其输出连接到扬声器上能发出相应的乐曲。

3、实验说明及步骤

一个音符对应一个频率，将对应一个音符频率的方波送到扬声器上，就可以发出这个音符的声音。音符与频率对照关系如表3-12-1所示。将一段乐曲的音符对应频率的方波依次送到扬声器，就可以发出这段乐曲的声音。

表 3-12-1 音符与频率对照表 (单位: Hz)

音符 音调	1 ·	2 ·	3 ·	4 ·	5 ·	6 ·	7 ·
A	221	248	278	294	330	371	416
B	248	278	312	330	371	416	467
C	131	147	165	175	196	221	248
D	147	165	185	196	221	248	278
E	165	185	208	221	248	278	312
F	175	196	221	234	262	294	330
G	196	221	248	262	294	330	371
音符 音调	1	2	3	4	5	6	7
A	441	495	556	589	661	742	833
B	495	556	624	661	742	833	935
C	262	294	330	350	393	441	495
D	294	330	371	393	441	495	556
E	330	371	416	441	495	556	624
F	350	393	441	467	525	589	661
G	393	441	495	525	589	661	742
音符 音调	· 1	· 2	· 3	· 4	· 5	· 6	· 7
A	882	990	1112	1178	1322	1484	1665
B	990	1112	1248	1322	1484	1665	1869
C	525	589	661	700	786	882	990
D	589	661	742	786	882	990	1112
E	661	742	833	882	990	1112	1248
F	700	786	882	935	1049	1178	1322
G	786	882	990	1049	1178	1322	1484

利用8254的方式3—“方波发生器”，将相应一种频率的计数初值写入计数器，就可产生对应频率的方波。计数初值的计算如下：

$$\text{计数初值} = \text{输入时钟} \div \text{输出频率}$$

例如输入时钟采用系统总线上CLK(1.041667MHz)，要得到800Hz的频率，计数初值即为 $1041667/800$ 。对于每一个音符的演奏时间，可以通过软件延时来处理。首先确定单位延时时间程序（这个要根据PC机的CPU频率做相应的调整）。然后确定每个音符演奏需要几个单位时间，将这个值送入DL中，调用DALLY子程序即可。

;单位延时时间

DALLY PROC

DO: MOV CX, 200H

```
D1: MOV AX, 0FFFFH  
D2: DEC AX  
JNZ D2  
LOOP D1  
RET  
DALLY ENDP  
;N个单位延时时间 (N送至DL)  
DALLY PROC  
D0: MOV CX, 200H  
D1: MOV AX, 0FFFFH  
D2: DEC AX  
JNZ D2  
LOOP D1  
DEC DL  
JNZ D0  
RET  
DALLY ENDP
```

下面提供了乐曲《友谊地久天长》的频率表和时间表。频率表是将曲谱中的音符对应的频率值依次记录下来（B调、四分之二拍），时间表是将各个音符发音的相对时间记录下来（由曲谱中节拍得出）。

```
FREQ_LIST DW 371, 495, 495, 495, 624, 556, 495, 556, 624 ;频率表  
DW 495, 495, 624, 742, 833, 833, 742, 624  
DW 624, 495, 556, 495, 556, 624, 495, 416, 416, 371  
DW 495, 833, 742, 624, 624, 495, 556, 495, 556, 833  
DW 742, 624, 624, 742, 833, 990, 742, 624, 624, 495  
DW 556, 495, 556, 624, 495, 416, 416, 371, 495, 0  
TIME_LIST DB 4, 6, 2, 4, 4, 6, 2, 4, 4 ;时间表  
DB 6, 2, 4, 4, 12, 1, 3, 6, 2
```

DB 4, 4, 6, 2, 4, 4, 6, 2, 4, 4

DB 12, 4, 6, 2, 4, 4, 6, 2, 4, 4

DB 6, 2, 4, 4, 12, 4, 6, 2, 4, 4

DB 6, 2, 4, 4, 6, 2, 4, 4, 12

频率表和时间表是一一对应的，频率表的最后一项为0，作为重复的标志。根据频率表中的频率算出对应的计数初值，然后依次写入8254的计数器。将时间表中相对时间值带入延时程序来得到音符演奏时间。实验参考程序流程如图3-12-1所示。

实验步骤如下。

- (1) 确认从PC机引出的两根扁平电缆已经连接在实验平台上。
- (2) 首先运行CHECK程序，查看I/O空间始地址。
- (3) 利用查出的地址参考流程图编写程序，然后编译链接。
- (4) 参考图3-12-2所示连接实验线路。
- (5) 运行程序，听电子发声单元的扬声器发出的音乐是否正确。

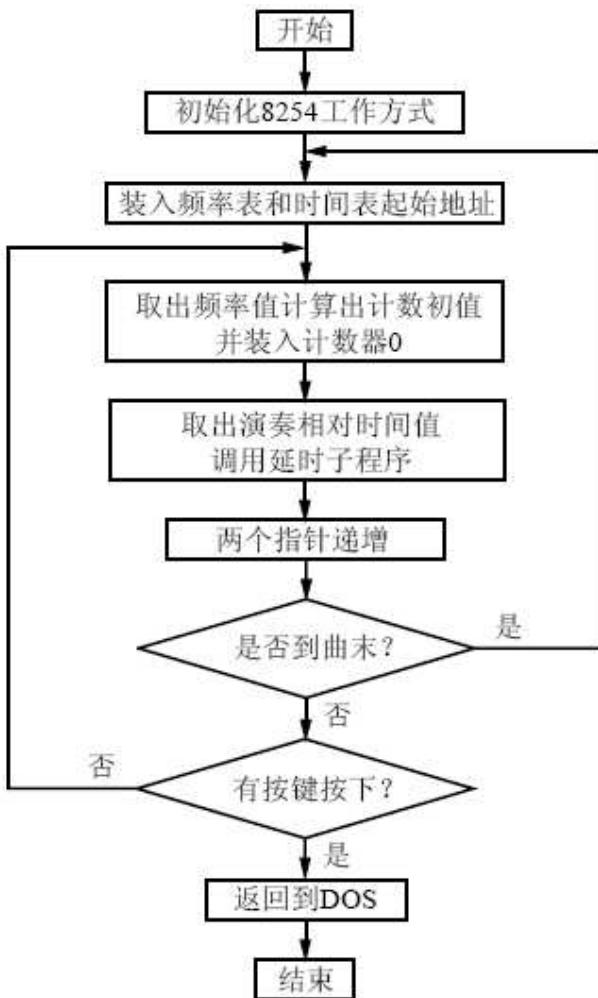


图 3-12-1 电子发声设计实验参考程序流程图

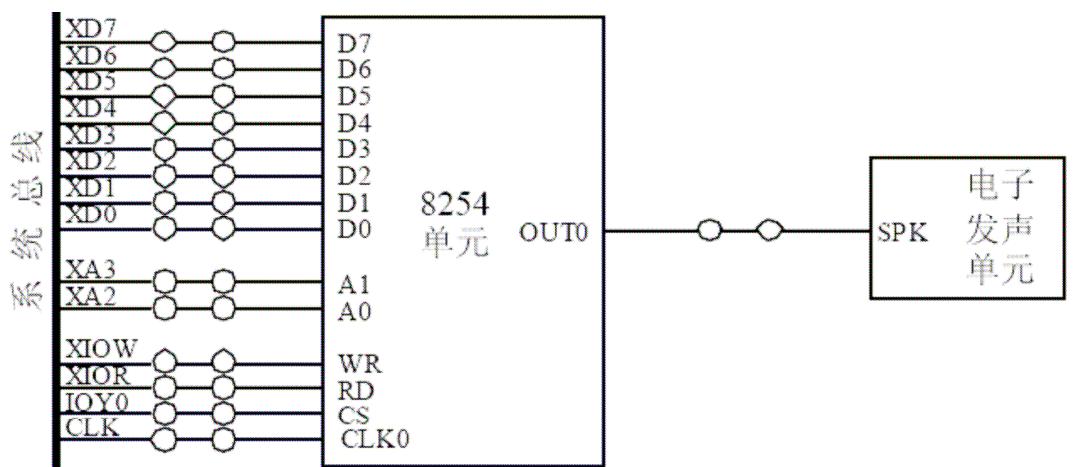


图 3-12-2 电子发声设计实验参考接线图

十五 键盘扫描及显示设计实验

1、实验目的

- (1) 学习按键扫描的原理及电路接法。
- (2) 掌握利用8255完成按键扫描及显示。

2、实验内容

连接8255与键盘扫描单元，编写程序完成按键扫描功能，并将读到的按键值依次显示在数码管上。

3、实验说明及步骤

实验系统中的键盘及数码管显示单元提供了4行×4列共16个按键，4个7段数码管也接成扫描电路方式。共用段位控制信号A~D_p，各自独立的公共端及列选择信号X₁~X₄，行扫描信号为Y₁~Y₄。电路原理如图3-13-1所示。

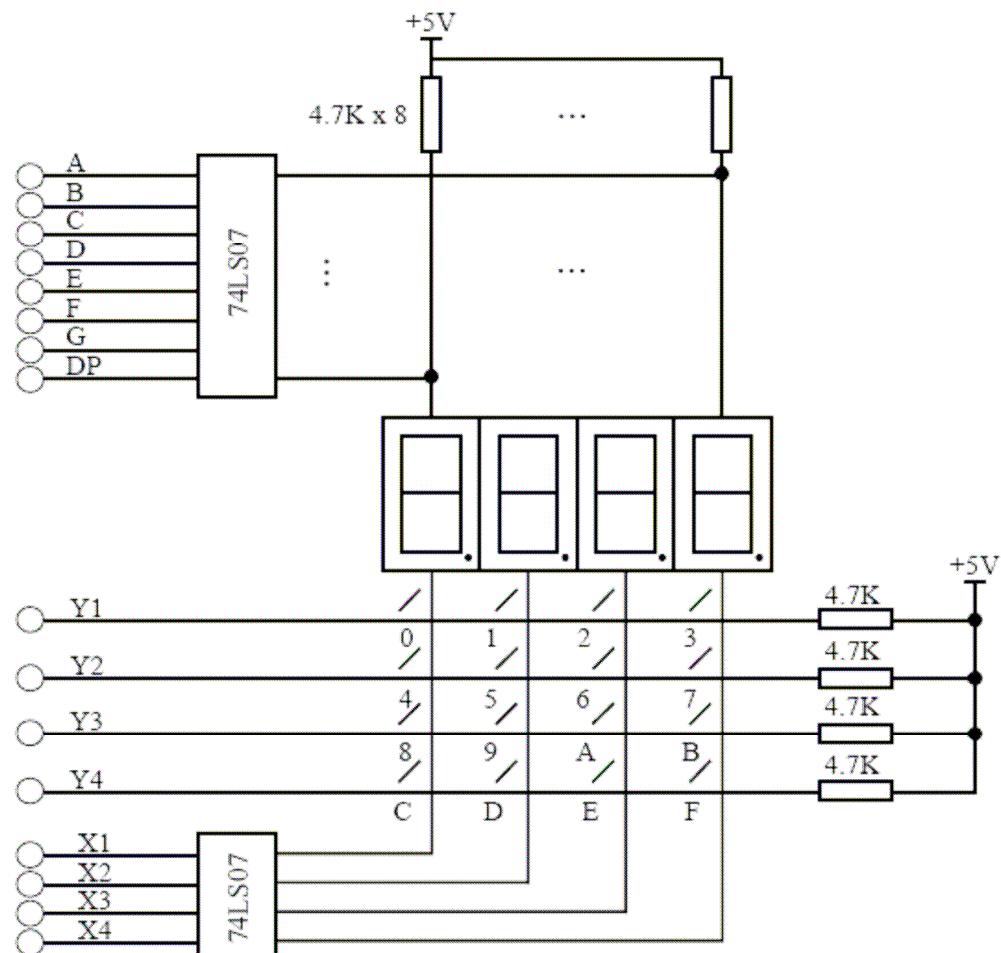


图 3-13-1 键盘及数码管显示单元电路结构图

X₁~X₄控制4位数码管的公共端，并连接到按键的一端，作为列选，确定是哪一列的按键按下。Y₁~Y₄连接到按键的另一端，作为行选，用于检测哪一行的按键按下。结合行选和列选即可得出是哪一个按键按下。在软件设计上要注意消除按键抖动的处理以及数码管显示的刷新。参考程序流程如图3-13-2所示，参考实验接线如图3-13-3所示。

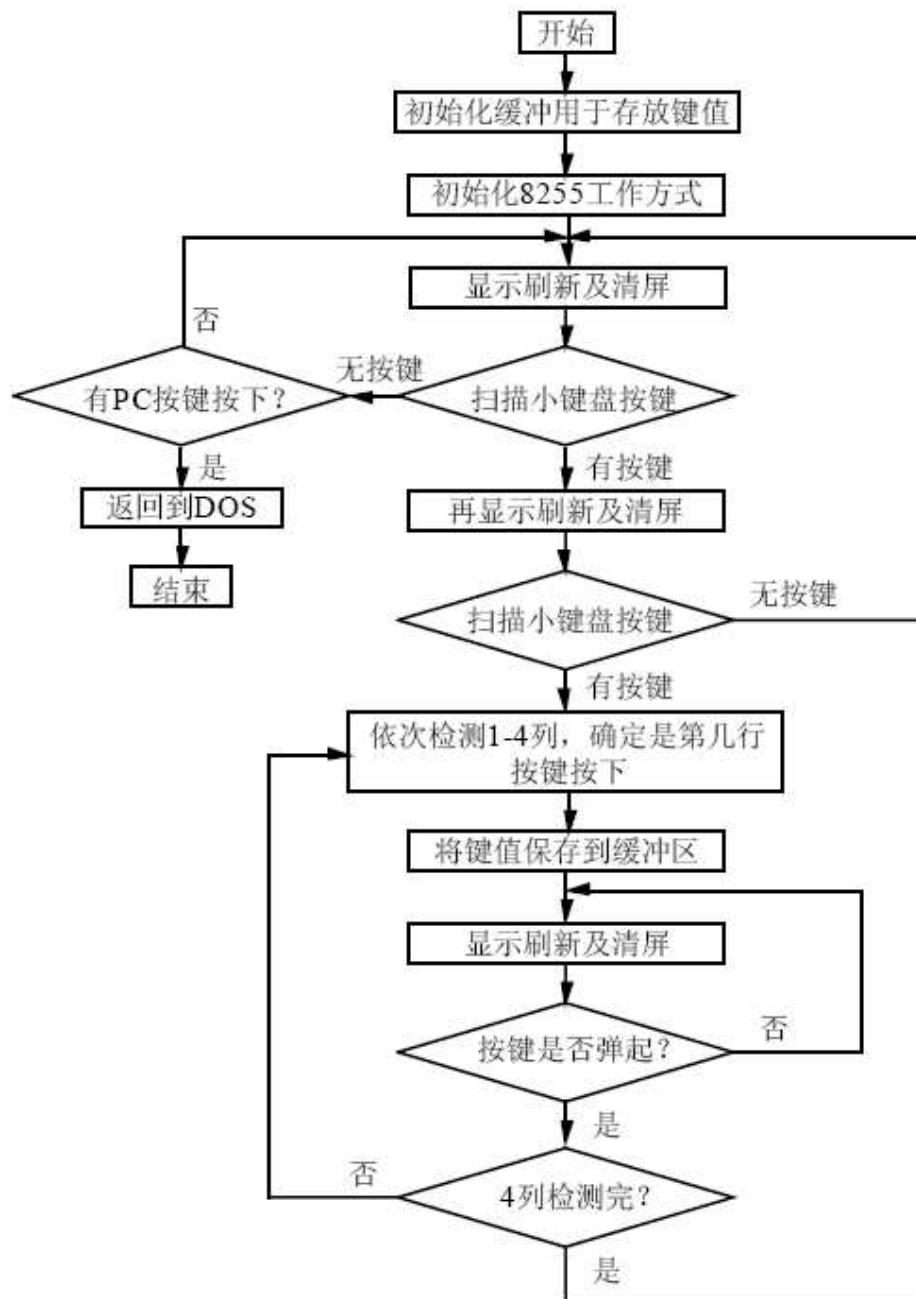


图 3-13-2 键盘扫描及显示实验参考程序流程图

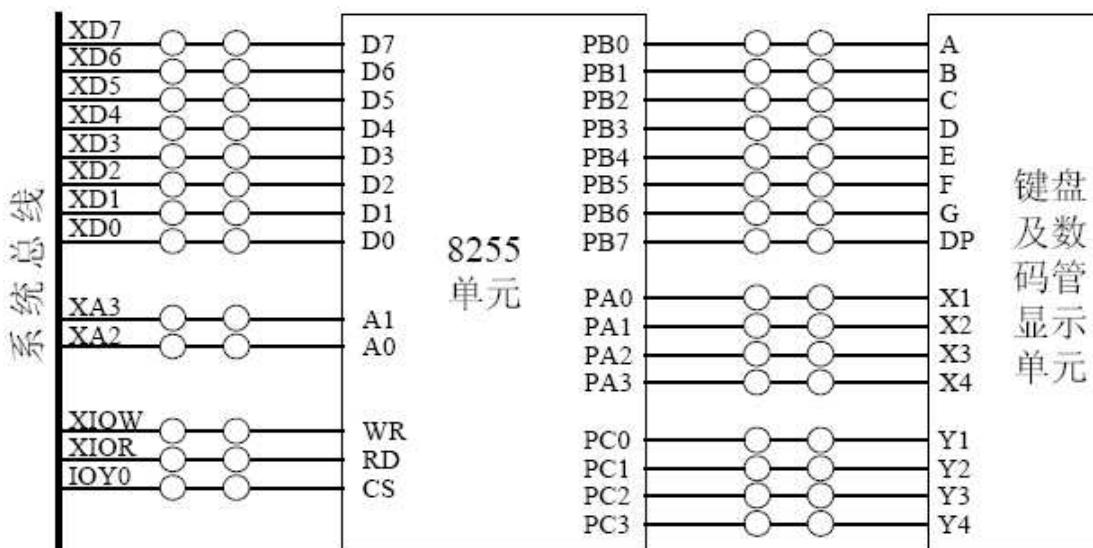


图 3-13-3 键盘扫描及显示实验参考接线图

实验步骤如下。

- (1) 确认从PC机引出的两根扁平电缆已经连接在实验平台上。
- (2) 首先运行CHECK程序，查看I/O空间始地址。
- (3) 利用查出的地址编写程序，然后编译链接。
- (4) 参考图3-13-3所示连接实验线路。
- (5) 运行程序，观察数码管显示是否正确。