**上海大学微机实践报告**

**实验一:开关状态显示**

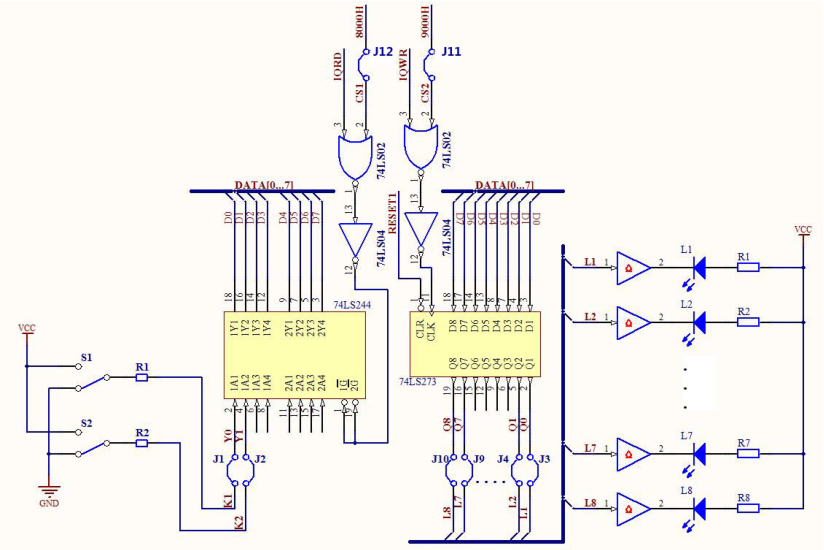
【实验目的】

熟悉实验箱和软件开发平台的使用。了解基本 I/O 端口的操作方法和技巧，掌握编程和调试基本技能。

【实验内容】

利用 74LS244 作为输入口，读取开关状态，根据给定表格中开关状态对应的输出关系，通过 74LS273 驱动发光二极管显示出来。

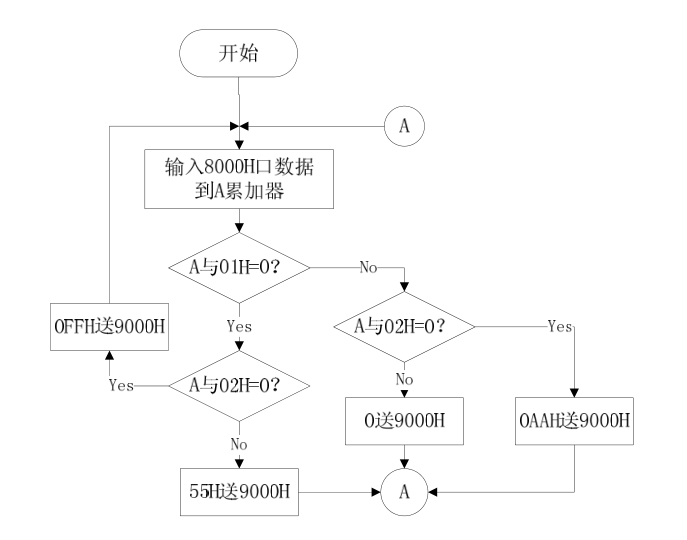
【实验区域电路连接图】



参考上图连线：

Y0～Y1 接 K1～K2（对应 J1、J2）；Q0～Q7 接 L1～L8（对应 J3 至 J10）；CS1 接 8000H孔（对应 J12）；CS2 接 9000H 孔（对应 J11）；IOWR→IOWR；IORD→IORD；然后用数据排线连接 JX7→JX17（BUS2）。

【程序框图】



【编程】

CODE SEGMENT

ASSUME CS:CODE

ORG 1200H

START:

MOV DX,8000H

IN AL,DX

TEST AL,01H

JNZ Y1 /测试AL是否等于01H，若不等于，则跳入循环Y1/

TEST AL,02H

JNZ Y2 /测试AL是否等于02H，若不等于，则跳入循环Y2/

MOV DX,9000H

MOV AL,0FFH /点亮所有的灯/

OUT DX,AL

JMP START

Y1:TEST AL,02H

JNZ Y3 /测试AL是否等于02H，若不等于，则跳入循环Y3/

MOV DX,9000H

MOV AL,0AAH /点亮奇数号灯/

OUT DX,AL

JMP START

Y2:MOV DX,9000H

MOV AL,55H /点亮偶数号灯/

OUT DX,AL

JMP START

Y3:MOV DX,9000H

MOV AL,00H /所有灯全灭/

OUT DX,AL

JMP START

CODE ENDS

END START

【问答题】

**1、I/O 端口的寻址方式有哪 2 种？在 x86 系统中，采用哪一种？**

解：I/O 端口的寻址方式有：一种是I/O地址与存储器地址统一编址，一种是I/O地址与存储器地址分别独立编址。在x86 系统中，采用I/O地址与存储器地址分别独立编址。

**2、在输入/输出电路中，为什么常常要使用锁存器和缓冲器？**

解：输入/输出按方式分为程序控制和DMA两类，程序控制方式又分为同步方式、异步方式和中断方式。

在程序控制方式中，同步方式的软件编程和硬件实现都十分简单，当需要时，就可以直接在需要输入/输出的地方写输入/输出指令。异步方式由于需要根据状态信息进行输入/输出操作，就需要增加必要的“握手”信号电路，程序设计也要包括对状态信息的判断，根据状态信息再决定是继续等待还是将信息读入或输出。中断方式在程序控制输入/输出方式中比较复杂，需要有中断控制器等硬件电路和接口芯片，程序也包括了中断处理程序来完成计算机输入、输出。

DMA是另一类输入/输出方式，在进行输入/输出时，不需要CPU直接参加操作，而仅是将总线空出给DMA控制器使用。当然，DMA方式要求的硬件电路更加复杂。其优点是响应速度快，缺点是需要增加硬件电路。

【心得体会】

通过本次试验的操作，我熟悉了试验箱和软件开发平台的使用，也了解到了智能芯片接口8255的端口配置方法，掌握了编程和调试基本技能，加深了对课堂上理论知识的学习，将理论与实践动手相结合，更好的掌握了知识的全面性。

**实验二 模拟交通灯实验**

【实验目的】

1. 学习在PC机系统中扩展简单I/O 接口的方法。

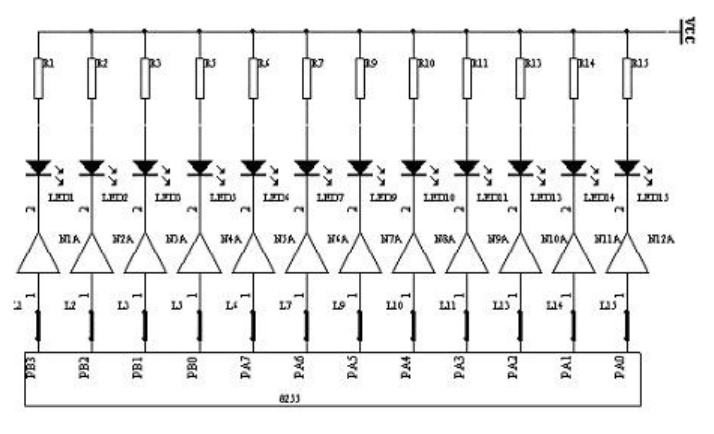
2. 进一步学习编制数据输出程序的设计方法。

3. 学习模拟交通灯控制的方法。

【实验内容】

用 8255 做输出口，控制十二个发光管（4 组红绿黄灯）的亮灭，模拟十字路口交通灯管理。

【实验区域电路连接图】



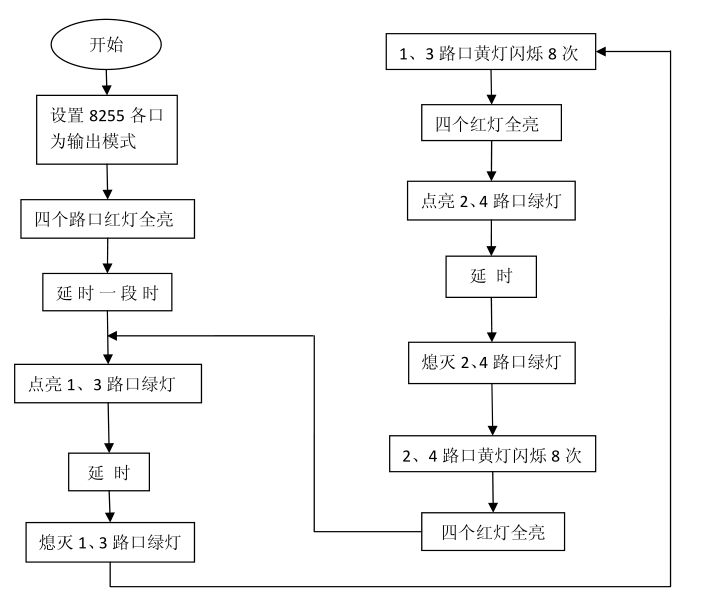
PC0→L3；PC1→L7；PC2→L11；PC3→L15；红灯

PC4→L2；PC5→L6；PC6→L10；PC7→L14；绿灯

PB4→L1；PB5→L5；PB6→L9；PB7→L13。 黄灯

8255 数据、控制线内部已线连好。

【程序框图】



【编程】

CODE SEGMENT

ASSUME CS:CODE

ORG 1200H

PORTA EQU 0FF28H

PORTB EQU 0FF29H

PORTC EQU 0FF2AH

PORTCN EQU 0FF2BH

START:

MOV DX,PORTCN

MOV AL,80H

OUT DX,AL

MOV DX,PORTB

MOV AL,0F0H

OUT DX,AL /黄灯熄灭/

MOV DX,PORTC

MOV AL,0F0H

OUT DX,AL /绿灯熄灭，红灯点亮/

CALL DELAY1 /延时1/

MOV DX,PORTC

MOV AL,0A5H

OUT DX,AL /1、3路口绿灯亮，2、4路口红灯亮/

CALL DELAY3 /延时3/

MOV DX,PORTC

MOV AL,0F5H

OUT DX,AL /1、3绿灯熄灭（2、4红灯亮）/

MOV CX,8

LOOP1:

MOV DX,PORTB

MOV AL,0A0H

OUT DX,AL /1、3黄灯亮/

MOV DX,PORTC

MOV AL,0F5H

OUT DX,AL /绿灯全灭，2、4红灯亮/

CALL DELAY2 /延时2/

MOV DX,PORTB

MOV AL,0FFH

OUT DX,AL /黄灯全部灭/

CALL DELAY2

DEC CX

JNZ LOOP1

MOV DX,PORTB

MOV AL,0F0H

OUT DX,AL /黄灯全灭/

MOV DX,PORTC

MOV AL,0F0H

OUT DX,AL /绿灯熄灭，红灯点亮/

CALL DELAY1 /延时1/

MOV DX,PORTC

MOV AL,5AH

OUT DX,AL /1、3红灯亮，2、4绿灯亮/

MOV DX,PORTB

MOV AL,0F0H

OUT DX,AL /黄灯全灭/

CALL DELAY3 /延时3/

MOV DX,PORTC

MOV AL,0FAH

OUT DX,AL /2、4绿灯灭（1、3红灯亮）/

MOV CX,8

LOOP2:

MOV DX,PORTB

MOV AL,50H

OUT DX,AL /2、4黄灯亮/

MOV DX,PORTC

MOV AL,0FAH

OUT DX,AL /1、3红灯亮，绿灯全灭/

CALL DELAY2 /延时2/

MOV DX,PORTB

MOV AL,0FFH

OUT DX,AL /黄灯全灭/

CALL DELAY2

DEC CX

JNZ LOOP2

JMP START

DELAY1:

MOV AX,5H

MOV BX,0H

X:DEC BX

JNZ X

DEC AX

JNZ X

RET

DELAY2:

MOV AX,1H

MOV BX,0H

Y:DEC BX

JNZ Y

DEC AX

JNZ Y

RET

DELAY3:

MOV AX,20H

MOV BX,0H

Z:DEC BX

JNZ Z

DEC AX

JNZ Z

RET

CODE ENDS

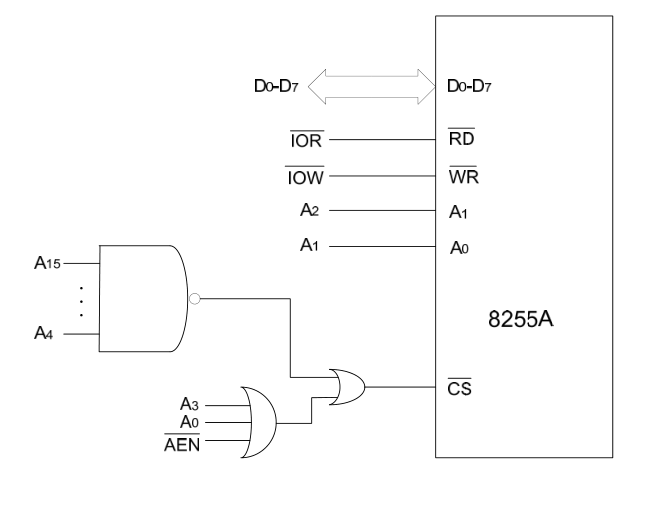
END START

【问答题】

**1、8255A 的哪个端口能实现位操作控制（按位置位/复位）？执行控制字写入操作的端口是哪个？**

解：8255A 的C端口能实现位操作控制（按位置位/复位），执行控制字写入操作的端口是命令字端口。

**2、写出下图中 Intel8255A 占用的 4 个端口地址。**



解：FFF0H、FFF2H、FFF4H、FFF6H。

【心得体会】

本实验是模拟交通灯工作的实验，通过8255A 控制发光二极管，PB4-PB7 对应黄灯，PC0-PC3 对应红灯，PC4-PC7 对应绿灯,以模拟交通路灯的管理。在实验中不但要满足题目所提出的要求，更要结合实际，考虑到现实问题，比如说绿灯时间要比红灯时间长。通过本次实验，不仅再次熟悉了8255A，更是对自己编程技术一个很好的锻炼。

**实验三：8253定时器/计数器实验**

**【实验目的】**

1.学会8253芯片与微机接口的原理和方法；

2.掌握8253定时器/计数器的工作原理和编程方法。

**【实验内容】**

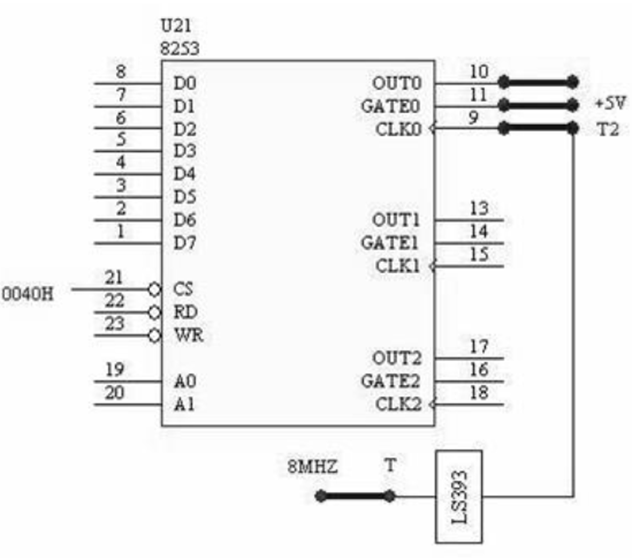
编写程序，将8253的计数器0设置为方式2（频率发生器），计数器1设置为方式3（方波频率发生器），计数器0的输出作为计数器1的输入，计数器1的输出接在一个LED上，运行后可观察到该LED在不停地闪烁。

1.编程时用程序框图中的二个计数初值，计算OUT1的输出频率，用表观察LED，进行核对；

2.修改程序中的二个计数初值，使OUT1的输出频率为1Hz，用手表观察LED，进行核对；

3.上面计数方式选用的是16进制，现若改用BCD码，试修改程序中的二个计数初值，使LED的闪亮频率仍为1Hz。

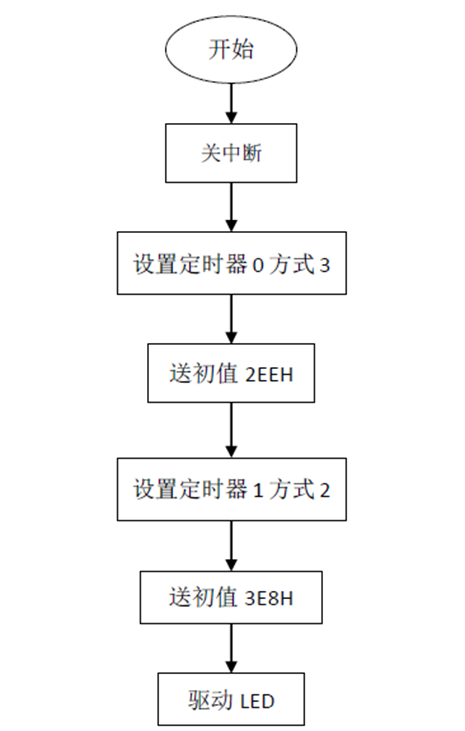
**【实验区域电路连接图】**



CS3→0040H；JX8→JX0；IOWR→IOWR；IORD→IORD；A0→A0；A1→A1；

GATE0→+5V；GATE1→+5V+5V；OUT0→CLK1；OUT1→L1；CLK0→0.5MHz；(单脉冲与时钟元)

**【程序框图】**



**【编程】**

（最终实验效果程序）

CODE SEGMENT

ASSUME CS:CODE

ORG 1000H

START:

MOV DX,43H ;写8253控制字

MOV AL,00110101B ;计数器0，选16位（先低后高），方式2，

OUT DX,AL ;BCD码计数

MOV DX,40H ;计数器0

MOV AL,00H ;送初值，低8位

OUT DX,AL

MOV AL,05H ;送初值，高8位

OUT DX,AL

MOV DX,43H ; 写8253控制字

MOV AL,01110111B ;计数器1，选16位（先低后高），方式3，

OUT DX,AL ;BCD码计数

MOV DX,41H ;计数器1

MOV AL,00H ;送初值，低8位

OUT DX,AL

MOV AL,10H ;送初值，高8位

OUT DX,AL

JMP $

CODE ENDS

END START

如要用二进制计数，则修改控制字最后一位为0，并修改相应初值。

问题二：注释和一基本相同

CODE SEGMENT

ASSUME CS:CODE

ORG 1000H

START:

MOV DX,43H

MOV AL,00110100B ;控制寄存器的初始化

OUT DX,AL

MOV DX,40H ;计数器0

MOV AL,0F4H ;写入初值

OUT DX,AL

MOV AL,01H

OUT DX,AL

MOV DX,43H ;控制寄存器的初始化

MOV AL,01110110B

OUT DX,AL

MOV DX,41H ;计数器1

MOV AL,0E8H ;写入初值

OUT DX,AL

MOV AL,03H

OUT DX,AL

JMP $ ;结束

CODE ENDS

END START

第三个问题：改成BCD码，注释和二一样。

CODE SEGMENT

ASSUME CS:CODE

ORG 1000H

START:

MOV DX,43H

MOV AL,00110101B

OUT DX,AL

MOV DX,40H

MOV AL,00H ;写入初值

OUT DX,AL

MOV AL,05H

OUT DX,AL

MOV DX,43H

MOV AL,01110111B

OUT DX,AL

MOV DX,41H

MOV AL,00H ;写入初值

OUT DX,AL

MOV AL,10H

OUT DX,AL

JMP $

CODE ENDS

END START

**【问答题】**

**1、8253初始化编程时需要遵循的原则是什么？**

解：答：在编程时必须遵守两条原则：

(1).在对某个计数器设置初值之前，必须先写入控制字；

(2).在设置计数器初始值时，要符合控制字的规定，即只写低位字节，还是只写高位字节，还是低。高位字节都写（分两次写，先低后高）。

**2、简述8253初始化编程的步骤。**

解：

1.首先向控制寄存器写入控制字，以选定计数通道（三个中之一），规定该计数的工作方式和计数方式以及计数初值的长度和装入顺序（初值写入方式）；

2.向已选定的计数器按控制字的要求 写入计数初值。

**【心得体会】**

这次实验主要用到8253的知识，要知道8253的控制字，每一位代表的意义，以及8253的控制字以及命令字的端口地址。要了解BCD码与数制之间的转换。通过这次的实验，巩固了这些知识，并加深了了解。

通过这次实验，我觉得，能够将学到的知识应用于实践中去，是非常有意义的，同时在实践中，又能巩固理论知识，是一件两全其美的事情。所以一定要认真的去完成每一次实验,这样在能在实践中得到锻炼，让所学过的知识更加深刻。通过本次实验，我更进一步熟悉了8253，并更好的锻炼了自己对于编程知识的运用。