

# 交通灯控制实验

## 一、实验目的

通过并行接口 8255 实现十字路口交通灯的模拟控制,进一步掌握对并行口的使用。

## 二、实验内容

如下图 1 所示, L7、L6、L5 作为南北路口的交通灯与 PC7、PC6、PC5 相连, L2、L1、L0 作为东西路口的交通灯与 PC2、PC1、PC0 相连。编程使六个灯按交通灯变化规律亮灭。

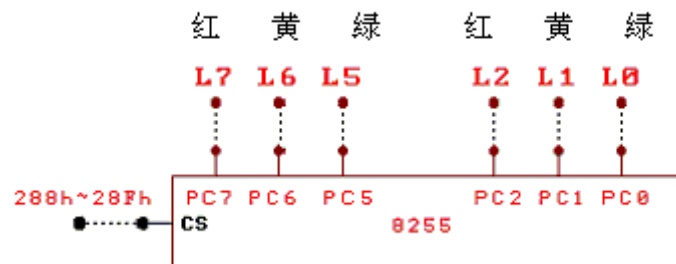


图 1

## 三、编程提示:

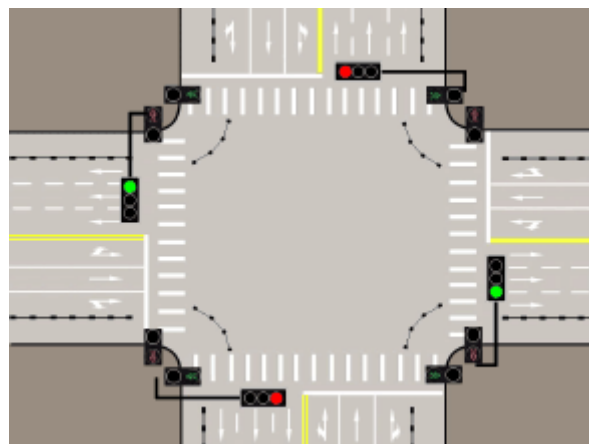


图 2 交通信号灯

十字路口交通灯的变化规律要求：

- (1) 南北路口的绿灯、东西路口的红灯同时亮 30 秒左右。
- (2) 南北路口的黄灯闪烁若干次，同时东西路口的红灯继续亮。
- (3) 南北路口的红灯、东西路口的绿灯同时亮 30 秒左右。
- (4) 南北路口的红灯继续亮、同时东西路口的黄灯亮闪烁若干次。
- (5) 转(1)重复。

算法流程图：

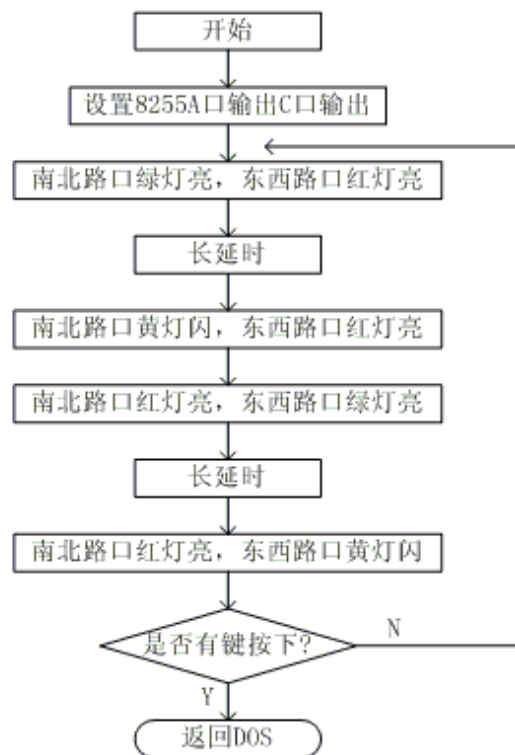


图 3 交通信号灯控制流程图

说明：六个灯状态转换图

1 红 1 黄 **1 绿** 0 0 **2 红** 2 黄 2 绿      灯亮长延时

1 红 **1 黄** 1 绿 0 0 **2 红** 2 黄 2 绿

1 红 1 黄 1 绿 0 0 **2 红** 2 黄 2 绿

1 红 **1 黄** 1 绿 0 0 **2 红** 2 黄 2 绿

1 红 1 黄 1 绿 0 0 **2 红** 2 黄 2 绿

1 红 **1 黄** 1 绿 0 0 **2 红** 2 黄 2 绿

1 红 1 黄 1 绿 0 0 **2 红** 2 黄 2 绿

**1 红** 1 黄 1 绿 0 0 2 红 2 黄 **2 绿**      灯亮长延时

**1 红** 1 黄 1 绿 0 0 2 红 **2 黄** 2 绿

**1 红** 1 黄 1 绿 0 0 2 红 2 黄 2 绿

**1 红** 1 黄 1 绿 0 0 2 红 **2 黄** 2 绿

**1 红** 1 黄 1 绿 0 0 2 红 2 黄 2 绿

**1 红** 1 黄 1 绿 0 0 2 红 **2 黄** 2 绿

**1 红** 1 黄 1 绿 0 0 2 红 2 黄 2 绿

```
portc1  db  24h, 44h, 04h, 44h, 04h, 44h, 04h    ;六个灯可能
          db  81h, 82h, 80h, 82h, 80h, 82h, 80h    ;的状态数据
```

#### 四、参考程序

```
;*****
;*      十字路口红绿灯模拟演示程序
;*  端口各灯的设置:    1 红 1 黄 1 绿 0 0 2 红 2 黄 2 绿    *;
;*****
```

```

data segment
io8255a      equ 28ah
io8255b      equ 28bh
portc1  db  24h, 44h, 04h, 44h, 04h, 44h, 04h ;六个灯可能
          db  81h, 82h, 80h, 82h, 80h, 82h, 80h ;的状态数据
          db  0ffh ;结束标志
data ends
code  segment
      assume  cs:code, ds:data
start:
      mov     ax, data
      mov     ds, ax
      mov     dx, io8255b
      mov     al, 90h
      out     dx, al ;设置 8255 为 C 口输出
      mov     dx, io8255a
re_on:
      mov     bx, 0
on:
      mov     al, portc1[bx]
      cmp     al, 0ffh
      jz      re_on
      out     dx, al ;点亮相应的灯
      inc     bx
      mov     cx, 200 ;参数赋初值
      test    al, 21h ;是否有绿灯亮
      jz      L0 ;没有,短延时
      mov     cx, 2000 ;有,长延时
L0:
      mov     di, 9000 ;di 赋初值 9000
L1:
      dec     di ;减 1 计数
      jnz     L1 ;di 不为 0
      loop    L0

      push    dx
      mov     ah, 06h
      mov     dl, 0ffh
      int     21h
      pop     dx ;判断是否有按键

```

```
        jz  on                ;没有,转到 on
exit:
        mov  ah,4ch          ;返回
        int  21h
code ends
end start
```