

# 系统实验报告

通信工程

1342401004

丁泽南

## 【实验目的】

设计出模拟路口交通灯程序，并结合数码管倒计时显示时间，在 PCB 板子上焊接出相应的模拟电路。

## 【设计任务及要求】

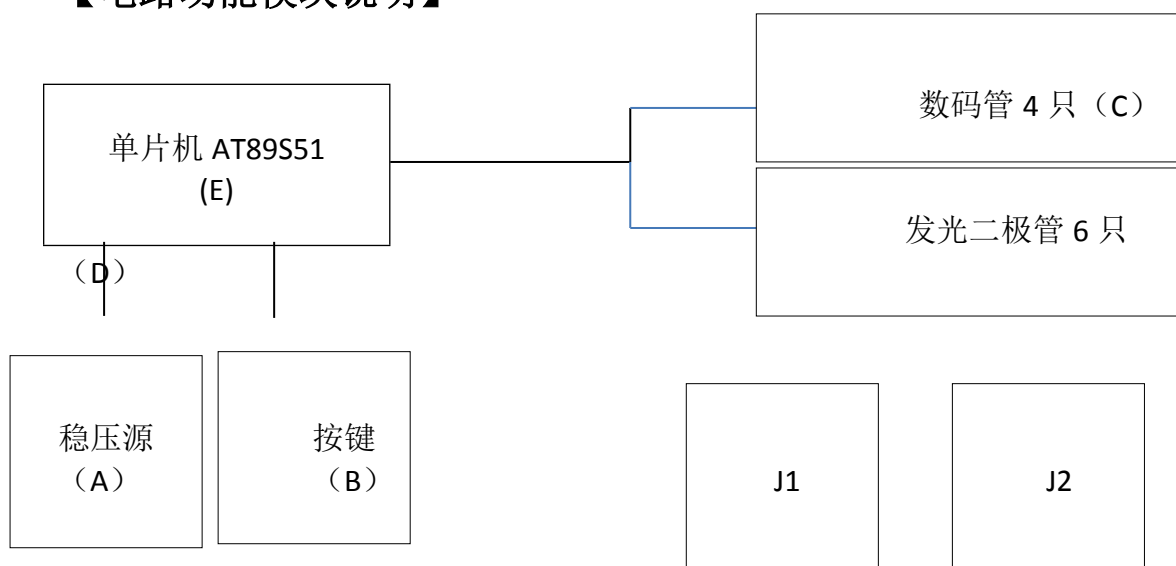
### \*时钟（计时器）

- 1.状态 1：一只数码管 0-9 技术，一只按键控制、结束；
- 2.状态 2：两只数码管 0-59 计数，一只按键控制开始、结束；
- 3.状态 3：三只数码管低二位 0-59 计数后进位给高位 0-9 计数，一只按键控制开始、结束；
- 4.状态 4：四只数码管低二位 0-59 计数后进位给高二位 0-59 计数（即分秒计时），一只按键控制开始、结束；

### \*交通灯

- 5.状态 1：仅亮灯，数码管不工作；  
按下键。红/黄/绿三色灯交替亮：红（20 秒）——黄（闪烁 5 秒）——绿（20 秒）——黄（闪烁 5 秒）——红。
- 6.状态 2：灯和数码管结合，模拟十字路口的交通灯；在以上功能的基础上数码管倒计时显示时间。

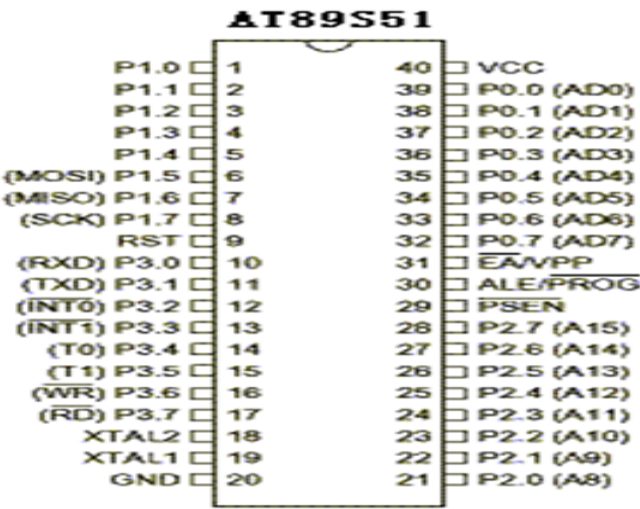
## 【电路功能模块说明】



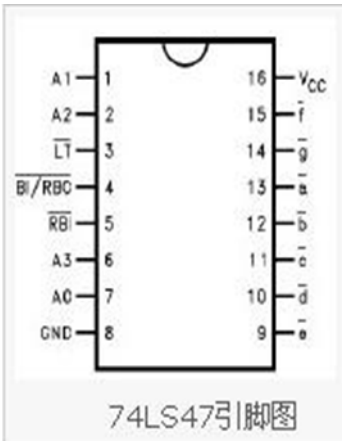
## 【器件功能】

- (A) 稳压源部分将输入的电流电转换为稳定的 5V 直流电压；由于目前提供的已经是直流变压器，所以整流桥可以不用（若不接整流桥 D1，需将电路板上相应的引脚连通，使得接入的 9V 电源能够作用到 7805 上，此处需注意正负极！）；
- (B) 作为单片机的输入部分（P1.0~P1.3）起控制作用；
- (C) 作为单片机的输出部分，P0.0~P0.3 经过 7 段译码器（74LS47）控制数码管的显示结果（0~9），P0.4~P0.7 数码管的选通控制口，提供数码管工作电源（公共端），p2.7 控制数码管的小数点是否点亮；
- (D) 发光二极管：输出部分，由 P2.0~P2.5 口控制，高电平灯亮；
- (E) 单片机 AT89S51:低功耗，高性能 CMOS8 位单片机。片内含 8KB 的 ISP(IN-SYSTEM-PROGRAMMABLE)的可反复擦写 1000 次的 FLASH 只读程序存储器，器件采用 ATMEL 公司的高密度、非易始性存储技术制造，兼容标准 MCS-51 指令系统及 80C51 引脚结构。其特点可概括为：  
40 个引脚；  
8KB FLASH 片内存储器（可反复擦写大于 1000 次）；  
256BYTES 的随机存储器（RAM）；  
32 个外部双向输入（I/O 口）；  
5 个中断优先级，2 层中断嵌套中断；  
2 个全双工串行通信口；  
看门狗（WDT）电路；片内时钟振荡器（0-33MHz）；  
4.5-5.5V 工作电压。
- (F) J1 口留作通信口使用，其中 J1.1(TXD)由 P3.1(11 脚)控制，J1.2(RXD)由 P3.0（10 脚）控制；
- (G) J2 留作在线烧录程序使用。

【引脚图】

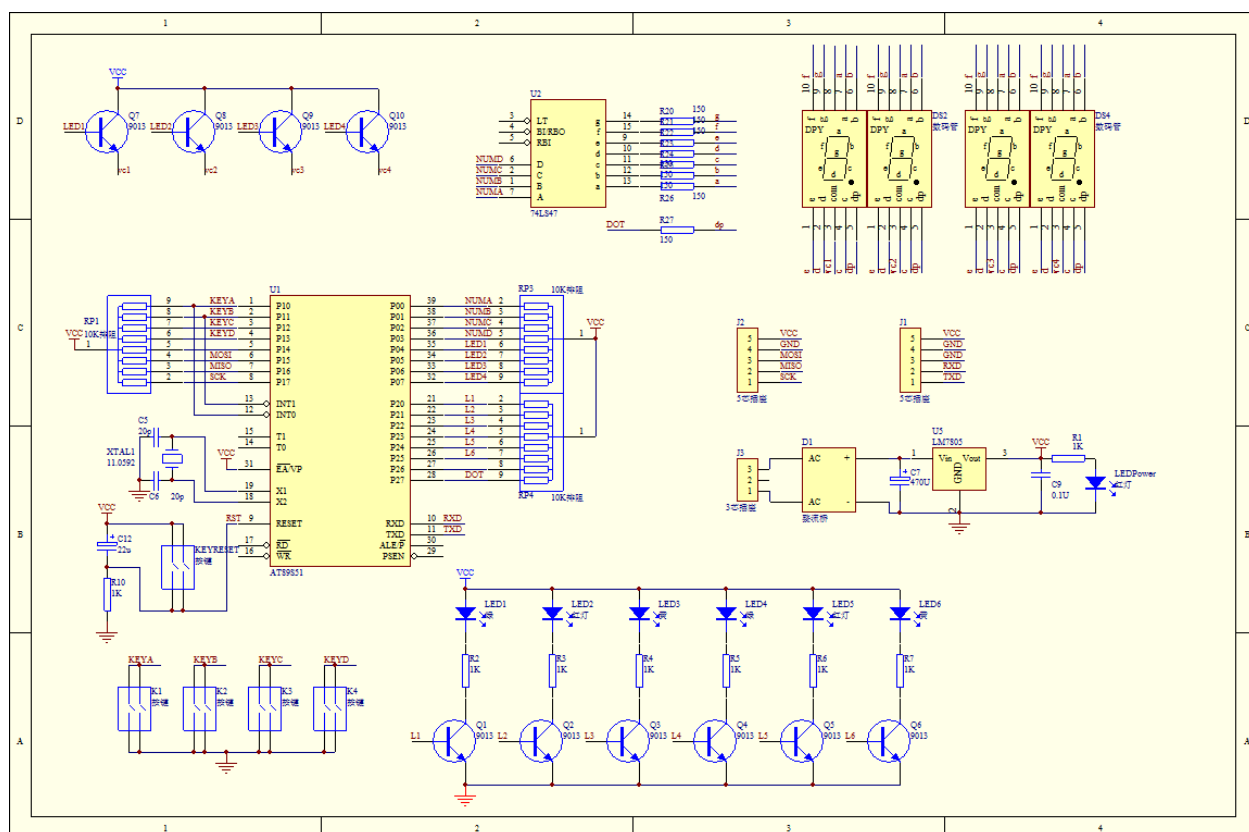


AT89S51 引脚功能图

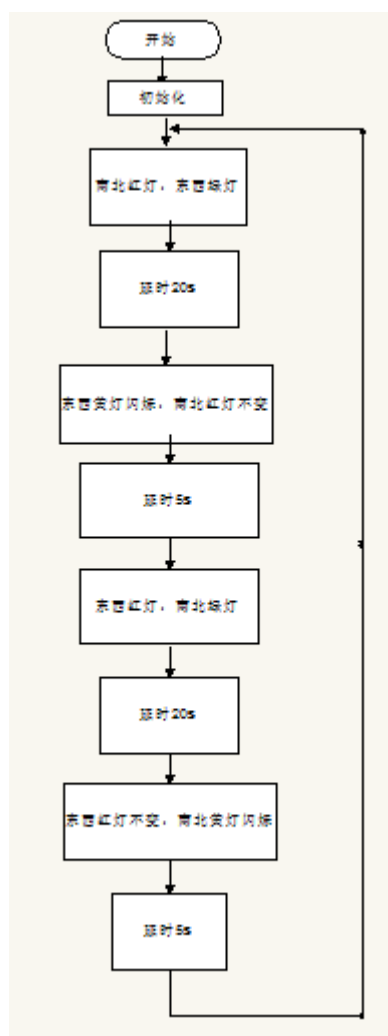


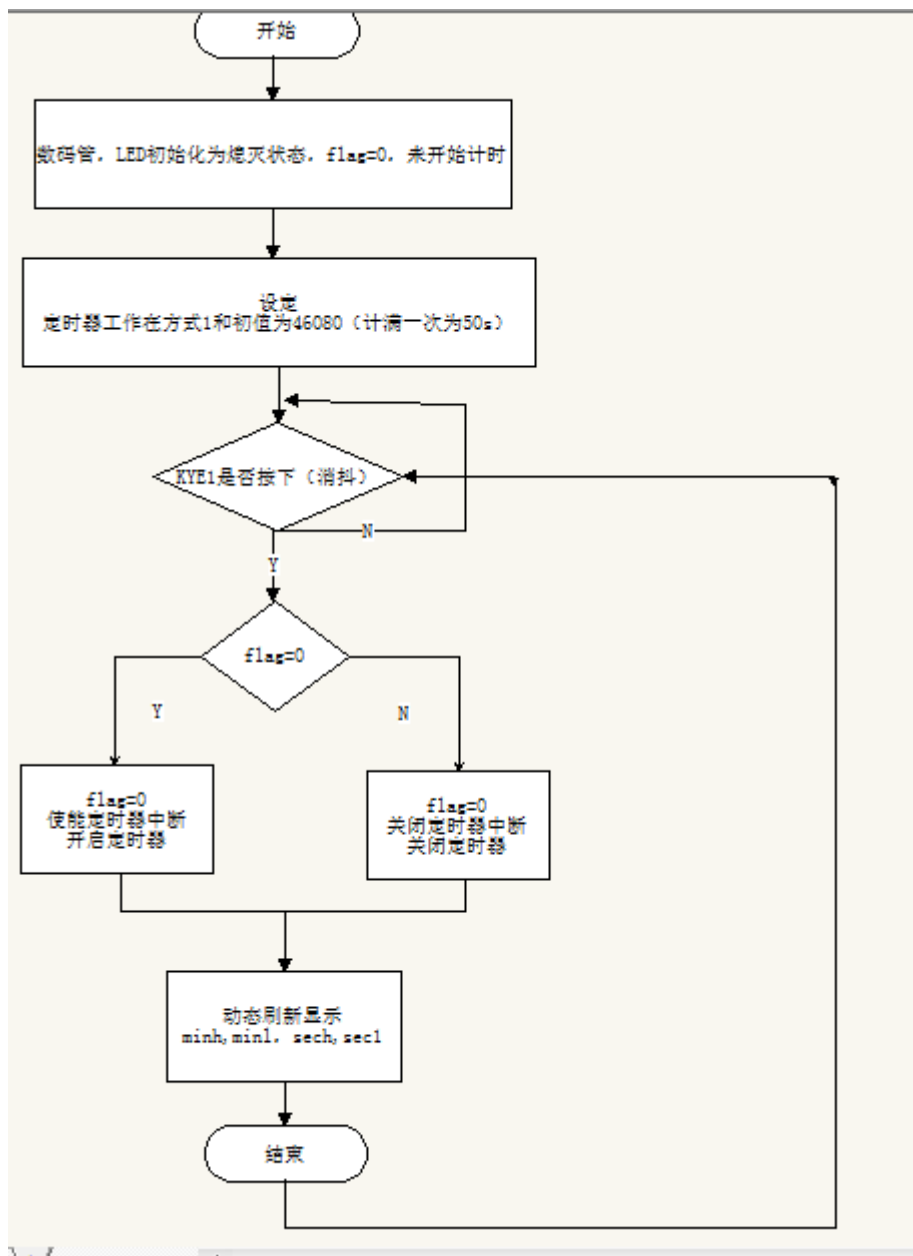
74LS47 引脚图

### 【电路原理图】



## 【软件设计流程图】





## 【实现功能】

交通灯功能和计时器功能结合，通过按键切换。

- ① 时器功能，低二位 0-59 计数进位给高二位 0-59 计数；K1 按键开始，K3 按键暂停/继续
- ② 拟实现十字路口交通灯功能，并结合数码管倒计时显示时间，初始化为南北侧（也可定义为东西侧）亮红灯，东西侧（南北侧）亮绿灯，倒计时二十秒结束后，东西侧绿灯熄灭，变为东西侧黄灯闪烁，并倒计时五秒，五秒倒计时过程中南北侧红灯继续保持五秒不；五秒倒计时结束后，南北侧红灯直接变为绿灯（可以通行车辆），东西侧闪烁的黄灯熄灭，变为红灯（禁止通行）；

继续开始 20 秒倒计时，循环往复；一个完整周期为 50s，一侧通行时间为 25s(包含黄灯 5s)；K2 按键开始，K4 按键暂停/继续

注：黄灯五秒时间实质为东西、南北两侧同时闪烁黄灯，但只要显示在绿灯通行一侧即可，另一侧保持红灯不变即能实现所需功能。

## 【调试过程】

一开始设计时钟状态 4 计时器任务，原因为状态 4 包含前三个状态，使用查询中断方式实现，并参阅看门狗电子程序，数码管 DS3、DS4 代表低二位，DS1、DS2 代表高二位，并设计按键 K1 为开始按键，K3 为暂停\回复按键（使计时器停止工作&恢复），hex 文件烧录进芯片后经过验证，功能完美实现。

接下来设计状态 5，仅亮灯，由于设计此程序仅有灯亮，数码管没有计时，所以设计起来相对简单，为了验证功能我把红绿灯的 20s 减少到了 5s 来加快实验进程。在调试过程中，由于只设计了一侧的红灯亮，而导致误认为红灯损坏，后来添加了 L5 灯的程序，即可。过程中还出现类似死循环部分 break 指令未添加，导致整体程序出错的低级错误，不再赘述。

再然后设计状态 6.模拟交通灯，结合状态 4 和状态 5 的程序，将状态 5 程序中的程序添加到状态 4 的过程中，在设计交通灯主程序中的核心程序是，第一次在调试时，想法是南北侧红灯亮，东西侧绿灯亮，二十秒倒计时后，南北侧黄灯闪烁五秒后，再将红绿灯状态互换，发现不对，会撞车；第二次调试时闪烁黄灯换到绿灯侧，但是绿黄灯同时存在，这种情况不允许，遂作罢；第三次调试第一个半周期出现了问题，黄灯在两次交替闪烁显然不对；第四次调试发现只能同时存在两个灯亮着（包括闪烁的黄灯），于是让两侧黄灯同时闪烁的时候将红绿灯都熄灭。。。第 N 调试终于发现，在 20s 倒计时后，绿灯熄灭，同时让绿灯一侧的黄灯闪烁，对向红灯保持不变，直到五秒倒计时后直接转为绿灯，而黄灯跳为红灯，符合实际情况。

在实现基本交通灯功能后，为了芯片不用一直插拔去烧录 hex 文件，遂设计计时器和交通灯结合程序，通过按键切换，因为前面遇到问题较多，所以此处基本很顺利。完成

## 【心得体会】

通过本次实验，将之前学过的单片机、C语言、嵌入式、PCB等知识运用到实际中，巩固并且为以后的运用铺垫。在硬件的焊制中得到了手工能力的锻炼，在软件的不断开发、调试中自主查阅资料，自主编程，发现错误，修正错误的过程中得到了软件能力的提升，对各种芯片也有了新的认识。