

系统试验

专业：通信工程

学号：1342401024

姓名：唐子文

系统实验报告

一、 设计实现的功能

*时钟（计时器）

1. 状态 1：一只数码管 0-9 计数，一只按键控制开始、结束；
2. 状态 2：二只数码管 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：仅灯亮，数码管不工作；按下键，红、黄、绿三色灯交替亮：红（20s）→黄（闪烁 5s）→绿（20s）→红。
6. 状态 2：灯和数码管相结合，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。

二、 器件的功能及引脚图

实验器件：单片机 AT89S51、集成块 74LS47、LM7805、数码管 4 只、发光二极管 6 只、稳压源、按键 4 个、电阻、排阻、三极管、电容。

1、 单片机 AT89S51

功能：AT89S51 具有如下特点：40 个引脚，4kBytesFlash 片内程序存储器，128bytes 的随机存取数据存储器（RAM），32 个外部双向输入/输出（I/O）口，5 个中断优先级 2 层中断嵌套中断，2 个 16 位可编程定时计数器，2 个全双工串行通信口，看门狗（WDT）电路，片内时钟振荡器。此外，AT89S51 设计和配置了振荡频率可为 0Hz 并可通过软件设置省电模式。空闲模式下，CPU 暂停工作，而 RAM 定时计数器，串行口，外中断系统可继续工作，掉电模式冻结振荡器而保存 RAM 的数据，停止芯片其它功能直至外中断激活或硬件复位。同时该

芯片还具有 PDIP、TQFP 和 PLCC 等三种封装形式，以适应不同产品的需求。

引脚：VCC：供电电压。

GND：接地。

P0 口：P0 口为一个 8 位漏级开路双向 I/O 口，每脚可吸收 8TTL 门电流。当 P1 口的管脚第一次写 1 时，被定义为高阻输入。P0 能够用于外部程序数据存储器，它可以被定义为数据/地址的第八位。在 FIASH 编程时，P0 口作为原码输入口，当 FIASH 进行校验时，P0 输出原码，此时 P0 外部必须被拉高。

P1 口：P1 口是一个内部提供上拉电阻的 8 位双向 I/O 口，P1 口缓冲器能接收输出 4TTL 门电流。P1 口管脚写入 1 后，被内部上拉为高，可用作输入，P1 口被外部下拉为低电平时，将输出电流，这是由于内部上拉的缘故。在 FLASH 编程和校验时，P1 口作为第八位地址接收。

P2 口：P2 口为一个内部上拉电阻的 8 位双向 I/O 口，P2 口缓冲器可接收，输出 4 个 TTL 门电流，当 P2 口被写“1”时，其管脚被内部上拉电阻拉高，且作为输入。并因此作为输入时，P2 口的管脚被外部拉低，将输出电流。这是由于内部上拉的缘故。P2 口当用于外部程序存储器或 16 位地址外部数据存储器进行存取时，P2 口输出地址的高八位。在给出地址“1”时，它利用内部上拉优势，当对外部八位地址数据存储器进行读写时，P2 口输出其特殊功能寄存器的内容。P2 口在 FLASH 编程和校验时接收高八位地址信号和控制信号。

P3 口：P3 口管脚是 8 个带内部上拉电阻的双向 I/O 口，可接收输出 4 个 TTL 门电流。当 P3 口写入“1”后，它们被内部上拉为高电平，并用作输入。作为输入，由于外部下拉为低电平，P3 口将输出电流（ILL）这是由于上拉的缘故。P3.0RXD（串行输入口）P3.1TXD（串行输出口）P3.2/INT0（外部中断 0）P3.3/INT1（外部中断 1）P3.4T0（记时器 0 外部输入）P3.5T1（记时器 1 外部输入）P3.6/WR（外部数据存储器写选通）P3.7/RD（外部数据存储器读选通）P3 口同时为闪烁编程和编程校验接收一些控制信号。

I/O: 口作为输入口时有两种工作方式即所谓的读端口与读引脚读端口时实际上并不从外部读入数据而是把端口锁存器的内容读入到内部总线经过某种运算或变换后再写回到端口锁存器只有读端口时才真正地把外部的数据读入到内部总线上面图中的两个三角形表示的就是输入缓冲器 CPU 将根据不同的指令分别发出读端口或读引脚信号以完成不同的操作这是由硬件自动完成的不需要我们操心 1 然后再实行读引脚操作否则就可能读入出错为什么看上面的图如果不对端口置 1 端口锁存器原来的状态有可能为 0Q 端为 0Q[^] 为 1 加到场效应管栅极的信号为 1 该场效应管就导通对地呈现低阻抗, 此时即使引脚上输入的信号为 1 也会因端口的低阻抗而使信号变低使得外加的 1 信号读入后不一定是 1 若先执行置 1 操作则可以使场效应管截止引脚信号直接加到三态缓冲器中实现正确的读入由于在输入操作时还必须附加一个准备动作所以这类 I/O 口被称为准双向口 89C51 的 P0/P1/P2/P3 口作为输入时都是准双向口接下来让我们再看另一个问题从图中可以看出这四个端口还有一个差别除了 P1 口外 P0/P2/P3 口都还有其他的功能

RST: 复位输入。当振荡器复位器件时, 要保持 RST 脚两个机器周期的高电平时间。

ALE/PROG: 当访问外部存储器时, 地址锁存允许的输出电平用于锁存地址的地位字节。在 FLASH 编程期间, 此引脚用于输入编程脉冲。在平时, ALE 端以不变的频率周期输出正脉冲信号, 此频率为振荡器频率的 1/6。因此它可用作对外部输出的脉冲或用于定时目的。然而要注意的是: 每当用作外部数据存储器时, 将跳过一个 ALE 脉冲。如想禁止 ALE 的输出可在 SFR8EH 地址上置 0。此时, ALE 只有在执行 MOVX, MOVC 指令是 ALE 才起作用。另外, 该引脚被略微拉高。如果微处理器在外部执行状态 ALE 禁止, 置位无效。

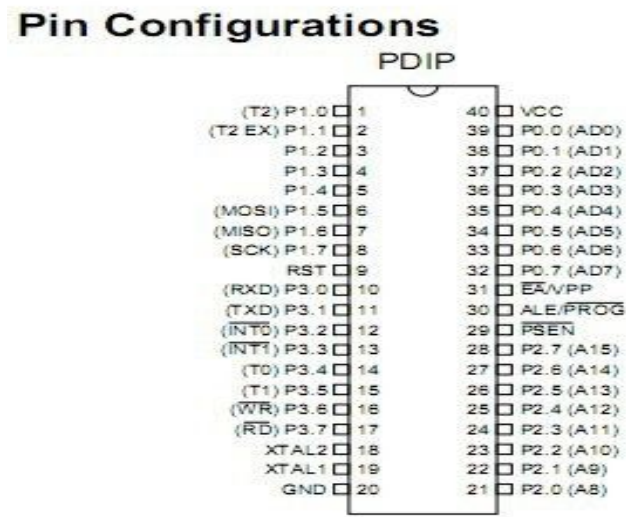
/PSEN: 外部程序存储器的选通信号。在由外部程序存储器取指期间, 每个机器周期两次/PSEN 有效。但在访问外部数据存储器时, 这两次有效的/PSEN 信号将不出现。

/EA/VPP: 当/E_A 保持低电平时, 则在此期间外部程序存储器 (0000H-FFFFH), 不管是否有内部程序存储器。注意加密方式 1 时, /EA 将内部锁定为

RESET；当/EA 端保持高电平时，此间内部程序存储器。在 FLASH 编程期间，此引脚也用于施加 12V 编程电源（VPP）。

XTAL1：反向振荡放大器的输入及内部时钟工作电路的输入。

XTAL2：来自反向振荡器的输出。



AT89S51

2、74LS47

功能：译码器的逻辑功能是将每个输入的二进制代码译成对应的输出的高、低电平信号。常用的译码器电路有二进制译码器、二—十进制译码器和显示译码器。译码为编码的逆过程。它将编码时赋予代码的含义“翻译”过来。实现译码的逻辑电路成为译码器。译码器输出与输入代码有唯一的对应关系。

74LS47 是输出低电平有效的七段字形译码器，它在这里与数码管配合使用，表 2.1 列出了 74LS47 的真值表，表示出了它与数码管之间的关系。

引脚：(1)LT(——)：试灯输入，是为了检查数码管各段是否能正常发光而设置的。当 LT(——)=0 时，无论输入 A3 ， A2 ， A1 ， A0 为何种状态，译码器输出均为低电平，也就是七段将全亮,若驱动的数码管正常，是显示 8。

(2)BI(——)：灭灯输入，是为控制多位数码显示的灭灯所设置的。当 BI(——)=0 时，不论 LT(——)和输入 A3 ， A2 ， A1, A0 为何种状态，译码器输出均为高电平，使共阳极数码管熄灭。

(3)RBI(——): 灭零输入, 它是为使不希望显示的 0 熄灭而设定的。当对每一位 $A_3=A_2=A_1=A_0=0$ 时, 本应显示 0, 但是在 $RBI(——)=0$ 作用下, 使译码器输出全为高电平。其结果和加入灭灯信号的结果一样, 将 0 熄灭。

(4)RBO(——): 灭零输出, 它和灭灯输入 BI(—)共用一端, 两者配合使用, 可以实现多位数码显示的灭零控制



74LS47

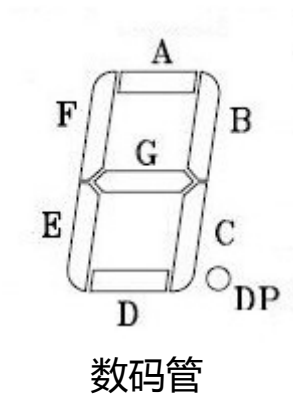
3、LM7805

功能: 将 9V 电压转化为 5V 直流电压

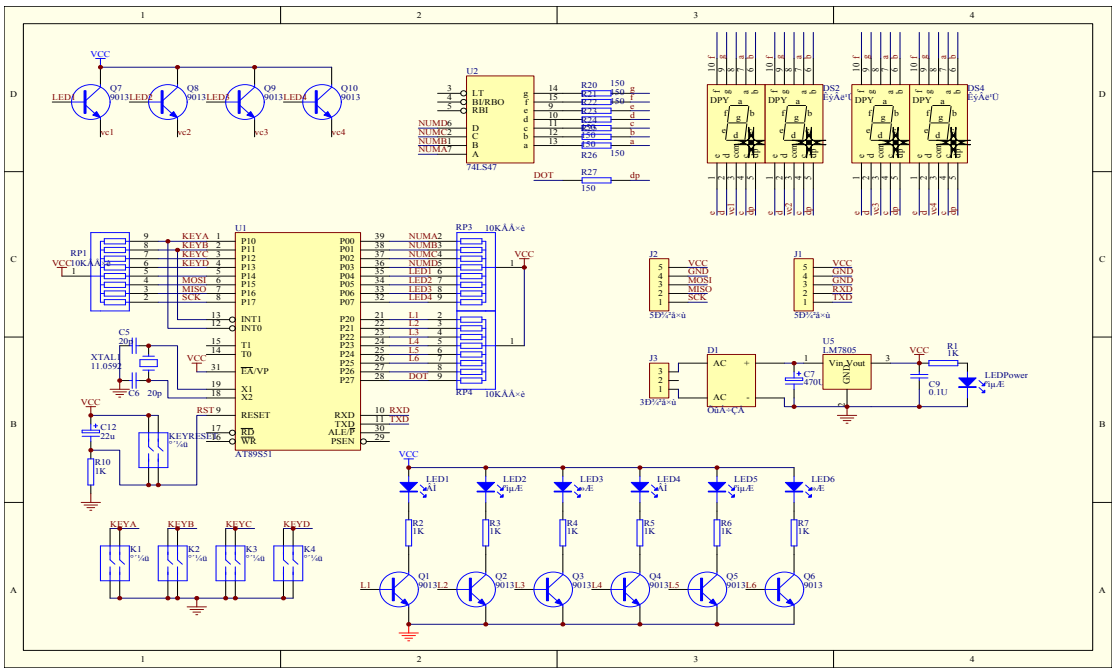
4、数码管

功能: 当数码管特定的段加上电压后, 这些特定的段就会发亮, 以形成我们眼睛看到的字样了。如: 显示一个“2”字, 那么应当是 a 亮 b 亮 g 亮 e 亮 d 亮 f 不亮 c 不亮 dp 不亮。LED 数码管有一般亮和超亮等不同之分, 也有 0.5 寸、1 寸等不同的尺寸。小尺寸数码管的显示笔画常用一个发光二极管组成, 而大尺寸的数码管由二个或多个发光二极管组成, 一般情况下,

单个发光二极管的管压降为 1.8V 左右，电流不超过 30mA。发光二极管的阳极连接到一起连接到电源正极的称为共阳数码管，发光二极管的阴极连接到一起连接到电源负极的称为共阴数码管。常用 LED 数码管显示的数字和字符是 0、1、2、3、4、5、6、7、8、9、A、B、C、D、E、F。

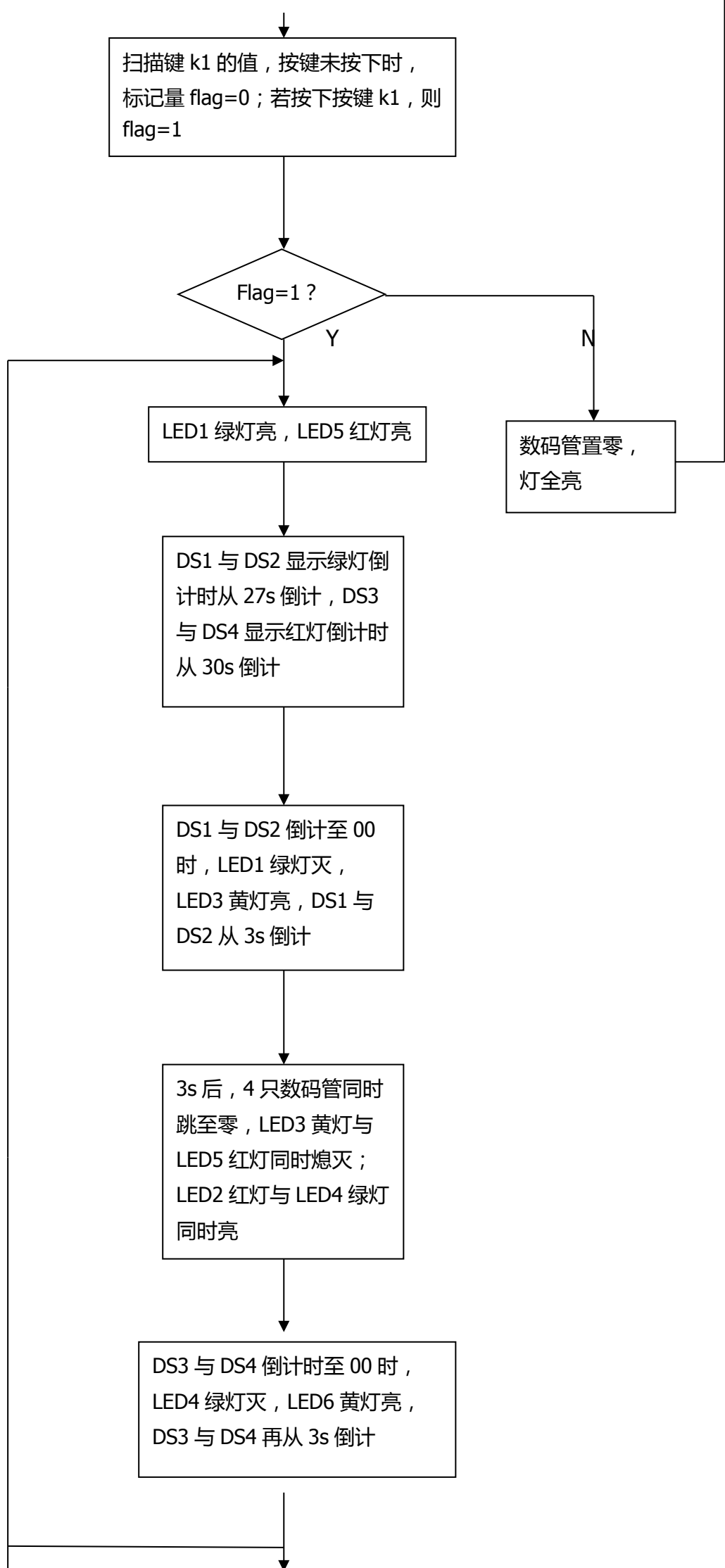


三、 电路原理图



四、 流程图





五、实现功能

K1 键开始运行键，k2 键停止键。灯与数码管相结合，模拟十字路口红绿灯。按下看 k1 开始运行，左边亮绿灯 27s，右边亮红灯 30s，左边 27s 倒计时结束两边数码管变为 3s，左边亮黄灯，右边亮红灯；继续倒计时 3s，左边亮红灯，右边亮绿灯，一直循环，按 k2 停止。

六、调试过程

编写好程序后，编译生成 HEX 文件，选择器件（Device）AT89S51 文件（File）→装入文件（Load）→擦除（Erase）→空检查（Blank check）→编程（program）→校验（Verify）烧录程序。

七、心得小结

本次系统试验让我意识到编程的重要性。进一步巩固了 C 语言知识，锻炼了动手操作能力。

