|  |
| --- |
| 石家庄铁道大学四方学院  课程设计报告 |

时间： 2018年 5 月 1 日至 2018 年 5 月 2 日

课程编号： S08197

课程名称： 单片机综合训练

学生姓名： 于祥麟

学号： 20186204

专业： 电子信息工程

班级： 方1809-1

指导教师： 孙秀婷

高迎霞

|  |
| --- |
| **一、设计任务及要求：**  **设计任务：**  在单片机设计一个节日彩灯控制器，通过四个按钮控制单片机的彩灯的各种变化。  **设计要求：**  1.设计4个按键S0、S1、S2、S3,  S0—开始按此键则灯开始流动(由上而下)；  S1—停止，按此键则停止流动，所有灯为暗；  S2—上，按此键则灯由上向下流动；  S3—下，按此键则灯由下向上流动。  2.由按键控制功能的流水灯,其中的LED采取共阳极接法，通过依次向连接LED的I/O口送出低电平，可实现题目要求的功能。  3. 要求做出实物。 |
| **二、指导教师成绩评定：**  **设计过程(40分)：**  **实物 (40分)：**  **报告 (20分)：** |
| **三、成绩**    指导教师签名：    年 月 日 |

目 录

[第1章　设计目的 1](#_Toc423104728)

[第2章　设计要求 1](#_Toc423104729)

[第3章　硬件电路设计 1](#_Toc423104730)

[3.1　系统结构框图 1](#_Toc423104731)

[3.2　STC89C52单片机 3](#_Toc423104732)

[3.3　总电路图 4](#_Toc423104733)

[第4章　软件程序设计 8](#_Toc423104737)

[4.1　主程序设计 8](#_Toc423104738)

[4.2　LED灯子程序设计 9](#_Toc423104739)

[4.3　延时子程序设计 9](#_Toc423104739)

[4.4　源程序 9](#_Toc423104739)

[第5章　结论 16](#_Toc423104740)

[参考文献 17](#_Toc423104741)

第1章　设计目的

(1)掌握单片机实际系统的开发步骤。

(2)熟悉节日彩灯控制器的工作原理。

(3)加深对单片机汇编语言的认识和理解，并会编程。

(4)学会使用Proteus对单片机进行仿真实验。

(5)学会使用Keil使用C语言对单片机进行编程。

第2章　设计要求

(1)设计4个按键S0、S1、S2、S3，

S0—开始按此键则灯开始流动(由上而下)；

S1—停止，按此键则停止流动，所有灯为暗；

S2—上，按此键则灯由上向下流动；

S3—下，按此键则灯由下向上流动。

(2)由按键控制功能的流水灯,其中的LED采取共阳极接法，通过依次向连接LED的I/O口送出低电平，可实现题目要求的功能。

(3)要求做出实物。

第3章　硬件电路设计

3.1　系统结构框图

节日彩灯控制系统主要由单片机、八个LED彩灯、开关及电源组成。其结构框图如图3-1所示。

八个LED彩灯

单片机

开关

5V电源

图3-1　系统结构框图

3.2　STC89C52单片机

STC89C52是STC公司生产的一种低功耗、高性能CMOS8位微控制器，具有8K字节系统可编程Flash存储器。STC89C52使用经典的MCS-51内核，但是做了很多的改进使得芯片具有传统的51单片机不具备的功能。在单芯片上，拥有灵巧的8 位CPU 和在系统可编程Flash，使得STC89C52为众多嵌入式控制应用系统提供高灵活、超有效的解决方案。

具有以下标准功能： 8k字节Flash，512字节RAM， 32 位I/O 口线，看门狗定时器，内置4KB EEPROM，MAX810复位电路，3个16 位定时器/计数器，4个外部中断，一个7向量4级中断结构（兼容传统51的5向量2级中断结构），全双工串行口。

STC89C52芯片有40个引脚，引脚图如图3-2所示，各引脚功能介绍如下：

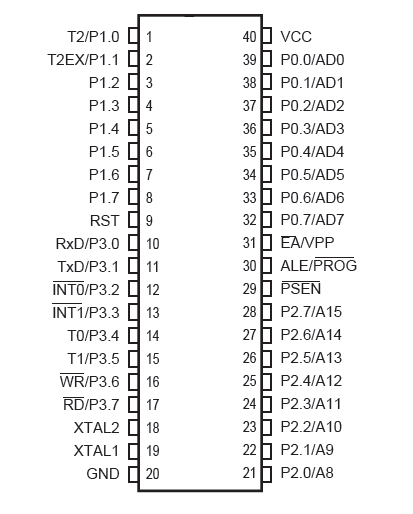


图3-2 STC89C52引脚图

VCC(40引脚)：电源电压

VSS(20引脚)：接地

P0口：P0口是一个8位漏极开路的双向I/O口。作为输出口，每位能驱动8个TTL逻辑电平。对P0端口写“1”时，引脚用作高阻抗输入。当访问外部程序和数据存储器时，P0口也被作为低8位地址/数据复用。在这种模式下，P0具有内部上拉电阻。在flash编程时，P0口也用来接收指令字节;在程序校验时，输出指令字节。程序校验时，需要外部上拉电阻。

P1口：P1口是一个具有内部上拉电阻的8 位双向I/O 口，p1 输出缓冲器能驱动4个TTL逻辑电平。对P1 端口写“1”时，内部上拉电阻把端口拉高，此时可以作为输入口使用。作为输入使用时，被外部拉低的引脚由于内部电阻的原因，将输出电流(IIL)。

此外，P1.0和P1.2分别作定时器/计数器2的外部计数输入(P1.0/T2)和时器/计数器2 的触发输入(P1.1/T2EX)，具体如下表所示。在flash编程和校验时，P1口接收低8位地址字节。

P2口：P2口是一个具有内部上拉电阻的8位双向I/O 口，P2输出缓冲器能驱动4个TTL逻辑电平。对P2端口写“1”时，内部上拉电阻把端口拉高，此时可以作为输入口使用。作为输入使用时，被外部拉低的引脚由于内部电阻的原因，将输出电流(IIL)。

在访问外部程序存储器或用16位地址读取外部数据存储器(例如执行MOVX @DPTR)时，P2口送出高八位地址。在这种应用中，P2口使用很强的内部上拉发送1。在使用8位地址(如MOVX @RI)访问外部数据存储器时，P2口输出P2锁存器的内容。在flash编程和校验时，P2口也接收高8位地址字节和一些控制信号。

P3口：P3口是一个具有内部上拉电阻的8位双向I/O 口，P3输出缓冲器能驱动4个TTL逻辑电平。对P3端口写“1”时，内部上拉电阻把端口拉高，此时可以作为输入口使用。作为输入使用时，被外部拉低的引脚由于内部电阻的原因，将输出电流(IIL)。

P3口亦作为STC89C52特殊功能(第二功能)使用，如下图3-3所示。在flash编程和校验时，P3口也接收一些控制信号。

图3-3 P3口的第二功能

RST——复位输入。当振荡器工作时，RST引脚出现两个机器周期以上高电平将是单片机复位。

ALE/PROG——当访问外部程序存储器或数据存储器时，ALE(地址锁存允许)输出脉冲用于锁存地址的低8位字节。一般情况下，ALE仍以时钟振荡频率的1/6输出固定的脉冲信号，因此它可对外输出时钟或用于定时目的。要注意的是：每当访问外部数据存储器时将跳过一个ALE脉冲。

对FLASH存储器编程期间，该引脚还用于输入编程脉冲(PROG)。

如有必要，可通过对特殊功能寄存器(SFR)区中的8EH单元的D0位置位，可禁止ALE操作。该位置位后，只有一条MOVX和MOVC指令才能将ALE激活。此外，该引脚会被微弱拉高，单片机执行外部程序时，应设置ALE禁止位无效。

PSEN——程序储存允许(PSEN)输出是外部程序存储器的读选通信号，当STC89C52由外部程序存储器取指令(或数据)时，每个机器周期两次PSEN有效，即输出两个脉冲，在此期间，当访问外部数据存储器，将跳过两次PSEN信号。

EA/VPP——外部访问允许，欲使CPU仅访问外部程序存储器(地址为0000H-FFFFH)，EA端必须保持低电平(接地)。需注意的是：如果加密位LB1被编程，复位时内部会锁存EA端状态。如EA端为高电平(接Vcc端)，CPU则执行内部程序存储器的指令。FLASH存储器编程时，该引脚加上+12V的编程允许电源Vpp，当然这必须是该器件是使用12V编程电压Vpp。

3.3　总电路图

在总电路图中，复位电路与晶振电路是单片机最小系统中包括的电路，复位连在AT89C52的9引脚，而晶振电路的两条接线分别与18、19引脚相连。对于本次设计，还需要将8个LED灯接入电阻中，并且分别与8个限流电阻相连保护发光二极管，与单片机的P1口相连。4个按键S0、S1、S2、S3分别连单片机的P3.2、P3.3、P3.4和P3.5由这四个按键来控制灯的启动、停止、由上而下、由下而上。

由按键控制功能的LED工作的方式，其中的LED采取共阳极接法，通过依次向连接LED的I/O口送出低电平,可实现题目要求的功能。

节日彩灯控制器的总电路图如图3-4所示。

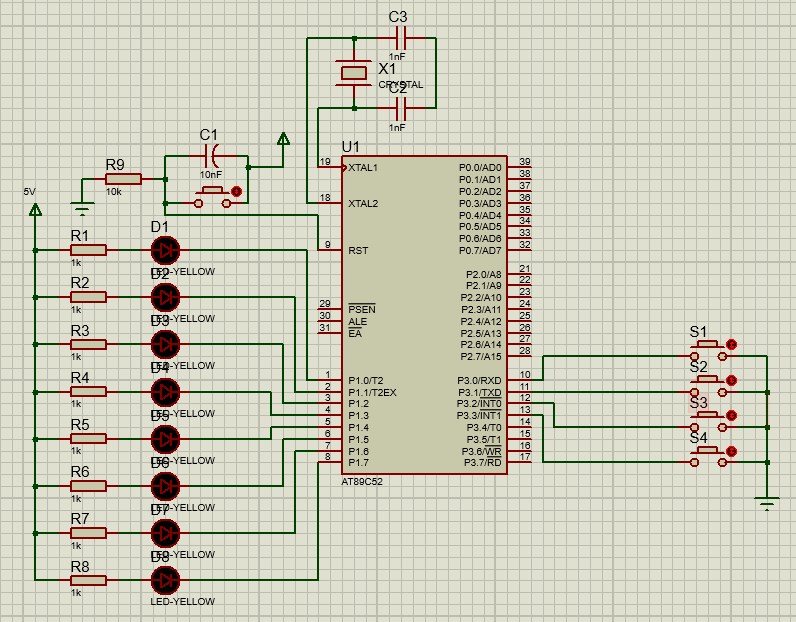


图3-4 总电路图

图3-5　总电路图

第4章　软件设计

4.1　主程序流程图

节日彩灯控制器设计的功能是：当在没有任何按键按下时，8个LED灯均呈现灭灯的状态，并一直在等待按键的按下。当按下开始键S0键后，8个LED灯自上而下循环流动，检测有没有按下停止键即S1，若按下停止键S1则立即停止流动，否则检测是否按下按键S2或者按键S3，若都没有按下则灯继续保持原先的状态流动。当按键S3被按下，灯自下而上流动，同样也要检测是否按下停止键S1，按下则停止，否则灯将继续自下而上流动。同理，当按键S2被按下时，灯自上而下流动，如果按下停止键S1，按下则停止，否则灯将继续自上而下流动。

此程序一共设计了两种灯的流动方式，分别使用两个函数展示。

节日彩灯控制器的主程序流程图如图4-1所示。

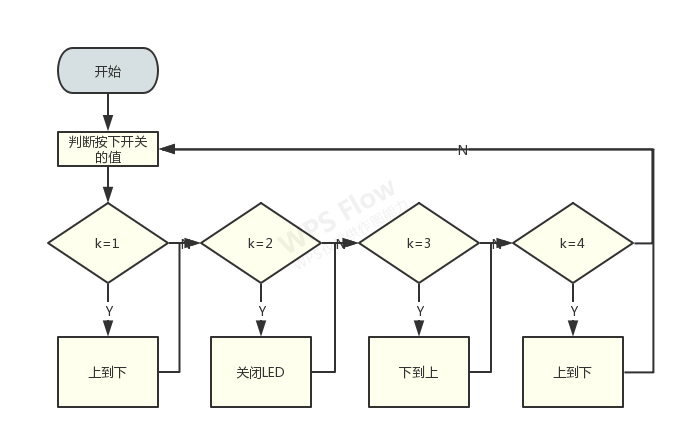
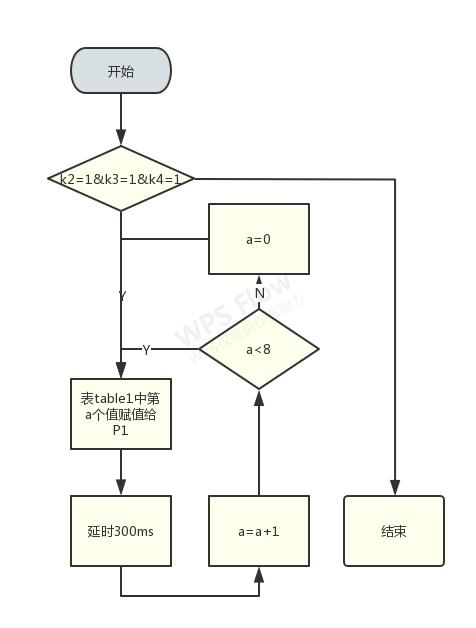
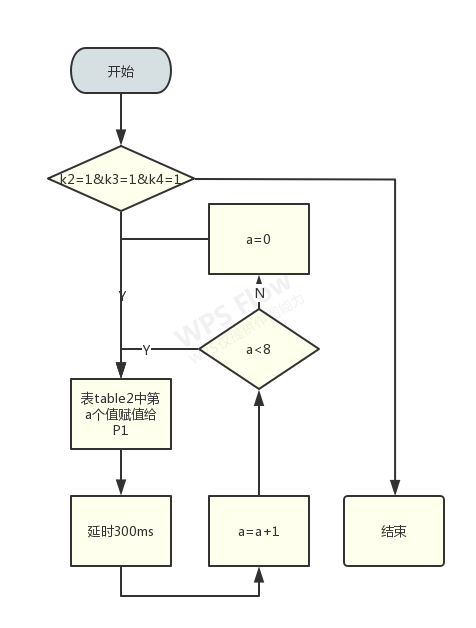


图4-1 主程序流程图

4.2　LED灯子程序流程图

在本设计中的LED灯的流动有两个方向，分别为自上而下和自下而上，首先赋初值，取表中的数，将数值给P1口，并延时输出，满足条件则自动加一或减一。因而得到两个流程图分别为图4-2的a、b。



a、自上而下 b、自下而上

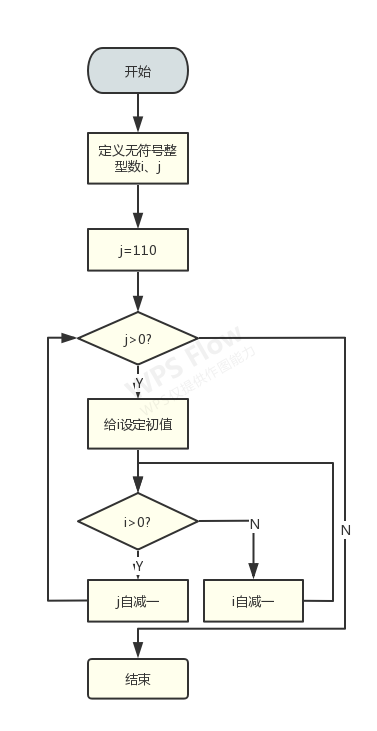
图4-2 LED灯子程序流程图

4.3　延时子程序流程图

对于LED灯循环显示程序来说，由于单片机运行速度非常快，以至于当灯亮时人眼看不见，不方便检测与监控，因此在程序中需要添加延时子程序。

同时延时子程序还可以有效的防止抖动。延时子程序运用了两个for循环嵌套，外循环执行一次，内循环执行j次，因此延时时间为100xms。

延时子程序的流程图如图4-3所示。

图4-3 延时子程序流程图

4.4　源程序

#include <reg52.h>

#define uchar unsigned char //宏定义

#define uint unsigned int

sbit k1 = P3 ^ 2; //位定义

sbit k2 = P3 ^ 3;

sbit k3 = P3 ^ 4;

sbit k4 = P3 ^ 5;

uchar a = 0; //定义所需变量

uchar k = 0; //开关状态

uchar code table1[] = {0xfe, 0xfd, 0xfb, 0xf7, 0xef, 0xdf, 0xbf, 0x7f}; //从上到下流水数组

uchar code table2[] = {0x7f, 0xbf, 0xdf, 0xef, 0xf7, 0xfb, 0xfd, 0xfe}; //从下到上流水数组

void delay\_ms(uint xms) //延时函数

{

uint i, j;

for (i = xms; i > 0; i--)

for (j = 110; j > 0; j--)

;

}

void key() //判断开关状态

{

if (k1 == 0)

{

k = 1;

}

if (k2 == 0)

{

k = 2;

}

if (k3 == 0)

{

k = 3;

}

if (k4 == 0)

{

k = 4;

}

}

void shangdaoxia() //上到下循环点亮

{

while (k2 == 1 && k3 == 1 && k4 == 1)

{

for (a = 0; a < 8; a++) //执行从上到下流水操作

{

P1 = table1[a];

delay\_ms(100);

}

a = 0;

}

}

void xiadaoshang() //下到上循环点亮

{

while (k1 == 1 && k2 == 1 && k4 == 1)

{

for (a = 0; a < 8; a++) //执行从下到上流水操作

{

P1 = table2[a];

delay\_ms(100);

}

a = 0;

}

}

void ledoff() //P1口置1

{

while (k1 == 1 && k3 == 1 && k4 == 1)

{

P1 = 0xff; //P0全部输出高电平，关闭LED

}

}

void main() //主函数

{

while (1)

{

key();

if (k == 1)

{

shangdaoxia();

}

if (k == 2)

{

ledoff();

}

if (k == 3)

{

xiadaoshang();

}

if (k == 4)

{

shangdaoxia();

}

}

}

第5章　结　论

本次课程设计软件程序部分使用了Keil5软件，使用C语言对单片机进行编程，硬件部分使用了AT89C52单片机，利用Proteus8对程序进行仿真，得到了应有的程序结果。在程序上对程序进行了多次修改测试，依据程序的流程图对程序进行调试，使程序符合要求，实现了彩灯的停止，彩灯从下到上依次点亮和彩灯从下到上依次点亮，两种模式之间的切换以及单个模式之间的循环。

本次实验使用C语言对单片机进行编程，相比使用汇编语言，程序的逻辑清晰了很多，相应的编程难度也容易了很多。这次程序使用了主程序和子程序的搭配，子程序有从上到下程序、从下到上程序、检查是否有按键按下程序、延时程序、灭灯程序。主程序使用了一个循环和检查按键按下程序调用其他子程序，实现了要求所需功能。延时子程序使用了两个for循环嵌套实现了毫秒级的延时，彩灯流动程序使用了一个for循环，将预设好的表内的值依次写入到P1口实现彩灯的依次点亮，编程之后利用Keil5软件编译，修改语法上的错误，进行调试。

此次设计过后我对单片机的认识更深了一步，相对之前有了更深的了解，对于Keil5和Proteus8软件的使用也更得心应手。通过这次单片机设计，我不仅加深了对单片机理论的理解，将理论很好地应用到实际当中去，而且我还学会了如何去培养我们的创新精神，从而不断地战胜自己，超越自己。

参考文献

[1] 高峰.单片微型计算机原理与接口技术.科学出版社,2007.

[2] 张志良.单片机原理与控制技术.机械工业出版社,2007.

[3] 谭浩强.C程序设计.北京:清华大学出版社,1991.