

大连理工大学

本科实验报告

题 目： 基于 MINI STM32 门廊监控系统
院 系： 电子信息与电气工程学部
专 业： 集成电路与集成系统
班 级： 电集 1402
学 号： 201484044
姓 名： 崔奥博
同 组 人： 梁轶峤
组 别： 第一组
报告日期： 2017 年 7 月

基于 MINI STM32 单片机的门廊监控系统

一、设计任务与功能

1、设计任务

使用 stm32 单片机开发板，自拟题目完成一个系统。

2、设计功能

测试端功能

- (1) 具有声控灯功能。
- (2) 具有触摸灯功能。
- (3) 具有检测雨量强度的功能。
- (4) 具有检测环境温度功能。
- (5) 具有对检测数据在屏幕上显示的功能。
- (6) 具有遥控器远程控制功能。
- (7) 具有检测是否有人通过门廊功能。
- (8) 具有检测障碍物距离的功能。
- (9) 具有通过蓝牙传输数据并显示的功能。

接收端功能

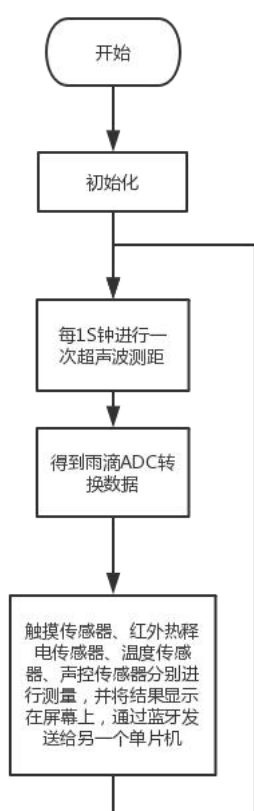
- (1) 遥控器为 1 显示距离情况，当距离小于 5 时进行报警
- (2) 遥控器为 2 显示是否有人
- (3) 遥控器为 3 当温度小于 32 度时接收端显示 warm 同时接收端灯微亮，当温度大于 32 度时接收端显示 hot 同时接收端灯剧烈亮
- (4) 遥控器为 4 显示雨滴湿度情况，当湿度小于 1000 显示 dry，当湿度大于 1000 显示 heavy。
- (5) 遥控器为 5 显示触摸情况，当有人触摸时显示 yes，无人触摸显示 no

二、硬件清单

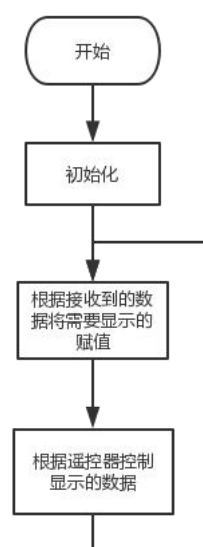
- (1) stm32 mini 开发板以及相应板上资源(液晶屏，AD 转换，时钟输出 PWM 波)
- (2) 雨滴传感器

- (3) DS18B20 温度传感器
- (4) 声控传感器
- (5) HC-SR501 红外热释传感器
- (6) 触摸开关传感器
- (7) 蓝牙模块
- (8) 红外遥控器
- (9) 超声波测距传感器
- (10) 蜂鸣器

三、程序设计框图



发送端程序框图



接收端程序框图

四、程序代码及说明

(1) 发送端主程序初始化代码:

```

u8 t=0;
u8 HC_SR501=0;
u8 touch=0;
u8 song=0;
    
```

```

u16 adcx;
    float distance;
short temperature;
delay_init();          //

uart_init(115200);      //
    Ultran_Init();
LED_Init();
LCD_Init();
Adc_Init();
KEY_Init();
fengming_Init();
song_Init();

HC_SR501_GPIO_Init();
touch_Init();
POINT_COLOR=RED;//

while(DS18B20_Init()) //DS18B20初始化
{
    LCD_ShowString(60,50,200,16,16,"DS18B20 Error");
    delay_ms(200);
    LCD_Fill(60,130,239,130+16,WHITE);
    delay_ms(200);
}
LCD_ShowString(60,50,200,16,16,"DS18B20 OK");
POINT_COLOR=BLUE;//
LCD_ShowString(60,70,200,16,16,"Temp:    . C");
    LCD_ShowString(60,90,200,16,16,"rain:    ");

```

(2) 发送端主程序循环体代码:

```

    if(t==0)//每1秒测量一次超声波测距距离
    {
        Ultra_Ranging(&distance);
    }

```

```
        LCD_ShowString(60,150,200,18,16,"distance:    cm");
LCD_ShowxNum(125,150,distance,4,16,0);
    if(distance<5)
        printf("a");
    else if(5<distance<10)
        printf("b");
}

    adcx=Get_Adc_Average(ADC_Channel_1,10);

HC_SR501=HC_SR501_StateRead();
touch=touch_StateRead();
    song=song_StateRead();

if(HC_SR501==1)
{
    GPIO_SetBits(GPIOA,GPIO_Pin_5);
    delay_ms(5000);
        LCD_ShowString(60,110,200,16,16,"MAN");
    printf("c");
}
else
{
    GPIO_ResetBits(GPIOA,GPIO_Pin_5);
        LCD_ShowString(60,110,200,16,16,"NOMAN");
    printf("d");
}
if(adcx<1000)
{
    LCD_ShowNum(60+40+8,90,adcx,4,16); //显示正数部分
    printf("e");
}
else
```

```
{
    printf("f");
}
if(song==1)
{
    LED0=1;

}
else{
    LED0=0;
    delay_ms(5000);

}
if(touch==1)
{
    LED1=0;
    printf("g");
    LCD_ShowString(60,130,200,16,16,"touch");
}
else{
    LED1=1;
    printf("h");
    LCD_ShowString(60,130,200,16,16,"NOtouch");
}

if(t%10==0)//每100ms读取一次
{
    temperature=DS18B20_Get_Temp();
    if(temperature<0)
    {
        LCD_ShowChar(60+40,70,'-',16,0);           //显示负号
        temperature=-temperature;                   //转为正数
    }else LCD_ShowChar(60+40,70,' ',16,0);          //去掉负号
}
```

```

        LCD_ShowNum(60+40+8,70,temperature/10,2,16);//显示正数部分
        LCD_ShowNum(60+40+32,70,temperature%10,1,16); //显示小数部分

    }
    if(temperature/10<32)
        printf("i");
    else
        printf("j");
        //POINT_COLOR=RED;//设置字体为蓝色
        //LCD_DrawLine(60,180,160,180); //
画线
    }
    delay_ms(10);
    HC_SR501=0;
}

```

(3) 接收端初始化代码

```

    u8 key;
    u16 adcx;
    u8 *str=0;
    u8 *tem=0;
    u8 *man=0;
    u8 *dis=0;
    u8 *rain=0;
    u8 *touch=0;
    float temp;
    float distance;
    Adc_Init();
    uart_init(115200);    //串口初始化为57600
    delay_init();        //延时函数初始化
    LED_Init();          //初始化与LED连接的硬件接口
    TIM1_PWM_Init(899,0);//不分频。PWM频率=72000/(899+1)=80Khz
    LCD_Init();
    Remote_Init();        //红外接收初始化
    LED1=1;

```

```
POINT_COLOR=RED;//设置字体为红色
LCD_ShowString(60,50,200,16,16,"Mini STM32");
LCD_ShowString(60,70,200,16,16,"SYMBOL:");
//显示提示信息
POINT_COLOR=BLUE;//设置字体为蓝色
```

(4) 接收端循环体代码

```
switch(USART_RX_BUF[0])
{
    case 97:{
        dis="danger5 ";
        GPIO_SetBits(GPIOC,GPIO_Pin_0);
        delay_ms(1000);
        GPIO_ResetBits(GPIOC,GPIO_Pin_0);
    }
    break;
    case 98:{
        dis="danger10";
        GPIO_SetBits(GPIOC,GPIO_Pin_0);
        delay_ms(3000);
        GPIO_ResetBits(GPIOC,GPIO_Pin_0);
    }
    break;
    case 99:man="man ";break;
    case 100:man="noman ";break;
    case 101:rain="heavy ";break;
    case 102:rain="dry ";break;
    case 103:touch="yes ";break;
    case 104:touch="no ";break;

    case 105:
        {tem="warm ";
        TIM_SetCompare1(TIM1,50); } }
```



```

        break;
    case 106: {tem="hot    ";
        TIM_SetCompare1(TIM1,800);

    }break;
}

    USART_RX_STA=0;
key=Remote_Scan();
    if(key)
    {
//        LCD_ShowNum(116,130,key,3,16);        //显示键值
        //LCD_ShowNum(116,150,RmtCnt,3,16); //显示按键次数
switch(key)
    {
        case 0:str="ERROR";break;
        case 162:str="POWER";break;
        case 98:str="UP";break;
        case 2:str="PLAY";break;
        case 226:str="ALIENTEK";break;
        case 194:str="RIGHT";break;
        case 34:str="LEFT";break;
        case 224:str="VOL-";break;
        case 168:str="DOWN";break;
        case 144:str="VOL+";break;
        case 104:
            {
                str="1";

LCD_ShowString(60,90,200,16,16,"distance:");
                LCD_ShowString(116,90,200,16,16,dis); //显示
SYMBOL
            if(dis)
                GPIO_SetBits(GPIOC,GPIO_Pin_0);
                delay_ms(1000);
                GPIO_ResetBits(GPIOC,GPIO_Pin_0);

```

```

        }
break;
case 152:{
    str="2";
    LCD_ShowString(60,90,200,16,16,"people:    ");
    LCD_ShowString(116,90,200,16,16,man);    //
显示SYMBOL
    }
break;
case 176:{
    str="3";
    LCD_ShowString(60,90,200,16,16,"tem:        ");
    LCD_ShowString(116,90,200,16,16,tem);//显示SYMBOL
}
break;
case 48:
{
    str="4";
    LCD_ShowString(60,90,200,16,16,"rain:        ");
    LCD_ShowString(116,90,200,16,16,rain);//显示SYMBOL
}
break;
case 24:
{str="5";
    LCD_ShowString(60,90,200,16,16,"touch:        ");
    LCD_ShowString(116,90,200,16,16,touch);    //显示
SYMBOL
}

break;
case 122:str="6";break;

```

```

        case 16:str="7";break;
        case 56:str="8";break;
        case 90:str="9";break;
        case 66:str="0";break;
        case 82:str="DELETE";break;
    }

    LCD_Fill(116,170,116+8*8,170+16,WHITE); //清楚之前的显示
    LCD_ShowString(116,70,200,16,16,str); //显示SYMBOL

}

else delay_ms(10);

}
}

```

(5) 其他子函数都是一些初始化以及延迟函数

总结：发送端主要就是初始化各个模块后依次进行判断，以及在不同的状态下发送不同的字母给串口，串口通过蓝牙将数据传输，其中比较不同的是雨滴传感器既可以用作数字也可以用作模拟，在此我们利用了 ADC 转换模块清晰的显示湿度情况。而接收端则接收不同的字母然后用 CASE 语句进行显示不同的情况，比较不同的是加入了距离报警，和当温度不同时通过输出 PWM 波控制灯的亮度。

五、实验经验总结

- (1) 建立工程时要注意为了今后的框架方便。
- (2) 在传输过程时是传输的ASCII码。
- (3) 注意学会耐心地看手册，机智利用库函数与例程
- (4) 合理分配板上资源，防止接口混用。
- (5) 今后在实现功能和使用传感器前都需要耐心地查找资料和手册。
- (6) 判断时用“==”而不是“=”

六、实验感想

曾经虽然有接触STM32单片机的机会，但没有在短时间内做出一个系统的经历，这也是第一次接触这么多传感器的机会。虽然在逻辑上很简单，但是把他们结合起来也很有成就感，学会了利用已有的资料，在别人的基础上进行改进从而完成指定的功能，这次实验对我的耐心和编写代码实现大系统功能都有了很大的提高。

在此特别感谢王开宇和李月老师的指导。