《智能控制系统》大作业报告

设计题目 基于STM32的农业大棚智能控制系统

小组成员

任课教师 ____________________

完成日期 ____________________

大连理工大学软件学院

目录

1 系统需求

1.1 功能需求

1、监测功能：监测温室的当前状态，包括空气温度、空气湿度、光照度、土壤湿度、等参数等的信息采集以及各个设备的开关状态。

2、设定功能：可以设定各个温室的运行参数，温室内的土壤湿度、时间等参数来自动控制水泵等的目标值，通过空气温度、空气湿度、光照、等参数来自动控制风机、延长光照设备等的目标值和设备的开启/关闭等等。

3、手动控制：可以实现强制手动控制温室内的设备的开关状态。

1.2 运用的元器件

1. TFT-LCD，薄膜晶体管液晶显示器。用于显示各种参数

2.土壤湿度检测器。用于检测土壤的湿度

3.三色灯。用于表示水泵，风扇的设备的工作状态

4.温度/湿度检测器。用于检测空气温度/湿度

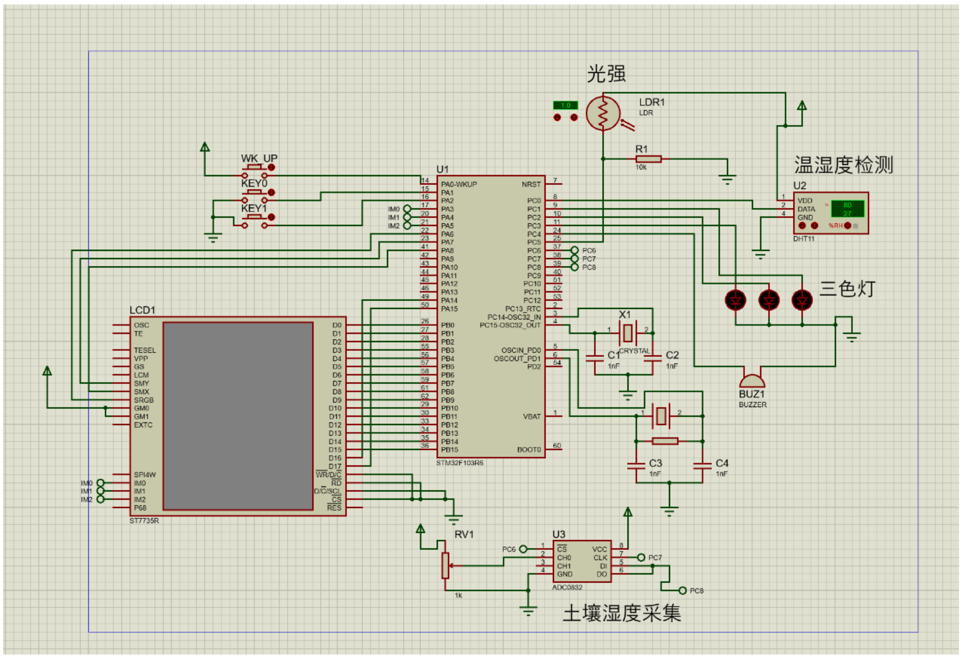
5.光照强度检测。用于检测光照强度

6.蜂鸣器。用于报警

2 系统原理图设计

电路原理图如图所示：

图 1 电路原理图



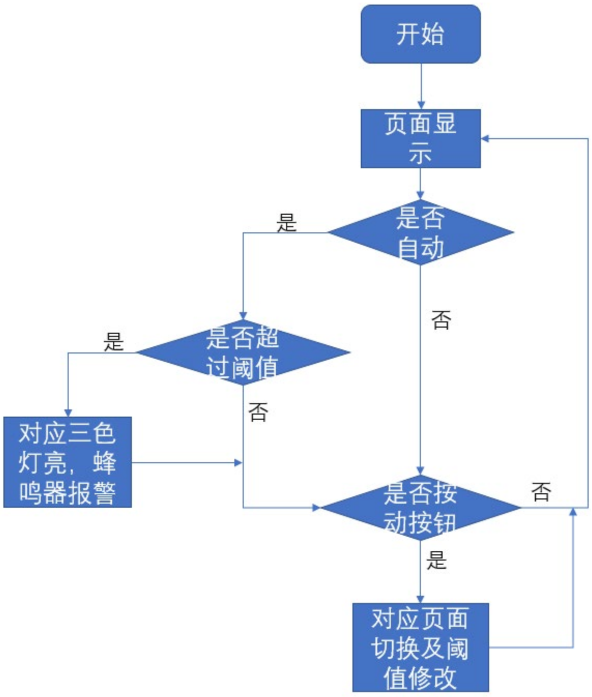
图中三个按钮分别连接到 PA01-PA03 这三个引脚上，用于控制 LCD 屏幕的页面切换以及改变一些阈值。LCD 连接到 PA06-PA08、PA14-PA15、PB0-PB15 这些引脚上，通过这些引脚来控制 LCD 显示。光照强度检测器连接到 PC5 引脚上，用来接收其传入的数据。温湿度检测器连接到 PC0 引脚上，用来接收其检测的数据。三色灯连接到 PC1-PC3 引脚上，通过控制不同颜色的灯来表示水泵、风扇等设备的运行。土壤湿度采集器连接到 PC6-PC8 引脚上，来接收采集到的数据。

3 系统程序设计

3.1 程序设计过程

程序设计流程图如下:

图 2 程序设计流程图



3.2 源代码

Start.c 文件：

```
#include "gpio.h"

#include "stdio.h"

#include "key.h"

#include "lcd.h"

#include "window.h"

#include "update.h"
```

```
#include "get_data.h"
#include "start.h"



struct threshold defa,alter;

struct Status{
    uint8_t Light_status,Water_status,Ven_status;
}status;



int flag_threshold = 0,flag_status = 0,save_threshold = 0;
uint8_t key_1,key_2,key_3;



void init(void)
{
    LCD_Init();
    defa.temp_h = 50,defa.humi_h=100,defa.soil_h=30,defa.light_h=500;
    defa.temp_l = 0,defa.humi_l=0,defa.soil_l=0,defa.light_l=0;

    status.Light_status = 0,status.Water_status = 0,status.Ven_status = 0;
}

void check_BEEP()
{
    uint8_t temp,humi;
    double soil;
    uint32_t light;

    get_temp_humi(&temp,&humi);
    get_soil(&soil);
```

```
        get_light(&light);
        if(temp < defa.temp_h || humi < defa.humi_h || soil < defa.soil_h || light < defa.light_h ||
          temp > defa.temp_l || humi > defa.humi_l || soil > defa.soil_l || light > defa.light_l)
        {
         HAL_GPIO_WritePin(GPIOB,GPIO_PIN_7,GPIO_PIN_SET);
        }
        else HAL_GPIO_WritePin(GPIOB,GPIO_PIN_7,GPIO_PIN_RESET);
}

void start(void)
{
        check_BEEP();
        key_1 = KEY_Scan(0);
        if(key_1 == KEY1_PRES)
        {
         LCD_Clear(WHITE);
         window_main_menu(status.Light_status,status.Water_status,status.Ven_status);
        }
        else if(key_1 == KEY0_PRES)
        {
         flag_status = 0;
         while(1)
         {
                update_status(flag_status%3+1);
                window_main_menu(status.Light_status,status.Water_status,status.Ven_status);
                key_2 = KEY_Scan(0);
                if(key_2 == KEY0_PRES)
                        flag_status++;
                else if(key_2 == KEY1_PRES)
                {
                        update_data_status(&status.Light_status,&status.Water_status,&status.Ven_status,flag_status%3+1);
                }
                else if(key_2 == WKUP_PRES) break;
         }
```

```
}
else if(key_1 == WKUP_PRES)
{
 HAL_GPIO_WritePin(GPIOB,GPIO_PIN_7,GPIO_PIN_RESET);
 LCD_Clear(WHITE);
 window_update_menu(&defa);
 while(1)
 {
        key_2 = KEY_Scan(0);
        if(key_2 == WKUP_PRES)
        {
            LCD_Clear(WHITE);
            window_main_menu(0,0,0);
            break;
        }
        else if(key_2 == KEY0_PRES)
        {
            flag_threshold = 0;
            save_threshold = 0;
            LCD_Clear(WHITE);
            alter = defa;
            while(1)
            {
                update_threshold(flag_threshold%6+1);
                window_update_menu(&alter);
                key_3 = KEY_Scan(1);
                switch(key_3)
                {
                    case KEY0_PRES:
                        flag_threshold++;
                        break;
                    case KEY1_PRES:
                        update_data_threshold(&alter,flag_threshold%10+1,&save_threshold);
                        break;
                    }
                    if(save_threshold == 1)
                    {
                        defa = alter;
                        break;
                    }
                    else if(save_threshold == 2)
                    {
                        window_update_menu(&defa);
                        break;
                    }
                }
            }
    }
    }
}
```

Windows.c 文件：

```
#include "window.h"
#include "lcd.h"
#include "stdio.h"
#include "get_data.h"

void window_main_menu(uint8_t Light_status,uint8_t Water_status,uint8_t Ven_status)
{
    uint8_t str_buff[100];
    char *status = "OFF";
    uint8_t temp=0,humi=0;
    double soil=0;
    uint32_t light=0;

    LCD_ShowString(20,10,220,12,12, "---------------------------------");
    LCD_ShowString(20,30,220,12,12, "|    Vegetable Greenhouse Control |");
    LCD_ShowString(20,50,220,12,12, "---------------------------------");
```

```
get_temp_humi(&temp,&humi);
sprintf((char *)str_buff,       "|     Temperature: %2d                  |",temp);
LCD_ShowString(20,70,220,12,12, str_buff);




sprintf((char *)str_buff,       "|     Humidity: %2d%%                      |",humi);
LCD_ShowString(20,90,220,12,12, str_buff);
get_soil(&soil);
sprintf((char *)str_buff,       "|     Soil Moisture: %3.2lf%%          |",soil);
LCD_ShowString(20,110,220,12,12,str_buff);
get_light(&light);
sprintf((char *)str_buff,       "|     Light Intensity: %4d lx      |",light);
LCD_ShowString(20,130,220,12,12,str_buff);

LCD_ShowString(20,150,220,12,12,"--------------------------------");
LCD_ShowString(20,170,220,12,12,"| Control Status:                    |");

if(Light_status%2 == 0)
{
 status = "OFF";
 led1_off;
}
else
{
 status ="ON";
 led1_on;
}
sprintf((char *)str_buff,       "| 1.Light: %3s                          |",status);
LCD_ShowString(20,190,220,12,12,str_buff);
if(Water_status%2 == 0)
{
 status = "OFF";
 led2_off;
```

```
    }
    else
    {
     status ="ON";
     led2_on;
    }
    sprintf((char *)str_buff,          "| 2.Watering: %3s                    |",status);
    LCD_ShowString(20,210,220,12,12,str_buff);
    if(Ven_status%2 == 0)
    {
     status = "OFF";
     led3_off;
    }
    else
    {
     status ="ON";
     led3_on;
    }
    sprintf((char *)str_buff,          "| 3.Ventilation: %3s                |",status);
    LCD_ShowString(20,230,220,12,12,str_buff);



    LCD_ShowString(20,250,220,12,12,"--------------------------------");
    LCD_ShowString(20,270,220,12,12,"|                    DLUT                    |");
    LCD_ShowString(20,290,220,12,12,"--------------------------------");



}

void window_update_menu(struct threshold *value)
{
    uint8_t str_buff[100];
    LCD_ShowString(20,10,220,12,12, "--------------------------------");
    LCD_ShowString(20,25,220,12,12, "|                                                  |");
LCD_ShowString(20,40,220,12,12, "|        Threshold Adjustment          |");
LCD_ShowString(20,55,220,12,12, "|                                      |");
LCD_ShowString(20,70,220,12,12, "--------------------------------");
LCD_ShowString(20,85,220,12,12, "|                                      |");



sprintf((char *)str_buff,          "|1.Temperature MIN:%2d                 |",value->temp_l);
LCD_ShowString(20,100,220,12,12,str_buff);

sprintf((char *)str_buff,          "|2.Temperature MAX:%2d                 |",value->temp_h);
LCD_ShowString(20,115,220,12,12,str_buff);

sprintf((char *)str_buff,          "|3.Humidity MIN         :%3d%%         |",value->humi_l);
LCD_ShowString(20,130,220,12,12,str_buff);

sprintf((char *)str_buff,          "|4.Humidity MAX         :%3d%%         |",value->humi_h);
LCD_ShowString(20,145,220,12,12,str_buff);

sprintf((char *)str_buff,          "|5.Soil Moisture MIN:%2d%%             |",value->soil_l);
LCD_ShowString(20,160,220,12,12,str_buff);

sprintf((char *)str_buff,          "|6.Soil Moisture MAX:%2d%%             |",value->soil_h);
LCD_ShowString(20,175,220,12,12,str_buff);



sprintf((char *)str_buff,          "|7.Light MIN: %3dlx                    |",value->light_l);
LCD_ShowString(20,190,220,12,12,str_buff);

sprintf((char *)str_buff,          "|8.Light MAX: %3dlx                    |",value->light_h);
LCD_ShowString(20,205,220,12,12,str_buff);

LCD_ShowString(20,220,220,12,12,"--------------------------------");
LCD_ShowString(20,230,220,12,12,"|                                      |");
LCD_ShowString(20,240,220,12,12,"|9. Save and Return                    |");
```

```
    LCD_ShowString(20,255,220,12,12,"|                                  |");
    LCD_ShowString(20,270,220,12,12,"|10. Discard Changes and Return    |");
    LCD_ShowString(20,285,220,12,12,"|                                  |");
    LCD_ShowString(20,300,220,12,12,"--------------------------------");
}
```

Key.c 文件：

```
#include "stdint.h"
#include "key.h"

uint8_t KEY_Scan(uint8_t mode)
{
    static uint8_t key_up=1;//°´¼ü°´ËÉ¿ª±êÖ¾
    if(mode)key_up=1;    //Ö§³ÖÁ¬°´
    if(key_up&&(KEY0==0||KEY1==0||WK_UP==1))
    {
     HAL_Delay(10);//È¥¶¶¶¯
     key_up=0;
     if(KEY0==0)return KEY0_PRES;
     else if(KEY1==0)return KEY1_PRES;
     else if(WK_UP==1)return WKUP_PRES;
    }else if(KEY0==1&&KEY1==1&&WK_UP==0)key_up=1;
    return 3;// ÎÞ°´¼ü°´ÏÂ
}
```

Updata.c 文件：

```
#include "update.h"
#include "lcd.h"
#include "stdio.h"

uint8_t update_light_status(void);
uint8_t update_water_status(void);
uint8_t update_ven_status(void);

void update_threshold(int flag)
{
	if(flag == 1){
	 LCD_Fill(25,100,30,111,YELLOW);
	}
	else if(flag == 2)
	{
	 LCD_Fill(25,130,30,141,YELLOW);
	}
	else if(flag == 3)
	{
	 LCD_Fill(25,160,30,171,YELLOW);
	}
	else if(flag == 4)
	{
	 LCD_Fill(25,190,30,201,YELLOW);
	}
	else if(flag == 5)
	{
	 LCD_Fill(25,240,30,251,YELLOW);
	}
	else if(flag == 6)
	{
	 LCD_Fill(25,270,30,281,YELLOW);

	}
}
void update_status(int flag)
{
	if(flag == 1)
	{
	 LCD_Fill(30,190,35,201,YELLOW);
	}
	else if(flag == 2)
	{
	 LCD_Fill(30,210,35,221,YELLOW);
```

```
    }
    else if(flag == 3)
    {
     LCD_Fill(30,230,35,241,YELLOW);
    }
}

void update_data_threshold(struct threshold *value,int flag,int *save)
{
    switch(flag)
    {
     case 1:
         (value->temp_l)++;
         if((value->temp_l) > (value->temp_h)) (value->temp_l) = 0;
         break;
     case 2:
         (value->temp_h)++;
         if((value->temp_h) > 100) (value->temp_h) = value->temp_l;
         break;
     case 3:
         (value->humi_l)++;
         if((value->humi_l) > (value->humi_h)) (value->humi_l) = 0;
         break;
     case 4:
         (value->humi_h)++;
         if((value->temp_h) > 100) (value->temp_h) = value->humi_l;
         break;
     case 5:
         (value->soil_l)++;
         if((value->soil_l) > (value->soil_h)) (value->soil_l) = 0;
         break;
     case 6:
         (value->soil_h)++;
         if((value->soil_h) > 100) (value->soil_h) = value->soil_l;
```

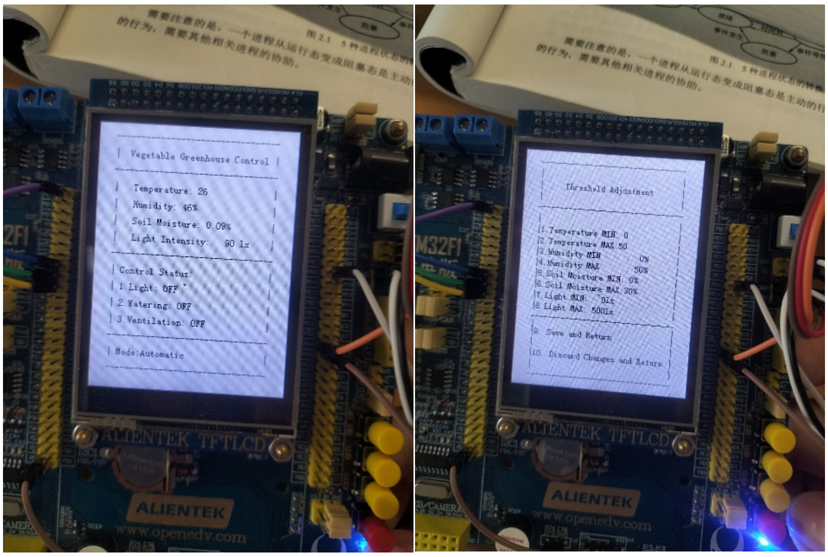
```
            break;
        case 7:
            (value->light_l)++;
            if((value->light_l) > (value->light_h)) (value->light_l) = 0;
            break;
        case 8:
            (value->light_h)++;
            if((value->light_h) > 100) (value->temp_h) = value->light_l;
            break;
        case 9:
            (*save) = 1;
            break;
        case 10:
            (*save) = 2;
            break;
        }
}

void update_data_status(uint8_t *Light_status,uint8_t *Water_status,uint8_t *Ven_status,int flag)
{
        switch(flag)
        {
        case 1:
            (*Light_status)++;
            break;
        case 2:
            (*Water_status)++;
            break;
        case 3:
            (*Ven_status)++;
            break;
        }
}
```

3.3 运行效果

运行效果展示：

图 3 运行效果展示（左图为页面 1，右图为页面 2）



```
-----------------------------------
| Vegetable Greenhouse Control |
-----------------------------------
|  Temperature: 26               |
|  Humidity: 46%                 |
|  Soil Moisture: 0.09%          |
|  Light Intensity:   90 lx      |
-----------------------------------
| Control Status:                |
| 1.Light: OFF                   |
| 2.Watering: OFF                |
| 3.Ventilation: OFF             |
-----------------------------------
| Mode:Automatic                 |
-----------------------------------
```

```
-----------------------------------
|                                 |
|     Threshold Adjustment        |
|                                 |
-----------------------------------
|                                 |
|1.Temperature MIN: 0             |
|2.Temperature MAX:50             |
|3.Humidity MIN       :   0%      |
|4.Humidity MAX       :  50%      |
|5.Soil Moisture MIN: 0%          |
|6.Soil Moisture MAX:30%          |
|7.Light MIN:   0lx               |
|8.Light MAX: 500lx               |
-----------------------------------
|9.  Save and Return              |
|                                 |
|10.  Discard Changes and Return  |
-----------------------------------
```

页面 1 上半页显示了当前空气温湿度及土壤湿度、光照强度等信息，下半页显示了水泵、风扇和灯光等设备的开关信息。页面 2 则显示了温湿度，光照强度，土壤湿度的阈值，并且可以通过按钮对阈值进行更改。

4 设计总结

不足及改进方法：

1. 该系统在功能方面不够全面，还可以加入增加空气湿度的装置，施肥的装置等等。
2. 该系统还有很多和农作物生长相关的条件没有检测到，如土壤温度、土壤的电导率、空气中二氧化碳浓度等。
3. 操作方面不够便捷，可以考虑加入矩阵键盘来简化操作。