自动灌溉系统设计报告

2020/06/29

# 引言

自古以来，我国就是一个以农业为主的国家，即便到了现代社会，农业仍是我国国民经济的基础。但我国农业生产效率低下，长期以来一直以经验种植为主，农业生产效率仅为发达国家的1/10。为解决三农问题，国家正大力发展现代农业，但我国农业管理水平落后，大多控制系统采用定时控制或者手动控制方式。在灌溉管理方面，通常存在浇水不及时、不均、灌水不足或过量灌水等现象。自动灌溉系统通常对作物根系的土壤湿度进行实时监测，获得作物根系的需水量，以此作为自动灌溉的依据。自动灌溉可实现土壤湿度和营养成分的有效管理，是保证设施作物优质高产的重要措施。随着精准感知技术、定量控制技术的迅速发展,自动控制技术在节水灌溉中有了新的发展，通过灌溉控制器适时、适量地灌水，在节省水、人工和提高作物产量方面取得 了一定的成效，可显著提高灌溉精准度，提高水的利用率。

本文设计一种操作简单、精确灌溉的低成本自动化控制灌溉系统，使之既能保证植物的良好的生长状态，又能做到尽量节水，对农业的发展具有重要意义。

# 1本设计的任务和主要内容

在设计过程中，初步掌握AD转换工作原理、控制电路的设计、LCD显示程序、串口通信程序以及控制程序的编写。

本设计主要包括4大部分组成，分别是:单片机最小系统、土壤干湿度检测

模块和LCD显示模块。

其中硬件设计部分包括：

(1)设计单片机与LCD1602的连接电路、

(2)设计单片机与土壤干湿度传感器连接电路

(3)设计光电限位传感器与单片机连接电路。

软件设计部分包括：

(1)显示内容电压值与临界值。通过LCD1602显示屏显示当前干湿度传感

器电压值与设置浇灌临界的电压值;

(2) 设置临界电压值。通过电脑端串口软件向单片机发送字符串指令“SetValue"进入设置浇灌临界值设置，输入“xxx”(其中x代表电压值)，单片机接收设置后反馈信息给电脑，串口软件显示“Set Value:xxxOK." (其中x替换成输入的电压值数字，例输入“1.28”， 返回应答字符串"Set Value: I .28 OK.");

(3)立即浇灌和停止浇灌控制。通过电脑端串口软件向单片机发送字符串指

令“Start”,继电器动作，电磁阀通电开始浇水，并且单片机反馈应答信息给电脑，串口软件显示“Operating";通过电脑端串口软件向单片机发送字符串指令“Stop”， 电磁阀停止工作，并且单片机反馈应答信息给电脑，串口软件显示“Stoped."

# 2本系统主要硬件电路设计及介绍

## 2.1本系统主要硬件介绍及总体说明

本设计主要任务是设计完成一个自动灌溉系统，使用到的主要硬件包括土壤干湿度传感器、STC12C5A60S2单片机、LCD1602液晶显示屏、单路继电器等。系统功能实现流程：通过土壤干湿度传感器循环检测土壤湿度情况，采集到的土壤湿度数据通过STC12C5A60S2单片机自带的AD转换功能将其转换为电压值，之后LCD1602显示AD转换的电压值和预先设定的浇灌电压值，并且单片机可以通过串口接收命令，根据不同命令执行相应灌溉操作。系统总体框图如下。

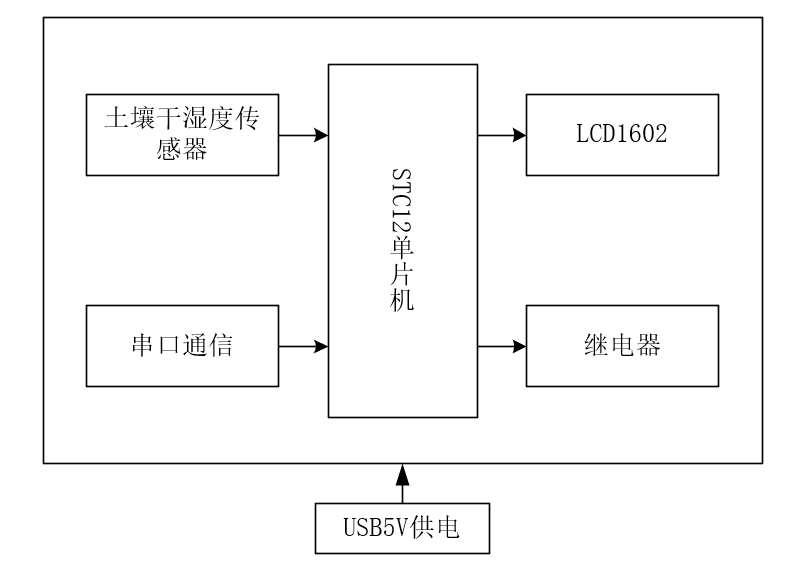


图 1系统总体框图

## 2.2简介STC12C5A60S2单片机

STC12C5A60S2/AD/PWM系列单片机是宏晶科技生产的单时钟/机器周期(1T)的单片机，是高速/低功耗/超强抗干扰的新一代8051单片机，指令代码完全兼容传统8051,但速度快8-12倍。内部集成MAX810专用复位电路,2路PWM,8路高速10位A/D转换(250K/S),针对电机控制，强干扰场合。

本系统的设计主要用到了STC12C5A60S2单片机的AD转换功能和串口通信功能以及LCD1602和继电器的驱动控制。STC12C5A60S2单片机实物图和原理图如下。

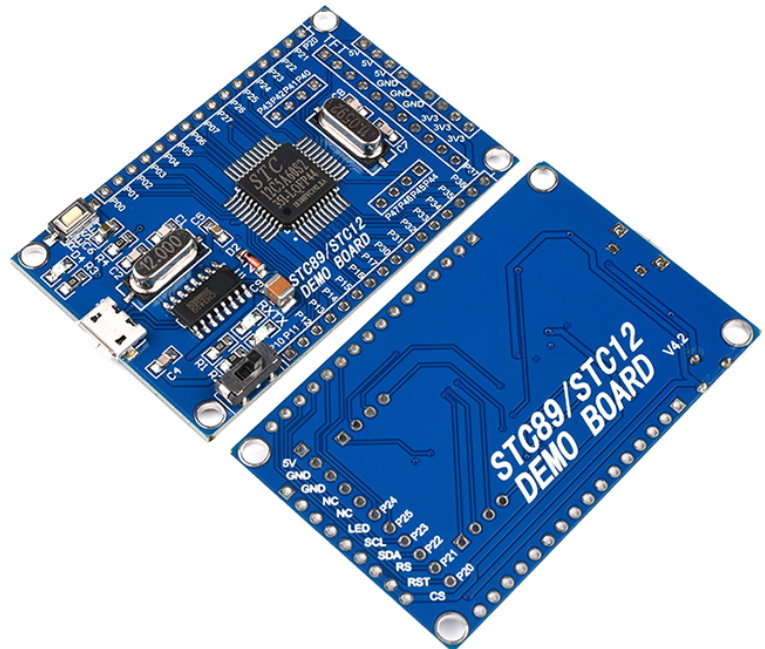


图 2单片机实物图

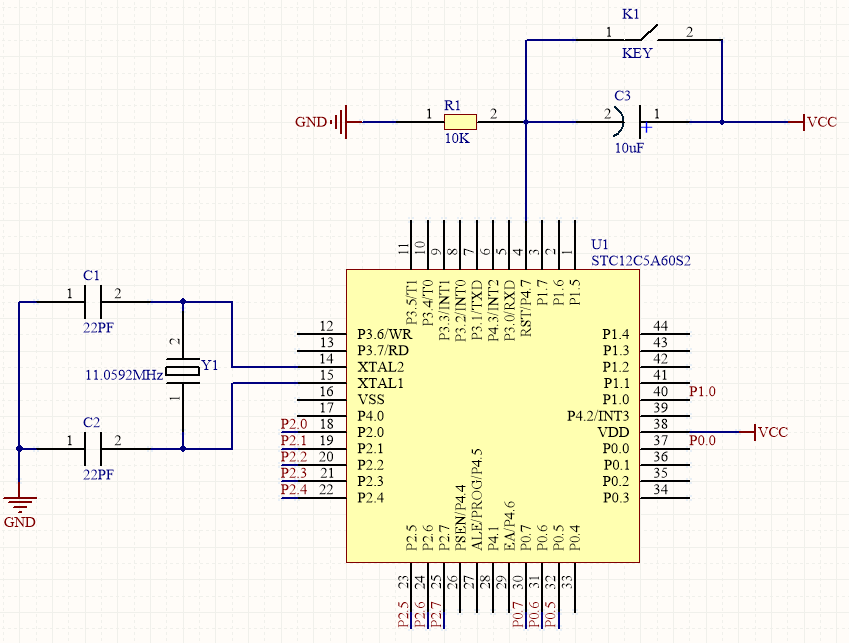


图 3单片机原理图

## 2.3土壤干湿度传感器模块介绍

采用电容式土壤湿度检测传感器模块，输出量为模拟量。

工作电压3.3~5.5V;输出电压:0~3.0V;湿度大时电压小，湿度小时电压大。土壤干湿度传感器实物图和原理图如下。

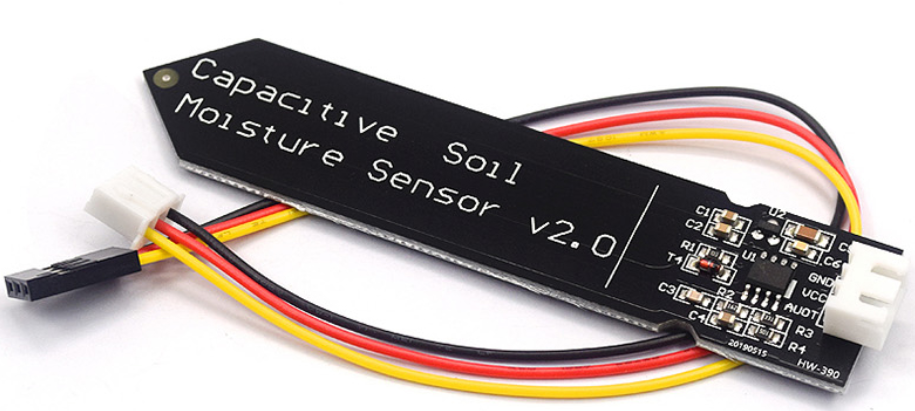


图 4土壤湿度传感器实物图

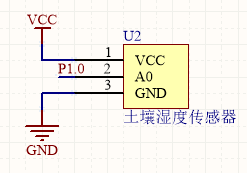


图 5土壤湿度传感器原理图

## 2.4 LCD1602液晶显示模块介绍

LCD1602是一种工业字符型液晶，能够同时显示16x02即32个字符。LCD1602液晶显示原理 LCD1602液晶显示的原理是利用液晶的物理特性，通过电压对其显示区域进行控制，有电就有显示，这样即可以显示出图形。

1602液晶也叫1602字符型液晶，它是一种专门用来显示字母、数字、符号等的点阵型液晶模块。它由若干个5X7或者5X11等点阵字符位组成，每个点阵字符位都可以显示一个字符，每位之间有一个点距的间隔，每行之间也有间隔，起到了字符间距和行间距的作用。

正因为如此所以它不能很好地显示图形（用自定义CGRAM，显示效果也不好）。1602LCD是指显示的内容为16X2，即可以显示两行，每行16个字符液晶模块（显示字符和数字）。LCD1602液晶显示模块实物图和原理图如下。

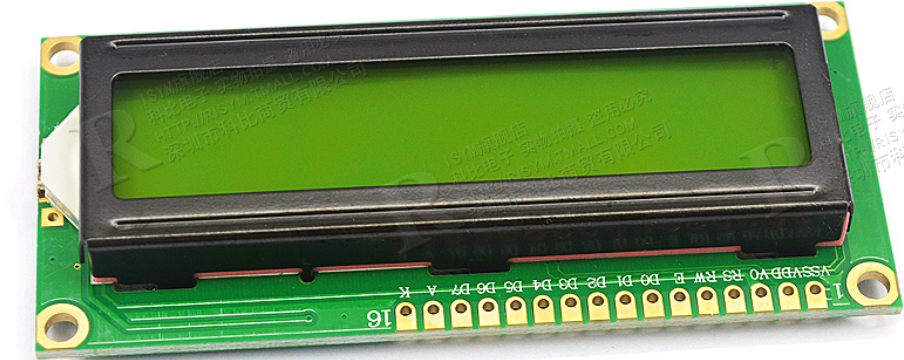


图 6 LCD1602实物图

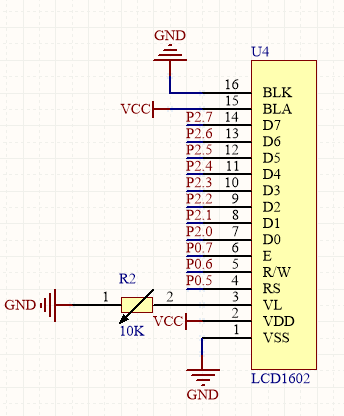


图 7 LCD1602原理图

## 2.5继电器模块介绍

单片机无法直接驱动电磁阀工作，故需用使用继电器模块来控制电磁阀工作，单路光耦继电器转接模块实物图和原理图如图所示。

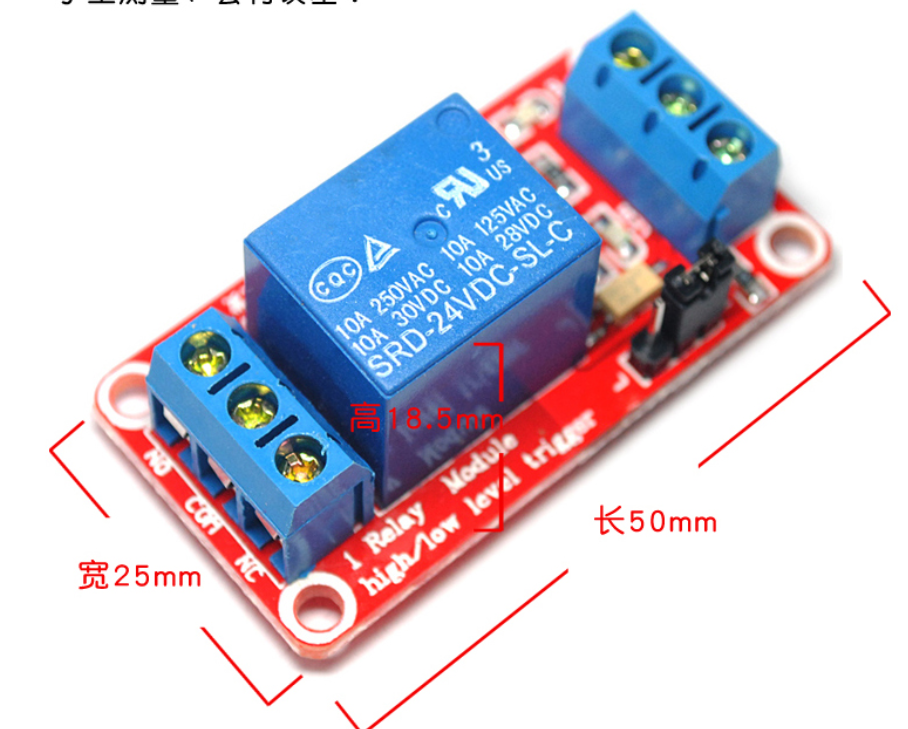


图 8继电器实物图

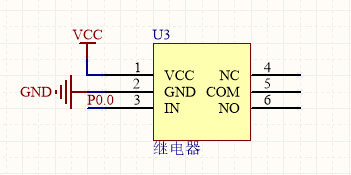


图 9继电器原理图

# 3程序介绍

软件设计思路：通过土壤干湿度传感器循环检测土壤湿度情况，采集到的土壤湿度数据通过STC12C5A60S2单片机自带的AD转换功能将其转换为电压值，之后LCD1602显示AD转换的电压值和预先设定的浇灌电压值，并且单片机可以通过串口接收命令，根据不同命令执行相应灌溉操作。软件设计流程图如下。

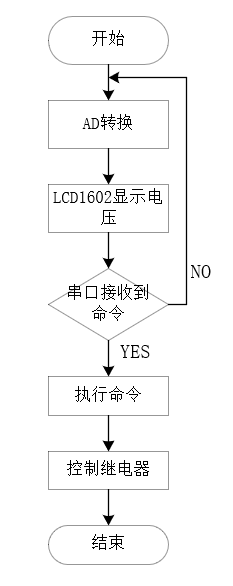


图 10软件设计流程图

## 3.1 AD转换模块 STC12C5A60S2使用的部分程序

void ADC\_Init(unsigned char port)

{

P1ASF=port;//设置AD转换通道

ADC\_RES=0;//清空转换结果

ADC\_CONTR=ADC\_POWER | ADC\_SPEEDLL\_540;//打开AD转化器电源

// IE=0XA0;//开启总中断，ADC中断

\_nop\_();

\_nop\_();

\_nop\_();

\_nop\_();

}

float GetADCResult(unsigned char channel)//读取通道ch的电压值

{

unsigned int ADC\_RESULT = 0;//用来存放结果

float result;

ADC\_CONTR = ADC\_POWER | ADC\_SPEEDLL\_540 | ADC\_START | channel;//开始转换，并设置测量通道为P1^0

\_nop\_();//需经过四个CPU时钟延时，上述值才能保证被设进ADC\_CONTR控制寄存器

\_nop\_();

\_nop\_();

\_nop\_();

while(!(ADC\_CONTR & ADC\_FLAG));//等待转换结束

ADC\_CONTR &= ~ADC\_FLAG;//软件清除中断控制位

ADC\_RESULT = ADC\_RES;

ADC\_RESULT = (ADC\_RESULT << 2) | (0x02 & ADC\_RESL); //默认数据存储方式：高八位在ADC\_RES,低二位在ADC\_RESL低二位

result = ADC\_RESULT \* 5.0 / 1024.0 ; //基准电压为电源电压5V，10的分辨率，即1024

return result;

}

## 3.2单片机的主函数

void main()

{

LCD\_1602\_Init(); //液晶显示前进行初始化

ADC\_Init(ADC\_PORT0); //配置通道P1^0为AD采集口，如要配置其他其他口，直接用或运算加进初始化函数中

USART\_Init();

moto=1;

res1=1.28;

while(1)

{

res0=GetADCResult(ADC\_CH0);

Display\_Now\_U();

Display\_Set\_U();

if(flag==1)

{

flag=0;

a=Analysis(data\_buffer,table0,8) ;

b=Analysis(data\_buffer,table2,5) ;

c=Analysis(data\_buffer,table4,4) ;

if(a==0)

{

table6[10]=data\_buffer[9];

table6[12]=data\_buffer[10];

table6[13]=data\_buffer[11];

Send\_String(table6,14);

res1=data\_buffer[9]+0.1\*data\_buffer[10]+0.01\*data\_buffer[11];

a=1;

}

if(b==0)

{

Send\_String(table3,10);

moto=0;

b=1;

}

if(c==0)

{

Send\_String(table5,7);

moto=1;

c=1;

}

}

Delay\_ms(10);

}

}

## 3.3 LCD显示函数

void Display\_Now\_U()

{

Write\_1602\_String("NOW:U=",0x80);

Write\_1602\_Data(0x30 + (uint)res0%10);

Write\_1602\_Data('.');

Write\_1602\_Data(0x30 + (uint)(res0\*10)%10);

Write\_1602\_Data(0x30 + (uint)(res0\*100)%10);

Write\_1602\_Data('V');

}

void Display\_Set\_U()

{

Write\_1602\_String("SET:U=",0x80+0x40);

Write\_1602\_Data(0x30 + (uint)res1%10);

Write\_1602\_Data('.');

Write\_1602\_Data(0x30 + (uint)(res1\*10)%10);

Write\_1602\_Data(0x30 + (uint)(res1\*100)%10);

Write\_1602\_Data('V');

}

# 4 调试

## 4.1软件调试

使用keil开发环境编写程序，经过仔细修改编译，未发现逻辑错误，编译后程序无错误无警告。

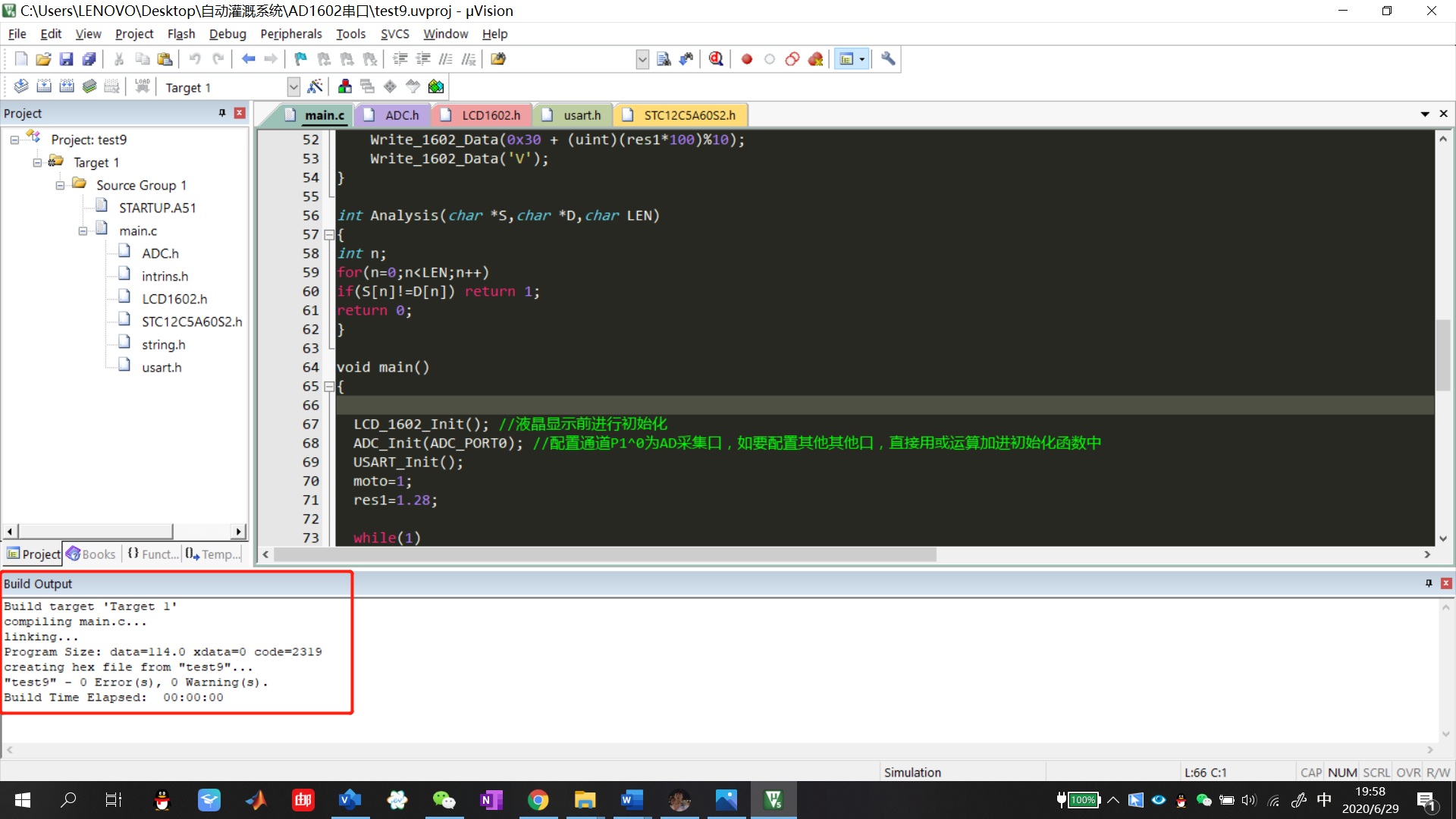


图 11软件调试图

## 4.2硬件调试

程序设计每完成一部分就会执行相应的硬件调试，经测试AD转换、LCD显示、串口接收、继电器驱动等都无问题。

# 5结论

在经过多次测试调试后，可得到如下结果：

1. 土壤湿度传感器通过AD转换后能准确测量出土壤的湿度电压值。
2. LCD1602能够准确显示出AD检测的电压，和设定电压值。
3. 单片机能够根据串口接受的不同命令执行不同操作。
4. 继电器能够准确的根据命令进行工作。

# 谢辞

在系统的设计过程中，老师严谨的治学态度、丰富渊博的知识、精益求精的工作态度以及侮人不倦的师者风范是我终生学习的楷模,老师的高深精湛的造诣与严谨求实的治学精神, 将永远激励着我。在此,谨向老师致以衷心的感谢和崇高的敬意!

另外,感谢学校给予这样一次机会,能够独立地完成一个课程设计，并在这个过程当中，给予我们各种方便,使我们在这学期快要结束的时候,能够将学到的知识应用到实践中，增强了我们实践操作和动手应用能力,提高了独立思考的能力。

在这次课程设计的撰写中,我得到了许多人的帮助。首先我要感谢我的老师在课程设计上给予我的指导、提供给我的支持和帮助，这是我能顺利完成这次设计的主要原因，更重要的 是老师帮我解决了许多技术上的难题, 让我能把系统做得更加完善。在此期间,我不仅学到了许多新的知识，而且也开阔了视野，提高了自己的设计能力。其次， 我要 感谢帮助过我的同学,他们也为我解决了不少我不太明白的设计上的难题。最后再一次感谢所有在设计中帮助过我的良师益友和同学

# 参考文献

[1]宋凤娟,付侃,薛雅丽.STC12C5A60S2单片机高速A/D转换方法[J].煤矿机械,2010,31(06):219-221.

[2]杨平,王威.MSP430系列超低功耗单片机及应用[J].国外电子测量技术,2008,27(12):48-50.

[3]刘志平,赵国良.基于nRF24L01的近距离无线数据传输[J].应用科技,2008(03):55-58.

[4]王海宁. 基于单片机的温度控制系统的研究[D].合肥工业大学,2008.

[5]赵娜,赵刚,于珍珠,郭守清.基于51单片机的温度测量系统[J].微计算机信息,2007(02):146-148.

[6]冯显英,葛荣雨.基于数字温湿度传感器SHT11的温湿度测控系统[J].自动化仪表,2006(01):59-61.

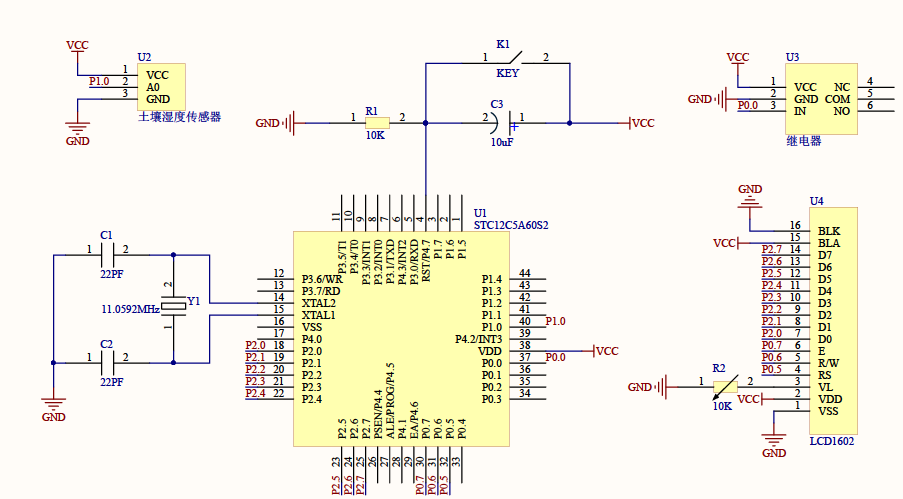
[7]曹宇,魏丰,胡士毅.用51单片机控制RTL8019AS实现以太网通讯[J].电子技术应用,2003(01):21-23.

[8]苏建徽. 光伏水泵系统及其控制的研究[D].合肥工业大学,2003.

[9]马云峰.单片机与数字温度传感器DS18B20的接口设计[J].计算机自动测量与控制,2002(04):278-280.

[10]金伟正.单线数字温度传感器的原理与应用[J].电子技术应用,2000(06):66-68.

# 附录



#include <STC12C5A60S2.h>

#include <LCD1602.h>

#include <ADC.h>

#include <usart.h>

#include <string.h>

#include <intrins.h>

#define uchar unsigned char

#define uint unsigned int

#define FSCLK 11059200

//#define FSCLK 12000000

sbit moto=P0^0;

float res0,res1;

uchar a,b,c;

uchar data\_buffer[20],data\_length = 0,flag = 0;

uchar table0[8]="SetValue";

uchar table1[16]={'S','e','t','V','a','l','u','e',':','x','.','x','x','O','K','.'};

uchar table2[5]="Start";

uchar table3[10]="Operating.";

uchar table4[4]="Stop";

uchar table5[7]="Stoped.";

uchar table6[14]="PleaseSetValue";

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* 函数功能 ：延时函数

\*\* 函数说明 ：利用软件延时，占用CPU，经调试最小单位大约为1ms

\*\* 入口参数 ：time:需要延时的时间，单位ms

\*\* 出口参数 ：无

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void Delay\_ms(uint time)

{

uint i,j;

for(i = 0;i < time;i ++)

for(j = 0;j < 930;j ++);

}

void Display\_Now\_U()

{

Write\_1602\_String("NOW:U=",0x80);

Write\_1602\_Data(0x30 + (uint)res0%10);

Write\_1602\_Data('.');

Write\_1602\_Data(0x30 + (uint)(res0\*10)%10);

Write\_1602\_Data(0x30 + (uint)(res0\*100)%10);

Write\_1602\_Data('V');

}

void Display\_Set\_U()

{

Write\_1602\_String("SET:U=",0x80+0x40);

Write\_1602\_Data(0x30 + (uint)res1%10);

Write\_1602\_Data('.');

Write\_1602\_Data(0x30 + (uint)(res1\*10)%10);

Write\_1602\_Data(0x30 + (uint)(res1\*100)%10);

Write\_1602\_Data('V');

}

int Analysis(char \*S,char \*D,char LEN)

{

int n;

for(n=0;n<LEN;n++)

if(S[n]!=D[n]) return 1;

return 0;

}

void main()

{

LCD\_1602\_Init(); //液晶显示前进行初始化

ADC\_Init(ADC\_PORT0); //配置通道P1^0为AD采集口，如要配置其他其他口，直接用或运算加进初始化函数中

USART\_Init();

moto=1;

res1=1.28;

while(1)

{

res0=GetADCResult(ADC\_CH0);

Display\_Now\_U();

Display\_Set\_U();

if(flag==1)

{

flag=0;

a=Analysis(data\_buffer,table0,8) ;

b=Analysis(data\_buffer,table2,5) ;

c=Analysis(data\_buffer,table4,4) ;

if(a==0)

{

table6[10]=data\_buffer[9];

table6[12]=data\_buffer[10];

table6[13]=data\_buffer[11];

Send\_String(table6,14);

res1=data\_buffer[9]+0.1\*data\_buffer[10]+0.01\*data\_buffer[11];

a=1;

}

if(b==0)

{

Send\_String(table3,10);

moto=0;

b=1;

}

if(c==0)

{

Send\_String(table5,7);

moto=1;

c=1;

}

}

Delay\_ms(10);

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* (C) \*\*\*\*\*END OF FILE\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/