**一、设计任务**

本系统满足室内环境变量实行全面、实时、长期监测的要求，实现室内环境温湿度、可燃气体浓度检测的自动化和智能化。系统以单片机为核心，以温度、湿度传感器，气敏传感器作为测量元件，通过单片机与智能传感器相连，采集并存储智能传感器的测量数据，经过分析处理将结果显示于LCD液晶屏。在单片机系统中，还要实现超限报警和数据辅助存储功能。

**二、方案设计**

**2.1硬件设计**

在室内环境监测硬件设计上，由单片机（AT89C52）控制整个系统的运作，MQ211气敏传感器模块实现监测室内可燃气体功能、SHT11温湿度传感器模块实现监测室内的温度和湿度功能、按键模块实现设置报警上限功能、LCD液晶模块实现显示功能、蜂鸣器报警功能。这六大模块组成的原理图来实现家庭环境检测系统的各个功能。

在该设计中，选用了AT89C52单片机作为控制芯片。该芯片有丰富的内部资源，丰富的I/O接口，低电压，低功耗等优点，并且内置看门狗电路，支持串口程序烧录，使用方便快捷，可以进行C语言程序编写，易于实现。温湿度测量方面选用瑞士SHT11芯片，该芯片内置A/D转换芯片，管脚接线简单，测量精度高等优点，气敏传感器使用多气体测量传感器MQ211，其具有多种可燃气体的测试功能，简单高效。A/D转换模块选用ADC0831，具有接口电路简单，成本低等优点，该芯片为一路八位数转换芯片需求。环境监测系统硬件结构图如图1.

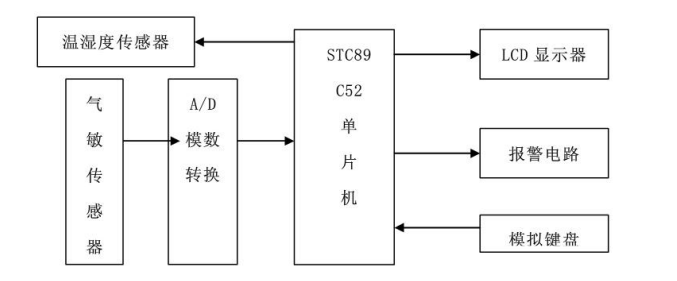


图1 系统硬件结构图

2.1.1芯片管脚连接

在该设计电路中，用单片机I/O口中的P1口作为LCD液晶屏的数据口，采用并口数据传输模式，P2口中P2.0、P2.1、P2.2作为控制信号输出口，分别接LCD的RS、R/W、E控制端；P2.3和P2.4分别接温湿度传感器SHT11的SCK和DATA，P2.5、P2.6和P2.7接按键电路。P1口的P1.1、P1.2分别接ADC0831的控制端，P1.4接报警器的蜂鸣器。

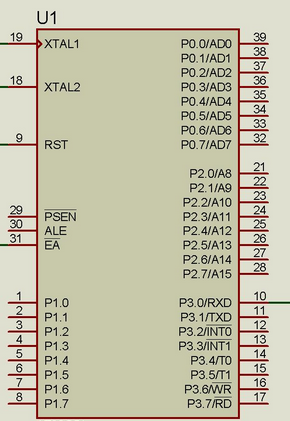


图2 STC89C52管脚分布图

2.1.2 晶振和复位电路

时钟电路用于产生时钟信号，时钟信号是单片机内部各种微操作的时间基准。复位对单片机来说，是程序还没有开始执行，是在做准备工作。晶振电路和复位电路见图3.

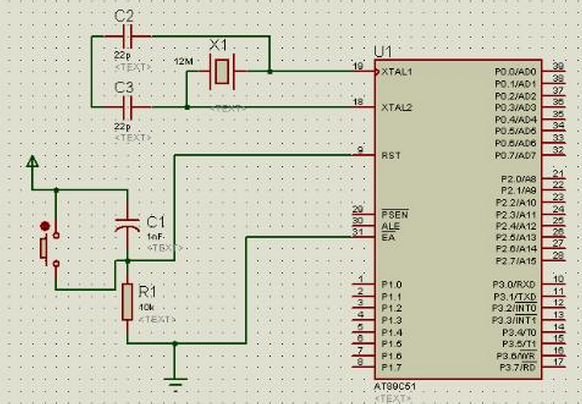


图3 晶振和复位电路

2.1.3 气敏传感器

MQ211型气敏元件是一种通用性较强的气敏元件，气敏传感器的敏感元件是气敏电阻器，气敏电阻器的阻值随吸附气体的浓度变化而变化，利用者一特性检测气体浓度。气敏传感器的连接如图4.

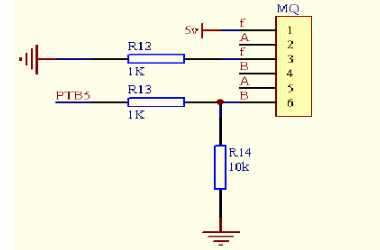


图4 气敏传感器的连接

2.1.4 温湿度传感器

传感器SHT11将温度感测、湿度感测、信号变换、A/D转换和加热器等功能集成到一个芯片上，其内部结构如图5所示。该芯片包括一个电容性聚合体湿度敏感元件和一个用能隙材料制成的温度敏感元件。这两个敏感元件分别将湿度和温度转换成电信号，该电信号首先进入微弱信号放大器进行放大，然后进入一个14位的A/D转换器，最后经过二线串行数字接口输出数字信号。

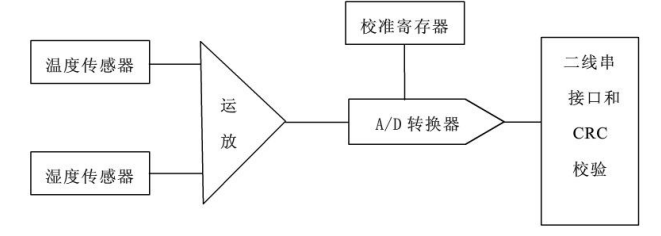


图5 温湿度传感器内部结构

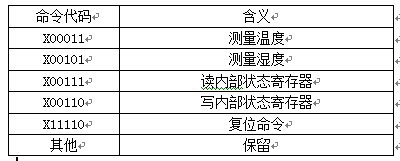


图6 温湿度传感器指令代码

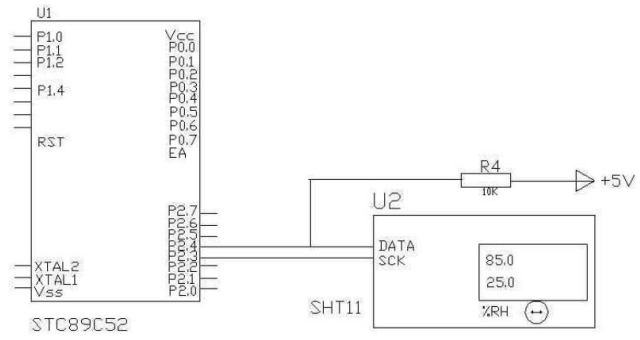
**

图7 温湿度传感器电路

2.1.5 LCD显示模块设计

在该设计中，LCD要显示温度、湿度、可燃气体是否超标的功能，所以使用了一块行列点阵数为128\*64，每行可以显示8个字符，共显示2行字的LCD。



图8 LCD管脚名称和功能

LCD的连接电路：当LCD选择并口时，要把8个数据送到芯片，而LCD选择串口时，只需要把3个数据送到芯片。由于芯片的管脚丰富够用，所以LCD的连接方式选择并口。LCD的连接方式如图9所示。

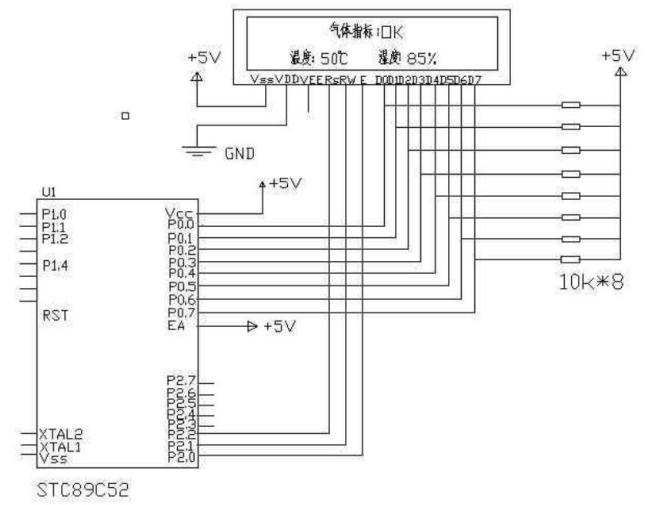


图9 LCD连接电路

2.1.6 按键电路块

图10是按键输入的典型电路，按键S1、S2、S3断开时，端口P2.5、P2.6、P2.7上的信号为高电平，按键S1、S2、S3合上时，端口P2.5、P2.6、P2.7上的信号为低电平。电阻起到分压、限流作用。当按键电路设计成这种硬件连接时，按键可以采取中断的形式进行读取，也可以采用循环扫描的形式读取，为程序的编写提供了选择的余地。

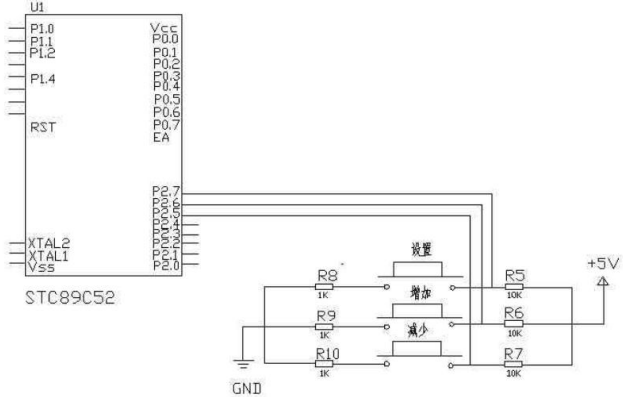


图10按键电路

2.1.7 报警电路设计

图11，在本系统中，采用P1.4口为报警电路输出口，当有报警信号时，蜂鸣器鸣叫，提醒安全指标超标或不正常，须采取安全措施改变室内环境质量，如配合智能建筑系统，可实现自动化无人操作，方便又安全。

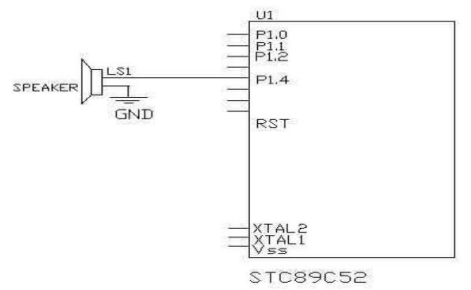


图11 报警电路

**2.2 软件设计**

在上述硬件基础上，编写系统程序，实现以下几项功能：

1. 通电时蜂鸣器响一声，LCD液晶显示室内温度、湿度、可燃气体浓度状态。
2. 当室内温度大于设定的温度、湿度（例如温度28度、湿度80%）、检测到可燃气体超标时，蜂鸣器报警。
3. 当第一次按设置键：温度值闪烁，这是可以通过按“+”或者“—”按键设定温度上限值，4秒钟无键按下停止闪烁，按“+”或者“—”设置温度下限值；第二次按设置按键：湿度值闪烁，这时可以通过按“+”或者“—”设置湿度上限值，4秒钟无键按下停止闪烁，按“+”或“—”设置湿度下限值；第三次按设置按键，退出设置状态。

2.2.1 软件系统设计

系统的软件设计采用C语言编程，软件结构为模块化结构，由主程序、自检程序、SHT11数据采集与处理程序、ADC0831数据采集处理程序、报警控制程序及显示程序等模块组成，其软件模块结构如图12所示。

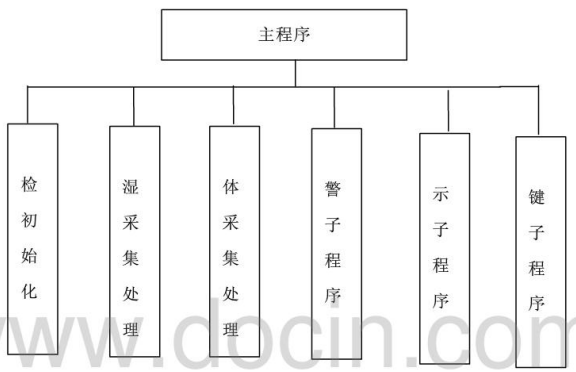
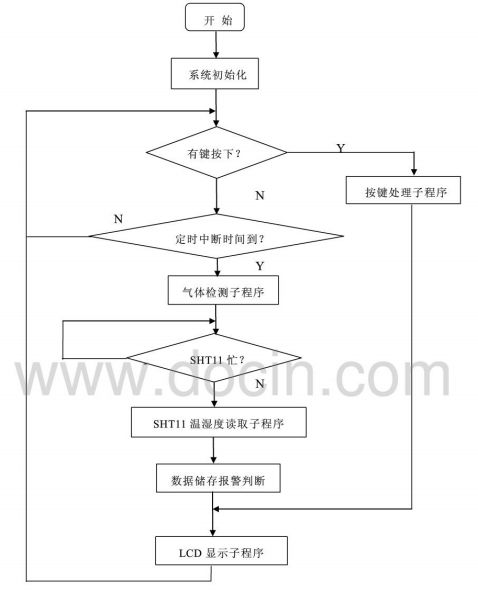
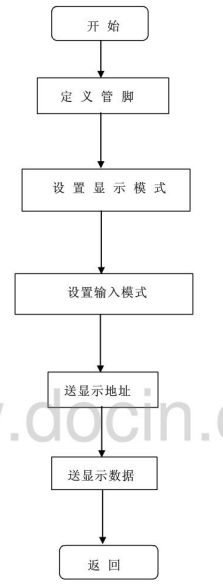


图12 软件结构方框图

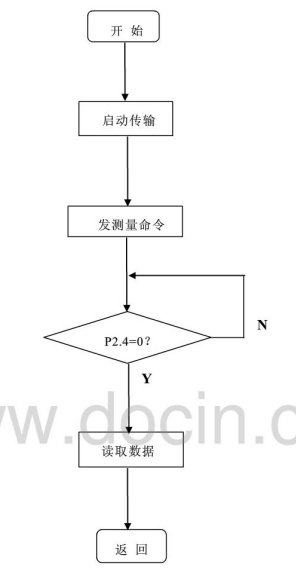
2.2.2主程序设计



2.2.3 LCD1602显示模块程序设计



2.2.4 SHT11温湿度测量模块程序



**三、程序代码**

#include<REG51.H>

#include<intrins.h>

#include<LCD1602.h>

#include<sht10.h>

#include<adc0831.h>

#define uchar unsigned char

#define uint unsigned int

Unsigned char TempBuffer[5];

Uchar humibuffer[4];

Uchar qitibuffer[3];

Void main wendu\_to\_str()

{ int i;

int TempBuffer[5];

for(i=0;i<5;i++)

Tempbutter[i]=i;

Tempbutter[0]=(SHUO[xts\_zi]);

Tempbutter[1]=(SHUO[xtg\_zi]);

Tempbutter[2]=0xdf;

Tempbutter[3]=’C’;

Tempbutter[4]=’\0’;

Printf(“%d”,TempBuffer[i]);

Printf(“\n”);

}

Void main shidu\_to\_str()

{int i;

for(i=0;i<4;i++)

humibuffer[i]=i;

humibuffer[0]=(SHUO[xzz\_zi]);

humibuffer[1]=(SHUO[xsg\_zi]);

humibuffer[2]=’%’;

humibuffer[3]=’\0’;

printf(“\n”);

}

Void main qiti\_to\_str()

{int I;

for(i=0;i<3;i++)

qitibuffer[0]=b1;

qitibuffer[1]=b2;

qitibuffer[2]=b3;

printf(“%d,”qitibuffer[i]);

printf(“\n”);

}

Void show\_LCD()

{

LCD\_Initial();

text\_jishuan\_temp11();

data\_pro();

wendu\_to\_str();

Goto XY(6,1);

Print(TempBuffer);

Goto XY(0,1);

Print(“wendu”);

shidu\_to\_str();

GotoXY(11,1);

Print(“shidu”);

GotoXY(17,1);

Print(humibuffer);

GotoXY(13,0);

Print(qitibuffer);

}

csh()

{

P2.0=0;

}

main()

{

csh();

show\_LCD();

delay\_n10us(100000);

}

LCD显示屏驱动程序

1602.h

#ifndef LCD\_CHAR\_1602\_2005\_4\_9

#define LCD\_CHAR\_1602\_2005\_4\_9

#include<intrins.h>

sbit LcdRs=P2^0;

sbit LcdRw=P2^1;

sbit LcdEn=P2^2;

sfr DBPort=0x80;

unsigned char LCD\_Wait(void)

{

LcdRs=0;

LcdRw=1;\_nop\_();

LcdEn=1;\_nop\_();

//while(DBPort&0x80);

LcdEn=0;

return DBPort;

}

//向LCD写入命令或数据

#define LCD\_COMMAND

1. //Command

#define LCD DATA 1 //DATA

#define LCD\_CLEAR\_SCREEN 0x01 //清屏

#define LCD\_HOMING 0x02 //光标返回原点

Void LCD\_Write(bit style,unsigned char input)

{

LcdEn=0;

LcdRs=style;

LcdRw=0; \_nop\_();

DBPort=input; \_nop\_();//注意顺序

LcdEn=1; \_nop\_();//注意顺序

LcdEn=0; \_nop\_();

LCD\_wait();

}

//设置显示模式

#define LCD\_SHOW 0x04 //显示开

#define LCD\_HIDE 0x00 //显示关

#define LCD\_CURSOR 0x02 //显示光标

#define LCD\_NO\_CURSOR 0x00 //无光标

#define LCD\_FLASH 0x01 //光标闪动

#define LCD\_NO\_FLASH 0x00 //光标不闪动

Void LCD\_SetDisplay(unsigned char DisplayMode)

{

LCD\_Write(LCD\_COMMAND,0x08|DisplayMode);

}

//设置输入模式

#define LCD\_AC\_UP 0x02

#define LCD\_AC\_DOWN 0x00 //default

#define LCD\_MOVE 0x01 //画面可移动

#define LCD\_NO\_MOVE 0x00 //default

Void LCD\_SetInput(unsigned char InputMode)

{

LCD\_Write(LCD\_COMMAND,0x04|InputMode);

}

//移动光标或屏幕

#define LCD\_CURSOR 0x02

#define LCD\_SCREEN 0x08

#define LCD\_LEFT 0x00

#define LCD\_RIGHT 0x04

Void LCD\_Move(unsigned char object,unsigned char direction)

{

if(object==LCD\_CURSOR)

LCD\_Write(LCD\_COMMAND,0x10|diriction);

if(object==LCD\_SCREEN)

LCD\_Write(LCD\_COMMAND,0x18|diriction);

}

//初始化LCD

Void LCD\_Initial()

{

LcdEn=0;

LCD\_Write(LCD\_COMMAND,0x38);

LCD\_Write(LCD\_COMMAND,0x38);

LCD\_SetDisplay(LCD\_SHOW|LCD\_NO\_CURSOR);

LCD\_Write(LCD\_COMMAND,LCD\_CLEAR\_SCREEN);

LCD\_SetInput(LCD\_AC\_UP|LCD\_NO\_MOVE);

}

Void GotoXY(unsigned char x,unsigned char y)

{

if(y==0)

LCD\_Write(LCD\_COMMAND,0x80|x);

if(y==1)

LCD\_Write(LCD\_COMMAND,0x80|(x-0x40));

}

Void Print(unsigned char \*str)

{

while(\*str!=’\0’)

{

LCD\_Write(LCD\_DATA,\*str);

str++;

}

}

Void LCD\_LoadChar(unsigned char user[8],unsigned char place)

{

Unsigned char i;

LCD\_Write(LCD\_COMMAND,0x40|(place\*8));

for(i=0;i<8;i++)

LCD\_Write(LCD\_DATA,user[i]);

}

#endif

温湿度传感器驱动程序

SHT11.h

#include<reg51.h> //头文件

#include<intrins.h>

#include<stdio.h>

#include<math.h> //Keil library

sbit DATA=P2^4; //数据

sbit SCK=P2^3; //时钟

#define TEMP\_ML 0x03 //000 0001 1 温度命令

#define HUMI\_ML 0x05 //000 0010 1 湿度命令

unsigned char error; //全局错误变量

unsigned char ack; //全局错误变量

//float temp\_zi; //全局应答变量

//float humi\_zi; //全局应答变量

unsigned char temp\_h; //全局应答变量

unsigned char temp\_LL; //全局应答变量

unsigned char humi\_h;

unsigned char temp\_LL

#define uchar unsigned char //定义一下方便使用

#define unit unsigned int

#define ulong unsigned long

unsigned int recs=0;//接收次数

uchar TempBuffer[5];uchar humibuffer[4];

const unsigned char SHUO[15]=”0123456789.%RH”;

int xts\_zi=0;//温度值

int xtg\_zi=0;//温度个位值

int xtd\_zi=0;//温度点值

int xss\_zi=0;//湿度值

int xsg\_zi=0;//湿度个位值

int xsd\_zi=0;//湿度点值

sbit soud=P1^4;//声音

//基本驱动程序

//系统初始化

Void start(void) //启动

{

DATA=1; SCK=0; //数据为1，SCK=0

\_nop\_();

SCK=1; //第一个脉冲

DATA=0; //数据跌落

\_nop\_();

SCK=0; //完成一个脉冲

\_nop\_(); \_nop\_(); \_nop\_();

SCK=1; //再一个脉冲

\_nop\_();

DATA=1; //数据变为1

\_nop\_();

SCK=0; //变为该脉冲

}

Void sht\_rest(void) //复位

{

unsigned char I;

DATA=1;SCK=0; //数据为1时钟为0

for（i=0;i<9;i++） //9个脉冲为复位

{SCK=1;

SCK=0;

}

Start（）; //启动

}

char write(unsigned char value)//写一个字节 返回应答信号

{

Unsigned char I;

ack=0;

for(i=0x80;i>0;i/=2)  //释放数据总线

{if(i&value)DATA=1; //写入值

Else DATA=0;

SCK=1; //上升沿写入

\_nop\_(); \_nop\_(); \_nop\_(); //延时

SCK=0;

}DATA=1; //释放数据总线

SCK=1;

return ack; //error=1表示没有应答

}

char read()//读一个字节返回应答信号

{

Unsigned char I,val=0;

temp\_LL=0;

temp\_h=0;

DATA=1; //释放数据总线

for（i=0x80;i>0;i/=2） //位移8位

{SCK=1; //上升沿读入

if(DATA)val=(val|i) //确定值

SCK=0;

}

DATA=0; //读应答信号应答为1，应答为0通过CPU下拉为应答

SCK=1; 第9个脉冲

\_nop\_(); \_nop\_(); \_nop\_(); //pulswith approx.5us

SCK=0;

DATA=1; //释放数据总线

temp\_h=val;

val=0;

//低8位

DATA=1; //释放数据总线

for(i=0x80;i>0;i/=2) //位移8位

{SCK=1; //上升沿读入

if(DATA)val=(val|i) //确定值

SCK=0;

}

DATA=1;//0; //不需要应答 通过CPU下拉为应答

SCK=1; 第9个脉冲

\_nop\_(); \_nop\_(); \_nop\_(); //pulswith approx.5us

SCK=0;

DATA=1; //释放数据总线

temp\_LL=val

return val;

}

//测量温度或者是湿度，返回校验值

text\_a(unsigned char ml)

{

unsigned int I;

start(); //启动

write(ml); //写入测温度

if（ack==1）

{

sht\_rest(); //复位

write(ml); //写入测温度

}

//DATA=1;// 释放数据总线

For(i=0;i<65535;i++)if(DATA==0)break;

read();//度温度

}

text\_jishuan\_temp()

{

float aa=0,bb=0,temp zi;

int abcd=0;

aa=(float)temp\_h\*256+(float)temp\_LL;

temp\_zi=0.01\*aa-40;

if (temp\_zi<0)

{

temp\_zi= -temp\_zi;

}

temp\_zi= temp\_zi\*10;

abcd=(int) temp\_zi;

xts\_zi=abcd/100;

abcd=abcd%100;

xtg\_zi=abcd/10;

abcd=abcd%10;

xtd\_zi=abcd/1;

}

text\_jishuan\_humi()

{

float aa=0,bb=0,humi\_zi;

int abcd=0;

aa=(float)temp\_h\*256+(float)temp\_LL;

baa\*aa\*2.8/1000000;

aa=0.0405\*aa;

aa=aa-4-bb;

humi\_zi= humi\_zi\*10;

abcd=(int)humi\_zi;

xss\_zi=abcd/100;

abcd=abcd%100;

xsg\_zi=abcd/10;

abcd=abcd%10;

xsd\_zi=abcd/1;

}

text\_jishuan\_temp11()

{

error=0;

ack=0;

sht\_rest(); //复位

text\_a(TEMP\_ML);

text\_jishuan\_temp();

text\_a(HUMI\_ML);

text\_jishuan\_humi();

}

unsigned int wendu,shidu;

void delay\_n10us(unit n) //延时n个10us@12M晶振

{

uint i;

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

{

\_nop\_(); \_nop\_(); \_nop\_(); \_nop\_(); \_nop\_(); \_nop\_();

}

**四、心得体会**

通过本次课程设计的学习与实践，我逐渐对单片机产生了兴趣，并且对温度气敏传感器、LED显示器都有了一定的了解，在设计过程中通过图书馆、网上查阅资料，自己的专业知识也有了一定的提高，相信这次设计对我会有很大的帮助。

在本系统中，采用模块化、层次化设计。单片机监测信息的采集、报警和处理，能够迅速、全面准确地获取系统的各种特征量，完成室内环境温湿度参数、有害气体浓度参数的实时监测。同时LCD液晶屏显示得到的数据，供人们浏览、查询和进行控制，满足了系统监测的要求。本系统的优良功能可实现对室内环境温湿度的全面、实时、自动的监测，及时发现问题，采取有效防护措施，减少或避免灾难性事故的发生。

**五、参考文献**

【1】李朝青.单片机原理及接口技术.北京：北京航空航天大学出版社，2006年版

【2】杨素行.模拟电子技术基础简明教程（第三版）.北京：高等教育出版社，2006年版

【3】徐安，陈耀，李锐华，郭其一.微型计算机控制技术.北京：电子工艺出版社，2008版

【4】高吉祥 全国大学生电子设计竞赛培训系列教程模拟电子线路设计 .北京 ：电子工业出版,2007版

【5】张毅刚 单片机原理及应用 .北京 ：高等教育出版社, 2003.

【6】郭天祥 新概念单片机语言教程—入门 、提高 、开发拓展全攻略 [M].北京 ：电子工业出版社,2009

【7】刘海成 单片机及应用系统设计原理与实践测试结果 北京 ：北京航空航天大学出版社,2009. 【8】于勇 ，戴佳 ，长江.51 单片机 C 语言常用模块与 综合系统设计[M].北京 ：电子工业出版社,2007. 【9】高吉祥 全国大学生电子设计竞赛培训系列教测试结果程模拟电子线路设计 [M].北京 ：电子工业出版社 ，2007.

【10】翟玉文 电子设计与实践 北京 ：北京中国电力出版社2005.

【11】肖忠祥 数据采集原理 西安 西北工业出版社 ，2002.

【12】谭浩强 程序设计 （第三版 ） 北京 清华大学出版社,2005.

【13】黄智伟 .全国大学生电子设计竞赛训练教程测试结果北京 ：电子工业出版社 2005.

【14】陈杰 黄鸿 传感器与检测技术 北京 高等教育出版社,2008.

【15】李广第.单片机基础.北京航空航天出版社.2001

返回

送显示数据

送显示地址

设置输入模式

设置显示模式

定义管脚

开始

返回

读取数据

开始

P2.4=0？

发测量命令

启动传输