

**本科生毕业论文(设计)**

|  |  |
| --- | --- |
| **题 目** | 基于单片机的智能火灾预警装置的设计 |
| 与实现 | |
| **院 系** | 计算机与信息工程学院 |
|  | |
| **班 级** | 物联142 |
|  |  |
| **姓 名** | 刘 玲 |
|  | |
| **学 号** 220143047 | |
|  | |
| **答辩时间** 2018年5月 | |
|  | |
| **新疆农业大学计算机与信息工程学院** | |

[1 前言 4](#_Toc2040651434)

[1.1背景及意义 4](#_Toc1975960378)

[1.2 发展情况及现状 4](#_Toc317097467)

[1.3 主要研究内容 5](#_Toc1892066601)

[2 系统设计 5](#_Toc1376710097)

[2.1 控制器模块 6](#_Toc927612902)

[2.2 烟雾采集模块 6](#_Toc1330573317)

[2.3 温度监测模块 6](#_Toc603570492)

[2.4 显示模块 6](#_Toc1687926652)

[2.5 整体系统描述 7](#_Toc660260756)

[3 硬件设计 7](#_Toc959997301)

[3.1 控制器模块 7](#_Toc485560280)

[3.1.1 STC89C52单片机主要说明 8](#_Toc402724286)

[3.1.2 单片机最小系统 9](#_Toc593209441)

[3.2 LCD液晶显示屏简介 11](#_Toc1194953865)

[3.2.1 液晶显示原理 11](#_Toc894429689)

[3.2.2 液晶模块简介 11](#_Toc364228444)

[3.2.3 液晶显示部分与STC89C52的接口 12](#_Toc1947346619)

[3.3报警模块设计 13](#_Toc221558440)

[3.3.1 蜂鸣器报警 13](#_Toc270744729)

[3.3.2 灯光报警 14](#_Toc1063958031)

[3.4 温度检测模块设计 15](#_Toc1633108117)

[3.4.1 温度传感器特性 15](#_Toc2114738097)

[3.4.2 DS18B20内部结构 15](#_Toc2007905771)

[3.4.3温度检测电路 16](#_Toc1469834481)

[3.5 烟雾传感器模块介绍 17](#_Toc822890675)

[3.5.1 烟雾检测报警器设计 17](#_Toc1610120709)

[3.5.2 MQ-2型烟雾传感器的工作原理 17](#_Toc791698927)

[3.5.3 MQ-2型传感器特性 18](#_Toc631704567)

[4 功能仿真验证分析 19](#_Toc498777856)

[4.1 关于仿真与编程软件 19](#_Toc1255179497)

[4.2 AD电路图 21](#_Toc524872353)

[5 系统调试 22](#_Toc327254586)

[5.1 硬件调试 22](#_Toc1572276965)

[5.2 软件调试 23](#_Toc269455306)

[5.3 调试结果 24](#_Toc1703964683)

[6 总结 24](#_Toc352406219)

[参考文献： 26](#_Toc1600028624)

[致谢 28](#_Toc160051528)

[附录 29](#_Toc2040332871)

[附录1：代码 29](#_Toc112805732)

[附录2：实物图 42](#_Toc1120048829)

基于单片机的智能火灾预警装置的设计与实现

刘玲 指导教师：孙伟

# 摘要:在我们的日常生活中，随着社会的进步和经济的发展，无论是什么场所，都有很大的火灾隐患存在。很多人都认识到了这一点，并针对火灾隐患研究并设计了火灾报警器，但是目前市面很多火灾报警器在小型场所并不适用，往往结构复杂成本又高。因此，很有必要设计一种结构简单，经济且实用的火灾预警装置。这次的设计核心控制器选用STC89C52单片机，用MQ-2型烟雾传感器和DS18B20温度传感器分别监测烟雾浓度和温度，将监测到的信息发送至单片机，单片机处理后发送给LDC1602液晶显示屏并显示在液晶屏上，最后单片机将数据与预设值对比，超出设定值则进行声光报警。这个设计有一定的实用性。

关键词：火灾报警器；STC89C52；实用性

# **Design and Implementation of Intelligent Fire Warning Device Based on SCM**

**LiuLing Tutor: SunWei**

**Abstract:**In our daily life, with the social progress and economic

development, no matter what place, there are a lot of fire hazards. Many people have realized this, and designed the fire alarm for the fire hidden trouble, but at present, many fire alarms are not

applicable in the small place, often the structure is complex and

costly. Therefore, it is necessary to design a simple, economical

and practical fire warning device. The design of the core controller selected STC89C52 single-chip microcomputer, with MQ-2 smoke

sensor and DS18B20 temperature sensor monitoring the smoke

concentration and temperature, the monitoring of the information

sent to the SCM, SCM processing after sending to LDC1602 LCD screen and display on the LCD screen, The last SCM compares the data with the preset value, and then the sound and light alarm is exceeded. This design has certain practicality.

**Key words:** Fire Alarm;STC89C52;Practicality

# 1 前言

## 1.1背景及意义

现代科技飞速发展，同时也伴随着越来越多的火灾隐患产生。为了我们的生活安全有保障，及早发现火灾，防范于未然，让我们的人身安全和财产安全受到足够的保护。尽可能的在火灾发生初期发现并消灭火灾，阻止火灾带来的严重事故。火灾一旦发生，不仅会给我们赖以生存的环境造成巨大的破坏，还可能危及我们的生命，造成妻离子散、家破人亡的不可预料的伤害。“星星之火，可以燎原”。因此我们为了减少这类事故的发生，为了我们生活的幸福，为了我们国家现代化建设顺利发展，我们必须在室内适当的位置放置用来感应烟雾浓度和室内温度的仪器，随时监测，一旦检测到异常立即报警，及早发现，及早遏止。这种装置就是火灾预警装置，所以我们应该在这方面多作研究和贡献。

## 1.2 发展情况及现状

近年来，单片机的应用已逐步深入到社会的每一个角落。现在市面上有各种各样适用于生活中、工业上、农业上等的单片机，在大学课程中我们也学到并运用过。在任何电子系统中都可以用到，因为它功能齐全，体积小，成本低的特点。因此同样，在报警技术领域对单片机也有广泛的运用，基于单片机的智能火灾预警装置无论是从功能的完善还是性能优差都比较有大优势，现在越来越多的人研究并开发各式各样的火灾报警装置，基于单片机的依然处于比较领先的地位。

我国火灾报警系统相对于发达国家起步晚一阶段，从我国开始研制生产火灾报警系列产品后，最先出现的是多线制开关量式火灾探测报警系统，现在基本处于淘汰的状态，可能还有极少数的厂家在生产。然后出现的总线制可寻址式目前用的比较多，二总线制可寻址开关量式火灾探测报警系统是目前最常规用的。随后出现的第三代产品模拟量传输式智能[火灾报警系统](https://wapbaike.baidu.com/item/%E7%81%AB%E7%81%BE%E6%8A%A5%E8%AD%A6%E7%B3%BB%E7%BB%9F)，它拥有很高的精确度，而且不容易出现误报的情况。我国目前这方面的水平越来越先进，火灾预警装置发方向就朝着这肿高精度、高可靠率、低误报率方向靠近。

## 1.3 主要研究内容

在此次设计的基于单片机的智能火灾预警装置主要实现以下功能：

（1）分别由烟雾传感器和温度传感器监测到烟雾浓度和温度实时值；

（2）在液晶显示屏上显示出烟雾浓度；

（3）在液晶屏上显示出实时温度；

（4）单片机通过接收到的烟雾浓度值和温度与预设阀值做比较，若大于阀值则进行声光报警。

# 2 系统设计

在这次的设计中，要想实现以上功能，则需要完成对控制器模块、烟雾采集模块、温度检测模块和显示模块的设计。

## 2.1 控制器模块

次次设计采用STC89C52单片机，该单片机是一个高可靠性，超低价，无法解密，高性能的8位单片机，32个IO口[1]。在所学的课程中也曾学过51系列单片机，运用起来比较熟练。该单片机还可以在线编程和调试，使用起来很方便。

## 2.2 烟雾采集模块

此次采用的MQ-2型烟雾传感器被广泛的应用在烟雾报警系统中，它采用了离子式烟雾传感，有较大的探测范围，使用寿命一般较长，可靠性高，性能比气敏电阻类火灾报警器好。只需要简单的驱动电路这是一个很大的优点，

## 2.3 温度监测模块

DS18B20数字温度传感器接线方便，封装成后可应用于多种场合，如管道式，螺纹式，磁铁吸附式，不锈钢封装式，型号多种多样，有LTM8877，LTM8874等等[2]。DS18B20数字传感器，输出数字量，可直接读取数据。使用简便，精确度高。

## 2.4 显示模块

字符型液晶显示模块是一种专门用于显示字母、数字、符号等点阵式LCD[3]，目前常用16\*1，16\*2，20\*2和40\*2行等的模块。LCD1602需要的接口线较多，但调试起来很方便。

## 2.5 整体系统描述

此次设计的智能火灾预警装置主要由单片机、液晶显示屏、烟雾传感器、温度传感器、模数转换芯片，蜂鸣器、LED灯等构成。可将系统划分为两大部分，烟雾及温度采集部分和声光报警部分。系统的结构框图如下图所示：



图2-1 系统结构图

# 3 硬件设计

## 3.1 控制器模块

单片机单片机是一个很微型但完整的计算机系统，也被称为微控制器，，是整个系统的核心，起着核心控制器的作用。在这个设计中，它起到从各个模块读取数据、输出数据到LCD液晶显示屏、比较烟雾浓度值、温度值，再驱动液晶将相关数据显示出来的作用。我选用了51系列单片机中的STC89C52作为控制器模块。

在STC89C52的40个引脚中，其中4个8位并行I/O口，1个全双工异步串行口，2个16位定时/计数器[4]。STC89C52的存储器系统由4K的程序存储器(掩膜ROM)，和128B的数据存储器(RAM)组成。

STC89C52单片机的内部结构框图见图3-1。



图3-1 STC89C52单片机结构图

### 3.1.1 STC89C单片机主要说明

最早期的单片机都是8位或者4位的，STC89C52是8位的CPU，内部有一个高增益反向放大器，可执行复位操作，这是单片机的初始化操作，复位操作高电平有效，此操作对其他寄存器有一些影响；当程序运行错误或者自己操作失误时，也可以复位重新启动。它还有一个特殊的寄存器映像功能，在这个设计中没有用到。可以读或者写的数据一般存在片内的数据存储器中，一些原始数据和表格放在片内程序存储器中；它的4组8 位可双向传输的I／O 接口P0-P3，其中P0口为三态双向口，P1和P2口为准双向口，P3为双功能口。通常情况下该单片机需要外接一个晶振。最高允许振荡频率为12MHz[5]。

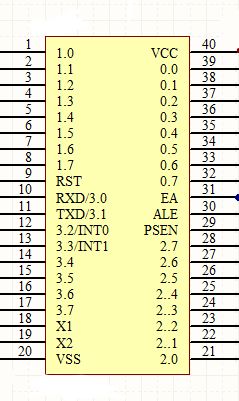


图3-2 STC89C52单片机管脚图

此次设计中使用到的引脚：

（1）19脚和20脚：这两个脚是时钟电路引脚，分别接外部晶体和微调电容的两端；

（2）9脚是复位引脚，可控制单片机的基本初始化操作；31脚也同为控制信号引脚，完成读指令操作，会根据所接电平的高低不同选择读取内部或者外部程序[6]。

（3）要完成单片机电路的完整设计需要接若干个可编程引脚。此次设计中用到了32脚-39、 1脚-6脚21-24脚共18个双向I/O口。

### 3.1.2 单片机最小系统

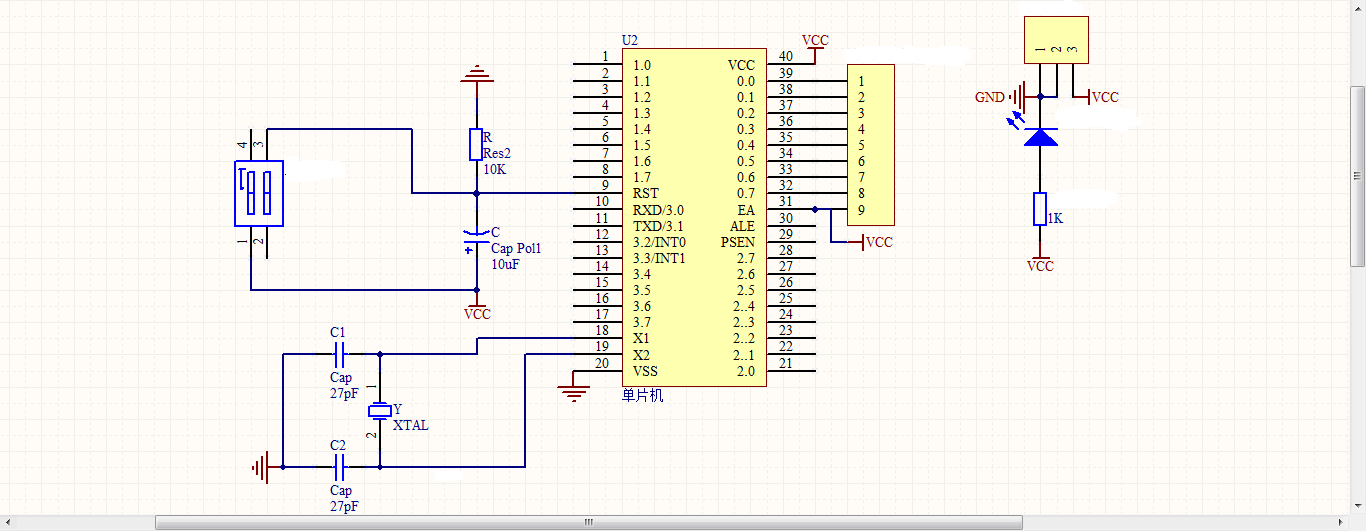


图3-3 单片机最小系统电路图

图3-3是单片机最小系统电路图，是指用最少的元件组成的单片机可以工作的系统。本次设计中51系列单片机来说，最小系统包括:单片机、晶振电路、复位电路和LED驱动电路。最小系统中的晶振大小由单片机中时钟周期大小要求确定，接入单片机的18脚和19脚，分别是晶振电路的输出端和输入端。复位电路在其中起到的作用很关键，单片机的9脚出现2个机器周期的高电平时，会进行复位。STC89C52P0口由于内部没有上拉电阻,是开漏的，无论它接的电源多大都相当于0V，需要外部的电源提供，且要避免输入时读错数据，所以要外接一个上拉电阻。在LED驱动电路中，由于LED发光工作条件，STC89C52发出的电流不足以点亮LED,需要接一个1K电阻。

## 3.2 LCD液晶显示屏简介

### 3.2.1 液晶显示原理

液晶组成物质是一种以碳为中心所构成的有机化合物，当通电的时候会导通，使液晶分子的排列变得有秩序，这些分子可以认为是完全平行的，中间有一定的缝隙，这样光线就可以通过。LCD有两块玻璃板，背面还有背光板和反光膜，光线穿过时进行有规律的折射从而产生光学效果，由此就可以在液晶屏上显示出来。也是因为其中的液晶分子，所以会显示出不会连续但是错落有致的的字符。利用此一原理，RCA公司发明了世界第一台使用液晶显示的屏幕[7]。

### 3.2.2 液晶模块简介

LCD1602由若干个5X7或者5X11等点阵字符位组成[8]，芯片工作电压4.5V-5.5V，工作电流为2.0Ma(5.0V)。LCD1602可执行四个基本操作，读状态、读数据、写命令和些数据。LCD1602液晶模块的引脚图如图3-4所示。

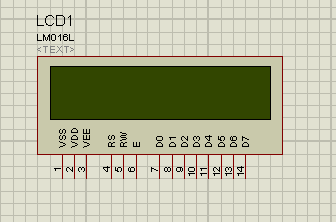


图3-4 LCD1602引脚图

LCD1602引脚接口说明:

1脚接地；2脚接5V电源的正极。3脚是液晶显示的偏压信号，可接10K的3296精密电位器。 4脚是[寄存器](https://baike.so.com/doc/1248822-1320752.html" \t "_blank)选择、5脚是读写信号线；4脚和5脚电平为0时显示地址，4脚电平为0、5脚电平为1时读忙信号；4脚电平为1、5脚电平为0时写入数据。6脚：当电平由1变为0，液晶模块执行命令。7～14是8位双向数据端[9]。

### 3.2.3 液晶显示部分与STC89C52的接口

如图3-5所示。在连接STC89C52与LCD1602液晶显示屏时时，用LCD的4脚 、5脚、6脚分别连接STC89C52的P0口用P1.0、P1.1、P1.2脚。6脚是使能端，5脚负责读写信号，4脚是寄存器选择信号，随后进行模块初始化操作。在本次的设计中，LCD显示两行字符，第一行为环境中的温度值，单位为摄氏度；第二行显示房间中的烟雾浓度比。要显示的字符或数据将由单片机驱动显示在LCD1602液晶上。

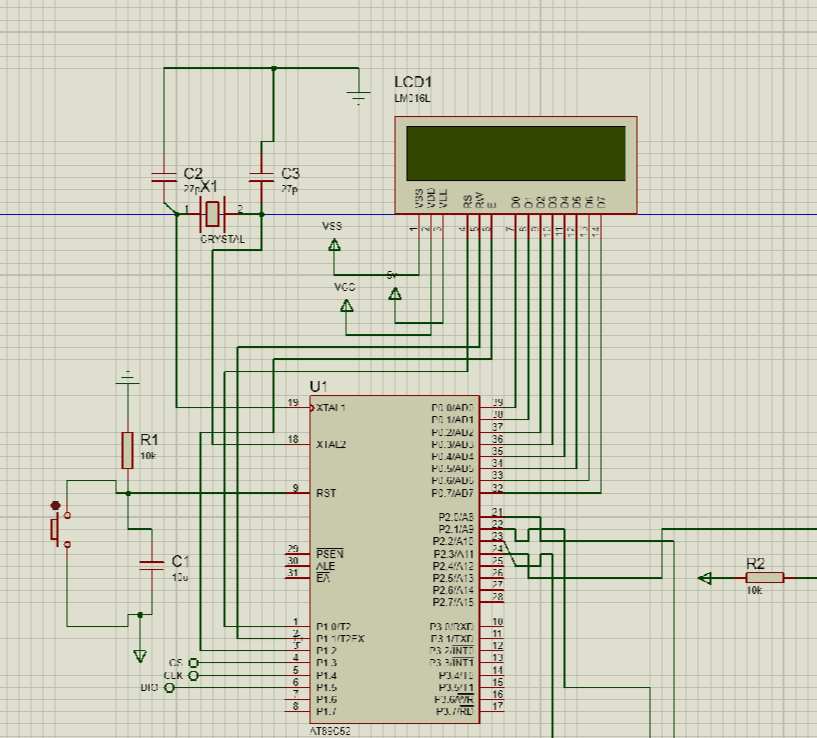


图3-5 LCD1602与STC89C52的接口

## 3.3报警模块设计

在此次设计的报警模块中，正常环境下，两个LED灯都处于熄灭状态，蜂鸣器不发出声音；当监测到烟雾浓度或温度超出设定阀值时会有对应的LED灯循环闪烁，蜂鸣器进入报警状态发出报警声；当监测到烟雾浓度和温度都超出阀值，两个LED灯会同时循环闪烁，蜂鸣器报警。

### 3.3.1 蜂鸣器报警

报警模块的声音报警电路如下图3-6所示。在电路中，需要使用三极管来放大电流，然后将他们比串联起来即可。声报警电路由单片机的P2.2引脚进行控制，当P2.2输出的电平为低电平时，三极管导通，蜂鸣器的电流形成回路，发出声音报警；否则，三极管截止，蜂鸣器不发出声音。

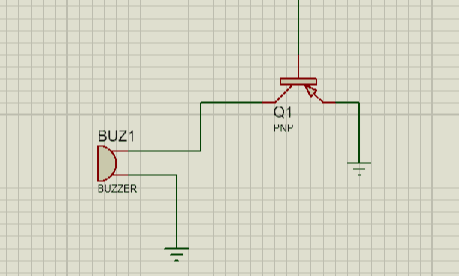


图3-6 蜂鸣器报警电路

### 3.3.2 灯光报警

报警模块中的光报警电路如下图3-7所示，用单片机的21脚和22脚分别连接LED,控制超过温度阀值的LED灯报警以及超过烟雾浓度阀值的LED灯报警。当单片机判断出监测到温度值和烟雾浓度值超过阀值时，单片机控制对应的的两个端口，完成灯光报警。

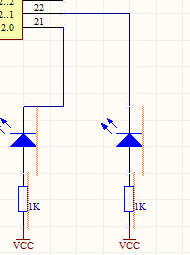


图3-7 光报警电路

## 3.4 温度检测模块设计

此次设计中用DS18B20温度传感器来检测温度，单片机通过读取DS18B20的温度数据并在LCD1602上显示，单片机比较室内温度与预设值，当室内温度值超过预设阀值时，系统就会报警。

### 3.4.1 温度传感器特性

DS18B20适应电压范围3.0V～5.5V，在寄生电源方式下可用数据线供电；它的温度时延为750ms[11]；该传感器的温度测量范围为零下55摄氏度至125摄氏度，精确度为0.625摄氏度；在零下10度至80的环境下，精确度可达0.5摄氏度。该传感器输出为数字温度信号，可直接串联传送给CPU，不需要用模数转换芯片来转换信号。如果在接电源时将正负极接反的情况下，芯片不会被烧坏，但是无法正常工作。DS18B20与微控制器只需要一条线就可以实现双向通讯。

### 3.4.2 DS18B20内部结构

DS18B20的内部结构如图3-7所示。主要包括：配置寄存器，温度传感器，64位ROM和单总线接口，高速暂存器RAM，用于存储用户设定温度上下限值的高温(TH)和低温（TL）触发器，存储器与控制逻辑，8位CRC发生器等7部分[11]。



图3-7 DS18B20内部结构

### 3.4.3温度检测电路

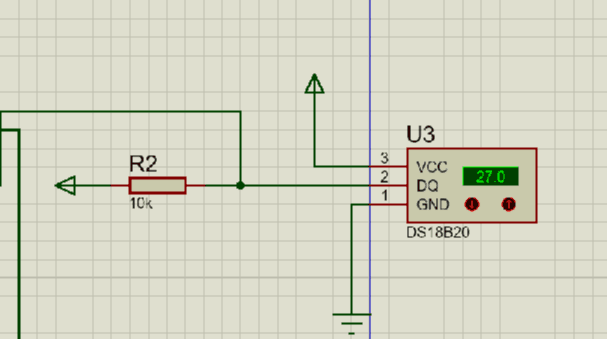


图3-8 DS18B20温度检测电路设计

图3-8为该设计中的温度检测部分电路图， DQ口为该DS18B20的数据口，单片机读DQ口的信息可得出实时温度值，由于DS18B20是单线通信，即输出0时通过三极管下拉为低电平，而输出1时，则为高阻，需要外接上拉电阻将其拉为高电平，因此是需要外接上拉电阻，否则无法输出。

## 3.5 烟雾传感器模块介绍

### 3.5.1 烟雾检测报警器设计

在此次设计的基于单片机的智能火灾预警装置中，选择合适的烟雾传感器至关重要，离子烟雾传感器比烟雾粒子传感器要灵敏，而前向式光电烟雾传感器比离子烟雾传感器感应大粒子颗粒灵敏，本次设计主要针对室内，这种情况下在发生火灾时，离子烟雾报警器往往比前向式光电烟雾报警器先报警，所以选用离子型烟雾传感器。再考虑到实用性、成本等原因，最后选用MQ-2半导体烟雾传感器完成烟雾检测部分。

### 3.5.2 MQ-2型烟雾传感器的工作原理

半导体烟雾传感器是用氧化物半导体陶瓷敏感材料和单晶半导体器件制作的烟雾传感器。

此次设计中采用的二氧化锡半导体气敏MQ-2型烟雾传感器属于表面离子式N型半导体。当处于200~300°C时，空气中的氧被二氧化锡吸附，形成氧的负离子吸附，使其电阻值增加。接触烟雾时，如果晶粒间界处的势垒受到该烟雾的调制而变化，就会使表面电导率的变化[12]。利用这一点就可以获得这种烟雾存在的信息。

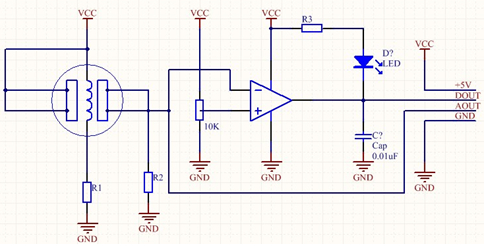


图3-9 烟雾传感器原理图

### 3.5.3 MQ-2型传感器特性

（1）MQ-2型传感器的主要特点：对天然气、液化石油气等烟雾有很高的灵敏度；具有良好的重复性和长期的稳定性。

（2）MQ-2型传感器的基本特性包括灵敏度特性、初期稳定特性。

灵敏度特性：MQ-2在最佳工作条件下，接触同一种烟雾，其电阻值RS随气体浓度变化的特性称之为灵敏度特性，用K表示。K=RS / R0 (2-1)。

初期稳定特性：MQ-2在不通电一段时间后，再通电时，MQ-2输出的电流和电压可能不准确，应该加热一段时间，这样才能正常工作。

（3）MQ-2型传感器的特性参数：回路电压：(Vc) 5～24V；取样电阻：(RL) 0.1～20K；加热电压：(VH)5±0.2V；加热功率：(P)约750mW；灵敏度：以甲烷为例R0(air)/RS (0.1%CH4)＞5；响应时间：Tres＜10秒；恢复时间：Trec＜30秒[13]。

# 4 功能仿真验证分析

## 4.1 关于仿真与编程软件

此次设计首先用Proteus软件画出仿真电路图，由于Proteus中不可仿真传感器，所以将本次设计中用到的传感器先用电流源替代，然后使用KeilC51编写程序，再通过Protues软件进行仿真调试，最后调试得到预期的效果。

1. Protues软件

在大学四年期间接触Proteus了不少，在一些课程的实验课上也是经常用到，所以这次的设计中我用了Proteus来做仿真。本设计的仿真电路图如下图4-1所示：

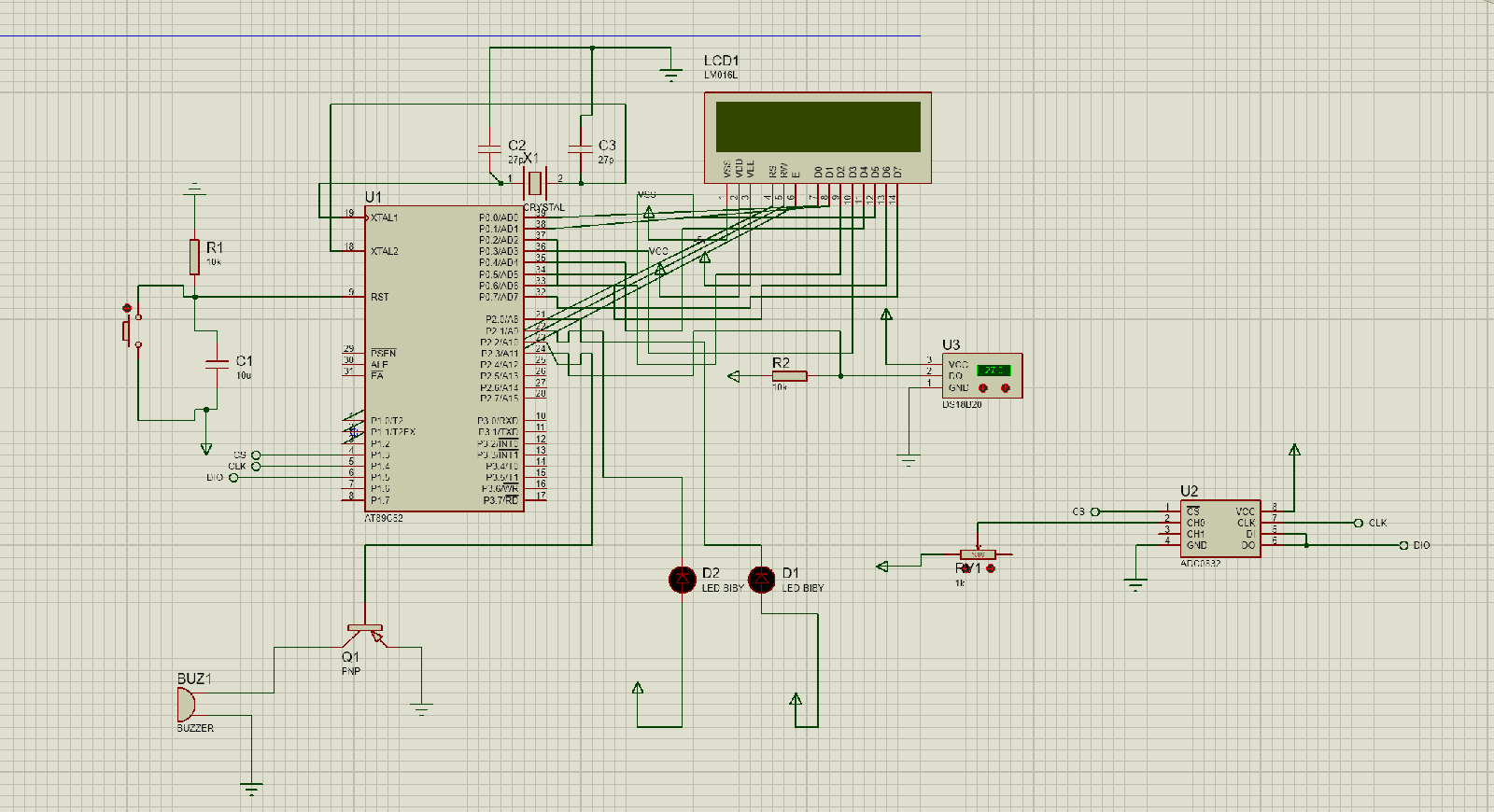


图4-1 仿真电路图

（2）Keil C51

Keil C51是美国Keil Software公司(ARM公司之一)出品的51系列兼容单片机C语言软件开发系统[14]。Keil C51在使用上不需要特别高的水准要求，在次设计中帮助了我完成了与STC89C52上的关键功能。下图4-2为此次设计的程序流程图：



图4-2 主程序流程图

## 4.2 AD电路图

为了更直观的了解本次设计中的电路设计，我使用了Altium Designer 绘出电路图。AD电路图如下图4-2所示

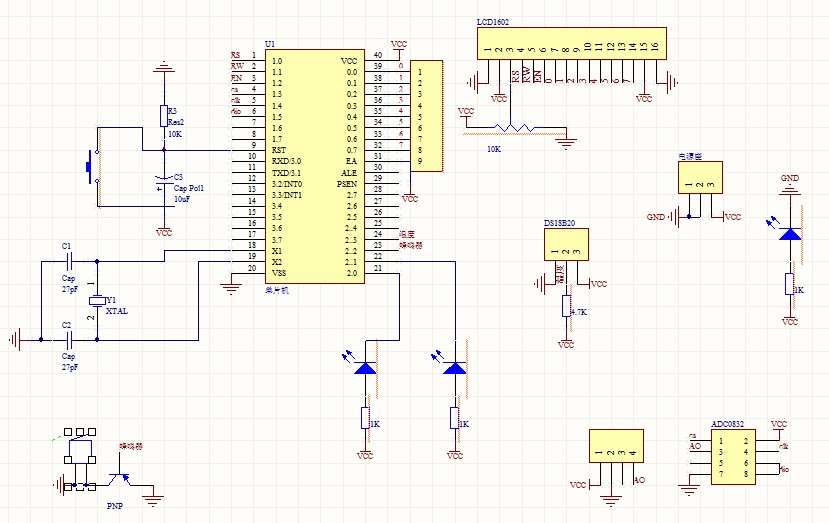


图4-2 AD电路图

# 5 系统调试

## 5.1 硬件调试

该系统在硬件放面的主要问题在于芯片的针脚比较稠密在焊接过程中极易发生短路。在调试过程中主要遇到了以下问题：

（1）蜂鸣器在触发后没有发出声音

解决：通过对于蜂鸣器的资料查找，和对于其触发时的电流的测量，发现电流不足以驱动蜂鸣器工作，在使用三极管对电流放大后，蜂鸣器正常工作。

（2）A/D转换器不能正确的读取到数值

解决：使用万用表对ADC0832的所有针脚进行排查，CLK针脚虚焊，补焊之后读取正常。

## 5.2 软件调试

软件的调试分模块进行，首先是对于LCD显示屏的调试，让其能显示出正确的字符，使的后面模块读取的数据能显示出来，方便后面模块的调试，然后依次是温度模块，烟雾模块，LCD模块和蜂鸣器模块。

在实际操作的过程中，最难的部分在对于温度模块，烟雾模块和LCD模块每次信号发出去之后的延时是不同，所以针对3种模块都有不同的延时函数，而且在选择延时时间的时候并不能得到一个准确的数值，只能在一个合理的范围内去多次尝试，来得到一个可以正确使用的延时数值。

这次的设计软件部分程序复杂，在编程和调试的时候遇到了一些阻碍。最后归纳出在软件的调试过程中主要遇到的问题如下：

（1）显示模块不能正常显示

解决：经排查发现在向LCD输出的显示字符地址最高为并不为1，在对当前地址加上80H后，输出正常；

（2）显示模块会在输出显示后，丢失几条指令

解决：查询资料该型号LCD，需要再输出指令前检查是否处于忙状态，修改代码后，输出正常。

（3）温度模块读取到的数值不正确

解决：经排查，在对温度模块传回的数据进行处理时发生错误，对处理数据的代码进行调整后，数值正常。

## 5.3 调试结果

虽然在整个设计过程中遇到了不少问题，但在我的坚持不懈下，在最后我终于解决了问题，完成了这个设计。最终实现的效果如图5-1所示：



图5-1 调试结果图

# 6 总结

经过我对这次设计的深入了解与研究，实现了对室内的烟雾浓度以及温度的实时监测，我们可以直观的在液晶显示屏上看到温度和烟雾浓度的显示值，如发现火情则触动声光报警。此次使用的单片机在我们的生活中也经常遇到，STC89C52性能好、处理数据快速，我认为采用该单片机很恰当，能较好的实现我所设计的功能。这个设计虽然结构简单，放置在室内用来预防火情变得严重是有一定的实用价值的，况且操作简单，会避免未曾接触过这方面的人不知如何使用。如发生火情，按下复位键即可停止。此次设计用到的芯片、传感器以及其他元件都比较平价，所以不存在价格方面的问题 。在此次设计中传感器起到的作用也是必不可少的，选用的MQ-2型半导体电阻式烟雾传感器对于小型室内的火灾预警装置是比较理想的，而且对烷类烟雾很敏感。温度传感器DS18B20有一个很大的优势，直接发出数字信号，串联到单片机即可，简单的电路并不妨碍我所实现的功能。

但是由于自身能力等其他因素，这个火灾预警装置的设计并不是很完美，很多地方还需要改善。此设计由于系统的算法不够成熟，有一定的误报率，灵敏度可能不够；BS18B20在条件允许的情况下可以并联8个，从而测试不同点的温度。DS18B20能测的温度范围较小，不适合在稳定的上升很快的场所使用；如果用户想要自己设定阀值比较难，也不能保证所有场所适用。

# 参考文献：

[1] 高林.单片机原理与微机原理综合仿真系统的设计及应用[J].实验技术与管理,2014,31(03):91-94.

[2] 郭志辉.DS18B20在数字温度计系统设计的理论分析与探究[J].新课程(教育学术),2011(04):172-173.

[3] 林嘉.基于89S52的LCD1602程序设计[J].电脑知识与技术,2012,8(26):6376-6378.

[4] 张鲲,陈美伊,李壮,段志扬,菅永利,刘绍宇.基于单片机火灾报警系统设计原理及方案的比较分析与研究[J].新型工业化,2014,4(02):74-77+81.

[5] 黄景早.真空微电子加速度传感器信号处理电路的研究[D].重庆大学,2006.

[6] 刘新春.网络通讯技术中软件故障讨论[J].信息通信,2013(01):91+43

[7] 黄祖沛.液晶显示器介绍——显示器的工作原理(二)[J].电子测试,2001(02):172-17.

[8] 汤慕豪. 基于微控制单元电容式湿度传感器的研究[D].华中科技大学,2012.

[9] 董继承,黄宇.带时钟的数字温度计的设计与制作[J].中国科技信息,2007(08):65-67.

7.

[10] 王文川. 基于ARM的嵌入式智能家居系统的研究[D].合肥工业大学,2013.

[11] 胡显华.火灾探测器误报警的原因及改进方法[J].电脑开发与应用,2007(11):60-62+65.

[12] 江杰,宋宏龙.基于GSM短信的烟雾传感报警系统[J].测控技术,2014,33(01):1-3.

[13] 盛艳. 基于单片机和GPRS实验室安全报警监控系统研究[D].电子科技大学,2012.

[14] 陈媛媛,陈菁.基于Proteus与Keil单片机实验教学的探讨与研究[J].电子世界,2016(09):28-29.

# 致谢

在完成本设计的写作过程中，我十分感谢我的指导老师。从选题到完成设计，老师一直都是很悉心的给我讲解着在设计中遇到的各种问题，循循善诱，严格把关，帮助我开拓设计思路，并不断地鼓舞着我，使我感到信心倍增，让我非常积极地投入到设计中，不断地完成设计中的一个个部分。在此，再次感谢老师在设计上不断地给与我帮助，让我在大学里的最后一次的学习过程中，充分感受到了自己对学习的兴趣和热情，使我能够圆满地完成自己的毕业设计。

回想大学四年的时光，仿佛尽在昨天。最后，我还要感谢在我大学四年的学习期间给我极大关心和支持的家人、各位老师以及我的同学和朋友。是你们在生活和学习上不断给与我支持、帮助和无微不至的关怀，是你们不断地给与了我信心，让我在人生中一次次坚强地走下去。

# 附录

## 附录1：代码

#include <reg52.h>

#include <intrins.h>

#define uint unsigned int

#define uchar unsigned char

typedef unsigned char BYTE;

typedef unsigned short WORD;

unsigned char word1[16]={"CH = "};

sbit ADCS =P1^3;

sbit ADDI =P1^5;

sbit ADDO =P1^5;

sbit ADCLK =P1^4;

sbit ds1=P2^3;

sbit rs=P1^0;

sbit rw=P1^1;

sbit en=P1^2;

sbit led1=P2^0;

sbit led2=P2^1;

sbit beer=P2^2;

unsigned char readad[2];

void Adc0832(unsigned char channel);

float tty;

int a = 0, b = 0, c = 0, t = 0;

float tt = 0;

void delay\_nms(unsigned int k)

{unsigned int i,j;

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

{for(j=0;j<121;j++){;}}}

void Delay5us()

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

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

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

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

}void Delay5ms()

{ WORD n = 560;

while (n--);}

void Delay50us() //@11.0592MHz

{ unsigned char i, j;

\_nop\_();

i = 1;

j = 134;

do

{while (--j);

} while (--i);}

void Delay80us() //@11.0592MHz

{ unsigned char i, j;

\_nop\_();

i = 1;

j = 217;

do

{while (--j);}

while (--i);}

void Delay30us() //@11.0592MHz

{unsigned char i;

\_nop\_();

\_nop\_();

i = 80;

while (--i);}

void Adc0832(unsigned char channel)

{ uchar i=0;

uchar j;

uint dat=0;

uchar ndat=0;

if(channel==0)channel=2;

if(channel==1)channel=3;

ADDI=1;

\_nop\_();

\_nop\_();

ADCS=0;

\_nop\_();

\_nop\_();

ADCLK=1;

\_nop\_();

\_nop\_();

ADCLK=0;

\_nop\_();

\_nop\_();

ADCLK=1;

ADDI=channel&0x1;

\_nop\_();

\_nop\_();

ADCLK=0;

\_nop\_();

\_nop\_();

ADCLK=1;

ADDI=(channel>>1)&0x1;

\_nop\_();

\_nop\_();

ADCLK=0;

ADDI=1;

\_nop\_();

\_nop\_();

dat=0;

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

{ dat|=ADDO;

ADCLK=1;

\_nop\_();

\_nop\_();

ADCLK=0;

\_nop\_();

\_nop\_();

dat<<=1;

if(i==7)dat|=ADDO;}

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

{ j=0;

j=j|ADDO;

ADCLK=1;

\_nop\_();

\_nop\_();

ADCLK=0;

\_nop\_();

\_nop\_();

j=j<<7;

ndat=ndat|j;

if(i<7)ndat>>=1;}

ADCS=1;

ADCLK=0;

ADDO=1;

readad[0]=dat;

readad[1]=ndat;}

void Tempprocess()

{tty=readad[0]/255.0\*99.0;

word1[4]=(unsigned char )(tty);}

void DelayUs2x(unsigned char t)

{while(--t);}

void DelayMs(unsigned char t)

{ while(t--)

{ DelayUs2x(245);

DelayUs2x(245);}}

void shuaxin(uint shuaxint)/

{while(shuaxint--);}

bit LCD\_Check\_Busy(void)

{P0=0xFF;

rs=0;

rw=1;

en=0;

\_nop\_();

en=1;

return (bit)(P0 & 0x80);}

void LCD\_Write\_Com(uchar com)

{DelayMs(5);

rs=0;

rw=0;;

en=1;

P0= com;

\_nop\_();

en=0;}

void LCD\_Write\_Data(uchar Data)

{ DelayMs(5);

rs=1;

rw=0;

en=1;

P0= Data;

\_nop\_();

en=0;}

void LCD\_Clear(void)

{ LCD\_Write\_Com(0x01);

DelayMs(5); }

void LCD\_Write\_Char(unsigned char x,unsigned char y,unsigned char Data)

{ if (y == 0)

{LCD\_Write\_Com(0x80 + x);}

else

{LCD\_Write\_Com(0xC0 + x);}

LCD\_Write\_Data( Data);}

void LCD\_Write\_String(unsigned char x,unsigned char y,unsigned char \*s) //写入字符串的函数

{ if (y == 0)

{LCD\_Write\_Com(0x80 + x);}

else

{LCD\_Write\_Com(0xC0 + x);}

while (\*s)

{LCD\_Write\_Data( \*s);

s ++; }}

void LCD\_Init(void)

{LCD\_Write\_Com(0x38);

DelayMs(5);

LCD\_Write\_Com(0x38);

DelayMs(5);

LCD\_Write\_Com(0x38);

DelayMs(5);

LCD\_Write\_Com(0x38);

LCD\_Write\_Com(0x08);

LCD\_Write\_Com(0x01);

LCD\_Write\_Com(0x06);

DelayMs(5);

LCD\_Write\_Com(0x0C); }

void DS18B20\_Delay(uint count)

{ while(count--);}

void delaytt(uint ttt)

{while(ttt--); }

uint DS18B20\_Init(void)

{ uint x = 0;

ds1 = 1;

DS18B20\_Delay(8);

ds1 = 0;

DS18B20\_Delay(80);

ds1 = 1;

DS18B20\_Delay(14);

x = ds1;

DS18B20\_Delay(20);

return (~x);}

uint DS18B20\_Read1Byte(void)

{uint i = 0;

uint dat = 0;

for(i=8;i>0;i--)

{ ds1 = 0;

dat >>= 1;

ds1 = 1;

if(ds1) dat |= 0x80;

DS18B20\_Delay(4);}

return (dat);}

void DS18B20\_Write1Byte(uint dat)

{uint i = 0;

or(i=8;i>0;i--)

{ ds1 = 0;

ds1 = dat&0x01;

DS18B20\_Delay(5);

ds1 = 1;

dat>>=1;}}

void Read\_Disp\_Temperature(void)

{ DS18B20\_Init();

DS18B20\_Write1Byte(0xCC);

DS18B20\_Write1Byte(0x44);

DS18B20\_Init();

DS18B20\_Write1Byte(0xCC);

DS18B20\_Write1Byte(0xBE);

a = DS18B20\_Read1Byte();

b = DS18B20\_Read1Byte();

t = b;

t<<= 8;

t = t | a;

tt = t \* 0.0625;

t = tt \* 10 + 0.5;

a = t / 100;

b = t / 10 - a \* 10;

c = t - a \* 100 - b \* 10;

LCD\_Write\_String(0,0,"T:");

LCD\_Write\_Char(4,0,'T');

LCD\_Write\_Char(5,0,':');

LCD\_Write\_Char(6,0,0x30+a);

LCD\_Write\_Char(7,0,0x30+b);

LCD\_Write\_Char(8,0,'.');

LCD\_Write\_Char(9,0,0x30+c);

LCD\_Write\_String(10,0,"C");}

void main (void)

{LCD\_Init();

LCD\_Clear();

while(1)

{ Adc0832(0);

Tempprocess();

Read\_Disp\_Temperature();

LCD\_Write\_String(0,1,"YW value:");

LCD\_Write\_Char(12,1,word1[4]/10+0x30);

LCD\_Write\_Char(13,1,word1[4]%10+0x30);

LCD\_Write\_Char(14,1,'%');

if(tty>20||tt>30)

{beer=0;}

else

{beer=1;}

if(tty>20)

{led1=0;}

else

{led1=1;}

if(tt>30)

{led2=0;}

else

{led2=1;}}}

## 附录2：实物图

