

# 西安邮电大学

## 毕业设计（论文）

题目：基于嵌入式的室内火灾检测系统\_\_\_\_\_

学院：\_\_\_\_\_自动化\_\_\_\_\_

专业：\_\_\_\_\_自动化\_\_\_\_\_

班级：\_\_\_\_\_自动 1403\_\_\_\_\_

学生姓名：\_\_\_\_\_杨文洁\_\_\_\_\_

学号：\_\_\_\_\_06141100\_\_\_\_\_

导师姓名：\_\_\_\_\_李育贤\_\_\_\_\_职称：\_\_\_\_\_教授\_\_\_\_\_

起止时间：2017 年 12 月 5 日至 2018 年 6 月 10 日

## 毕业设计（论文）声明书

本人所提交的毕业论文《基于嵌入式的室内火灾检测系统》是本人在指导教师指导下独立研究、写作的成果，论文中所引用他人的文献、数据、图件、资料均已明确标注；对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式注明并表示感谢。

本人完全理解《西安邮电大学本科毕业设计（论文）管理办法》的各项规定并自愿遵守。

本人深知本声明书的法律责任，违规后果由本人承担。

论文作者签名：

日期：        年    月    日

## 西安邮电大学本科毕业设计(论文)选题审批表

申 报 人	李育贤	职 称	教授	学 院	自动化			
题 目 名 称	基于嵌入式的室内火灾检测系统							
题目来源	科研				教学		其它	是
题目类型	硬件设计	是	软件设计		论文		艺术作品	
题目性质	应用研究		是		理论研究			
题目简述	<p>我国工业化水平不断提高,经济高速发展的同时也带来了严重的环境污染和安全问题,有害气体挥发泄漏、室内失火等问题频发,严重威胁到了人们的健康及生命财产安全,从而对室内环境实时监测的需求越来越大、要求也越来越高。基于无线网络技术的嵌入式室内火灾监测系统性能更优越,具有成本低、组网灵活、布线简单、扩展性好等优点。</p>							
对学生知识与能力要求	<p>掌握微机原理与接口技术及单片机原理及应用,理解嵌入式系统的基本组织结构与工作原理,具有一定的模拟电路技术基础及网络通信原理,熟悉常用传感器原理及使用方法,具有嵌入式软件设计能力及解决实际问题的动手能力。</p>							
具体任务以及预期目标	<p>设计基于嵌入式的室内火灾检测系统, ,选取室内温度、燃气泄露等环境参数作为监测对象,实现对室内环境及安全状况的实时监测监控,并在本地及网络端实时监测显示。</p>							
时间进度	<div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> 2018. 03. 01-2018. 03. 15:  2018. 03. 16-2018. 03. 30:  2018. 04. 01-2018. 04. 10:  2018. 04. 15-2018. 04. 30:  2018. 05. 01-2018. 05. 25:  2018. 05. 26-2018. 06. 01: </div> <div style="width: 50%;"> 查阅资料, 撰写提交开题报告。  确定系统架构设计方案, 购买元器件。  系统软件程序设计。  软硬件综合调试, 系统功能完善。  撰写毕业设计论文。  修改、装订论文。 </div> </div>							
系(教研室)主任签字	年 月 日			主管院长 签字	年 月 日			

## 西安邮电大学本科毕业设计（论文）开题报告

学生姓名	杨文洁	学号	06141100	专业班级	自动 1403
指导教师	李育贤	题目	基于嵌入式的室内火灾检测系统		
<p>选题目的：</p> <p>随着社会进步科技发展，对火灾预防技术的要求也越来越高。为了确保人员生命和公私财产的安全，对室内所设置的消防设施提出了更高的要求。火灾自动报警系统是这些消防设施中最基本和最重要的组成部分。火灾自动探测报警技术是将传感技术、微机原理与接口技术及单片机原理应用于火灾预防的一项综合性技术。它具有成本低、组网灵活、布线简单、扩展性好等点。该设计是在嵌入式的基础上应用熟练的专业知识技能实现对室内环境及安全状况的实时监测监控并在本地及网络端实时监测显示。</p>					
<p>前期基础：</p> <p>已学课程：</p> <p>单片机原理及应用、传感器原理及应用、数字控制系统设计、嵌入式控制系统、微机原理与接口技术。</p> <p>掌握的工具：电烙铁、电脑、万用表、实验箱等。</p> <p>参考文献：【1】李全钊 单片机原理及接口技术 【2】何道清 传感器原理及应用 【3】杨栓科 模拟电子技术基础</p> <p>软硬件条件：</p> <p>软件：Keil uVision4、ISIS 7 Professional。</p> <p>硬件：电源模块、温湿度传感器模块、单片机模块、显示系统模块等。</p>					
<p>要研究和解决的问题：</p> <p>本课题研究的内容是建立在嵌入式平台上的火灾测系统。主要是构建嵌入式平台，在平台上利用嵌入式的特点完成对火灾检测系统的软件设计，并实现软件与硬件设备的对接，已达到对室内温度过高、燃气泄露等环境参数的监测和报警。火灾检测由温度检测模块和烟雾检测模块构成。</p> <p>本课题研究的重点是如何有效识别火灾信息，避免误报。火灾检测由温度检测模块和烟雾检测模块构成，其温度传感器选用 AD590，烟雾传感器选用 MQ-2 烟雾传感器。当收到探测器、手动报警开关、消火栓开关及输入模块所配接的设备所发来的火警信号时，报警器会发出火灾报警信号。</p>					

工作思路和方案：

系统结构图如图 1 所示：

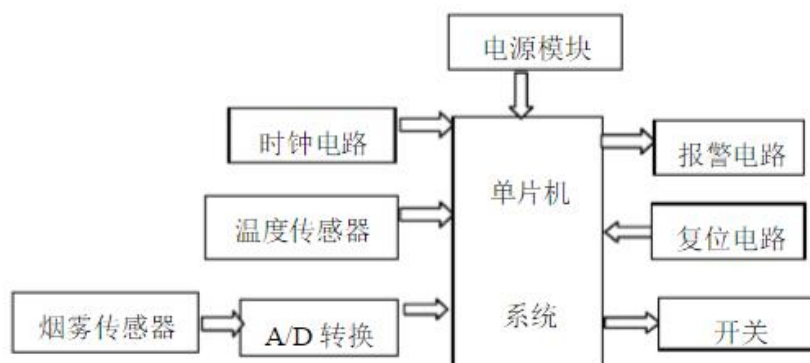


图 1 系统结构框图

该系统主要包括单片机控制模块，电源模块，温湿度传感器模块，烟雾传感器模块。利用了多传感器融合技术和嵌入式技术，可实现室内火灾环境综合监测功能。工作进度如下：

2018.03.01-2018.03.15:	查阅资料，撰写提交开题报告。
2018.03.16-2018.03.30:	确定系统架构设计方案，购买元器件。
2018.04.01-2018.04.10:	系统软件程序设计。
2018.04.15-2018.04.30:	软硬件综合调试，系统功能完善。
2018.05.01-2018.05.25:	撰写毕业设计论文。
2018.05.26-2018.06.01:	修改、装订论文。

指导教师意见

签字：                    年    月    日

# 西安邮电大学毕业设计（论文）成绩评定表

学生姓名	杨文洁	性别	女	学号	06141100	专业班级	自动 1403
课题名称	基于嵌入式的室内火灾检测系统						
指导教师意见	评分（百分制）：_____ 指导教师(签字)：_____ _____年__月__日						
评阅教师意见	评分（百分制）：_____ 评阅教师(签字)：_____ _____年__月__日						
验收小组意见	评分（百分制）：_____ 验收教师(组长)(签字)：_____ _____年__月__日						
答辩小组意见	评分（百分制）：_____ 答辩小组组长(签字)：_____ _____年__月__日						
评分比例	指导教师评分__ (%) 评阅教师评分__ (%) 验收小组评分__ (%) 答辩小组评分__ (%)						
学生总评成绩	百分制成绩			等级制成绩			
答辩委员会意见	毕业论文(设计)最终成绩(等级)：_____  学院答辩委员会主任(签字)：_____ _____年__月__日						

# 目 录

摘 要	I
ABSTRACT	II
第一章 绪论	1
1.1 论文研究目的和意义	1
1.2 火灾检测系统的国内外现状	1
1.3 本论文主要任务	1
第二章 背景知识介绍	2
2.1 烟雾传感器模块	2
2.1.1 烟雾传感器介绍	2
2.1.2 MQ-2 型烟雾传感器的工作原理	2
2.1.3 MQ-2 型传感器的特性	3
2.2 温度传感器模块	4
2.2.1 DS18B20 的介绍	4
2.2.2 温度传感器的控制方法	4
2.3 单片机模块	4
2.3.1 主要功能特性	5
2.4 GPRS 系统	5
2.5 LCD 液晶显示屏介绍	6
2.5.1 LCD 的主要技术参数	6
2.5.2 LCD 的引脚说明	6
2.6 软件介绍	7
第三章 系统方案设计	8
3.1 系统功能	8
3.2 系统框图	8
第四章 硬件设计	9
4.1 单片机最小系统电路	9
4.1.1 复位电路	9
4.1.2 晶振电路	9

4.1.3 P0 口的上拉电阻-----	9
4.2 温度检测模块-DS18B20 测温电路-----	10
4.3 报警模块-----	11
4.3.1 蜂鸣器报警电路-----	11
4.3.2 按键电路-----	12
4.4 显示模块-液晶显示电路-----	12
4.5 烟雾检测模块-----	13
4.6 网络模块-----	13
4.7 本章小结-----	14
第五章 软件设计-----	15
5.1 软件下载和安装-----	15
5.2 使用教程-----	15
5.3 温度检测模块-----	15
5.3.1 温度检测模块流程图-----	15
5.4 烟雾检测模块-----	16
5.4.1 烟雾检测流程图-----	16
5.5 A/D 转换模块-----	17
5.5.1 A/D 转换模块流程图-----	17
5.6 显示模块-----	18
5.6.1 LCD 液晶显示流程图-----	18
5.7 网络模块-----	19
5.7.1 GPRS 模块流程图-----	19
5.8 报警模块-----	19
5.8.1 报警模块流程图-----	19
5.9 本章小结-----	20
第六章 系统测试及调试-----	21
6.1 系统调试-----	21
6.2 烟雾报警测试-----	21
6.2.1 测试目的-----	21
6.2.2 测试结果-----	21
6.3 温度报警测试-----	23
6.3.1 测试目的-----	23
6.3.2 测试结果-----	23



6.4 实验结果分析-----	25
6.5 本章小结-----	25
结束语-----	27
致谢-----	28
参考文献-----	29
附录-----	30

## 摘 要

随着“信息时代”的到来，作为获取信息的手段——传感器技术得到了显著的进步，其应用领域越来越广泛，对其要求越来越高，需求越来越迫切。传感器技术已成为衡量一个国家科学技术发展水平的重要标志之一。因此，了解并掌握各类传感器的基本结构、工作原理及特性是非常重要的。

为了提高对传感器的认识和了解，尤其是对烟雾传感器的深入研究以及其用法与用途，基于实用、广泛和典型的原则而设计了本系统。本文利用单片机结合传感器技术而开发设计了这一烟雾监控系统。

本论文以电阻式烟雾传感器和单片机技术为核心并与其他电子技术相结合，设计出一种技术水平较好的烟雾报警器。其中选用 MQ-2 型半导体可燃气体敏感元件烟雾传感器实现烟雾的检测，具有灵敏度高、响应快、抗干扰能力强等优点，而且价格低廉，使用寿命长。选用的 AT89S52 单片机，其整合了 A/D 转换、硬件乘法器、硬件脉宽调制器等资源，具有高速、低功耗、超强抗干扰等优点，是目前同类技术中性价比较高的产品。

以 AT89S52 单片机和 MQ-2 型半导体电阻式烟雾传感器为核心设计的烟雾报警器可实现报警、故障自诊断、浓度级别显示、报警限设置、温度显示及与温度报警值设定等功能。是一种结构简单、性能稳定、使用方便、价格低廉、智能化的烟雾报警器。具有一定的实用价值。

**关键词：**烟雾；报警器；AT89S52；传感器； GPRS 模块

# ABSTRACT

With the advent of the "Information Age", sensor technology has made remarkable progress as a means of obtaining information. Its application fields have become more and more extensive, and its requirements have become higher and higher, and demand has become more pressing. Sensor technology has become one of the important indicators to measure the development level of a country's science and technology. Therefore, it is very important to understand and master the basic structure, working principle and characteristics of various types of sensors.

In order to improve the understanding and understanding of the sensor, especially the in-depth study of the smoke sensor and its usage and use, the system was designed based on practical, extensive and typical principles. This paper develops and designs this smoke monitoring system using single-chip microcomputer and sensor technology.

In this dissertation, the smoke sensor and single-chip microcomputer technology are taken as the core and combined with other electronic technologies to design a smoke alarm with a good technical level. Among them, MQ-2 type semiconductor flammable gas sensor smoke sensor is used to detect smoke. It has the advantages of high sensitivity, fast response, and strong anti-interference ability. It also has low price and long service life. The selected AT89S52 MCU integrates resources such as A/D conversion, hardware multipliers, and hardware pulse width modulators. It has the advantages of high speed, low power consumption, and superior anti-interference. It is currently the most cost-effective product in its class.

The smoke alarm designed with AT89S52 micro and MQ-2 semiconductor resistive smoke sensor as the core can realize the functions of alarm, fault self-diagnosis, concentration level display, alarm limit setting, temperature display and temperature alarm value setting. It is a kind of smoke alarm with simple structure, stable performance, convenient use, low price and intelligence. Has a certain practical value.

**Keywords:** smoke, alarm, AT89S52, sensor; GPRS module

## 第一章 绪论

### 1.1 论文研究目的和意义

随着社会进步科技发展,防火技术的要求也越来越高。为了保证人员生命安全和公私财产安全,对室内设置的消防设施提出了更高的要求。而这些消防设施中最基本、最重要的组成单元就是火灾自动报警系统。该系统主要技术是将传感器和单片机互相结合创新之后衍生的一项检测火灾的技术。这一项技术的主要优点在于其的综合性、组网灵活、布线简单、扩展性好等点。该设计是在嵌入式的基础上应用熟练的专业知识技能实现对室内环境及安全状况的实时监测监控并在本地及网络端实时监测显示。

### 1.2 火灾检测系统的国内外现状

中国以外的国家从很早就开始探索及发明火灾检测系统,在科技革命的推动下进展非常快。人们的警惕性增强和传感器市场增长是取得这样的成果的主要原因,据有关统计,美国在20世纪末到21世纪出的5年时间里烟雾传感器技术不断成熟。随着传感器生产技术水平日趋成熟、传感器逐渐简单化、灵敏度越来越高,使得火灾检测系统的体积也越来越小,使其能够更加方便投入到人们的生产和生活之中。我国对烟雾报警器的研究,功能全面,以前只是用于单一的工业领域,随着人们生活水平的提高,人们对生活质量的要求也逐步升高,警惕性越来越强,身边时时刻刻都在发生着各种由于没有及时检测到火灾的发生而导致一件件不幸的意外,但是这些意外是可以被避免的,因此火灾检测就应运而生到生活领域。我们在学习国外先进的传感器技术和先进生产技术的同时,进行发展和创新形成自己别具一格的特色。最近几年来,在烟雾灵敏性和产品可靠性上也取得了中大突破。

### 1.3 本论文主要任务

本课题研究的题材是基于在嵌入式平台上的火灾测系统。主要是构建嵌入式平台,在平台上利用嵌入式的特点完成对火灾检测系统的软件设计,并实现软件与硬件设备的对接,已达到对室内温度过高、燃气泄露等环境参数的监测和报警。该检测系统主要由温度和烟雾两大检测部分构成。

本次设计研究的核心是怎样高效率高精度度的实现火灾报警,提前预防火灾的发生,尽量不要发生错误报警的情况。其中在温度检测部分使用型号DS18B20的传感器,烟雾检测部分使用MQ-2型半导体烟雾传感器。如若设备接受到检测器传来的失火报警信息,报警器将发出失火报警。

## 第二章 背景知识介绍

### 2.1 烟雾传感器模块

#### 2.1.1 烟雾传感器介绍

烟雾传感器在人类社会生活和生产中不可或缺的器件,它是通过对烟雾浓度的高低来实现火灾预防和报警的。现今科随着学技术的不断创新使得市面上的烟雾传感器越发的多样化,但大都以离子式烟雾传感器为主流。由于离子式烟雾传感器灵敏度远远优于其它类型传感器,所以其被人们广泛的应用于各类火灾报警系统中。现代烟雾传感器的发展才刚刚起步,我相信随着科学技术的不断革新,将会涌现出更多类型的传感器。

#### 2.1.2 MQ-2 型烟雾传感器的工作原理

本设计中采用的烟雾传感器属于半导体式传感器,当发生火灾温度升高时,传感器中半导体材料与空气中的氧气作用形成氧负离子,结果,半导体材料中的电子密度降低,电阻增加。当传感器与烟雾接触时,刚开始吸附的氧与电子脱离,从而释放出大量电子。由于其中的电子密度的增加,促使阻值其降低。当空气中没有烟雾时,半导体材料将很快恢复氧的负离子吸附,并将阻值增加到原来的值。由此过程可感知烟雾是否存在的状态,并及时发出报警。图 2.1, 2.2 分别为该类型传感器的结构图和实物图。

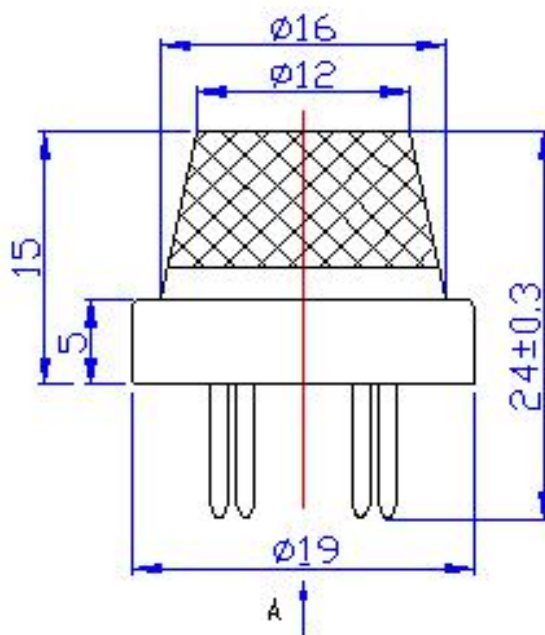


图 2.1 MQ-2 型传感器的结构图



图 2.2 MQ-2 型传感器的实物图

### 2.1.3 MQ-2 型传感器的特性

(1) MQ-2 型传感器的特点:

(a)MQ-2 传感器的工作能力很强,能够准确地检测出一些烷类可燃性气体。

(b)MQ-2 型传感器长期使用后其性能不会改变,并且其重复性好。

(c)MQ-2 型传感器能够准确排除外界干扰,精确性高。

(d)设计中有效电压范围很广。

(2) MQ-2 型传感器的基本特性

(a)灵敏度特性

烟雾传感器在最佳工作条件下暴露于相同的烟雾。其电阻值随气体浓度的变化而变化的现象即为传感器灵敏度的体现。用  $K$  表示灵敏度系数,用公式  $K=RS/R_0$  表示,其中  $R_0$  表示传感器在正常空气中的阻值, $RS$  是传感器在烟雾中的阻值。

(b)初始稳定性特征

当半导体烟雾传感器以功率状态存储一段时间时,该装置在通电之后不能立即工作。这是因为传感器中的半导体材料在不工作时会吸附空气中的水分子,工作时需要蒸发掉水分子,才可以保证传感器能够准确接受失火信号。我们把通电后传感器恢复到正常工作的这段时间称为初始稳定时间。

(c)加热特性

由于半导体烟雾传感器的工作环境通常在较高的环境下,如果将电热丝直接和电源连接在一起,可能会因为电热丝的短路以致设备温度高于安全工作温度而发生危险。因此,我们必须使用合理的加热电压使其在更安全的范围内工作。

## 2.2 温度传感器模块

### 2.2.1 DS18B20 的介绍

DS18B20 温度传感器是国外半导体公司通过引进先进的技术经过大量的实验和研究改良的一种温度传感器。与传统温度传感器相比,可直接反映出测量的实际温度,并可根据人们的需要通过简单软件语言实现十几位的数字读数方法。在图 2.3 中示出了在包封过程中的温度传感器的引脚布置,其各个管脚的功能在表 1 中描述。

表 1 DS18B20 详细引脚功能描述

引脚号	对应名称	管脚作用详细描述
1	GN D	接地信号
2	DQ	信号输入/输出引脚。打开和泄漏单总线接口引脚。
3	VD D	可替换的 VDD 引脚。当在寄生电源中应用时,此引脚必须和地相连。



图 2.3 DS18B20 (底视图)

### 2.2.2 温度传感器的控制方法

DS18B20 有多条控制指令,下面介绍几种主要的操作指令

CCH: 跳过扫描温度传感芯片序列号

44H: 启动 DS18B20 进行温度转换

BEH: 读温度值

## 2.3 单片机模块

AT89S52 的单片机和传统的其它单片机型号相比凸显出很多的优点,比如说有着不但不高且非常低的能耗,有良好的性能。这个型号的单片机芯片的内部结构设计有 8KB 的充足的存储空间,这样有助于单片机在进行编程时实现高速率的存储。还有这种型号的芯片以高密度非易失性存储的技术进行量产,它所用的指令与国际标准的 51 单片机命令集是可以相互兼容的。这种型号的单片机

内部的软件语言存储器非常灵活，支持重复在线编程，支持程序存储器在内部进行两种方式改写，一种是串行口改写，还有一种是共用的非易失性存储器改写。一般情况下，将 8 位中央处理器 CPU 和支持在线下载的 Flash 集成在一个芯片上，由此，就产生了 AT89S52 这样一个效率非常高的微型计算机。它的意义在于可以使其作用在更广泛的各种领域，更多的应用于电子产品设计领域，价格合理，可以解决很多不同场景下较为复杂的控制难题。其引脚图如图 2.4 所示

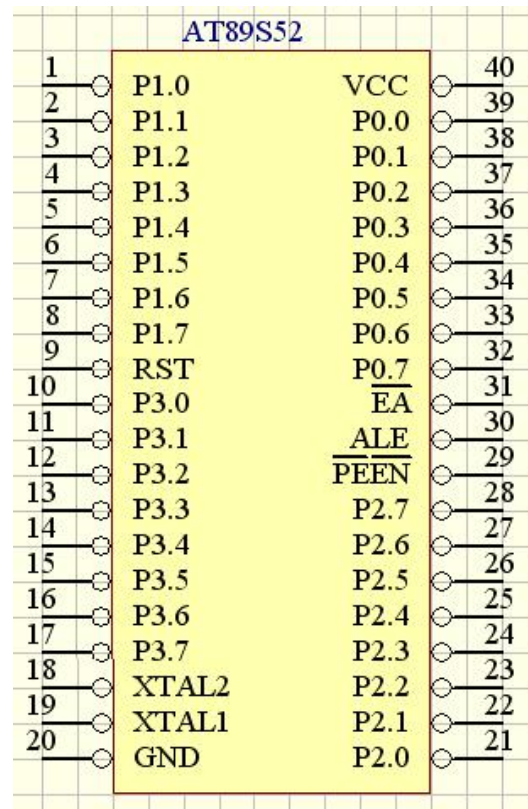


图 2.4 52 引脚图

### 2.3.1 主要功能特性

1. 多功能 I/O 口，四个 8 位的并行 I/O 口：P0、P1、P2、P3 端口。
2. RST 复位输入端。
3. ALE/PROG 地址锁存允许信号。
4. PSEN 程序存储器允许信号
5. EA/Vpp 外部存取允许信号。
6. XTAL1/XTAL2 振荡器的反相放大器输入/输出。

### 2.4 GPRS 系统

通俗来讲，GPRS 是通信领域的科技人才为了方便人们上网而研究出来的，它可以将人们想要发送的数据分成单个部分传输数据，提高了通信的效率，



而且收费也很合理，速度比 GSM 快很多。它有很多优点。

1. 广泛覆盖区域：GPRS 在全国 很多个地方都能良好的覆盖，并且它是全世界通行的 2G 通讯标准。基本上可以用手机通信的地方都可以通过 GPRS 上网。

2. 永久工作状态：只要启动 GPRS 系统应用后，在不人工关闭前提下它会一直保持工作状态的。

3. 按使用量收费：GPRS 服务虽然一直保持工作状态，但完全不用害怕会产生高额的费用，因为只有在通信过程中产生了流量才会计费。

4. 快速传输速率：目前 GPRS 可支持的传输速率高达 85.6KB ，并且通过不断地优化，其传输速率还有提高的空间。

5. 价格便宜：相对于 SIM 系列的模块 价格只有其的一半，大幅度的降低了通信设备的成本，其外观图如图 2.5 所示。

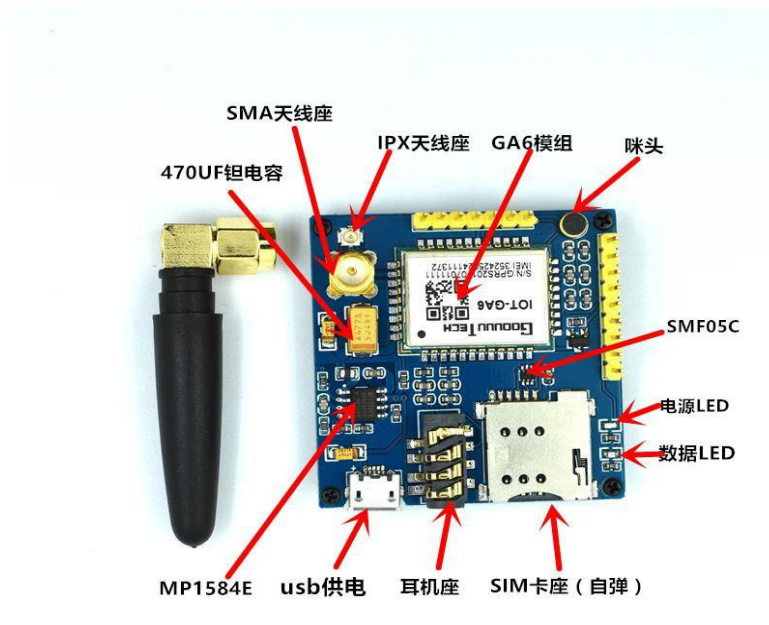


图 2.5 GPRS 外貌图

## 2.5 LCD 液晶显示屏介绍

### 2.5.1 LCD 的主要技术参数

显示容量为16×2个字符；

工作电压范围为4.5~5.5V；

工作电流为2.0mA（5.0V）；

字符长度 为2.95×4.35（W×H）mm。

### 2.5.2 LCD的引脚说明

LCD1602采用标准的14脚接口，其中：

1号脚：VSS为地电源。

2号脚：VDD接5V正电源。

3号脚：V0为液晶显示器对比度调整端。

4号脚：RS为寄存器选择，如果要选择数据寄存器的话在高电平时有效、选择指令寄存器则相反。

5号脚：RW为读写信号线，进行读取操作时其处于高电平状态，进行记录操作时处于低电平状态。当RS和RW共同处于低电平的状态可以录入指令亦或显示地址，当RS处于低电平状态而RW处于高电平状态时可以获取信号，当RS处于高电平状态而RW处于低电平状态时可以录入数据。

6号脚：E端为使能端，当此端电平状态由1变为0时，开始执行显示模块的指令。

7~14号脚：双向传输数据线。

15~16号脚：空脚

## 2.6 软件介绍

设计中软件工具选用Keil uvision4，是一家外国公司出品的兼容单片机C语言软件开发系统，这个工具现在非常成熟并且比较好用，对C语言来说是非常好的选择，具有方便的窗口设计，详细的功能库，方便用户编译和仿真调试，克服了很多传统工具的局限，用户学得快，编译效率也高。

### 第三章 系统方案设计

#### 3.1 系统功能

该系统使用单片机采集外界的温度和烟雾浓度, 并且温度的精确度很高, 运用 LCD 液晶显示屏显示温度的大小, 通过 GPRS 将数据输送到网络端, 设置三个按钮调整报警的温度。当温度超过所设定的最大值或烟雾浓度超过设定值时, 蜂鸣器接受到信号并且发出警报且指示灯闪烁。

#### 3.2 系统框图

图 3.1 为本次设计的系统框图。

#### 嵌入式火灾检测系统设计

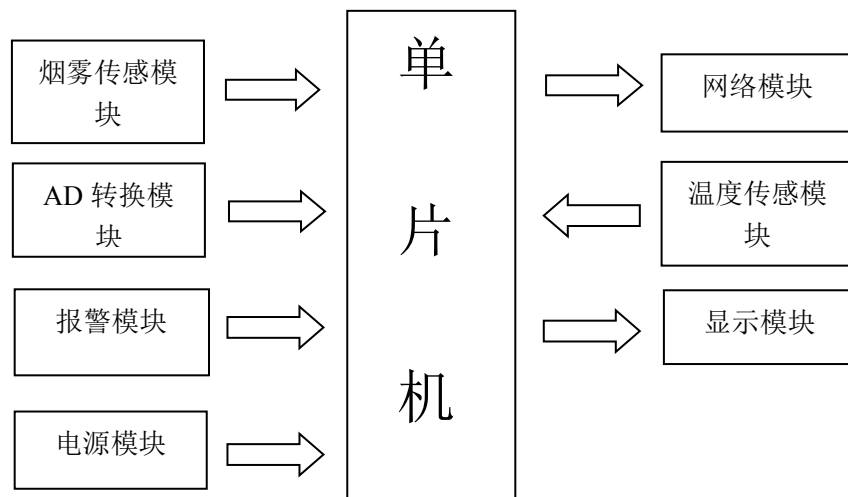


图 3.1 系统设计框图

本次设计的系统主要由传感器模块, 网络模块, 显示模块, 单片机控制模块, 电源模块, 报警模块及 AD 转换模块组成。传感器模块包括烟雾传感器和温度传感器模块, 检测室内空气浓度和室内温度, 是整个系统中最核心的部分; 由单片机控制整个系统数据的采集; 网络模块负责将数据发送到网络端 (如电脑, 数据等通讯设备); 显示模块则显示本地数据, AD 转换模块负责将传感器的模拟数据转换为单片机可以识别的数字信号, 报警模块负责整个系统的异常报警, 电源负责给整个系统供电。

## 第四章 硬件设计

### 4.1 单片机最小系统电路

单片机的最小系统亦可以称之为最小应用系统，可以理解为用最简单的基本器件相结合实现单片机能够稳定的运行，且抗干扰能力比较强。对于型号为 52 的单片机所具有的最小系统通常包括晶振电路、复位电路等。

#### 4.1.1 复位电路

单片机的复位电路在整个单片机系统里面有着举足轻重的地位。它的主要作用在于当系统在运行过程中受到不良因素的影响导致运行错误或者导致程序运行偏离正常的路径时，按下复位按钮其内部的程序会主动重新开始运行避免因为程序跑飞使系统瘫痪。

#### 4.1.2 晶振电路

晶振晶是指体振荡器，它由两部分组成，一部分为电容和电阻并联，另一部分为串联的电容形成的网络。该网络通过频率的高低分为并联、串联谐振。虽然分为高、低频率可是因为晶体本身的性质导致这两个频率相差并不是很大。因此在这个频率允许的范围内晶振可以看做为电感，在其两边加上适合的电容将其改变成为并联谐振电路，该电路通过与负反馈电路的结合可以构成正弦振荡电路，因为频率相差不是很大所以这个振荡器的频率受其它参数的影响是不会很大的。负载电容值是一个特别重要的技术指标，该值的大小通常是接到晶振的两端的电容串联值。

#### 4.1.3 P0 口的上拉电阻

如若 P0 口作为输出的时候在低电平有效时输出 0，但在高电平时需要加一个上拉电阻才能使其给负载提供足够的电流。下面举几个例子：

- 1.通用 51 单片机的 P0 口在作为地址/数据复用时无需接上拉电阻。
- 2.一般情况下使用 I/O 口时，由于内部没有上拉电阻，故要接上上拉电阻。
- 3.当 p0 口用来驱动 PNP 三极管的时候，低电平有效，不需要接入上拉电阻。

图 4.1 是单片机的最小系统。

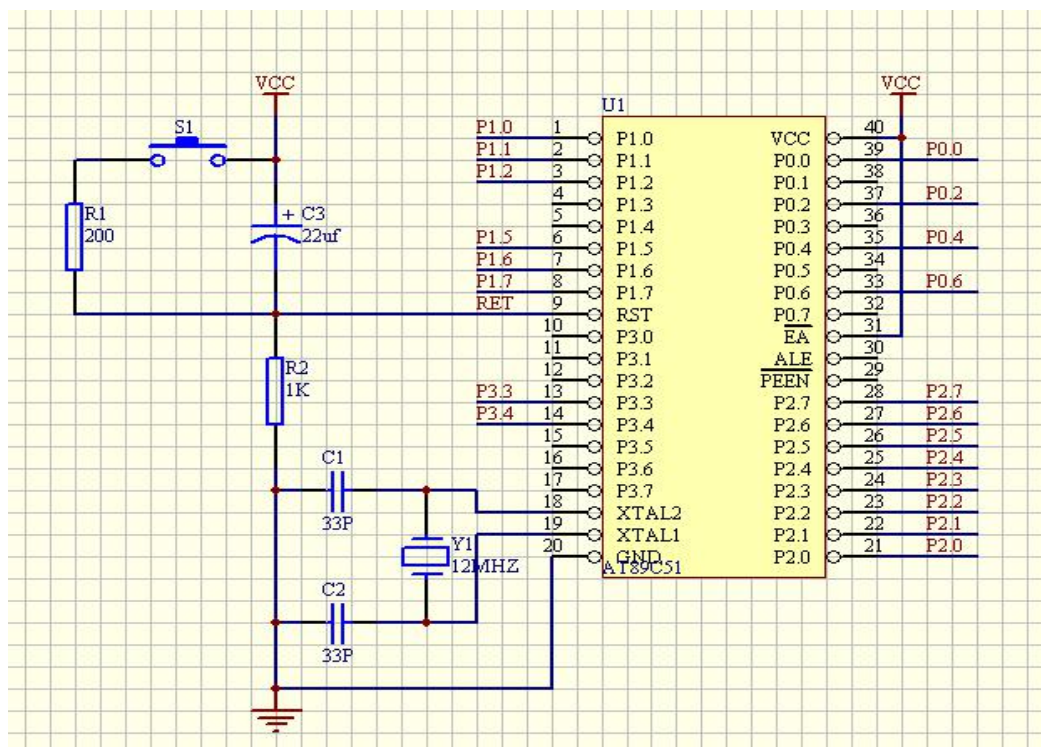


图 4.1 单片机最小系统

- (1) 9 号引脚连接复位电路，可通过按按钮 S1 使单片机进行复位。
- (2) 晶振采用 12MHZ。
- (3) 由于单片机只访问片内 Flash ROM 并执行内部程序存储器中的指令，因此单片机的 31 脚接高电平 VCC。

## 4.2 温度检测模块-DS18B20 测温电路

在实际操作与运用中应该注意以下几点事项：

- (1) 在读编程和写编程的时候，因为温度传感器以串行的方式和微处理器连接，因此，一定要严格把关读写程序，如果不能做到这一点测试者就得不到正确的测温数据。
- (2) 测试者需要根据实际情况，也就是在设计测量温度电路的时候，会遇到大多数人遇到的问题，即微处理器的总线驱动这个问题。
- (3) 和 DS18B20 相连的总线是有长度的。在用 DS18B20 进行远距离测温系统设计时要时刻注意总线分布等问题。
- (4) 在 DS18B20 测温软件程序设计中，在执行温度转换的指令后，程序要接受到返回信号，如果接触有问题或者断路，将使返回信号无法被接收，从而导致程序进入死循环。图 4.2 是 DS18B20 电路。

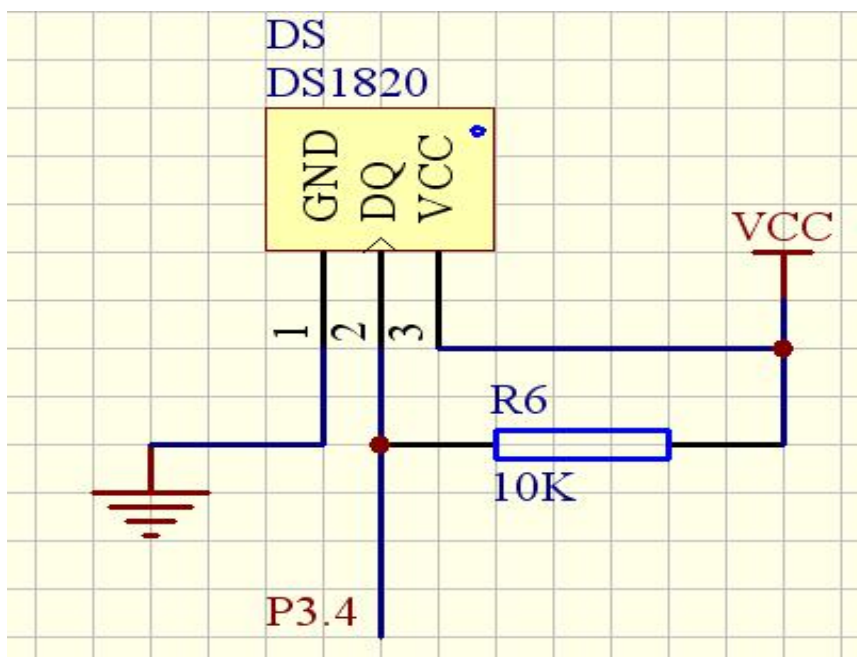


图 4.2 DS18B20 电路

DS18B20 的 1 号引脚接地，2 号引脚数据端接单片机的 P3.4，3 号引脚接 VCC，为了确保 DS18B20 工作在准确的环境下，2 号脚需要接合理阻值的电阻。

### 4.3 报警模块

#### 4.3.1 蜂鸣器报警电路

图 4.3 为蜂鸣器报警电路。

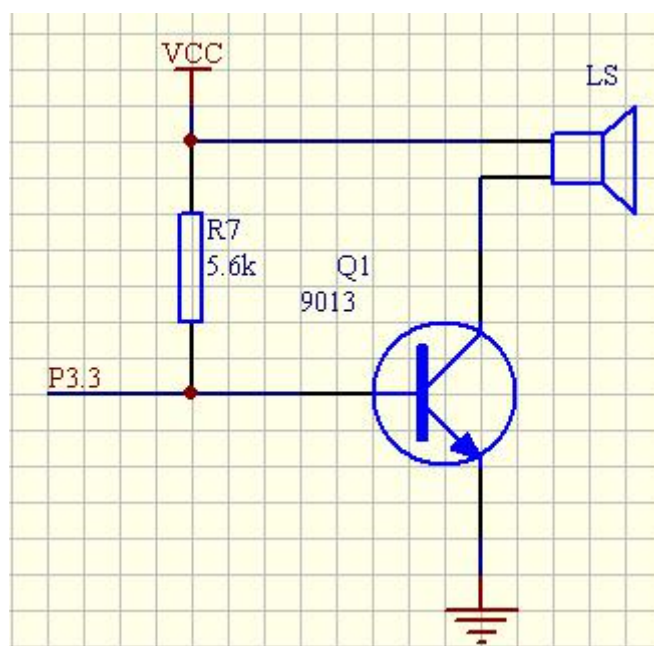


图 4.3 蜂鸣器报警电路

本次报警模块选用蜂鸣器报警，因为单片机输出的电流不足以驱动报警，所以使用三极管对蜂鸣器进行驱动并使其出声。

整个报警电路由蜂鸣器，三极管，电阻组成，蜂鸣器有正负两个管脚，正极接 VCC，负极接三极管发射级，基级接单片机 P3.3，另外一端接地。

### 4.3.2 按键电路

图 4.4 为报警温度设定按键电路。

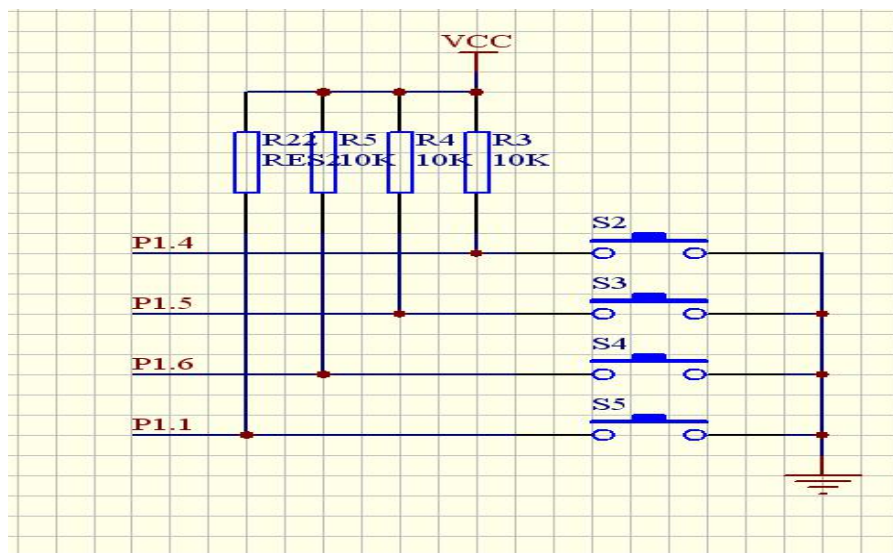


图 4.4 按键电路

报警温度的设置通过按键调整，分别为 S2，S3，S4. S2 是选择按钮，按第一次就可以选择设置温度报警的最小值，按第二次就可选择设置温度报警的最大值，按下第三次则使其数值显示为正常温度。S5 是紧急情况按钮。

分别将 S2，S3，S4，S5 的 1 管脚接地，3 管脚依次接单片机的 P1.4，P1.5，P1.6，P1.1。

### 4.4 显示模块-液晶显示电路

本系统的显示器采用型号为 LCD1602 显示器，该显示器比数码管的硬件连接和软件调节等方面均具有很大的优势。只需要把所显示的内容放入显示存储器中即可直接的显示出指定的内容，操作上更加的简便和快捷方便。图 4.5 为显示模块接线图。



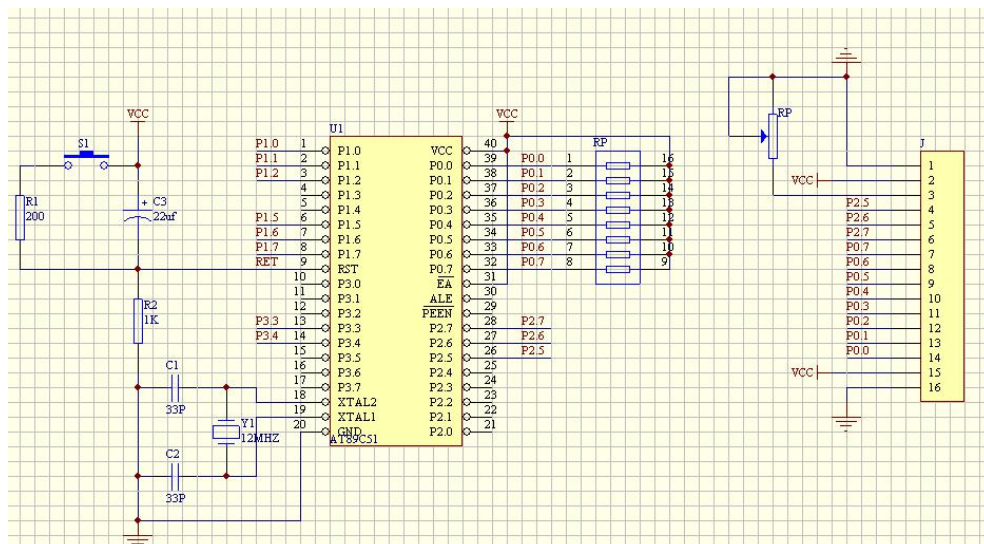


图 4.5 显示模块接线图

液晶显示屏的 1 管脚接到 GND, 2 管脚接 VCC, 3 管脚用来调整液晶的灰度, 大多数情况下接 10k 电位器, 14 管脚接单片机的 I/O 口, 15 管脚为背光的电源, 16 管脚为背光的地。

## 4.5 烟雾检测模块

图 4.5 为烟雾传感器转换电路

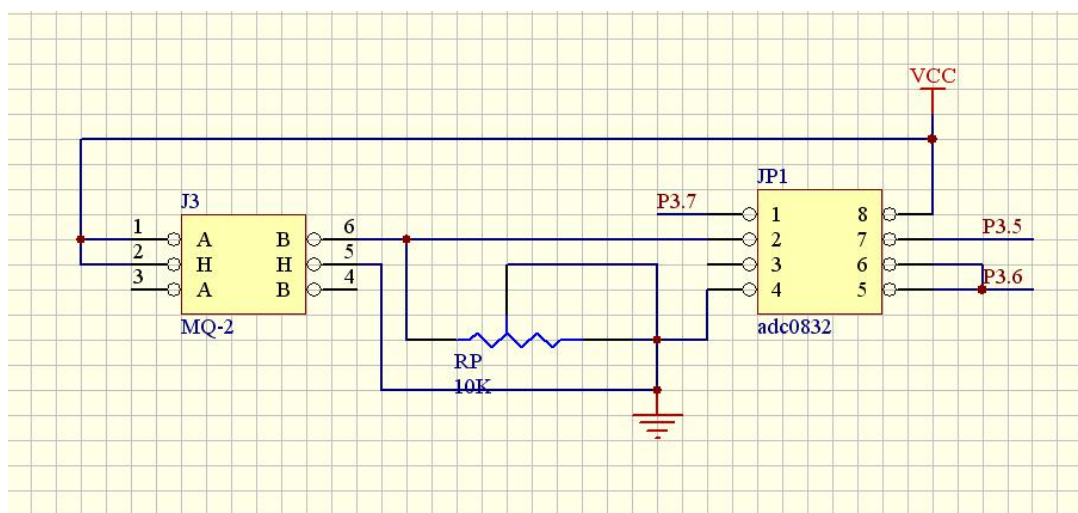


图 4.5 烟雾传感器转换电路

烟雾传感器 MQ-2 的烟雾浓度数据需要通过 AD0832 转换模块处理后才能输送给单片机的接口。

## 4.6 网络模块

网络模块负责将采集的温度值和烟雾浓度值发送到服务器, 由服务端显示数



据并且进行分析，图 4.6 为 GPRS 电路图。

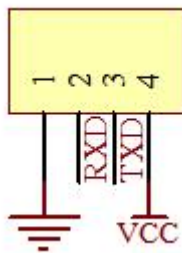


图 4.6 GPRS 电路图

GPRS 的 1 管脚接地，2 管脚接单片机的 RXD,3 管脚接单片机的 TXD，4 管脚接 VCC。

#### 4.7 本章小结

在这一章主要介绍了本次设计的各个硬件模块以及每个硬件的接线情况和每一个模块在整个系统中承担着什么样的角色起着什么样的作用。硬件电路设计还是比较麻烦的事情，要选择什么样的器件来实现什么样的功能，各个模块之间的连接方法，都需要对每个模块的各个管脚很清楚，而且还要有一定的焊接电路板的功底，本人的硬件电路比较复杂，千万注意焊接的时候一定要按层次划分，检查每一根线是否有断开或者短路的情况，学会用万用表检查电路故障。

## 第五章 软件设计

### 5.1 软件下载和安装

如图 5.1 所示。



图 5.1 软件下载

### 5.2 使用教程

- 5.2.1 建立工程
- 5.2.2 建立源文件
- 5.2.3 添加源文件到工程里并键入程序
- 5.2.4 编译
- 5.2.5 调试

### 5.3 温度检测模块

该模块实现了温度检测的功能，当实际温度值超过自己定的上下限，会启动报警程序，体现人们发生火灾了。

#### 5.3.1 温度检测模块流程图

图 5.2 为温度检测模块的流程图，初始化完成后启动温度转换模块，发生延时后进行数据处理判断，同时把数据输送到显示模块。

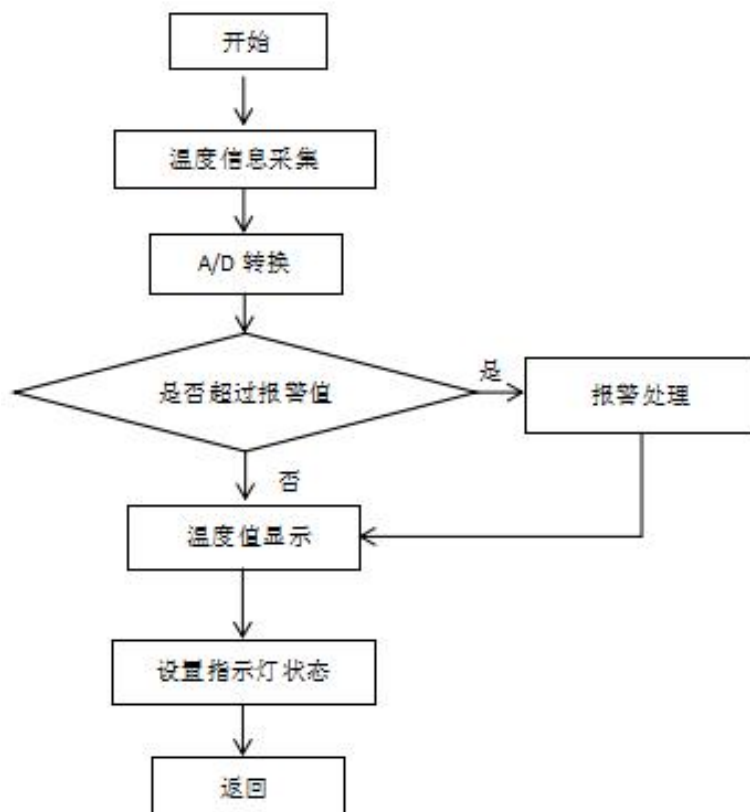


图 5.2 温度检测模块流程图

## 5.4 烟雾检测模块

由于烟雾传感器在很长时间不用的状态下再次进入工作状态时它会有一个缓冲期，不能立即检测烟雾浓度，所以要先预热一端时间，然后控制系统对传感器检测到的烟雾浓度信号作 A/D 转换，取中间值过滤，一次函数化操作后，把采集到的浓度值与程序里面设的值相比较，决定是不是要报警，并且把数值输送到液晶显示屏上，这样就可以实现本地显示了。

### 5.4.1 烟雾检测流程图

图 5.3 为烟雾检测模块的流程图

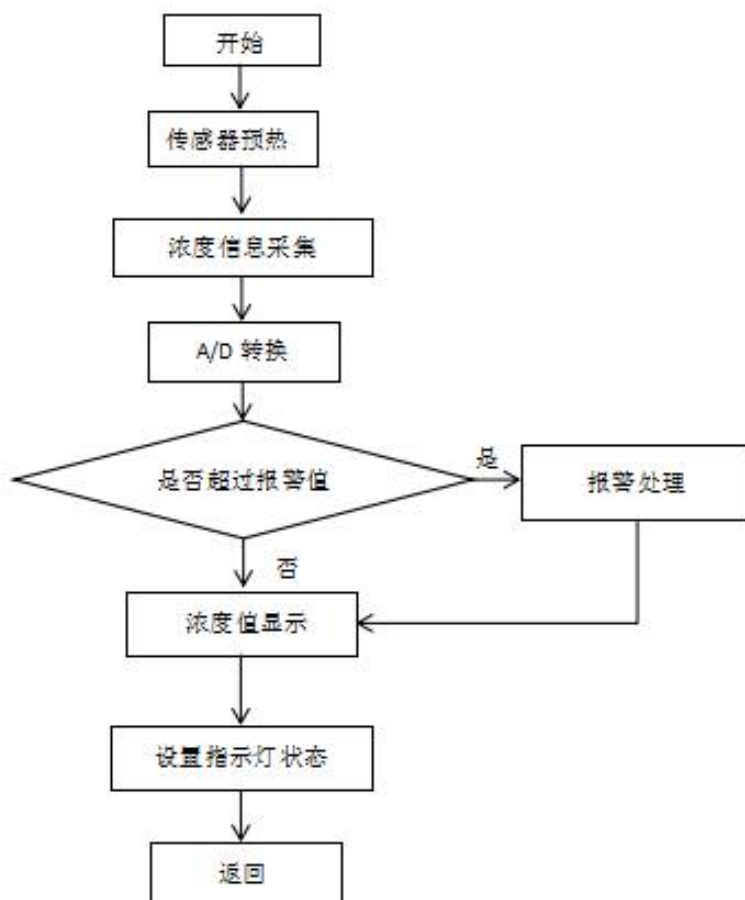


图 5.3 烟雾检测模块流程图

## 5.5 A/D 转换模块

这个模块的功能就是将模拟信号转换为数字信号的过程。

### 5.5.1 A/D 转换模块流程图

图 5.4 为 A/D 转换模块流程图

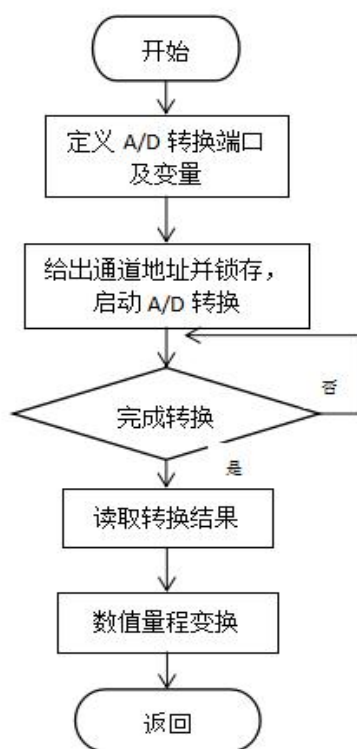


图 5.4 A/D 转换模块流程图

## 5.6 显示模块

### 5.6.1 LCD 液晶显示流程图

LCD 液晶显示流程图如图 5.5 所示

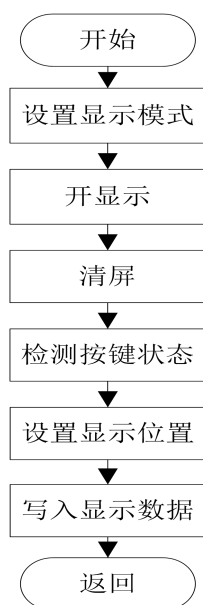


图 5.5 LCD 液晶显示流程图

## 5.7 网络模块

### 5.7.1 GPRS 模块流程图

图 5.6 为 GPRS 模块流程图。

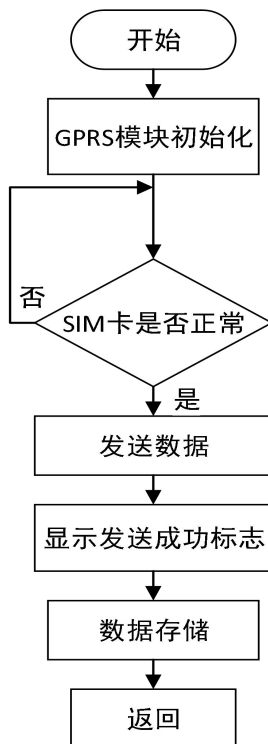


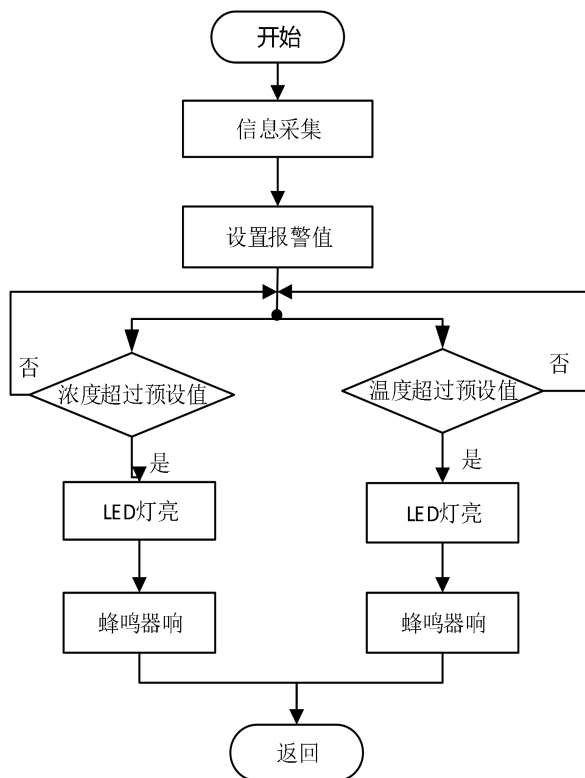
图 5.6 GPRS 模块流程图

## 5.8 报警模块

当烟雾浓度值超过本身定好的值，蜂鸣器会发出声音以提示人们做好安全措施，避免发生一些令人叹息的不安全事件。在软件语言设计中采用很短的时间恢复检测并且将报警延迟一段时长。这样做的目的是为了避免报警器报错，如果只是有人只是打开阀门的一瞬间检测到了烟雾，那这种情况就属于误报。

### 5.8.1 报警模块流程图

图 5.7 为报警模块流程图



5.7 报警模块流程图

## 5.9 本章小结

在本章主要介绍了软件设计这一部分，描述了每一个模块的流程图和主要程序，这个完全考验我的逻辑思维能力，要通过什么样的语言模块来实现每一个功能，还要考虑到每一种情况下该执行哪一个行为语句，这个一定要对自己要实现和设计的东西非常熟悉，换一种说法其实就是把硬件设计中的电路连接用软件语言翻译过来，从另一个层次实现系统的功能。

## 第六章 系统测试及调试

### 6.1 系统调试

在硬件和软件部分的工作全部做完之后，当然还要对系统进行调试，整个调试的过程中会对整个设计有更深入的理解，首先要具有分析问题的思路和基本方法，主要分为两部分，硬件和软件。

硬件：

主要还是硬件电路焊接的问题比较多，而且实际过程中的导线非常细很容易断掉，这就要根据硬件设计电路图用万用表一步一步测量最有可能出问题的模块的管脚焊接是否有问题，或是虚焊或是短路，一开始当温度值超过设定温度值时，蜂鸣器却没有报警，那最有可能是蜂鸣器的管脚焊接有问题，对照电路图检查电路连接，果然是蜂鸣器的 VCC 没有接地，其实还有很多这样类似的硬件电路问题，都逐个排查就好了。

软件：

软件部分主要是查看中断程序执行是否有误，调试过程中也出现了 LCD 液晶显示屏无法显示本地数据，在检查硬件电路没有问题之后，在软件中找到了答案，是因为 LCD 初始化显示的子程序里面没有加延迟，因为有时候会发生上电失败的情况，导致初始化失败从而 LCD 无法显示，在 LCD 的初始化程序里面加上延迟就好了。修改过程如图 6.1 所示。

```
lcd_init();//初始化显示  
delaylms(100);  
lcd_init();//初始化显示
```

图 6.1 修改过程

### 6.2 烟雾报警测试

#### 6.2.1 测试目的

制造一定烟雾浓度的环境，当超过设定浓度级别时发生报警并且在 LCD 显示屏及网络端上显示实时数据。这样人们就可以提前预知火灾发生，做好相应的防护措施，将伤害减少到最小。

#### 6.2.2 测试结果

当报警设备处于正常洁净空气下的环境时，LCD 显示烟雾浓度为 0 级，无任何报警提示，本地结果显示如图 6.2 所示。



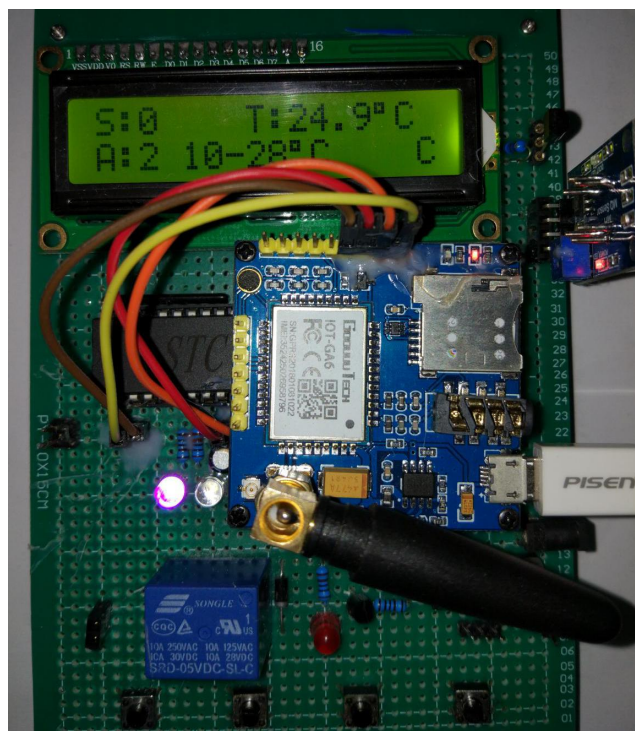


图 6.2 本地结果显示

当报警设备处于一定浓度下的烟雾环境时，蜂鸣器发出嘀嘀嘀的声响，同时报警灯亮。

本地显示如下图 6.3 所示。



图 6.3 本地显示

网络端显示如图 6.4 所示

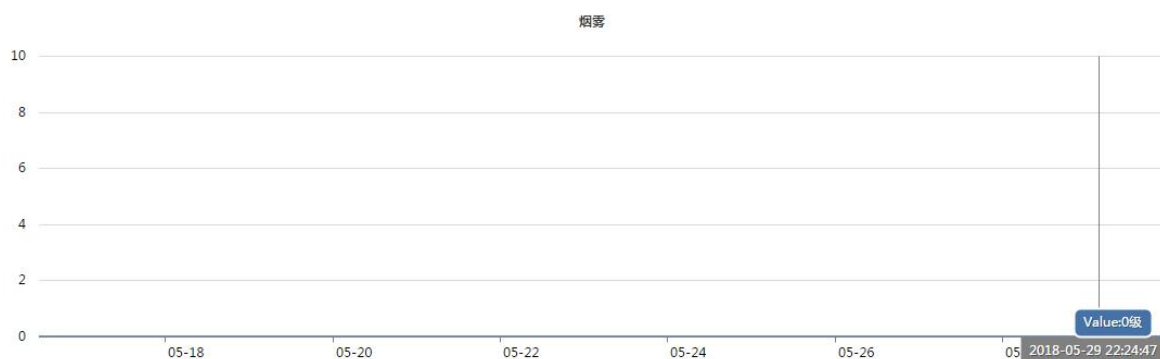


图 6.4 网络端显示

## 6.3 温度报警测试

### 6.3.1 测试目的

当发生火灾时，室内温度升高，通过温度传感器检测温度通过本地显示并且通过网络模块将数据传输到网络端。防止火灾给人们带来的损失，保护人生安全和财产安全。

### 6.3.2 测试结果

当设备处于正常室温环境下时，本地在 LCD 屏幕上显示实时温度，同时在网络端显示实时数据。测试结果如图 6.5 所示。

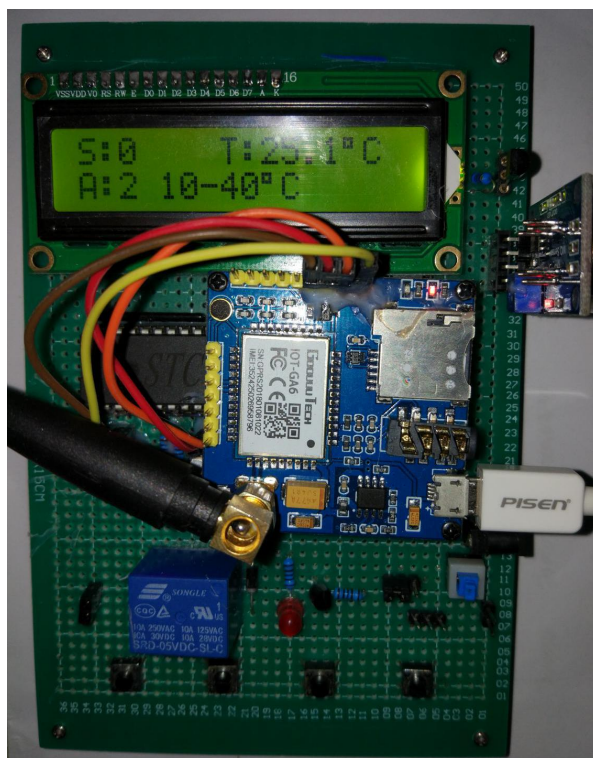


图 6.5 测试结果

当设备处于异常室温的情况下时，在本地和网络端显示实时数据，并且蜂鸣器报警且指示灯闪烁。测试结果本地显示如图 6.6 所示。

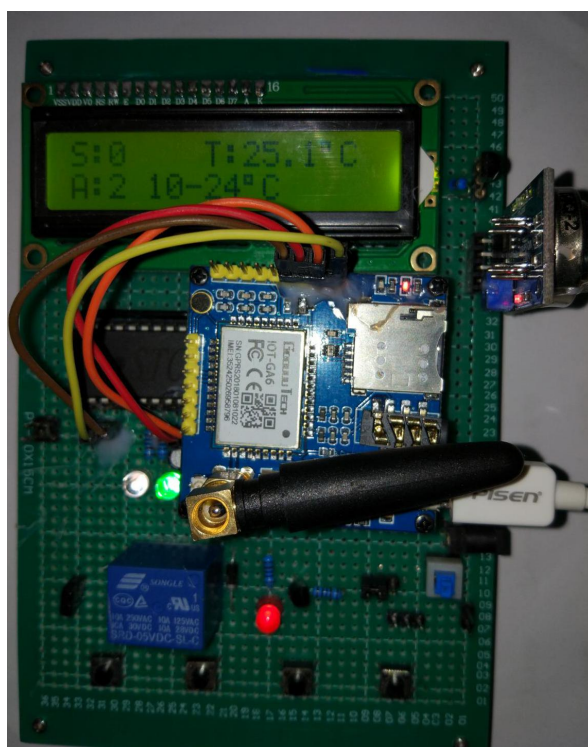


图 6.6 本地显示

测试结果在网络端显示结果如图 6.7 所示



图 6.7 网络端显示

## 6.4 实验结果分析

从上面的结果图可以很明显的看到网络端的实时显示数据和本地显示数据是存在一定的误差的，主要的误差和软件与硬件都有关系。

从软件方面来讲的主要原因是软件设计的过程中采用中值平均数字滤波的时候用到了基本的乘除运算，那么在这个过程中由于不能够保证数据的每一位有效位，难免会出现舍入误差，可能一次计算并不会造成误差，但是这是一个累计计算的过程，必然会引起误差。

从硬件方面来讲就有很多造成误差的原因了，下面列举了一些造成误差的主要原因

(1) 本设计中采用的 MQ-2 型烟雾传感器，烟雾浓度和输出电压存在着非线性的关系，所以采用曲折线插值法来减少误差。

(2) 由于传感器输出的信号一般都很微弱，我们要通过运放电路将其放大调整后再输入单片机，在此过程中由于本来运送到放大电路的信号就有误差，在经过放大电路，误差就会很明显。还可以在旁边加一些设计需要的阻容器件组合起来，调整以后就可以减小误差。

(3) 外部环境也是一个重要的因素，空气中的尘埃颗粒等杂物影响并不大，但是温度起着一定的干扰作用，只要在设计电路时加上抗干扰布局就好了。

## 6.5 本章小结

在本章中主要描述了完成软硬件的设计在调试以及测试的过程中出现的的问题，其次介绍了由于硬件电路和软件语言算法引起到的问题的检测结果不准确，详细分析了为什么会出现这样的误差，以及应该采取什么样的措施来避免这样的误差。到最后把误差程度降到了最低，达到预期的实验结果。

## 结束语

经过四年的学习，终于完成了我的烟雾报警的设计，虽然没有完全达到设计要求，但从心底里说，还是比较高兴的。过程当中用到的单片机作为控制内核，LCD 液晶作为显示部分，及其他电路，共同组成了我的硬件系统嵌入式火灾检测系统。首先设计的是它的硬件电路，最重要的部分是 89S52 控制内核，所有的数据处理都是采用的单片机，其次是 DS18B20 温度采集电路，其他还有晶振电路，复位电路，报警点及上下限温度调整电路。其次我们设计了它的灵魂软件电路，通过用 C 语言编程实现对器件的控制。

从这次的课程设计中，我才真正的意识到，在以后的学习中，要理论联系实际，把我们所学的理论知识用到实际当中，不仅使我真正的学会了数字温度计的设计，而且我相信通过这次的学习我能够达到举一反三的效果，同时这次的不足之处是在有些细节方面刚开始做的很不详细，我想原因在于自己平时对自己的学习要求的不够严格，才造成遇到一些问题显得惊慌失措，在日后我会克服这些缺点的。

## 致谢

经过几个月的忙碌和工作，本次毕业设计已经接近尾声，作为一个本科生的毕业设计，由于经验的匮乏，难免有许多考虑不周的地方，如果没有导师的督促指导。以及一起工作的同学们的支持，想要完成这个设计是难以想象的。

在这里要感谢李育贤老师，她循循善诱的教导和不拘一格的思路给予我无尽的启迪；她细致，一丝不苟的作风一直是我们的工作、学习中的榜样；并将积极影响我今后的学习和工作。然后要感谢大学 3 年来的所有老师，为我们打下了自动化专业知识的基础；同时还要感谢 14 级自动化学院所有的同学们，正是因为有你们的支持和鼓励，我才能在温暖的环境中茁壮成长并顺利完成毕业设计。

我的毕设作品不是很成熟，还有很多不足之处。但是这次做论文的经历使我终身受益。我感受到做论文是要真真正正用心去做的一件事情，是真正的自己学习的过程和研究的过程，没有学习就不可能有研究的潜力，没有自己的研究，就不会有所突破，那也就不叫论文了。期望这次的经历能让我在以后学习中激励我继续进步。

## 参考文献

- [1] 李全钊,单片机原理及接口技术(第2版),高等教育出版社,2009年
- [2] 李广弟,单片机基础,北京航空航天大学出版社,1994
- [3] 阎石,数字电子技术基础(第三版),高等教育出版社,1989
- [4] 石东海等,单片机数据通信技术从入门到精通,西安电子科技大学出版社,2002
- [5] 张毅刚,彭喜源,谭晓昀等,MSC-51单片机应用设计,哈尔滨工业大学出版社,1999
- [6] 周坚.单片机C语言轻松入门,北京航空航天大学出版社,2006
- [7] 王曙燕,C语言程序设计教程,人民邮电出版社,2009
- [8] K. Vardar, E. Akpınar, T. Sürgevil. Evaluation of reference current extraction methods for DSP implementation in active power filters[J]. Electric Power Systems Research, 2009, 79(10).
- [9] D. Hazarika, D.S. Pegu. Micro-controller based air pressure monitoring instrumentation system using optical fibers as sensor[J]. Optical Fiber Technology, 2013, 19(2).
- [10] Chun Ru Xiong, Fu Zhen Xie. Application about Proteus Simulation on SCM Teaching[J]. Applied Mechanics and Materials, 2014, 3207(556).



## 附录

### 附录 A 程序源代码:

```
#include <AT89X52.h>
#include <intrins.h>
#include "DS18B20.h"
#define uint unsigned int
#define uchar unsigned char //宏定义
#define SET P1_4 //定义调整键
#define DEC P1_5 //定义减少键
#define ADD P1_6 //定义增加键
#define BEEP P3_4 //定义蜂鸣器
#define hujiao P1_1
sbit ADCS = P3^7;
sbit ADCLK = P3^5;
sbit ADDI = P3^6;
sbit ADDO = P3^6;
bit shanshuo_st; //闪烁间隔标志
bit beep_st; //蜂鸣器间隔标志
bit flag=0; //紧急呼叫标志
sbit DIAN = P2^5; //小数点
uint abc;
uchar x=0; //计数器
signed char m; //温度值全局变量
uchar n; //温度值全局变量
uchar set_st=0; //状态标志
signed char shangxian=30; //上限报警温度，默认值为 38
signed char xiaxian=5; //下限报警温度，默认值为 38
uchar nongdu=1;
uchar code
LEDData[]={0x28, 0xeb, 0x32, 0xa2, 0xe1, 0xa4, 0x24, 0xea, 0x20, 0xa0};
/*****延时子程序*****/
void Delay(uint num)
{
while( --num );
}
/*****初始化定时器 0*****/
void InitTimer(void)
{
TMOD=0x1;
TH0=0x4c;
```

```

TL0=0x00;      //50ms（晶振 11.0592M）
}
/*****定时器 0 中断服务程序*****/
void timer0(void) interrupt 1
{
    TH0=0x4c;
    TL0=0x00;
    x++;
}
/*****读取温度*****/
void check_wendu(void)
{
    uint a, b, c;
    c=ReadTemperature()-5; //获取温度值并减去 DS18B20 的温漂误差
    a=c/100;      //计算得到十位数字
    b=c/10-a*10;  //计算得到个位数字
    m=c/10;       //计算得到整数位
    n=c-a*100-b*10; //计算得到小数位
    if(m<0){m=0;n=0;} //设置温度显示上限
    if(m>99){m=99;n=9;} //设置温度显示上限
}
/*****显示开机初始化等待画面*****/
Disp_init()
{
    P2 = 0xf7;      //显示-
    P0 = 0xbf;
    Delay(200);
    P0 = 0xef;
    Delay(200);
    P0 = 0xfb;
    Delay(200);
    P0 = 0xfe;
    Delay(200);
    P0 = 0xff;      //关闭显示
}
/*****显示温度子程序*****/
Disp_Temperature() //显示温度
{
    P2 =LEDData[m%10]; //显示 C
    P0 = 0xbf;

```

```

Delay(300);
P2 =LEDData[m/10];    //显示个位
P0 = 0xef;
Delay(300);
P2 =0xf7;    //显示十位
P0 = 0xfb;
Delay(300);
P2 =LEDData[abc];    //显示百位
P0 = 0xfe;
Delay(300);
P0 = 0xff;    //关闭显示
}
/*****显示报警温度子程序*****/
Disp_alarm(uchar baojing)
{
P2 =0x3c;    //显示 C
P0 = 0xbf;
Delay(200);
P2 =LEDData[baojing%10]; //显示十位
P0 = 0xef;
Delay(200);
P2 =LEDData[baojing/10]; //显示百位
P0 = 0xfb;
Delay(200);
if(set_st==1)P2 =0x61;
else if(set_st==2)P2 =0x3d; //上限 H、 下限 L 标示
P0 = 0xfe;
Delay(200);
P0 = 0xff;    //关闭显示
}
Disp_nongdu(uchar baojing)//烟雾浓度报警显示
{
P2 =0xff;
P0 = 0xbf;
Delay(200);
P2 =0xff;
P0 = 0xef;
Delay(200);
P2 =0xff;
P0 = 0xfb;

```

```

Delay(200);
P2 =LEDData[baojing]; //显示浓度
P0 = 0xfe;
Delay(200);
P0 = 0xff;           //关闭显示
}
/*****报警子程序*****/
void Alarm()
{
if(x>=10) {beep_st=~beep_st;x=0;}
if((m>=shangxian&&beep_st==1) || (m<xiaxian&&beep_st==1))BEEP=1;
else if(abc>=nongdu&&beep_st==1) BEEP=1;
else BEEP=0;
}
void Alarm1()
{
if(x>=10) {beep_st=~beep_st;x=0;}
if(beep_st==1)BEEP=1;
else BEEP=0;
}
uchar ADC0832(bit mode,bit channel)    //AD 转换，返回结果
{
    uchar i, dat, ndat;

    ADCS = 0; //拉低 CS 端
    _nop_();
    _nop_();

    ADDI = 1; //第 1 个下降沿为高电平
    ADCLK = 1; //拉高 CLK 端
    _nop_();
    _nop_();
    ADCLK = 0; //拉低 CLK 端, 形成下降沿 1
    _nop_();
    _nop_();

    ADDI = mode; //低电平为差分模式，高电平为单通道模式。
    ADCLK = 1; //拉高 CLK 端
    _nop_();
    _nop_();
}

```

```

ADCLK = 0;//拉低 CLK 端, 形成下降沿 2
_nop_();
_nop_();

ADDI = channel; //低电平为 CH0, 高电平为 CH1
ADCLK = 1;//拉高 CLK 端
_nop_();
_nop_();
ADCLK = 0;//拉低 CLK 端, 形成下降沿 3

ADDI = 1;//控制命令结束(经试验必需)
dat = 0;
//下面开始读取转换后的数据, 从最高位开始依次输出 (D7~D0)
for(i = 0;i < 8;i++)
{
    dat <<= 1;
    ADCLK=1;//拉高时钟端
    _nop_();
    _nop_();
    ADCLK=0;//拉低时钟端形成一次时钟脉冲
    _nop_();
    _nop_();
    dat |= ADD0;
}
ndat = 0; //记录 D0
if(ADD0 == 1)
ndat |= 0x80;
//下面开始继续读取反序的数据 (从 D1 到 D7)
for(i = 0;i < 7;i++)
{
    ndat >>= 1;
    ADCLK = 1;//拉高时钟端
    _nop_();
    _nop_();
    ADCLK=0;//拉低时钟端形成一次时钟脉冲
    _nop_();
    _nop_();
    if(ADD0==1)
    ndat |= 0x80;
}

```

```

    ADCS=1;//拉高 CS 端, 结束转换
    ADCLK=0;//拉低 CLK 端
    ADDI=1;//拉高数据端, 回到初始状态
    if(dat==ndat)
        return(dat);
    else
        return 0;
}
/*****主函数*****/
void main(void)
{
    uint z;
    InitTimer();    //初始化定时器
    EA=1;           //全局中断开关
    TR0=1;
    ET0=1;          //开启定时器 0
    BEEP=0;
    check_wendu();
    check_wendu();
    for(z=0;z<300;z++)
    {
        Disp_init();
    }
    while(1)
    {
        if(hujiao==0)
        {
            Delay(2000);
            do{}while(hujiao==0);
            flag=~flag;
        }
        if(SET==0)
        {
            Delay(2000);
            do{}while(SET==0);
            set_st++;x=0;shanshuo_st=1;
            if(set_st>3)set_st=0;
        }
        if(set_st==0)
        {

```

```

        abc = ADC0832(1,0); //差分模式, CH0-CH1
        abc = abc*19.607843; //转换为实际电压便于显示
        abc=abc/1000%10;
        check_wendu();
        Disp_Temperature();
    if(flag==1)Alarm1();
    else Alarm(); //报警检测
}
else if(set_st==1)
{
    BEEP=0; //关闭蜂鸣器
    if(DEC==0)
    {
        Delay(2000);
        do {} while(DEC==0);
        shangxian--;
        if(shangxian<xiaxian) shangxian=xiaxian;
    }
    if(ADD==0)
    {
        Delay(2000);
        do {} while(ADD==0);
        shangxian++;
        if(shangxian>99) shangxian=99;
    }
    if(x>=10) {shanshuo_st=~shanshuo_st;x=0;}
    if(shanshuo_st) {Disp_alarm(shangxian);}
}
else if(set_st==2)
{
    BEEP=0; //关闭蜂鸣器
    if(DEC==0)
    {
        Delay(2000);
        do {} while(DEC==0);
        xiaxian--;
        if(xiaxian<0) xiaxian=0;
    }
    if(ADD==0)
    {

```

```

        Delay(2000);
        do {} while (ADD==0);
        xiaxian++;
        if(xiaxian>shangxian) xiaxian=shangxian;
    }
    if(x>=10) {shanshuo_st=~shanshuo_st;x=0;}
    if(shanshuo_st) {Disp_alarm(xiaxian);}
}
else if(set_st==3)
{
    BEEP=0;    //关闭蜂鸣器
    if(DEC==0)
    {
        Delay(2000);
        do {} while (DEC==0);
        if(nongdu>0)
            nongdu--;
        if(nongdu<1) nongdu=0;
    }
    if(ADD==0)
    {
        Delay(2000);
        do {} while (ADD==0);
        nongdu++;
        if(nongdu>5) nongdu=5;
    }
    if(x>=10) {shanshuo_st=~shanshuo_st;x=0;}
    if(shanshuo_st) {Disp_nongdu(nongdu);}
}
}
}
/*****END*****/
#include <AT89X52.h>
#define DQ P1_0    //定义 DS18B20 总线 I/O
/*****延时子程序*****/
void Delay_DS18B20(int num)
{
    while(num--);
}
/*****初始化 DS18B20*****/

```



```

void Init_DS18B20(void)
{
    unsigned char x=0;
    DQ = 1;          //DQ 复位
    Delay_DS18B20(8); //稍做延时
    DQ = 0;          //单片机将 DQ 拉低
    Delay_DS18B20(80); //精确延时，大于 480us
    DQ = 1;          //拉高总线
    Delay_DS18B20(14);
    x = DQ;          //稍做延时后，如果 x=0 则初始化成功，x=1 则初始化失败
    Delay_DS18B20(20);
}
/*****读一个字节*****/
unsigned char ReadOneChar(void)
{
    unsigned char i=0;
    unsigned char dat = 0;
    for (i=8;i>0;i--)
    {
        DQ = 0;      // 给脉冲信号
        dat>>=1;
        DQ = 1;      // 给脉冲信号
        if(DQ)
            dat|=0x80;
        Delay_DS18B20(4);
    }
    return(dat);
}
/*****写一个字节*****/
void WriteOneChar(unsigned char dat)
{
    unsigned char i=0;
    for (i=8; i>0; i--)
    {
        DQ = 0;
        DQ = dat&0x01;
        Delay_DS18B20(5);
        DQ = 1;
        dat>>=1;
    }
}

```

```

}
/*****读取温度*****/
unsigned int ReadTemperature(void)
{
    unsigned char a=0;
    unsigned char b=0;
    unsigned int t=0;
    float tt=0;
    Init_DS18B20();
    WriteOneChar(0xCC); //跳过读序号列号的操作
    WriteOneChar(0x44); //启动温度转换
    Init_DS18B20();
    WriteOneChar(0xCC); //跳过读序号列号的操作
    WriteOneChar(0xBE); //读取温度寄存器
    a=ReadOneChar();    //读低 8 位
    b=ReadOneChar();    //读高 8 位
    t=b;
    t<<=8;
    t=t|a;
    tt=t*0.0625;
    t= tt*10+0.5;      //放大 10 倍输出并四舍五入
    return(t);
}
/*****END*****/

```

附录 B 电路图

