东 北 石 油 大 学

综 合 实 践

课 程 专业技术综合实践

题 目

学 院 电气信息工程学院

专业班级 自动化17-1班

小组成员 自己填

指导教师 自己填

2020年12月04日

目录

[第1章 概述 1](#_Toc57812324)

[第2章系统功能需求及方案设计 2](#_Toc57812325)

[2.1 设计需求分析 2](#_Toc57812326)

[2.2 系统设计方案及论证 2](#_Toc57812327)

[2.3 主要元器件 2](#_Toc57812328)

[第3章 硬件设计 4](#_Toc57812329)

[3.1 主控制器电路设计 4](#_Toc57812330)

[3.2 LCD1602显示模块设计 6](#_Toc57812331)

[3.3 NRF2401无线通信模块 8](#_Toc57812332)

[3.4 Proteus仿真电路图 9](#_Toc57812333)

[第4章 软件设计 10](#_Toc57812334)

[4.1 主程序设计 10](#_Toc57812335)

[4.2 NRF2401模块程序设计 10](#_Toc57812336)

[第5章 系统仿真及调试 12](#_Toc57812337)

[5.1系统的硬件仿真 12](#_Toc57812338)

[5.2系统调试中出现的问题 12](#_Toc57812339)

[5.3 对于问题的原因分析及改正后的效果 12](#_Toc57812340)

[第6章工程项目管理和经济决算 13](#_Toc57812341)

[6.1 工程项目管理概述 13](#_Toc57812342)

[6.2 工程项目管理内容 13](#_Toc57812343)

[结论 14](#_Toc57812344)

[参考文献 14](#_Toc57812345)

[附录 14](#_Toc57812346)

# 第1章 概述

石油化工产业是化学工业的重要组成部分，在国民经济的发展中有重要作用，是我国的支柱产业部门之一。石油化工指以石油和天然气为原料，生产[石油产品](https://baike.baidu.com/item/%E7%9F%B3%E6%B2%B9%E4%BA%A7%E5%93%81)和石油化工产品的加工工业。石油产品又称[油品](https://baike.baidu.com/item/%E6%B2%B9%E5%93%81)，主要包括各种燃料油（汽油、[煤油](https://baike.baidu.com/item/%E7%85%A4%E6%B2%B9)、[柴油](https://baike.baidu.com/item/%E6%9F%B4%E6%B2%B9)等）和润滑油以及[液化石油气](https://baike.baidu.com/item/%E6%B6%B2%E5%8C%96%E7%9F%B3%E6%B2%B9%E6%B0%94)、石油[焦碳](https://baike.baidu.com/item/%E7%84%A6%E7%A2%B3)、石蜡、沥青等。生产这些产品的加工过程常被称为石油炼制，简称炼油。石油化工产品以炼油过程提供的原料油进一步化学加工获得。石油化工罐区就是储存了大量经过炼制的石油产品，原料以及石油制品的区域，这些气体或是固体对实时的温度以及湿度都非常的敏感，故对于石油化工罐区域内的温湿度的实时监测非常重要。

本实践以单片机为基础，实现了石油化工罐区室内温湿度多点监测系统的设计，系统由单片机、多个数字温湿度传感器、NRF2401无线收发模块和液晶显示屏组成的-主-从双单片机数据检测,能实现罐区室内温湿度多点(至少3个点)的实时检测，采集的数据通过无线传输实时地显示在液晶屏上。

# 第2章系统功能需求及方案设计

## 2.1 设计需求分析

系统选用STC89C52单片机为主控芯片，使用NRF2401模块为无线通信模块，DHT11作为温湿度采集模块。且分为主从单片机，主单片机实时采集3个点位的温度数据，通过无线数据传输模块发送给从机。从机接受到数据后显示在LCD液晶屏上。实现数据的远程传输。

## 2.2 系统设计方案及论证

本系统由两片52单片机之间通过NRF24L01无线数据传输模块传输数据。主单片机实时获取DHT11模块获取的温湿度数据，发送给主发射模块。2.4G主机发送温湿度数据给从机，从机接收数据给从单片机，从单片机控制LCD模块显示接收到的温湿度数据。

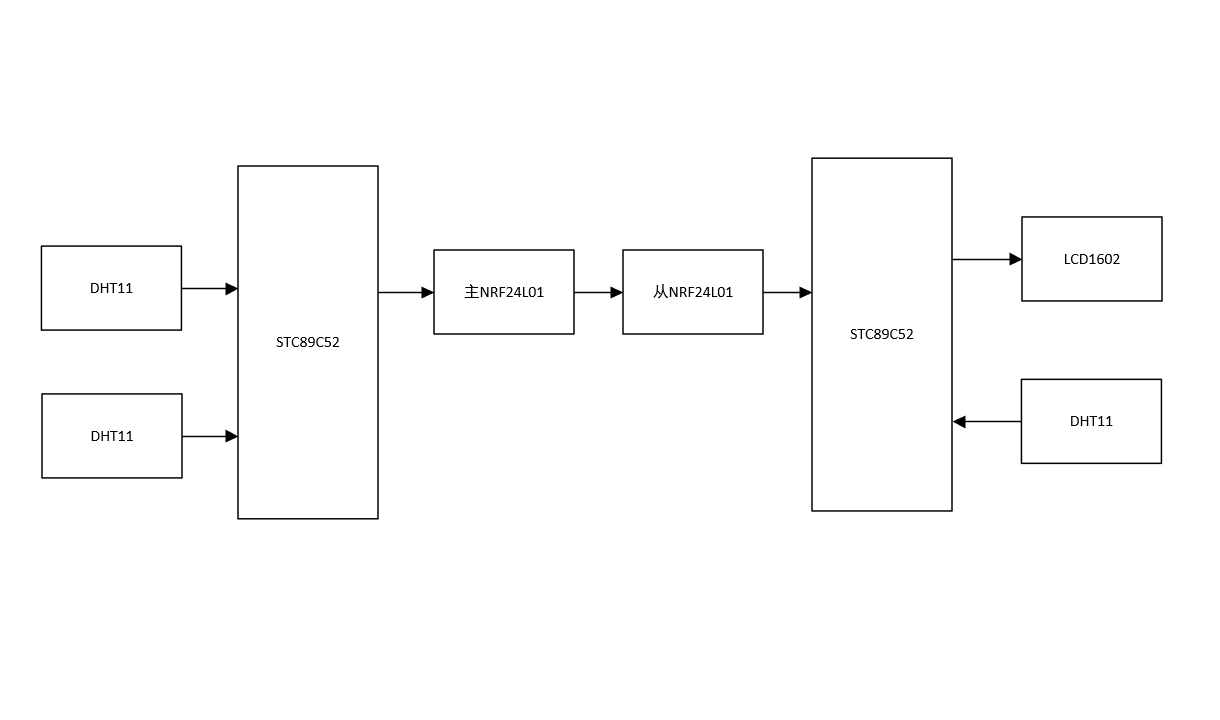


图2.2 程序框图

## 2.3 主要元器件

本系统设计到的元器件位DHT11，单片机，NRF2401模块以及LCD液晶显示模块。

单片机为STC89C52RC单片机，其与电脑通过串口通信。LCD1602为液晶显示模块，8位并行通信。NRF24L01的通信方式为SPI通信。SPI通讯使用3条总线及片选线，3条总线分别为SCK、MOSI、MISO，片选线为CS。CS:从设备选择信号线，常称为片选信号线，也称为NSS。当有多个SPI从设备与SPI主机相连时，设备的其它信号线SCK、MOSI及MISO同时并联到相同的SPI总线上，即无论有多少个从设备，都共同只使用这3条总线；而每个从设备都有独立的这一条CS信号线，本信号线独占主机的一个引脚，即有多少个从设备，就有多少条片选信号线。SPI协议中没有设备地址，它使用CS信号线来寻址，当主机要选择从设备时，把该从设备的CS信号线设置为低电平，该从设备即被选中，即片选有效，接着主机开始与被选中的从设备进行SPI通讯。所以SPI通讯以CS线置低电平为开始信号，以NSS线被拉高作为结束信号。SCK:时钟信号线，用于通讯数据同步。它由通讯主机产生，决定了通讯的速率，不同的设备支持的最高时钟频率不一样，两个设备之间通讯时，通讯速率受限于低速设备。

# 第3章 硬件设计

## 3.1 主控制器电路设计

主控模块是整个系统中最重要的一部分，就像人的大脑一样起着统筹的作用，这里使用STC89C52单片机作为主控制器。STC89C52是一种运行时耗功低、但却有着高性能的CMOS 8位微控制器，该芯片有40个引脚，其Flash有8k字节、RAM能储存512字、4个8位并行I/O口、3个16位定时器/计数器；[10]由于这种芯片自身具有灵巧的8位主控制单元处理器与能直接在系统上能进行功能写入的相关功能,这种芯片的控制功能也很强大，具有自身的掉电保护功能，可以有控制的停止工作，因此被广泛的应用在电子产品的各个方面。

这种型号的单片机内部带有富裕的存储空间，可以通过写入程序进行编程进行控制，它具有强大的自保与存储功能，再出现故障的情况下也不会出现数据丢失找不到的情况。这种芯片可以控制命令与动作中断的优先级。此单片机的工作频率与传统的单片机频率大概相似，范围几乎相同。详细的情况如以下介绍所述。

引脚的相关功能详述：

1.芯片的主电源引脚VCC(40)与VSS(20)的介绍：

VCC：供电电压引脚。

VSS：接地端子的引脚。

2.时钟电路引脚XTAL1(19)和XTAL2(18)

单片机的芯片里面配置有反向放大器装置，这些段子口中XTAL1是输入信号的；XTAL2是用作信号输出的端子，它也是芯片里面时钟工作电路信号输入的端子。

3.控制信号引脚

(1) RST口（9）：芯片上的引脚9的端口处，它表示的意义是复位输入。是单片机芯片的复位位置，当出现两个周连续的高电平周期的情况下，单片机会执行复位的动作。

(2)ALE/PROG端口(30)：当需要对数据存储器进行访问时，ALE的引脚端口的输出脉冲在通常情况下都可以使用固定的大小的脉冲进行输出。所以，它既有对外输出时钟信号的功能以及计时器的相关功能。

(3)EA/VPP(31)：它是一个信号输入的端口，能够对外部的存储器进行访问，还具有对外部的控制信号进行访问的功能。EA端为低电平(接地)，CPU则执行外部程序存储器的指令。

(4)并行I/O口P0~P3端引脚

P0口(32~39)：P0的端口功能主要由两个方面，一个方面是把它用作数据通信的接口，意思也就是把它当做基本I/0输入输出口进行应用。另一方面就是对芯片的功能进行扩展时可以进行反不同时段的应用。

单片机最小系统：对于每一个51系列的单片机，电源电路、时钟电路、复位电路都是构成整个单片机电路的基础，是系统组成的重要部分，对于一个完整的系统，每一个小电路都是不能缺少的一个小部分。因此，最小应用系统中也具有这三个重要的基本电路。本文所使用的单片机控制单元的最小系统电路如下图3-4的系统结构图所示：图中标注为10K的排阻是PO口的上拉电阻，由于P0口跟其他I/O口组成结构不相同，其自身结构比较特殊，所以，要对其加入上拉电阻之后才可以被正常应用。

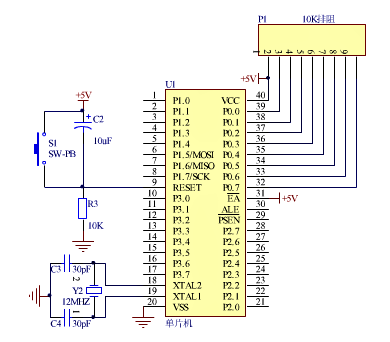


图3.4最小系统示意图

时钟电路：XTAL1 (19脚) ：为高增益反相放大器的输入端。XTAL2 (18脚) ：为高增益反相放大器的输出端。自激震荡电路就是在XTAL1与XTAL2之间接入一个晶振和一个微调电容。在多数情况下，晶振所处的大小范围一般在1.2MHz到12MHz大小的范围内，处于这个范围内的频率，数值都可以被随意选择。晶振的频率大小可以更高。但是频率与功率损耗在数值上是成正比的关系，因此，频率越高，功率的损失也就会更多[5]。

复位电路：复位的动作存在两种模式：一种是通过给电信号的方式，即上电自动复位的方式，另一个复位方式是手动的，通过按动按键的方式进行复位。最简洁的上电复位电路仅仅用一个电容与一个阻值大小的合适电阻进行串联进行组成，它是利用电容可充电的特性来实现的。它是利用电阻的分压来实现。本次设计选择按键复位电路，因为它在功能上与上电复位电路相同，但它操作起来与上电复位电路相比更简单。按键复位电路如图3.5所示

## 3.2 LCD1602显示模块设计

液晶屏是一种比较常见的基本组件，它是以液晶材料为基础的；液晶显示屏是一种高分子材料，现在普遍被应用在轻薄型显示器上。这种显示器的成像原理是通过电流对液晶分子的刺激，从而很快速的产生点、线、面然后与背部灯管共同组成画面。其中，1602字符型液晶屏是常见的一种，它是一种被广泛应用的液晶显示模块，它经常被用来显示字母、数字、符号。这种显示器是由几个点阵字符位组成，所以在进行代码编写的时候要注意对其二进制指令的应用，更让人值得关注的是，1602液晶可以显示十六进制数，所以在读取1302时钟芯片的记数时不用对其进制数进行转换[6]。

LCD1602屏幕可以显示两行字符，每行16个，总共32个字符。字符显示的位置和相关地址的关系如下表3.2.1所示：

表3.2.1 字符显示位与DDRAM的对应关系

|  |  |  |
| --- | --- | --- |
| 显示位置序号 | DDRAM地址 | |
| 第一行 | 第二行 |
| 1 | 00 | 40 |
| 2 | 01 | 41 |
| 3 | 02 | 42 |
| 4 | 03 | 43 |
| 5 | 04 | 44 |
| 6 | 05 | 45 |
| 7 | 06 | 46 |
| … | … | … |
| 39 | 26 | 66 |
| 40 | 27 | 67 |

型号为1602的显示器采用标准16引脚的接口，其中各个引脚的功能与位置说明如下表3.2.2所示：

表3.2.2 各引脚接口说明

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | 符号 | 引脚的意义说明 | 编号 | 符号 | 引脚说明 |
| 1 | VSS | 电源的接地端子 | 9 | D2 | Data I/O |
| 2 | VDD | 电源正极 | 10 | D3 | Data I/O |
| 3 | VL | 液晶显示偏压 | 11 | D4 | Data I/O |
| 4 | RS | 数据/命令选择 | 12 | D5 | Data I/O |
| 5 | R/W | 读/写选择 | 13 | D6 | Data I/O |
| 6 | E | 使能信号 | 14 | D7 | Data I/O |
| 7 | D0 | Data I/O | 15 | BLA | 背光源正极 |
| 8 | D1 | Data I/O | 16 | BLK | 背光源负极 |

第1脚：GND表示的是接地

第2脚：与5V的正极性进行连接

第3脚：VL是显示效果的对比度调整引脚。

第4脚：寄存器位置功能选择的端子。

第5脚：通过电位的高低活大小执行读或者写的命令。

第6脚：使能端的端口，当电位的大小发生变化时，液晶模块根据电位指令的变化执行接收到的命令指令。

第7至14脚：这几个引脚都具有双向通信的功能。

第15脚：背景电源的正极性。

第16脚：背景电源的负极性。

1.并行连接。LCD1602的8条数据线和3条控制线（E，RS，R/W）与单片机相连后就可以正常工作。对于一般应用来说，只需要往LCD1602中写入命令和数据即可，因此，可将LCD1602的R/W引脚直接接地，这样便可以节省1条数据线。VL引脚是液晶对比度调试端，想要实现对比度的调整通常情况下连接一个10kΩ的电位器即可；也可以选择将一个适当大小的电阻通过该引脚接地的方式进行调整，不过电阻的大小需要通过调试后才能决定。

2.串行连接。这个连接也称为四线控制工作方式，是利用HD44780的4位数据总线的功能，将电路接口进行简化的一种方式。为了减少接线数量，所以在与单片机进行通信时只采用引脚DB4~DB7，先传数据或命令的高4位，再传低4位。[13]采用的是四线并口通信方式，这样可以减少对微控制器I/O口的需求，如果在产品的设计过程中出现单片机的I/O口资源紧张时，可以考虑使用此方法。

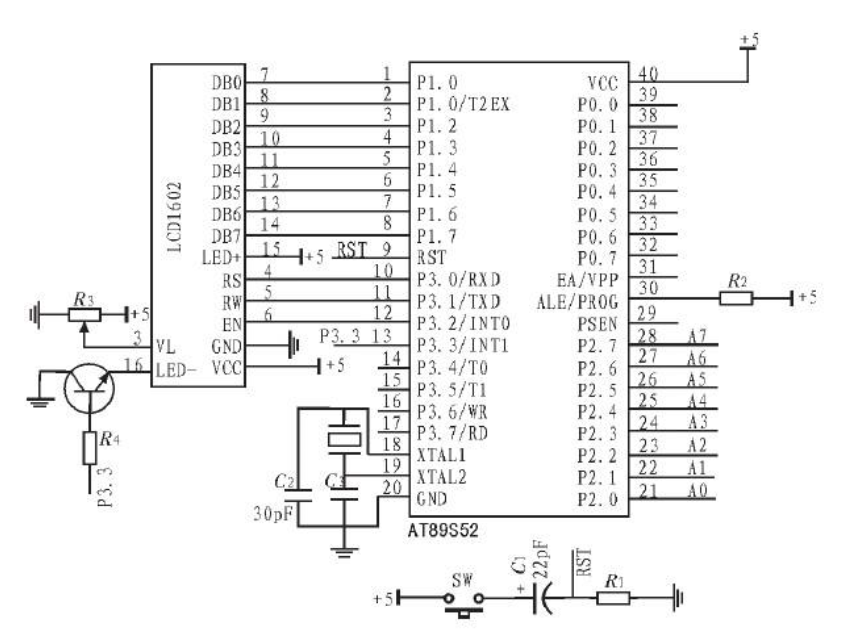


图3.2 单片机与LCD连接方式

## 3.3 NRF2401无线通信模块

nRF2401是单片射频收发芯片，工作于2.4～2.5GHz ISM频段，芯片内置频率合成器、功率放大器、晶体振荡器和调制器等功能模块，输出功率和通信频道可通过程序进行配置。芯片能耗非常低，以-5dBm的功率发射时，工作电流只有10.5mA，接收时工作电流只有18mA，多种低功率工作模式，节能设计更方便。其DuoCeiverTM技术使nRF2401可以使用同一天线，同时接收两个不同频道的数据。nRF2401适用于多种无线通信的场合，如无线数据传输系统、无线鼠标、遥控开锁、遥控玩具等。

nRF2401（最新版本为nRF2401A，nRF2401AG为无铅工艺版本）是由Nordic公司出品的单芯片无线收发芯片，工作于2.4GHz～2.5GHz的全球免申请（ISM）频率。芯片包括一个完全集成的频率合成器，功率放大器，晶体振荡器和调制器。发射功率和工作频率等工作参数可以很容易的通过3线SPI端口完成。极低的电流消耗，在-5dBm的输出功率时仅为10.5mA，在接收模式时仅为18mA。掉电模式可以很容易的实现低功耗需求。

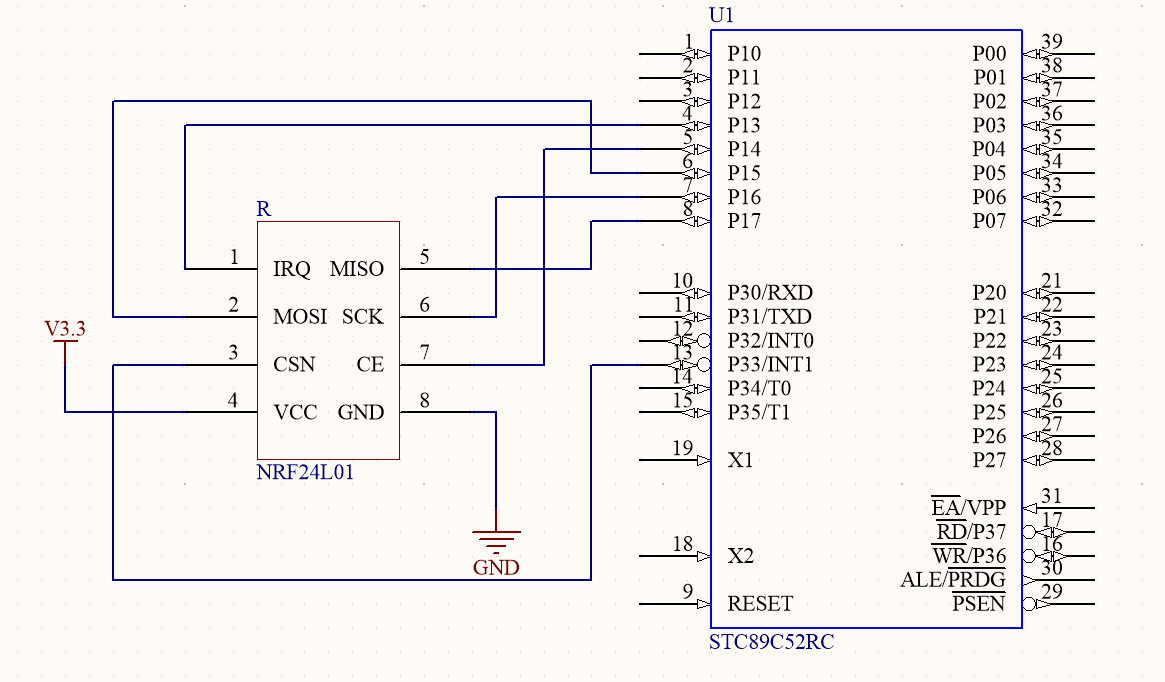


图3.3 模块与单片机连接方式

## 3.4 Proteus仿真电路图

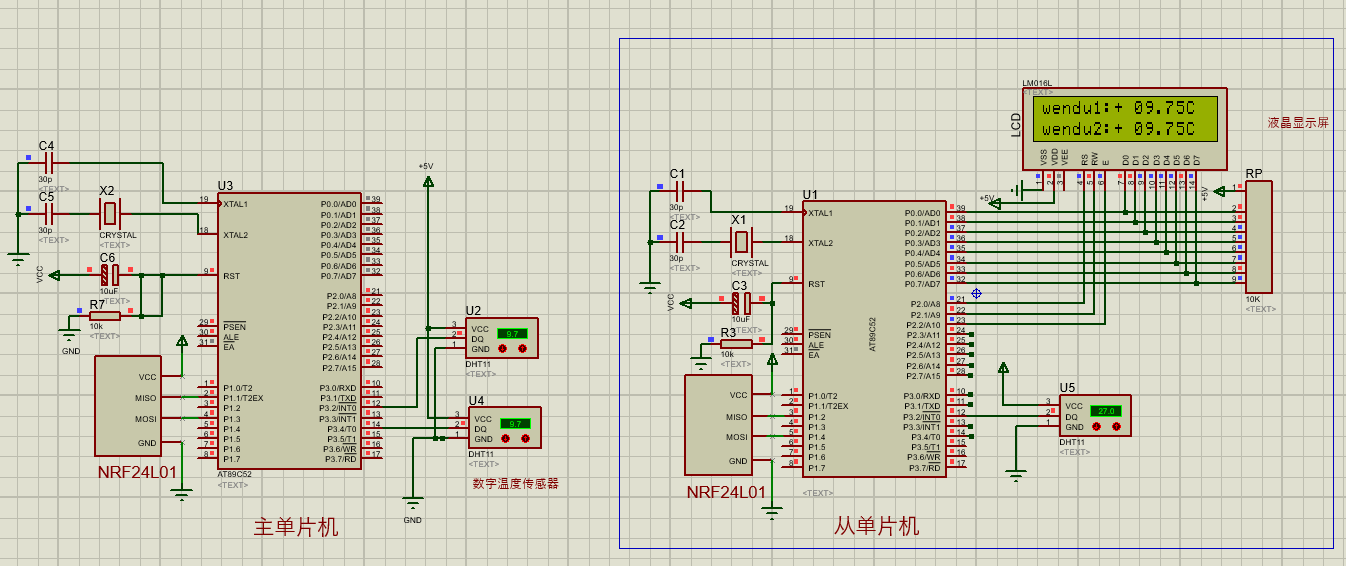
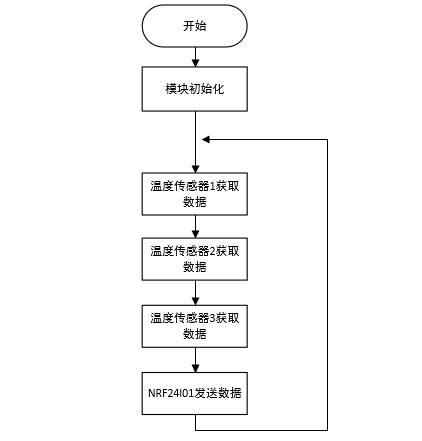
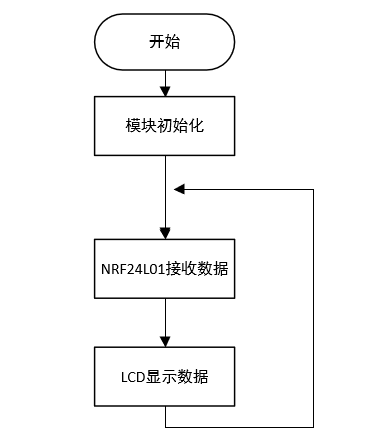


图5.3 仿真图

# 第4章 软件设计

## 4.1 主程序设计

本系统分为主从单片机，两者通过2.4G无线通信模块相互交互数据。主单片机获取3路的温湿度模块的数据，并通过无线通信模块发送给从单片机。从单片机接受到数据后，进行一些处理后，将数据显示在LCD1602液晶显示屏上。实现一个远程的多点温湿度监控。

主单片机 从单片机

图4.1 主程序流程图

## 4.2 NRF2401模块程序设计

2.4G无线通信模块在本系统起数据通信的作用。主单片机通过此模块的发送数据给从单片机上的相应模块。两个单片机上的无线通信模块也有一个是主机，一个是从机，主从机可以相互交互数据。

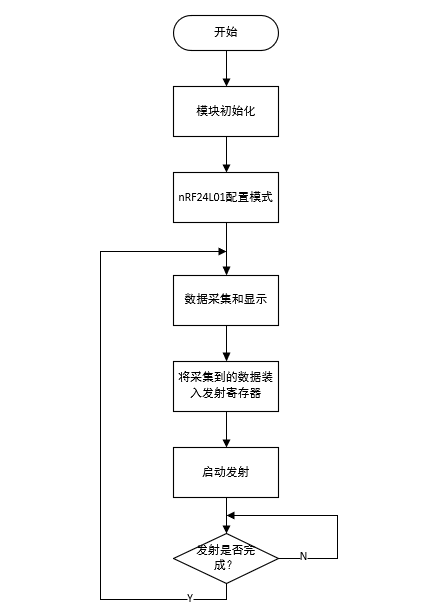


图4.2 NRF2401程序流程图

# 第5章 系统仿真及调试

## 5.1系统的硬件仿真

系统的硬件主要有52单片机，多个温度检测传感器，LCD驱动电路，两点2.4G无线通信模块电路等等。

每个温湿度传感器只有3个引脚，两个用于供电，一个用于与单片机的数据传输。NRF24L01模块有8个引脚，采用SPI通信协议通信。本系统有两个单片机，分别在两个万用板上焊接完成，供电通过一个系统供电。

## 5.2系统调试中出现的问题

在实物仿真调试完成后，进行程序的调试，将程序下载入单片机，查看运行结果。在实物调试时，遇到了一下问题，发现后都被一一解决。

问题一：无法获取3个点的温度数据。

问题二：2.4G模块数据传输有干扰数据。

## 5.3 对于问题的原因分析及改正后的效果

对于问题一：每个模块的的数据口虽不同，但是相互之间有冲突。我们查阅了相关资料，最终解决了问题，可以几乎同时采集3路温度数据，通过串口显示在单片机的串口助手上。

对于问题二：开始虽能成功实现2.4G主从机模块的连接，但有时会有无关数据传输，使得显示出现问题。我们参考网上的程序修改，同时加上了限定值，使得显示正常。

# 第6章 工程项目管理和经济决算

潮湿、发霉、腐蚀是石油化工罐日常工作中需要解决的问题，故仓库中这些变量信是否达标则成为了衡量仓库管理贡量重要特征。这些问题都将影响到储备物资的安全以及工作的可靠性。因此，现在绝大部分的仓库都配备了库内温度和湿度的监测设施。与传统的人工检测，通过采用湿度计、毛发湿度计和湿纸等器件去检测环境中温湿度信息相比，现在的温湿度传感器及其外部设施，具有无比的优越性，给现代化的仓库管理系统带来

## 6.1 工程项目管理概述

发送端:由数字温湿度传感器AM2301，实时地采集当前的温湿度信息，经单片机分析处理后，通过模拟SPI接口控制无线射频模块nRF24L01装载温湿度信息，由无线射频模块nRF24L01发射给主机接收端并显示温湿度信息。接收端:由nRF24L01无线射频模块接收终端采集过来的数据信息，经模拟SPI接口发给单片机，由单片机控制液晶LCD实时显示温湿度信息，并且显示报警温湿度，上限值。

## 6.2 工程项目管理内容

1.投资

本系统单片机部分采用AT89C51，一般价格在5-10元左右，显示部分采用有LCD1602，一般价格在30-50元不等，NR2401一个一般在50-60一个，DHT11一个一般在10-15元一个。如表6-1所示。

表6-1研发成本

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **组成部分** | **AT89C51** | **LCD1602** | **NRF2401** | **电子元器件** | **编程**  **技术** | **测试**  **技术** | **总计** |
| **成本/元** | 5-10 | 30-50 | 50-60 | 10-20 | 0 | 0 | 300 |

2.决算

因为每个地方生活水平消费能力不同，场地费用和维护费用等会有所不同，所以在不同的地方需要要考虑当地成本。

# 结论

通过这一学期的努力，我的综合实践在我的导师的耐心教导下终于顺利完成。在整个设计过程中，从一开始的选题一直到最后完成，每一步都耗费了老师和我大量的汗水。在老师的耐心的指导帮助下，我查阅了大量的课题相关资料，并对单片机、温度传感器、湿度传感器、proteus、KeilC、等理论知识重新进行了系统的梳理与学习，在设计中所接触的知识让我受益匪浅。

# 参考文献

[1] 蔡美琴等.MCS-51系列单片机系统及其应用[M]北京:高等教育出版社， 2004.[2] 谭浩强.C程序设计(第四版) [M]北京:清华大学出版社，2003.  
[3] 肖洪兵、跟我学用单片机[M].北京:北京航空航天大学出版社，2002.  
[4] 胡汉才.单片机原理及其接口技术(第二版)，清华大学出版社,2004.  
[5] 沈红卫.基于单片机的智能系统设计与实现，电子工业出版社,2005.  
[6]张娟，万隆,曾实现. Protel电路设计教程.清华大学出版社,2010.  
[7] 付少波.传感器及其应用电路.化学工业出版社，2011.  
[8] 陈善久.单片机微型计算机原理及应用[M].西安电子科技大学出版社，1997.  
[9] 纪宗南、单片机外围器件实用手册输入通道器件分册[M]. 北京航空航天大学出版社，1998.

[10] 涂时亮，张友德，单片微机控制技术[M]. 复旦大学出版社，1994.

# 附录

//DHT11驱动

void DHT11\_start()

{

Data=1;

DHT11\_delay\_us(2);

Data=0;

DHT11\_delay\_ms(30); //延时18ms以上

Data=1;

DHT11\_delay\_us(30);

}

uchar DHT11\_rec\_byte() //接收一个字节

{

uchar i,dat=0;

for(i=0;i<8;i++) //从高到低依次接收8位数

{

while(!Data); ////等待50us低电平过去

DHT11\_delay\_us(8); //延时60us，如果还为高则数据为1，否则为0

dat<<=1; //移位使正确接收8位数据，数据为0时直接移位

if(Data==1) //数据为1时，使dat加1来接收数据1

dat+=1;

while(Data); //等待数据线拉低

}

return dat;

}

void DHT11\_receive(uchar \*temp,uchar \*humi) //接收40位的数据

{

uchar R\_H,R\_L,T\_H,T\_L,RH,RL,TH,TL,revise;

DHT11\_start();

if(Data==0)

{

while(Data==0); //等待拉高

DHT11\_delay\_us(40); //拉高后延时80us

R\_H=DHT11\_rec\_byte(); //接收湿度高八位

R\_L=DHT11\_rec\_byte(); //接收湿度低八位

T\_H=DHT11\_rec\_byte(); //接收温度高八位

T\_L=DHT11\_rec\_byte(); //接收温度低八位

revise=DHT11\_rec\_byte(); //接收校正位

DHT11\_delay\_us(25); //结束

if((R\_H+R\_L+T\_H+T\_L)==revise) //校正

{

RH=R\_H;

RL=R\_L;

TH=T\_H;

TL=T\_L;

\*temp=TH;

\*humi=RH;

}

}

//接收主程序

void main()

{

uchar RxBuf[20]={0};

// init();

init\_io() ;

UsartInit();

LcdInit();

LcdWriteCom(0x80);

LcdWriteData('H');

LcdWriteData('1');

LcdWriteData(':');

LcdWriteCom(0x80+9);

LcdWriteData('H');

LcdWriteData('2');

LcdWriteData(':');

LcdWriteCom(0x80+0x40+4);

LcdWriteData('H');

LcdWriteData('3');

LcdWriteData(':');

while(1)

{

DHT11\_receive(&tem,&hmi);

LcdWriteCom(0x80+3);

LcdWriteData(tem/10+'0');

LcdWriteData(tem%10+'0');

LcdWriteData(' ');

LcdWriteData(hmi/10+'0');

LcdWriteData(hmi%10+'0');

SetRX\_Mode();

PutString("no receive!\r\n");

if(nRF24L01\_RxPacket(RxBuf))

{

LcdWriteCom(0x80+12);

LcdWriteData(RxBuf[1]+'0');

LcdWriteData(RxBuf[2]+'0');

LcdWriteData(' ');

LcdWriteCom(0x80+0x40);

LcdWriteData(RxBuf[3]+'0');

LcdWriteData(RxBuf[4]+'0');

LcdWriteCom(0x80+0x40+7);

LcdWriteData(RxBuf[5]+'0');

LcdWriteData(RxBuf[6]+'0');

LcdWriteData(' ');

LcdWriteData(RxBuf[7]+'0');

LcdWriteData(RxBuf[8]+'0');

PutString("I receive\r\n");

PutString(RxBuf);

PutString("\r\n");

}

}

}

//发送主程序

void main()

{

int a=0;

uchar TxBuf[20]={0}; //

init\_io() ;

UsartInit();

nRF24L01\_TxPacket(TxBuf); // Transmit Tx buffer data

Delay1(6000);

while(1)

{

DHT11\_receive(&tem,&hmi);

DHT11\_receive\_1(&tem\_1,&hmi\_1);

PutString("OK\r\n");

TxBuf[0]='#';

TxBuf[1]=tem/10+1-1;

TxBuf[2]=tem%10+1-1;

TxBuf[3]=hmi/10+1-1;

TxBuf[4]=hmi%10+1-1;

TxBuf[5]=tem\_1/10+1-1;

TxBuf[6]=tem\_1%10+1-1;

TxBuf[7]=hmi\_1/10+1-1;

TxBuf[8]=hmi\_1%10+1-1;

nRF24L01\_TxPacket(TxBuf); // Transmit Tx buffer data

PutString(TxBuf);

PutString("\r\n");

PutString("I send\r\n");

DHT11\_delay\_ms(1000);

}

}