基于超声波的空气污染物浓度测量系统研制

摘要

阅读并翻译相关文献，从理论上对使用超声波传感器进行空气污染物浓度测定的可行性进行分析。基于超声传播在不同浓度的混合气体中具有微小时差的特性，设计了空气污染物浓度检测系统。该系统选用瑞士盛世瑞恩(SENSIRION)公司的SHT10数字温湿度传感器测量空气的温度和湿度，使用超声传感器测量超声波在一段定长空气中传播的时间。同时该系统选用恩智浦(NXP)半导体公司的32位基于ARM Cortex-M0内核的LPC1114微控制器对传感器获得的温度，湿度和时间数据进行采集，并将这些数据以特定格式封装成帧并使用蓝牙芯片通过无线通信方式将数据帧发出。该系统使用自己编写的基于安卓的手机程序，通过调用手机自带的蓝牙模块对数据进行接收。手机程序对接收的数据帧进行解帧，并对解帧后的数据进行处理，将结果显示在手机界面上。整个项目使用Git以及Subversion作为版本控制工具。

关键词：空气污染物浓度，超声波，微控制器，无线通信，安卓，版本控制

The Development of Air Pollutants’ Concentration- Measuring System Based on Ultrasonic

ABSTRACT

Reading and translating relevant documents and materials, analyze the feasibility of applying ultrasonic sensor to measure the concentration of air pollutants in theory. Design the air pollutants’ concentration-detecting system based on the characteristic that there will be minor time difference when ultrasonic spreads in mixed gas of different concentration. This system makes use of SHT10 digital temperature and humidity sensor manufactured by SENSIRION Company in Switzerland to measure temperature as well as humidity of air and measures the time of ultrasonic spreading in the air of selected length by using ultrasonic sensor. At the same time, this research chooses NXP Company’s LPC1114 32-bit ARM Cortex-M0 microcontroller to collect temperature, humidity and time data gained from ultrasonic sensor, later packages these data into frames with certain format and eventually sends out those data frames through wireless communication by using a Bluetooth chip. The system takes advantage of self-written mobile-phone application based on Android OS to receive data by calling Bluetooth module that exists initially in phones. Mobile-phone applications then de-frame and process the received data frames, and display results in the interface of mobile-phones. The whole project uses Git and Subversion as the version controlling tools.

Key words: air pollutants’ concentration, ultrasonic, microcontroller, wireless communication, Android OS, version control

1 引 言

* 1. 研制空气污染物浓度检测系统的意义

随着社会经济的发展和人们生活水平的提高，城市环境空气质量也越来越成为人们普遍关注的热点话题，因而对城市大气环境质量做出客观、全面、实时的认识和评价是极其必要的。

空气污染指数(air pollution index, API)是反映大气环境质量水平的重要标准。根据环境空气质量标准和各项污染物对人体健康和生态环境的影响，将常规检测的集中空气污染物浓度简化为单一的概念性指数值形式，它将空气污染程度和空气质量状况分级表征，适合于表示城市的短期空气质量状况和变化。

* 1. 空气污染物浓度检测的常用方法

1.2.1 利用电阻式气敏元件测量气体浓度

电阻式气敏元件经过一段时间的预热后，其阻值与周围气体浓度有着确定的关系。因此，只需测出运行状态下气敏元件的电阻值，便可推算出当前运行状态下待测气体的浓度。这是气体浓度测量中最常用的传统方法。

利用气敏元件测量气体浓度原理简单，操作也比较方便，但存在以下不足：(a)由于在检测仪中一般将气敏元件与标准原件组成测量电桥电路，因此电桥电路的非线性会影响测量精度；(b)电桥供电电压的大小对测量精度有很大影响；(c)检测时还需考虑现场温度、空气扰动等因素，需要传接上补偿电路。所以这种方法主要用在测量精度不高的场合。

1.2.2 气相色谱法测量气体浓度

气相色谱法是基于不同的气体在通过色谱柱时速度不同的原理，主要适用于工厂生产现场的检测、设备检修过程中动力作业的安全检测。该方法对混合气体进行多次采样，注入色谱仪直至色谱峰值全部出现后再对其进行色谱分析。很多地方也称其为光谱测量法。不同浓度的混合气体的色谱存在明显差异，根据当前呈现的色谱便能分析出当前气体浓度的高低。这种检测方法测量方便、快捷、线性范围宽、灵敏度高、可靠性重现性号，在气体浓度检测中占有很重要的地位，但在测量中必须合理地选择色谱柱及载气流速等参数，只有进行多次重复实验后方能获得理想的测量效果。

1.2.3 载体催化燃烧法检测气体浓度

这种检测方法线性和稳定性较好。以爆炸下限百分体积浓度为单位的浓度标度方法能统一衡量各种可燃气体浓度所呈现出的爆炸危险度，而且量程符合工业要求，故常被用于爆炸危险场所可燃性气体的测爆。该检测方法以催化载体型气敏元件作为浓度的传感器，该元件由在铂丝上烧结一层陶瓷载体后，再涂覆催化活性物（Rh,Pd等）构成。当铂丝中通以工作电流使之达到临界反应温度（320~350℃）时，可燃气就在元件表面催化燃烧，铂丝电阻增加，在完全燃烧且热辐射可忽略时，电阻增量ΔR与可燃气体浓度C成正比，将ΔR转换成电信号，即可用于检测可燃气体的浓度。检测中一般将催化元件（检知元件）及与之配对的参比元件组成的电桥，作为浓度信号采样单元。由于催化元件的气敏特性除与可燃气体浓度有关外，还受到工作电流、环境的温度和湿度、气压的影响，所以在设计过程中应注意采用桥式单元或其他参比元件对其予以补偿。

1.2.4 光干涉法测量气体浓度

光干涉法是一种创新的检测方法，它基于同一光源发出的光波会被平面镜分为2束光，经不同路线后又汇集成一束光，发生光干涉，产生干涉条纹的物理现象。光干涉式气体浓度测量器目前被广泛用于煤矿井下空气中的甲烷、瓦斯等浓度，以及其他环境下各种气体浓度的测定；在实验室也往往使用这种测定仪器作为相对参考的仪器。用该方法测定气体浓度时，必须考虑到气体的温度、压力、含量、换算结果，以及光在该气体中的折射率等问题[1]。

* 1. 超声技术测量气体浓度的基本概况

使用超生原理测量气体浓度是近10年来由于电子技术和测量技术的发展而出现的一种新技术。气体浓度的超声检测法为非接触式测量，具有测量范围宽、测量精度高等特点，而且测试设备体积小，无需维护，使用寿命长，在计量检测领域收到极大关注。近年来，超声技术作为一种新型的检测技术已广泛应用于气体浓度检测领域。国内外一些公司经过大量研究，已将超声技术成功地应用于气体浓度的高精度检测装置中，实现了气体浓度检测技术的重大突破[2]。

* 1. 本文所作的工作

本文首先从理论上分析利用超声波测量气体浓度的原理，通过数学推导验证超声法对气体污染物浓度进行监测的可行性。并从以下几个方面介绍毕业设计空气污染物浓度监测系统所做的工作:

(1) 对于LPC1114微控制器底层寄存器的配置，包括32位计时器的匹配输出与捕获输入，利用IO口模拟应用于温湿度传感器SHT10的两线串行接口(DATA总线与CLK总线)；

(2) 硬件电路的设计，包括原理图设计和印刷电路板(PCB)图设计；

(3) 基于安卓的手机上位机程序设计，包括使用设计模式中的依赖倒置原则；

(4) 版本控制，包括程序工程进行版本控制的工具(Git)和对电路工程进行版本控制的工具(Subversion)；

(5) 实验结果。

1. 理论部分

2.1 浓度声速理论

将空气简化为二元混合气体（普通空气与微量污染物气体），在常温常压实验条件下可利用理想气体模型对该气体进行分析。混合气体平均速度为[2]：



式中为混合气体的平均速度；为混合气体平均定压定容比热比；*P*是气体压力；是气体密度。

使用理想气体方程：



式中为气体体积；为通用气体常数，也称普适气体恒量，；T为气体的热力学温度；为气体的质量；为气体的平均相对分子质量。

将式(2.2)代入式(2.1)中可得：



对于二元混合气体，设其两种组分分别由下标组成，有：





式中为类气体的浓度，为类气体的定容热比，分别为两种气体的相对分子质量。

根据式(2.3)，令参数*Y*为：



由以上等式整理可得：



其中，系数均为参数*Y*的函数。；； 。

由于，所以方程(2.5)有单根，其解为：



通过测量出的声速和温度两个量，可得出参数的值，并由此计算出方程的3个系数，代入式(2.6)，求出浓度n值[3]。

2.2 嵌入式系统相关理论

本毕业设计使用的微控制器型号为恩智浦半导体公司基于ARM 架构Cortex-M0内核的32位低成本微控制器LPC1114。该款微控制器的外设组件包括：64KB片内FLASH程序存储器，8KB片内SRAM，一路I2C总线接口，一路RS-485/EIA-485 UART，两路SPI总线接口，四个通用定时/计数器以及多达42个通用I/O口[4]。

LPC1114 Cortex-M0微控制器具有以下特性：

1. Cortex-M0内核，工作频率高达50MHz；
2. 内置嵌套向量中断控制器（NVIC）；
3. 高达32KB片内Flash程序存储器；
4. 高达8KB片内SRAM存储器；
5. 在系统编程（ISP）和在应用编程（IAP）可通过片内引导装载程序软件实现；
6. 串行接口包括：

--可产生小数波特率、具有调制解调器、内部FIFO和支持RS-485/EIA-485标准的UART；

--SSP控制器，带FIFO和多协议功能（仅在LQFP48和PLCC44封装中有两路SSP）；

--I2C总线接口，完全支持I2C总线规范和快速模式，数据速率为1Mbit/s，具有多个地址识别功能和监控模式。

1. 其它外设：

--多达42个通用I/O（GPIO）引脚，带可配置的上拉/下拉电阻；

--P0.7引脚支持20mA的高驱动电流；

--I2C总线引脚在FM+模式下可支持20mA的灌电流；

--4个通用定时器/计数器，共有4路捕获输入和13路匹配输出；

--可编程的看门狗定时器（WDT）；

--系统节拍定时器。

1. 带有SWD调试功能；
2. 集成了PMU（电源管理单元），可在睡眠、深度睡眠和深度掉电模式中极大限度地减少功耗；
3. 具有三种低功耗模式：睡眠模式、深度睡眠模式和深度掉电模式；
4. 3.3V单电源供电（2.0V~3.6V）；
5. 10位ADC，在8个引脚中实现输入多路复用；
6. GPIO均可以配置为边沿或电平中断；
7. 带驱动的时钟输出功能可以反映主振荡器时钟、IRC时钟、CPU时钟和看门狗时钟；
8. 13个起始逻辑功能可以将CPU从深度睡眠模式中唤醒；
9. 掉电检测，具有4个独立的阀值，用于中断和强制的复位；
10. 上电复位（POR）；
11. 主振荡器工作范围：1Mhz~25MHz；
12. 12MHz内部RC振荡器可调节到1%的精度，可将其选择为系统时钟；
13. PLL允许CPU在最大的CPU速率下操作，而无需高频晶振，可从主振荡器、内部RC振荡器或从看门狗振荡器中运行；
14. 可采用LQFP48、PLCC44或HVQFN33封装。

2.3 Android操作系统及JAVA编程相关知识

2.3.1 Android操作系统

Android，中文俗称安卓，是一个以Linux为基础的开放源代码移动设备操作系统，主要用于智能手机和平板电脑，由Google成立的Open Handset Alliance（OHA，开放手持设备联盟）持续领导和开发中。Android已发布的最新版本为Android 5.1.1（Lollipop）。

Android系统最初由安迪·鲁宾（Andy Rubin）等人开发制作，最初开发这个系统的目的是创建一个数码相机的先进操作系统；但是后来发现市场需求不够大，加上智能手机市场快速成长，于是Android被改造为一款面向智能手机的操作系统。于2005年8月被美国科技企业Google收购。2007年11月，Google与84家硬件制造商、软件开发商及电信营运商成立开放手持设备联盟来共同研发改良Android系统，随后，Google以Apache免费开放源代码许可证的授权方式，发布了Android的源代码，让生产商推出Android的智能手机，Android操作系统后来更逐渐拓展到平板电脑及其他领域上。

2010年末数据显示，仅正式推出两年的Android操作系统在市场占有率上已经超越称霸十年的诺基亚Symbian系统，成为全球第一大智能手机操作系统[5]。

2.3.2 JAVA程序设计理论

Java是一种计算机编程语言，拥有跨平台、面向对象、泛型编程的特性，广泛应用于企业级Web应用开发和移动应用开发。

Java编程语言的风格十分接近C++语言。继承了C++语言面向对象技术的核心，Java舍弃了C++语言中容易引起错误的指针，改以引用取代，同时移除原C++与原来运算符重载，也移除多重继承特性，改用接口取代，增加垃圾回收器功能。在Java SE 1.5版本中引入了泛型编程、类型安全的枚举、不定长参数和自动装/拆箱特性。太阳微系统对Java语言的解释是：“Java编程语言是个简单、面向对象、分布式、解释性、健壮、安全与系统无关、可移植、高性能、多线程和动态的语言”。由于JAVA是面向对象编程的语言，因此在编程中需要使用设计模式的理论作为指导进行编程。

2.3.3 设计模式

Christopher Alexander说过：“每一个模式描述了一个在我们周围不断重复发生的问题，以及该问题的解决方案的核心。这样，你就能一次又一次地使用该方案而不必做重复劳动。”尽管Alexander所指的是城市和建筑模式，但他的思想也同样适用于面向对象设计模式，只是在面向对象的解决方案里，我们用对象和接口代替了墙壁和门窗。两类模式的核心都在于提供了相关问题的结局方案[6]。

2.4 版本控制相关知识

本毕业设计仅对保存源代码的文本文件和PCB设计文件作版本控制管理，而实际上，你可以对任何类型的文件进行版本控制。

如果你是一位图形或网页设计师，可能会需要保存某一副图片或页面布局文件的所有修订版本[7]。采用版本控制系统（VCS）是个明智的选择。有了它你就可以将某个文件回溯到之前的状态，甚至将整个项目都回退到过去某个时间点的状态。你可以比较文件的变化细节，查出是谁最后修改了什么地方从而造成某些怪异问题，又是谁在何时报告了某个功能缺陷，等等。使用版本控制系统通常还意味着，就算你胡来搞砸了整个项目，把文件改的改，删的删，你也可以轻松恢复到原先的样子。而由此额外增加的工作量确微乎其微。版本控制系统分本地版本控制系统，集中化的版本控制系统和分布式的版本控制系统。

在本次毕业设计中，我使用了Git作为程序源代码工程的版本控制系统，又使用了Subversion作为基于Altium Designer软件设计的PCB电路工程的版本控制系统。

1. 需求分析与系统架构
   1. 需求分析

（1） 由于需要测量空气中的温度，湿度和超声波在声音中传播的速度，故需要设计硬件电路板，可以实现上述采集数据的功能；

（2） 需要设计基于LPC1114微控制器的程序，能控制数据采集的时序，并将采集回来的程序进行装帧并通过蓝牙串口发出；

（3） 在接收端需要设计基于Android OS的手机上位机，将单片机从蓝牙串口发出的数据进行接收并进行处理；

（4） 整体设计应该足够小，另外应该具有尽量低的功耗，这样可以满足采用电池供电的需求。

3.2 系统架构

3.2.1 硬件设计框架

对于温度和湿度的测量，该系统采用瑞士的盛世瑞恩公司的SHT10数字式温湿度传感器，该款传感器在环境温度25℃和供电电压3.3V的条件下测湿精度在±4.5%RH，测温精度在±0.5℃，基本可以满足系统对于温度和湿度测量的要求。SHT10传感器的接口为两线制串行接口，接口说明见下图。



图3.1 SHT10温湿度传感器典型电路

SHT10的串行接口，在传感器信号的读取及电源损耗方面，都做了优化处理；但与通用I2C接口不兼容[8]，因此，需要使用两个通用I/O口模拟串行时钟输入接口（SCK）和串行数据接口（DATA）。

对于超声波速度的测量，则需要使用单片机的计时功能。LPC1114提供了两种计时方式，一种是使用系统节拍定时器，最快可以每10毫秒产生一个中断，通过中断进行计时。另一种是使用微控制器内部的2个32位和2个16位的可编程定时器/计数器，最快可以以系统时钟频率（8MHz）的速度进行计时。

因为对于超声波在一小段距离内的计时要求非常精确，且激发超声波的信号频率在180kHz，故在这里采用硬件计时方式，即使用定时器/计数器进行计时。

超声接收部分的输出信号为180kHz的正弦波包络，为了计时，将正弦包络

由于最后单片机需要将数据发送到基于Android OS的手机终端，故还需使用微控制器的UART模块，通过UART将数据发送到蓝牙芯片上，再从蓝牙芯片发送到手机。

由以上条件可知，硬件电路的设计应基于LPC1114单片机，围绕单片机的GPIO，TIMER和UART通讯建立。

图3.2 硬件设计框架

3.2.2 软件设计架构

软件可以看做两部分：前端测量模块，由LPC1114单片机和传感器电路组成，主要完成温湿度与超声波传播时间信号的采集和数据的无线传输；终端接收模块，由运行在Android OS手机上的上位机组成，主要负责完成蓝牙数据的接收，解帧和处理。

图3.3 软件设计框架

考虑到程序的可扩展性，在此使用分层的思想对数据帧进行设计。底层帧为帧头，帧长度，数据域和校验位；而数据域内又分命令字位和数据位。底层帧结构如下图所示：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 帧头1(0xAA) | 帧头2(0xBB) | 帧长度 | 数据域 | 校验字 |

图3.4 底层数据帧格式

数据域分命令字和数据位，命令字由一个字节构成，考虑到在监测过程中需要测量环境温度，环境湿度和超声波声时三个数据，故令该字节从低到高的三位为有效位，分别表示是否使能测量温度，湿度和声时。相应位置1说明使能了对应数据的测量。数据位则依次填入温度，湿度和声时的数据，每个数据值占两个字节，若未进行测量，则该数据省略，具体格式如下图：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 命令字(1 Byte) | 温度1 | 温度2 | 湿度1 | 湿度2 | 声时1 | 声时2 |

图3.5 数据域（上层数据帧）格式

* 1. 各模块设计思路

3.3.1 温湿度传感器模块设计思路

温湿度传感器SHT10的硬件设计比较简单，可以直接使用用户手册里的参考电路进行应用。同时，可在上拉电阻边串联一个led，通过led的亮灭状态判断数据总线上是否有数据通过。

由于SHT10为一款数字式传感器，采用串行接口进行通信，但SHT10的串行接口与I2C通用接口不兼容，因此需要使用通用I/O对串行接口进行模拟。具体原则为DATA总线在SCK时钟下降沿之后改变状态，并仅在SCK时钟上升沿有效。数据传输期间，在SCK时钟高电平时，DATA必须保持稳定。为避免信号冲突，微处理器应驱动DATA在低电平，即MCU不输出数据时DATA应为高电平。因此，需要一个外部的上拉电阻（例如：10kΩ）将信号提拉至高电平。

3.3.2 超声波传感器模块实现计时功能设计思路

超声波传感器是利用超声波的特性研制而成的传感器。在输入端，由换能晶片在180kHz脉冲输入电压的激励下发生震动产生超声波；在输出端，换能晶片再将接收到的超声波信号转换为频率为180kHz的正弦波包络。因此，该模块的硬件设计包括将单片机输出的脉冲信号的峰峰值放大到10V和输入端将1mV左右的信号放大并对放大之后的信号进行电压过零比较。

软件设计思路为输入端计数器采用匹配输出，经过实测可得当把计数器的匹配寄存器配置为132时输出的矩形脉冲为180kHz，在匹配输出的同时使能另一个计数器的捕获输入功能，当超声波传感器发出的正弦波经过放大和过零比较为矩形脉冲波输入到计数器引脚时，计数器产生中断将捕获寄存器中的数取出即为发送到接收经历的时间。

3.3.3 蓝牙接收端模块程序设计思路

通过蓝牙接收数据的程序是基于Android OS的手机开发的。使用主线程（UI线程）实现手机屏幕中控件的显示，并新建一个线程实现对数据的接收。在JAVA中实现多线程有两种途径：继承Thread类或者实现Runnable接口。但在程序开发中只要是多线程肯定永远以实现Runnable接口为主，因为实现Runnable接口相比继承Thread类有如下好处：

1. 避免点继承的局限，一个类可以继承多个接口；
2. 适用于资源的共享。

将接收的字节型数据进行处理即可得到目标数据。

4 硬件设计与开发