# 摘要

智能家居已逐渐成为一个人人都熟悉的概念，每年都有厂商推出数不清的产品来为我们构建未来智能家居的愿望。越来越多的智能设备出现在市场中，如ISee mini智能投影、智能路由器、智能烤箱等。2017年CES大会，我国国内厂商TCL、海信、创维以及海尔等展出多种智能家电。智能家电的多样化将推动我国智能家居自动控制系统的发展，具备环境监测与自动控制功能的智能家居系统将成为我国未来智能家居发展趋势。

本课题设计的智能家居环境监测与自动控制系统是利用传感器技术、ZigBee通信技术以及嵌入式技术，通过用户界面实现对多个智能家居设备的统一监测或控制。该系统可实现室内环境的监测，将智能家居设备联网完成自动控制，同时还可以手动控制智能家居设备。

本系统具体采用CC2530 芯片作为ZigBee无线网络收发模块，利用ZigBee传感器节点采集数据信息，通过ZigBee协调器组建无线传感网完成数据采集，然后通过协调器将采集到的数据发送至QT设计的上位机。系统通过温湿度和光线传感器来实时采集室内温湿度和光线信息，并根据获取到的信息利用继电器进行风扇及灯光的自动控制，同时根据实际需要也可以实现手动控制。

**关键词：**智能家居；环境监测；自动控制；传感器；ZigBee通信技术；QT

Abstract

Smart home has gradually become a everyone familiar with the concept, has launched numerous products every year to build the future for us the desire of the smart home. More and more intelligent device appeared in the market, such as ISee mini smart projection, intelligent router, smart oven, etc. CES conference in 2017, China's domestic manufacturers, TCL, hisense, skyworth and haier display a variety of intelligent home appliances. Diversity of the smart appliances will promote the development of intelligent household automatic control system in our country, environmental monitoring and automatic control function of smart home system will become the development trend of future smart home.

This topic design of intelligent household environment monitoring and automatic control system is the use of sensor technology, ZigBee communication technology and embedded technology, through a user interface to the unity of the multiple intelligent household equipment monitoring and control. The system can realize indoor environment monitoring, smart home devices connected to the Internet to complete automatic control, at the same time can also be manually controlled intelligent household equipment.

This system specific USES CC2530 chip as ZigBee wireless transceiver module, data acquisition using ZigBee sensor node information, through the ZigBee coordinator to form a wireless sensor network to complete data collection, and then through the coordinator will be sent to the data collected QT upper machine design. System with temperature and humidity, and light and light sensors to collect real-time indoor temperature and humidity information, and according to the access to information using relay fan and light automatic control, also can realize manual control according to the actual needs.

**Keywords:** Intelligent household; Environmental monitoring; Automatic control; The sensor; ZigBee communication technology; QT

目录

[1 引言 1](#_Toc20631)

[1.1背景及意义 1](#_Toc9627)

[1.2研究现状 1](#_Toc13201)

[1.3论文结构 2](#_Toc7900)

[2 ZigBee协议及组网 3](#_Toc10471)

[2.1 ZigBee简介 3](#_Toc17586)

[2.2 ZigBee协议框架 3](#_Toc10874)

[2.3 ZigBee协议栈 4](#_Toc22745)

[2.4 ZigBee组网流程 4](#_Toc19725)

[3 总体方案设计 7](#_Toc22315)

[3.1系统设计方案 7](#_Toc5774)

[3.1.1系统结构框图 7](#_Toc18782)

[3.1.2运行流程框图 8](#_Toc32626)

[3.2上位机功能介绍 9](#_Toc19803)

[3.3下位机功能介绍 10](#_Toc16413)

[4 硬件设计 11](#_Toc2963)

[4.1上位机设计 11](#_Toc21107)

[4.2下位机设计 11](#_Toc1216)

[4.2.1 SHT10温湿度传感器模块 11](#_Toc4353)

[4.2.2 TPS852光线传感器模块 12](#_Toc6902)

[4.2.3继电器模块 13](#_Toc26721)

[5 软件设计 14](#_Toc6127)

[5.1上位机设计 14](#_Toc29893)

[5.1.1自然模式设计流程 15](#_Toc17138)

[5.1.2自动控制模式设计流程 16](#_Toc29900)

[5.1.3手动模式设计流程 16](#_Toc27341)

[5.2协调器数据发送与接收 18](#_Toc1037)

[5.3温湿度传感器模块软件设计 19](#_Toc11444)

[5.4光线传感器模块软件设计 20](#_Toc25345)

[5.5继电器模块软件设计 21](#_Toc18178)

[6 系统测试与分析 23](#_Toc32561)

[6.1协调器与上位机的串口通讯测试 23](#_Toc32570)

[6.2 ZigBee组网测试 24](#_Toc17873)

[7 结论 25](#_Toc7339)

[参考文献 26](#_Toc23596)

[致谢 27](#_Toc17836)

1、引言

## 1.1背景及意义

物联网是继互联网与移动通信网之后的又一次信息产业浪潮。物联网以其便捷、灵活、实时性的优势，在各个领域迅速发展起来。随着经济水平不断提高，科学技术的不断创新，人们对其家居生活环境的要求也越来越高，以往传统的家居环境已经不能够满足现代人们对高舒适度、高智能化家居的要求。智能家居的发展也由此拉开序幕。

## 1.2研究现状

2016年，全球范围内信息技术创新不断加快，我国市场规模庞大，正处于居民消费升级和信息化、工业化、城镇化、农业现代化加快融合发展的阶段。我国政府为了推动信息化、智能化城市发展也发表了关于促进信息消费扩大内需的若干意见，大力发展宽带普及、宽带提速，加快推动信息消费持续增长，这都为智能家居、物联网行业的发展打下了坚实的基础。

近几年，智能家居在全球范围内呈现强进的生命力。2016年，美国以5.8%的普及率位居第一，而中国仅为0.1%。造成国内智能家居发展缓慢的主要原因是国内智能家居生态体系尚未完全建立，设备兼容差、操作复杂、协议标准的统一等问题还未完全解决，消费者难以深切体会到智能家居的智慧与便捷。除此之外，智能家居设备的价格与消费者的消费习惯也是阻碍国内智能家居普及的重要原因之一。

尽管如此，国内智能家居市场仍被十分看好，目前国内知名厂商已推出的智能家居平台有海尔的智能U+、美的的M-Smart、华为的HiLink和小米的米家等，这从相关厂商的态度和金融机构的投资热情中都能看出。首先，我国整体经济形势稳中有进，中产阶级崛起，消费不断升级;其次，中国成为全球最大的互联网市场，制造业与互联网融合发展逐步深入，这两大因素使得国内智能家居市场具备了迅速爆发的潜力。

## 1.3论文结构

本设计论文结构及主要内容如下：

1. ，引言。本章介绍了项目的背景以及目前智能家居发展的现状，同时介绍了论文的主要结构。
2. ，ZigBee协议及网络。本章详细介绍了ZigBee技术以及其组网用到的Z-Stack协议栈，并说明Z-Stack协议栈运行流程。
3. ，总体方案设计。本章提出本系统上下位机设计的总体方案以及运行流程。
4. ，硬件设计。本章介绍了该系统所用到的硬件设备，即上位机用到的开发板，下位机所用到的各个传感器节点。
5. ，软件设计。本章提出了本系统的软件设计，包括上位机的界面功能设计，下位机的各个传感器的数据格式等。
6. ，系统测试与分析。本章说明系统在设计过程中遇到的问题以及解决方法。
7. ，结论。本章首先得出该系统达到的功能，其次提出本系统所存在的问题，最后对智能家居的展望。

# ZigBee协议及组网

ZigBee技术是一种功耗低、距离短的无线通信技术，其遵循的协议是IEEE802.15.4，支持点对点、一对多通信。目前ZigBee技术已在医疗、农业、电力、智能化控制等领域有着广泛的运用。

## 2.1 ZigBee简介

ZigBee技术是一种新兴的短距离无线通信技术，建立在IEEE 802.15.4标准之上，采用的频段主要是在2.4GHz(全球)、868MHz(欧洲)、和915MHz(美国)三个频段，分别提供的传输速率是250kb/s、20kb/s、40kb/s，可容纳设备节点数为254个。ZigBee技术具有低功耗、低成本、时延小、低速率、有效范围小、可靠性高和安全性好等特点。低功耗，由于ZigBee网络设备节点工作周期短、收发数据功耗低，不接收数据时处于休眠状态，接收数据则由协调器唤醒节点设备。据估算，两节五号电池可维持其工作长达6个月到两年。成本低，则是由于ZigBee协议栈设计简单且免专利费，节点硬件设备成本较低。时延小，是由于其通信时延与唤醒时延都非常短，搜索设备时延为30ms，唤醒时延为15ms,与蓝牙等技术的时延相比，各项指标值都非常小。因此ZigBee技术主要用于距离短、功耗低的各种电子设备之间进行数据传输以及典型的有周期型数据、间歇性数据和低反应时间数据传输的应用，非常适用于家电和小型电子设备的无线控制指令传输。

## 2.2 ZigBee协议框架

ZigBee协议栈主要有物理层、MAC层、网络层、安全层、应用层组成。物理层和MAC层由IEEE 802.15.4标准定义，网络层、安全层和应用接口层由ZigBee联盟制定。MAC层可提供与网络层接口，应用接口层负责向用户提供简单的应用软件接口。如图3-1所示。

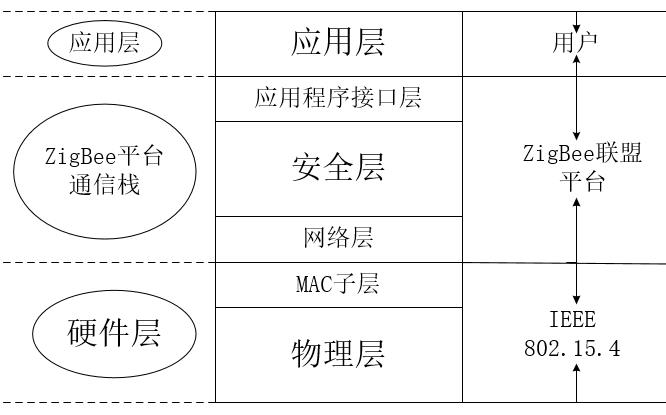


图2-1 ZigBee协议框架图

## 2.3 ZigBee协议栈

Z-Stack是TI公司开发的ZigBee协议栈，其运行环境是瑞典公司IAR开发的IAR Embedded Workbench for 8051。Z-Stack是一款经过ZigBee联盟认可的商业及协议栈，可以帮助程序员更好、更方便开发ZigBee。在ZigBee网络中有三种设备类型：协调器（Coordinator）、路由器(Router)、终端设备(End-Device)。协调器的功能主要是启动网络，选择一个空闲的信道以及一个PANID，然后启动网络。路由器主要是允许其他设备加入网络，终端设备可以休眠或者被唤醒加入网络。

ZigBee在2.4G频段上，共划分了16个信道（11~26），每个信道的频带宽度为5MHz。各个设备节点主要通过16位的PANID来区分网络，PANID相同的设备节点则在同一个ZigBee网络内，这样就可以使得同一地区可以存在多个ZigBee网络。

此外，ZigBee的每个设备节点均有两种类型的地址，一种是64位的MAC地址，另一种是16位的网络地址。MAC地址是设备由制造商或被安装时设置的，在设备节点的生命周期中一直拥有它，是全球唯一的地址。网络地址是设备加入网络由协调器分配的，在网络中是唯一的，用来在网络中鉴别设备和发送数据。

## 2.4 ZigBee组网流程

建立一个ZigBee网络主要是利用Z-Stack协议栈来组网进行数据的发送与接收，在组网前要在Z-Stack协议栈中对设备类型、信道、PANID等网络参数进行设置，设置完成后将协议栈下载至设备节点上的CC2530模块中，在该模块会从Z-Stack协议栈中main()函数开始运行，进行系统初始化以及执行函数osal\_start\_system(),进入轮转查询式操作系统事件的死循环，不停地调用osal\_run\_system()处理发生事件。具体流程如图2-2所示。



图2-2 Z-stack协议栈main()函数流程图

ZigBee网络中数据的发送与接收主要通过Z-Stack协议栈中的无线接收与发送函数实现的，协调器与上位机是通过串口实现数据的发送与接收。对于串口初始化的配置就是配置波特率、数据位、流控制、校验位和停止位等等。Z-Stack MT层提供了串口初始化函数MT\_UartInit();进入函数可以看见串口初始化的结构体：

uartConfig.configured = TRUE;

uartConfig.baudRate = MT\_UART\_DEFAULT\_BAUDRATE;

uartConfig.flowControl = MT\_UART\_DEFAULT\_OVERFLOW;

uartConfig.flowControlThreshold= MT\_UART\_DEFAULT\_THRESHOLD;

uartConfig.rx.maxBufSize= MT\_UART\_DEFAULT\_MAX\_RX\_BUFF;

uartConfig.tx.maxBufSize= MT\_UART\_DEFAULT\_MAX\_TX\_BUFF;

uartConfig.idleTimeout= MT\_UART\_DEFAULT\_IDLE\_TIMEOUT;

uartConfig.intEnable= TRUE;

通过改变相对应的宏定义设置来初始化串口，设置波特率为38400，关闭流控（如果串口只连接RX和TX一定要关闭流控，否则不能发送信息），其余保持默认配置即可。

# 3、总体方案设计

## 3.1系统设计方案

智能家居环境监测与自动控制系统的设计是基于ARM Cortex-A8为核心处理器，以嵌入式版本的linux为操作系统，采用ZigBee通信技术实现设备节点网络的组建。

ARM Cortex-A8处理器是第一款基于ARMV7架构的处理器，可以运行Linux、WinCE、Android等常用的嵌入式实时操作系统，无论是在消费电子、汽车电子、医疗电子和工控领域都扮演着重要的角色。

Linux操作系统是以Unix操作系统为基础，常用于嵌入式软件开发中。本系统运行环境为嵌入式Linux操作系统，由于其具有源码开放、可移植、内核小、速度快、安全性能高和性能稳定等特点，常运用于开发领域。

### 3.1.1系统结构框图

本系统的在硬件设备连接方式主要分为两部分，一部分是使用RS232串口线进行连接，另外一部分则用ZigBee无线连接。上位机与协调器利用串口线进行数据与指令的传送，协调器与各个传感器节点之间通过ZigBee无线连接方式进行数据传输。整体连接图如图3-1所示。 

图3-1 整体连接图

### 3.1.2运行流程框图

本系统的上位机是以嵌入式Linux为运行环境，ARM Cortex-A8为核心处理器，通过Linux版本的QT软件实现上位机控制界面的设计。下位机则是由传感器模块构成，用来采集数据，利用ZigBee模块将指令发送给协调器，经由协调器通过串口将数据发送给上位机，通过上位机可以实现环境温湿度实时监测以及控制指令的发送，本系统的总体运行流程图如图3-2所示。



图3-2 系统总体运行流程图

本系统共有三种模式，分别是自动控制模式、自然模式与手动模式。选择自动模式，系统可以根据用户设置的温湿度值和光线值自行控制开关；选择自然模式，设备节点会根据自然情况下温湿度值和光线值自行控制开关；手动模式则可以通过按键控制设备的开关。具体模式运行流程图如图3-3所示。



图3-3 具体模式运行流程图

## 3.2上位机功能介绍

本系统上位机的显示界面是基于Linux的QT程序设计，主要实现数据接收并显示以及对设备的控制。当所有传感节点全部开启，上位机实现数据的实时监测。本系统共设计三种模式，分别为手动模式、自动控制模式和自然模式，系统默认模式为自然模式，手动模式在自动控制模式和自然模式下均可使用。手动模式，用户则可以直接通过界面上的按键对设备进行控制；当选择自动控制模式时，用户需先设定临界值，然后上位机将接收到的数据与临界值进行对比，自行发送相应控制指令；当由自动控制模式切换为默认的自然模式时，上位机将根据室内环境情况自行发送相应控制指令。

以光线传感器为例，三种模式具体处理方式说明如下。光线传感器检测到数据后，通过ZigBee模块将数据发送至协调器，协调器通过串口将数据发送给上位机进行显示。手动模式即用户可以直接通过界面上的按键打开或关闭灯光；当选择自动控制模式时，光线值小于用户设定的光线下限值时，上位机则会发出打开灯光指令到协调器，协调器再将指令发送给ZigBee继电器模块，然后打开灯光；当切换回自然模式时，若外界环境黑暗，上位机则自行发送打开灯光指令。

## 3.3下位机功能介绍

本系统下位机主要由四个Zigbee节点组成，分别是温湿度传感器、光线传感器和两个继电器节点。

温湿度传感器节点用来采集室内环境的温度和湿度，光线传感器用来采集室内光线值，两个传感器采集到数据后通过Zigbee模块将数据发送到协调器，协调器再通过串口发送至上位机，上位机根据收到的数据发出相应指令进行控制。

两个继电器节点分别用来控制风扇和灯光的开关。当上位机发送关闭设备指令至协调器时，协调器将指令发送至继电器节点上的Zigbee模块，继电器节点接收到关闭指令后，断开电路，使得设备关闭。同理，若继电器节点接收到打开指令，则连通电路，使得设备关打开。

# 

# 4、硬件设计

## 4.1上位机设计

本系统的上位机平台是以无锡泛太嵌入式试验箱为平台，是整个系统的核心。该上位机平台以ARM Cortex-A8开发板为核心处理器，ARM Cortex-A8处理器是第一款基于ARMV7架构的处理器，其核心芯片为E210，该处理器集成了LCD控制器、USB Host、NAND Flash控制器、中断控制器、内存控制器、SPI、UART、I2C和丰富的GPIO资源，通过外部的存储器、JTAG 接口、串口等实现硬件平台。而传感器节点采集到的数据将通过以CC2530为核心芯片的ZigBee协调器上的串口（UART）传送至ARM Cortex-A8，利用ARM Cortex-A8上的LCD显示屏对数据进行显示以及实现控制命令的输入。上位机硬件设计图如图4-1所示。



图4-1 上位机硬件设计图

## 4.2下位机设计

本系统下位机由一个风扇、一个灯和四个Zigbee节点组成。四个Zigbee节点分别是温湿度传感器、光线传感器和两个继电器节点。

### 4.2.1 SHT10温湿度传感器模块

SHT10温湿度传感器模块是集测量温度和湿度于一体，具有湿度传感器自检测能力的传感器元件。该元件是两线制数字接口，具有高可靠性和长期的优化稳定性。本元件温度的测量范围大致在-40~+123.8℃，湿度的测量范围在0至100%，其响应时间小于20S。温湿度实物图如图4-2所示。



图4-2温湿度实物图

本模块连接在CC2530模块的P1\_1和P1\_2引脚，CC2530的P1\_2引脚发送时钟信号，SHT10芯片通过CC2530上的P1\_1引脚将采集的数据发送至CC2530模块。连接图如图4-3所示。



图4-3 温湿度传感器模块连接图

### 4.2.2 TPS852光线传感器模块

TPS852光线传感器以大范围的输入光为测量对象，内置A/D转换器，可将光信号由模拟信号转换为数字信号，具有高灵敏度、工作稳定和可靠，检测的范围广的优点。本模块连接在CC2530模块的P1\_7引脚。实物图如图4-4所示。



图4-4 光线传感器模块实物图

连接图如图4-5所示。



图4-5 光线传感器模块连接图

### 4.2.3继电器模块

本模块由CC2530模块和继电器组成，当CC2530模块收到使得继电器闭合的指令时，会给继电器上的电磁铁通电，使得衔铁被吸下来，工作电路闭合。实物图如图4-6所示。

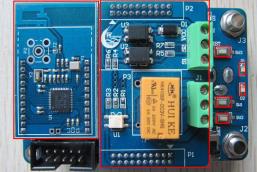


图4-6 继电器模块实物图

连接图如图4-7所示。P1\_0控制继电器开关，1为关闭，0为接通；P1\_1为固定值1；P1\_2为固定值0。



图4-7 继电器模块连接图

# 5、软件设计

## 5.1上位机设计

本系统上位机软件部分主要是基于Linux系统下的QT人机交互界面设计。界面左下方可显示串口信息和接收到的数据，界面左上部分可显示室内当前温湿度和光线以及分别控制灯和风扇的继电器地址，右上方可以设置光线和温度值，右下方是设置值时使用的按键。上位机界面 显示如图5-1所示。

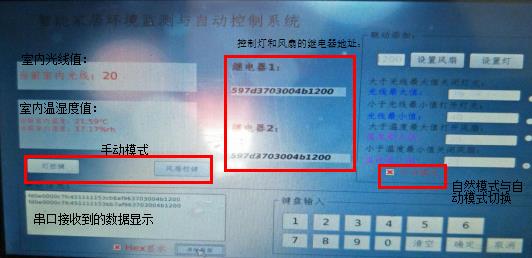
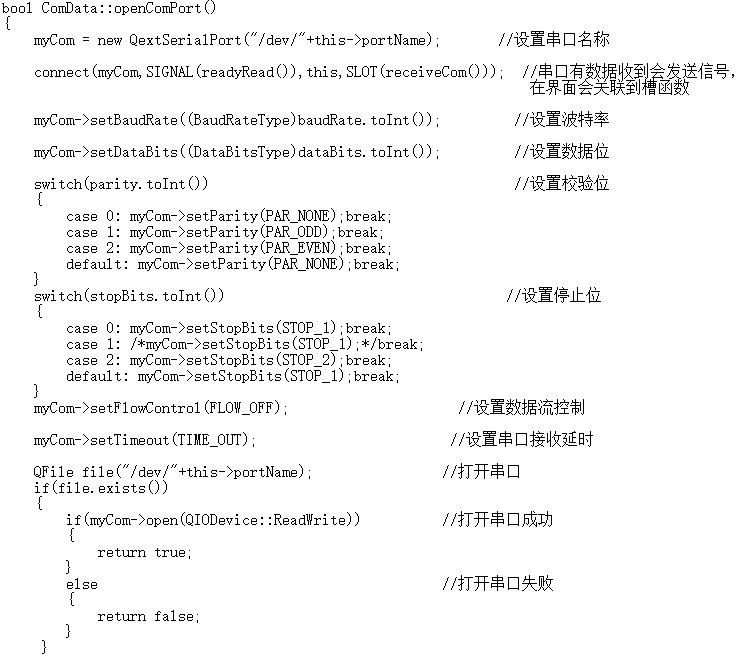


图5-1 上位机界面显示图

核心代码如下：



### 5.1.1自然模式设计流程

在自然模式下，用户不必设置任何参数，串口传来的数据值后，首先会判断数据类型。当数据为光线类，在默认范围之间，不会做出任何处理，等待串口再次传来数据；若数据小于默认范围，串口会发送打开灯光命令；若数据大于默认范围，串口会发送关闭灯光命令。当数据为温湿度类，在默认范围之间，不会做出任何处理，等待串口再次传来数据；若数据小于默认范围，串口会发送关闭风扇命令；若数据大于默认范围，串口会发送打开风扇命令。流程如图5-2所示。



图5-2 自然模式流程图

### 5.1.2自动控制模式设计流程

在自动控制模式下，用户需先设定自认为合适的光线值和温度值，完成设置参数后，联动模式才有效。串口传来的数据值后，会判断数据类型，然后与设置的参数进行对比，对比完成后发送相应的命令。例如：数据为光线类，在用户设置的最小最大值之间，不会做出任何处理，等待串口再次传来数据；若数据小于用户设置的最小值，串口会发送打开灯光命令；若数据大于用户设置的最大值，串口会发送关闭灯光命令。流程如图5-3所示。



图5-3 自动控制模式流程图

### 5.1.3手动模式设计流程

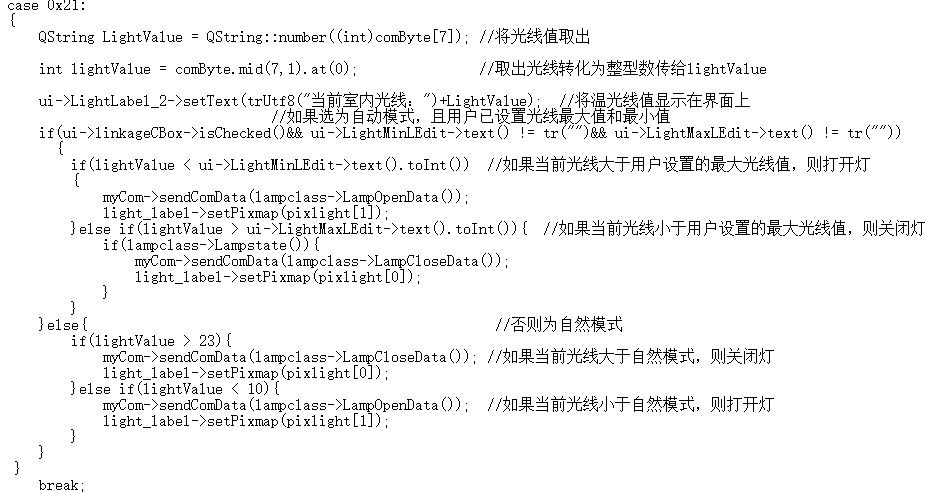
手动模式不用选择，无论上位机处于自然模式还是联动模式，用户都可通过界面按键直接控制灯光或者风扇的开关。流程如图5-4所示。



图5-4 手动模式流程图

核心代码如下：





## 5.2协调器数据发送与接收

在本系统中，协调器有着非常重要的作用。首先，协调器需要组建网络，然后为每个设备节点分配网络地址。具体运行流程如图5-5所示。



图5-5 协调器运行流程图

其次，各个设备终端节点采集到的数据必须经由协调器通过串口发送至上位机，而上位机向终端节点发送控制命令也必须先由上位机的串口发送至协调器，然后协调器再以无线发送数据的方式将控制命令传至终端节点。协调器在接收数据时，通常会先判断接收到数据的第一个字节是否与数据的标志位相同，从而可以取得完整的数据。设备节点发送的数据包含很多信息，如：标志位、长度、父节点地址、本节点地址、类型、数据、校验和等。具体协调器通用数据格式如表5-1所示。

表5-1协调器通用数据表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 标志 | 长度 | 父节点  地址 | 本节点  地址 | 类型 | 数据值 | 校验和 | IEEE地址 |
| FDH | 03H | 00H 00H | 05H  09H | 41H |  | 98H | 01H 02H 03H 04H 05H 06H 07H 08H |
| 数据标志 | 类型+数据+校验和+IEEE地址 | 父节点短地址 | 本身的短地址 | 传感器类型 | 传感器数据 | (类型+数据)/256 | Zigbee的IEEE地址（永远在最后8位） |

## 5.3温湿度传感器模块软件设计

温湿度传感器模块在本系统中用来采集温湿度，属于采集类设备。采集到数据后，按下本模块上的按键便可以向协调器发送数据。具体数据格式如表5-2所示。

表5-2 温湿度传感器数据表

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 名称 | 标志 | 长度 | 父节点  地址 | 本节点  地址 | 类型 | 数据值 | 校验和 | IEEE地址 |
| 内容 | FDH | 03H | 00H 00H | 05H  09H | 45H （E） | 00H(湿度整数), 00H(湿度小数)%RH, 00H(温度整数),00H(温度小数)oC(摄氏度) | 00H |  |
| 字节数 | 1字节 | 1字节 | 2字节 | 2字节 | 1字节 | 4字节 | 1字节 | 8字节 |

当接收数据的第12字节也就是校验和与发送的校验和相同时，则接收的数据无误，可以继续传至上位机进行处理，上位机先需判断类型，然后取出数据中的数据值四个字节，转化成整型显示在界面。

图5-6为串口助手接收到的温湿度数据:



图5-6 串口助手接收温湿度数据图

分析图5-6：

FDH：标志位；0EH: 类型位到最后一个字节的数据长度；00H 00H：父节点地址；C7H FBH: 本节点地址；45H：节点类型为温湿度；20H 21H 19H 28H：转化为十进制后，湿度为32.33，温度为25.40；C7H：校验和；AFH 96H 37H 03H 00H 4BH 12H 00H：64位IEEE地址。

## 5.4光线传感器模块软件设计

光线传感器模块在本系统中用来采集室内光线值，属于采集类设备。采集到数据后，按下本模块上的按键便可以向协调器发送数据。具体数据格式如表5-3所示。

表5-3 光线传感器数据表

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 名称 | 标志 | 长度 | 父节点  地址 | 本节点  地址 | 类型 | 数据值 | 校验和 | IEEE地址 |
| 内容 | FDH | 03H | 00H 00H | 05H  09H | 21H | 00H | 00H |  |
| 字节数 | 1字节 | 1字节 | 2字节 | 2字节 | 1字节 | 1字节 | 1字节 | 8字节 |

上位机接收到正确的数据后，先判断类型，若第七字节为光线类，则取出第八字节的内容，将其转换为整型显示在上位机界面。

图5-7为串口助手接收的光线数据:



图5-6 串口助手接收光线数据图

分析图5-7：

FDH：标志位；0BH: 类型位到最后一个字节的数据长度；00H 00H：父节点地址；C7H FCH: 本节点地址；21H：节点类型为温湿度；29H：转化为十进制后，光线值为41；4AH：校验和；CAH 97H 37H 03H 00H 4BH 12H 00H：64位IEEE地址。

## 5.5继电器模块软件设计

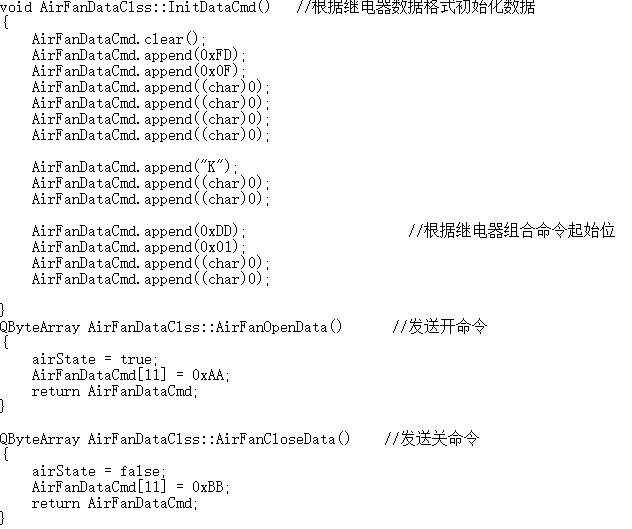
继电器模块在本系统中用来控制设备的开或关，属于受控类设备。上位机发送指令后，通过协调器将指令传送至继电器模块，继电器模块分析数据判断收到的命令位是开还是关，若是开则将设备电路连通，关则断开设备电路。具体数据格式如表5-4所示。

表5-4 继电器数据表

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 名称 | 标志 | 长度 | 节点  地址 | 本机  地址 | 类型 | 节点  地址 | 组合命令 | 校验和 | IEEE地址 |
| 内容 | FDH | 0EH | x1H x2H | 00H  00H | 4BH  （K） | x1H x2H |  | 00H |  |
| 字节数 | 1字节 | 1字节 | 2字节 | 2字节 | 1字节 | 2字节 | 3字节 | 1字节 | 8字节 |

在该数据中的组合命令的内容为3个字节的开继电器或关继电器指令，继电器模块在收到完整数据后，将分析该数据中组合命令内容。若三个字节分别为DDH、01H、AAH，则继电器为开；若三个字节分别为DDH、01H、BBH，则继电器为关。

核心代码如下：



# 系统测试与分析

## 6.1协调器与上位机的串口通讯测试

智能家居环境监测与自动控制系统的界面设计是基于Linux系统下的QT人机交互界面，Linux下的QT是不提供串口类函数的，所以选取了第三方串口类qextserialport实现串口的驱动。在测试过程中，主要遇到以下问题并解决。

1. 现象：串口无法接受数据。如图6-1。

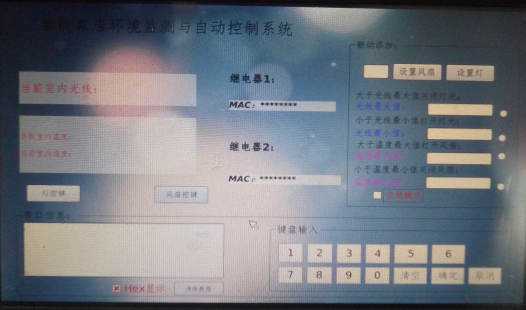


图6-1

原因：串口部分程序出现问题，没有找到，重新写了串口程序。

解决方法：认真阅读了实验案例以及查找网上资料观看视频，完成串口部分内容，可以显示数据。如图6-2。

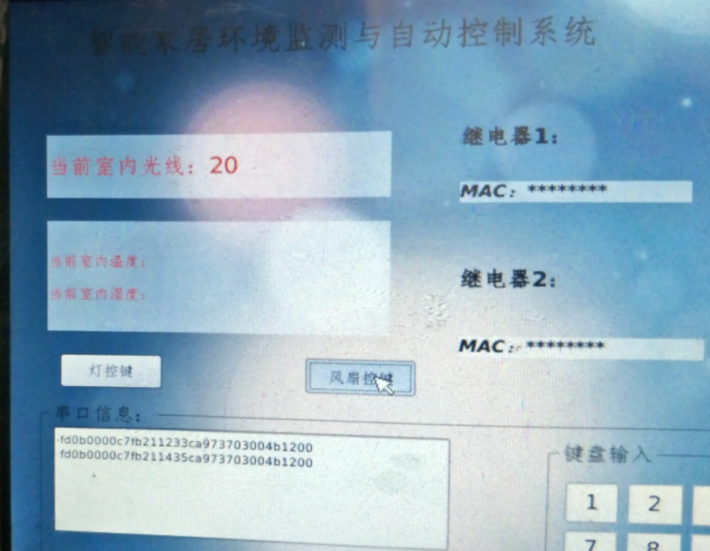


图6-2

2.现象：串口可接收数据后，只能够显示温湿度数据，自动控制不能实现，同理自然模式也无法实现。

错误原因：经过参考网上案例，查阅资料，发现由于从节点传回的数据取出温度的高位和低位后存放在的字符串型的tempValue里，然后直接用tempValue与用户设定的值进行比较。

解决方法：利用Qstring中的函数mid( int position, int n = -1)和at(int position)，这样可以将温度的高位取出并将其返回所在数组，然后传给设置的int型temp得到温度高位，在用temp与用户设定的值进行比较。如图6-3。

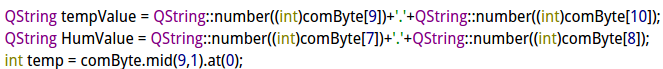
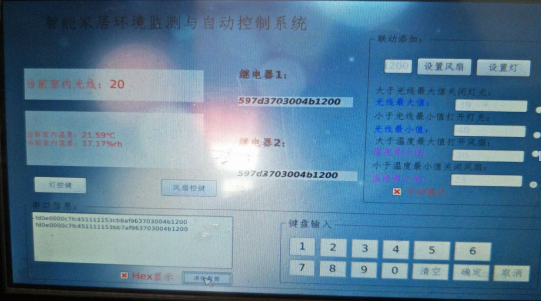


图6-3

## 6.2 ZigBee组网测试

本系统在ZigBee组网过程中，首先需安装Z-Stack协议栈，并配置IAR，然后选择设备类型，修改信道值，将所有的设备节点信道值设为同一个，然后依次下载至各个节点中。

组网成功后，各节点上的D1会先闪烁三下表示正在寻找网络。若寻找到网络D2会闪烁一下，表示节点加入网络；若没找到网络D1会一直闪烁。完成组网后，上位机界面显示如图6-4所示。

上位机界面显示图6-4

# 7、结论

在智能家居环境监测与自动控制系统中，ARM Cortex-A8主机上的界面是整个系统的核心，也是人机交互重要部分，ZigBee终端节点采集的温湿度和光线信息通过无线发送至协调器，协调器发送至A8的QT界面实现环境监测，界面上的按键、自然模式以及联动模式实现室内环境的自动控制。

本系统在ZigBee技术的基础上，利用传感器节点实现了温湿度和光线的采集、灯与风扇设备节点的控制。在此基础之上添加了自然模式与联动模式，使得家居更加智能化，更加节省人力与电力。

每个系统都有其缺陷，本系统也不例外。从技术方面来看，本文主要通讯技术为ZigBee无线通讯，目前该技术存在问题是穿透性不好，障碍物厚度等因素将严重影响通信距离。从界面来看，该系统在扩展性方面还不够完善，无法添加其他新的节点。传感器节点的位置对采集到的数据也有影响，界面单凭一个节点传来的数据便作出判断，并不十分可靠。因此，本系统还应该多添加几个相同类型节点，并且将多个节点放置在室内不同位置，在采集到数据传来后，应取平均值，然后再做出相应控制。从硬件角度来看，目前智能家居设备价格并不是很便宜，这也是智能家居面临的主要问题之一。本系统今后会在此基础之上做出完善，希望今后可以运用在实际生活之中。

智能家居有很好的应用前景，它既符合现代人对高品质生活的追求，又符合人们节能环保的生活追求。国家大力扶持物联网产业，相信在未来几年会有很大的进展，成本方面也会有很大的突破，到时候人人都能智能生活。

# 参考文献

[1] 张鸿涛.物联网关键技术及系统应用[M]北京：机械工业出版社，2012.5.

[2] 杜军朝.Zigbee原理与实战[M] 北京:机械工业出版社,2015.2.

[3] 王小强.ZigBee无线传感器网络设计与实现[M] 北京:化学工业出版社, 2012.6.

[4] （加）布兰切特 （英）萨默菲尔德.C++ GUI Qt 4编程（第二版）[M] 北京:电子工业出版社，2013.5.

[5] 霍亚飞.Qt Creator快速入门(第2版)[M] 北京:航空航天大学出版社,2014.1.

[6] 韩少云 奚海蛟 谌利.基于嵌入式Linux的Qt图形程序实战开发[M] 北京:航空航天大学出版社，2012.10.

[7] 于海斌 曾鹏.智能无线传感器网络系统[M]北京：科学出版社，2006.

[8]张新顺.基于ZigBee技术的家居环境监控系统设置[D]大连：大连工业大学, 2012.3

[9]刘勇.基于物联网的智能家居控制系统的设计[D]西安：西安科技大学,2015.3

[10]郭奇.智能家居控制平台的设计与实现[D]沈阳：沈阳理工大学,2012.3