

**课 程 设 计**

**课 程：物联网工程开发与应用实例**

**题 目：基于ZigBee的家居安防系统设计**

**学生姓名：谢作程**

**学 号：15115061028**

**二级学院：信息科学与工程学院**

**专 业：物联网工程**

**班 级：2015级01班**

**指导教师姓名及职称：刘晓樑 讲师**

**起止时间：2018年 9 月—— 2019年 1月**

（教务处制）

**基于ZigBee的家居安防系统设计**

**摘 要**：随着科技的发展，越来越多的人开始重视自身隐私的安全。隐私安全不仅仅是个人信息的安全，同时也是家居隐私的安全。目前，绝大部分家庭对房屋的安防措施主要是利用普通的门窗锁，而市面上的智能门锁价格又非常昂贵，为此我们设计了这套智能家居安防系统，在传统门窗的基础上，只需要投入少许资金就能实现温湿度、人员入侵的检测，在发生险情时系统会自动报警。

本套系统主要使用了ZigBee技术，同时QT作为应用层来实时显示温湿度等信息，同时在收集数据的基础上，把数据存入数据库，使用户能够对以往数据进行管理及分析。手机端同时有APP提供给用户实现对系统的无线控制以及数据的实时查看。一目了然地掌控家居安防。

**关键字**：ZigBee；QT；数据库；APP；家居安防

Design of home security system based on ZigBee

**Abstract:** With the development of technology, more and more people are beginning to pay attention to the security of their privacy. Privacy and security are not only the security of personal information, but also the security of home privacy. At present, most home security measures for houses mainly use ordinary door and window locks, and the price of smart door locks on the market is very expensive. For this reason, we designed this smart home security system based on traditional doors and windows. Only a small amount of money is needed to realize the detection of temperature and humidity and personnel intrusion, and the system will automatically alarm when a dangerous situation occurs.

This system mainly uses ZigBee technology. At the same time, QT serves as the application layer to display information such as temperature and humidity in real time. At the same time, based on the data collection, the data is stored in the database, enabling users to manage and analyze the previous data. At the same time, the mobile terminal provides the user with the wireless control of the system and the real-time viewing of the data. Control home security at a glance.

**Key words:**ZigBee;QT;Database;Home security

**目录**

[1 家居安防设计的相关技术 1](#_Toc25217)

[1.1 ZigBee技术 1](#_Toc329)

[1.2 Z-Stack协议栈 1](#_Toc494)

[1.3 IAR开发环境 2](#_Toc2863)

[1.4 WAMP集成开发环境 2](#_Toc9849)

[1.5 Qt Creator 3](#_Toc16055)

[2 系统总体设计方案 4](#_Toc17614)

[2.1 系统总体设计 4](#_Toc16223)

2.2 各部分功能 4

2.2.1 数据采集部分 4

2.2.2 网关部分 5

2.2.3 无线控制部分 5

2.2.4 数据库部分 5

[3 系统软件设计 6](#_Toc3124)

[3.1 Zigbee数据收发部分 6](#_Toc4016)

[3.1.1 协调器回调函数 6](#_Toc13650)

[3.1.2 协调器消息处理函数 8](#_Toc30109)

[3.1.3 终端消息发送函数 1](#_Toc21124)0

[3.2 传感器检测部分 1](#_Toc4016)3

[3.2.1 DHT11温湿度传感器 1](#_Toc13650)3

[3.2.2 红外热释电传感器](#_Toc30109) 15

[3.3 QT网关程序](#_Toc4016) 16

[3.4 安卓APP 1](#_Toc22185)7

[4 系统测试 1](#_Toc27057)7

[4.1 硬件测试 1](#_Toc7215)7

[4.2 软件测试 1](#_Toc28516)8

[5 总结与展望 1](#_Toc30836)9

[5.1 总结 1](#_Toc16776)9

[5.2 展望 1](#_Toc24964)9

[参考文献](#_Toc12103) 20

**基于ZigBee的家居安防系统设计**

**1 家居安防设计的相关技术**

* 1. ZigBee技术

ZigBee技术是一种近距离、低复杂度、低功耗、低速率、低成本的双向无线通讯技术。主要用于距离短、功耗低且传输速率不高的各种电子设备之间进行数据传输以及典型的有周期性数据、间歇性数据和低反应时间数据传输的应用。

ZigBee模块是一种物联网无线数据终端，它利用ZigBee网络为用户提供无线数据传输功能。该产品采用高性能的工业级ZigBee方案，提供SMT与DIP接口，可直接连接TTL接口设备，实现数据透明传输功能；低功耗设计，最低功耗小于1mA；提供6路I/O，可实现数字量输入输出、脉冲输出；其中有3路I/O还可实现模拟量采集、脉冲计数等功能。

同时ZigBee技术是本设计的重要组成部分，本系统是基于ZigBee开发并实现的。在本系统中，用了两个ZigBee节点，一个作为协调器用于接收终端节点发来的数据同时作为数据收发的中继站，并且协调器节点还连接了人体红外热释电传感器，用于检测人员出入。协调器节点与PC机通过串口相连接。另一个作为终端节点用于实现温湿度检测，并向协调器节点发送温湿度信息。

1.2 Z-Stack协议栈

ZigBee的协议栈基于IEEE802.15.4开发，其中IEEE802.15.4定义了物理层和MAC层技术规范，ZigBee联盟定义了网络层、安全层和应用层技术规范，ZigBee协议栈就是将各个层定义的协议都集合在一起，以函数的形式实现，并给用户提供一些应用层API，供用户调用。其工作流程如图1-1所示。

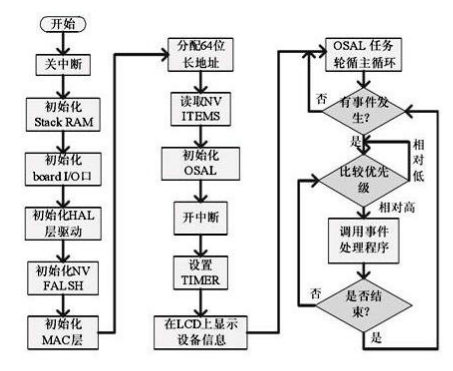


图1-1 Z-Stack软件流程图

1.3 IAR开发环境

硬件开发用的是Windows环境下的IAR8.0版本，IAR可以用于CC2530的开发，它支持多种芯片的开发，例如ARM、CC2540、CC2530、8051芯片等。在IAR里面，我们可以很方便地打开Z-Stack协议栈，对协议栈进行开发。开发界面如图2-2所示。

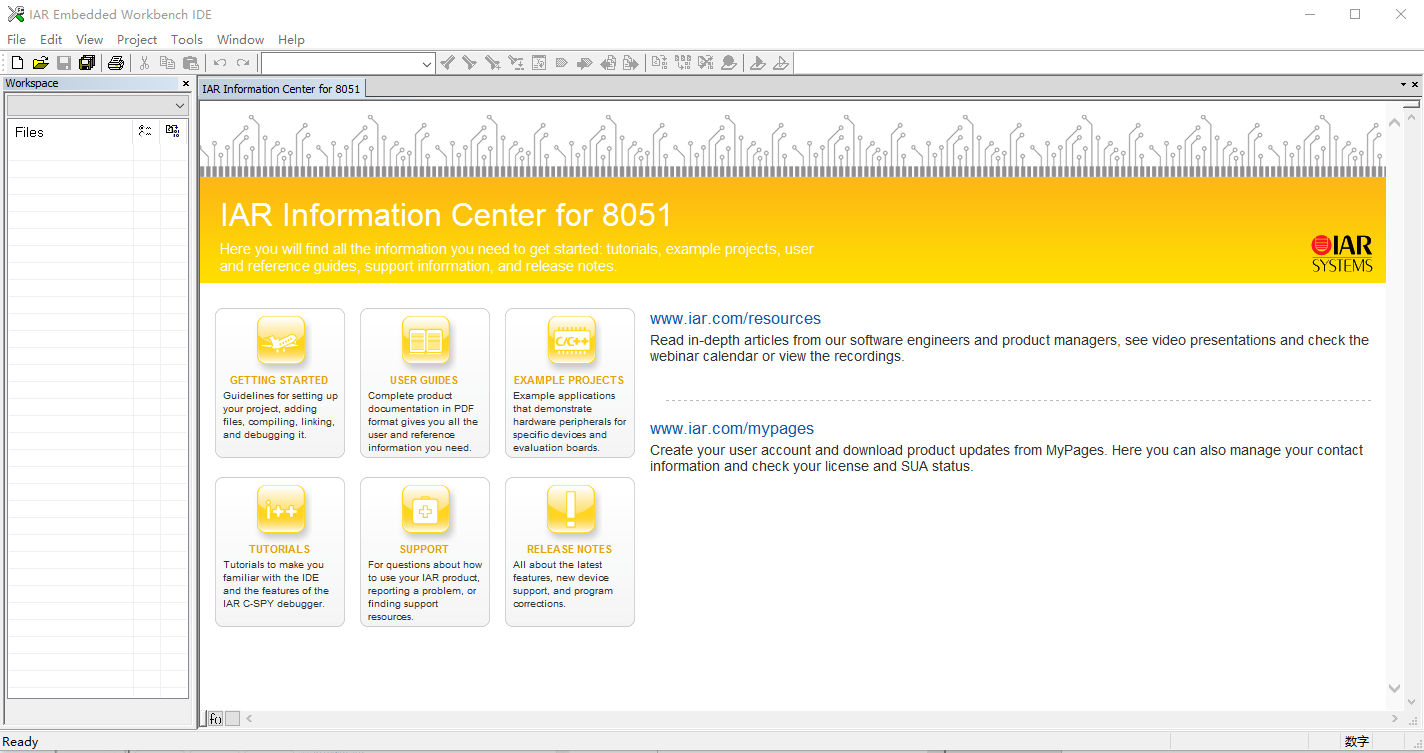


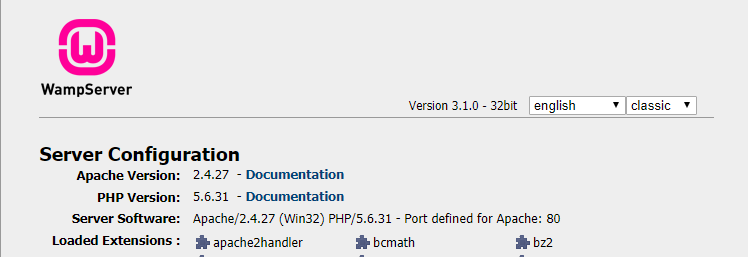
图1-2 IAR集成开发软件界面

1.4 WAMP集成开发环境

WAMP是Windows下的Apache+Mysql/MariaDB+Perl/PHP/Python，一组常用来搭建动态网站或者服务器的开源软件，本身都是各自独立的程序，但是因为常被放在一起使用，拥有了越来越高的兼容度，共同组成了一个强大的Web应用程序平台。随着开源潮流的蓬勃发展，开放源代码的LAMP已经与J2EE和.Net商业软件形成三足鼎立之势，并且该软件开发的项目在软件方面的投资成本较低，因此受到整个IT界的关注。

WAMP里的P代表PHP，本系统网页开发主要是用PHP语言进行开发的。PHP（外文名:PHP: Hypertext Preprocessor，中文名：“[超文本](https://baike.baidu.com/item/%E8%B6%85%E6%96%87%E6%9C%AC" \t "_blank)[预处理器](https://baike.baidu.com/item/%E9%A2%84%E5%A4%84%E7%90%86%E5%99%A8" \t "_blank)”）是一种通用[开源](https://baike.baidu.com/item/%E5%BC%80%E6%BA%90/246339" \t "_blank)[脚本语言](https://baike.baidu.com/item/%E8%84%9A%E6%9C%AC%E8%AF%AD%E8%A8%80/1379708" \t "_blank)。[语法](https://baike.baidu.com/item/%E8%AF%AD%E6%B3%95/2447258" \t "_blank)吸收了[C语言](https://baike.baidu.com/item/C%E8%AF%AD%E8%A8%80" \t "_blank)、[Java](https://baike.baidu.com/item/Java" \t "_blank)和[Perl](https://baike.baidu.com/item/Perl" \t "_blank)的特点，利于学习，使用[广泛](https://baike.baidu.com/item/%E5%B9%BF%E6%B3%9B/6246786" \t "_blank)，主要适用于[Web](https://baike.baidu.com/item/Web" \t "_blank)开发领域。PHP 独特的[语法](https://baike.baidu.com/item/%E8%AF%AD%E6%B3%95/2447258" \t "_blank)混合了[C](https://baike.baidu.com/item/C" \t "_blank)、[Java](https://baike.baidu.com/item/Java" \t "_blank)、[Perl](https://baike.baidu.com/item/Perl" \t "_blank)以及[PHP](https://baike.baidu.com/item/PHP" \t "_blank)自创的语法。它可以比[CGI](https://baike.baidu.com/item/CGI" \t "_blank)或者[Perl](https://baike.baidu.com/item/Perl" \t "_blank)更快速地执行[动态网页](https://baike.baidu.com/item/%E5%8A%A8%E6%80%81%E7%BD%91%E9%A1%B5/6327050" \t "_blank)。用PHP做出的[动态页面](https://baike.baidu.com/item/%E5%8A%A8%E6%80%81%E9%A1%B5%E9%9D%A2/8586386" \t "_blank)与其他的[编程语言](https://baike.baidu.com/item/%E7%BC%96%E7%A8%8B%E8%AF%AD%E8%A8%80/9845131" \t "_blank)相比，[PHP](https://baike.baidu.com/item/PHP/9337" \t "_blank)是将[程序](https://baike.baidu.com/item/%E7%A8%8B%E5%BA%8F/71525" \t "_blank)嵌入到[HTML](https://baike.baidu.com/item/HTML" \t "_blank)（[标准通用标记语言](https://baike.baidu.com/item/%E6%A0%87%E5%87%86%E9%80%9A%E7%94%A8%E6%A0%87%E8%AE%B0%E8%AF%AD%E8%A8%80/6805073" \t "_blank)下的一个应用）文档中去执行，执行效率比完全生成[HTML](https://baike.baidu.com/item/HTML" \t "_blank)标记的[CGI](https://baike.baidu.com/item/CGI/607810" \t "_blank)要高许多；PHP还可以执行[编译](https://baike.baidu.com/item/%E7%BC%96%E8%AF%91/1258343" \t "_blank)后代码，编译可以达到[加密](https://baike.baidu.com/item/%E5%8A%A0%E5%AF%86/752748" \t "_blank)和[优化](https://baike.baidu.com/item/%E4%BC%98%E5%8C%96/94618" \t "_blank)代码运行，使代码运行更快。

本系统采用PHP作为网页开发技术，WAMP作为开发环境，MySQL作为数据库，做了一个简单页面用于可视化显示数据库以及查看最新数据。WAMP的localhost页面如图1-3所示。



1-3 WAMP的localhost页面

1.5 Qt Creator

Qt是1991年有Qt公司开发的跨平台C++图形用户界面应用开发框架。可用于开发GUI程序也可用于开发非GUI程序。Qt Creator是一个用于Qt开发的轻量级跨平台集成开发环境。它为开发人员带来了两个好处，一是提供了首个专为支持跨平台开发而设计的集成开发环境（IDE），并确保首次接触Qt框架的开发人员能够迅速上手操作。

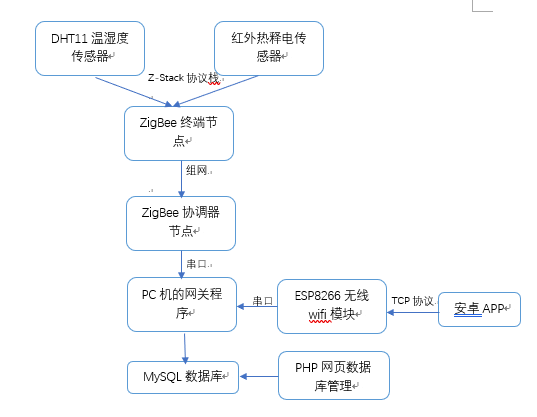
作为一个跨平台的开发环境，QT有着多种优势。

1. 优良的跨平台特性。Qt支持多种操作系统，如Windows、Linux、Solaris等系统。
2. 面向对象。Qt的良好封装机制使得 Qt 的模块化程度非常高，可重用性较好，对于用户开发来说是非常 方便的。
3. 丰富的API。Qt 包括多达 250 个以上的 C++ 类，还提供基于模板的 collections， serialization， file， I/O device， directory management， date/time 类。甚至还包括正则表达式的处理功能。
4. 支持2D/3D图形渲染，支持OpenGL

Qt的优势绝不仅仅局限于以上四个部分，它有着大量的开发文档，大量开发者的使用和支持。

1. **系统总体设计方案**
   1. 系统总体设计

本系统总体设计如图2-1所示。在本系统中，DHT11和红外热释电两个传感器与两个ZigBee设备构成一个无线传感网络，两个传感器收集数据后由终端节点通过Z-Stack协议栈与协调器节点组网，并向协调器发送传感器手机的数据，协调器再分析处理数据，发送到串口，由串口将数据传输到PC机的网关程序，同时PC机的QT网关程序对传来的数据进行分析处理，通过屏幕显示出来的同时，每收集到温度、湿度、红外热释电信息都会将其输入MySQL数据库存储。PHP编写的web页面给用户一个可视化数据库管理页面。PC机连接esp8266wifi模块，使PC机能够和手机进行通信，在手机app上能够实现对ZigBee的无线控制及数据显示。



* 1. 系统整体框图

2.2 各部分功能

2.2.1 数据采集部分

数据采集部分主要是ZigBee无线传感网络实现的。一个节点作为终端收集温湿度信息。另一个节点作为协调器节点，连接红外热释电传感器，负责检测是否有人员走动。温湿度信息通过Z-Stack协议栈组网将数据从终端节点发送到协调器。协调器通过串口连接PC机，PC机上的QT程序对收集到的数据进行处理显示。

2.2.2 网关部分

本系统中，PC机作为网关，PC上运行的QT程序作为网关程序。此程序负责接收同时来自ZigBee和ESP8266的信息。ZigBee部分接收的数据为传感器数据，当接收到传感器数据后，对数据进行分析处理，在对应部分显示对应数据，如实时温度、湿度、人员检测。同时QT程序集成了串口设置，实现对ZigBee和ESP8266串口的控制和设置。

程序同时还会采集来自ESP8266的数据。ESP8266wifi模块与手机APP在处于同一个网络环境的条件下，能够接受来自app的指令，再通过串口发送到QT程序上，QT再做出如开始检测、关闭检测、传回实时温湿度信息和人员检测信息。

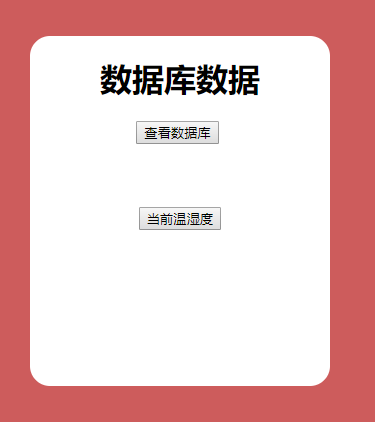
在收集到各部分数据以后，QT程序再将数据实时同步到MySQL数据库中。

2.2.3 无线控制部分

该部分包含两个模块。第一个模块是esp8266wifi模块，esp8266是一个容易上手使用的wifi模块，通过模块内部的AT指令集，连接UART串口实现对模块的控制。模块本身具有三种模式，分别是AP模式、STA模式、AP+STA模式。本系统中用的是AP模式，即使esp8266具有路由器的功能，能够发射wifi让其他终端连接，以便接收来自其他终端的数据。第二个模块为安卓APP。安卓APP通过Socket编程，用TCP协议，使得手机在连接上来自esp8266模块的wifi后能够与esp8266通过TCP协议进行连接，进而实现对ZigBee的无线控制。

2.2.4 数据库部分

本系统中采用的数据库为MySQL，MySQL是一个[关系型数据库管理系统](https://baike.baidu.com/item/%E5%85%B3%E7%B3%BB%E5%9E%8B%E6%95%B0%E6%8D%AE%E5%BA%93%E7%AE%A1%E7%90%86%E7%B3%BB%E7%BB%9F/696511" \t "_blank)，由瑞典MySQL AB公司开发，目前属[Oracle](https://baike.baidu.com/item/Oracle" \t "_blank)旗下产品。MySQL 是最流行的[关系型数据库管理系统](https://baike.baidu.com/item/%E5%85%B3%E7%B3%BB%E5%9E%8B%E6%95%B0%E6%8D%AE%E5%BA%93%E7%AE%A1%E7%90%86%E7%B3%BB%E7%BB%9F/696511" \t "_blank)之一，在WEB应用方面，MySQ是最好的[RDBMS](https://baike.baidu.com/item/RDBMS/1048260" \t "_blank)(Relational Database Management System，关系数据库管理系统)应用软件。MySql数据库能够有效提高数据操作的速度和灵活性。在QT程序中添加插入代码使数据能够实时存入数据库。另外系统中编写了WEB网页作为可视化数据库管理，如图2-2所示。



* 1. web数据库页面

1. **系统软件设计**

3.1 ZigBee数据收发部分

3.1.1 协调器回调函数

回调函数判断串口发来的数据，“start”就开始接收终端发来的数据，“stop”就停止收集，其他命令就返回命令错误。

static void rxCB(uint8 port, uint8 event)

{

unsigned char error[9]="ERROR CMD";

unsigned char stop[10]="STOP CHECK";

HalUARTRead(0,uartbuf,5);

if(osal\_memcmp(uartbuf,"start",5))

{

GenericApp\_MessageMSGCB;

}

else if(osal\_memcmp(uartbuf,"stop",4))

{

HalUARTWrite(0,stop,10);

}

else

{

HalUARTWrite(0,error,9);

}

}

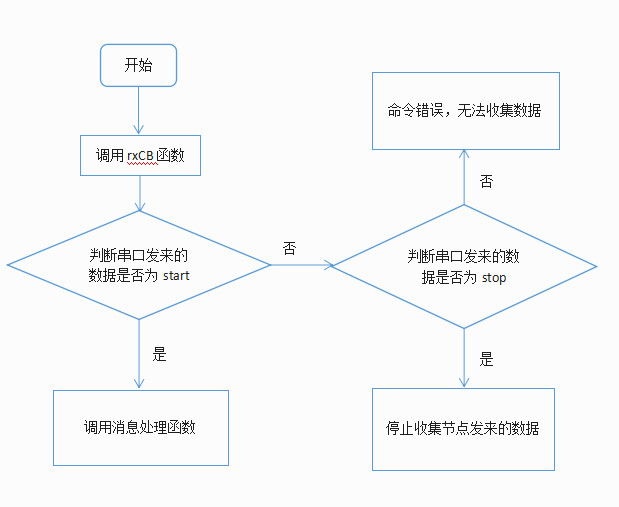


图3-1 回调函数流程图

3.1.2 协调器消息处理函数

负责处理终端发来的数据以及向串口写入数据。

static void GenericApp\_MessageMSGCB( afIncomingMSGPacket\_t \*pkt )

{

if(osal\_memcmp(uartbuf,"start",5))

{

switch ( pkt->clusterId )

{

case GENERICAPP\_CLUSTERID:

HalUARTWrite(0,"$",1);

HalUARTWrite(0,&pkt->cmd.Data[0],2);

HalUARTWrite(0,&pkt->cmd.Data[2],2);

HalUARTWrite(0,"#",1);

#if defined( LCD\_SUPPORTED )

HalLcdWriteScreen( (char\*)pkt->cmd.Data, "rcvd" );

#elif defined( WIN32 )

WPRINTSTR( pkt->cmd.Data );

#endif

break;

case GENERICAPP\_CLUSTERID\_YW:

HalUARTWrite(0,pkt->cmd.Data,pkt->cmd.DataLength);

//红外热释电传感器判断代码

byte state;

if(DATA\_PIN\_HW ==1)

{

MicroWait(10000);

if(DATA\_PIN\_HW ==1)

{

state = 0x31; //有人进入

HalUARTWrite(0,"1",1);

HalLedBlink(HAL\_LED\_1,0,50,100);

HalLedBlink(HAL\_LED\_2,0,50,200);

}

}

else

{

state = 0x30; //无人

HalUARTWrite(0,"0",1);

HalLedSet(HAL\_LED\_1,HAL\_LED\_MODE\_ON);

HalLedSet(HAL\_LED\_2,HAL\_LED\_MODE\_ON);

} break;

}}

}

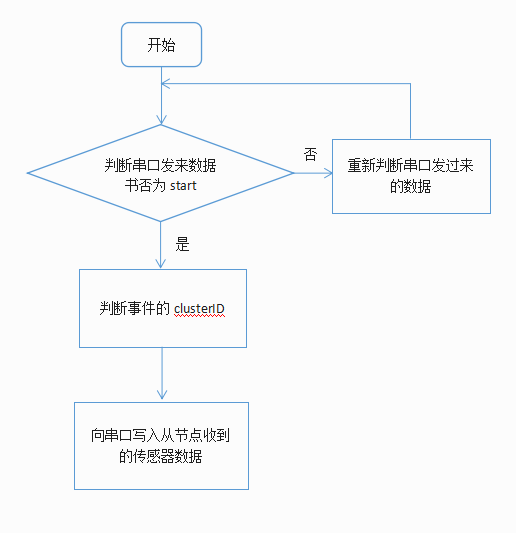


图3-2 消息处理函数流程图

3.1.3 终端消息发送函数

本函数主要是收集传感器数据并通过ZigBee无线网络向协调器发送数据。

static void GenericApp\_SendTheMessage( void )

{

DHT11();

uint8 wenshi[4];

byte state;

wenshi[0]=wendu\_shi+0x30;

wenshi[1]=wendu\_ge+0x30;

wenshi[2]=shidu\_shi+0x30;

wenshi[3]=shidu\_ge+0x30;

if(DATA\_PIN\_YW == 1)

{

state = 0x31;

}

else

{

MicroWait(10000);

if(DATA\_PIN\_YW == 0)

{

state = 0x30;

}

}

afAddrType\_t my\_DstAddr;

my\_DstAddr.addrMode=(afAddrMode\_t)Addr16Bit;//addr16bit表示单播

my\_DstAddr.endPoint = GENERICAPP\_ENDPOINT;

my\_DstAddr.addr.shortAddr = 0x0000; //表示发送对象为协调器

if ( AF\_DataRequest( &my\_DstAddr, &GenericApp\_epDesc,

GENERICAPP\_CLUSTERID,

osal\_strlen("1234")+1,

wenshi,

&GenericApp\_TransID,

AF\_DISCV\_ROUTE, AF\_DEFAULT\_RADIUS ) == afStatus\_SUCCESS )

{

HalLedBlink(HAL\_LED\_1,0,50,1300);// Successfully requested to be sent.

}

/\*else

{

// Error occurred in request to send.

}\*/

if(AF\_DataRequest(&my\_DstAddr, &GenericApp\_epDesc,

GENERICAPP\_CLUSTERID\_YW,

1,

&state,

&GenericApp\_TransID,

AF\_DISCV\_ROUTE, AF\_DEFAULT\_RADIUS ) == afStatus\_SUCCESS

)

{

HalLedBlink(HAL\_LED\_2,0,50,1300);

}

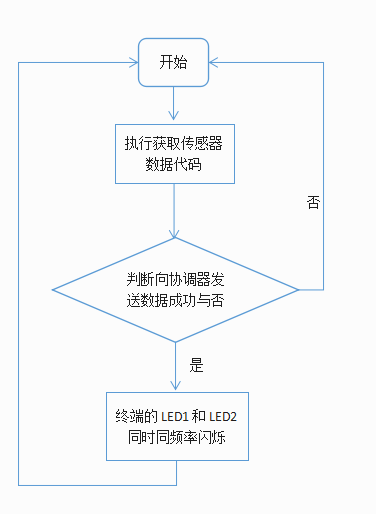


图3-3 消息发送函数流程图

3.2 传感器检测部分

3.2.1 DHT11温湿度传感器

void DHT11(void) //温湿传感启动

{

DATA\_PIN=0;//置低电平

Delay\_ms(19); //>18MS

DATA\_PIN=1; //置高电平，开始信号发送结束

P0DIR &= ~0x80; //重新配置IO口方向，输入

Delay\_10us();

Delay\_10us();

Delay\_10us();

Delay\_10us(); //高电平，延时40us等待

if(!DATA\_PIN) //如果电平变低，则证明检测到开始信号

{

ucharFLAG=2;

while((!DATA\_PIN)&&ucharFLAG++);//低电平，等待响应结束

ucharFLAG=2;

while((DATA\_PIN)&&ucharFLAG++); //高电平，准备输出

COM();//延时函数

ucharRH\_data\_H\_temp=ucharcomdata;//湿度高八位

COM();

ucharRH\_data\_L\_temp=ucharcomdata;//湿度低八位

COM();

ucharT\_data\_H\_temp=ucharcomdata;//温度高八位

COM();

ucharT\_data\_L\_temp=ucharcomdata;//温度低八位

COM();

ucharcheckdata\_temp=ucharcomdata;//校验八位

DATA\_PIN=1; //置高，空闲状态

uchartemp=(ucharT\_data\_H\_temp+ucharT\_data\_L\_temp+ucharRH\_data\_H\_temp+ucharRH\_data\_L\_temp);

if(uchartemp==ucharcheckdata\_temp)//校验

{

ucharRH\_data\_H=ucharRH\_data\_H\_temp;

ucharRH\_data\_L=ucharRH\_data\_L\_temp;

ucharT\_data\_H=ucharT\_data\_H\_temp;

ucharT\_data\_L=ucharT\_data\_L\_temp;

ucharcheckdata=ucharcheckdata\_temp;

}

wendu\_shi=ucharT\_data\_H/10;

wendu\_ge=ucharT\_data\_H%10;

shidu\_shi=ucharRH\_data\_H/10;

shidu\_ge=ucharRH\_data\_H%10;

}

else //没用成功读取，返回0

{

wendu\_shi=0;

wendu\_ge=0;

shidu\_shi=0;

shidu\_ge=0;

}

P0DIR |= 0x80; //IO口需要重新配置

}

3.2.2 红外热释电传感器

输入引脚高电平说明检测到有人，低电平无人。

if(DATA\_PIN\_HW ==1)

{

MicroWait(10000);

if(DATA\_PIN\_HW ==1)

{

state = 0x31; //有人进入

HalUARTWrite(0,"1",1);

HalLedBlink(HAL\_LED\_1,0,50,100);

HalLedBlink(HAL\_LED\_2,0,50,200);

}

}

else

{

state = 0x30;

HalUARTWrite(0,"0",1);

HalLedSet(HAL\_LED\_1,HAL\_LED\_MODE\_ON);

HalLedSet(HAL\_LED\_2,HAL\_LED\_MODE\_ON);

}

3.3 QT网关程序

网关程序中有串口设置、实时数据显示、基础控制三部分。

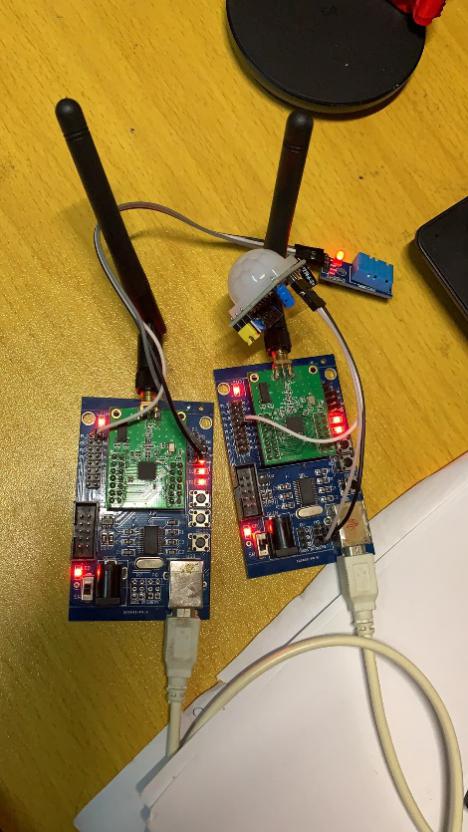
在串口设置部分，由于需要同时接收来自ZigBee和wifi模块的数据，故需要分别对两部分的串口进行设置。当接收到数据以后，QT程序立刻对数据进行分析处理。在收集、分析、处理数据后，通过可视化界面将数据显示出来。

3.4 安卓APP

本系统中安卓app能够实现对ZigBee传感网络的无线控制，主要是通过在Adroid Studio集成开发软件上开发程序实现。程序内部通过Socket编程，利用TCP协议，与ESP8266wifi模块进行连接，从而实现对数据的采集和控制。

1. **系统测试**

4.1 硬件测试



* 1. ZigBee及传感器

如图4-1所示，在两个节点组网之前，LED3等会匀速闪烁，当组网成功时，LED3会常亮，同时终端节点的LED2和LED1每发送一次数据就闪烁一次。当检测到有人时，协调器节点的LED1和LED2会交替闪烁模拟报警。

4.2 软件测试



* 1. 检测页面



* 1. 检测有人界面



* 1. APP检测页面。

图4-2和4-3是QT网关程序接收的温湿度和红外信息，图4-4是手机端APP接收到的数据以及无线控制的功能按钮。数据正确传输，符合预期结果。

1. **总结与展望**

5.1 总结

该系统已经初步实现了智能家居安防的基本功能，例如温湿度监测、入侵检测报警。

本次设计成本低廉，效率高，数据实时监测能够满足用户的需求，但由于条件有限，系统中仍存在部分地方需要完善，比如温度无法精确到小数点后的位数。

通过本次课程设计，对ZigBEE、QT、传感器等有了更深入的了解，同时为以后深入学习以及了解新技术打下良好的基础。

5.2 展望

由于本人目前时间及能力有限，不能够将该系统完善到方方面面，在部分功能方面还有很大的改进空间。例如为该系统加入真正的报警系统，像发生险情时，能够通过WiFi网络及时告知屋主或者自动联网报警；又或者添加新的传感器功能，例如声音、光照等的检测，为家居安全做到最大程度的保障。

参考文献

1. 王小强，欧阳骏，黄宁淋 《ZigBee无线传感器网络设计与实现》[M].北京：化学工业出版社.
2. 李正明，张兴伟，基于ZigBee技术的无线智能安防系统的设计[J],计算机与现代化，2012,9.
3. 孙波，基于ZigBee的智能安防系统的研究[J],通讯与网络,2013,7(4)。