

# Design and Research of Environmental Monitoring System Based on ZigBee

Ren Gao

College of Information Engineering, Hubei University of Economics, Wuhan Hubei  
Email: gr@hbue.edu.cn

Received: Sep. 2<sup>nd</sup>, 2016; accepted: Sep. 17<sup>th</sup>, 2016; published: Sep. 22<sup>nd</sup>, 2016

Copyright © 2016 by author and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

---

## Abstract

With rapid development of the society, people pay more attention to the quality of lives, especially the environmental conditions: the weather phenomenon as the haze and mist and PM2.5. As the hottest wireless communication technology, ZigBee-based environmental monitoring system is designed and discussed in this paper. From the perspectives of technical research and practical application, it elaborates the design of the environmental monitoring on the cost-effective Chip CC2530, which implements a variety of functions including data collection, data transmission through ZigBee, data processing through Web Server and data monitoring through intelligent terminal.

## Keywords

ZigBee, Z-Stack, Environmental Monitoring, Internet of Things, Sensor

---

# 基于ZigBee的环境监测系统的设计与实现

高 刃

湖北经济学院信息工程学院, 湖北 武汉  
Email: gr@hbue.edu.cn

收稿日期: 2016年9月2日; 录用日期: 2016年9月17日; 发布日期: 2016年9月22日

---

## 摘 要

随着社会不断的发展, 人们对生活质量要求越来越高, 特别对人们生存环境的关注, 如当前最关注的雾

霾现象和PM2.5。这些加剧人们对居住环境关注。本文结合当今最炙手可热的无线通信技术ZigBee，实现了基于ZigBee技术的环境监测系统。该系统从实际应用的角度出发，使用当前高性价比的CC2530芯片，设计出整套环境监测方案，该方案使用各种传感器采集数据，使用ZigBee节点传输数据，最后通过Web服务器对数据继续处理，并能通过智能终端查看和控制。

## 关键词

ZigBee, Z-Stack, 环境监测, 物联网, 传感器

## 1. 引言

当前，环境问题是一个全球都关注的热门话题，和人们的生活息息相关。无线传感网络是大量静止或者移动的传感器以某种方式构成的无线网络，把其网络监测的区域各种有用的环境信息通过采集、传输和处理，最后对这些信息进行分析应用。本文基于 ZigBee 技术[1]，设计出的无线传感网络能够部署在监测区域，通过无线通信的方式构成了一个多跳自组织网络，监测周围的环境信息，并将信息实时传递到终端。

## 2. 基于 ZigBee 的无线通信技术

### 2.1. ZigBee 无线通信协议

ZigBee 是符合 IEEE 802.15.4 规定的短距离无线通信协议，它的通信参考模型如图 1 所示。ZigBee 协议的底层是遵循 IEEE 802.15.4 标准的物理层和 MAC 组成，ZigBee 联盟[2]在此基础上又增加了网络层和应用层。

(1) 应用层：ZigBee 协议的应用层可以为在同一网络的不同设备之间提供数据传输服务，发现设备并绑定设备。(2) 网络层：ZigBee 协议中，网络层主要负责网络管理、路由管理、报文以及网络安全管理等任务。(3) MAC 层：MAC 层主要负责对物理的信道的访问。(4) 物理层：物理层主要负责控制无线收发器的激活和关闭，选择信道的频率，发送和接受数据等等[3]。

### 2.2. ZigBee 拓扑结构

ZigBee 无线网络结构可以分为星形、树形和网状[4]。其中全能设备(FFD)在三种网络中承担协调器的作用，管理网络中的其它网络节点。简化设备(RFD)则只是作为普通网络节点，在网络中与 RFD 或 FFD 间进行通信。

(1) 星形拓扑，星形拓扑网络结构的中心节点必须由协调器构成，其它节点则可以说终端设备或路由器，协调器作为整个网络的核心。(2) 树形拓扑，典型的 ZigBee 的树形拓扑网络由协调器、路由器和终端设备组成，不需要每个每一个网络节点都和协调器相连，可通过节点与协调器之间的路由器的路由功能进行通信。以路由器的方式扩大这个网络的覆盖范围。(3) 网状拓扑，网状拓扑较为复杂，但同时设计起来，灵活性高，在整个网络中，路由器作为一个很重要的节点，可以根据需要自动创建并维护网络路径，可以和任何节点之间镜像通信，帮助附近的节点传递数据。

### 2.3. Z-Stack 协议栈

Z-Stack 协议栈的实现方式是采用分层的思想，分为物理层、介质访问控制层、网络层和应用层。本设计采用 TI 公司推出的 Z-Stack 协议栈，如图 2 是用 IAR 软件打开的 Z-Stack 协议栈的目录。



Figure 1. ZigBee reference model for communications  
图 1. ZigBee 的通信参考模型

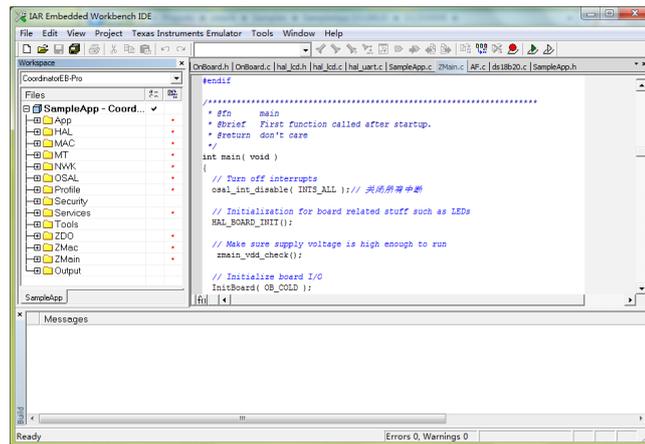


Figure 2. Z-Stack protocol stack project  
图 2. Z-Stack 协议栈工程

(1) App: 应用层目录; (2) HAL: 硬件层目录; (3) MAC: MAC 层目录; (4) MT: 监视测试目录; (5) NWK: 网络层目录; (6) OSAL: 协议栈的操作系统抽象层; (7) Profile: AF 层的目录; (8) Security: 安全层的目录; (9) Services: 地址处理函数目录; (10) Tools: 工程工具的配置目录; (11) ZDO: ZDO 目录; (12) ZMac: ZMac 目录; (13) ZMain: 整个协议栈的主函数目录; (14) Output: 输出文件的目录。在 Z-Stack 协议栈中使用了操作系统的概念, 即框架目录中的 OSAL 层。OSAL 采用的是时间片轮转的方法去实现多任务的切换, 以此实时处理各种任务。

Z-Stack 是基于操作系统设计的, 以事件机制加时间片轮询的进行工作构建的。Z-Stack 协议栈以 ZMain 目录中 main 函数入手, 可以看出整个 main 函数首先进行了相关硬件的初始化已经 APP 应用的初始化, 在初始化完成后, 就进行时间片轮询, 等待事件发生, 然后调用相应的处理函数, 完成相应的操作。

整个 Z-Stack 协议栈工作流程如图 3。首先是进行各项初始化, 包括初始化系统时钟、检查芯片的工作电压、初始化并配置相应定时器等; 完成系统初始化后就开始进入操作系统的循环等待事件发生。Z-Stack 是基于 OS 进行运作的, 每个事件都有相应的初始化函数和处理事件函数, 如开发者会用到的 GenericApp\_Init(taskID), 其中参数 taskID 是 Z-Stack 根据其优先级分配的 ID 号, 对应的数值越小, 则优先级越高。

### 3. 环境监测系统的系统设计

整个系统分为四大部分, 分别是终端部分、Web 服务器部分、无线通信部分和传感器部分。整个系统首先通过传感器部分对传感器进行设置并对数据进行采集; 接着在无线通信部分就可以将数据按照一

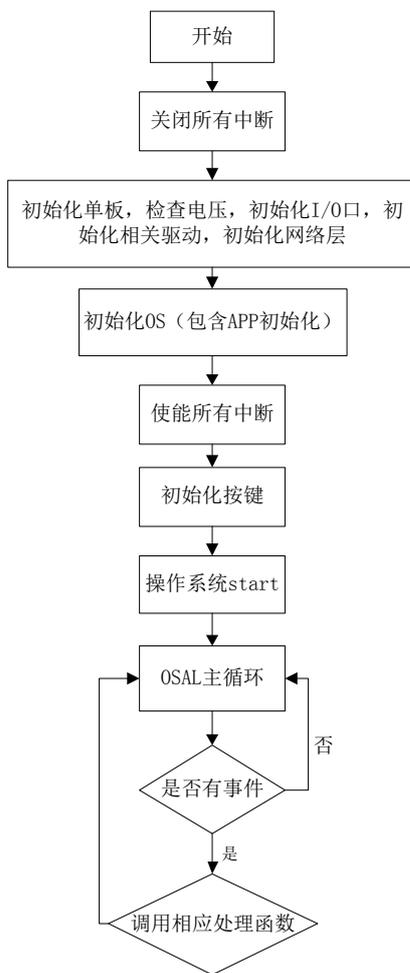


Figure 3. Z-Stack system flow chart

图 3. Z-Stack 系统工作流程图

定网络拓扑路径，从终端节点出发，最终传送到协调器节点，协调器在接到数据后，通过串口将数据上传到 PC 机；在 Web 服务器这部分，调用 Linux 系统下读写串口函数，对协调器上传的数据进行读取，并调用文件读写函数处理数据，存放在本地数据库；在终端这部分，可以对数据进行管理、查询并做出相应的环境预报，同时可以方向发生控制信号对传感器部分进行远程控制[5]。图 4 是系统的整体方案。

硬件模块采用 TI 公司提供的 CC2530。CC2530 是用于 2.4 GHz 的 IEEE 802.15.4 ZigBee 等技术的 SoC 解决方案。在建立大范围的网络节点时，为其解决成本。在本设计中，是增强型 CC2530 单片机为核心，组建具备数据采集、数据通信功能的无线传感网络[6]。

### 3.1. 采集节点

采集节点硬件模块主要由 CC2530 和各类传感器(如温度传感器、湿度传感器、烟雾传感器和气压传感器等等)以及电源模块组成，如图 5 所示。采集节点在网络结构中定位为终端节点，长时间工作在野外环境，具有超低功耗的运行模式，一般采用普通的 5 号电池就可以在野外工作 6~24 个月。

### 3.2. 路由器节点和协调器节点

协调器和路由器模块的硬件基本相同，均是采用 CC2530 作为整个电路的核心。需要区别的是，协

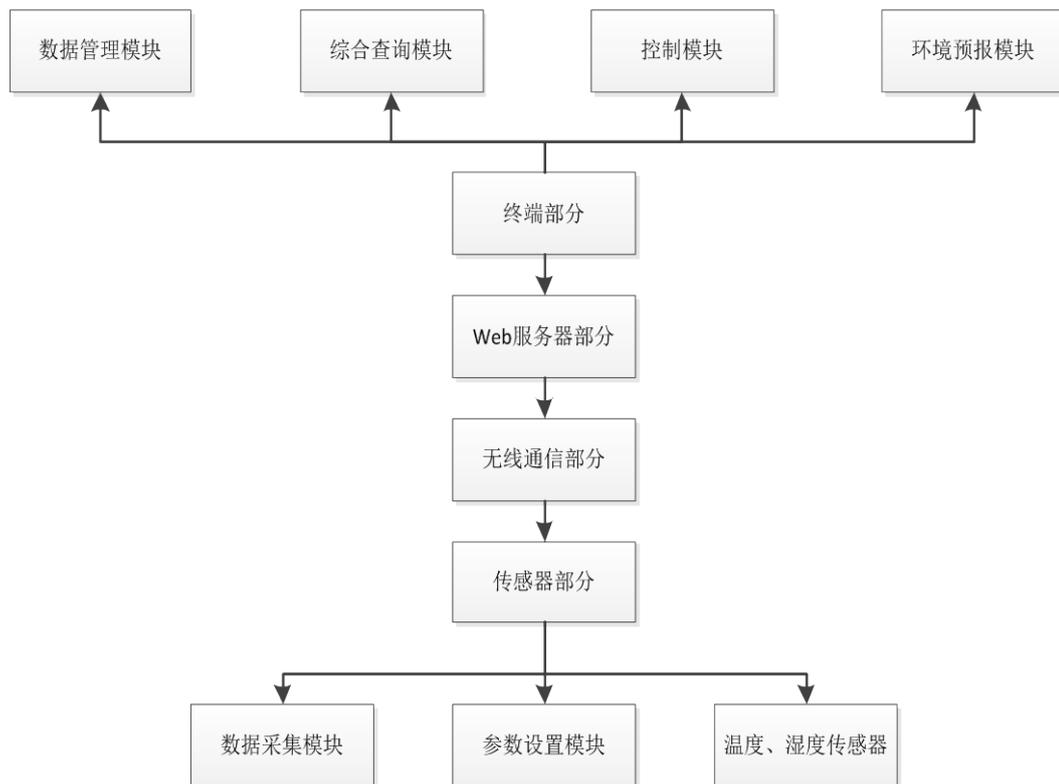


Figure 4. System scheme  
图 4. 系统整体方案

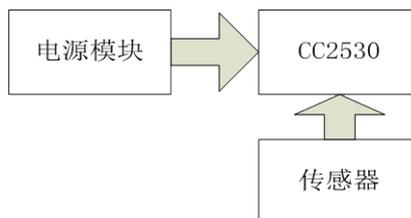


Figure 5. Collection nodes  
图 5. 采集节点

调器会通过串口将接收到的数据发送给 PC 机，PC 机则通过串口读写软件显示收到的数据。如图 6 为路由器节点和协调器节点。

## 4. 环境监测系统的软件设计

### 4.1. 软件模块

整个软件部分由底层协议 Z-Stack 协议栈、Linux 系统下 Web 服务器和手持终端组成，手持终端上运行 Android 系统[7]。

整个系统从终端节点开始，由采集几点上传感器采集数据，通过 Z-Stack 协议把采集到数据传输到协调器，协调器则通过串口向 PC 机上传采集数据。在 Web 服务器上，Linux 系统通过读串口数据函数把数据存储在本机，等待用户访问和控制。在 Android 系统部分，手持终端通过用户模式，登录 Linux 系统下的 Web 服务器，获取数据，得到需要采集环境的环境数据。

在使用 Z-Stack 协议栈 API 之前，需要进行一些简单的初始化，如图 7 所示。打开 App 目录下 SampleApp.c 文件，找到 SampleApp\_Init 函数，在函数开头加入串口初始化和需要外借温度传感器 I/O 口初始化。

在完成初始化后，在 SampleApp.c 源文件中找到 SampleApp\_ProcessEvent 函数，加入相应的内容，代码如下。系统通过周期性判断是否有事件发生并处理。当发生事件时，则对时间发生的情况进行解析。当事件为 SYS\_EVENT\_MSG 时，可判断事件具体情况为 CMD\_SERIAL\_MSG、KEY\_CHANGE、AF\_INCOMING\_MSG\_CMD、ZDO\_STATE\_CHANGE 中的哪一种，并调用相应的函数进行处理。当事件为 SAMPLEAPP\_SEND\_PERIODIC\_MSG\_EVT 时，在这里加入对原始数据的处理。

### 4.2. Web 服务器

整个服务器部分通过双进程的原理分别进行操作，其中一个进程构建 socket 网络，并初始化网络各项参数，等待用户访问，另一个进程读取串口发来的数据，并通过文件操作函数，将数据通过文件的形式存放在本地。

在 server.c 中，由 server\_start、init\_sock 和 do\_process 函数组成。server\_start 函数在函数最开始调用

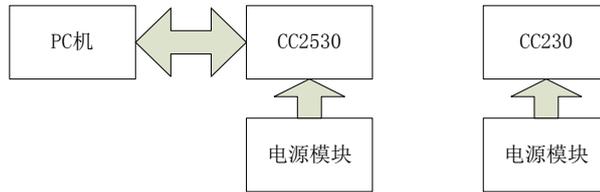


Figure 6. The coordinator nodes and routing nodes  
图 6. 协调器节点和路由器节点

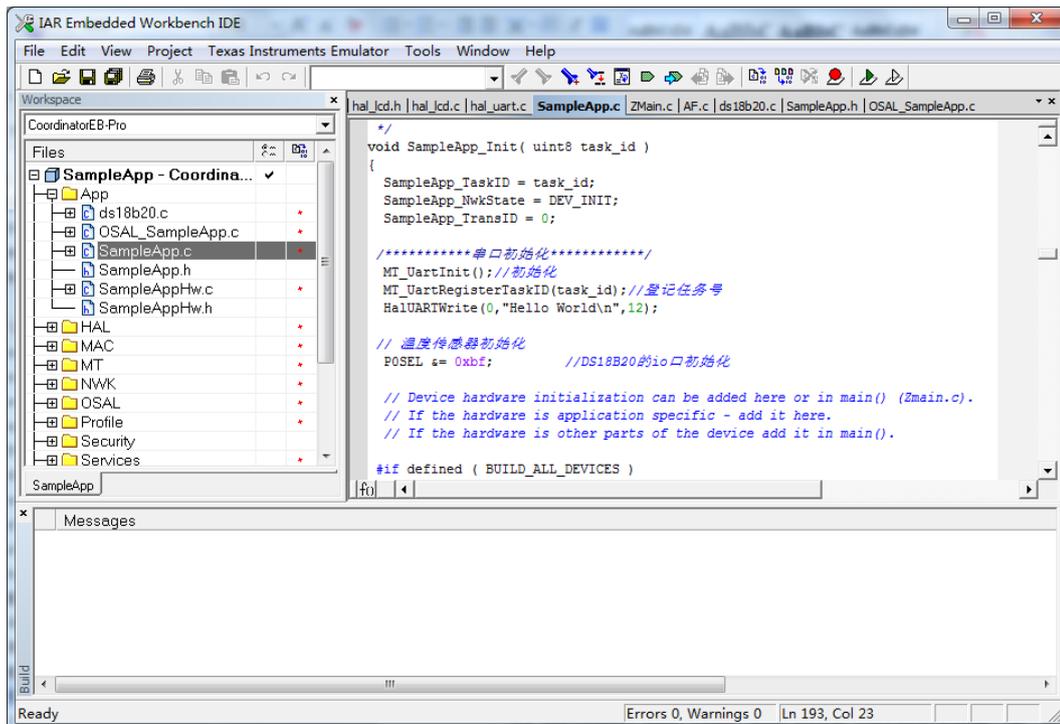


Figure 7. Serial port initialization  
图 7. 串口初始化

```

1 000 2013-11-04 12:42:40 H0796F 90 C
2 0 2013-11-04 12:42:40 2013-11-04 12:42:40 H0796F 91 C
3 0 2013-11-04 12:42:40 2013-11-04 12:42:50 H0796F 91 C
4 0 2013-11-04 12:42:50 2013-11-04 12:42:50 H0796F 90 C
5 00 H0796F 2013-11-04 12:42:53.7 V
6 0 2013-11-04 12:42:50 2013-11-04 12:43:00 H0796F 90 C
7 0 2013-11-04 12:43:00 2013-11-04 12:43:00 H0796F 91 C
8 0 2013-11-04 12:43:00 2013-11-04 12:43:00 H0796F 91 C
9 0 2013-11-04 12:43:10 2013-11-04 12:43:10 H0796F 89 C
10 0 2013-11-04 12:43:10 2013-11-04 12:43:10 H0796F 3.7 V
11 0 2013-11-04 12:43:10 2013-11-04 12:43:10 H0796F 91 C
12 0 2013-11-04 12:43:20 2013-11-04 12:43:20 H0796F 89 C
13 0 2013-11-04 12:43:20 2013-11-04 12:43:20 H0796F 91 C
14 0 2013-11-04 12:43:30 2013-11-04 12:43:30 H0796F 91 C
15 0 2013-11-04 12:43:30 2013-11-04 12:43:30 H0796F 3.7 V
16 0 2013-11-04 12:43:30 2013-11-04 12:43:30 H0796F 90 C
17 0 2013-11-04 12:43:40 2013-11-04 12:43:40 H0796F 91 C
18 0 2013-11-04 12:43:40 2013-11-04 12:43:40 H0796F 89 C
19 0 2013-11-04 12:43:50 2013-11-04 12:43:50 H0796F 90 C
20 0 2013-11-04 12:43:50 2013-11-04 12:43:50 H0796F 3.7 V
21 0 2013-11-04 12:43:50 2013-11-04 12:43:50 H0796F 90 C

```

Figure 8. Web server is receiving data

图 8. Web 服务器接受数据

init\_sock 函数，接下来循环等待用户访问。init\_sock 函数通过 socket 网络编程中，socket、bind 和 listen 三个核心函数，完成 Web 服务器的初始化，最好返回 sock\_fd 的值。do\_process 函数当有用户访问时，通过 getpeername 函数获取用户 IP 等一些信息，并对用户请求做出相应的处理。

### 4.3. Android 终端

在整个系统中，Android 终端属于用户操作，通过访问 Web 服务器获取监测环境的一些数据，并在手持终端上显示出来。手持终端通过类似网址访问模式。

## 5. 系统调试

在 Linux 系统下 eclipse 编程工具获取的协调器通过串口上传的数据，并调用 Linux 系统时间函数获取本地时间得到原始数据。如图 8 所示。

## 6. 总结

本系统是在 CC2530 单片机和 Z-Stack 协议栈的基础上，实现了基于 ZigBee 环境监测系统。(1) 主要借助 IAR 软件开发环境对 Z-Stack 协议栈的关键 API 函数进行剖析，并在相应位置加入需应用的代码。(2) 基于 ZigBee 无线通信技术[8]-[12]，实现了环境监测系统的 Web 服务器。(3) 实现 Android 客户端对环境监测系统控制。

## 基金项目

本文由湖北省教育厅科研项目 B2013032 支持。

## 参考文献 (References)

- [1] 王小强, 欧阳骏, 黄宁琳. ZigBee 无线传感网络设计与实现[M]. 北京: 化学工业出版社, 2012.
- [2] 高守玮, 吴灿阳, 杨超, 等. ZigBee 技术实战教程——基于 CC2430/31 的无线传感网络解决方案[M]. 北京: 北京航空航天大学出版社, 2009: 3-10.
- [3] 吕志安. ZigBee 原理与应用开放[M]. 北京: 北京航空航天大学出版社, 2007: 3-19.
- [4] 蒋挺, 赵成林. 紫峰技术及其应用[M]. 北京: 北京邮电大学出版社, 2006.
- [5] 周枫. 基于 ZigBee 协议的无线温度传感网络的设计与实现[D]: [硕士学位论文]. 南京: 南京理工大学, 2007.
- [6] 王江峰. 基于 ZigBee 无线传感网络的实现[D]: [硕士学位论文]. 济南: 济南大学, 2010.
- [7] 孙利民. 无线传感网络[M]. 北京: 清华大学出版社, 2004.
- [8] 李晓维, 徐勇军. 无线传感网络技术[M]. 北京: 北京理工大学出版社, 2007.
- [9] 宋文. 无线传感网络技术与应用[M]. 北京: 电子工业出版社, 2009.
- [10] 杨丰盛. Android 应用开发揭秘[M]. 北京: 北京机械工业出版社, 2010.
- [11] 余志龙, 陈昱勋, 郑名杰. Android SDK 开发范例大全[M]. 北京: 北京人民邮电出版社, 2010.
- [12] 张运芳. Android 路线与产品开发实战[M]. 北京: 电子工业出版社, 2010.

### 期刊投稿者将享受如下服务:

1. 投稿前咨询服务 (QQ、微信、邮箱皆可)
2. 为您匹配最合适的期刊
3. 24 小时以内解答您的所有疑问
4. 友好的在线投稿界面
5. 专业的同行评审
6. 知网检索
7. 全网络覆盖式推广您的研究

投稿请点击: <http://www.hanspub.org/Submission.aspx>

期刊邮箱: [csa@hanspub.org](mailto:csa@hanspub.org)