

# 毕业设计中期检查报告

班级：14 机电二班

姓名：郑江湖

学号：2014330300129

导师：金玉珍

## 一：毕业设计总括

### 1.1、论文题目

论文题目：“基于 Control4 主机与 Zigbee 外扩 AP 的智能家庭安防系统”。

### 1.2、论文内容

在我们的日常生活中，安全是至关重要的，离开了安全，那么生活就会毫无幸福、快乐而言，它关系到每个人的生活及财务是否得以保障。而家庭室内作为我们生活中度过大多数时光的区域，它的安全防护工作就显得尤其重要，稍有不慎就会造成一定的财产损失甚至是生命危险。以室内有害气体浓度为例，在我们不进行监督的情况下，如果任由其积累升高，渐渐地危害我们的呼吸健康，迟早会造成对身体的危害。除此之外还有其他的一系列平时会被我们忽略的安防问题，比如室内易燃气体检测、水浸情况检测、门磁安全监督、人体感应等。而传统的家庭安防基本都存在很大问题，要么是传感器不够智能，又或是户主无法实时获知监控数据，并且各传感器零散分布，管理困难，使得用户的使用体验极差，并不能达到很好的智能家庭安防效果。与此同时，局域网通信技术蓬勃发展，WiFi 和其他无线通信技术，比如 Zigbee 等都得到了广泛的应用，并且无论是硬件还是软件，其研发成本都大大降低，使得一系列智能家居产品开始走进万千普通用户的家中。因此利用无线局域网通信技术将诸多的安防传感器连接起来，搭配智能家居主机和智能网关组成一个智能家庭安防系统，便可以很好的解决上述中的家庭安防问题，实现对各种安全问题与潜在隐

患的实时监控，将可能发生或已经存在的危险扼杀在摇篮里，从而保障日常家庭生活的安全与财产保障。

本文首先调查和分析了目前国内外在智能安防家居领域的发展情况以及所依赖技术的发展现状，然后根据最需要解决的模块，设计整合出该安防系统。具体研究内容如下：

首先分析了市场需求现状和相关局域网无线通信技术发展的成熟度，决定采用 WiFi 主机搭配智能网关通过 Zigbee 方式来监控各安防传感器的设计思路，将室内的易燃气体浓度、有害烟雾浓度、是否水浸、门磁安全、人体感应等安防模块的检测结果进行实时采集与监督，从而实现一整套家庭安防问题的整合。根据以上安防系统的设计思路，先研究并做出了智能网关和一个传感器的硬件样品，将 Control4 主机、网关、传感器进行一个最基本系统的测试，在此基本系统良好运行的基础上，再着手开发其他剩余安防传感器模块，并逐一整合到系统上，最后实现完整系统的性能测试与运行演示。

1.3、设计中所用到的设备与软件

硬件电路开发调试：Allegro Cadence 系列 PCB 画图软件、IAR、CAM350

软件开发：Eclipse、Linux（Ubuntu）、Jlink/espFlashDownload 程序烧写软件、串口调试助手、socktools

测试设备：万用表、示波器、测试底板

二：论文进度安排

起止时间	内容
2017.11.20-2017.11.23	毕业设计前期资料准备、毕业设计任务书、外文翻译任务布置
2017.11.24-2017. 12.17	文献阅读、外文翻译、综述报告、开题报告
2017.12.18-2018.01.20	硬件选型
2018.03.05-2018.03.28	硬件电路设计
2018.03.29-2018.04.09	软件设计
2018.04.10-2018.04.27	实验验证

2018.04.28-2018.05.09	论文撰写
2018.05.12-2018.05.23	论文评阅
2018.05.26-2018.05.30	论文答辩

这是原先制定的论文计划，但是在硬件电路模块设计开发过程中，由于通过请教其他人员的原因，使得硬件模块的进度提前了，在接下来的软件设计模块，通过实习单位现有资料的参考之下，同样很快就完成了设计与开发，从而将整个毕业设计的进度加快了，更正后的进度安排如下：

2018.03.05-2018.03.20	硬件电路设计
2018.03.21-2018.04.01	软件设计
2018.04.2-2018.04.16	实验验证
2018.04.17-2018.04.30	论文撰写
2018.05.01-2018.05.10	论文评阅
2018.05.11-2018.05.16	论文答辩

### 三：论文完成情况

#### 3.1、硬件模块

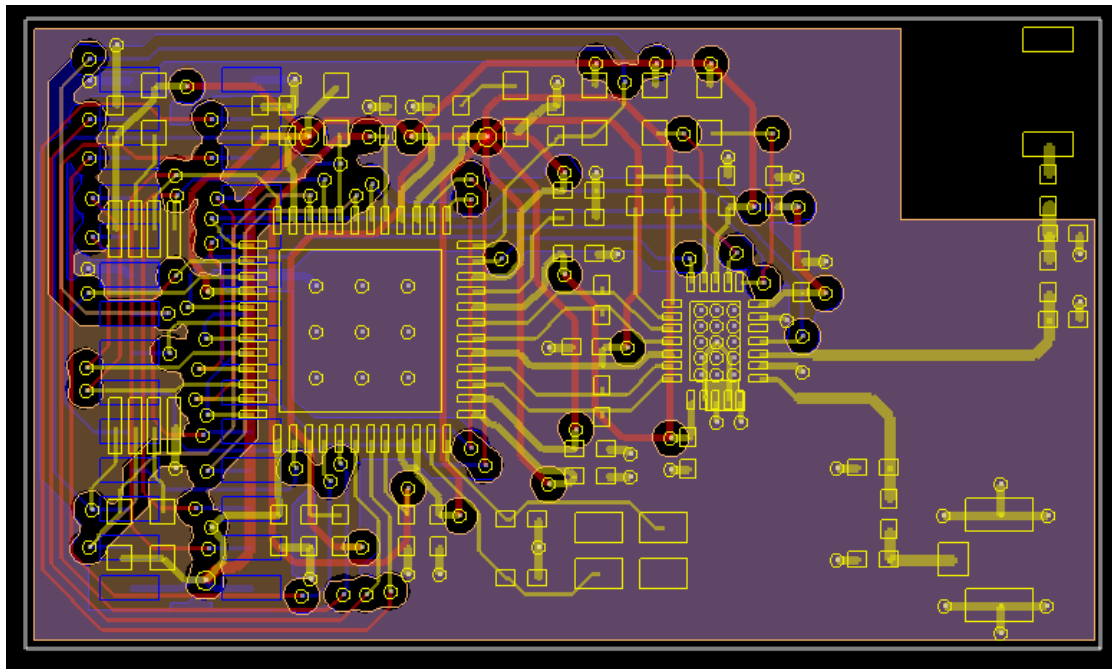
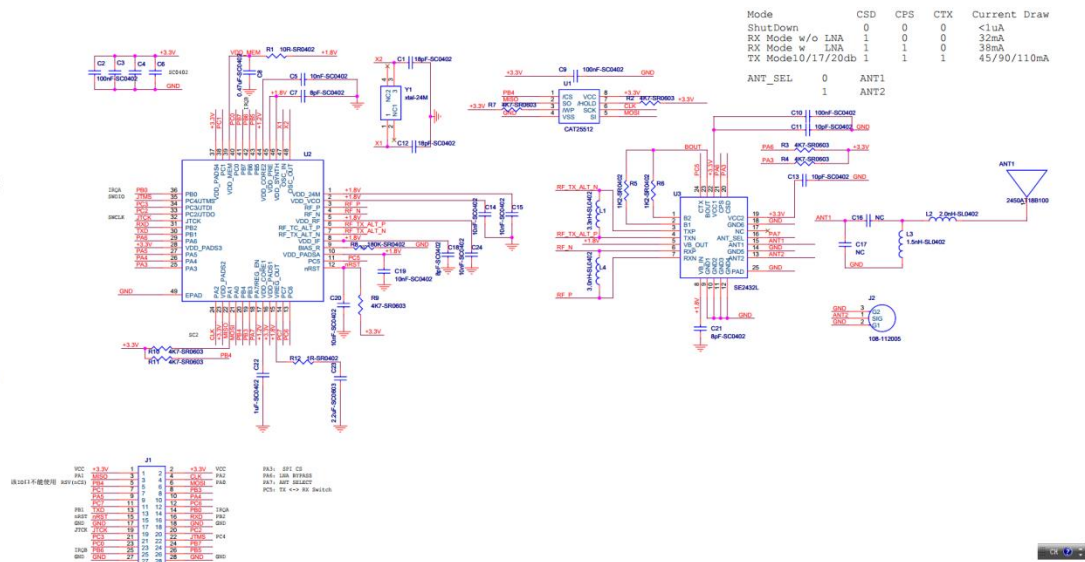
在该设计中，需要设计与开发的硬件有 ZAP 智能网关、AQS 空气质量检测仪、EM357 的 Zigbee 模块，其他的传感器部分，由于是实习单位直接进行购买所得，因此并不需要太大的修改与调整，但是在进行软件修改的时候需要进行各种电路焊接方便进行调试。

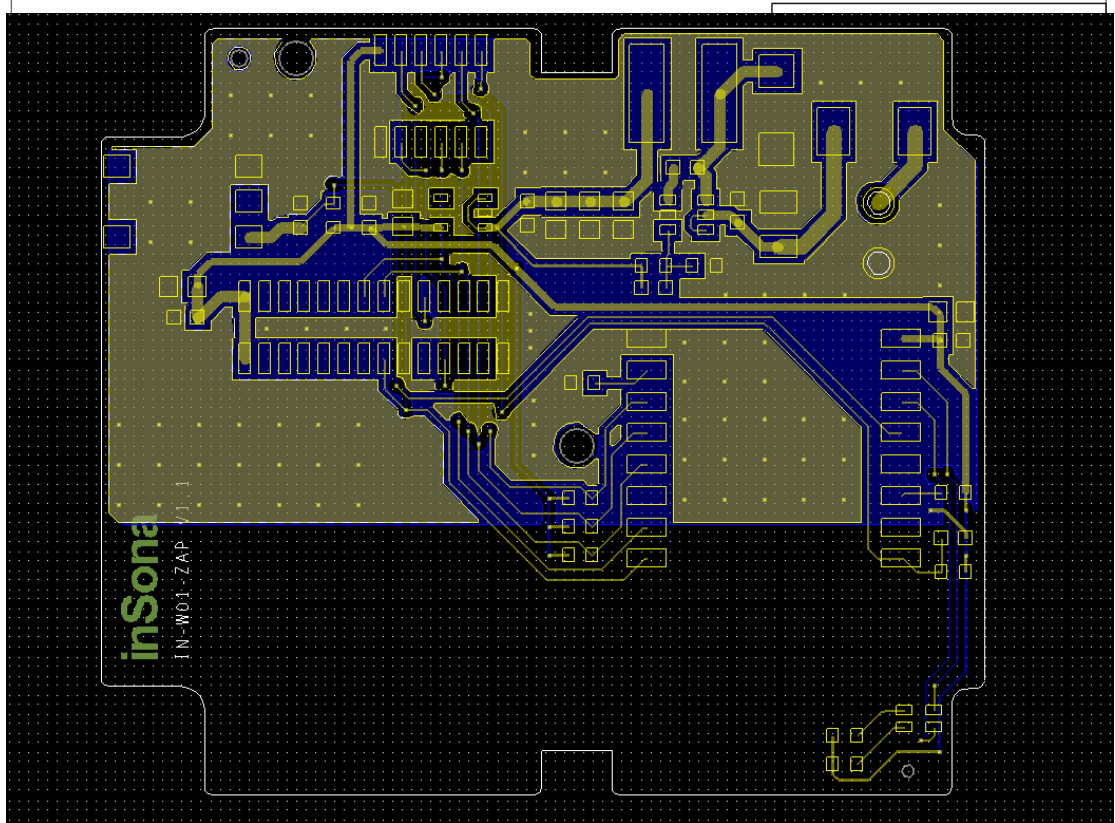
目前智能网关 ZAP 已经完成了原理图绘制、PCBA 线、打样、手动焊接、调试这几个步骤，在硬件方面基本没有太大问题，可以投入软件模块的程序烧写与调试了，展示如下：

原理图与 PCB 布线：

2

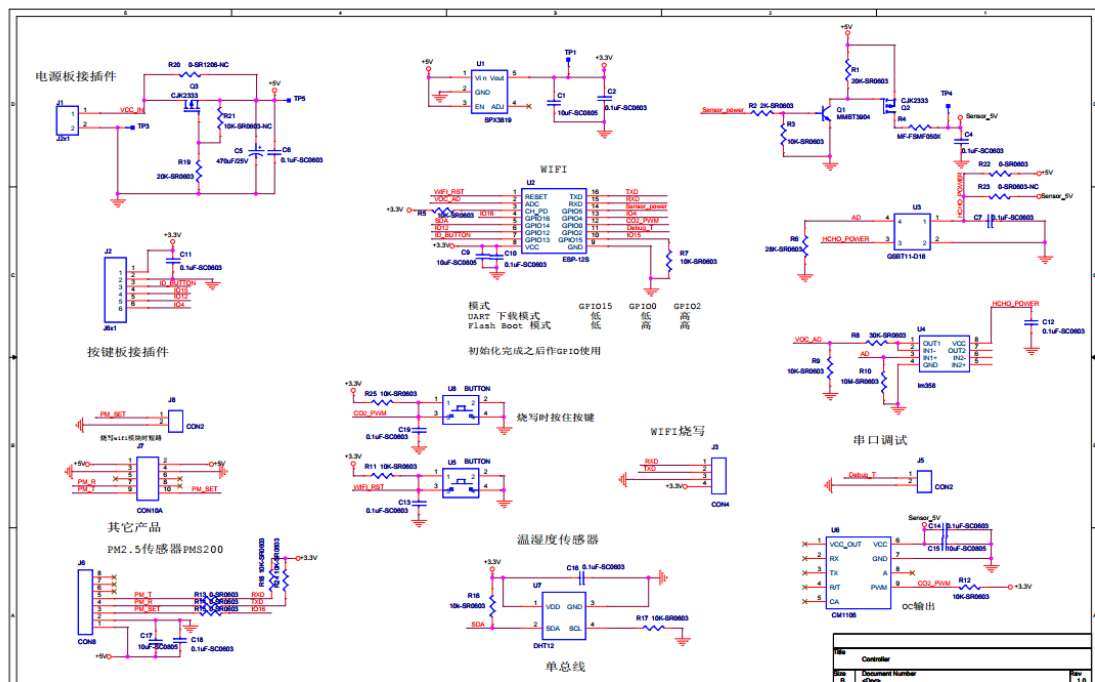
3







AQS：



产品图：



### 3.2、软件模块

软件模块分为 WIFI 模块、Zigbee 模块、主机的 Composer 模块、各传感器模块。其中：

WiFi 模块已经设计验证完毕，可以通过 socket tools 等网络调试工具和主机以及 ZAP 之间进行通信测试，部分代码如下：

```
23
24 /*****
25  * FunctionName : user_init
26  * Description  : entry of user application, init user function here
27  * Parameters   : none
28  * Returns      : none
29  *****/
30 void user_init(void)
31 {
32     static const char ssid[] = "inSona";
33     static const char password[] = "Control4";
34     struct station_config stationConf;
35
36
37     uart1_init();
38     Logger_Init(&myLoggerConfig);
39     Logger_SetGlobalLevel(LOG_DEBUG);
40     LogInfo("***** user_init %d %d*****\n", os_random(), os_random());
41     LogInfo("SDK version:%s\n", system_get_sdk_version());
42     /* need to set opmode before you set config */
43     wifi_set_opmode(STATION_MODE);
44
45     stationConf.bssid_set = 0;
46     strcpy(stationConf.ssid, ssid);
47     strcpy(stationConf.password, password);
48     wifi_station_set_config(&stationConf);
49
50     xTaskCreate(WifiTask, "WifiTask", 384, NULL, 5, NULL);
51 }
52
53
```

```

193 xTaskHandle jsonrpc_task_handle;
194
195 void ICACHE_FLASH_ATTR WifiTask(void * para)
196 {
197     u32 jsonrpc_uxHighWaterMark = 0;
198
199     //mytestsoso();
200
201     jsonrpc_uxHighWaterMark = uxTaskGetStackHighWaterMark(jsonrpc_task_handle);
202     LogDebug("%s \n", __FUNCTION__);
203     /* need to set opmode before you set config */
204     WaitConnected();
205     PrintInfo();
206
207     //xTaskCreate(tcptest_task, "tcptest_task", 512, NULL, 3, &jsonrpc_task_handle);
208
209     xTaskCreate(jsonrpc_task, "jsonrpc_task", 512, NULL, 3, &jsonrpc_task_handle);
210     //xTaskCreate(jsonrpc_task, "jsonrpc_task", 512, NULL, 4, NULL);
211     //xTaskCreate(zap_task, "zap_task", 512, NULL, 4, NULL);
212
213     while(1)
214     {
215         LogDebug("%s jsonrpc_uxHighWaterMark = %d\n", __FUNCTION__, jsonrpc_uxHighWaterMark);
216         vTaskDelay(500);
217         jsonrpc_uxHighWaterMark = uxTaskGetStackHighWaterMark(jsonrpc_task_handle);
218     }
219     vTaskDelete(NULL);
220 }
221
222
223
224 /*****

```

Zigbee 模块也已经基本设计完毕，可以进行传感器控制，能完成基本的 Zigbee 功能，可以嵌入到各传感器中去，部分代码如下：

```

7 void ICACHE_FLASH_ATTR
8 jsonrpc_task(void * para)
9 {
10     uint32_t nowTime, lastTime, mtorrIntervalTime;
11     jsonrpc_error_t error = JSONRPC_ERROR_OK;
12
13     LogDebug("create birpc server used port %d\n", 8212);
14     self = jsonrpc_server_open(0, 8212);
15     //LogDebug("create birpc server used port %d\n", 8219);
16     //self = jsonrpc_server_open(0, 8219);
17
18     error = jsonrpc_server_register_method(self, JSONRPC_TRUE, server_zapFormNetwork, "zapFormNetwork", "netType:s,panId:i,radio");
19     error = jsonrpc_server_register_method(self, JSONRPC_TRUE, server_zapLeaveNetwork, "zapLeaveNetwork", "netType:s");
20     error = jsonrpc_server_register_method(self, JSONRPC_TRUE, server_zapPermitJion, "zapPermitJion", "seconds:i");
21     error = jsonrpc_server_register_method(self, JSONRPC_TRUE, server_zapRemoveDevice, "zapRemoveDevice", "ieee:s");
22     error = jsonrpc_server_register_method(self, JSONRPC_TRUE, server_zapSetChannel, "zapSetChannel", "channel:i");
23     error = jsonrpc_server_register_method(self, JSONRPC_TRUE, server_zapSendUnicast, "zapSendUnicast", "ieee:s,profileId:i,clusterId:i,sourceEndpoint:i,destinationEndpoint:i,message:s,needConfi");
24
25     //resetZap();
26     zapInit();
27     if(zapNetworkInit() == 1)
28     {
29         LogInfo("zapNetworkInit ok\n");
30     }
31
32     lastTime = system_get_time();
33     mtorrIntervalTime = 5*60*1000*1000;
34
35     while(1)
36     {
37         zapTick();
38         if(self != NULL)
39         {
40             zapInit();
41             if(zapNetworkInit() == 1)
42             {
43                 LogInfo("zapNetworkInit ok\n");
44             }
45
46             lastTime = system_get_time();
47             mtorrIntervalTime = 5*60*1000*1000;
48
49             while(1)
50             {
51                 zapTick();
52                 if(self != NULL)
53                 {
54                     error = jsonrpc_server_run(self, 0);
55                 }
56                 else
57                 {
58                     LogError("%s %d err\n", __FUNCTION__, __LINE__);
59                 }
60
61                 nowTime = system_get_time();
62                 if(nowTime - lastTime > mtorrIntervalTime)
63                 {
64                     LogDebug("nowTime = %d, lastTime = %d\n", nowTime, lastTime);
65                     lastTime = nowTime;
66                     if(gNetWorkStatus == 1)
67                     {
68                         zapSendMTORR();
69                     }
70                 }
71             }
72             LogError("%s return err\n", __FUNCTION__);
73             jsonrpc_server_close(self);
74         }
75     }
76 }

```



### 3.3、测试模块

在硬件和软件单独完成并测试后，下一步就是整个系统整合后的整机测试，该阶段目前只完成了最小系统的测试，即将主机、ZAP、个别传感器进行整合，可以实现基本功能要求，但是由于天线功率等原因，信号不是很稳定，灵敏度不强。下一步就将进行信号调试，准备将剩余的所有传感器一律整合，完成完整系统的测试，并拍摄实验测试视频。

### 3.4、论文撰写

由于该系统比较庞大，所涉及的硬件和软件，以及相关技术比较繁多，在预估到论文内容会很长的原因，因此在完成硬件开发和软件设计之后，整机测试之前就开始了论文的撰写，以防止最后时间来不及撰写论文的情况发生。论文在撰写之前就先做出了一份论文的详细目录，设计的技术实现与陈述处于论文的中后端，而前面开头部分属于综述性质的内容，因此，虽然设计还并没有完全结束，但是论文的前面部分仍然可以先开始完成。目前就论文而言，已经基本做完一半，完成了论文基本介绍、主机模块、ZAP 模块、只剩传感器模块与后续的总结部分，预计在四月末就可以完成，然后五月十号之前完成论文的审阅。

## 四：论文遇到的问题及拟解决办法

论文设计中主要遇到了以下问题：

- 1：个别传感器由于是购买所得，因此硬件电路不是很熟悉，在测试中遇到了阻碍。对于这个问题，准备通过联系商家，获得详细原理图与厂家售后技术支持的帮助。
- 2：设备的 Zigbee 信号不够强，也不稳定。准备更改陶瓷天线与电路的 PA 参数设置来增强信号强度。
- 3：整机测试环境如何搭建。由于该系统运行时需要多种触发条件，和多种设备操作动作，因此需要一个良好的测试环境，能够将所有测试设备和操作都包含进来，同时也方便后期录制测试视频。对于该问题，决定采用将所有相关设备借助实习公司的展厅来完成，该展厅提供了一套完整的测试环境。

## 五：后期安排

目前在一边进行论文的撰写，一边进行系统的测试与改进，还有测试视频未录制。接下来计划先把系统的测试与调试改进完成，顺便将视频录制完毕，这样就可以顺畅的撰写论文了，否则论文一直处于修改状态，使得效率低下，浪费时间。安排如下：

- 1、解决信号与个别传感器电路问题
- 2、完成测试环境搭建
- 3、完成整机测试并录制视频
- 4、完成论文撰写
- 5、联系导师并进行论文审阅
- 6、正式答辩