

---

# 毕业设计（论文）

题目 蓝牙技术在安卓系统中的应用

专业名称 软件工程

班级学号 XXXXXX

学生姓名 XXX

指导教师 XXX

二〇一五年六月

---

## 摘要

当今世界是信息化世界，一切都在走信息化道路，数据的传输是信息化道路上的重要问题。蓝牙作为一种低能耗，高效率的稳定无线传输方式，必会在未来得到广泛应用。对于用户而言，开发一款安卓手机上的蓝牙功能集合软件具有必要性。

本文已经实现了一款汇总安卓手机蓝牙功能的安卓应用。软件展示了蓝牙的打开与关闭，蓝牙可见性设置，蓝牙搜索，蓝牙聊天，蓝牙文件传输，蓝牙共享网络，通过蓝牙访问周围数据源等功能在安卓手机上的具体实现。

软件采用C/S架构，以安卓SDK为开发平台，ALCATEL ONETOUCH PIXI 3-4.5（3g）为测试机，CoR责任链模式作为行为模式进行开发和测试。本文描述了软件从需求分析，架构设计，到详细设计，再到实现与测试的开发过程。所设计的软件，满足了设计要求。

**关键词：**安卓 蓝牙 应用

---

## Abstract

Nowadays is a world called informational world, all operations have been informationized gradually and transmission of data is an important issue of information technology. Bluetooth technology will be widely used in the future because of its stable wireless transmission way is low-power and high efficiency. Therefore the developing of a set of Bluetooth software in Android phone is necessary for users.

This paper has achieved an android application with comprehensive Bluetooth functions. The software include the opening and closing of Bluetooth, Bluetooth visible settings, Bluetooth Search, Bluetooth Chat, Bluetooth file transfer, Bluetooth sharing networks and Bluetooth access data sources around. These can be implemented specifically in Android phone.

The software using C / S structure and the Android SDK development platform .And the paper choose the ALCATEL ONETOUCH PIXI 3-4.5 (3g) as test machine. The CoR chain of responsibility pattern is employed to develop and test software. This paper describes the software development process include requirements analysis, structure design, the specific detail design and implementation and testing of the software. Eventually, the software meet the design requirements.

**Keywords:** Android Bluetooth Application

---

# 目 录

1 绪论	
1.1 课题的来源, 意义及目标	2
1.2 蓝牙功能的现状与发展	2
1.3 需要完成的任务及设计结果	3
1.3.1 任务内容	3
2 需求分析及总体设计	
2.1 需求分析	5
2.1.1 可行性分析	5
2.1.2 总体需求说明	5
2.1.3 实现平台需求	5
2.2 总体设计	6
2.2.1 软件结构	6
2.2.2 软件主要功能	7
3 程序设计及编码实现	
3.1 程序设计	8
3.1.1 软件业务流程	8
3.1.3 包格式设计	12
3.2 编码实现	13
3.2.1 编程综述	13
3.2.2 基础功能	15
3.2.3 连接与 OBEX	16
3.2.4 PAN 与数据源访问	24
3.2.5 Android APK 构建	25
4 测试	
4.1 单元测试用例及测试报告	27
4.2 测试方法	27
5 评估与改进	
5.1 软件设计评价	28
5.2 改进计划与未来目标	28
5.3 今后进一步改进计划	29
参考文献	30
致 谢	31

# 1 绪论

## 1.1 课题的来源，意义及目标

本课题是 TCL 通讯惠州研发中心所提供，本人根据指导老师意见修改而来。在现实生活中，身边的亲戚在使用手机蓝牙功能时，都难以找到蓝牙功能菜单，而根据公司的用户反馈，蓝牙功能不易于找到并使用，使得公司手机用户体验受到影响，甚至于有些蓝牙功能很少有人知道。研发一款有关蓝牙应用的软件势在必行。

完成本次毕设，有利于提升本人的编程水平，便于身边朋友亲人使用蓝牙，为公司提供完整的蓝牙功能集合，节约使用蓝牙时间，提升用户体验。

本软件所要实现的是安卓手机已有的蓝牙基本功能的汇总和拓展，需要完成的功能包括蓝牙的打开与关闭，蓝牙可见性设置，蓝牙数据传输，蓝牙共享网络，通过蓝牙实现对周围数据源的访问。以上功能通过调用安卓 SDK 封装好的 API 实现，用户仅需要打开软件通过点击按钮就可以使用这些安卓手机上的蓝牙功能。

## 1.2 蓝牙功能的现状与发展

随着安卓手机在全世界范围内流行，整个时代进入移动互联网时代。互联网的发展，使得人们渐渐淡忘了在塞班系统中风靡一时的蓝牙技术。但是在流量不够充足的情况下，想要面对面分享资源，蓝牙又是必不可少的一种分享手段。随着可穿戴式计算的发展，不管是智能手表还是智能手环，都需要蓝牙连接，蓝牙必将再一次进入大众的眼球，成为衡量手机质量的一大标准。

蓝牙技术已广泛应用于智能家居市场，具有成本低，低功耗的特点，而且当今的移动通信设备大多支持蓝牙。可以通过蓝牙来控制门窗，电器，家具，车库的工作，而这些仅仅需要一部蓝牙手机就可以实现，而蓝牙低功耗的特性可以弥补手机电源不够充足的缺陷，蓝牙技术将瞄准一切低功耗的场景来发展。为了更好的实现智能化，未来蓝牙芯片将和传感器进行深度融合，可以通过传感器感知使用人，将使用人需要的工作学习资料信息上传云端，并通过蓝牙技术传输到任何一台使用人所需使用的支持蓝牙的设备上，方便使用人的工作学习。基于蓝牙的 Beacon 定位技术精度高，成本低，未来蓝牙技术可以应用到销售领域。当客户手持蓝牙设备走进商店，商店蓝牙设备将根据蓝牙设备上的数据为你推荐所需商品从而避免了长时间挑选的尴尬。总之，蓝牙技术将在未来大放异彩。

## 1.3 需要完成的任务

### 1.3.1 任务内容

软件要实现蓝牙的打开与关闭, 蓝牙可见性设置, 蓝牙聊天, 蓝牙文件传输, 蓝牙共享网络, 通过蓝牙对周围数据源访问六个功能模块。

#### (1) 蓝牙的打开与关闭

调用安卓蓝牙 API 自带的命令实现此功能, 所要实现的效果为用户通过点击“打开/关闭”蓝牙按钮控制蓝牙无线电的开关, 状态栏中的蓝牙图标随着用户操作产生变化。

#### (2) 蓝牙可见性设置

同样调用 API 自带命令实现, 蓝牙可见性即为蓝牙设备可被其他设备检验到的属性, 设置设备可见, 该设备才会被搜索到并且添加到正在搜索设备的机器内置可连接设备列表。所需要实现效果为用户点击按钮设置设备可见, 安卓系统弹出提示框询问用户是否执行该操作。

#### (3) 蓝牙聊天

蓝牙聊天即为两个用户使用两台已连接的蓝牙设备进行文字交流。一端用户在输入框键入想要表达的消息, 然后点击按钮发送消息, 另一端的用户在消息框看到接收到的消息并回复从而达到聊天的目的。在此过程中, 聊天数据通过蓝牙传输。

#### (4) 蓝牙文件传输

蓝牙文件传输就是通过蓝牙连接将用户想要分享的文件发送到需要接收文件的蓝牙设备上, 要保证传输文件的完整性和传输前后格式的一致性。用户在软件实现的文件列表中选择所需要传输的文件, 然后点击按钮进行文件传输, 接收方接收文件并将其放入存储空间, 文件与传输之前完全一致。

#### (5) 蓝牙共享网络

蓝牙共享网络就是将蓝牙设备的网络通过蓝牙分享, 使得其他蓝牙设备可以不打开 WiFi 而通过蓝牙连接获取网络连接。因为蓝牙低能耗的特性, 这种方式较之 WiFi 更为省电。用户通过点击网络共享按钮进入网络设置并设置通过蓝牙共享网络, 其他蓝牙设备连接蓝牙后将共享网络后的设备设为接入点, 这样就可以通过该设备上网。

#### (6) 通过蓝牙对周围数据源访问

通过蓝牙对周围数据源访问就是蓝牙设备通过蓝牙连接访问 PC（个人电脑）上的数据库。用户在软件中与 PC 建立蓝牙连接并点击接受动态按钮，数据库将通过用户的蓝牙地址找到的用户动态展示给用户。PC 端管理员可以对用户动态进行增删查改的操作。

## 2 需求分析及总体设计

### 2.1 需求分析

#### 2.1.1 可行性分析

目前安卓系统已发展至 Android 5.0 Lollipop 版本，而市面上手机更多的还是基于 Android 4.4 KitKat 版本，所以软件会基于 Android 4.4 KitKat 版本。安卓系统支持蓝牙基础功能，提供了蓝牙相关 API。当前安卓手机之间传文件，更多人会使用蓝牙，所以蓝牙在安卓系统上的应用有研究的必要性，而且技术上可行。

#### 2.1.2 总体需求说明

在蓝牙软件需要实现以下功能：

- (1) 基础功能：打开/关闭蓝牙，蓝牙可见，蓝牙连接
- (2) 数据传输：蓝牙聊天，文件传输
- (3) 数据共享：数据源访问，网络共享

蓝牙软件必须符合以下特性：

- (1) 基础功能可用且有特征显示。
- (2) 数据传输安全，可靠。蓝牙聊天内容格式不能发生改变，文件传输所传文件完整，格式正确且文件仍然可用。
- (3) 网络共享的网络连接稳定，数据源访问用户身份正确。
- (4) 界面友好易用。
- (5) PC 端服务器可维护，数据操作正确。

#### 2.1.3 实现平台需求

选择正确的语言及环境，是软件成功实现的保障。为了完美实现该软件，选择以下的编程环境：



硬件环境：ALCATEL onetouch PIXI3-4.5 手机，G480/1024M/8G 电脑，悦人心智蓝牙适配器

操作系统：Android 4.4 KitKat，WIN7 64 位

编译环境：Eclipse+安卓 SDK，MyEclipse+SQL Server 2008

安卓 SDK 是谷歌提供的安卓软件开发工具包，其中包含了各个版本的安卓系统开发所需的所有的开发工具，Eclipse 是由 IBM 公司贡献给开源社区的 JAVA IDE（跨平台开源集成开发环境）。将安卓 SDK 部署到 Eclipse 上就可以通过虚拟机对大多数安卓应用进行编译和调试。

MyEclipse 是在 eclipse 基础上部署自己的插件从而能开发成的功能强大的企业级集成开发环境，毕设中用 MyEclipse 部署访问 PC 服务器的 WEB 工程。

SQL Server 2008 是美国微软公司提供的一种关系数据库管理平台，毕设中使用它部署 PC 数据源。

安卓虚拟机没有蓝牙组件，无法对蓝牙软件进行支持，蓝牙软件必须进行震级调试，所以在这里引进公司提供的测试手机 ALCATEL onetouch PIXI3-4.5，这款手机完美提供了软件所需测试环境。

## 2.2 总体设计

### 2.2.1 软件结构

蓝牙软件设计为三大模块。第一个模块是基础功能模块，该模块包含蓝牙的打开与关闭，蓝牙可见性设置操作，这些功能是其它功能的基础；第二个模块是蓝牙连接与数据传输模块，该模块包含蓝牙搜索，蓝牙 OBEX（对象交换功能），蓝牙数据传输包含蓝牙聊天和文件传输等功能，这是当前蓝牙的常用功能；第三个模块是高级功能模块，该模块包含蓝牙共享网络和通过蓝牙 PAN（个人局域网）访问 PC 端数据源功能，其中，访问数据远功能可作为蓝牙物联网研究模板。图 2.1 为软件设计结构图。

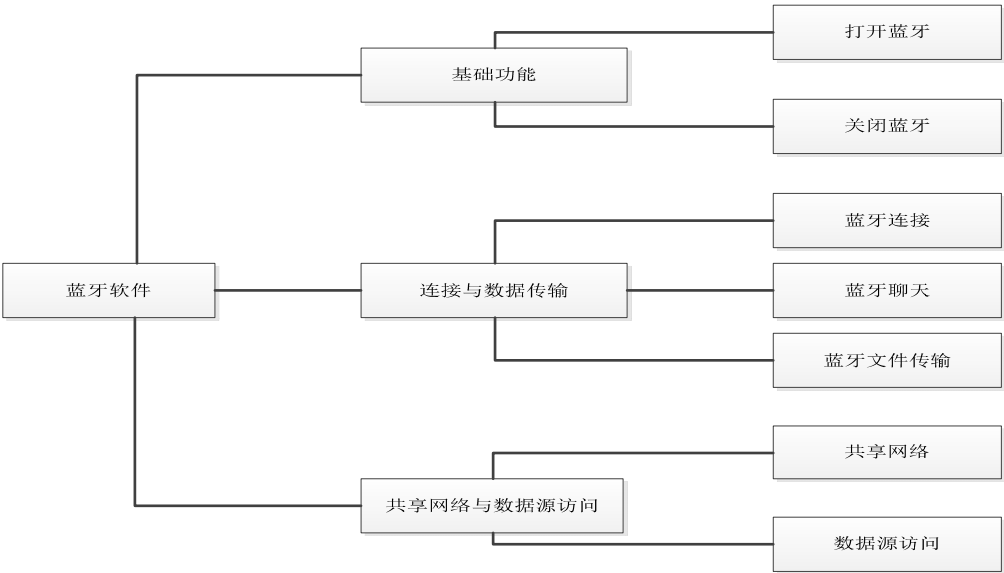


图 2.1 蓝牙软件系统结构图

2.2.2 软件主要功能

蓝牙软件所实现的功能如表 2.1 所示，表格最后一列是功能的具体描述。

表2.1 软件功能表

编号	功能名称	功能描述
1.	基础功能	
1.1	打开/关闭蓝牙	用户通过点击按钮实现蓝牙无线电的开关
1.2	蓝牙可见性设置	用户通过点击按钮实现蓝牙可被搜索属性的设置
2.	蓝牙连接与数据传输	
2.1	蓝牙搜索	一个用户搜索其它可见用户并将其添加到可连接设备列表
2.2	蓝牙连接	用户选择一个可连接设备进行蓝牙连接

2.3	蓝牙聊天	用户与已连接的蓝牙设备进行文字交流
2.4	文件传输	用户选择文件列表中的文件分享给其它已连接用户
3.	<b>共享网络与 数据源访问</b>	
3.1	共享网络	用户通过蓝牙无线电分享自己的网络给其它已连接用户
3.2	数据源访问	用户通过蓝牙个人局域网访问 PC 端的数据源

## 3 程序设计及编码实现

### 3.1 程序设计

#### 3.1.1 软件业务流程

软件使用流程如下：

##### （1） 基础功能设置

用户打开软件，进入基础功能界面，将蓝牙开关打开。点击蓝牙可见按钮并同意操作，使该设备可被其它设备搜索到。这一步是基础，其它操作都建立在此操作之上。

##### （2） 蓝牙搜索与连接

此操作需要用户拥有两台蓝牙手机。用户进入蓝牙连接与数据传输界面，使用其中一台点击搜索按钮进行搜索。这台机器会将所有搜索到的设备添加到搜索到但未连接的设备列表中。用户选择搜索到的另一台手机进行连接，确认验证码并同意操作，设备会将完成连接的设备添加到已连接列表。

##### （3） 蓝牙聊天

用户选择已连接设备进入聊天界面，在输入框输入想要表达的信息并点击发送按钮。已连接的设备会收到发送端传来的信息并将其添加到显示框中。传输来的信息文字和格式不能发生任何改变。

##### （4） 文件传输

用户选择已连接设备进入文件传输界面，点击选择文件进入做好的文件列表中，选择想要传输的文件并确定，返回文件传输界面点击发送。已连接设备接收文件并将文件放入存储空间。传输来的文件格式和内容不能发生改变并且文件仍可用。

### (5) 网络共享

用户进入网络共享和数据源访问页面，点击网络共享按钮进入网络设置界面，选择通过蓝牙共享网络并返回软件。

### (6) PC 端操作

用户将蓝牙适配器插入电脑并安装驱动，在控制面板搜索并添加共享网络的设备，右键点击设备并将其设为接入点，此时电脑就可以通过共享的网络连接上网。管理员在 PC 端打开蓝牙软件所对应的 WEB 服务，可根据需要管理最新动态。

### (7) 访问数据源

用户进入网络共享和数据源访问页面，点击接受动态即可访问数据源，才是要保证蓝牙软件所在设备与 PC 数据源在同一局域网或者连接网络，PC 服务器已启动。

总体流程如下图，其中 OBEX 为对象交换功能，DUN 为拨号上网功能：

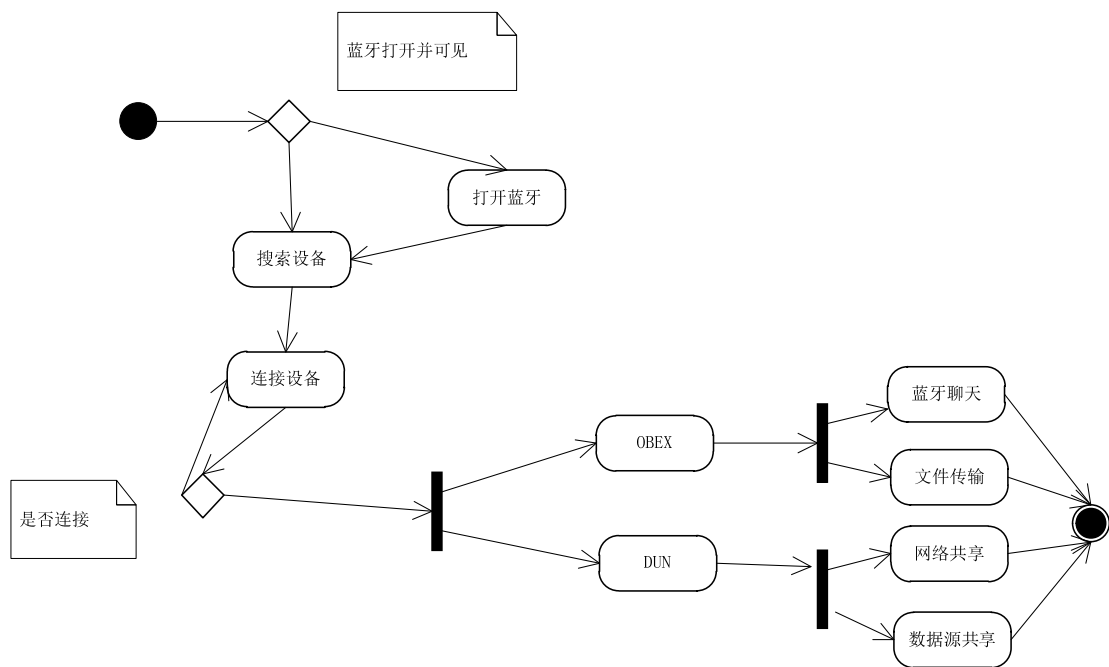


图 3.1 软件总体流程图

3.1.2 蓝牙软件各模块顺序图

基础功能模块顺序图如图 3.2 所示，用户在使用蓝牙软件基础功能时，只需要通过向 Android 系统发送相应命令，等待系统响应即可。因其实现简单，仅需单机测试，所以将这些功能汇总到一个模块中。

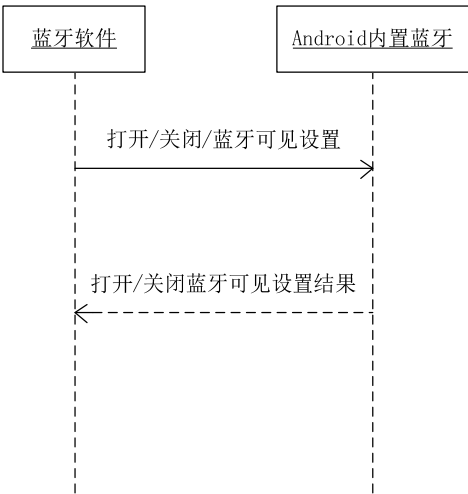


图 3.2 基础功能模块顺序图

蓝牙连接与数据传输模块顺序图如图 3.3 所示，需要分别建立客户端和服务端，客户端与服务端之间通过无线电广播进行通信，实现该模块需要有两台手机。因为该模块所有功能都需要两台手机进行测试，并且全部有关与移动终端异步无连接通信，所以将这些功能设为一个模块。

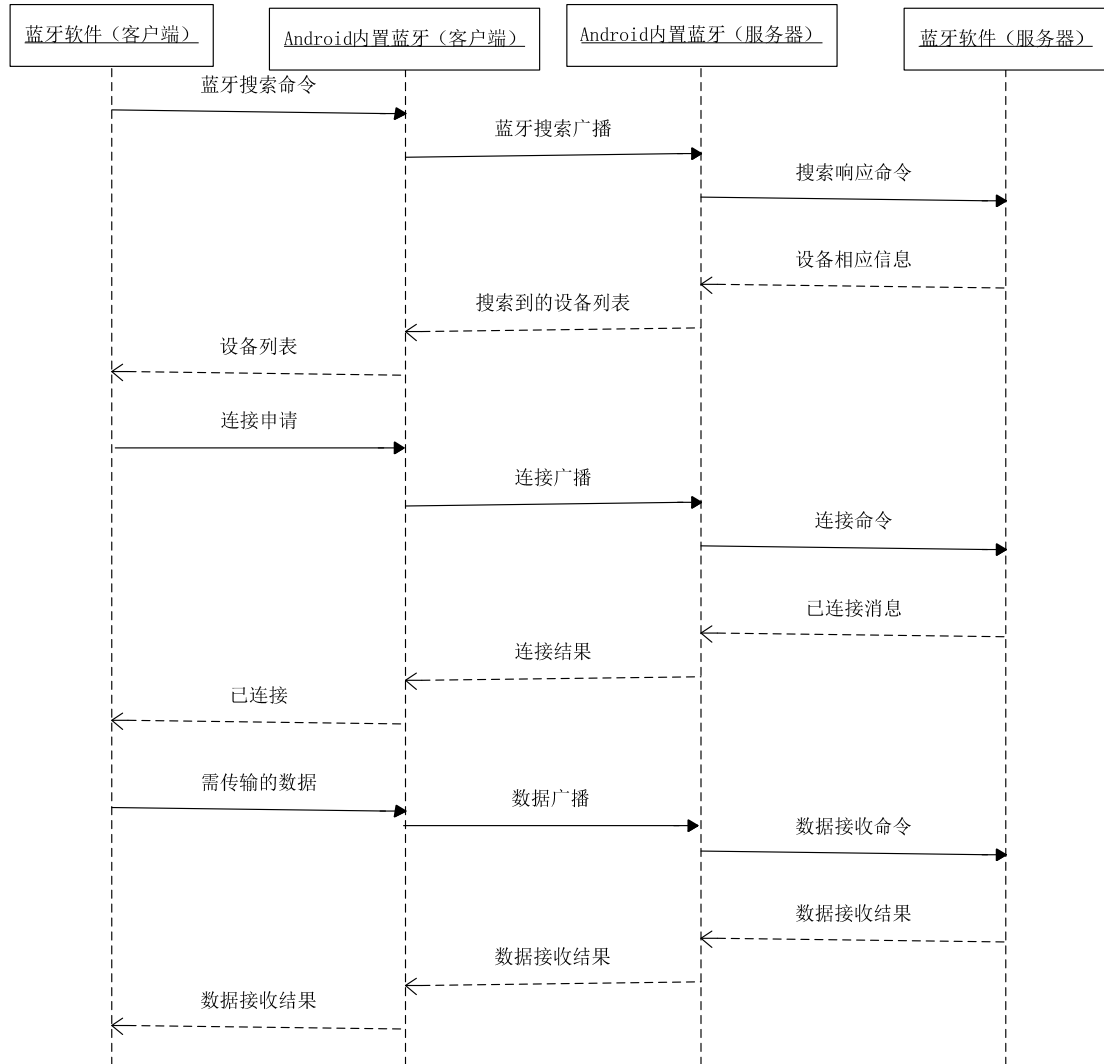


图 3.3 蓝牙连接与数据传输顺序图

高级功能模块顺序图如图 3.4 所示，高级功能部分的实现需要分别在 PC 端和移动终端部署服务器和客户端，PC 端与移动终端通过蓝牙建立的 PAN（个人局域网）进行数据传输，因为这些功能全部需要 PC 以及移动终端，并且实现过程中需要保持连接，所以本文将其部署到了一个模块中。

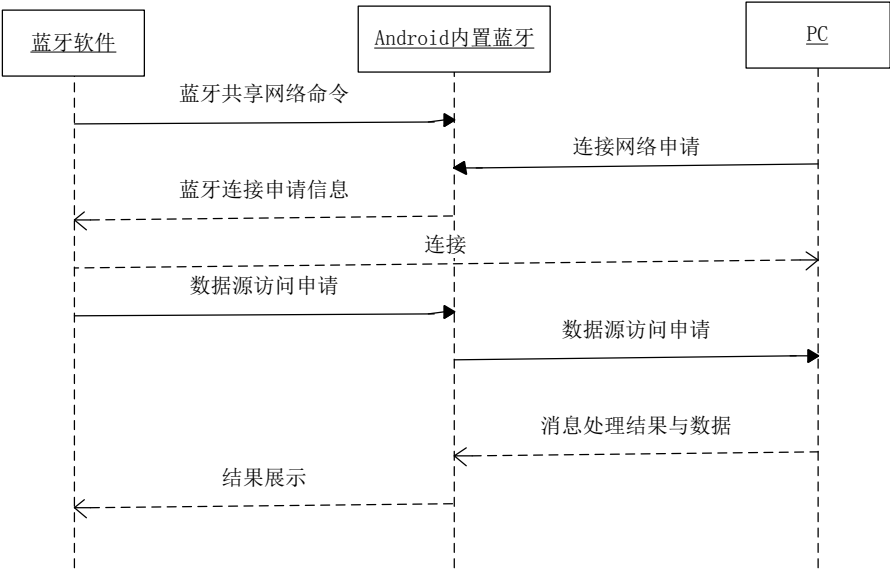


图 3.4 高级功能顺序图

3.1.3 包格式设计

根据软件需求分析和设计，设计了五个包，分别为主界面包，蓝牙连接与数据传输包，蓝牙格式化包，文件输入输出流包，功能改进包。如图 3.5 所示。

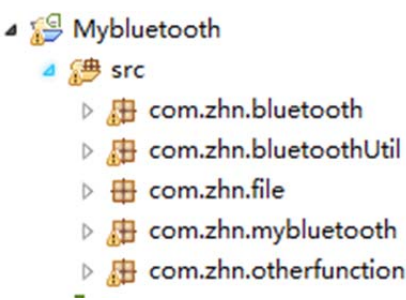


图 3.5 软件包设置图

mybluetooth 包是主界面包，包含了欢迎界面，基础功能界面和高级功能界面。

Bluetooth 包为蓝牙连接与数据传输包，包含了该功能的所有界面（蓝牙连接和数据传输功能主界面，客户端界面，服务器界面，数据传输界面）。

bluetoothUtil 包为蓝牙格式化包，包含蓝牙适配器，UUID 声明，数据传输套接字与线程的声明等一系列需要经常调用的工具类。

file 包为文件输入输出流得格式化包，该包调用了 Android 文件输入输出流，文件适配器等功能，实现文件格式化。

Otherfunction 包为功能改进包，为以后改进软件提供便利，在本次毕设中未启用。

### 3.1.4 PC 端服务器详细设计

因本次毕设主要展示蓝牙功能，所以所搭建的 PC 端服务器仅含一张表，该表主键为蓝牙 mac 地址，作为机主身份的唯一标识。note 属性为机主该接收的最新动态。表 3.1 为数据库表结构：

表 3.1 PC 服务器数据库表设计

字段名称	数据类型及长度	允许为空	中文说明	备注
mac	varchar(100)	NO	蓝牙 mac 地址	主键
note	varchar(1000)	yes	个人动态	

PC 端服务器数据的增删查改使用 JSP+Servlet 技术实现，设计增加信息页面，信息列表页面，显示个人信息页面和跳转页面。数据处理放入 DAO 包中，servlet 放入 action 包中，表数据结构放入 entity 包中，biz 包中放入数据处理单元的抽象类。

## 3.2 编码实现

### 3.2.1 编程综述

该节论述了软件实现所必须的蓝牙功能和蓝牙属性，包括蓝牙物理地址，蓝牙适配器，唯一服务标识符 UUID，套接字与线程，PC 服务器。以下为详细介绍。

#### (1) 蓝牙 mac 地址

BD\_ADDR：蓝牙设备地址

蓝牙地址是 48 位 IEEE 802 地址，蓝牙地址由三部分组成：

LAP——>24 位地址低端部分。



UAP——>16 位地址高端部分。

NAP——>8 位无意义地址部分。

LAP 是由厂商自由指定的地址段，用于自己品牌蓝牙设备的标记区分，UAP 和 NAP 由权威蓝牙部门分配，又称之为单播地址，是厂商的唯一标识。

举例说明：

20: 15: 23: 14: 22: 42

在上面的蓝牙地址中，20:15 就是 UAP， 23 就是 NAP，而之后的 14: 22: 42 则为 LAP，也就是说 20: 15: 23 就是此蓝牙地址对应厂商的唯一标志（单播地址）。请注意，蓝牙地址第二位是奇数的时候是作为广播地址，是不能转发数据包，只有作为单播地址才能有效的转发。

## （2）蓝牙适配器

BluetoothAdapter

代表本地蓝牙适配器（蓝牙无线电）。

BluetoothAdapter 是所有蓝牙活动的入口点。蓝牙软件将利用这一点搜索其他蓝牙设备，查询可配对的设备列表，使用已知的 MAC 地址实例化一个 BluetoothDevice 类，并创建一个 BluetoothServerSocket 侦听来自其他设备的通信。

安卓系统封装了蓝牙 Adapter 类，软件只需要使用一句代码调用这个类建立蓝牙适配器。代码如下：

```
Private BluetoothAdapter bluetooth= BluetoothAdapter.getDefaultAdapter();
```

## （3）UUID

UUID 含义是通用唯一识别码（Universally Unique Identifier），用于标志蓝牙服务。在软件实现中所需要的服务，都必须要建立服务所对应的 UUID 号。如下代码用于建立用于安卓手机与安卓手机之间的蓝牙通讯的 UUID：

```
public static final UUID PRIVATE_UUID =  
UUID.fromString("00001101-0000-1000-8000-00805F9B34FB");  
//将 UUID 格式化为字符串
```

## （4）套接字与线程

蓝牙异步通信需要套接字作为连接点，本程序使用一个客户端和一个服务器进行通信，即单线程编程。

### (5) PC 服务器

采用 JSP+servlet 技术搭建 web 网站，sql server 作为后台数据库构建 PC 端服务器，蓝牙移动设备通过蓝牙个人局域网访问服务器页面获取信息。

## 3.2.2 基础功能

### (1) 蓝牙的打开与关闭

在安卓系统使用蓝牙须在 xml 文件中加入蓝牙权限。

```
<uses-permission android:name="android.permission.BLUETOOTH" />
<!-- 蓝牙可用权限 -->
<uses-permission android:name="android.permission.BLUETOOTH_ADMIN"/>
<!-- 蓝牙所有功能可用权限 -->
```

安卓提供的打开蓝牙的命令为 enable(), 关闭蓝牙的命令为 disable()。蓝牙无线电的状态用蓝牙适配器的 STATE\_ON, STATE\_OFF 参数标志，这些功能和状态由 BluetoothAdapter 创建的对象来调用。如图 3.6 为打开/关闭蓝牙功能实现效果，其中左图为打开蓝牙状态，右图为关闭：

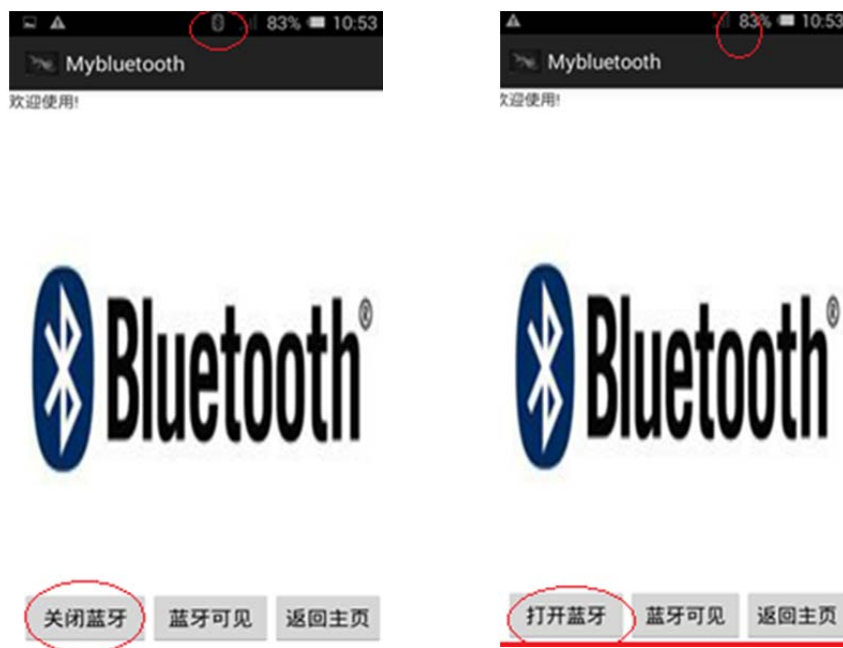


图 3.6 打开/关闭蓝牙功能效果图

## （2）蓝牙可见性

蓝牙可见性即蓝牙可被其他设备搜索到的特性，安卓系统默认蓝牙设备为不可见状态，需手动开启蓝牙可见。安卓赋予蓝牙的可见性标志为 ACTION\_REQUEST\_DISCOVERABLE，此参数同样利用 BluetoothAdapter 的对象调用，使用安卓 Intent 加载并使用 Activity 开启。如图 3.7 为蓝牙可见性设置效果图：



3.7 蓝牙可见性设置效果图

### 3.2.3 连接与 OBEX

#### （1）蓝牙服务器

安卓系统蓝牙连接需建立一个套接字服务器作为服务器接入点，该服务器需要具有蓝牙适配器，蓝牙 mac 地址，并且服务器蓝牙无线电状态需设为可见。

蓝牙服务器需要通过线程来注册一个 BluetoothServerSocket，它将作为服务器的接入点，这个 BluetoothServerSocket 需要具有名称并且还要有唯一标识 UUID 号用以标志服务。同样的，还需要给服务器一个可用的蓝牙适配器 BluetoothAdapter。然后服务器的工作就是一直监听蓝牙客户端的请求，并对应这些请求做出相应处理。

在毕设作品中，因需要建立很多次蓝牙适配器和用于通讯的 UUID 号，并且需要多次进行基础设置，所以在软件中将这常用对象放到了一个类里并取名为蓝牙工具箱（BluetoothTools.java），部分实现代码如下：

```
/**
 * 蓝牙工具类
```

```
* @author 张海南
*
*/
public class BluetoothTools {

    private static BluetoothAdapter adapter = BluetoothAdapter.getDefaultAdapter();

    /**
     * 本程序所使用的 UUID
     */
    public static final UUID PRIVATE_UUID =
UUID.fromString("00001101-0000-1000-8000-00805F9B34FB");
    /**
     * 打开蓝牙功能
     */
    public static void openBluetooth() {
        adapter.enable();
    }

    /**
     * 关闭蓝牙功能
     */
    public static void closeBluetooth() {
        adapter.disable();
    }

    /**
     * 设置蓝牙发现功能
     * @param duration 设置蓝牙发现功能打开持续秒数（值为 0 至 300 之间的整数）
     */
    public static void openDiscovery(int duration) {
        if (duration <= 0 || duration > 300) {
            duration = 200;
        }
        Intent intent = new
Intent(BluetoothAdapter.ACTION_REQUEST_DISCOVERABLE);
        intent.putExtra(BluetoothAdapter.EXTRA_DISCOVERABLE_DURATION,
duration);
    }

    /**
     * 停止蓝牙搜索
     */
    public static void stopDiscovery() {
        adapter.cancelDiscovery();
    }
}
```

```
}
```

## (2) 蓝牙客户端

有了服务器接入点，想要连接成功，蓝牙软件还需要建立蓝牙客户端，服务器和客户端需要部署到不同的两个手机中。在毕业设计中，将蓝牙搜索功能放入了蓝牙客户端。搜索远程蓝牙设备需要调用 `BuletoothDevice` 类，安卓提供了该类。蓝牙软件所需要做的就是调用该类的 `getRemoteDevice` 方法来获取远程设备信息。

蓝牙搜索是蓝牙的重要组成部分。这是一个过程，其中一个蓝牙设备试图找到所有相邻蓝牙设备。正试图扫描其它设备的设备被认为是处于“搜索模式”。设备正在侦听查询请求是“查询扫描模式”。这种“查询扫描模式”通常是设定在一个设定为可见的蓝牙设备上。

当启动搜索时，该设备进入“搜索模式”，并加速其跳频频率。另一方面，该设备在“查询扫描模式”降低了它的跳频频率。该算法将允许在“查询扫描模式”时查询设备赶上发射频率设备。这是因为，频率的跳频算法是蓝牙使用中较为重要的算法。当频率一致时，扫描设备将充当从设备，发送其掌握的地址和时钟信息。搜索结束后，查询设备将能够发起连接到所查询设备。因为由查询设备发起连接，它将作为主设备。此初始连接被称为分页。寻呼可能从频率插槽由主设备完成发送寻呼请求。这个频率间隙计算从蓝牙地址来的和在接受调查的时钟信息。蓝牙搜索流程图如图 3.8:

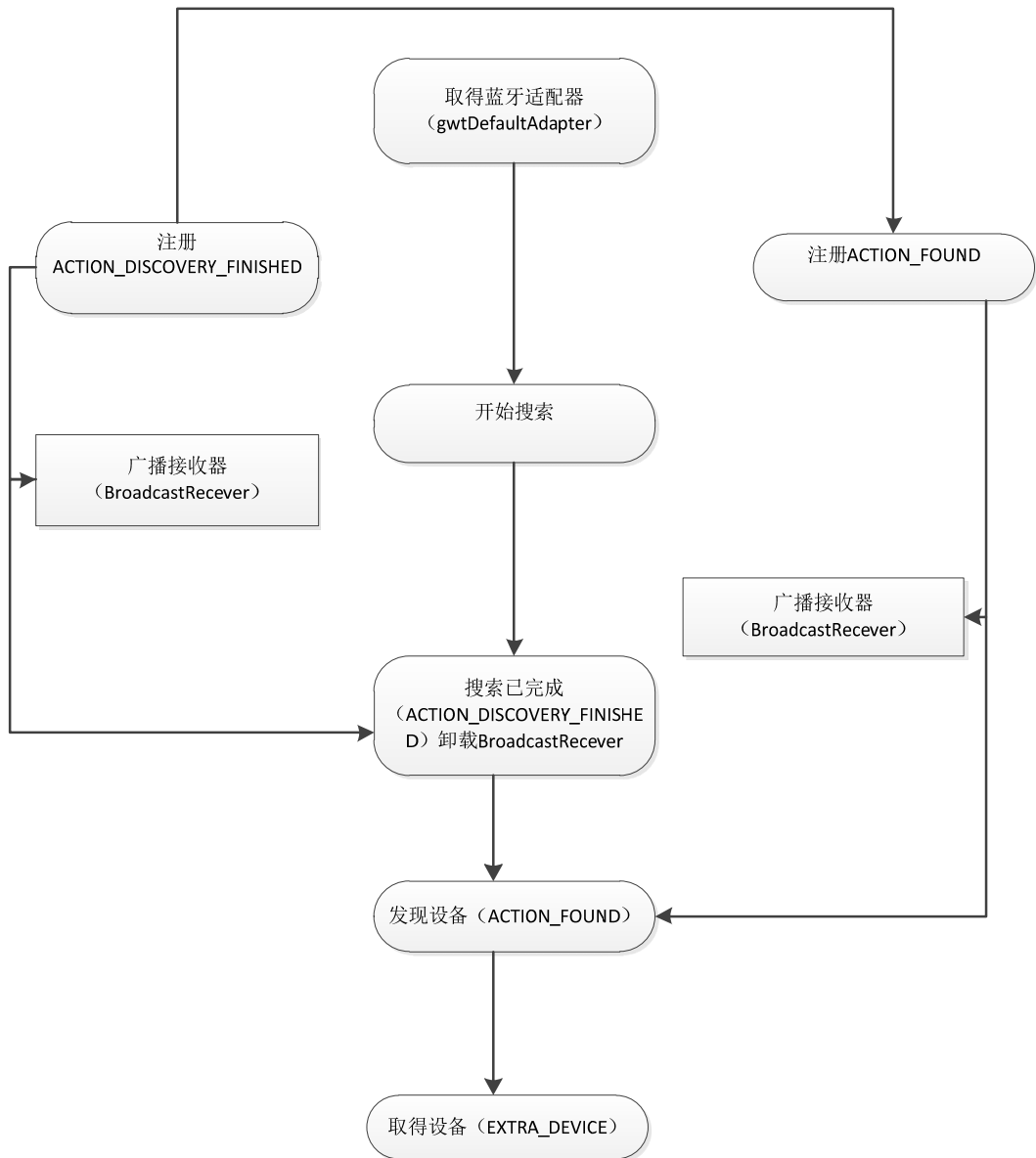


图 3.8 蓝牙搜索流程图

搜索完成后，蓝牙客户端需要做的是选择一个服务器（搜索出来的蓝牙设备）然后通过唯一标识符 UUID 创建一个 BluetoothSocket 对象来进行连接，连接需要用到 BluetoothSocket 的 connect 方法。蓝牙软件在连接时会将已经连接的设备添加到已连接的设备列表中。效果图如图 3.9 所示：



图 3.9 蓝牙搜索连接效果图

实现代码如下：

```
private BroadcastReceiver searchDevices = new BroadcastReceiver() {  
    //注册广播接受器  
    @Override
```

```

public void onReceive(Context context, Intent intent) {
    String action = intent.getAction();
    if (BluetoothAdapter.ACTION_DISCOVERY_STARTED.equals(action)) {
        progressDialog = ProgressDialog.show(context, "请稍等...",
            "搜索蓝牙设备中...", true);    //搜索显示提示语
    } else if
(BluetoothAdapter.ACTION_DISCOVERY_FINISHED.equals(action)) {
        System.out.println("设备搜索完毕");
        progressDialog.dismiss();
        /*客户端列表监听器，加入设备列表
        UnbondDevices 是未连接设备， BondDevices 是已连接设备    */
        clientListListener.addUnbondDevicesToListView();
        clientListListener.addBondDevicesToListView();
    } else if (BluetoothDevice.ACTION_FOUND.equals(action)) {
        //获取到设备对象
        BluetoothDevice device =
intent.getParcelableExtra(BluetoothDevice.EXTRA_DEVICE);
        String str= device.getName() + "|" + device.getAddress();
        System.out.println(str);

        short rssi = intent.getExtras().getShort(BluetoothDevice.EXTRA_RSSI);
        Log.v("调试", "rssi:"+rssi); //安卓调试用句，输出日志

        if (device.getBondState() == BluetoothDevice.BOND_BONDED) {

            //信号强度。

            if (!bondDevices.contains(device)) {
                bondDevices.add(device); //将已连接设备添加至已连接列表
            }
        } else {
            if (!unbondDevices.contains(device)) {
                unbondDevices.add(device); //将未连接设备添加至未连接列表
            }
        }
    }
}

```

### (3) 套接字

在同一网络上的两个不同程序通过一个双向的信连接来实现数据的交换，这个连接的一端称为一个 socket。也就是说，Socket 为双向通信的接入点。



蓝牙服务器与客户端的连接需要建立 socket，上文已讲述，此处不再赘述。

#### （4）线程

线程是一个轻量级程序执行单元。本次毕设使用线程的目的是为了用子线程处理耗时操作，例如蓝牙搜索，更新 UI 等，以减轻主线程压力，避免报错。在这里蓝牙软件引用了 Android 的 Handler 机制实现主线程和子线程的通讯。

#### （5）数据传输

蓝牙数据传输，在本次毕设中，数据格式包含字符串和文件流两种，全部使用输入输出流进行传输。蓝牙聊天，使用 inStream 传输，然后使用 outStream 读取显示。本文在这里使用了 TransmitBean 数据类型来获取和展示文件以及字符串。效果图如图 3.1 所示：

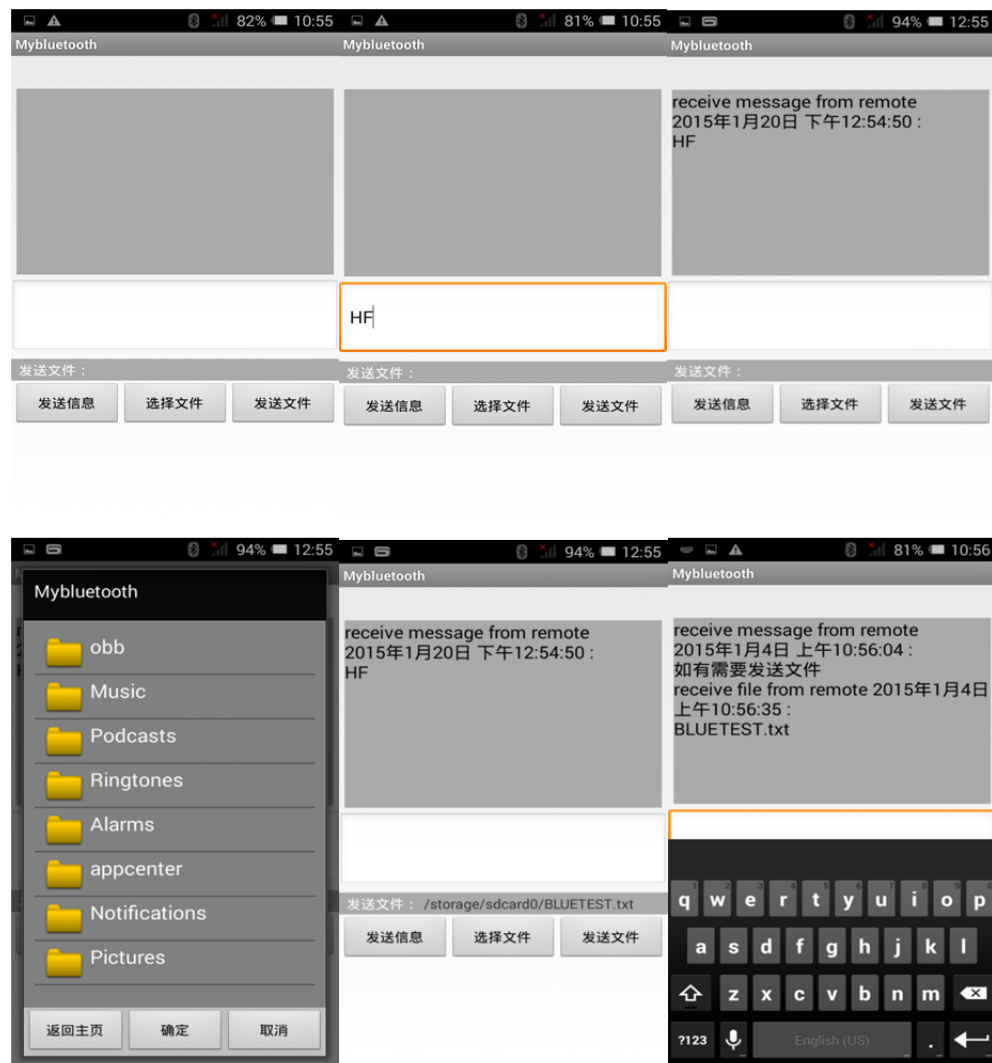


图 3.10 蓝牙数据传输效果图

消息传输代码如下：

```
TransmitBean data = new TransmitBean ();//创建 TransmitBean 对象
data.setMsg(sendEditText.getText().toString());
//将发送框数据转换为 String
Intent sendDataIntent = new Intent(BluetoothTools.ACTION_DATA_TO_SERVICE);
//使用 intent 创建对象
sendDataIntent.putExtra(BluetoothTools.DATA, data);
//将数据加入 intent
sendBroadcast(sendDataIntent);
将 intent 广播出去
sendEditText.setText("");
//提示消息为空
```

文件传输与消息传输实现过程类似，只是文件需要获取其名称和路径，本人在 file 包中建立了 file 相关类。文件转换为 byte[] 形式传输并在接受端转换为文件。在安卓系统中找寻文件需先声明文件适配器。在毕设中，声明放在了 AdapterManager 类中。根据给定路径获取所有路径下文件实现代码如下：

```
public static List<File> getFileList(String path){
    System.out.println("path:"+path);//测试语句
    List<File> fileList = new ArrayList<File>();//建立 file 列表
    File[] files = new File(path).listFiles();
    if(files.length > 0){//获取路径下所有文件夹和文件
        List<File> allFolder = new ArrayList<File>();
        List<File> allFile = new ArrayList<File>();
        for(File file : files){
            if(file.isFile()){
                allFile.add(file);
            }else {
                allFolder.add(file);
            }
        }
        fileList.addAll(allFolder);//将文件夹添加到列表
        fileList.addAll(allFile);//将文件添加到列表
    }

    return fileList;
}
```

数据输入流实现代码如下：

```
public void write(Object obj) {
```

```

try {
    TransmitBean transmit_s = (TransmitBean) obj;
    if(transmit_s.getFilename()!=null&&"".equals(transmit_s.getFilename())){

        Log.v("调试", "type:"+2); //安卓特有调试方式, log 输出
        String filename=transmit_s.getFilename();
        byte type = 2; //类型为 2, 即传文件
        //读取文件长度
        FileInputStream fins=new FileInputStream(transmit_s.getFilepath());
        long fileDataLen = fins.available(); //文件的总长度
        int f_len=filename.getBytes("GBK").length; //文件名长度
        byte[] data=new byte[f_len];
        data=filename.getBytes("GBK");
        long totalLen = 4+1+1+f_len+fileDataLen; //数据的总长度
        outputStream.writeLong(totalLen); //写入数据的总长度
        outputStream.writeByte(type); //写入类型
        outputStream.writeByte(f_len); //写入文件名的长度
        outputStream.write(data); //写入文件名的数据
        outputStream.flush();
        //读取文件并发送
    }
}

```

数据传输通过输入输出流实现, 输出流与输入流相似, 将写入数据取出即可, 这里不再展示。

### 3.2.4 PAN 与数据源访问

#### (1) 蓝牙 PAN (个人局域网)

蓝牙提供的网络共享服务, 需要调用蓝牙 DUN (拨号上网) 功能。PC 端将移动终端作为网络适配器, 然后拨号上网, 此过程中, 所有的数据通讯, 将通过蓝牙无线电传输。在这里调用了安卓系统的设置菜单。

#### (2) 数据源设置

在本次毕设中, 本人搭建了 PC 端数据库, 采用 sql sever 2008 作为数据库管理软件, jsp+servlet 技术搭建 WEB 服务器, 因本次毕设主要焦点放在蓝牙方面, 所以该网站以测试驱动和数据桩为标准搭建。数据库包含 news 表, 以 mac 作为唯一标识符, news 字段存放通知。移动终端连接蓝牙局域网后, 可以通过服务器 ip 访问 WEB 服务器。访问协议 http, 端口号 8080。

#### (3) 数据库数据访问

本节叙述蓝牙设备访问数据库数据的过程。大体流程为蓝牙软件通过浏览器访问 WEB 页面。消息请求通过 PAN 到达 PC 端, servlet 根据蓝牙软件传来的信

息调用相应的 DAO 通过 JDBC 获取数据库数据并将其重定向到相应页面。PC 端将数据和页面通过 PAN 传给移动终端。因其过于简单且不是毕设主题，所以这里不再详细描述实现过程。

### 3.2.5 Android APK 构建

#### (1) Android Activity

Android Activity 是安卓应用的最基本模块，可以根据它的字面意思叫它为活动。通常来讲，在安卓 APK 中看到的一个屏幕就是一个 Activity。Activity 之间用 Intent 进行转换。本软件共有 7 个界面，主界面 Activity\_index，基础功能界面 Activity\_main，蓝牙连接数据传输界面 MainActivity，服务器界面 ServerActivity，客户端界面 ClientActivity，蓝牙通信界面 ClientActivity1，高级功能界面 AdvancedActivity。其他 Activity 用于测试蓝牙通信。在安卓工程中需要为每一个 Activity 在 layout 文件夹下配置 XML 文件，并且在 Androidmanifest.xml 文件下声明每一个 Activity，语法如下：

```
<activity android:name="com.zhn.bluetooth.ServerActivity"
        android:theme="@android:style/Theme.Light"></activity>
```

#### (2) Android 调用内置组件

Android 为每一个内置组件提供了调用接口。在本次毕设中，本人调用了 Android 设置界面来使用蓝牙分享网络，调用 Android 内置浏览器访问 PC 端 WEB 服务器。图 3.11 中左图为蓝牙共享网络，右图为数据共享：



图 3.11 蓝牙共享网络，接收动态效果图

实现代码如下:

```
Intent tetherSettings = new Intent (); //打开安卓 WLAN 设置界面
tetherSettings.setClassName("com.android.settings","com.android.settings.TetherSettings");
startActivity(tetherSettings);
Intent intent= new Intent ();          //打开安卓浏览器并访问 Uri
                                intent.setAction("android.intent.action.VIEW");
                                Uri content_url =
Uri.parse("http://192.168.44.73:8080/Mybluetooth/show_one?mac="+adapter.getmac());
                                intent.setData(content_url);
                                startActivity(intent);
```

### (3) Handler

**Handler** 主要用于处理异步消息: 当发出一个消息之后, 消息将首先进入一个消息队列, 发送消息的函数就会即刻返回, 而接收消息的部分会从消息队列中逐一将消息取出, 然后对消息进行处理, 也就是发送消息和接收消息不是同步的处理。这种机制通常用来处理相对耗时比较长的操作, 比如子线程配合主线程更新界面。

在毕业设计中, 比如搜索操作, 它耗时很长, 超过了 5 秒, 如果将其放到主线程中处理, 那么主界面会在一段时间内陷入假死状态, 主界面陷入假死状态一旦超过 5 秒, 安卓系统将强制关闭软件。那么如果本人利用 **Handler** 的消息分发机制将此操作分配给一个子线程, 并安排其执行的时间和位置, 那么强制关闭的错误就不会再出现。

## 4 测试

### 4.1 单元测试用例及测试报告

本次单元测试聚焦在蓝牙连接，蓝牙聊天以及蓝牙文件传输功能模块。采用测试方法为判定覆盖测试方法。

经分析可知，在此功能模块中有 4 次判定，根据判定，设计测试用例如下：

Case 1: 打开蓝牙情况下进行连接并传输数据

Case 2: 蓝牙关闭情况下连接并使用 DUN

测试结果：

Case 1 不存在问题，Case 2 蓝牙无法连接。

### 4.2 测试方法

（1）在代码中添加 log 输出。Android 调试方法，语法为

`Log.v("调试", "type:"+2);`这种格式。

（2）运行程序，执行用例。结果见 4.1 节。

（3）用户测试，寻找他人使用软件。软件符合用户需求。

（4）压力测试，不停地连接，发送，断开连接，搜索设备。软件经受压力测试，运行完好。

## 5 软件优缺点与改进

### 5.1 软件优点

本文设计的软件是一个对于安卓蓝牙功能汇总的具体实现,该软件实现了蓝牙的打开与关闭,蓝牙可见性设置,蓝牙数据传输,蓝牙网络共享这些安卓所提供的基本功能,为用户使用安卓手机蓝牙功能提供了方便,大大节省了用户使用蓝牙功能的时间,也为用户展示了手机蓝牙的新奇功能。总体来讲,本次毕设是成功的。软件总体特点表现在以下几点:

(1) 用户仅需要点击按钮就可以完成使用蓝牙功能的操作,不需要再为寻找功能花费时间。

(2) 软件中加入了返回按钮,不再需要点击安卓手机返回按键。

(3) 安卓手机中虽然自带蓝牙共享网络功能,但很多用户不知道这个功能;软件中的蓝牙聊天功能是安卓手机所不具备的,手机蓝牙的免费特性使得中程交流不再需要支付高昂的流量费用。

(4) 界面简单易用。

(5) 汇总了大多数用户所不了解的蓝牙共享网络功能。

### 5.2 软件缺陷

蓝牙软件经过三个月的努力基本上达到了设计要求,但由于时间紧迫和个人水平等问题,软件还是存在很多缺陷,有很多问题需要改进,具体问题如下:

(1) 需求分析和系统设计基本完成,但是因时间紧迫,设计阶段不够仔细,导致编码过程中思路紊乱,以后需改进。

(2) 需求分析中的功能基本实现。但是存在界面不美观的缺陷。

(3) 软件不具备强鲁棒性,需按照特定流程执行,不然会出现错误

(4) 测试过程未完成计划,蓝牙软件需真机调试,因水平有限,无法在单元测试阶段设计驱动以及数据桩,整个测试过程偏向于黑盒测试,未满足测试的生命周期,不符合敏捷测试的测试优先思想,在未来工作中,需努力提升并改进。

总之,本次毕业设计总体成功,但还有瑕疵。

### 5.3 今后进一步改进计划

(1) 优化界面。成功的软件要有优美的界面，这样才能给用户良好的用户体验。

(2) 满足更多当下的蓝牙外设需求，如蓝牙打印机与安卓手机的连接需求，蓝牙手表以及蓝牙手环的连接需求。

(3) 实现蓝牙的自动连接。



## 参考文献

- [1]. 钱志鸿, 杨帆, 周求湛 蓝牙技术原理开发与应用 [M]. 北京: 北京航空航天大学出版社, 2006
- [2]. 马建仓等编著 蓝牙核心技术及应用 [M]. 北京: 科学出版社, 2001
- [3]. 金纯等编著 蓝牙技术 电子工业出版社 [M]. 北京: 电子工业出版社, 2001
- [4]. 张禄林等编著 蓝牙协议及其实现 [M]. 北京: 人民邮电出版社, 2001
- [5]. 周正译 蓝牙揭秘 [M]. 北京: 人民邮电出版社, 2003
- [6]. Jaap C. Haartsen, Ericsson Radio Systems B.V., “The Bluetooth Radio System”, IEEE Personal Communications, February 2000
- [7]. Per Johansson, Ericsson Research Manthos Kazantzidis, Rohit Kapoor, and Mario Gerla, UCLA, “Bluetooth: An Enabler for Personal Area Networking”, IEEE Network, September/October 2001
- [8]. Charlie Kaufman, Radia Perlman, Mike Speciner, “Network Security: PRIVATE Communication in a PUBLIC World, 2nd Edition”
- [9]. Matt Ziegler, “An Overview of Bluetooth: Architecture, Power Consumption and Performance” [http://www.ece.virginia.edu/~mmz4s/papers/ECE613project\\_bluetooth.pdf](http://www.ece.virginia.edu/~mmz4s/papers/ECE613project_bluetooth.pdf)
- [10]. John R. Hayes, “Modular Programming in C”,  
<http://www.electronicengineering.com/features/software/OEG20011129S0051>
- [11]. Crystal Sloan, “State Diagram Tips”, [http://www.ertn.com/pr\\_state\\_diagrams.html](http://www.ertn.com/pr_state_diagrams.html)

## 致 谢

为期四年的大学本科生活即将结束，在这里，我想对四年以来对我有过帮助的老师 and 同学说声感谢，没有他们，就没有我今天的这样的成果。这篇论文历时三个月终于完成，在论文写作过程中遇到了很多障碍，但这些障碍都在老师和同学们的帮助下解决了。

这次毕业设计能够成功完成，尤其要特别感谢我的指导老师xx老师，老师在软件实现和论文写作过程中为我提供了很多方向性的指导 and 建设性的意见，使我的论文能够及时完成。

其次，我要感谢 TCL 惠州研发中心和公司导师软件测试工程师 xxx 师傅，感谢公司在软件实现阶段提供测试所需手机和各种条件，感谢师傅在毕业设计阶段提供的技术指导，本次毕设的完成，与这些支持是分不开的。

最后，我要感谢 xx 大学软件学院，自从我 11 年来到学院，我就被学院“IT 精英，傲视群雄”的口号所感染，四年以来一直以这句口号为目标而努力。软件学院的全体老师和同学在这四年里给予我的帮助，使得我成功的成为了一名合格的职业人，这才有今天这份成功的毕业设计。

在此预祝 xx 大学软件学院蒸蒸日上、再创辉煌！祝各位老师，身体健康、万事如意！祝 xx 大学越办越好。