### 代码统计：

1. 工程结构截图：

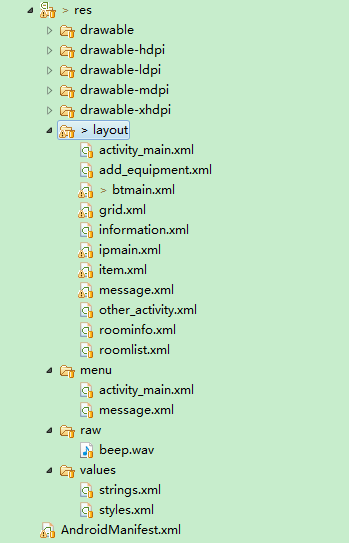
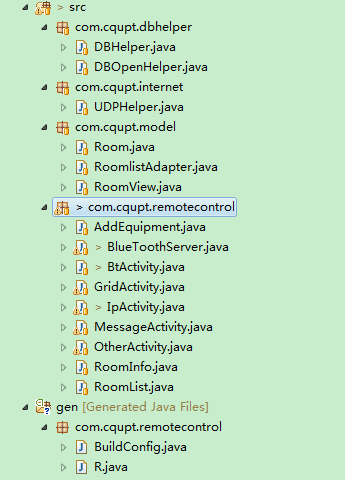
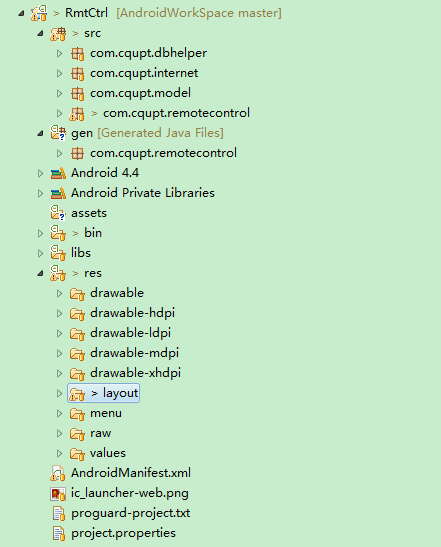


图1 工程结构图 图2 java代码结构图 图3 布局文件结构图

1. java代码统计：

（1）Com.cqupt.dbhelper： DBHelper.java

package com.cqupt.dbhelper;

import java.util.LinkedList;

import java.util.List;

import com.cqupt.model.Room;

import android.content.ContentValues;

import android.content.Context;

import android.database.Cursor;

import android.database.sqlite.SQLiteDatabase;

/\*\*

\* 功能：实现Room信息数据库增删改查

\* @author LS

\*

\*/

public class DBHelper{

private static DBOpenHelper dbOpenHelper;

public DBHelper(Context context){

dbOpenHelper = new DBOpenHelper(context, "room\_data\_db");

}

public Long insert(String name, String ip, String port){

ContentValues values = new ContentValues();

// 向该对象中插入键值对，其中键是列名，值是希望插入到这一列的值，值必须和数据库当中的数据类型一致

values.put("name", name);

values.put("ip", ip);

values.put("port", port);

SQLiteDatabase sqliteDatabase = dbOpenHelper.getWritableDatabase();

// 调用insert方法，就可以将数据插入到数据库当中

// 第一个参数:表名称

// 第二个参数：SQl不允许一个空列，如果ContentValues是空的，那么这一列被明确的指明为NULL值

// 第三个参数：ContentValues对象

Long backinfo = sqliteDatabase.insert("room", null, values);

sqliteDatabase.close();

return backinfo;

}

public void delete(String name){

SQLiteDatabase sqliteDatabase = dbOpenHelper.getWritableDatabase();

sqliteDatabase.delete("room", "name=?", new String[]{name});

sqliteDatabase.close();

}

public void delete(String name, String ip, String port){

SQLiteDatabase sqliteDatabase = dbOpenHelper.getWritableDatabase();

sqliteDatabase.delete("room", "name=? and ip=? and port=?", new String[]{name,ip,port});

sqliteDatabase.close();

}

//从数据库中查询数据

public List<Room> queryData(){

List<Room> list = new LinkedList<Room>();

//获得数据库对象

SQLiteDatabase sqliteDatabase = dbOpenHelper.getReadableDatabase();

//查询表中的数据

Cursor cursor = sqliteDatabase.query("room", null, null, null, null, null, null);

//获取name列的索引

int nameIndex = cursor.getColumnIndex("name");

//获取level列的索引

int ipIndex = cursor.getColumnIndex("ip");

//获取level列的索引

int portIndex = cursor.getColumnIndex("port");

for (cursor.moveToFirst();!(cursor.isAfterLast());cursor.moveToNext()) {

String name = cursor.getString(nameIndex);

String ip = cursor.getString(ipIndex);

String port = cursor.getString(portIndex);

list.add(new Room(name, ip, port));

}

cursor.close();//关闭结果集

sqliteDatabase.close();//关闭数据库对象

return list;

}

/\*\*

\* 关闭数据库资源

\*/

public void close(){

dbOpenHelper.close();

dbOpenHelper = null;

}

}

（2）Com.cqupt.dbhelper： DBOpenHelper.java

package com.cqupt.dbhelper;

import android.content.Context;

import android.database.sqlite.SQLiteDatabase;

import android.database.sqlite.SQLiteDatabase.CursorFactory;

import android.database.sqlite.SQLiteOpenHelper;

/\*\*.

\* 功能：继承SQLiteOpenHelper，创建初始化数据库

\* @author LS

\*

\*/

public class DBOpenHelper extends SQLiteOpenHelper {

private static final int VERSION = 1;

private static final String CREATE\_SQL =

"create table room" +

"(" +

"id int PRIMARY KEY," +

"name varchar(20) NOT NULL, " +

"ip varchar(20) NOT NULL, " +

"port varchar(6) NOT NULL" +

")";

public DBOpenHelper(Context context, String name, CursorFactory factory, int version) {

super(context, name, factory, version);

// TODO Auto-generated constructor stub

}

public DBOpenHelper(Context context, String name, int version){

this(context,name,null,version);

}

public DBOpenHelper(Context context, String name){

this(context,name,VERSION);

}

/\*\*

\* 创建数据库

\*/

@Override

public void onCreate(SQLiteDatabase db) {

// TODO Auto-generated method stub

System.out.println("create a database");

db.execSQL(CREATE\_SQL);

}

/\*\*

\* 数据库更新方法

\*/

@Override

public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {

// TODO Auto-generated method stub

System.out.println("upgrade a database");

}

}

（3）com.cqupt.internet：UDPHelper.java

package com.cqupt.internet;

import java.io.IOException;

import java.net.DatagramPacket;

import java.net.DatagramSocket;

import java.net.InetSocketAddress;

import java.net.SocketException;

/\*\*

\* 功能：实现基于ＵＤＰ协议的信息发送与接受

\* @author LS

\*

\*/

public class UDPHelper {

public static final int PORT = 20000;

public DatagramSocket ds;

public UDPHelper(){

if(ds == null){

try {

ds = new DatagramSocket(PORT);

} catch (SocketException e) {

e.printStackTrace();

}

}

}

/\*\*

\* 发送消息

\* @param ip

\* @param port

\* @param info

\*/

public void sendData(String ip, int port, String info){

byte[] data = info.getBytes();

try {

DatagramPacket dp = new DatagramPacket(data, data.length, new InetSocketAddress(ip, port));

ds.send(dp);

} catch (Exception e) {

e.printStackTrace();

}

}

/\*\*

\* 接收消息

\* @return

\*/

public String receive(){

byte[] data = new byte[1024];

DatagramPacket dp = new DatagramPacket(data, data.length);

try {

ds.receive(dp);

} catch (IOException e) {

e.printStackTrace();

return null;

}

String info = new String(dp.getData());

return info.trim();

}

/\*\*

\* 关闭资源

\*/

public void close(){

if(ds != null){

ds.close();

ds = null;

}

}

}

（4）com.cqupt.model: Room.java

package com.cqupt.model;

/\*\*

\* 功能：房间信息类

\* @author LS

\*

\*/

public class Room {

private String roomName;

private String roomIp;

private String roomPort;

private String analog\_1 = "analog\_1";

private String analog\_2 = "analog\_2";

private String analog\_3 = "analog\_3";

private String analog\_4 = "analog\_4";

private String digital\_1 = "digital\_1";

private String digital\_2 = "digital\_2";

private String digital\_3 = "digital\_3";

private String digital\_4 = "digital\_4";

public Room(String roomName, String roomIp, String roomPort) {

this.roomName = roomName;

this.roomIp = roomIp;

this.roomPort = roomPort;

}

public String getRoomName() {

return roomName;

}

public void setRoomName(String roomName) {

this.roomName = roomName;

}

public String getRoomIp() {

return roomIp;

}

public void setRoomIp(String roomIp) {

this.roomIp = roomIp;

}

public String getRoomPort() {

return roomPort;

}

public void setRoomPort(String roomPort) {

this.roomPort = roomPort;

}

public String getAnalog\_1() {

return analog\_1;

}

public void setAnalog\_1(String analog\_1) {

this.analog\_1 = analog\_1;

}

public String getAnalog\_2() {

return analog\_2;

}

public void setAnalog\_2(String analog\_2) {

this.analog\_2 = analog\_2;

}

public String getAnalog\_3() {

return analog\_3;

}

public void setAnalog\_3(String analog\_3) {

this.analog\_3 = analog\_3;

}

public String getAnalog\_4() {

return analog\_4;

}

public void setAnalog\_4(String analog\_4) {

this.analog\_4 = analog\_4;

}

public String getDigital\_1() {

return digital\_1;

}

public void setDigital\_1(String digital\_1) {

this.digital\_1 = digital\_1;

}

public String getDigital\_2() {

return digital\_2;

}

public void setDigital\_2(String digital\_2) {

this.digital\_2 = digital\_2;

}

public String getDigital\_3() {

return digital\_3;

}

public void setDigital\_3(String digital\_3) {

this.digital\_3 = digital\_3;

}

public String getDigital\_4() {

return digital\_4;

}

public void setDigital\_4(String digital\_4) {

this.digital\_4 = digital\_4;

}

}

（5）com.cqupt.model：RoomlistAdapter.java

package com.cqupt.model;

import java.util.List;

import android.app.Activity;

import android.content.Context;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.ArrayAdapter;

/\*\*

\* 功能：ListView适配器，继承自ArrayAdapter

\* @author LS

\*

\*/

public class RoomlistAdapter extends ArrayAdapter<Room>{

public RoomlistAdapter(Context context, List<Room> roomlist) {

super(context, 0, roomlist);

}

@Override

public View getView(int position, View convertView, ViewGroup parent) {

// TODO Auto-generated method stub

Activity activity = (Activity) getContext();

View rowView = convertView;

RoomView roomView;

if (rowView == null) {

LayoutInflater inflater = activity.getLayoutInflater();

rowView = inflater.inflate(com.cqupt.remotecontrol.R.layout.roomlist, null);

roomView = new RoomView(rowView);

rowView.setTag(roomView);

} else {

roomView = (RoomView)rowView.getTag();

}

final Room room = getItem(position);

roomView.setRoomView(room);

return rowView;

}

}

（6）com.cqupt.mode：RoomView.java

package com.cqupt.model;

import android.view.View;

import android.widget.TextView;

/\*\*

\* 功能：显示房间具体信息

\* @author LS

\*

\*/

public class RoomView {

private View baseView;

private TextView room;

private TextView ip;

public RoomView(View baseView) {

this.baseView = baseView;

}

public void setRoomView(Room room){

getRoomName().setText(room.getRoomName());

getRoomIP().setText(room.getRoomIp());

}

public TextView getRoomName() {

if(room == null){

room = (TextView) baseView.findViewById(com.cqupt.remotecontrol.R.id.roomName);

}

return room;

}

public TextView getRoomIP() {

if(ip == null){

ip = (TextView) baseView.findViewById(com.cqupt.remotecontrol.R.id.roomIP);

}

return ip;

}

}

（7）com.cqupt.remotecontrol：RoomView.java

package com.cqupt.remotecontrol;

import com.cqupt.dbhelper.DBHelper;

import android.app.Activity;

import android.content.Intent;

import android.os.Bundle;

import android.os.Handler;

import android.os.Message;

import android.view.View;

import android.view.Window;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Toast;

/\*\*

\* 功能：添加房间信息Activity

\* @author LS

\*

\*/

public class AddEquipment extends Activity {

private EditText nameEditText;

private EditText ipEditText;

private EditText portEditText;

private Button addButton;

private Handler handler\_show;

private String roomname;

private String roomip;

private String roomport;

@Override

protected void onCreate(Bundle savedInstanceState) {

// TODO Auto-generated method stub

super.onCreate(savedInstanceState);

requestWindowFeature(Window.FEATURE\_NO\_TITLE);//去掉标题栏

setContentView(R.layout.add\_equipment);

Intent intent = getIntent();

Bundle infoBundle = intent.getExtras();

if(infoBundle != null){

handler\_show = new Handler(){

@Override

public void handleMessage(Message msg) {

// TODO Auto-generated method stub

Toast.makeText(AddEquipment.this, (String)msg.obj, Toast.LENGTH\_SHORT).show();

}

};

nameEditText = (EditText)findViewById(R.id.add\_name);

ipEditText = (EditText)findViewById(R.id.add\_ip);

portEditText = (EditText)findViewById(R.id.add\_port);

addButton = (Button)findViewById(R.id.add\_button);

roomname = infoBundle.getString("name");

roomip = infoBundle.getString("ip");

roomport = infoBundle.getString("port");

nameEditText.setText(roomname);

ipEditText.setText(roomip);

portEditText.setText(roomport);

addButton.setText("修改");

addButton.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

// TODO Auto-generated method stub

String name = nameEditText.getText().toString().trim();

String ip = ipEditText.getText().toString().trim();

String port = portEditText.getText().toString().trim();

if(name.equals("")){

show("设备名不能为空");

nameEditText.requestFocus();

return ;

}

if(ip.equals("")){

show("IP地址不能为空");

ipEditText.requestFocus();

return ;

}

if(port.equals("")){

show("端口号不能为空");

portEditText.requestFocus();

return ;

}

if(name.equals(roomname) && ip.equals(roomip) && port.equals(roomport)){

show("没有修改");

return ;

}

DBHelper dbHelper = new DBHelper(AddEquipment.this);

Long backinfo = dbHelper.insert(name, ip, port);

if(backinfo == -1){

show("修改失败");

dbHelper.close();

return ;

}

dbHelper.delete(roomname,roomip,roomport);

dbHelper.close();

show("修改成功");

AddEquipment.this.finish();

}

});

}else{

init();

}

}

public void init(){

handler\_show = new Handler(){

@Override

public void handleMessage(Message msg) {

// TODO Auto-generated method stub

Toast.makeText(AddEquipment.this, (String)msg.obj, Toast.LENGTH\_SHORT).show();

}

};

nameEditText = (EditText)findViewById(R.id.add\_name);

ipEditText = (EditText)findViewById(R.id.add\_ip);

portEditText = (EditText)findViewById(R.id.add\_port);

addButton = (Button)findViewById(R.id.add\_button);

addButton.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

// TODO Auto-generated method stub

String name = nameEditText.getText().toString().trim();

String ip = ipEditText.getText().toString().trim();

String port = portEditText.getText().toString().trim();

if(name.equals("")){

show("设备名不能为空");

nameEditText.requestFocus();

return ;

}

if(ip.equals("")){

show("IP地址不能为空");

ipEditText.requestFocus();

return ;

}

if(port.equals("")){

show("端口号不能为空");

portEditText.requestFocus();

return ;

}

DBHelper dbHelper = new DBHelper(AddEquipment.this);

Long backinfo = dbHelper.insert(name, ip, port);

if(backinfo == -1){

show("添加失败");

return ;

}

dbHelper.close();

show("添加成功");

AddEquipment.this.finish();

}

});

}

public void show(String string){

Message message = new Message();

message.obj = string;

handler\_show.sendMessage(message);

}

}

（8）com.cqupt.remotecontrol： BlueToothServier.java

package com.cqupt.remotecontrol;

import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStream;

import java.io.InputStreamReader;

import java.io.OutputStream;

import java.util.ArrayList;

import java.util.List;

import java.util.UUID;

import android.app.Service;

import android.bluetooth.BluetoothAdapter;

import android.bluetooth.BluetoothDevice;

import android.bluetooth.BluetoothSocket;

import android.content.BroadcastReceiver;

import android.content.Context;

import android.content.Intent;

import android.content.IntentFilter;

import android.os.AsyncTask;

import android.os.IBinder;

/\*\*

\* 功能：蓝牙连接后台服务类

\* 1.实现蓝牙设备搜索

\* 2.蓝牙设备配对

\* 3.蓝牙间消息发送与接受

\* @author LS

\*

\*/

public class BlueToothServer extends Service {

public List<BluetoothDevice> deviceList;

public boolean DISCOVERY\_OVER = false;

public ArrayList<String> infor;

private static final UUID MY\_UUID = UUID

.fromString("00001101-0000-1000-8000-00805F9B34FB");

public static final String CONNECT = "com.example.action.connect";// 连接蓝牙消息

public static final String DEVICE\_LIST = "com.example.action.device\_list";// 反馈搜到的蓝牙设备

public static final String LIGHT\_ONE\_OPEN = "com.example.action.light\_one\_open";// 第一盏灯亮

public static final String LIGHT\_ONE\_CLOSE = "com.example.action.light\_one\_close";// 第一盏灯亮

public static final String LIGHT\_TWO\_OPEN = "com.example.action.light\_tow\_open";// 第一盏灯亮

public static final String LIGHT\_TWO\_CLOSE = "com.example.action.light\_two\_close";// 第一盏灯亮

public static final String LIGHT\_THREE\_OPEN = "com.example.action.light\_three\_open";// 第一盏灯亮

public static final String LIGHT\_THREE\_CLOSE = "com.example.action.light\_three\_close";// 第一盏灯亮

public static final String LIGHT\_FOUR\_OPEN = "com.example.action.light\_four\_open";// 第一盏灯亮

public static final String LIGHT\_FOUR\_CLOSE = "com.example.action.light\_four\_close";// 第一盏灯亮

public static final String WORK\_STATE = "com.example.action.work\_state";

public static final String REFRESH = "com.example.action.refresh";// 刷新

public static final String CUT\_CONNECT\_BT = "com.example.action.cut\_connect\_bt";// 断开蓝牙连接

public BluetoothDevice device;

public BluetoothSocket socket;

public BluetoothAdapter btAdapter;

public Intent sendToOtherAc;

public Intent sendToBtMain;

public int position;

public OutputStream outStream;

public byte[] myByte;

public InputStream inPutStream;

public BtDiscovery discovery;

public StringBuffer buffer;

public ControlReceiver sr;

public String sendInfor;

byte b[] = new byte[4];

@Override

public void onCreate() {

// TODO Auto-generated method stub

super.onCreate();

btAdapter = BluetoothAdapter.getDefaultAdapter();

btAdapter.startDiscovery();

deviceList = new ArrayList<BluetoothDevice>();

infor = new ArrayList<String>();

myByte = new byte[1024];

sendToBtMain = new Intent();

sendToOtherAc = new Intent();

sendToOtherAc.setAction(DEVICE\_LIST);

position = -1;

SetupBtDiscoverReceiver();

setUpControlReceiver();

}

/\*\*

\* 注册蓝牙设备信息接受广播

\*/

private void SetupBtDiscoverReceiver() {

// TODO Auto-generated method stub

// 注册系统广播接受蓝牙设备

discovery = new BtDiscovery();

IntentFilter filter = new IntentFilter(BluetoothDevice.ACTION\_FOUND);

registerReceiver(discovery, filter);

filter = new IntentFilter(BluetoothAdapter.ACTION\_DISCOVERY\_FINISHED);

registerReceiver(discovery, filter);

}

/\*\*

\* 注册接受指令广播

\*/

private void setUpControlReceiver() {

// TODO Auto-generated method stub

// /注册接受指令广播

sr = new ControlReceiver();

IntentFilter filter2 = new IntentFilter();

filter2.addAction(LIGHT\_ONE\_OPEN);

filter2.addAction(LIGHT\_ONE\_CLOSE);

filter2.addAction(LIGHT\_TWO\_OPEN);

filter2.addAction(LIGHT\_TWO\_CLOSE);

filter2.addAction(LIGHT\_THREE\_OPEN);

filter2.addAction(LIGHT\_THREE\_CLOSE);

filter2.addAction(LIGHT\_FOUR\_OPEN);

filter2.addAction(LIGHT\_FOUR\_CLOSE);

filter2.addAction(REFRESH);// 刷新操作

filter2.addAction(CONNECT);

filter2.addAction(CUT\_CONNECT\_BT);

registerReceiver(sr, filter2);

}

@Override

public void onDestroy() {

// TODO Auto-generated method stub

super.onDestroy();

cutConnect();

unregisterReceiver(discovery);

unregisterReceiver(sr);

}

@Override

public int onStartCommand(Intent intent, int flags, int startId) {

// TODO Auto-generated method stub

System.out.println("BTService開始搜索");

new Work().execute(DISCOVERY\_OVER);// 開始搜索

return super.onStartCommand(intent, flags, startId);

}

@Override

public IBinder onBind(Intent intent) {

// TODO Auto-generated method stub

return null;

}

/\*\*

\* 异步任务发送所得蓝牙设备信息

\*

\* @author Administrator

\*

\*/

protected class Work extends AsyncTask<Boolean, String, String> {

@Override

protected void onPostExecute(String result) {

// TODO Auto-generated method stub

super.onPostExecute(result);

}

@Override

protected void onPreExecute() {

// TODO Auto-generated method stub

super.onPreExecute();

}

@Override

protected void onProgressUpdate(String... values) {

// TODO Auto-generated method stub

super.onProgressUpdate(values);

}

@Override

protected String doInBackground(Boolean... params) {

// TODO Auto-generated method stub

while (true) {

if (DISCOVERY\_OVER) {// 搜索結束

System.out.println("搜索結束");

sendToOtherAc.putStringArrayListExtra("device",

turnString(deviceList));// 获得发送列表条目

sendBroadcast(sendToOtherAc);

break;

}

}

return null;

}

}

/\*\*

\* 接受蓝牙设备信息

\*

\* @author Administrator

\*

\*/

class BtDiscovery extends BroadcastReceiver {

@Override

public void onReceive(Context context, Intent intent) {

// TODO Auto-generated method stub

String action = intent.getAction();

if (BluetoothDevice.ACTION\_FOUND.equals(action)) {

device = intent

.getParcelableExtra(BluetoothDevice.EXTRA\_DEVICE);

deviceList.add(device);// 加入驱动表中

System.out.println("Discovery\_\_ing" + device.getName() + "/n"

+ device.getAddress());

// 发送Device名称

} else if (BluetoothAdapter.ACTION\_DISCOVERY\_FINISHED

.equals(action)) {

DISCOVERY\_OVER = true;

}

}

}

/\*\*

\* 将驱动列表包装

\*

\* @param deviceList2

\* @return

\*/

public ArrayList<String> turnString(List<BluetoothDevice> deviceList2) {

// TODO Auto-generated method stub

String[] infor1 = new String[deviceList2.size()];

for (int i = 0; i < deviceList2.size(); i++) {

if (deviceList2.get(i).getName() == null) {// 设备名为空处理

System.out.println("deiceList name is null");

infor1[i] = "null";

} else

infor1[i] = deviceList2.get(i).getName();

if (infor.indexOf(infor1[i]) == -1) {

infor.add(infor1[i]);

}

}

return infor;

}

/\*\*

\* 获取Device连接

\*

\* @param position

\*/

protected void connectBlue(int position) {

// TODO Auto-generated method stub

// device = deviceList.get(position);

device = btAdapter.getRemoteDevice(deviceList.get(position)

.getAddress());

System.out.println("BeforeContect~~~"

+ deviceList.get(position).getAddress());

new Connect().start();// 连接线程

}

/\*\*

\* 连接蓝牙操作线程

\*

\* @author Administrator

\*

\*/

class Connect extends Thread {

// private BluetoothSocket tmp;

@Override

public void run() {

// TODO Auto-generated method stub

System.out.println("BeforeContect~~~");

btAdapter.cancelDiscovery();

try {

socket = device.createRfcommSocketToServiceRecord(MY\_UUID);// 获取Socket

socket.connect();

outStream = socket.getOutputStream();

inPutStream = socket.getInputStream();

new JieShou().start();

System.out.println("Contect~~~");

} catch (IOException e) {

e.printStackTrace();

System.out.println("ContectError~~~");

try {

socket.close();

outStream.flush();

outStream.close();

inPutStream.close();

} catch (IOException e1) {

// TODO Auto-generated catch block

e1.printStackTrace();

}

}

super.run();

}

}

/\*\*

\* 信息接收线程

\* @author LS

\*

\*/

class JieShou extends Thread {

@Override

public void run() {

// TODO Auto-generated method stub

super.run();

while (true) {

int j;

try {

j = inPutStream.read(b);

if (j > 0) {

sendWorkStateToBt(b);

}

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

}

}

/\*\*

\* 连接蓝牙后读取操作 一写一读

\*

\* @author Administrator

\*

\*/

class Oper extends Thread {

String infor = "";

public Oper(String infor) {

this.infor = infor;

System.out.println(infor);

}

@Override

public void run() {

// TODO Auto-generated method stub

String line = "";

String backInfor = "";

try {

if (infor != "")

outStream.write(infor.getBytes());// 发送

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

super.run();

}

}

/\*\*

\* 响应用户操作

\*

\* @author Administrator

\*

\*/

class ControlReceiver extends BroadcastReceiver {

@Override

public void onReceive(Context arg0, Intent intent) {

// TODO Auto-generated method stub

String action = intent.getAction();

if (action.equals(CONNECT)) {// 连接蓝牙设备

position = intent.getIntExtra("position", -1);

if (position > -1) {

connectBlue(position);// 连接

}

} else if (action.equals(CUT\_CONNECT\_BT)) {// 断开连接

System.out.println(CUT\_CONNECT\_BT + "asdasd");

cutConnect();

} else if (action.equals(LIGHT\_ONE\_OPEN)) {

sendInfor = "m1";

System.out.println("打开灯1" + sendInfor);

new Oper(sendInfor).start();

} else if (action.equals(LIGHT\_ONE\_CLOSE)) {

sendInfor = "m5";

new Oper(sendInfor).start();

} else if (action.equals(LIGHT\_TWO\_OPEN)) {

sendInfor = "m2";

new Oper(sendInfor).start();

} else if (action.equals(LIGHT\_TWO\_CLOSE)) {

sendInfor = "m6";

new Oper(sendInfor).start();

} else if (action.equals(LIGHT\_THREE\_OPEN)) {

sendInfor = "m3";

new Oper(sendInfor).start();

} else if (action.equals(LIGHT\_THREE\_CLOSE)) {

sendInfor = "m7";

new Oper(sendInfor).start();

} else if (action.equals(LIGHT\_FOUR\_OPEN)) {

sendInfor = "m4";

new Oper(sendInfor).start();

} else if (action.equals(LIGHT\_FOUR\_CLOSE)) {

sendInfor = "m8";

new Oper(sendInfor).start();

} else if (action.equals(REFRESH)) {

sendInfor = "m9";

new Oper(sendInfor).start();

}

}

}

/\*\*

\* 发送反馈显示状态

\*

\* @param string

\*/

public void sendWorkStateToBt(byte[] b) {

// TODO Auto-generated method stub

sendToBtMain.setAction(WORK\_STATE);

sendToBtMain.putExtra("lights\_is\_open", b);

sendBroadcast(sendToBtMain);

}

/\*\*

\* 结束蓝牙连接，释放相关资源

\*/

public void cutConnect() {

// TODO Auto-generated method stub

if (socket != null && outStream != null && inPutStream != null) {

try {

socket.close();

outStream.flush();

inPutStream.close();

outStream.close();

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

}

}

（9）com.cqupt.remotecontrol：BtActivity.java

package com.cqupt.remotecontrol;

import android.app.Activity;

import android.content.BroadcastReceiver;

import android.content.Context;

import android.content.Intent;

import android.content.IntentFilter;

import android.os.Bundle;

import android.view.View;

import android.view.View.OnClickListener;

import android.widget.Button;

import android.widget.ImageView;

/\*\*

\* 功能：蓝牙控制主界面Activity

\* @author LS

\*

\*/

public class BtActivity extends Activity implements OnClickListener {

public static final String LIGHT\_ONE\_OPEN = "com.example.action.light\_one\_open";// 第一盏灯亮

public static final String LIGHT\_ONE\_CLOSE = "com.example.action.light\_one\_close";// 第一盏灯亮

public static final String LIGHT\_TWO\_OPEN = "com.example.action.light\_tow\_open";// 第一盏灯亮

public static final String LIGHT\_TWO\_CLOSE = "com.example.action.light\_two\_close";// 第一盏灯亮

public static final String LIGHT\_THREE\_OPEN = "com.example.action.light\_three\_open";// 第一盏灯亮

public static final String LIGHT\_THREE\_CLOSE = "com.example.action.light\_three\_close";// 第一盏灯亮

public static final String LIGHT\_FOUR\_OPEN = "com.example.action.light\_four\_open";// 第一盏灯亮

public static final String LIGHT\_FOUR\_CLOSE = "com.example.action.light\_four\_close";// 第一盏灯亮

public static final String WORK\_STATE = "com.example.action.work\_state";

public static final String REFRESH = "com.example.action.refresh";

public static final String CUT\_CONNECT\_BT = "com.example.action.cut\_connect\_bt";// 断开蓝牙连接

WorkStatReceiver receiver;

private ImageView lightOne;

private ImageView lightTwo;

private ImageView lightThree;

private ImageView lightFour;

private ImageView[] vi = new ImageView[4];

private Intent sendToServiceContr0l;

@Override

protected void onCreate(Bundle savedInstanceState) {

// TODO Auto-generated method stub

super.onCreate(savedInstanceState);

setContentView(R.layout.btmain);

setUpReceiver();

setUpViews();

}

private void setUpViews() {

// TODO Auto-generated method stub

((Button) findViewById(R.id.btAc\_btn1\_open)).setOnClickListener(this);

((Button) findViewById(R.id.btAc\_btn1\_close)).setOnClickListener(this);

((Button) findViewById(R.id.btAc\_btn2\_open)).setOnClickListener(this);

((Button) findViewById(R.id.btAc\_btn2\_close)).setOnClickListener(this);

((Button) findViewById(R.id.btAc\_btn3\_open)).setOnClickListener(this);

((Button) findViewById(R.id.btAc\_btn3\_close)).setOnClickListener(this);

((Button) findViewById(R.id.btAc\_btn4\_open)).setOnClickListener(this);

((Button) findViewById(R.id.btAc\_btn4\_close)).setOnClickListener(this);

((Button) findViewById(R.id.btAc\_btn\_refersh)).setOnClickListener(this);

lightOne = (ImageView) findViewById(R.id.btAc\_iv\_1);

lightTwo = (ImageView) findViewById(R.id.btAc\_iv\_2);

lightThree = (ImageView) findViewById(R.id.btAc\_iv\_3);

lightFour = (ImageView) findViewById(R.id.btAc\_iv\_4);

vi[0] = lightOne;

vi[1] = lightTwo;

vi[2] = lightThree;

vi[3] = lightFour;

}

private void setUpReceiver() {

// TODO Auto-generated method stub

sendToServiceContr0l = new Intent();

receiver = new WorkStatReceiver();

IntentFilter filter = new IntentFilter();

filter.addAction(WORK\_STATE);

registerReceiver(receiver, filter);

}

@Override

protected void onDestroy() {

// TODO Auto-generated method stub

unregisterReceiver(receiver);

super.onDestroy();

}

/\*\*

\* 按键监听发送指令广播

\*/

@Override

public void onClick(View v) {

// TODO Auto-generated method stub

switch (v.getId()) {

case R.id.btAc\_btn1\_open:// / 1 open

sendToServiceContr0l.setAction(LIGHT\_ONE\_OPEN);

lightOne.setBackgroundResource(R.drawable.openlight);

sendBroadcast(sendToServiceContr0l);

break;

case R.id.btAc\_btn1\_close:

sendToServiceContr0l.setAction(LIGHT\_ONE\_CLOSE);// 1 close

lightOne.setBackgroundResource(R.drawable.closelight);

sendBroadcast(sendToServiceContr0l);

break;

case R.id.btAc\_btn2\_open:

sendToServiceContr0l.setAction(LIGHT\_TWO\_OPEN);

lightTwo.setBackgroundResource(R.drawable.openlight);

// lightThree.setBackgroundResource(R.drawable.closelight);

sendBroadcast(sendToServiceContr0l);

break;

case R.id.btAc\_btn2\_close:

sendToServiceContr0l.setAction(LIGHT\_TWO\_CLOSE);

lightTwo.setBackgroundResource(R.drawable.closelight);

sendBroadcast(sendToServiceContr0l);

break;

case R.id.btAc\_btn3\_open:

sendToServiceContr0l.setAction(LIGHT\_THREE\_OPEN);

lightThree.setBackgroundResource(R.drawable.openlight);

// lightTwo.setBackgroundResource(R.drawable.closelight);

sendBroadcast(sendToServiceContr0l);

break;

case R.id.btAc\_btn3\_close:

sendToServiceContr0l.setAction(LIGHT\_THREE\_CLOSE);

lightThree.setBackgroundResource(R.drawable.closelight);

sendBroadcast(sendToServiceContr0l);

break;

case R.id.btAc\_btn4\_open:

sendToServiceContr0l.setAction(LIGHT\_FOUR\_OPEN);

lightFour.setBackgroundResource(R.drawable.openlight);

sendBroadcast(sendToServiceContr0l);

break;

case R.id.btAc\_btn4\_close:

sendToServiceContr0l.setAction(LIGHT\_FOUR\_CLOSE);

lightFour.setBackgroundResource(R.drawable.closelight);

sendBroadcast(sendToServiceContr0l);

break;

case R.id.btAc\_btn\_refersh:

sendToServiceContr0l.setAction(REFRESH);

sendBroadcast(sendToServiceContr0l);

break;

}

}

class WorkStatReceiver extends BroadcastReceiver {

@Override

public void onReceive(Context context, Intent intent) {

// TODO Auto-generated method stub

String action = intent.getAction();

byte[] b;

if (action.equals(WORK\_STATE)) {

b = intent.getByteArrayExtra("lights\_is\_open");

System.out.println("接收到的Wifi--->" + new String(b, 0, 9)

+ "字节数组长度：--》" + b.length);

for (int i = 0; i < 4; i++) {

if (new String(b, i, 1).equals("1")) {

vi[i].setBackgroundResource(R.drawable.openlight);

System.out.println("light open-->" + i);

} else

vi[i].setBackgroundResource(R.drawable.closeip);

}

}

}

}

}

（10）com.cqupt.remotecontrol：GridActivity.java

package com.cqupt.remotecontrol;

import java.util.ArrayList;

import java.util.HashMap;

import android.app.Activity;

import android.app.AlertDialog;

import android.app.AlertDialog.Builder;

import android.bluetooth.BluetoothAdapter;

import android.content.Context;

import android.content.DialogInterface;

import android.content.Intent;

import android.os.Bundle;

import android.os.Handler;

import android.os.Message;

import android.view.View;

import android.view.Window;

import android.widget.AdapterView;

import android.widget.AdapterView.OnItemClickListener;

import android.widget.GridView;

import android.widget.SimpleAdapter;

import android.widget.Toast;

/\*\*

\* 功能：程序主界面

\* @author LS

\*

\*/

public class GridActivity extends Activity {

public BluetoothAdapter adapter;

private GridView gridview;

public Context cxt;

private boolean isOpenBtPermission = false;

public static final int REQUEST\_DISCOVERABLE = 1;

public static final String CUT\_CONNECT\_BT = "com.example.action.cut\_connect\_bt";// 断开蓝牙连接

public Handler handler = new Handler() {

@Override

public void handleMessage(Message msg) {

// TODO Auto-generated method stub

String string = (String) msg.obj;

if (string.equals("RoomInfo")) {

Intent intent = new Intent();

intent.setClass(cxt, RoomInfo.class);

startActivity(intent);

} else if (string.equals("AddEquipment")) {

Intent intent = new Intent();

intent.setClass(cxt, AddEquipment.class);

startActivity(intent);

} else if (string.equals("RoomList")) {

Intent intent = new Intent();

intent.setClass(cxt, RoomList.class);

startActivity(intent);

} else if (string.equals("MessageActivity")) {

Intent intent = new Intent();

intent.setClass(cxt, MessageActivity.class);

startActivity(intent);

}

}

};

public void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

requestWindowFeature(Window.FEATURE\_NO\_TITLE);// 去掉标题栏

setContentView(R.layout.grid);

cxt = this;

gridview = (GridView) findViewById(R.id.gridview);

adapter = BluetoothAdapter.getDefaultAdapter();

// 生成动态数组，并且转入数据

ArrayList<HashMap<String, Object>> lstImageItem = new ArrayList<HashMap<String, Object>>();

HashMap<String, Object> map1 = new HashMap<String, Object>();

map1.put("ItemImage", R.drawable.p1);

map1.put("ItemText", "IP模式监控");

lstImageItem.add(map1);

HashMap<String, Object> map2 = new HashMap<String, Object>();

map2.put("ItemImage", R.drawable.p2);

map2.put("ItemText", "IP模式设置");

lstImageItem.add(map2);

HashMap<String, Object> map3 = new HashMap<String, Object>();

map3.put("ItemImage", R.drawable.p3);

map3.put("ItemText", "添加IP房间");

lstImageItem.add(map3);

HashMap<String, Object> map4 = new HashMap<String, Object>();

map4.put("ItemImage", R.drawable.p4);

map4.put("ItemText", "蓝牙模式监控");

lstImageItem.add(map4);

HashMap<String, Object> map5 = new HashMap<String, Object>();

map5.put("ItemImage", R.drawable.p5);

map5.put("ItemText", "蓝牙模式打开");

lstImageItem.add(map5);

HashMap<String, Object> map6 = new HashMap<String, Object>();

map6.put("ItemImage", R.drawable.p6);

map6.put("ItemText", "蓝牙模式关闭");

lstImageItem.add(map6);

HashMap<String, Object> map7 = new HashMap<String, Object>();

map7.put("ItemImage", R.drawable.p7);

map7.put("ItemText", "短信模式监控");

lstImageItem.add(map7);

HashMap<String, Object> map8 = new HashMap<String, Object>();

map8.put("ItemImage", R.drawable.p8);

map8.put("ItemText", "短信模式设置");

lstImageItem.add(map8);

HashMap<String, Object> map9 = new HashMap<String, Object>();

map9.put("ItemImage", R.drawable.p9);

map9.put("ItemText", "关闭");

lstImageItem.add(map9);

// 生成适配器的ImageItem <====> 动态数组的元素，两者一一对应

SimpleAdapter saImageItems = new SimpleAdapter(this, // 没什么解释

lstImageItem,// 数据来源

R.layout.item,// night\_item的XML实现

// 动态数组与ImageItem对应的子项

new String[] { "ItemImage", "ItemText" },

// ImageItem的XML文件里面的一个ImageView,两个TextView ID

new int[] { R.id.ItemImage, R.id.ItemText });

// 添加并且显示

gridview.setAdapter(saImageItems);

// 添加消息处理

gridview.setOnItemClickListener(new ItemClickListener());

}

public void exitProgram() {

AlertDialog.Builder builder = new Builder(cxt);

builder.setTitle("提示");

builder.setMessage("你确定要退出吗？");

builder.setPositiveButton("是的",

new android.content.DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface dialog, int which) {

dialog.dismiss();

android.os.Process.killProcess(android.os.Process

.myPid());

}

});

builder.setNegativeButton("取消",

new android.content.DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface dialog, int which) {

dialog.dismiss();

}

});

builder.create().show();

}

// 当AdapterView被单击(触摸屏或者键盘)，则返回的Item单击事件

class ItemClickListener implements OnItemClickListener {

public void onItemClick(AdapterView<?> arg0,// The AdapterView where the

// click happened

View arg1,// The view within the AdapterView that was clicked

int arg2,// The position of the view in the adapter

long arg3// The row id of the item that was clicked

) {

// 在本例中arg2=arg3

@SuppressWarnings("unchecked")

HashMap<String, Object> item = (HashMap<String, Object>) arg0

.getItemAtPosition(arg2);

// 显示所选Item的ItemText

setTitle((String) item.get("ItemText"));

if (item.get("ItemText").equals("IP模式监控")) {

Message s = new Message();

s.obj = "RoomList";

handler.sendMessage(s);

} else if (item.get("ItemText").equals("IP模式设置")) {

Message s = new Message();

s.obj = "RoomList";

handler.sendMessage(s);

} else if (item.get("ItemText").equals("添加IP房间")) {

Message s = new Message();

s.obj = "AddEquipment";

handler.sendMessage(s);

} else if (item.get("ItemText").equals("蓝牙模式监控")) {

// /跳转蓝牙列表界面

if (isOpenBtPermission) {

Intent intent3 = new Intent();

intent3.setClass(GridActivity.this, OtherActivity.class);

startActivity(intent3);

} else

Toast.makeText(GridActivity.this, "未打开蓝牙或者未获取权限！", 0)

.show();

} else if (item.get("ItemText").equals("蓝牙模式打开")) {

//开启蓝牙设备

Intent enabler = new Intent(

BluetoothAdapter.ACTION\_REQUEST\_DISCOVERABLE);

startActivityForResult(enabler, REQUEST\_DISCOVERABLE);

isOpenBtPermission = true;

} else if (item.get("ItemText").equals("蓝牙模式关闭")) {

//关闭蓝牙

Intent sendToService = new Intent();

sendToService.setAction(CUT\_CONNECT\_BT);

sendBroadcast(sendToService);

adapter.disable();

} else if (item.get("ItemText").equals("短信模式监控")) {

Message s = new Message();

s.obj = "MessageActivity";

handler.sendMessage(s);

} else if (item.get("ItemText").equals("短信模式设置")) {

Toast.makeText(GridActivity.this, "暂时没开通该功能", 0).show();

} else if (item.get("ItemText").equals("关闭")) {

exitProgram();

}

}

}

}

（11）com.cqupt.remotecontrol：IpActivity.java

package com.cqupt.remotecontrol;

import java.io.IOException;

import java.net.DatagramPacket;

import java.net.DatagramSocket;

import java.net.InetAddress;

import android.app.Activity;

import android.content.Intent;

import android.os.Bundle;

import android.os.Handler;

import android.os.Message;

import android.view.View;

import android.view.Window;

import android.widget.Button;

import android.widget.EditText;

import android.widget.ImageView;

import android.widget.TextView;

import android.widget.Toast;

/\*\*

\* 功能：ＷＩＦＩ控制主界面Ａｃｔｉｖｉｔｙ

\* @author LS

\*

\*/

public class IpActivity extends Activity {

public static String IP = "192.168.1.1";

public static final int PORT = 8888;

public static final byte LIST = (byte) 0xA1;

public static final byte CTRL = (byte) 0xA2;

public static final byte END = (byte) 0xA3;

public static final byte RSP = (byte) 0xA4;

public static final byte[] KEYS = { (byte) 0xAA, (byte) 0xBB, (byte) 0xCC,

(byte) 0xDD };

public Button b1;

public EditText ipEdit;

public Button bo1;

public Button bo2;

public Button bo3;

public Button bo4;

public Button bc1;

public Button bc2;

public Button bc3;

public Button bc4;

public TextView t1;

public ImageView[] iv = new ImageView[4];

public DatagramSocket ds = null;

public Receive rec;

public Handler handler = new Handler() {

@Override

public void handleMessage(Message msg) {

byte[] b = (byte[]) msg.obj;

System.out.println("接收到的Wifi--->" + new String(b, 0, 4)

+ "字节数组长度：--》" + b.length);

for (int i = 0; i < 4; i++) {

if (new String(b, i, 1).equals("1")) {

iv[i].setImageResource(R.drawable.openlight);

System.out.println("light open-->" + i);

} else

iv[i].setImageResource(R.drawable.closeip);

}

}

};

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

requestWindowFeature(Window.FEATURE\_NO\_TITLE);// 去掉标题栏

setContentView(R.layout.ipmain);

b1 = (Button) findViewById(R.id.button1);

t1 = (TextView) findViewById(R.id.text1);

bo1 = (Button) findViewById(R.id.buttono1);

bo2 = (Button) findViewById(R.id.buttono2);

bo3 = (Button) findViewById(R.id.buttono3);

bo4 = (Button) findViewById(R.id.buttono4);

bc1 = (Button) findViewById(R.id.buttonc1);

bc2 = (Button) findViewById(R.id.buttonc2);

bc3 = (Button) findViewById(R.id.buttonc3);

bc4 = (Button) findViewById(R.id.buttonc4);

iv[0] = (ImageView) findViewById(R.id.image1);

iv[1] = (ImageView) findViewById(R.id.image2);

iv[2] = (ImageView) findViewById(R.id.image3);

iv[3] = (ImageView) findViewById(R.id.image4);

ipEdit = (EditText) findViewById(R.id.ipEdit);

try {

if (ds == null) {

ds = new DatagramSocket(PORT);

}

} catch (Exception e) {

System.out.println(e);

}

if (ds == null) {

Toast.makeText(this, "请重启应用！", Toast.LENGTH\_SHORT).show();

finish();

}

rec = new Receive();

rec.start();

b1.setOnClickListener(new MyButtonListener());

bo1.setOnClickListener(new MyButtonListener());

bo2.setOnClickListener(new MyButtonListener());

bo3.setOnClickListener(new MyButtonListener());

bo4.setOnClickListener(new MyButtonListener());

bc1.setOnClickListener(new MyButtonListener());

bc2.setOnClickListener(new MyButtonListener());

bc3.setOnClickListener(new MyButtonListener());

bc4.setOnClickListener(new MyButtonListener());

Intent intent = getIntent();

Bundle infoBundle = intent.getExtras();

ipEdit.setText(infoBundle.getString("ip"));

}

@Override

protected void onResume() {

// TODO Auto-generated method stub

super.onResume();

try {

if (ds == null) {

ds = new DatagramSocket(PORT);

}

} catch (Exception e) {

System.out.println(e);

}

if (rec == null) {

rec = new Receive();

rec.start();

}

}

@Override

protected void onPause() {

// TODO Auto-generated method stub

super.onPause();

try {

if (ds != null) {

ds.close();

ds = null;

}

} catch (Exception e) {

System.out.println(e);

}

if (rec != null) {

rec.destroy();

rec = null;

}

}

public void changeIP(View v) {

IP = ipEdit.getText().toString();

ipEdit.setEnabled(false);

}

class MyButtonListener implements View.OnClickListener {

@Override

public void onClick(View v) {

byte[] data = null;

String[] sendData = new String[] { "m1", "m2", "m3", "m4", "m5",

"m6", "m7", "m8", "m9" };

switch (v.getId()) {

case R.id.button1:// 刷新

data = sendData[8].getBytes();

break;

case R.id.buttono1:// //1 open

data = sendData[0].getBytes();

iv[0].setImageResource(R.drawable.openlight);

break;

case R.id.buttonc1:// 1 close

data = sendData[4].getBytes();

iv[0].setImageResource(R.drawable.closeip);

break;

case R.id.buttono2:// 2 open

data = sendData[1].getBytes();

iv[1].setImageResource(R.drawable.openlight);

break;

case R.id.buttonc2:

data = sendData[5].getBytes();

iv[1].setImageResource(R.drawable.closeip);

break;

case R.id.buttono3:

data = sendData[2].getBytes();

iv[2].setImageResource(R.drawable.openlight);

break;

case R.id.buttonc3:

data = sendData[6].getBytes();

iv[2].setImageResource(R.drawable.closeip);

break;

case R.id.buttono4:

data = sendData[3].getBytes();

iv[3].setImageResource(R.drawable.openlight);

break;

case R.id.buttonc4:

data = sendData[7].getBytes();

iv[3].setImageResource(R.drawable.closeip);

break;

default:

return;

}

try {

InetAddress addr = InetAddress.getByName(IP);

DatagramPacket dp = new DatagramPacket(data, data.length, addr,

PORT);

ds.send(dp);

} catch (Exception e) {

System.out.println("fail");

}

}

}

public byte[] encrypt(byte[] b) {

int i;

byte[] temp = new byte[b.length];

for (i = 0; i < 3; i++) {

temp[i] = (byte) (b[i] ^ KEYS[i]);

}

temp[i] = b[i];

return temp;

// return b;

}

class Receive extends Thread {

@Override

public void run() {

byte b[] = new byte[1024];

DatagramPacket dp = new DatagramPacket(b, b.length);

while (true) {

try {

ds.receive(dp);

Message msg = new Message();

msg.obj = dp.getData();

handler.sendMessage(msg);

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

}

}

}

（12）com.cqupt.remotecontrol：MessageActivity.java

package com.cqupt.remotecontrol;

import java.util.HashMap;

import android.annotation.SuppressLint;

import android.app.Activity;

import android.app.AlertDialog;

import android.app.AlertDialog.Builder;

import android.content.BroadcastReceiver;

import android.content.Context;

import android.content.DialogInterface;

import android.content.Intent;

import android.content.IntentFilter;

import android.media.AudioManager;

import android.media.SoundPool;

import android.os.Bundle;

import android.telephony.SmsManager;

import android.telephony.SmsMessage;

import android.view.Menu;

import android.view.MenuItem;

import android.view.MotionEvent;

import android.view.View;

import android.view.View.OnClickListener;

import android.widget.Button;

import android.widget.EditText;

import android.widget.ImageButton;

import android.widget.ImageView;

import android.widget.TextView;

import android.widget.Toast;

/\*\*

\* 功能：短信控制主界面Ａｃｔｉｖｉｔｙ

\* @author LS

\*

\*/

public class MessageActivity extends Activity {

public static String PHONENUMBER = "1123";

public static final String COMMAND\_1\_close = "close1";

public static final String COMMAND\_1\_open = "open1";

public static final String COMMAND\_2\_close = "close2";

public static final String COMMAND\_2\_open = "open2";

public static final String COMMAND\_query = "query3";

public static final String COMMAND\_warning = "warning";

public static TextView phoneTextView;

public static ImageButton button1;

private static boolean button1State;

public static ImageButton button2;

private static boolean button2State;

public static Button button3;

public static Button button4;

public static ImageView light1;

public static ImageView light2;

public static ImageView light3;

public static ImageView light4;

public static SoundPool sp;

public static HashMap<Integer, Integer> spMap;

@SuppressLint("UseSparseArrays")

@Override

protected void onCreate(Bundle savedInstanceState) {

// TODO Auto-generated method stub

super.onCreate(savedInstanceState);

this.setContentView(R.layout.message);

sp = new SoundPool(2,AudioManager.STREAM\_MUSIC,0);

spMap = new HashMap<Integer,Integer>();

spMap.put(1, sp.load(this, R.raw.beep, 1));

startListen();

light1 = (ImageView)findViewById(R.id.light1);

light2 = (ImageView)findViewById(R.id.light2);

light3 = (ImageView)findViewById(R.id.light3);

light4 = (ImageView)findViewById(R.id.light4);

phoneTextView = (TextView)findViewById(R.id.phoneNumber);

button1 = (ImageButton) this.findViewById(R.id.button1);

button1State = false;

button1.setOnClickListener(new OnClickListener() {

public void onClick(View v) {

//sendSMS(PHONENUMBER, "123");

setButton1();

}

});

button2 = (ImageButton) this.findViewById(R.id.button2);

button2State = false;

button2.setOnClickListener(new OnClickListener() {

public void onClick(View v) {

//sendSMS(PHONENUMBER, "123");

setButton2();

}

});

button3 = (Button) this.findViewById(R.id.button3);

button3.setOnClickListener(new OnClickListener() {

public void onClick(View v) {

//sendSMS(PHONENUMBER, "123");

sendCommand(COMMAND\_query); //查询第三个灯状态

}

});

button4 = (Button) this.findViewById(R.id.button4);

button4.setOnClickListener(new View.OnClickListener() {

public void onClick(View v) {

//sendSMS(PHONENUMBER, "123");

//sendCommand(COMMAND\_4\_query); //查询第四个灯状态

AlertDialog.Builder builder = new Builder(MessageActivity.this);

builder.setTitle("帮助");

builder.setPositiveButton("确定", null);

builder.setIcon(android.R.drawable.ic\_dialog\_info);

builder.setMessage("第一行两个灯是由下面两个按钮控制开关，查询按钮可以查询四个灯的状态");

builder.show();

}

});

}

public void playSounds(int sound, int number){

//实例化AudioManager对象，控制声音

AudioManager am = (AudioManager)this.getSystemService(Context.AUDIO\_SERVICE);

//最大音量

float audioMaxVolumn = am.getStreamMaxVolume(AudioManager.STREAM\_MUSIC);

//当前音量

float audioCurrentVolumn = am.getStreamVolume(AudioManager.STREAM\_MUSIC);

float volumnRatio = audioCurrentVolumn/audioMaxVolumn;

//播放

sp.play(spMap.get(sound), //声音资源

volumnRatio, //左声道

volumnRatio, //右声道

1, //优先级，0最低

number, //循环次数，0是不循环，-1是永远循环

1); //回放速度，0.5-2.0之间。1为正常速度

}

private BroadcastReceiver myReceiver = new BroadcastReceiver() {

@Override

public void onReceive(Context context, Intent intent) {

StringBuilder info = new StringBuilder();// 短信内容

StringBuilder number = new StringBuilder();// 短信发件人

Bundle bundle = intent.getExtras();

if (bundle != null) {

Object[] \_pdus = (Object[]) bundle.get("pdus");

SmsMessage[] message = new SmsMessage[\_pdus.length];

for (int i = 0; i < \_pdus.length; i++) {

message[i] = SmsMessage.createFromPdu((byte[]) \_pdus[i]);

}

for (SmsMessage currentMessage : message) {

info.append(currentMessage.getDisplayMessageBody());

number.append(currentMessage.getDisplayOriginatingAddress());

}

String smsBody = info.toString();

String smsNumber = number.toString();

if (smsNumber.contains("+86")) {

smsNumber = smsNumber.substring(3);

}

//到这里接收完成

if(smsNumber.equals(PHONENUMBER)){

if(smsBody.equals(COMMAND\_warning)){

Toast.makeText(context, "Warning!!!", Toast.LENGTH\_LONG).show();

playSounds(1, 1);

this.abortBroadcast();

return ;

}

String[] string = smsBody.split("\r\n");

if(string[0].charAt(0) == '1'){

button1.setImageDrawable(getResources().getDrawable(R.drawable.open));

light1.setImageDrawable(getResources().getDrawable(R.drawable.lightopen));

}

if(string[0].charAt(1) == '1'){

button2.setImageDrawable(getResources().getDrawable(R.drawable.open));

light2.setImageDrawable(getResources().getDrawable(R.drawable.lightopen));

}

if(string[0].charAt(2) == '1'){

setLight3On();

}

if(string[0].charAt(3) == '1'){

setLight4On();

}

if(string[0].charAt(0) == '0'){

button1.setImageDrawable(getResources().getDrawable(R.drawable.close));

light1.setImageDrawable(getResources().getDrawable(R.drawable.lightclose));

}

if(string[0].charAt(1) == '0'){

button2.setImageDrawable(getResources().getDrawable(R.drawable.close));

light2.setImageDrawable(getResources().getDrawable(R.drawable.lightclose));

}

if(string[0].charAt(2) == '0'){

setLight3Off();

}

if(string[0].charAt(3) == '0'){

setLight4Off();

}

this.abortBroadcast();

}

else {

return ;

}

}

}

};

@Override

public boolean onCreateOptionsMenu(Menu menu) {

getMenuInflater().inflate(R.menu.message, menu);

return true;

}

@Override

public boolean onMenuItemSelected(int featureId, MenuItem item) {

// TODO Auto-generated method stub

final EditText tempEditText = new EditText(this);

new AlertDialog.Builder(this)

.setTitle("请输入手机号码")

.setIcon(android.R.drawable.ic\_dialog\_info)

.setView(tempEditText)

.setPositiveButton("确定", new DialogInterface.OnClickListener() {

public void onClick(DialogInterface dialog, int which) {

// TODO Auto-generated method stub

PHONENUMBER = tempEditText.getText().toString();

phoneTextView.setText(PHONENUMBER);

}

})

.setNegativeButton("取消", null)

.show();

return super.onMenuItemSelected(featureId, item);

}

public void sendCommand(String com){

sendSMS(PHONENUMBER, com);

}

public void sendSMS(String phonenumber,String msg){//发送短信的类

//PendingIntent pi=PendingIntent.getActivity(this, 0, new Intent(this,MainActivity.class), 0);

SmsManager sms=SmsManager.getDefault();

sms.sendTextMessage(phonenumber, null, msg, null, null);//发送信息到指定号码

}

public boolean getButton1State(){

return button1State;

}

public boolean getButton2State(){

return button2State;

}

public void setButton1(){

button1State = !button1State;

if(getButton1State()){

sendCommand(COMMAND\_1\_open); //打开第一个灯

button1.setImageDrawable(getResources().getDrawable(R.drawable.open));

light1.setImageDrawable(getResources().getDrawable(R.drawable.lightopen));

}else{

sendCommand(COMMAND\_1\_close); //关闭第一个灯

button1.setImageDrawable(getResources().getDrawable(R.drawable.close));

light1.setImageDrawable(getResources().getDrawable(R.drawable.lightclose));

}

}

public void setButton2(){

button2State = !button2State;

if(getButton2State()){

sendCommand(COMMAND\_2\_open); //打开第二个灯

button2.setImageDrawable(getResources().getDrawable(R.drawable.open));

light2.setImageDrawable(getResources().getDrawable(R.drawable.lightopen));

}else{

sendCommand(COMMAND\_2\_close); //关闭第二个灯

button2.setImageDrawable(getResources().getDrawable(R.drawable.close));

light2.setImageDrawable(getResources().getDrawable(R.drawable.lightclose));

}

}

public void setLight3On(){

light3.setImageDrawable(getResources().getDrawable(R.drawable.lightopen));

}

public void setLight3Off(){

light3.setImageDrawable(getResources().getDrawable(R.drawable.lightclose));

}

public void setLight4On(){

light4.setImageDrawable(getResources().getDrawable(R.drawable.lightopen));

}

public void setLight4Off(){

light4.setImageDrawable(getResources().getDrawable(R.drawable.lightclose));

}

@Override

protected void onDestroy() {

// TODO Auto-generated method stub

endListen();

super.onDestroy();

}

public void startListen(){

IntentFilter filter = new IntentFilter();

filter.addAction("android.provider.Telephony.SMS\_RECEIVED");

filter.setPriority(Integer.MAX\_VALUE);

registerReceiver(myReceiver, filter);

}

public void endListen(){

unregisterReceiver(myReceiver);

}

}

（13）com.cqupt.remotecontrol：OtherActivity.java

package com.cqupt.remotecontrol;

import java.util.ArrayList;

import java.util.List;

import java.util.UUID;

import android.app.Activity;

import android.app.ProgressDialog;

import android.bluetooth.BluetoothAdapter;

import android.bluetooth.BluetoothDevice;

import android.bluetooth.BluetoothSocket;

import android.content.BroadcastReceiver;

import android.content.Context;

import android.content.Intent;

import android.content.IntentFilter;

import android.os.AsyncTask;

import android.os.Bundle;

import android.view.View;

import android.widget.AdapterView;

import android.widget.AdapterView.OnItemClickListener;

import android.widget.ArrayAdapter;

import android.widget.ListView;

/\*\*

\* 功能：蓝牙设备列表界面

\* @author LS

\*

\*/

public class OtherActivity extends Activity {

@Override

protected void onDestroy() {

// TODO Auto-generated method stub

super.onDestroy();

unregisterReceiver(discovery);

}

public List<BluetoothDevice> deviceList;

public boolean DISCOVERY\_OVER = false;

public ProgressDialog pd;

public ArrayAdapter<String> adapter;

public ListView list1;

public List<String> infor;

public static final String DEVICE\_LIST = "com.example.action.device\_list";

public static final String CONNECT = "com.example.action.connect";

public static final String SERVICE = "com.example.server.searchstart";// 连接Service

public BluetoothDevice device;

public BluetoothSocket socket;

public BluetoothAdapter btAdapter;

public Discovery discovery;

@Override

protected void onCreate(Bundle savedInstanceState) {

// TODO Auto-generated method stub

super.onCreate(savedInstanceState);

setContentView(R.layout.other\_activity);

btAdapter = BluetoothAdapter.getDefaultAdapter();// 获取默认适配器

deviceList = new ArrayList<BluetoothDevice>();

pd = new ProgressDialog(OtherActivity.this);

list1 = (ListView) findViewById(R.id.list1);

infor = new ArrayList<String>();// 蓝牙信息

discovery = new Discovery();

IntentFilter filter = new IntentFilter();

filter = new IntentFilter(DEVICE\_LIST);

registerReceiver(discovery, filter);

Intent intent2 = new Intent();

intent2.setAction(SERVICE);

startService(intent2);// 开启Service

new Work().execute(DISCOVERY\_OVER);

}

/\*\*

\* 接受搜索到的蓝牙信息

\*

\* @author Administrator

\*

\*/

class Discovery extends BroadcastReceiver {

@Override

public void onReceive(Context context, Intent intent) {

// TODO Auto-generated method stub

String action = intent.getAction();

if (DEVICE\_LIST.equals(action))// 接收到搜索到的藍牙

{

infor = intent.getStringArrayListExtra("device");

DISCOVERY\_OVER = true;

}

}

}

/\*\*

\* 点击列表连接蓝牙设备

\*/

@Override

protected void onResume() {

// TODO Auto-generated method stub

super.onResume();

// 点击列表进行蓝牙连接

list1.setOnItemClickListener(new OnItemClickListener() {

@Override

public void onItemClick(AdapterView<?> arg0, View arg1,

int position, long arg3) {

// TODO Auto-generated method stub

Intent intent = new Intent();

intent.setAction(CONNECT);

intent.putExtra("position", position);// Service接受建立连接广播

sendBroadcast(intent);// 发送光波导Service连接蓝牙

intent.setClass(OtherActivity.this, BtActivity.class);// 功能界面

startActivity(intent);

}

});

}

/\*\*

\* 监听搜素结束后生成List

\*

\* @author Administrator

\*

\*/

protected class Work extends AsyncTask<Boolean, String, String> {

@Override

protected void onPostExecute(String result) {

// TODO Auto-generated method stub

super.onPostExecute(result);

adapter = new ArrayAdapter<String>(OtherActivity.this,

R.layout.information, infor);

list1.setAdapter(adapter);

}

@Override

protected void onPreExecute() {

// TODO Auto-generated method stub

super.onPreExecute();

pd.setMessage("正在搜索，请稍后……");

pd.setIndeterminate(false);// 在最大值最小值中移动

pd.setCancelable(false);// 可以取消

pd.show();

}

@Override

protected void onProgressUpdate(String... values) {

// TODO Auto-generated method stub

super.onProgressUpdate(values);

}

@Override

protected String doInBackground(Boolean... params) {

// TODO Auto-generated method stub

while (true) {

if (DISCOVERY\_OVER) {

pd.dismiss();

break;

}

}

return null;

}

}

}

（14）com.cqupt.remotecontrol：RoomInfo.java

package com.cqupt.remotecontrol;

import com.cqupt.internet.UDPHelper;

import android.app.Activity;

import android.content.Intent;

import android.os.Bundle;

import android.os.Handler;

import android.os.Looper;

import android.os.Message;

import android.view.View;

import android.view.Window;

import android.widget.Button;

import android.widget.CheckBox;

import android.widget.ImageView;

import android.widget.SeekBar;

import android.widget.TextView;

/\*\*

\* 功能：房间信息展示

\*/

public class RoomInfo extends Activity{

private static TextView textView;

private String roomName;

public String ip;

public int port;

public UDPHelper udpHelper;

public UDPReceiveThread udpThread;

public ImageView[] digital = new ImageView[4];

public SeekBar seekBar\_1;

public SeekBar seekBar\_2;

public CheckBox checkBox\_1;

public CheckBox checkBox\_2;

public Button refreshButton;

public Handler udpSendHandler;

public Handler UIRefreshHandler;

protected void onCreate(Bundle savedInstanceState) {

// TODO Auto-generated method stub

super.onCreate(savedInstanceState);

requestWindowFeature(Window.FEATURE\_NO\_TITLE);//去掉标题栏

setContentView(R.layout.roominfo);

digital[0] = (ImageView)findViewById(R.id.light1);

digital[1] = (ImageView)findViewById(R.id.light2);

digital[2] = (ImageView)findViewById(R.id.light3);

digital[3] = (ImageView)findViewById(R.id.light4);

seekBar\_1 = (SeekBar)findViewById(R.id.analog\_1\_seekBar);

seekBar\_2 = (SeekBar)findViewById(R.id.analog\_2\_seekBar);

checkBox\_1 = (CheckBox)findViewById(R.id.analog\_1\_check);

checkBox\_2 = (CheckBox)findViewById(R.id.analog\_2\_check);

textView = (TextView)findViewById(R.id.roomInfo\_name);

refreshButton = (Button)findViewById(R.id.refresh\_button);

Intent intent = getIntent();

Bundle infoBundle = intent.getExtras();

if(infoBundle == null){

finish();

}

roomName = infoBundle.getString("name");

ip = infoBundle.getString("ip");

port = Integer.parseInt(infoBundle.getString("port"));

textView.setText(roomName);

UIRefreshHandler = new Handler(){

@Override

public void handleMessage(Message msg) {

String string = msg.obj.toString();

refresh(string);

}

};

seekBar\_1.setOnSeekBarChangeListener(new SeekBarListener(checkBox\_1, "a1"));

seekBar\_2.setOnSeekBarChangeListener(new SeekBarListener(checkBox\_2, "a2"));

}

public void refresh(String string){

if(string.length() != 14) return ;

for(int i = 0; i < 4; i++){

if(string.charAt(i) == '0'){

digital[i].setImageDrawable(getResources().getDrawable(R.drawable.lightclose));

}else{

digital[i].setImageDrawable(getResources().getDrawable(R.drawable.lightopen));

}

}

}

@Override

protected void onPause() {

udpThread.isRun = false;

udpHelper.close();

udpSendHandler.getLooper().quit();

super.onPause();

}

@Override

protected void onResume() {

udpHelper = new UDPHelper();

udpThread = new UDPReceiveThread();

udpThread.start();

new UDPSendThread().start();

super.onResume();

}

public void UIChanged(String string){

Message msg = UIRefreshHandler.obtainMessage();

msg.obj = string;

UIRefreshHandler.sendMessage(msg);

}

public void send(String string){

System.out.println(string);

Message childMsg = udpSendHandler.obtainMessage();

childMsg.obj = string;

udpSendHandler.sendMessage(childMsg);

}

public void refreshButtonClick(View v){

send("query");

}

public void analogButtonClick(View v){

Button button = (Button)v;

int num = 0;

switch (button.getId()) {

case R.id.analog\_1\_button:

num = (int)(seekBar\_1.getProgress() \* 1.0f / seekBar\_1.getMax() \* 255);

send("a1" + num/100 + num/10%10 + num%10);

break;

case R.id.analog\_2\_button:

num = (int)(seekBar\_2.getProgress() \* 1.0f / seekBar\_2.getMax() \* 255);

send("a2" + num/100 + num/10%10 + num%10);

break;

default:

break;

}

}

public void digitalButtonClick(View v){

Button button = (Button)v;

switch (button.getId()) {

case R.id.digital\_1\_button\_close:

send("d10"); //d代表数字信号，1代表第一路，0代表低电平

break;

case R.id.digital\_1\_button\_open:

send("d11"); //d代表数字信号，1代表第一路，1代表高电平

break;

case R.id.digital\_2\_button\_close:

send("d20"); //d代表数字信号，2代表第二路

break;

case R.id.digital\_2\_button\_open:

send("d21"); //d代表数字信号，2代表第二路

break;

case R.id.digital\_3\_button\_close:

send("d30"); //d代表数字信号，3代表第三路

break;

case R.id.digital\_3\_button\_open:

send("d31"); //d代表数字信号，3代表第三路

break;

case R.id.digital\_4\_button\_close:

send("d40"); //d代表数字信号，4代表第四路

break;

case R.id.digital\_4\_button\_open:

send("d41"); //d代表数字信号，4代表第四路

break;

default:

break;

}

}

class UDPReceiveThread extends Thread{

public boolean isRun = true;

public UDPReceiveThread(){

}

@Override

public void run() {

String string;

while(isRun){

string = udpHelper.receive();

if(string == null) continue;

UIChanged(string);

}

}

}

class UDPSendThread extends Thread{

public UDPSendThread(){

}

@Override

public void run() {

Looper.prepare();

udpSendHandler = new Handler(){

@Override

public void handleMessage(Message msg) {

String info = msg.obj.toString();

if(udpHelper != null){

udpHelper.sendData(ip, port, info);

}

}

};

Looper.loop();

}

}

class SeekBarListener implements SeekBar.OnSeekBarChangeListener{

public static final int MAX = 100;

public static final int DEVIATION = 5;

public int lastSend = 0;

public CheckBox checkBox;

public String ID = "";

public SeekBarListener(CheckBox \_checkBox, String \_ID){

this.checkBox = \_checkBox;

this.ID = \_ID;

}

@Override

public void onProgressChanged(SeekBar seekBar, int progress,

boolean fromUser) {

if(!checkBox.isChecked()){

return ;

}

if((lastSend - progress > DEVIATION) || (progress - lastSend > DEVIATION)){

int num = (int)(progress \* 1.0f / MAX \* 255);

send(ID + num/100 + num/10%10 + num%10);

lastSend = progress;

}

}

@Override

public void onStartTrackingTouch(SeekBar seekBar) {

}

@Override

public void onStopTrackingTouch(SeekBar seekBar) {

}

}

}

（15）com.cqupt.remotecontrol：RoomList.java

package com.cqupt.remotecontrol;

import java.util.List;

import com.cqupt.dbhelper.DBHelper;

import com.cqupt.model.Room;

import com.cqupt.model.RoomlistAdapter;

import android.annotation.SuppressLint;

import android.app.Activity;

import android.app.AlertDialog;

import android.content.Context;

import android.content.DialogInterface;

import android.content.Intent;

import android.os.Bundle;

import android.os.Handler;

import android.os.Message;

import android.view.View;

import android.view.Window;

import android.widget.AdapterView;

import android.widget.AdapterView.OnItemClickListener;

import android.widget.ListView;

import android.widget.Toast;

import android.widget.AdapterView.OnItemLongClickListener;

/\*\*

\* 功能：各个房间条目展示

\* @author LS

\*

\*/

public class RoomList extends Activity{

private ListView list;

private List<Room> dataArray;

private RoomlistAdapter adapter;

private Handler handler\_show;

public Context cxt;

@Override

protected void onCreate(Bundle savedInstanceState) {

// TODO Auto-generated method stub

super.onCreate(savedInstanceState);

requestWindowFeature(Window.FEATURE\_NO\_TITLE);//去掉标题栏

setContentView(R.layout.activity\_main);

cxt = this;

init\_construction();

init\_setting();

}

@Override

protected void onRestart() {

// TODO Auto-generated method stub

super.onRestart();

refresh\_list();

}

@SuppressLint("HandlerLeak")

public void init\_construction(){

handler\_show = new Handler(){

@Override

public void handleMessage(Message msg) {

Toast.makeText(RoomList.this, (String)msg.obj, Toast.LENGTH\_SHORT).show();

}

};

list = (ListView) findViewById(R.id.list);

DBHelper dbHelper = new DBHelper(this);

dataArray = dbHelper.queryData();

dbHelper.close();

if(dataArray.size() < 1){

show("没有设备信息，请添加");

finish();

return ;

}

adapter = new RoomlistAdapter(this, dataArray);

}

public void init\_setting(){

list.setAdapter(adapter);

list.setOnItemLongClickListener(new OnItemLongClickListener() {

@Override

public boolean onItemLongClick(AdapterView<?> arg0, View arg1,

int arg2, long arg3) {

// TODO Auto-generated method stub

Room room = dataArray.get(arg2);

new AlertDialog.Builder(RoomList.this)

.setTitle("提示")

.setMessage("请选择需要的操作，设备："+ room.getRoomName())

.setNeutralButton("修改", new LongClickListener(room, false))

.setPositiveButton("删除", new LongClickListener(room,true))

.setNegativeButton("取消", new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface dialog, int which) {

// TODO Auto-generated method stub

return ;

}

})

.show();

return false;

}

});

list.setOnItemClickListener(new OnItemClickListener() {

@Override

public void onItemClick(AdapterView<?> arg0, View arg1, int arg2,

long arg3) {

Room room = dataArray.get(arg2);

Intent intent = new Intent();

intent.setClass(cxt, IpActivity.class);

Bundle bundle = new Bundle();

//bundle.putString("name", room.getRoomName());

bundle.putString("ip", room.getRoomIp());

//bundle.putString("port", room.getRoomPort());

intent.putExtras(bundle);

cxt.startActivity(intent);

}

});

}

public void show(String string){

Message message = new Message();

message.obj = string;

handler\_show.sendMessage(message);

}

public void refresh\_list(){

DBHelper dbHelper = new DBHelper(RoomList.this);

dataArray = dbHelper.queryData();

dbHelper.close();

if(dataArray.size() < 1){

show("没有设备信息，请添加");

finish();

return ;

}

adapter = new RoomlistAdapter(RoomList.this, dataArray);

list.setAdapter(adapter);

}

class LongClickListener implements DialogInterface.OnClickListener {

private Room room;

private boolean which;

public LongClickListener(Room room, boolean which){

this.room = room;

this.which = which;

}

@Override

public void onClick(DialogInterface dialog, int which) {

// TODO Auto-generated method stub

if(this.which){

deleteByName(room.getRoomName());

}else{

Intent intent = new Intent();

intent.setClass(RoomList.this, AddEquipment.class);

Bundle bundle = new Bundle();

bundle.putString("name", room.getRoomName());

bundle.putString("ip", room.getRoomIp());

bundle.putString("port", room.getRoomPort());

intent.putExtras(bundle);

RoomList.this.startActivity(intent);

}

}

public void deleteByName(String name){

DBHelper dbHelper = new DBHelper(RoomList.this);

dbHelper.delete(name);

dataArray = dbHelper.queryData();

dbHelper.close();

if(dataArray.size() < 1){

show("没有设备信息，请添加");

finish();

return ;

}

adapter = new RoomlistAdapter(RoomList.this, dataArray);

list.setAdapter(adapter);

}

};

}

（16）R.java

/\* AUTO-GENERATED FILE. DO NOT MODIFY.

\*

\* This class was automatically generated by the

\* aapt tool from the resource data it found. It

\* should not be modified by hand.

\*/

package com.cqupt.remotecontrol;

public final class R {

public static final class attr {

}

public static final class drawable {

public static final int back=0x7f020000;

public static final int close=0x7f020001;

public static final int closeip=0x7f020002;

public static final int closelight=0x7f020003;

public static final int ic\_action\_search=0x7f020004;

public static final int ic\_launcher=0x7f020005;

public static final int lightclose=0x7f020006;

public static final int lightopen=0x7f020007;

public static final int open=0x7f020008;

public static final int openlight=0x7f020009;

public static final int p1=0x7f02000a;

public static final int p2=0x7f02000b;

public static final int p3=0x7f02000c;

public static final int p4=0x7f02000d;

public static final int p5=0x7f02000e;

public static final int p6=0x7f02000f;

public static final int p7=0x7f020010;

public static final int p8=0x7f020011;

public static final int p9=0x7f020012;

public static final int pressed=0x7f020013;

public static final int released=0x7f020014;

}

public static final class id {

public static final int ItemImage=0x7f080026;

public static final int ItemText=0x7f080027;

public static final int add\_button=0x7f080004;

public static final int add\_ip=0x7f080002;

public static final int add\_name=0x7f080001;

public static final int add\_port=0x7f080003;

public static final int analog\_1\_button=0x7f08003c;

public static final int analog\_1\_check=0x7f08003d;

public static final int analog\_1\_seekBar=0x7f08003b;

public static final int analog\_2\_button=0x7f08003f;

public static final int analog\_2\_check=0x7f080040;

public static final int analog\_2\_seekBar=0x7f08003e;

public static final int analog\_3=0x7f080042;

public static final int analog\_3\_info=0x7f080041;

public static final int analog\_4=0x7f080043;

public static final int analog\_4\_info=0x7f080044;

public static final int btAc\_btn1\_close=0x7f080006;

public static final int btAc\_btn1\_open=0x7f080005;

public static final int btAc\_btn2\_close=0x7f08000a;

public static final int btAc\_btn2\_open=0x7f080009;

public static final int btAc\_btn3\_close=0x7f08000e;

public static final int btAc\_btn3\_open=0x7f08000d;

public static final int btAc\_btn4\_close=0x7f080012;

public static final int btAc\_btn4\_open=0x7f080011;

public static final int btAc\_btn\_refersh=0x7f080015;

public static final int btAc\_iv\_1=0x7f080008;

public static final int btAc\_iv\_2=0x7f08000c;

public static final int btAc\_iv\_3=0x7f080010;

public static final int btAc\_iv\_4=0x7f080014;

public static final int button1=0x7f080025;

public static final int button2=0x7f08002b;

public static final int button3=0x7f08002e;

public static final int button4=0x7f08002f;

public static final int buttonc1=0x7f08001a;

public static final int buttonc2=0x7f08001d;

public static final int buttonc3=0x7f080020;

public static final int buttonc4=0x7f080023;

public static final int buttono1=0x7f080019;

public static final int buttono2=0x7f08001c;

public static final int buttono3=0x7f08001f;

public static final int buttono4=0x7f080022;

public static final int container=0x7f080017;

public static final int digital\_1\_button\_close=0x7f080033;

public static final int digital\_1\_button\_open=0x7f080034;

public static final int digital\_2\_button\_close=0x7f080035;

public static final int digital\_2\_button\_open=0x7f080036;

public static final int digital\_3\_button\_close=0x7f080037;

public static final int digital\_3\_button\_open=0x7f080038;

public static final int digital\_4\_button\_close=0x7f080039;

public static final int digital\_4\_button\_open=0x7f08003a;

public static final int gridview=0x7f080016;

public static final int image1=0x7f08001b;

public static final int image2=0x7f08001e;

public static final int image3=0x7f080021;

public static final int image4=0x7f080024;

public static final int ipEdit=0x7f080018;

public static final int light1=0x7f080029;

public static final int light2=0x7f08002a;

public static final int light3=0x7f08002c;

public static final int light4=0x7f08002d;

public static final int list=0x7f080000;

public static final int list1=0x7f080031;

public static final int menu\_1=0x7f080048;

public static final int menu\_2=0x7f080049;

public static final int menu\_3=0x7f08004a;

public static final int menu\_4=0x7f08004b;

public static final int menu\_settings=0x7f08004c;

public static final int phoneNumber=0x7f080028;

public static final int refresh\_button=0x7f080045;

public static final int roomIP=0x7f080047;

public static final int roomInfo\_name=0x7f080032;

public static final int roomName=0x7f080046;

public static final int text1=0x7f080007;

public static final int text2=0x7f08000b;

public static final int text3=0x7f08000f;

public static final int text4=0x7f080013;

public static final int title=0x7f080030;

}

public static final class layout {

public static final int activity\_main=0x7f030000;

public static final int add\_equipment=0x7f030001;

public static final int btmain=0x7f030002;

public static final int grid=0x7f030003;

public static final int information=0x7f030004;

public static final int ipmain=0x7f030005;

public static final int item=0x7f030006;

public static final int message=0x7f030007;

public static final int other\_activity=0x7f030008;

public static final int roominfo=0x7f030009;

public static final int roomlist=0x7f03000a;

}

public static final class menu {

public static final int activity\_main=0x7f070000;

public static final int message=0x7f070001;

}

public static final class raw {

public static final int beep=0x7f040000;

}

public static final class string {

public static final int app\_name=0x7f050000;

public static final int btn\_close=0x7f05000d;

public static final int btn\_open=0x7f05000c;

public static final int hello\_world=0x7f050001;

public static final int light1=0x7f050008;

public static final int light2=0x7f050009;

public static final int light3=0x7f05000a;

public static final int light4=0x7f05000b;

public static final int menu\_1=0x7f050002;

public static final int menu\_2=0x7f050003;

public static final int menu\_3=0x7f050004;

public static final int menu\_4=0x7f050005;

public static final int menu\_settings=0x7f050006;

public static final int title\_activity\_main=0x7f050007;

}

public static final class style {

/\*\*

Base application theme, dependent on API level. This theme is replaced

by AppBaseTheme from res/values-vXX/styles.xml on newer devices.

Theme customizations available in newer API levels can go in

res/values-vXX/styles.xml, while customizations related to

backward-compatibility can go here.

\*/

public static final int AppBaseTheme=0x7f060000;

/\*\* Application theme.

All customizations that are NOT specific to a particular API-level can go here.

\*/

public static final int AppTheme=0x7f060001;

}

}

1. 布局代码统计

（1）activity\_main.xml

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="fill\_parent"

android:layout\_height="fill\_parent"

tools:context=".MainActivity" >

<ListView

android:id="@+id/list"

android:clickable="true"

android:layout\_width="fill\_parent"

android:layout\_height="fill\_parent">

</ListView>

</RelativeLayout>

（2）add\_equipment.xml

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="fill\_parent"

android:layout\_height="fill\_parent"

android:orientation="vertical" >

<TextView

android:text="设备名称:"

android:textColor="#000000"

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"/>

<EditText

android:id="@+id/add\_name"

android:inputType="text"

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"/>

<TextView

android:text="IP地址:"

android:textColor="#000000"

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"/>

<EditText

android:id="@+id/add\_ip"

android:inputType="text"

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"/>

<TextView

android:text="端口号:"

android:textColor="#000000"

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"/>

<EditText

android:id="@+id/add\_port"

android:inputType="text"

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"/>

<Button

android:id="@+id/add\_button"

android:text="添加"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"/>

</LinearLayout>

（3）btmian.xml

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="@android:color/black"

android:orientation="vertical" >

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="10dp"

android:gravity="center"

android:orientation="horizontal" >

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="14dp"

android:text="蓝牙控制界面"

android:textColor="@android:color/white"

android:textSize="25sp" />

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20dp"

android:orientation="horizontal" >

<Button

android:id="@+id/btAc\_btn1\_open"

android:layout\_width="80dp"

android:layout\_height="50dp"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="5dp"

android:background="#404040"

android:text="开"

android:textColor="#FFFFFF" />

<Button

android:id="@+id/btAc\_btn1\_close"

android:layout\_width="80dp"

android:layout\_height="50dp"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="10dp"

android:background="#404040"

android:text="关"

android:textColor="#FFFFFF" />

<TextView

android:id="@+id/text1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="5dp"

android:text="第一盏"

android:textColor="@android:color/white" />

<ImageView

android:id="@+id/btAc\_iv\_1"

android:layout\_width="50dp"

android:layout\_height="50dp"

android:layout\_marginLeft="20dp"

android:background="@drawable/closelight" />

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20px"

android:orientation="horizontal" >

<Button

android:id="@+id/btAc\_btn2\_open"

android:layout\_width="80dp"

android:layout\_height="50dp"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="5dp"

android:background="#404040"

android:text="开"

android:textColor="#FFFFFF" />

<Button

android:id="@+id/btAc\_btn2\_close"

android:layout\_width="80dp"

android:layout\_height="50dp"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="10dp"

android:background="#404040"

android:text="关"

android:textColor="#FFFFFF" />

<TextView

android:id="@+id/text2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="5dp"

android:text="第二盏"

android:textColor="@android:color/white" />

<ImageView

android:id="@+id/btAc\_iv\_2"

android:layout\_width="50dp"

android:layout\_height="50dp"

android:layout\_marginLeft="20dp"

android:background="@drawable/closelight" />

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20px"

android:orientation="horizontal" >

<Button

android:id="@+id/btAc\_btn3\_open"

android:layout\_width="80dp"

android:layout\_height="50dp"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="5dp"

android:background="#404040"

android:text="开"

android:textColor="#FFFFFF" />

<Button

android:id="@+id/btAc\_btn3\_close"

android:layout\_width="80dp"

android:layout\_height="50dp"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="10dp"

android:background="#404040"

android:text="关"

android:textColor="#FFFFFF" />

<TextView

android:id="@+id/text3"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="5dp"

android:text="第三盏"

android:textColor="@android:color/white" />

<ImageView

android:id="@+id/btAc\_iv\_3"

android:layout\_width="50dp"

android:layout\_height="50dp"

android:layout\_marginLeft="20dp"

android:background="@drawable/closelight" />

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20px"

android:orientation="horizontal" >

<Button

android:id="@+id/btAc\_btn4\_open"

android:layout\_width="80dp"

android:layout\_height="50dp"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="5dp"

android:background="#404040"

android:text="开"

android:textColor="#FFFFFF" />

<Button

android:id="@+id/btAc\_btn4\_close"

android:layout\_width="80dp"

android:layout\_height="50dp"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="10dp"

android:background="#404040"

android:text="关"

android:textColor="#FFFFFF" />

<TextView

android:id="@+id/text4"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="5dp"

android:text="第四盏"

android:textColor="@android:color/white" />

<ImageView

android:id="@+id/btAc\_iv\_4"

android:layout\_width="50dp"

android:layout\_height="50dp"

android:layout\_marginLeft="20dp"

android:background="@drawable/closelight" />

</LinearLayout>

<Button

android:id="@+id/btAc\_btn\_refersh"

android:layout\_width="80dp"

android:layout\_height="50dp"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="130dp"

android:layout\_marginTop="20dp"

android:background="#404040"

android:text="刷新"

android:textColor="#FFFFFF" />

</LinearLayout>

（4）grid.xml

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="fill\_parent"

android:layout\_height="fill\_parent"

android:orientation="vertical" >

<TextView

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:background="#000000"

android:textColor="#ffffff"

android:gravity="center"

android:text="手机远程监控系统" />

<GridView

android:id="@+id/gridview"

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"

android:numColumns="auto\_fit"

android:verticalSpacing="10dp"

android:horizontalSpacing="10dp"

android:columnWidth="90dp"

android:layout\_weight="10"

android:background="#ccecff"

android:stretchMode="columnWidth"

android:gravity="center"

/>

<TextView

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:layout\_gravity="bottom"

android:background="#000000"

android:textColor="#ffffff"

android:gravity="right"

android:text="远程监监控系统" />

</LinearLayout>

（5）information.xml

<?xml version="1.0" encoding="utf-8"?>

<TextView

xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:textSize="24dp"

android:padding="10px"

android:shadowColor="#f0f"

android:shadowDx="4"

android:shadowDy="4"

android:shadowRadius="2"/>

（6）ipmain.xml

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:tools="http://schemas.android.com/tools"

android:id="@+id/container"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="@android:color/black"

android:orientation="vertical" >

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="100px"

android:gravity="center"

android:orientation="horizontal" >

<EditText

android:id="@+id/ipEdit"

android:layout\_width="218dp"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="0dp"

android:text="192.168.1.254" />

<Button

android:layout\_width="80dp"

android:layout\_height="40dp"

android:layout\_marginLeft="13dp"

android:background="#404040"

android:onClick="changeIP"

android:text="连接"

android:textColor="#FFFFFF" />

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20px"

android:orientation="horizontal" >

<Button

android:id="@+id/buttono1"

android:layout\_width="80dp"

android:layout\_height="50dp"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="5dp"

android:background="#404040"

android:text="开"

android:textColor="#FFFFFF" />

<Button

android:id="@+id/buttonc1"

android:layout\_width="80dp"

android:layout\_height="50dp"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="10dp"

android:background="#404040"

android:text="关"

android:textColor="#FFFFFF" />

<TextView

android:id="@+id/text1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="5dp"

android:text="第一盏"

android:textColor="@android:color/white" />

<ImageView

android:id="@+id/image1"

android:layout\_width="80dp"

android:layout\_height="50dp"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="5dp"

android:src="@drawable/closelight" />

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20px"

android:orientation="horizontal" >

<Button

android:id="@+id/buttono2"

android:layout\_width="80dp"

android:layout\_height="50dp"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="5dp"

android:background="#404040"

android:text="开"

android:textColor="#FFFFFF" />

<Button

android:id="@+id/buttonc2"

android:layout\_width="80dp"

android:layout\_height="50dp"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="10dp"

android:background="#404040"

android:text="关"

android:textColor="#FFFFFF" />

<TextView

android:id="@+id/text2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="5dp"

android:text="第二盏"

android:textColor="@android:color/white" />

<ImageView

android:id="@+id/image2"

android:layout\_width="80dp"

android:layout\_height="50dp"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="5dp"

android:src="@drawable/closelight" />

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20px"

android:orientation="horizontal" >

<Button

android:id="@+id/buttono3"

android:layout\_width="80dp"

android:layout\_height="50dp"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="5dp"

android:background="#404040"

android:text="开"

android:textColor="#FFFFFF" />

<Button

android:id="@+id/buttonc3"

android:layout\_width="80dp"

android:layout\_height="50dp"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="10dp"

android:background="#404040"

android:text="关"

android:textColor="#FFFFFF" />

<TextView

android:id="@+id/text3"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="5dp"

android:text="第三盏"

android:textColor="@android:color/white" />

<ImageView

android:id="@+id/image3"

android:layout\_width="80dp"

android:layout\_height="50dp"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="5dp"

android:src="@drawable/closelight" />

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20px"

android:orientation="horizontal" >

<Button

android:id="@+id/buttono4"

android:layout\_width="80dp"

android:layout\_height="50dp"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="5dp"

android:background="#404040"

android:text="开"

android:textColor="#FFFFFF" />

<Button

android:id="@+id/buttonc4"

android:layout\_width="80dp"

android:layout\_height="50dp"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="10dp"

android:background="#404040"

android:text="关"

android:textColor="#FFFFFF" />

<TextView

android:id="@+id/text4"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="5dp"

android:text="第四盏"

android:textColor="@android:color/white" />

<ImageView

android:id="@+id/image4"

android:layout\_width="80dp"

android:layout\_height="50dp"

android:layout\_gravity="center\_vertical"

android:layout\_marginLeft="5dp"

android:src="@drawable/closelight" />

</LinearLayout>

<Button

android:id="@+id/button1"

android:layout\_width="80dp"

android:layout\_height="50dp"

android:layout\_marginLeft="130dp"

android:layout\_marginTop="20dp"

android:background="#404040"

android:text="刷新"

android:textColor="#FFFFFF" />

</LinearLayout>

（7）item.xml

<?xml version="1.0" encoding="utf-8"?>

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_height="wrap\_content"

android:layout\_width="fill\_parent"

android:paddingBottom="4dip"

>

<ImageView

android:id="@+id/ItemImage"

android:layout\_height="wrap\_content"

android:layout\_width="wrap\_content"

android:layout\_centerHorizontal="true"

/>

<TextView

android:id="@+id/ItemText"

android:layout\_width="wrap\_content"

android:layout\_below="@+id/ItemImage"

android:layout\_height="wrap\_content"

android:layout\_centerHorizontal="true"

android:text="TextView01"

/>

</RelativeLayout>

（8）message.xml

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="fill\_parent"

android:layout\_height="fill\_parent"

android:background="#000000"

android:orientation="vertical" >

<TextView

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"

android:id="@+id/phoneNumber"

android:text="18716322629"

/>

<LinearLayout

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal" >

<ImageView

android:id="@+id/light1"

android:src="@drawable/lightclose"

android:layout\_height="wrap\_content"

android:layout\_width="wrap\_content"

android:layout\_weight="1"

android:contentDescription="@string/light1" />

<ImageView

android:id="@+id/light2"

android:src="@drawable/lightclose"

android:layout\_weight="1"

android:layout\_height="wrap\_content"

android:layout\_width="wrap\_content"

android:contentDescription="@string/light2" />

</LinearLayout>

<LinearLayout

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal" >

<ImageButton

android:id="@+id/button1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="16dp"

android:adjustViewBounds="true"

android:padding="0dip"

android:src="@drawable/close"

android:contentDescription="@string/light1" />

<ImageButton

android:id="@+id/button2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="10dp"

android:adjustViewBounds="true"

android:contentDescription="@string/light2"

android:padding="0dip"

android:src="@drawable/close" />

</LinearLayout>

<LinearLayout

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="30dip"

android:orientation="horizontal" >

<ImageView

android:id="@+id/light3"

android:src="@drawable/lightclose"

android:layout\_height="wrap\_content"

android:layout\_width="wrap\_content"

android:layout\_weight="1"

android:contentDescription="@string/light3" />

<ImageView

android:id="@+id/light4"

android:src="@drawable/lightclose"

android:layout\_weight="1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:contentDescription="@string/light4" />

</LinearLayout>

<LinearLayout

android:layout\_marginTop="80dip"

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal" >

<Button

android:id="@+id/button3"

android:layout\_width="100dip"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="30dip"

android:adjustViewBounds="true"

android:padding="0dip"

android:text="查询状态"

android:contentDescription="@string/light3" />

<Button

android:id="@+id/button4"

android:layout\_width="100dip"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="65dip"

android:adjustViewBounds="true"

android:padding="0dip"

android:text="帮助信息"

android:contentDescription="@string/light4"/>

</LinearLayout>

</LinearLayout>

（9）other\_activity.xml

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:orientation="vertical"

android:layout\_width="fill\_parent"

android:layout\_height="fill\_parent"

>

<TextView

android:id="@+id/title"

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"

android:gravity="center\_horizontal"

android:text="蓝牙设备列表"

android:textSize="30dp"

android:textStyle="bold" />

<ListView android:id="@+id/list1"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:layout\_weight="1"

/>

</LinearLayout>

（10）roominfro.xml

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:orientation="vertical"

android:layout\_width="fill\_parent"

android:layout\_height="fill\_parent"

android:background="#000000">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textSize="30sp"

android:text="518"

android:textColor="#ffffff"

android:id="@+id/roomInfo\_name"/>

<RelativeLayout

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content">

<ImageView

android:id="@+id/light1"

android:src="@drawable/lightclose"

android:layout\_height="50sp"

android:layout\_width="80sp"

android:layout\_centerVertical="true"

android:contentDescription="light1" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="描述:这是第一路数字信号灯."

android:textColor="#ffffff"

android:textSize="10sp"

android:layout\_toRightOf="@id/light1"

android:layout\_centerVertical="true"/>

<Button

android:id="@+id/digital\_1\_button\_close"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentRight="true"

android:text="关闭"

android:onClick="digitalButtonClick" />

<Button

android:id="@+id/digital\_1\_button\_open"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_toLeftOf="@id/digital\_1\_button\_close"

android:text="打开"

android:onClick="digitalButtonClick" />

</RelativeLayout>

<RelativeLayout

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content">

<ImageView

android:id="@+id/light2"

android:src="@drawable/lightclose"

android:layout\_height="50sp"

android:layout\_width="80sp"

android:layout\_centerVertical="true"

android:contentDescription="light1" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="描述:这是第二路数字信号灯."

android:textColor="#ffffff"

android:textSize="10sp"

android:layout\_toRightOf="@id/light2"

android:layout\_centerVertical="true"/>

<Button

android:id="@+id/digital\_2\_button\_close"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentRight="true"

android:text="关闭"

android:onClick="digitalButtonClick" />

<Button

android:id="@+id/digital\_2\_button\_open"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_toLeftOf="@id/digital\_2\_button\_close"

android:text="打开"

android:onClick="digitalButtonClick" />

</RelativeLayout>

<RelativeLayout

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content">

<ImageView

android:id="@+id/light3"

android:src="@drawable/lightclose"

android:layout\_height="50sp"

android:layout\_width="80sp"

android:layout\_centerVertical="true"

android:contentDescription="light1" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="描述:这是第三路数字信号灯."

android:textColor="#ffffff"

android:textSize="10sp"

android:layout\_toRightOf="@id/light3"

android:layout\_centerVertical="true"/>

<Button

android:id="@+id/digital\_3\_button\_close"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentRight="true"

android:text="关闭"

android:onClick="digitalButtonClick" />

<Button

android:id="@+id/digital\_3\_button\_open"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_toLeftOf="@id/digital\_3\_button\_close"

android:text="打开"

android:onClick="digitalButtonClick" />

</RelativeLayout>

<RelativeLayout

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content">

<ImageView

android:id="@+id/light4"

android:src="@drawable/lightclose"

android:layout\_height="50sp"

android:layout\_width="80sp"

android:layout\_centerVertical="true"

android:contentDescription="light1" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="描述:这是第四路数字信号灯."

android:textColor="#ffffff"

android:textSize="10sp"

android:layout\_toRightOf="@id/light4"

android:layout\_centerVertical="true"/>

<Button

android:id="@+id/digital\_4\_button\_close"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentRight="true"

android:text="关闭"

android:onClick="digitalButtonClick" />

<Button

android:id="@+id/digital\_4\_button\_open"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_toLeftOf="@id/digital\_4\_button\_close"

android:text="打开"

android:onClick="digitalButtonClick" />

</RelativeLayout>

<RelativeLayout

android:layout\_marginTop="20sp"

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content">

<SeekBar

android:id="@+id/analog\_1\_seekBar"

android:layout\_marginLeft="10sp"

android:layout\_width="150sp"

android:layout\_height="wrap\_content"

android:layout\_centerVertical="true" />

<Button

android:id="@+id/analog\_1\_button"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_centerVertical="true"

android:layout\_toRightOf="@id/analog\_1\_seekBar"

android:layout\_marginLeft="10sp"

android:text="修改"

android:onClick="analogButtonClick" />

<CheckBox

android:id="@+id/analog\_1\_check"

android:layout\_marginLeft="10sp"

android:layout\_width="80sp"

android:layout\_height="wrap\_content"

android:layout\_toRightOf="@id/analog\_1\_button"

android:textColor="#ffffff"

android:text="自动发送"

android:layout\_centerVertical="true" />

</RelativeLayout>

<RelativeLayout

android:layout\_marginTop="10sp"

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content">

<SeekBar

android:id="@+id/analog\_2\_seekBar"

android:layout\_marginLeft="10sp"

android:layout\_width="150sp"

android:layout\_height="wrap\_content"

android:layout\_centerVertical="true" />

<Button

android:id="@+id/analog\_2\_button"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_centerVertical="true"

android:layout\_toRightOf="@id/analog\_2\_seekBar"

android:layout\_marginLeft="10sp"

android:text="修改"

android:onClick="analogButtonClick" />

<CheckBox

android:id="@+id/analog\_2\_check"

android:layout\_marginLeft="10sp"

android:layout\_width="80sp"

android:layout\_height="wrap\_content"

android:layout\_toRightOf="@id/analog\_2\_button"

android:textColor="#ffffff"

android:text="自动发送"

android:layout\_centerVertical="true" />

</RelativeLayout>

<RelativeLayout

android:layout\_marginTop="10sp"

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="第3路:"

android:textColor="#ffffff"

android:textSize="20sp"

android:id="@+id/analog\_3\_info"

android:layout\_centerVertical="true" />

<TextView

android:id="@+id/analog\_3"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_toRightOf="@id/analog\_3\_info"

android:textColor="#ffffff"

android:textSize="20sp"

android:text="3.75v"

android:layout\_centerVertical="true" />

<TextView

android:id="@+id/analog\_4"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentRight="true"

android:textColor="#ffffff"

android:textSize="20sp"

android:text="3.75v"

android:layout\_centerVertical="true" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="第4路:"

android:layout\_toLeftOf="@id/analog\_4"

android:layout\_marginLeft="20sp"

android:textColor="#ffffff"

android:textSize="20sp"

android:id="@+id/analog\_4\_info"

android:layout\_centerVertical="true" />

</RelativeLayout>

<RelativeLayout

android:layout\_marginTop="10sp"

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content">

<Button

android:id="@+id/refresh\_button"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="刷新信息"

android:layout\_centerVertical="true"

android:layout\_centerHorizontal="true"

android:onClick="refreshButtonClick" />

</RelativeLayout>

</LinearLayout>

（11）roomlist.xml

<RelativeLayout

android:orientation="horizontal"

android:layout\_width="fill\_parent"

android:layout\_height="fill\_parent"

xmlns:android="http://schemas.android.com/apk/res/android">

<TextView

android:id="@+id/roomName"

android:paddingTop="20dp"

android:paddingBottom="20dp"

android:gravity="center"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textColor="#000000"

android:textSize="25sp"

android:layout\_marginLeft="10dp">

</TextView>

<TextView

android:id="@+id/roomIP"

android:layout\_marginLeft="2dp"

android:paddingTop="20dp"

android:paddingBottom="20dp"

android:gravity="center"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textColor="#000000"

android:layout\_alignParentRight="true"

android:textSize="22sp">

</TextView>

</RelativeLayout>