

文章编号: 1005-8451 (2013) 08-0053-04

基于Android的列车综合无线通信装置的 仿真研究

宛紫晶, 郎诚廉

(同济大学 电气工程系, 上海 201804)

摘要: 本文研究一种基于Android的列车综合无线通信装置 (CIR) 仿真系统的方案, 旨在利用Android系统下提供的各种资源建立无线通信装置终端, 以此在局域网内仿真列车综合无线通信装置的相关功能。该方案采用系统硬件设备简单, 成本低, 易于扩展, 可用于模拟驾驶仿真台中, 对驾驶员的培训有着实际的应用意义与价值。

关键词: Android; 语音; 实时传输; 列车综合无线通信装置

中图分类号: U285.2 **文献标识码:** A

Simulation and study on train integrated radio communication equipment based on Android System

WAN Zijing, LANG Chenglian

(Department of Electrical Engineering, Tongji University, Shanghai 201804, China)

Abstract: In this paper, a program of simulation of train integrated radio communication equipment based on Android System was addressed in which the resources provided by Android System were used to build the radio communication device terminal so that the function of TIR in local area network could be simulated. The program has advantages such as simple hardware, low cost, good scalability and it could be applied into the driving simulation simulator which was with practical meaning and value for the training of driver.

Key words: Android; voice; real-time transmission; train integrated radio communication equipment

列车综合无线通信调度装置 (CIR), 简称无线列调, 它是铁路无线通信的主要组成部分, 是组织铁路运输、行车安全、提高生产效率的重要通信设施, 其通信质量的好坏直接关系到铁路的行车安全。因此, 通过仿真手段, 对列车综合无线通信装置的相关功能进行研究并应用, 对于列车驾驶员上岗前的培训, 尤其是列车运行时出现故障情况下的训练, 有着极大的意义和应用前景。

本文设计了一种基于 Android 系统的列车无线通信装置通信功能的仿真方案。

1 CIR功能简介

机车综合无线通信设备是基于 GSM-R 数字移动通信技术, GPS 全球定位技术, 450 MHz 模拟无线电台通信等技术的综合车载通信设备。它

与地面的 GSM-R 设备和 450 MHz 设备共同组成一个完整的铁路综合无线通信网。CIR 中关键功能包括基于 GSM-R 和 450 MHz 的语音调度通信, 无线调度命令信息传送, 无线车次号发送与校核信息, 列尾风压信息传送, GPS 定位信息, MMI 终端显示等功能。

本文首先介绍了 Android 终端及开发环境, 然后分模块在 Android 终端上对各部分仿真功能及方案进行阐述, 其中, 对关键的实时语音通信功能仿真方案进行了详细说明。

2 Android系统及开发环境

Android 是基于 Linux 内核, 完全开源和免费使用的, 其优异的性能以及免费性和开放性, 使开发人员可以随时取得程序的源代码。Android 开发工具包括: JDK5 或 JDK6, Eclipse, Android SDK, Eclipse 的插件 ADT (Android

收稿日期: 2012-12-31

作者简介: 宛紫晶, 在读硕士研究生; 郎诚廉, 副教授。

development tools)。ADT 扩展了 Eclipse 的功能，可以快速建立新的 Android 项目，添加了基于 Android 框架的 API 组件，为 Eclipse 下 Android 应用程序的开发提高了效率。

3 语音调度通信模块

语音调度通信模块的功能包括车到车，车到调度中心的实时语音通信调度。通过构建实时语音通信系统，Android 端到 PC 端的实时语音通信系统达到仿真功能的目的。其中 Android 端程序中主要分语音采集播放模块，语音传输模块。语音采集播放模块完成对语音信号的采集、存储和播放等。语音传输模块运用 UDP 协议对语音信号进行封包和传输。程序流程图如图 1 所示。

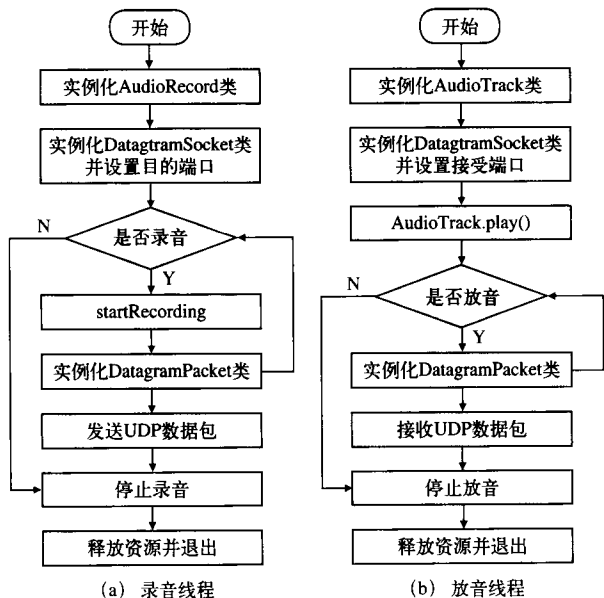


图1 程序流程图

3.1 音频采集播放模块

语音采集模块主要负责对音频信号的采集与存储。音频是一种连续变化的模拟信号，但计算机只能处理和记录二进制的数字信号，由自然音源得到的音频信号必须经过一定的变化，成为数字音频信号后才能送到计算机中作进一步的处理。数字音频系统通过将声波的波形转换成一系列二进制数据（A/D 转换），A/D 转换器以每秒钟上万次的速率对声波进行采样，每个采样点都记录下了原始模拟声波在某一时刻的状态，通常称之为样本，而每一秒钟所采样的数目则称为采样频率，通过将一串连续的样本连接起来，就可以在

计算机中描述一段声音。对于采样过程中的每一个样本，数字音频系统会分配一定存储位来记录声波的振幅，一般称之为采样分辨率或采样精度，采样精度越高，声音还原时就会越细腻。对于在 Android 系统下进行音频采集，最重要的是理解声音数字化的两个关键步骤：采样和量化。采样就是每隔一定时间就读一次声音信号的幅度，而量化则是将采样得到的声音信号幅度转化为数字值，从本质上讲，采样是时间上的数字化，而量化则是幅度上的数字化。

在 Android 系统下对音频的采集播放实现主要是通过 AudioRecord 和 AudioTrack 两个类实现，这两个类是 Android 获取和播放音频流的重要类，放置在 Android.media 包中。与该包中的 MediaRecorder 和 MediaPlayer 类不同，AudioRecord 和 AudioTrack 类在获取和播放音频数据流时无需通过文件保存和读取，可以动态地直接获取和播放音频流，在实时处理音频数据流时非常有用。首先分别对 AudioRecord 和 AudioTrack 类进行实例化，在实例化时，参数中各有一个缓冲数组大小，经过试验这个数组大小应和 AudioRecord 和 AudioTrack 能正常实例化所需的最小 Buffer 大小相等，这块 Buffer 的大小可以通过 AudioRecord 和 AudioTrack 类的 getMinBufferSize () 方法获得。然后在实例化中设置录放音的采样频率，量化位数以及声道数，本例中采用采样频率 8 000 Hz、16 bit、单声道。实例化成功后还要分别设置一块数组以存储收发的音频数据。

3.2 语音传输模块

本方案中的音频传输主要采用局域网内的基于 UDP/IP 的流式传送，UDP 协议的全称是用户数据包协议，是一种无连接的协议。UDP 协议的主要作用是将网络数据流量压缩成数据包的形式，每一个数据包的前 8 个 byte 用来包含报头信息，剩余 byte 则包含具体的传输数据。本例中，通信双方，在录音线程内通过一个套接字将音频数据发送到另一方的放音线程的套接字，自己的放音线程内则有另一个套接字通过新端口接收另一方传送来的数据，这样就实现了双向数据的传送。程序中 UDP 的创建主要通过 DatagramSocket 类和 DatagramPacket 类来实现，首先实例化

DatagramSocket 类,在实例化的同时指定要发送目的地的端口号。接着实例化 DatagramPacket 类将数据进行打包并绑定目的端口号与 IP 地址。其中发送端部分代码如下:

```
public void run() {
    record.startRecording();
    Log.d("startRecording","ok");
    while (isrunning) {
        record.read(recordbuf, 0,
        recordbufsize);
        Log.d("read", "ok");
        try {
            recordpacket = new
            DatagramPacket(recordbuf,
            recordbuf.length,
            serveraddress, 8888);
            Log.d("packet", "ok");
            udps.send(recordpacket);
            Log.d("send", "ok");
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}
```

其中 record 是 AudioRecord 录音类实例化的对象, startRecording() 方法表示开始录音, read 方法将从 MIC 采集到的音频存入指定的 Buffer 中, udps 是 DatagramSocket 类实例化对象, recordpacket 是 DatagramPacke 的实例化对象, 负责将采集的数据进行 UDP 打包封装。

为方便调试,在同一个局域网内的两台 Android 系统手机上测试程序,经测试能够实现通信功能,可以达到仿真目的。

4 GPS定位及数据收发模块

GPS 定位模块通过在 Android 上采集 GPS 定位信息发送给 PC 端并显示在 UI 界面上。Android 下的 GPS 定位信息采集主要通过 Location-Manager, LocationListener, Location, GpsStatus. Listener 等类来实现。LocationMangager 类实现位置管理器。要想操作定位相关设备,必须先定义 LocationManager。LocationListener 类实现位

置监听,位置变化,设备开关与状态。Location 类可以获取时间、经纬度、海拔等信息。数据收发模块通过实现 Android 终端与 PC 机之间的文本收发达到对调度命令信息传送,无线车次号校核信息,列尾风压信息传送等功能的仿真。数据输入通过 Android 端下的 setText 控件实现,数据传送通过局域网内的 UDP/IP 协议实现。

调度命令信息收发界面如图 2 所示。

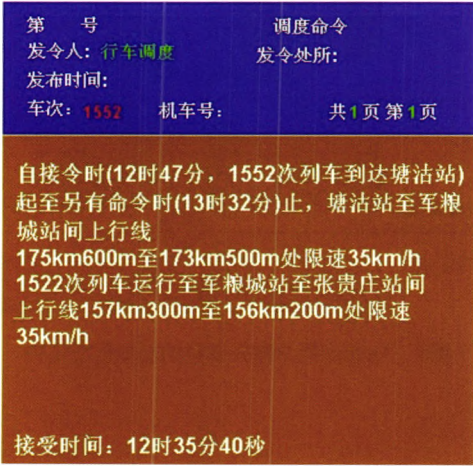


图2 调度命令信息收发界面

5 MMI显示模块

本例中按照实际的列车综合无线列控装置 MMI 界面样式,设计了 Android 平台下的仿真界面如图 3 所示,其中显示参数通过数据收发模块,在 PC 机上发送,Android 端接收并显示在界面上。最下方的 7 个 BUTTON 中调度按钮发起与 PC 端列控中心仿真软件的通信呼叫,最右 4 个按钮按下任意键都可以与另一台 Android 终端进行呼叫通信。

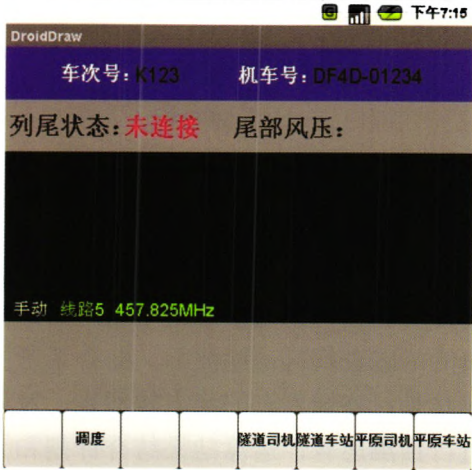


图3 450Mhz模式主界面

6 按键模块

要制作按键硬件主要涉及 Android 系统下的驱动程序编写。在 Android 的上层中，可以通过获得这些设备产生的事件，并对设备事件做出响应。在 Java 框架和应用程序层，用按键事件获得各种键盘信息。Android 用户输入系统的基本层次结构如图 4 所示。

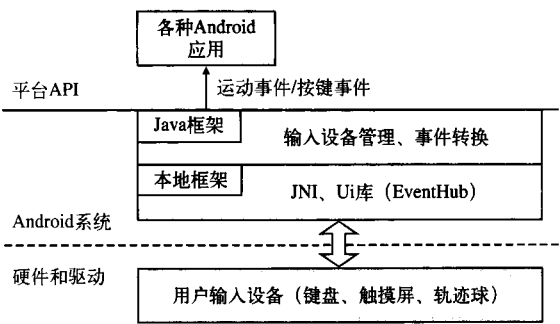


图4 Android用户输入系统的基本层次结构

7 结束语

本文利用 Android 嵌入式系统的特性，将其

责任编辑 陈 蓉

(上接 P52)

3.2.3 容错模型

最终的车次追踪容错模型如图 3 所示。

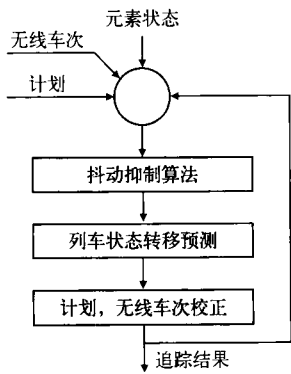


图3 追踪模型

模型的输入信息为车站的元素状态，计划信息和无线车次号信息。元素的状态信息和已建立的元素间的邻接关系是推算列车位置的基础，在对元素的状态进行抖动滤波后，结合系统上一次的运算结果，查询列车的状态转移表，对列车的状态进行预测运算，运算结果结合计划和无线车次号信息进行校正，校正后的结果作为下一次预

作为列车综合无线通信系统的仿真终端，提出并设计一套基于 Android 系统的列车综合无线通信装置仿真系统方案。对每一部分的实现做了阐述，尤其对无线列调通信功能仿真模块进行了详细说明，未来可将其运用在模拟仿真驾驶台中，对于驾驶员上岗前在突发事故情况下的训练与教学有着比较实际的应用价值和现实意义。

参考文献：

[1] 张大千, 崔建国. 无线列车调度实现数据话音同传 [J]. 沈阳航空工业学院学报, 2006, 14 (8): 12-13.
[2] 陈志杰, 徐 钧, 郑 敏. 机车综合无线通信设备 (CIR) 的技术方案 [J]. 铁道通信信号, 2006, 18 (5): 24-25.
[3] 林 城. Android2.3 应用开发实践 [M]. 北京: 机械工业出版社, 2011.
[4] 周 巍, 何 涛, 林嘉宇. Android 系统智能手机语音应用开发环境构架 [J]. 微处理机, 2011.

4 结束语

本文在对车次追踪问题分析的基础上，基于有限自动机模型理论，提出了元素抖动抑制算法，建立了具有容错性质的车次追踪模型，该模型具有高可靠性，高灵活性，并能有效提高车次追踪系统的可靠性，对改善调度集中系统的车次号正确性具有重要意义。

参考文献：

[1] 王建英. 铁路行车调度指挥仿真系统 [J]. 中国铁道科学, 2007 (5): 110-116.
[2] 王建英. 分散自律调度集中系统中车次追踪算法的研究 [J]. 铁路计算机应用, 2007, 16 (9): 45-48.
[3] John E.Hopcroft. 自动机理论、语言和计算导论 [M]. 孙家骕, 译. 北京: 机械工业出版社, 2008.

责任编辑 陈 蓉