

Android

朱元波 ◆ 编著

传感器开发与智能设备 案例实战



本书涵盖了 Android 传感器和外设开发的主要内容

如 Android 传感器系统分析、地图定位、光线传感器、接近警报、磁场传感器、加速度传感器、方向传感器、陀螺仪传感器、旋转向量传感器、距离传感器、气压传感器、温度传感器、湿度传感器、Android 蓝牙系统、语音识别、手势识别、NFC 近场通信、拍照解析条形码、麦克风音频录制、基于图像处理的人脸识别、行走轨迹记录等。



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS



Android

传感器开发与智能设备 案例实战

朱元波◆编著

人民邮电出版社
北京

图书在版编目 (C I P) 数据

Android传感器开发与智能设备案例实战 / 朱元波编
著. -- 北京 : 人民邮电出版社, 2016. 5
ISBN 978-7-115-41474-8

I. ①A… II. ①朱… III. ①移动终端—应用程序—
程序设计 IV. ①TN929.53

中国版本图书馆CIP数据核字(2016)第080065号

内 容 提 要

本书主要介绍 Android 传感器和外设的开发, 共 29 章, 主要包括 Android 开发技术基础、获取并编译源码、Android 技术核心框架分析、Android 传感器系统分析、使用地图定位、光线传感器详解、接近警报传感器详解、磁场传感器详解、加速度传感器详解、方向传感器详解、陀螺仪传感器详解、旋转向量传感器详解、距离传感器详解、气压传感器详解、温度传感器详解、湿度传感器详解、Android 蓝牙系统概述、低功耗蓝牙技术详解、语音识别技术详解、手势识别技术详解、NFC 近场通信技术详解、拍照解析条形码技术详解、基于图像处理的人脸识别技术详解、行走轨迹记录器、手势音乐播放器和智能家居系统等。本书几乎涵盖了 Android 传感器和外设开发所需的所有主要内容, 讲解方法通俗易懂。

本书适合 Android 初学者、Android 爱好者以及 Android 底层开发人员、传感器开发人员、智能设备开发人员、Android 外部设备开发工程师学习, 也可以作为相关培训学校和大专院校相关专业的教学用书。

-
- ◆ 编 著 朱元波
责任编辑 张 涛
责任印制 焦志炜
 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京圣夫亚美印刷有限公司印刷
 - ◆ 开本: 787×1092 1/16
印张: 47
字数: 1203 千字 2016 年 5 月第 1 版
印数: 1-2 000 册 2016 年 5 月北京第 1 次印刷
-

定价: 108.00 元

读者服务热线: (010)81055410 印装质量热线: (010)81055316

反盗版热线: (010)81055315

广告经营许可证: 京东工商广字第 8052 号

前 言

2007年11月5日谷歌公司正式向外界展示了基于Linux平台的开源手机操作系统Android。该平台由操作系统、中间件、用户界面和应用软件组成，号称首个为移动终端打造的真正开放和完整的移动软件平台。随着Android系统应用的不断扩大，Android系统在移动设备、可穿戴设备上的开发越来越多，本书就是为了适应新智能设备的开发特意策划编写的。

本书的内容

本书共计29章，全面讲解了基于Android系统开发传感器程序和智能设备应用的技术，主要内容为Android技术核心框架分析、Android传感器系统分析、光线传感器详解、接近警报传感器详解、磁场传感器详解、加速度传感器详解、方向传感器详解、陀螺仪传感器详解和基于图像处理的人脸识别技术详解，以及智能家居系统等知识。

本书特色

本书内容丰富，分析细致。我们的目标是通过一本图书，提供多本图书的价值，在内容的编写上，本书具有以下特色。

(1) 遵循“基础讲解—技术剖析—实战演练”这一主线，全面剖析了与Android传感器和外设应用开发相关的核心技术，并通过具体实例进行了实践演练和举例说明。

(2) 实用性强。本书彻底摒弃了枯燥的理论和简单的操作，注重实用性和可操作性，用通俗的语言详细讲解了各个知识点的基本原理。

(3) 内容全面。本书堪称市面上“内容较全面的一本Android传感器和外设应用开发书”，无论是获取源码，还是各个相关技术的运作机制和具体使用方法，在本书中您都能找到解决问题的答案。

读者对象

初学Android编程的自学者

Linux开发人员

大中专院校的老师 and 学生

毕业设计的学生

Android编程爱好者

相关培训机构的老师和学员

从事Android开发的程序员

Android智能穿戴设备开发工程师

Android外部设备开发工程师

本书在编写过程中，得到了人民邮电出版社工作人员的大力支持，正是各位编辑的求实、耐心和效率，才使得本书在这么短的时间内出版。另外，也十分感谢我的家人在我写作的时候给予的巨大支持。由于本人水平有限，书中的纰漏和不尽如人意之处在所难免，诚请读者提出意见和建议，以便修订并使之更臻完善。另外，我们也提供了售后支持和源程序下载网站：<http://www.toppr.net/>，读者如有疑问可以在此提出，一定会得到满意的答复。编辑联系邮箱：zhangtao@ptpress.com.cn。

编 者

目 录

第 1 章 Android 开发技术基础	1	3.5.2 获取 MSM 内核代码	34
1.1 智能手机操作系统介绍	1	3.5.3 获取 OMAP 内核代码	35
1.2 Android 的巨大优势	2	3.5.4 编译 Android 的 Linux 内核	35
1.2.1 系出名门	2	3.6 编译源码生成 SDK	36
1.2.2 强大的开发团队	2	第 4 章 Android 技术核心框架分析	40
1.2.3 Android 系统开源	2	4.1 分析 Android 的系统架构	40
第 2 章 搭建 Android 应用开发环境	3	4.1.1 Android 体系结构介绍	40
2.1 安装 Android SDK 的系统要求	3	4.1.2 Android 应用工程文件组成	42
2.2 安装 JDK	3	4.2 Android 的五大组件	45
2.3 安装 Eclipse 和 Android SDK	6	4.2.1 Activity 界面组件	45
2.3.1 获取并安装 Eclipse 和 Android SDK	6	4.2.2 Intent 切换组件	45
2.3.2 快速安装 SDK	8	4.2.3 Service 服务组件	45
2.4 安装 ADT	8	4.2.4 用 Broadcast/Receiver 广播 机制组件	46
2.5 验证设置	10	4.2.5 ContentProvider 存储组件	46
2.5.1 设定 Android SDK Home	10	4.3 进程和线程	46
2.5.2 验证开发环境	10	4.3.1 什么是进程	47
2.5.3 创建 Android 虚拟设备 (AVD)	11	4.3.2 什么是线程	47
2.6 启动 AVD 模拟器	12	4.3.3 应用程序的生命周期	47
2.6.1 模拟器和真机究竟有何区别	13	4.4 Android 和 Linux 的关系	49
2.6.2 启动 AVD 模拟器的基本流程	13	4.4.1 Android 继承于 Linux	50
第 3 章 获取并编译源码	14	4.4.2 Android 和 Linux 内核 的区别	50
3.1 在 Linux 系统获取 Android 源码	14	4.5 第一段 Android 程序	52
3.2 在 Windows 平台获取 Android 源码	15	4.5.1 新建 Android 工程	52
3.3 分析 Android 源码结构	17	4.5.2 调试程序	54
3.3.1 应用程序	18	4.5.3 运行程序	54
3.3.2 应用程序框架	19	第 5 章 Android 传感器系统分析	56
3.3.3 系统服务	20	5.1 Android 传感器系统概述	56
3.3.4 系统程序库	22	5.2 分析 Java 层	57
3.3.5 系统运行库	24	5.3 分析 Frameworks 层	62
3.3.6 硬件抽象层	25	5.3.1 监听传感器的变化	62
3.4 编译源码	26	5.3.2 注册监听	62
3.4.1 搭建编译环境	26	5.4 分析 JNI 层	71
3.4.2 开始编译	26	5.4.1 分析 android_hardware_ SensorManager.cpp	72
3.4.3 在模拟器中运行	28	5.4.2 处理客户端数据	75
3.4.4 常见的错误分析	28	5.4.3 处理服务端数据	77
3.4.5 实践演练——演示两种编译 Android 程序的方法	29	5.4.4 封装 HAL 层的代码	87
3.5 编译 Android Kernel	32	5.4.5 消息队列处理	91
3.5.1 获取 Goldfish 内核代码	32	5.5 分析 HAL 层	94

第 6 章 Android 传感器系统概览	102	9.1.3 公共方法	190
6.1 Android 设备的传感器系统	102	9.1.4 Geocoder 的主要功能	192
6.1.1 包含的传感器	102	9.1.5 地理编码和地理反编码	193
6.1.2 检测当前设备支持的传感器	103	9.2 实战演练——在设备地图中快速查询某个位置	195
6.2 使用 SensorSimulator	105	9.3 实战演练——接近某个位置时实现自动提醒	197
6.3 查看传感器的相关信息	107	第 10 章 磁场传感器详解	207
第 7 章 地图定位	120	10.1 磁场传感器基础	207
7.1 位置服务	120	10.1.1 什么是磁场传感器	207
7.1.1 android.location 功能类	120	10.1.2 磁场传感器的分类	207
7.1.2 实现定位服务功能	121	10.2 Android 系统中的磁场传感器	208
7.1.3 实战演练——在 Android 设备中实现 GPS 定位	123	10.3 实战演练——获取磁场传感器的 3 个分量	209
7.2 及时更新位置信息	125	10.4 实战演练——演示常用传感器的基本用法	210
7.2.1 Maps 库类	125	10.4.1 实现布局文件	210
7.2.2 使用 LocationManager 及时监听	126	10.4.2 实现程序文件	210
7.2.3 实战演练——在 Android 设备中显示当前位置的坐标和海拔	127	第 11 章 加速度传感器详解	217
7.3 在 Android 设备中使用地图	134	11.1 加速度传感器基础	217
7.3.1 准备工作	134	11.1.1 加速度传感器的分类	217
7.3.2 使用 Map API 密钥的基本流程	135	11.1.2 加速度传感器的主要应用领域	218
7.3.3 实战演练——在 Android 设备中使用谷歌地图实现定位	137	11.2 Android 系统中的加速度传感器	219
7.4 在 Android 设备中实现地址查询	141	11.2.1 实战演练——获取 x 、 y 、 z 轴的加速度值	220
7.5 在 Android 设备中实现路径导航	144	11.2.2 实战演练——实现控件的抖动效果	222
第 8 章 光线传感器详解	149	11.2.3 实战演练——实现仿微信“摇一摇”效果	224
8.1 光线传感器基础	149	11.3 线性加速度传感器详解	229
8.1.1 光线传感器介绍	149	11.3.1 线性加速度传感器的原理	229
8.1.2 在 Android 中使用光线传感器的方法	150	11.3.2 实战演练——测试小球的运动	230
8.2 实战演练——获取设备中光线传感器的值	151	第 12 章 方向传感器详解	235
8.3 实战演练——显示设备中光线传感器的强度	152	12.1 方向传感器基础	235
8.4 实战演练——显示设备名称和光线强度	154	12.1.1 方向传感器必备知识	235
8.5 实战演练——智能楼宇灯光控制系统	155	12.1.2 Android 中的方向传感器	236
8.5.1 布局文件	156	12.2 实战演练——测试当前设备的 3 个方向值	236
8.5.2 实现程序文件	166	12.2.1 实现布局文件	236
第 9 章 接近警报传感器详解	190	12.2.2 实现主程序文件	237
9.1 类 Geocoder 详解	190	12.3 实战演练——开发一个指南针程序	238
9.1.1 类 Geocoder 基础	190	12.3.1 实现布局文件	238
9.1.2 公共构造器	190	12.3.2 实现程序文件	238

12.4	开发一个具有定位功能的指南针	240	16.2	实战演练——开发一个 Android 气压计系统	348
12.4.1	实现布局文件	240	16.2.1	编写插件调用文件	348
12.4.2	实现程序文件	242	16.2.2	编写 Cordova 插件文件	349
第 13 章	陀螺仪传感器详解	251	16.2.3	定义每个时间点的压力值	351
13.1	陀螺仪传感器基础	251	16.2.4	监听传感器传来的和存储的新压力值	351
13.2	Android 中的陀螺仪传感器	252	16.3	实战演练——获取当前相对海拔和绝对海拔的数据	355
13.2.1	陀螺仪传感器和加速度传感器的对比	252	16.3.1	实现布局文件	355
13.2.2	智能设备中的陀螺仪传感器	252	16.3.2	实现主 Activity	357
13.3	实战演练——联合使用互补滤波器和陀螺仪传感器	254	第 17 章	温度传感器详解	364
13.3.1	实现布局文件	255	17.1	温度传感器基础	364
13.3.2	实现主 Activity 文件	260	17.2	Android 系统中的温度传感器	364
13.3.3	实现均值滤波器	266	17.3	实战演练——让 Android 设备变为温度计	366
13.3.4	测量各个平面的值	267	17.3.1	实现布局文件	367
13.3.5	传感器处理	278	17.3.2	检测温度传感器的温度变化	367
第 14 章	旋转向量传感器详解	290	17.4	实战演练——电池温度测试仪	368
14.1	Android 中的旋转向量传感器	290	17.4.1	实现布局文件	368
14.2	实战演练——确定设备当前的具体方向	291	17.4.2	实现程序文件	369
14.2.1	实现主 Activity	291	17.5	实战演练——测试温度、湿度、光照和压力	377
14.2.2	获取设备的旋转向量	298	17.5.1	实现 Arduino 文件	377
第 15 章	距离传感器详解	301	17.5.2	实现 Android APP	385
15.1	距离传感器基础	301	第 18 章	湿度传感器详解	388
15.1.1	距离传感器介绍	301	18.1	湿度传感器基础	388
15.1.2	Android 系统中的距离传感器	302	18.2	Android 系统中的湿度传感器	389
15.2	实战演练——使用距离传感器实现自动锁屏功能	303	18.3	实战演练——获取远程湿度传感器的数据	389
15.3	实战演练——根据设备的距离实现自动锁屏功能	308	18.3.1	编写布局文件	390
15.4	实战演练——绘制运动曲线	310	18.3.2	监听用户触摸单击屏幕控件事件并处理	391
15.4.1	实现布局文件	310	18.3.3	设置远程湿度传感器的初始 URL 地址	393
15.4.2	实现 Activity 程序文件	312	18.4	实战演练——开发一个湿度测试仪	394
15.4.3	实现监听事件处理	316	18.4.1	实现主界面	394
15.5	实战演练——开发一个健身计步器	319	18.4.2	设置具体值	397
15.5.1	系统功能模块介绍	319	18.4.3	显示当前的值	401
15.5.2	系统主界面	319	18.4.4	保存当前数值	404
15.5.3	系统设置模块	329	18.4.5	图形化显示测试结果	405
第 16 章	气压传感器详解	347	18.4.6	湿度跟踪器	414
16.1	气压传感器基础	347	第 19 章	Android 蓝牙系统概述	416
16.1.1	什么是气压传感器	347	19.1	蓝牙概述	416
16.1.2	气压传感器在智能手机中的应用	347			

19.1.1	蓝牙技术的发展历程	416	20.4	蓝牙规范	455
19.1.2	蓝牙的特点	416	20.4.1	Bluetooth 系统中的 常用规范	455
19.2	Android 系统中的蓝牙模块	416	20.4.2	蓝牙协议体系结构	456
19.3	分析蓝牙模块的源码	418	20.4.3	低功耗 (BLE) 蓝牙协议	457
19.3.1	初始化蓝牙芯片	418	20.4.4	现有的基于 GATT 的 协议/服务	457
19.3.2	蓝牙服务	418	20.4.5	双模协议栈	458
19.3.3	管理蓝牙电源	419	20.4.6	单模协议栈	458
19.4	和蓝牙相关的类	419	20.5	低功耗蓝牙协议栈详解	459
19.4.1	BluetoothSocket 类	419	20.5.1	蓝牙协议栈基础	459
19.4.2	BluetoothServerSocket 类	421	20.5.2	蓝牙协议体系中的协议	460
19.4.3	BluetoothAdapter 类	421	20.5.3	Android 的低功耗蓝牙 协议栈	461
19.4.4	BluetoothClass.Service 类	428	20.6	TI 公司的低功耗蓝牙	462
19.4.5	BluetoothClass.Device 类	428	20.6.1	获取 TI 公司的低功耗 蓝牙协议栈	462
19.5	在 Android 平台开发蓝牙 应用程序	429	20.6.2	分析 TI 公司的低功耗 蓝牙协议栈	463
19.6	实战演练——开发一个控制玩具 车的蓝牙遥控器	432	20.7	使用蓝牙控制电风扇	469
19.7	实战演练——开发一个蓝牙 控制器	438	20.7.1	准备 DHT 传感器	469
19.7.1	界面布局	439	20.7.2	实现 Android 测试 APP	473
19.7.2	响应单击按钮	440	第 21 章	语音识别技术详解	479
19.7.3	和指定的服务器建立连接	441	21.1	语音识别技术基础	479
19.7.4	搜索附近的蓝牙设备	442	21.1.1	语音识别的发展历史	479
19.7.5	建立和 OBEX 服务器的 数据传输	443	21.1.2	技术发展历程	480
19.7.6	实现蓝牙服务器端的 数据处理	446	21.2	Text-To-Speech 技术详解	480
第 20 章	低功耗蓝牙技术详解	448	21.2.1	Text-To-Speech 基础	480
20.1	短距离无线通信技术概览	448	21.2.2	Text-To-Speech 的 实现流程	481
20.1.1	ZigBee——低功耗、 自组网	448	21.2.3	实战演练——使用 Text-To-Speech 技术实现 语音识别	483
20.1.2	Wi-Fi——大带宽支持 家庭互联	449	21.3	Voice Recognition 技术详解	484
20.1.3	蓝牙——4.0 进入 低功耗时代	449	21.3.1	Voice Recognition 技术 基础	484
20.1.4	NFC——近场通信	449	21.3.2	实战演练——使用 Voice Recognition 技术 实现语音识别	486
20.2	蓝牙 4.0 BLE 基础	450	21.4	实战演练——开发一个语音 识别系统	489
20.2.1	蓝牙 4.0 的优势	450	21.4.1	验证是否支持所需要 的语言	489
20.2.2	Bluetooth 4.0 BLE 推动了 智能设备的兴起	451	21.4.2	实现 TTS 的初始化工作	489
20.3	低功耗蓝牙基础	452	21.4.3	开启语言检查功能	491
20.3.1	低功耗蓝牙的架构	452	21.4.4	跟踪语言数据的安装状况	492
20.3.2	低功耗蓝牙分类	452	21.4.5	转换语言并处理结果	493
20.3.3	集成方式	453			
20.3.4	低功耗蓝牙的特点	454			
20.3.5	BLE 和传统蓝牙 BR/EDR 技术的对比	454			

21.4.6	实现语音阅读测试	495	第 24 章	拍照解析条形码技术详解	566
21.4.7	保证系统可以实现正确的语音识别	499	24.1	Android 拍照系统介绍	566
21.4.8	显示语音识别的结果	501	24.1.1	分析拍照系统的底层程序	568
21.4.9	处理回调	502	24.1.2	分析拍照系统的硬件抽象层	574
第 22 章	手势识别技术详解	508	24.1.3	分析拍照系统的 Java 部分	577
22.1	手势识别技术基础	508	24.2	开发拍照应用程序	582
22.1.1	类 GestureDetector 基础	508	24.2.1	通过 Intent 调用系统的照相机 Activity	583
22.1.2	使用类 GestureDetector	509	24.2.2	调用 Camera API 拍照	583
22.1.3	手势识别处理事件和方法	511	24.2.3	总结 Camera 拍照的流程	584
22.2	实战演练——通过触摸方式移动图片	512	24.2.4	实战演练——获取系统现有相机拍摄的图片	586
22.2.1	实例说明	512	24.2.5	实战演练——使用 Camera 预览并拍照	590
22.2.2	具体实现	512	24.2.6	实战演练——使用 Camera API 方式拍照	594
22.3	实战演练——实现各种手势识别	515	24.3	使用拍照方式解析二维码	596
22.4	实战演练——实现手势拖动和缩放图片效果	517	24.3.1	QR Code 码的特点	596
22.4.1	实现布局文件	518	24.3.2	实战演练——使用 Android 相机解析二维码	597
22.4.2	监听用户选择的设置选项	518	第 25 章	麦克风音频录制技术详解	604
22.4.3	获取并设置移动位置和缩放值	519	25.1	使用 MediaRecorder 接口录制音频	604
22.4.4	在不同的缩放状态下绘制图像视图	520	25.1.1	类 MediaRecorder 详解	604
22.4.5	根据监听到的手势实现图片缩放	522	25.1.2	实战演练——使用 MediaRecorder 录制音频	605
第 23 章	NFC 近场通信技术详解	524	25.2	使用 AudioRecord 接口录制音频	609
23.1	近场通信技术基础	524	25.2.1	AudioRecord 的常量	609
23.1.1	NFC 技术的特点	524	25.2.2	AudioRecord 的构造函数	610
23.1.2	NFC 的工作模式	524	25.2.3	AudioRecord 的公共方法	610
23.1.3	NFC 和蓝牙的对比	525	25.2.4	AudioRecord 的受保护方法	612
23.2	射频识别技术详解	525	25.2.5	实战演练——使用 AudioRecord 录制音频	612
23.2.1	RFID 技术简介	526	25.3	实战演练——麦克风录音综合实例	614
23.2.2	RFID 技术的组成	526	25.3.1	获取录音源的最大振幅	614
23.2.3	RFID 技术的特点	526	25.3.2	实现异步音频录制功能	616
23.2.4	RFID 技术的工作原理	527	25.3.3	监听是否超越最大值	618
23.3	Android 系统中的 NFC	528	25.3.4	录制音频	618
23.3.1	分析 Java 层	528	25.3.5	巨响检测	621
23.3.2	分析 JNI 部分	548	25.3.6	检测一致性频率	622
23.3.3	分析底层	556	第 26 章	基于图像处理的人脸识别技术详解	624
23.4	编写 NFC 程序	556	26.1	二维图形处理详解	624
23.4.1	在 Android 系统编写 NFC APP 的方法	556			
23.4.2	实战演练——使用 NFC 发送消息	558			
23.4.3	实战演练——使用 NFC 读写 Mifare Tag	562			

26.1.1	类 Graphics 基础	624	28.3.1	布局文件	678
26.1.2	实战演练——使用 Graphics 类	624	28.3.2	程序文件	680
26.1.3	实战演练——使用 Color 类和 Paint 类实现绘图处理	626	28.4	实现公共类	691
26.2	二维动画处理详解	628	28.4.1	核心公共类 Jamendo Application	691
26.2.1	类 Drawable 详解	628	28.4.2	缓存图片资源	694
26.2.2	实现 Tween Animation 动画	629	28.4.3	类 RequestCache	694
26.2.3	实战演练——实现 Tween 动画效果	630	28.5	手势操作	695
26.2.4	实战演练——使用 Tween Animation 实现 Tween 动画效果	631	28.5.1	Android 提供的手势操作 API	695
26.2.5	实现 Frame Animation 动画效果	632	28.5.2	使用命令模式构建手势识别系统	698
26.2.6	实战演练——播放 GIF 动画	633	28.5.3	实现抽象命令角色 Command	698
26.3	Android 人脸识别技术详解	634	28.5.4	实现具体命令角色 ConcreteCommand	698
26.3.1	分析人脸识别模块的源码	634	28.5.5	实现命令接收者角色 Receiver	699
26.3.2	实战演练——使用内置模块实现人脸识别	635	28.5.6	实现调用者角色 Invoker	700
26.3.3	实战演练——实现人脸识别	636	28.5.7	实现装配者角色 Client	701
26.3.4	实战演练——从照片中取出人脸	640	28.6	播放处理	701
第 27 章	行走轨迹记录器	642	28.6.1	设计播放界面	702
27.1	系统功能模块介绍	642	28.6.2	分析播放流程	702
27.2	系统主界面	642	第 29 章	智能家居系统	710
27.2.1	布局文件	642	29.1	需求分析	710
27.2.2	实现主 Activity	645	29.1.1	背景介绍	710
27.3	系统设置	658	29.1.2	传感技术的推动	710
27.3.1	选项设置	659	29.1.3	Android 与智能家居的紧密联系	711
27.3.2	生成 GPX 文件和 KML 文件	661	29.2	系统功能模块介绍	711
27.4	邮件分享提醒	665	29.3	系统主界面	711
27.4.1	基本邮箱箱设置	665	29.3.1	实现布局文件	711
27.4.2	实现邮件发送功能	668	29.3.2	实现程序文件	712
27.5	上传 OSM 地图	671	29.4	系统设置	714
27.5.1	授权提示布局文件	671	29.4.1	总体配置	714
27.5.2	实现文件上传	673	29.4.2	系统总体配置	714
第 28 章	手势音乐播放器	675	29.4.3	构建数据库	719
28.1	系统功能模块介绍	675	29.5	电器控制模块	721
28.2	系统主界面	675	29.5.1	电器控制主界面	721
28.3	系统列表界面	678	29.5.2	温度控制界面	723
			29.5.3	电灯控制界面	724
			29.6	预案管理模块	727
			29.6.1	天气情况	727
			29.6.2	历史数据	735
			29.6.3	系统设置	737

第 1 章 Android 开发技术基础

Android 是科技界巨头谷歌公司推出的一款智能设备操作系统,是以 Linux 开源系统架构为基础的。Android 功能十分强大。从 2008 年推出到现在,Android 一直在全球智能手机操作系统市场中占居第一的宝座。本章简单介绍 Android 的发展历程和背景,并介绍智能设备和 Android 的密切关系,为读者步入本书后面知识的学习打下基础。

1.1 智能手机操作系统介绍

在 Android 系统诞生之前,智能手机这个新鲜事物大大丰富了人们的生活,得到了广大手机用户的青睐。各大手机厂商在利益的驱动之下,纷纷建立了各种智能手机操作系统,并全力抢夺市场份额。Android 系统就是在这个风起云涌的时代背景下诞生的。

何谓智能手机

智能手机是指具有像个人计算机那样强大的功能,拥有独立的操作系统,用户可以自行安装应用软件、游戏等第三方服务商提供的程序,并且可以通过移动通信网络接入到无线网络中。在 Android 系统诞生之前已经有很多优秀的智能手机操作系统产品,例如,家喻户晓的 Symbian 系列和微软的 Windows Mobile 系列等。

对于初学者来说,可能还不知道怎样来区分什么是智能手机。某大型专业统计公司曾经为智能手机的问题做过一项市场调查,经过大众讨论并投票之后,总结出了智能手机所必须具备的功能标准,其中下面是当时投票后得票率最高的前 5 个选项:

- (1) 操作系统必须支持新应用的安装;
- (2) 高速度处理芯片;
- (3) 支持播放式的手机电视;
- (4) 具有大容量存储芯片和存储扩展能力;
- (5) 支持 GPS 导航。

根据大众投票结果,手机联盟制定了一个标准。并根据这个标准为基础,总结出了如下智能手机的主要特点:

- (1) 具备普通手机的全部功能,例如可以进行正常的通话和发短信等手机应用;
- (2) 是一个开放性的操作系统,在系统平台上可以安装更多的应用程序,从而实现功能的无限扩充;
- (3) 具备上网功能;
- (4) 具备 PDA 的功能,实现个人信息管理、日程记事、任务安排、多媒体应用和浏览网页;
- (5) 可以根据个人需要扩展机器的功能;
- (6) 扩展性能强,并且可以支持很多第三方软件。

1.2 Android 的巨大优势

为什么 Android 能在这么多的智能系统中脱颖而出，成为市场占有率第一的手机系统呢？要想分析其原因，需要先了解它的巨大优势，分析究竟是哪些优点吸引了厂商和消费者的青睐。本节将对上述问题一一解答。

1.2.1 系出名门

Android 是出身于 Linux 世家，是一款开源的手机操作系统。Android 功成名就之后，各大手机联盟纷纷加入，这个联盟由中国移动、摩托罗拉、高通、HTC 和 T-Mobile 在内的 30 多家技术和无线应用的领军企业组成。通过与运营商、设备制造商、开发商和其他有关各方结成深层次的合作伙伴关系，希望借助建立标准化、开放式的移动电话软件平台，在移动产业内形成一个开放式的生态系统。

1.2.2 强大的开发团队

Android 的研发队伍阵容强大，包括摩托罗拉、Google、HTC（宏达电子）、PHILIPS、T-Mobile、高通、魅族、三星、LG 以及中国移动在内的 34 家企业，这都是在手机开发生产领域中享誉盛名的企业。它们都将基于该操作平台开发手机的新型业务，应用之间的通用性和互联性将在最大程度上得到保持。此外，相关生产厂家还成立了手机开放联盟，联盟中的成员名单如下所示。

1. 手机制造商

中国台湾宏达国际电子（HTC）（Palm 等多款智能手机的代工厂）、摩托罗拉（美国最大的手机制造商）、韩国三星电子（仅次于诺基亚的全球第二大手机制造商）、韩国 LG 电子、中国移动（全球最大的移动运营商）、日本 KDDI（2900 万用户）、日本 NTT DoCoMo（5200 万用户）、美国 Sprint Nextel（美国第三大移动运营商，5400 万用户）、意大利电信（Telecom Italia）（意大利主要的移动运营商，3400 万用户）、西班牙 Telefónica（在欧洲和拉美有 1.5 亿用户）、T-Mobile（德意志电信旗下公司，在美国和欧洲有 1.1 亿用户）。

2. 半导体公司

Audience Corp（声音处理器公司）、Broadcom Corp（无线半导体主要提供商）、英特尔（Intel）、Marvell Technology Group、Nvidia（图形处理器公司）、SiRF（GPS 技术提供商）、Synaptics（手机用户界面技术）、德州仪器（Texas Instruments）、高通（Qualcomm）、惠普 HP（Hewlett-Packard Development Company,L.P）。

3. 软件公司

Aplix、Ascender、eBay 的 Skype、Esmertec、Living Image、NMS Communications、Noser Engineering AG、Nuance Communications、PacketVideo、SkyPop、Sonix Network、TAT-The Astonishing Tribe、Wind River Systems。

1.2.3 Android 系统开源

开源意味着对开发人员和手机厂商来说，Android 是完全无偿免费使用的。因为源代码公开的原因，所以吸引了全世界各地无数程序员的热情。于是很多手机厂商都纷纷采用 Android 作为自己产品的系统。因为免费，所以降低了成本，提高了利润。而对于开发人员来说，众多厂商的采用就意味着人才需求大，所以纷纷加入到 Android 开发大军中来。于是有一些干的还可以的程序员禁不住高薪的诱惑，都纷纷改行做 Android 开发。至于“混”的不尽如人意的程序员，就更加坚定了“改行做 Android 手机开发”，目的是想寻找自己程序员生涯的转机。而像笔者这样遇到发展瓶颈的程序员，也决定做 Android 开发，因为这样可以学习一门新技术，使自己的未来更加有保障。

第2章 搭建 Android 应用开发环境

“工欲善其事，必先利其器”出自《论语》，意思是要想高效地完成一件事，需要有一个合适的工具。对于安卓开发人员来说，开发工具同样至关重要。作为一项新兴技术，在进行开发前首先要搭建一个对应的开发环境。而在搭建开发环境前，需要了解安装开发工具所需要的硬件和软件配置条件。Android 开发包括底层开发和应用开发，底层开发大多数是指和硬件相关的开发，并且是基于 Linux 环境的，例如开发驱动程序。应用开发是指开发能在 Android 系统上运行的程序，例如游戏、地图等程序。本书的重点是讲解多媒体应用开发，即使讲一些底层的知识，也是为上层的应用服务的。因为开发 Android 智能设备程序既需要底层开发知识，也需要上层应用开发的知识。所以在本书需要讲解底层和应用开发环境的搭建知识。本章将首先介绍在 Windows 下搭建 Android 应用开发环境的过程。

2.1 安装 Android SDK 的系统要求

在搭建之前，一定先确定基于 Android 应用软件所需要开发环境的要求，具体如表 2-1 所示。

表 2-1 开发系统所需求参数

项 目	版本要求	说 明	备 注
操作系统	Windows XP 或 Vista Mac OS X 10.4.8+Linux Ubuntu Drapper	根据自己的计算机自行选择	选择自己最熟悉的操作系统
软件开发包	Android SDK	选择最新版本的 SDK	截止到目前，最新手机版本是 2.3
IDE	Eclipse IDE+ADT	Eclipse3.3 (Europa), 3.4 (Ganymede) ADT (Android Development Tools) 开发插件	选择“for Java Developer”
其他	JDK Apache Ant	Java SE Development Kit 5 或 6 Linux 和 Mac 上使用 Apache Ant 1.6.5+, Windows 上使用 1.7+版本	(单独的 JRE 是不可以的，必须要有 JDK)，不兼容 Gnu Java 编译器 (gcj)

Android 工具是由多个开发包组成的，具体说明如下。

- JDK：可以到网址 <http://java.sun.com/javase/downloads/index.jsp> 处下载。
- Eclipse (Europa)：可以到网址 <http://www.eclipse.org/downloads/> 下载 Eclipse IDE for Java Developers。
 - Android SDK：可以到网址 <http://developer.android.com> 下载。
 - 还有对应的开发插件。

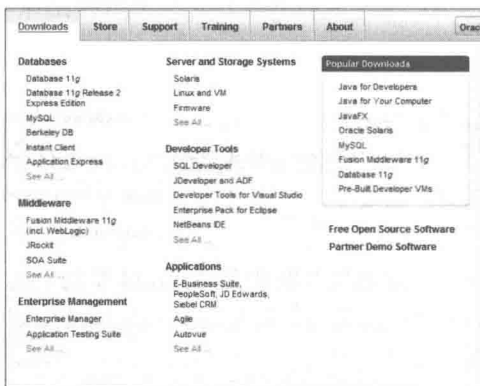
2.2 安装 JDK

JDK (Java Development Kit) 是整个 Java 的核心，包括了 Java 运行环境、Java 工具和 Java

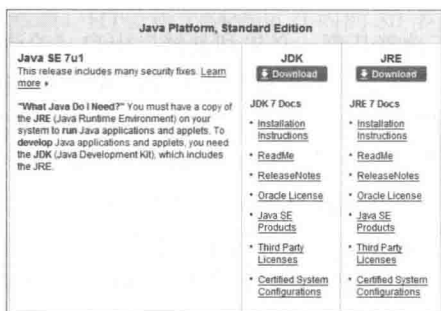
基础的类库。JDK 是学好 Java 的第一步，是开发和运行 Java 环境的基础，当用户要对 Java 程序进行编译的时候，必须先获得对应操作系统的 JDK，否则将无法编译 Java 程序。在安装 JDK 之前需要先获得 JDK，获得 JDK 的操作流程如下所示。

(1) 登录 Oracle 官方网站，网址为 <http://developers.sun.com/downloads/>，如图 2-1 所示。

(2) 在图 2-1 中可以看到有很多版本，在此选择当前最新的版本 Java 7，下载页面如图 2-2 所示。



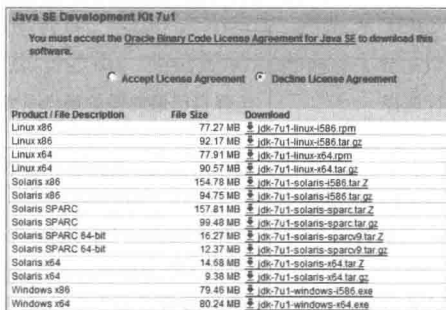
▲图 2-1 Oracle 官方下载页面



▲图 2-2 JDK 下载页面

(3) 在图 2-2 中单击 JDK 下方的“Download”按钮，在弹出的新界面中选择将要下载的 JDK，本书在此选择的是 Windows X86 版本，如图 2-3 所示。

(4) 下载完成后双击下载的“.exe”文件开始进行安装，将弹出“安装向导”对话框，在此单击“下一步”按钮，如图 2-4 所示。



▲图 2-3 选择 Windows X86 版本



▲图 2-4 “许可证协议”对话框

(5) 弹出“安装路径”对话框，在此选择文件的安装路径，如图 2-5 所示。

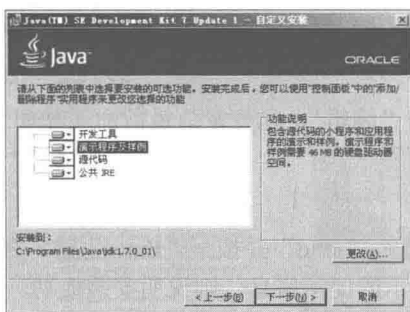
(6) 在此设置安装路径是“E:\jdk1.7.0_01”，然后单击“下一步”按钮开始在安装路径解压缩下载的文件，如图 2-6 所示。

(7) 完成后弹出“目标文件夹”对话框，在此选择要安装的位置，如图 2-7 所示。

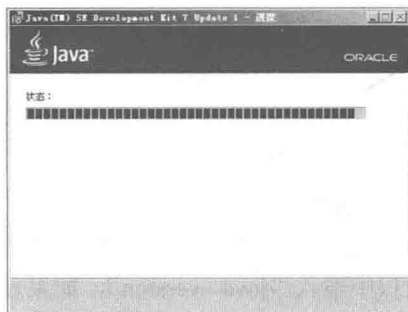
(8) 单击“下一步”按钮后开始正式安装，如图 2-8 所示。

(9) 完成后弹出“完成”对话框，单击“完成”按钮后完成整个安装过程，如图 2-9 所示。

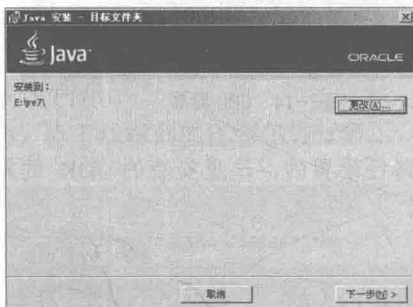
完成安装后可以检测是否安装成功，检测方法是依次单击“开始”|“运行”，在运行框中输入“cmd”并按下“Enter”键，在打开的 CMD 窗口中输入“java -version”，如果显示图 2-10 所示的提示信息，则说明安装成功。



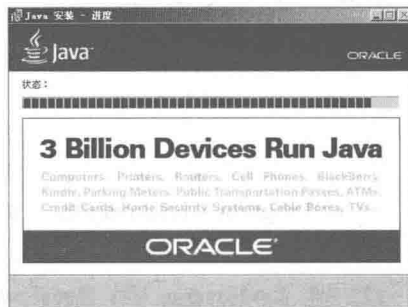
▲图 2-5 “安装路径”对话框



▲图 2-6 解压缩下载的文件



▲图 2-7 “目标文件夹”对话框



▲图 2-8 继续安装



▲图 2-9 完成安装



▲图 2-10 CMD 窗口



注意

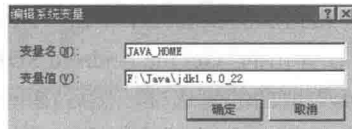
完成安装后可以检测是否安装成功，方法是依次单击“开始”|“运行”，在运行框中输入“cmd”并按下“Enter”键，在打开的CMD窗口中输入“java -version”，如果显示图 2-11 所示的提示信息，则说明安装成功。

如果检测没有安装成功，需要将其目录的绝对路径添加到系统的 PATH 中。具体做法如下所示。

(1) 右键单击“我的电脑”，依次选择“属性”|“高级”，单击下面的“环境变量”，在下面的“系统变量”处选择新建，在变量名处输入 JAVA_HOME，变量值中输入刚才的目录，例如设置为“F:\Java\jdk1.6.0_22”。如图 2-12 所示。



▲图 2-11 CMD 窗口



▲图 2-12 设置系统变量

(2) 再次新建一个变量名为 `classpath`，其变量值如下所示。

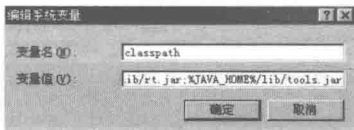
```
.;%JAVA_HOME%/lib/rt.jar;%JAVA_HOME%/lib/tools.jar
```

单击“确定”按钮找到 `PATH` 的变量，双击或单击“编辑”，在变量值最前面添加如下值。

```
%JAVA_HOME%/bin;
```

具体如图 2-13 所示。

(3) 再依次单击“开始”|“运行”，在运行框中输入“`cmd`”并按下“Enter”键，在打开的 `CMD` 窗口中输入“`java -version`”，如果显示图 2-14 所示的提示信息，则说明安装成功。



▲图 2-13 设置系统变量



▲图 2-14 CMD 界面

注意

上述变量设置中，是按照个人的安装路径设置的，这里安装的 `JDK` 的路径是 `C:\Program Files\Java\jdk1.7.0_02`。

2.3 安装 Eclipse 和 Android SDK

在安装好 `JDK` 后，接下来需要安装 `Eclipse` 和 `Android SDK`。`Eclipse` 是进行 `Android` 应用开发的一个集成工具，而 `Android SDK` 是开发 `Android` 应用程序所必须具备的框架。在 `Android` 官方公布的最新版本中，已经将 `Eclipse` 和 `Android SDK` 这两个工具进行了集成，一次下载即可同时获得这两个工具。

2.3.1 获取并安装 Eclipse 和 Android SDK

获取并安装 `Eclipse` 和 `Android SDK` 的具体步骤如下所示。

(1) 登录 `Android` 的官方网站 <http://developer.android.com/index.html>，如图 2-15 所示。

(2) 单击图 2-15 左上方“`Developers`”右侧的 \vee 符号，在弹出的界面中单击“`Get the SDK`”链接。如图 2-16 所示。



▲图 2-15 Android 的官方网站



▲图 2-16 单击“Get the SDK”链接

(3) 在弹出的新页面中单击“`Download the SDK`”按钮，如图 2-17 所示。

(4) 在弹出的“`Get the Android SDK`”界面中选中“`I have read and agree with the above terms and conditions`”前面的复选框，然后在下面的单选按钮中选择系统的位数。如果机器是 32 位的，就可以选中“`32-bit`”前面的单选按钮。如图 2-18 所示。



▲图 2-17 单击“Download the SDK”按钮



▲图 2-18 “Get the Android SDK”界面

(5) 单击图 2-18 中的 **Download the SDK ADT Bundle for Windows** 按钮后开始下载工作，下载的目标文件是一个压缩包。如图 2-19 所示。

(6) 将下载得到的压缩包进行解压，解压后的目录结构如图 2-20 所示。

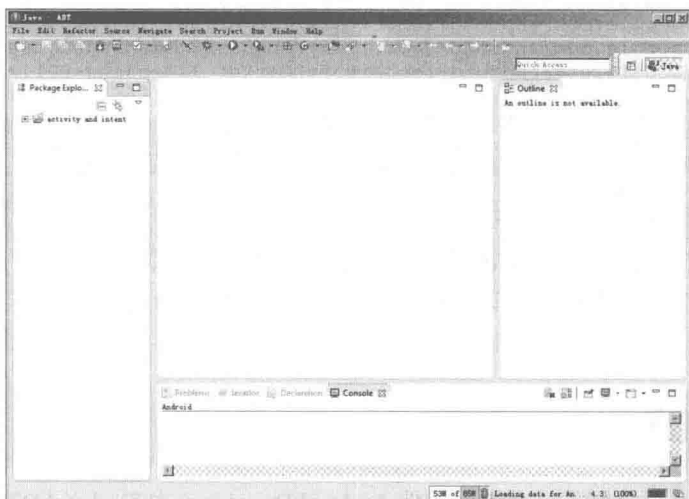


▲图 2-19 开始下载目标文件压缩包


eclipse	2013/10/11 16:28	文件夹
sdk	2013/7/26 18:16	文件夹
SDK Manager.exe	2013/5/22 18:57	应用程序

▲图 2-20 解压后的目录结构

由此可见，Android 官方已经将 Eclipse 和 Android SDK 实现了集成。双击“eclipse”目录中的“eclipse.exe”可以打开 Eclipse，界面效果如图 2-21 所示。



▲图 2-21 打开 Eclipse 后的界面效果

(7) 打开 Android SDK 的方法有两种，第一种是双击下载目录中的“SDK Manager.exe”文件，第二种是在 Eclipse 工具栏中单击  图标。打开后的效果如图 2-22 所示。