



Android

底层接口与驱动开发技术详解

The Technology Development Detailed Annotation of
Device Driver and Lower Layer Interface in Android

陈强 编著



底层驱动开发是通往Android殿堂级高手的
必经之路



遵循“内核分析-驱动实现-JNI层分析-Java
应用层-系统移植”主线



细致分析底层驱动, 提炼Android系统的本
质与精华

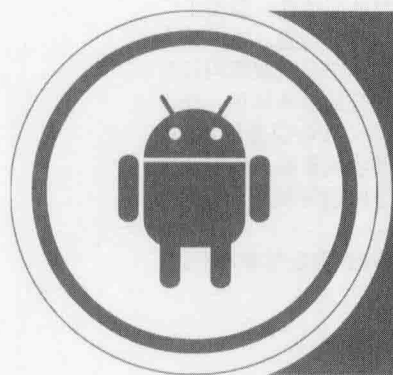


完全从实际需要出发, 作者十年开源系统开
发经验和盘托出

内容简介

目录

前言



Android

底层接口与驱动开发技术详解

陈强 编著



中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

Android 系统从诞生到现在短短的几年时间, 凭借其操作易用性和开发的简洁性, 赢得了广大消费用户和开发者的支持, 目前一直位居智能手机操作系统市场占有率榜首的位置。本书分为三篇, 共计 24 章, 循序渐进地讲解了 Android 系统中底层驱动方面的知识。本书从获取源码和源码结构分析讲起, 依次讲解了基础知识篇、专用驱动篇和设备驱动篇三大部分的基本知识。在讲解每一个驱动时, 都从 Android 系统源码开始入手, 遵循“内核分析—驱动实现—JNI 层分析—Java 应用层—系统移植和改造”这一主线, 透彻地剖析了每一个驱动系统的完整实现流程。本书几乎涵盖了 Android 驱动方面的所有主要内容, 并且全书内容言简意赅, 讲解方法通俗易懂, 不但适用于应用高手们, 也适合初学者学习并消化。

本书定位于有一定基础的 Android 开发人员, 帮助其细致分析 Android 源码和底层驱动开发, 积累经验, 快速向 Android 殿堂级高手迈进。

图书在版编目 (CIP) 数据

Android 底层接口与驱动开发技术详解 / 陈强编著

— 2 版. — 北京: 中国铁道出版社, 2015. 3

ISBN 978-7-113-19795-7

I. ①A… II. ①陈… 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 (2015) 第 000718 号

书 名: Android 底层接口与驱动开发技术详解

作 者: 陈 强 编著

责任编辑: 荆 波

特邀编辑: 安云飞

责任印制: 赵星辰

读者服务热线: 010-63560056

封面设计: 多宝格·付巍

出版发行: 中国铁道出版社 (北京市西城区右安门西街 8 号 邮政编码: 100054)

印 刷: 三河市宏盛印务有限公司

版 次: 2012 年 8 月第 1 版 2015 年 3 月第 2 版 2015 年 3 月第 1 次印刷

开 本: 787 mm×1092 mm 1/16 印张: 41.5 字数: 977 千

书 号: ISBN 978-7-113-19795-7

定 价: 89.00 元

版权所有 侵权必究

凡购买铁道版图书, 如有印制质量问题, 请与本社读者服务部联系调换。电话: (010) 51873174

打击盗版举报电话: (010) 51873659

Android 是一款于 2007 年 11 月 5 日宣布基于 Linux 平台的开源手机操作系统的名称,该平台由操作系统、中间件、用户界面和应用软件组成,号称是首个为移动终端打造的真正开放和完整的移动软件。根据国际数据公司 (IDC) 公布的统计数据,在 2013 年第一季度,Android 和 iOS 系统装机量占到所有智能手机出货量的 92.3%。在 2014 年第一季度,安装 Android 系统的新智能手机数量跃升至 1.621 亿部,超过 2013 年同期的 9 030 万部。这意味着,在运往世界各地的所有新智能手机中,移动操作系统 Android 的市场占有率已经达到 75%,比 2012 年第一季度的 59.1% 有显著提高。75%,这是一个具有突出性优势的比重,足以令后面的追赶者汗颜。我们有理由相信,在未来一段时间内,Android 依旧牢牢地占据着智能手机操作系统第一的宝座。

市场需求分析

强大的市场占有率造就了更多开发人员关注于这款神奇的系统,同时 Android 系统的应用者对该系统的功能需求也与日俱增,这也说明应用市场正在渴求着更多的 Android 开发高手,因此更多稍有基础的 Android 开发人员需要不断掌握更多的 Android 关键技术,提升开发功能。

可以说底层驱动开发是通往 Android 殿堂级高手的必经之路!为了让广大初学者可以对 Android 系统实现“灵与肉”的感知,而不是停留在原始、抽象的原理和概念之上,本书对 Android 系统中的底层驱动方面的知识进行了细致的分析,这样做的目的是“提炼”出 Android 系统埋藏在深处的本质,了解这款神奇的系统究竟是怎样诞生的。并依此为基础,详细讲解了系统移植和驱动开发的基本原理和具体实现。相信阅读完本书,经“细嚼慢咽”并消化之后,您一定会成为殿堂级的 Android 高手。

本书的内容

本书内容分为三篇,共计 24 章,循序渐进地讲解了 Android 系统中底层驱动方面的知识。本书从获取源码和源码结构分析讲起,依次讲解了 Android 驱动开发基础,获取并编译 Android 源码,Android 驱动移植基础,分析三大内核系统,分析 JNI,硬件设备驱动程序,Binder 驱动程序,Logger 日志驱动,内存系统驱动,分析电源管理模块,分析低内存管理模块,分析物理内存驱动,分析 Time Device 驱动,分析调试机制驱动 Ram Console,分析时钟系统驱动 Alarm,分析 USB Gadget 驱动,FrameBuffer 显示系统驱动,输入系统驱动应用,音频系统驱动,视频输出系统驱动,振动器系统驱动,传感器系统驱动,照相机系统,Wi-Fi 系统、蓝牙系统和 GPS 系统等高级知识。本书几乎涵盖了 Android 源码中的所有主要内容,并且全书内容言简意赅,

讲解方法通俗易懂，不但适用于应用高手们，也适合初学者学习并消化。

本书的版本

Android 系统自 2008 年 9 月发布第一个版本 1.1 以来，截至 2014 年 6 月发布新版本 Preview L，一共存在十多个版本。由此可见，Android 系统升级频率较快，一年中最少有两个新版本诞生。如果过于追求新版本，就会造成力不从心的结果。在此建议广大读者，不必追求最新的版本，我们只需关注最流行的版本即可。据官方统计，截至 2014 年 3 月 25 日，占据前三位的版本分别是 Android 4.3，Android 4.4 和 Android 4.2，其实这三个版本的区别并不是很大，只是在某个领域的细节上进行了更新。因此，在本书中使用的版本是 Android 4.4。

对于 Android 4.4 来说，对应的 Linux 内核是 3.10，本书将以 Linux 3.10 和 Android 4.4 为对象，详细分析 Android 平台中驱动系统的实现和移植。

本书特色

本书内容丰富全面，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，读者可以根据自己的需要有选择地进行阅读。在内容的编写上，本书具有以下特色。

（1）结构合理

从用户的实际需要出发，科学安排知识结构，内容由浅入深，叙述清楚。全书详细地讲解了和 Android 应用开发有关的源码，内容循序渐进，由浅入深。

（2）遵循“内核分析—驱动实现—JNI 层分析—Java 应用层—系统移植”这一主线

为了使广大读者彻底弄清楚 Android 平台中的各个驱动系统，在讲解每一个驱动系统时，都从 Linux 内核开始讲起，依次剖析了驱动层实现、JNI 层分析、Java 应用和系统移植改造等内容。遵循了从底层到顶层，实现了驱动系统大揭秘的目标。

（3）易学易懂

本书内容条理清晰、语言简洁，可以帮助读者快速掌握每个知识点。使读者既可以按照本书编排的章节顺序进行学习，也可以根据自己的需求对某一章节进行有针对性的学习。和传统难懂的计算机书籍相比，阅读本书会带来很大的乐趣。

（4）实用性强

本书彻底摒弃枯燥的理论和简单的操作，注重实用性和可操作性，通过简洁的语言和细腻的笔法，详细讲解了各个知识点的基本知识。

（5）内容全面

本书可以号称“内容非常全面的一本 Android 驱动开发书”，无论是获取源码，还是各个常用、常见的驱动系统，您都能在本书中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；
- 大中专院校的老师和学生；
- 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。

本团队在编写过程中，得到了中国铁道出版社工作人员的大力支持，特别是荆波编辑的大力支持。正是各位编辑的求实、耐心和高效，才能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。由于本团队水平有限，如有纰漏和不尽如人意之处在所难免，诚请读者提出意见或建议，以便修订并使之更加完善。

编 者

2014 年 12 月

第 1 篇 基础知识篇

第 1 章 Android 驱动开发基础

1.1 认识什么是驱动	1
1.1.1 驱动程序的作用	1
1.1.2 PC 中的驱动	2
1.1.3 手机中的驱动程序	2
1.2 开源还是不开源	2
1.2.1 雾里看花的开源	3
1.2.2 从为什么选择 Java 谈为什么不开源驱动程序	3
1.2.3 对驱动开发者来说是一把双刃剑	4
1.3 Linux 开发基础	4
1.3.1 Linux 历史简介	4
1.3.2 主要版本	5
1.3.3 Linux 的巨大前景	5
1.4 Android 和 Linux 的关系	6
1.4.1 Android 继承于 Linux	6
1.4.2 Android 和 Linux 内核的区别	6
1.5 简析 Linux 内核	8
1.5.1 Linux 内核的体系结构	8
1.5.2 和 Android 密切相关的 Linux 内核知识	10
1.6 分析 Linux 内核源码	14
1.6.1 源码目录结构	14
1.6.2 浏览源码的工具	16
1.6.3 Linux 内核的显著特性	18
1.6.4 学习 Linux 内核的方法	24

第 2 章 获取并编译 Android 源码

2.1 获取 Android 源码	29
2.1.1 在 Linux 系统获取 Android 源码	29

2.1.2	在 Windows 平台获取 Android 源码.....	30
2.2	分析 Android 源码结构.....	36
2.3	编译 Android 源码.....	37
2.3.1	搭建编译环境.....	38
2.3.2	开始编译.....	39
2.3.3	在模拟器中运行.....	40
2.3.4	常见的错误分析.....	40
2.3.5	实践演练——演示两种编译 Android 程序的方法.....	41
2.4	编译 Android Kernel.....	44
2.4.1	获取 Goldfish 内核代码.....	45
2.4.2	获取 MSM 内核代码.....	47
2.4.3	获取 OMAP 内核代码.....	47
2.4.4	编译 Android 的 Linux 内核.....	48

第 3 章 Android 驱动移植基础

3.1	驱动开发需要做的工作.....	50
3.2	Android 移植基础.....	52
3.2.1	移植的任务.....	52
3.2.2	需要移植的内容.....	53
3.2.3	驱动开发需要做的工作.....	54
3.3	Android 对 Linux 的改造.....	55
3.3.1	Android 的独有驱动.....	55
3.3.2	Android 的核心驱动.....	55
3.3.3	Android 对 Linux 内核文件的改动.....	57
3.3.4	为 Android 构建 Linux 的操作系统.....	58
3.4	内核空间 and 用户空间之间接口.....	59
3.4.1	内核空间和用户空间的相互作用.....	59
3.4.2	系统和硬件之间的交互.....	59
3.4.3	使用 Relay 实现内核到用户空间的数据传输.....	61
3.5	三类驱动程序.....	65
3.5.1	字符设备驱动.....	65
3.5.2	块设备驱动.....	72
3.5.3	网络设备驱动.....	75

第 4 章 分析三大内核系统

4.1 Goldfish 内核和驱动解析	77
4.1.1 Goldfish 基础	78
4.1.2 Android 专用驱动简介	79
4.1.3 Ashmem 驱动介绍	82
4.1.4 Pmem 驱动介绍	82
4.1.5 Alarm 驱动程序	83
4.1.6 USB Gadget 驱动程序	83
4.1.7 Paranoid 驱动介绍	84
4.1.8 Goldfish 的设备驱动	85
4.2 MSM 内核和驱动解析	87
4.2.1 MSM 基础	87
4.2.2 移植 MSM 内核简介	90
4.2.3 移植 MSM	91
4.3 OMAP 内核和驱动解析	98
4.3.1 OMAP 基础	98
4.3.2 OMAP 内核	100
4.3.3 移植 OMAP 体系结构	101
4.3.4 移植 Android 专用驱动和组件	108
4.3.5 OMAP 的设备驱动	109

第 2 篇 专用驱动篇

第 5 章 分析 JNI

5.1 JNI 的本质	114
5.2 分析 Java 层	116
5.2.1 加载 JNI 库	116
5.2.2 实现扫描工作	118
5.2.3 读取并保存信息	119
5.2.4 删除不是 SD 卡中的文件信息	121
5.2.5 processDirectory	121
5.2.6 扫描函数 scanFile()	122
5.2.7 JNI 中的异常处理	122
5.3 分析 JNI 层	123

5.3.1	将 Native 对象的指针保存到 Java 对象.....	123
5.3.2	创建 Native 层的 MediaScanner 对象.....	124
5.4	Native (本地) 层.....	124
5.4.1	注册 JNI 函数.....	124
5.4.2	完成注册工作.....	126
5.4.3	动态注册.....	128
5.4.4	处理路径参数.....	130
5.4.5	扫描文件.....	130
5.4.6	添加 TAG 信息.....	131
5.4.7	总结函数 JNI_OnLoad()与函数 JNI_OnUnload()的用途.....	131
5.4.8	Java 与 JNI 基本数据类型转换.....	132
5.4.9	JNIEnv 接口.....	133
5.4.10	JNI 中的环境变量.....	135
5.5	Camera 模块的 JNI 实例分析.....	135
5.5.1	Java 层预览接口.....	135
5.5.2	注册预览的 JNI 函数.....	137
5.5.3	C/C++层的预览函数.....	140
5.6	开发自己的 JNI 程序.....	140

第 6 章 硬件设备驱动程序

6.1	初识 HAL 层.....	143
6.1.1	HAL 层简介.....	144
6.1.2	HAL_legacy 和 HAL 的对比.....	145
6.2	分析 HAL 层源码.....	145
6.2.1	分析 HAL module.....	146
6.2.2	文件 hardware.h.....	146
6.2.3	文件 hardware.c.....	148
6.3	分析硬件抽象层的加载过程.....	151
6.4	分析硬件访问服务.....	153
6.4.1	定义硬件访问服务接口.....	153
6.4.2	实现硬件访问服务.....	154
6.5	分析 mokoid 工程.....	155
6.5.1	直接调用 service 方法的实现代码.....	156
6.5.2	通过 Manager 调用 service 的实现代码.....	160
6.6	分析 Sensor (传感器) 在 HAL 层的实现.....	163

6.6.1 HAL 层的 Sensor 代码	163
6.6.2 Sensor 编程的流程	165
6.7 HAL 和系统移植	166
6.7.1 移植各个 Android 部件的方式	166
6.7.2 辅助工作	166

第 7 章 Binder 驱动程序

7.1 引出 Binder 机制	172
7.2 Binder 机制概述	173
7.3 分析 Binder 驱动程序的数据结构	175
7.4 设备初始化	185
7.5 打开 Binder 设备文件	186
7.6 实现内存映射	188
7.6.1 分析流程	188
7.6.2 分析具体代码	188
7.7 释放物理页面	193
7.8 处理内核缓冲区	193
7.8.1 分配内核缓冲区函数	193
7.8.2 添加到红黑树	195
7.8.3 释放内核缓冲区	196
7.8.4 查询内核缓冲区	198
7.9 Binder 封装库	198
7.9.1 类 BBinder	199
7.9.2 类 BpRefBase	202
7.9.3 类 IPCThreadState	203

第 8 章 Logger 日志驱动

8.1 Logger 系统基础	207
8.2 分析 Logger 驱动程序	207
8.2.1 分析文件 logger.h	208
8.2.2 分析文件 logger.c	209
8.3 日志库 liblog	223
8.4 分析日志写入接口	227
8.4.1 分析 C/C++ 层的写入接口	228
8.4.2 分析 Java 层的写入接口	229

第 9 章 内存系统驱动

9.1 分析 Android 的进程通信机制	238
9.1.1 Android 的进程间通信 (IPC) 机制 Binder	238
9.1.2 Service Manager 是 Binder 机制的上下文管理者	240
9.2 分析 Ashmem 驱动程序	241
9.2.1 基础数据结构	241
9.2.2 初始化处理	242
9.2.3 打开匿名共享内存设备文件	244
9.2.4 内存映射	246
9.2.5 读/写操作	247
9.2.6 锁定和解锁	249
9.2.7 回收内存块	254
9.3 分析 C++ 访问接口层	255
9.3.1 接口 MemoryBase	255
9.3.2 接口 MemoryBase	264
9.4 分析 Java 访问接口层	267

第 10 章 分析电源管理模块

10.1 Android Power Management 基础	271
10.2 Framework 层分析	272
10.2.1 文件 PowerManager.java	272
10.2.2 文件 PowerManagerService.java	273
10.3 JNI 层分析	294
10.3.1 文件 android_os_Power.cpp	294
10.3.2 文件 power.c	295
10.4 Kernel (内核) 层分析	296
10.4.1 文件 power.c	296
10.4.2 文件 earlysuspend.c	298
10.4.3 文件 wakelock.c	299
10.4.4 文件 resume.c	301
10.4.5 文件 suspend.c	301
10.4.6 文件 main.c	302
10.4.7 proc 文件	303
10.5 wakelock 和 early_suspend	303

10.5.1	wake lock 的原理	303
10.5.2	early_suspend 的原理	304
10.5.3	Android 休眠	305
10.5.4	Android 唤醒	307
第 11 章 分析低内存管理模块		
11.1	分析 OOM 机制	308
11.1.1	OOM 机制基础	308
11.1.2	分析 OOM 机制的具体实现	309
11.2	分析 Low Memory Killer 的原理和机制	315
11.3	分析 Low Memory Killer 驱动	316
11.3.1	Low Memory Killer 驱动基础	316
11.3.2	分析核心功能	317
11.3.3	设置用户接口	320
第 12 章 分析物理内存驱动		
12.1	PMEM 基础	323
12.1.1	PMEM 概述	323
12.1.2	Platform 设备基础	324
12.1.3	PMEM 与 Ashmem 的区别	325
12.2	分析 PMEM 系统驱动程序	325
12.2.1	分析设备实现	325
12.2.2	分析驱动的具体实现	327
12.2.3	总结 PMEM 驱动的调用流程	350
12.3	分析用户接口空间	350
12.3.1	释放位图内存	351
12.3.2	释放位图内存空间	351
12.3.3	获取位图占用内存	352
第 13 章 分析 Time Device 驱动		
13.1	分析 Timed Output 驱动程序	354
13.1.1	设备类	354
13.1.2	分析 Timed Output 驱动的具体实现	359
13.1.3	实现 timed_output_dev 设备的读写操作	361
13.2	分析 Timed Gpio 驱动程序	362

13.2.1	Timed Gpio 驱动的基本原理	363
13.2.2	分析 Timed Gpio 驱动的实现文件	363

第 14 章 分析调试机制驱动 Ram Console

14.1	Ram Console 基础	368
14.2	分析 Ram Console 的具体实现	369
14.2.1	分析文件 ram_console.h	369
14.2.2	分析文件 ram_console.c	369

第 15 章 分析时钟系统驱动 Alarm

15.1	Alarm 系统基础	379
15.1.1	Alarm 层次结构介绍	379
15.1.2	需要移植的内容	381
15.2	分析 RTC 驱动程序	381
15.3	分析 Alarm 驱动程序	381
15.3.1	分析文件 android_alarm.h	382
15.3.2	分析文件 alarm.c	383
15.3.3	分析文件 alarm-dev.c	394
15.4	分析 JNI 层	402
15.5	分析 Java 层	403
15.5.1	分析 AlarmManagerService 类	404
15.5.2	分析 AlarmManager 类	412
15.6	模拟器环境的具体实现	413

第 16 章 分析 USB Gadget 驱动

16.1	分析 Linux 内核的 USB 驱动程序	415
16.1.1	USB 设备基础	415
16.1.2	分析 USB 和 sysfs 的联系	419
16.1.3	分析 urb 通信方式	422
16.1.4	分析 USB 驱动的例程	427
16.2	分析 USB Gadget 驱动	438
16.2.1	分析软件结构	438
16.2.2	层次整合	447
16.2.3	USB 设备枚举	457

第 3 篇 设备驱动篇

第 17 章 FrameBuffer 显示系统驱动

17.1	显示系统介绍	473
17.2	分析内核层	474
17.2.1	分析接口文件 fb.h	474
17.2.2	分析内核实现文件 fbmem.c	477
17.3	分析硬件抽象层	499
17.3.1	分析头文件	500
17.3.2	分析硬件帧缓冲区	501
17.3.3	实现缓冲区的分配	503
17.3.4	显示缓冲映射	504
17.3.5	分析管理库文件 LayerBuffer.cpp	505

第 18 章 输入系统驱动应用

18.1	输入系统介绍	507
18.1.1	Android 输入系统结构元素介绍	507
18.1.2	移植工作	509
18.2	分析 Input (输入) 系统驱动	509
18.2.1	分析头文件	510
18.2.2	分析核心文件 input.c	513
18.2.3	分析 event 机制	527
18.3	分析硬件抽象层	530
18.3.1	分析文件 KeycodeLabels.h	530
18.3.2	分析文件 KeyCharacterMap.h	534
18.3.3	分析 K1 格式文件	535
18.3.4	分析 KCM 格式文件	536
18.3.5	分析文件 EventHub.cpp	536

第 19 章 音频系统驱动

19.1	音频系统基础	541
19.2	分析音频系统的层次	542
19.2.1	层次说明	543
19.2.2	Media 库中的 Audio 框架	543

19.2.3	本地代码.....	545
19.2.4	分析 JNI 代码.....	548
19.2.5	Java 层代码简介.....	549
19.3	移植 Audio 系统的工作.....	549
19.3.1	我们的工作.....	549
19.3.2	分析硬件抽象层.....	550
19.3.3	分析 AudioFlinger 中的 Audio 硬件抽象层的实现.....	551
19.3.4	真正实现 Audio 硬件抽象层.....	556
19.4	在 MSM 平台实现 Audio 驱动系统.....	557
19.4.1	实现 Audio 驱动程序.....	557
19.4.2	实现硬件抽象层.....	558

第 20 章 视频输出系统驱动

20.1	视频输出系统结构.....	562
20.2	需要移植的部分.....	564
20.3	分析硬件抽象层.....	564
20.3.1	Overlay 系统硬件抽象层的接口.....	564
20.3.2	实现 Overlay 系统的硬件抽象层.....	566
20.3.3	实现接口.....	567
20.4	实现 Overlay 硬件抽象层.....	568

第 21 章 振动器系统驱动

21.1	振动器系统结构.....	571
21.2	分析硬件抽象层.....	572
21.3	分析 JNI 层部分.....	574
21.4	分析 Java 层部分.....	575
21.5	实现移植工作.....	578
21.5.1	移植振动器驱动程序.....	578
21.5.2	实现硬件抽象层.....	579
21.6	在 MSM 平台实现振动器驱动.....	580

第 22 章 传感器系统驱动

22.1	传感器系统的结构.....	583
22.2	分析需要移植的内容.....	585
22.2.1	移植驱动程序.....	585

22.2.2 移植硬件抽象层.....	586
22.2.3 实现上层部分.....	587
22.3 在模拟器中实现传感器驱动.....	591

第 23 章 照相机系统

23.1 Camera 系统的结构.....	596
23.2 分析需要移植的内容.....	598
23.3 移植和调试.....	599
23.3.1 V4L2 驱动程序.....	599
23.3.2 硬件抽象层.....	605
23.4 实现 Camera 系统的硬件抽象层.....	609
23.4.1 Java 程序部分.....	609
23.4.2 Camera 的 Java 本地调用部分.....	610
23.4.3 Camera 的本地库 libui.so.....	610
23.4.4 Camera 服务 libcameraservice.so.....	611

第 24 章 Wi-Fi 系统、蓝牙系统和 GPS 系统

24.1 Wi-Fi 系统.....	616
24.1.1 Wi-Fi 系统的结构.....	616
24.1.2 分析需要移植的内容.....	618
24.1.3 分析本地实现.....	618
24.1.4 分析 JNI 层.....	621
24.1.5 分析 Java FrameWork 层.....	623
24.1.6 分析 Setting 中的 Wi-Fi 设置.....	624
24.2 蓝牙系统.....	625
24.2.1 蓝牙系统的结构.....	625
24.2.2 分析需要移植的内容.....	627
24.2.3 分析驱动程序.....	628
24.2.4 分析本地代码.....	628
24.2.5 在 MSM 平台实现蓝牙驱动系统.....	630
24.3 定位系统.....	631
24.3.1 分析定位系统的结构.....	631
24.3.2 分析需要移植的内容.....	633
24.3.3 分析驱动程序.....	633
24.3.4 分析硬件抽象层.....	633
24.3.5 分析上层应用部分.....	635