Android 深度探索（卷1）HAL与驱动开发

李宁 编著

第一本结合 S3C6410 开发板学习 Android HAL 开发和驱动开发的实战书籍

- **3大完整的驱动开发案例：** 蜂鸣器驱动
- **3大真实的实验环境：** Ubuntu Linux 12.04 LTS、Android 模拟器和 S3C6410 开发板
- **必知必会的驱动开发高级技术：** 并发控制、阻塞和非阻塞 I/O、异步编程、Linux 中断和底半部、时间管理、内存管理和 I/O 访问等
- **6 大核心 Linux 驱动代码分析与实战：** RTC 驱动、LCD 驱动、音频驱动、块设备驱动、网络设备驱动和 USB 驱动

人民邮电出版社
POSTS & TELECOM PRESS

实验环境与源程序

Android

深度探索（卷1）

HAL与驱动开发

李宁 编著

人民邮电出版社
北京

图书在版编目（CIP）数据

Android深度探索. 第1卷, HAL与驱动开发 / 李宁编著. -- 北京 : 人民邮电出版社, 2013.1
ISBN 978-7-115-29802-7

Ⅰ. ①A… Ⅱ. ①李… 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(2012)第249939号

内 容 提 要

全书分为4篇，分别从搭建开发环境，Linux驱动和Android HAL的基础知识，开发Linux驱动的高级技术和分析典型的Linux驱动源代码4个方面介绍Android和Linux的底层开发。本书使用的试验环境是Ubuntu Linux12.04 LTS、Android模拟器和S3C6410开发板。在第1篇详细介绍了如何搭建和使用这3个试验环境。第2篇通过3个Linux驱动的完整案例（统计单词个数驱动、LED驱动和蜂鸣器驱动）从不同角度来讨论如何开发一个完整的Linux驱动。并且通过完整的案例介绍了如何编写Android HAL，以及如何与Linux驱动交互。第3篇则介绍了开发Linux驱动所需要的高级技术，这些技术包括并发控制、阻塞和非阻塞I/O、异步编程、Linux中断和底半部、时间管理、内存管理和I/O访问。最后一部分分析了一些典型Linux驱动的源代码（RTC驱动、LCD驱动、音频驱动、块设备驱动、网络设备驱动和USB驱动）。

本书注重理论和实践相结合。在介绍了大量的基础知识的同时，为每一个知识点提供了完整的案例，使读者可以通过实际的代码更好地理解Linux驱动和Android底层技术。

为了使读者更好地实践本书提供的实例代码，在随书光盘中除了提供源代码文件外，还提供了一个VMWare Ubuntu Linux12.04 LTS的虚拟环境。读者可以在Windows、Linux和Mac OS X上，通过VMWare打开该虚拟机文件来学习和测试本书的例子（虚拟环境中也带了一套本书提供的例子代码）。

本书适合底层开发的程序员和编程爱好者使用，也适合作为相关培训学校的Android底层开发培训教材。

Android深度探索（卷1）：HAL与驱动开发

◆ 编　　著　李　宁
责任编辑　张　涛
◆ 人民邮电出版社出版发行　　北京市崇文区夕照寺街14号
邮编　100061　　电子邮件　315@ptpress.com.cn
网址　http://www.ptpress.com.cn
北京鑫正大印刷有限公司印刷
◆ 开本：800×1000　1/16
印张：40.75
字数：969千字　　2013年1月第1版
印数：1－3 000册　　2013年1月北京第1次印刷

ISBN 978-7-115-29802-7

定价：99.00元（附光盘）

读者服务热线：(010)67132692　印装质量热线：(010)67129223
反盗版热线：(010)67171154
广告经营许可证：京崇工商广字第0021号

目　录

第一篇

Android 驱动开发前的准备

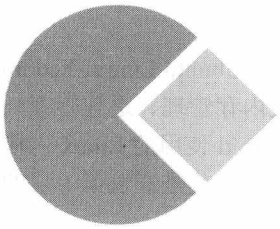
Android 系统移植与驱动开发概述

搭建 Android 开发环境

Git 使用入门

源代码的下载和编译

搭建 S3C6410 开发板的测试环境



第 1 章　Android 系统移植与驱动开发概述

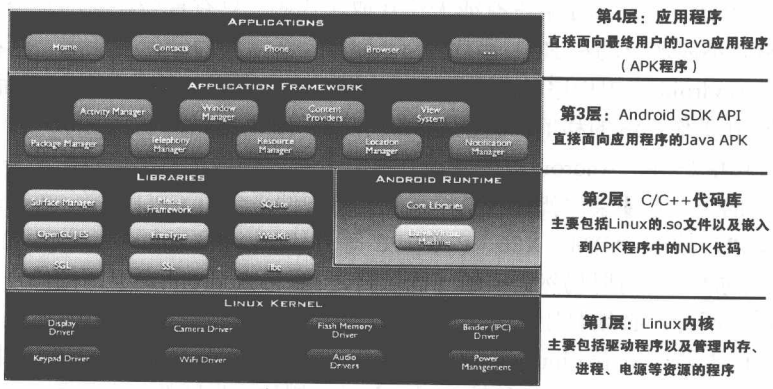
毋庸置疑，Android 已经成为当前智能手机操作系统的老大，市场占有率已遥遥领先于 iOS（IPhone 和 IPad）。Android 在几年时间发展如此神速，在很大程度上取决于任何人都可以利用 Android 的源代码定制完全属于自己的嵌入式系统，而不需要向 Google 交一分钱。

由于 Android 原生的代码支持的设备并不多，因此，要想在自己的设备（包括手机、MP4、智能电视、平板电脑、车载系统等）上完美运行 Android，就需要另外开发一些程序，使得 Android 可以识别相应设备中的硬件（显示器、蓝牙、音频、Wi-Fi 等）。这个为特定设备定制 Android 的过程被称为“移植”。那么，在移植的过程中开发得最多的就是支持各种硬件设备的 Linux 驱动程序（Android 是基于 Linux 内核的）。因此，讲移植就必须要讲驱动开发。

本章作为学习 Linux 驱动的第一道门，将对 Android 以及 Linux 驱动做一个总体的介绍，以便使读者对开发 Linux 驱动有一个感性的认识，并为更好地学习 Linux 驱动的方法和技巧打下基础。

1.1 Android 系统架构

Android 是一个非常优秀的嵌入式操作系统。经过几年的发展和演进，Android 已经形成了非常完善的系统架构，如图 1-1 所示。



▲图 1-1　Android 系统架构

从图 1-1 可以看出，Android 的系统架构分为 4 层。这 4 层所包含的内容如下。

第 1 层：Linux 内核

由于 Android 是基于 Linux 内核的，因此，Android 和其他 Linux 系统（如 Ubuntu Linux、Fedora Linux 等）的核心部分差异非常小。这一层主要包括 Linux 的驱动程序以及内存管理、进程管理、电源管理等程序。Android 使用 Linux 2.6 作为其内核。不过不同版本的 Android 使用的 Linux 内核版本有细微的差异，所以不同 Android 版本的驱动可能并不通用。本书主要讲的就是开发第 1 层的驱动程序，以及如何在不同 Linux 版本、硬件平台移植驱动程序。

第 2 层：C/C++代码库

这一层主要包括使用 C/C++编写的代码库（Linux 下的.so 文件），也包括 Dalivk 虚拟机的运行时（Runtime）。

第 3 层：Android SDK API

由于 Android SDK API 是用 Java 语言编写的，因此，这一层也可称为 Java API 层。实际上，这一层就是用 Java 编写的各种 Library。只不过这些 Library 是基于 Dalvik 虚拟机格式的。笔者所著《Android 开发权威指南》主要就是介绍了这一层的 Android SDK API 的使用方法及技巧。

第 4 层：应用程序

这一层是所有的 Android 用户（包括程序员和非程序员）都要接触到的。因为这一层相当于 Android 的 UI。所有的 Android 应用程序（包括拍照、电话、短信、Android 的桌面、浏览器以及各种游戏）都属于这一层。而这一层主要依靠第 3 层中的 Android SDK API 来完成各种功能。

1.2 Android 系统移植的主要工作

Android 移植可分为两部分：应用移植和系统移植。应用移植是指将如图 1-1 所示第 4 层的应用程序移植到某一个特定硬件平台上。由于不同硬件平台之间的差异，Android SDK API 也有可能存在差异（有的厂商会修改部分 Android SDK API 以适应自身硬件的需要），或者将应用程序从低版本 Android 移植到高版本的 Android 上。为了保证应用程序可以在新的硬件平台正常运行，需要对源代码进行一些修改。当然，如果没有或无法获取源代码，只有重新在新的平台上实现了。一般 Android 应用移植并不涉及驱动和 HAL 程序库（Android 新增加的硬件抽象层，将在后面的章节介绍）的移植，而且 Android 应用程序移植也不在本书讨论的范围内，因此，本书后面出现的 Android 移植都是指 Android 操作系统的移值（包括 Linux 驱动、HAL 程序库的移植）。

Android 系统移植是指让 Android 操作系统在某一个特定硬件平台上运行。使一个操作系统在特定硬件平台上运行的一个首要条件就是该操作系统支持硬件平台的 CPU 架构。Linux 内核本身已经支持很多常用的 CPU 架构（ARM、X86、PowerPC 等），因此，将 Android 在不同的 CPU 架构之间移植并不用做过多的改动（有时仍然需要做一些调整）。要想 Android 在不同硬件平台上正常运行，只支持 CPU 架构还不行，必须要让 Android 可以识别平台上的各种硬件（如声卡、显示器、蓝牙设备等）。这些工作主要也是由 Linux 内核完成的。其中的主角就是 Linux 驱动。因此，系统移植除了移植 CPU 架构外，最重要的就是移植 Linux 驱动。例如，为硬件平台增加了一个新型的 Wi-Fi 模块，就需要为这个 Wi-Fi 模块编写新的驱动程序，或修改原来的驱动程序，已使得 Linux 内核可以与 Wi-Fi 模块正常交互。

除了 Linux 驱动需要移植外，在 Android 系统中还增加了一个硬件抽象层（HAL，Hardware Abstraction Layer），为了方便，本书后面的部分都使用 HAL 表示硬件抽象层。

HAL 位于如图 1-1 所示的第 2 层，也是普通的 Linux 程序库（.so 文件），只是 Android SDK 通过 HAL 直接访问 Linux 驱动。也就是说，Android 并不像其他的 Linux 系统一样由应用程序直接访问驱动，而是中间通过 HAL 隔了一层。Google 这样设计的原因很多，例如，由于 Linux 内核基于 GPL 开源协议，而很多驱动厂商不想开放源代码，所以增加了 HAL 层后，可以将 Linux 驱动的业务逻辑放在 HAL 层，这样处理 Linux 驱动开源技术，也只是一个空架子而已。关于 Android 支持 HAL 的原因将在后面的章节详细介绍。

如果为 Android 增加了新的驱动或修改原来的驱动代码，HAL 中的代码就要做相应的调整。因此，Android 移植的主要工作如下：

- 移植 Linux 驱动；
- 移植 HAL。

移植的工作也可能不多，当然，也可能非常多。如果要移植的 Android 系统提供了驱动源代码，那就好办多了，直接根据移植的目标平台修改驱动代码就可以了。不过很多时候由于某些原因，无法获得驱动的源代码，或者要实现的驱动程序所对应的硬件是自己特有的，这就需要从头开始编写驱动程序以及相关的配置文件。对于 HAL 的移植也和 Linux 驱动差不多。总之，Android 移植的基本原则是尽可能找到驱动和 HAL 的源代码，在源代码的基础上改要比从头开始编写容易得多，实在无法获取源代码，就只有从头开始做起了。不过在了解了编写 Linux 驱动和 Android HAL 程序库的步骤和规则以后，看着也没那么复杂。因为驱动和 HAL 的代码远没有 Android SDK 和 Android 应用程序的代码量大。

注意

Android 移植在很大程度上是 Linux 内核的移植。Linux 内核移植主要就是移植驱动程序。不同 Linux 版本的驱动程序不能通用，需要重新修改源代码，并在新的 Linux 内核下重新编译才可以运行在新的 Linux 内核版本下。Android 版本和 Linux 版本不同。无论哪个 Android 版本，其 Linux 内核版本都是 Linux 2.6 或 Linux 3.0（将来有可能使用更高版本的 Linux 内核），只是小版本号不同。由于 Android 开放源代码，所以就算同一个 Android 版本，Linux 的内核也可能不同（有很多自制的 ROM 会更换不同的 Linux 内核，以至于和官方同一 Android 版本的 Linux 内核不同），例如，笔者曾见过有的 Android 2.3 使用了 Linux 2.6.29，而官方的 Android 2.3 使用了 Linux 2.6.35。在移植 Linux 驱动时，主要应考虑 Linux 内核的版本，就算 Android 版本不同，只要 Linux 内核版本相同，Linux 驱动就可以互相替换（有时也需要考虑 HAL 是否和 Linux 驱动兼容）。

1.3 查看 Linux 内核版本

目前 Linux 内核主要维护 3 个版本：Linux 2.4、Linux 2.6 和 Linux 3.x，大多数 Linux 系统都使用了这 3 个版本的内核，其中 Linux 2.6 是目前使用最广泛的 Linux 内核版本，Android 就使用了该

▲图 1-2　查看 Android 的 Linux 内核版本

内核版本。而 Linux 2.4 由于其内部设计缺陷（主要是进程调度上的缺陷），除了一些遗留 Linux 系统，已很少有新的 Linux 系统使用 Linux 2.4 了。Linux 3.x 是最新推出的 Linux 内核版本。最新的 Android 4.x 采用了这个新的 Linux 3.0.8 内核版本，还有很多新推出的 Linux 系统（如 Ubuntu Linux 11.10）都使用了 Linux 3.0。读者可在 Android 系统中的“设备”>“关于手机”中查看当前 Android 系统所采用的 Linux 内核版本，如图 1-2 所示。

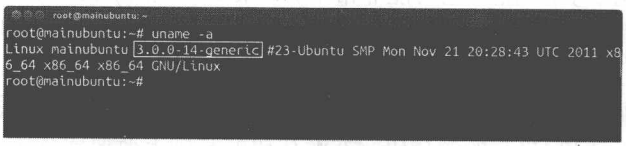
如果想查其他 Linux 系统的内核版本，可使用下面两种方法。

方法 1

在 Linux 终端执行下面的命令。

```
uname -a
```

如果当前系统是 Ubuntu Linux 11.10，会在 Linux 终端输出如图 1-3 所示的信息。白框内是 Linux 内核的版本。



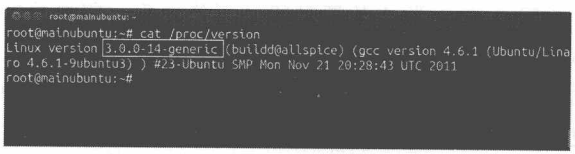
▲图 1-3　使用 uname 命令查看 Linux 内核版本

方法 2

在 Linux 终端执行下面的命令。

```
cat /proc/version
```

在 Linux 终端输出如图 1-4 所示的信息。白框内是 Linux 内核的版本。



▲图 1-4　查看 proc/version 文件获取 Linux 内核版本

注意

/proc 不是普通的文件系统，而是系统内核的映像，也就是说，该目录中的文件是存放在系统内存之中的，它以文件系统的方式为访问系统内核数据的操作提供接口。而 uname 命令就是从/proc/version 文件中获取信息的，当然直接查看/proc/version 文件的内容（方法 2）也可以获取同样的信息。uname 命令加上参数“-a”可以获取更多的信息，否则只显示当前的系统名，也就是只会输出“Linux”。

1.4 Linux 内核版本号的定义规则

Linux 内核版本号由下面几部分组成。

- 主版本号；