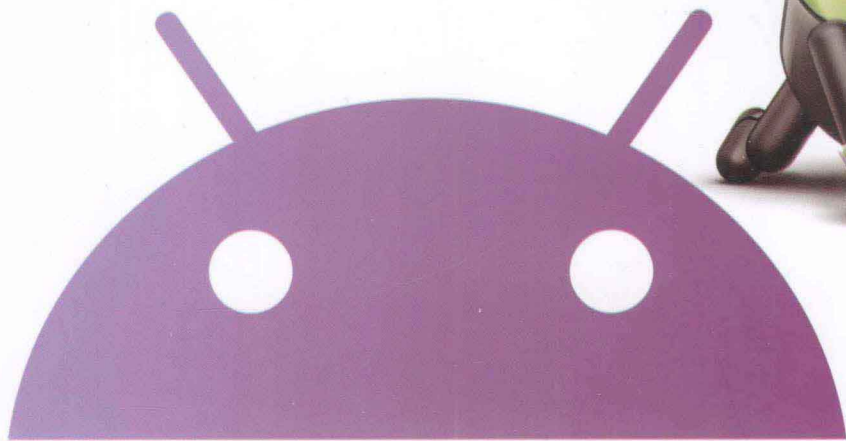
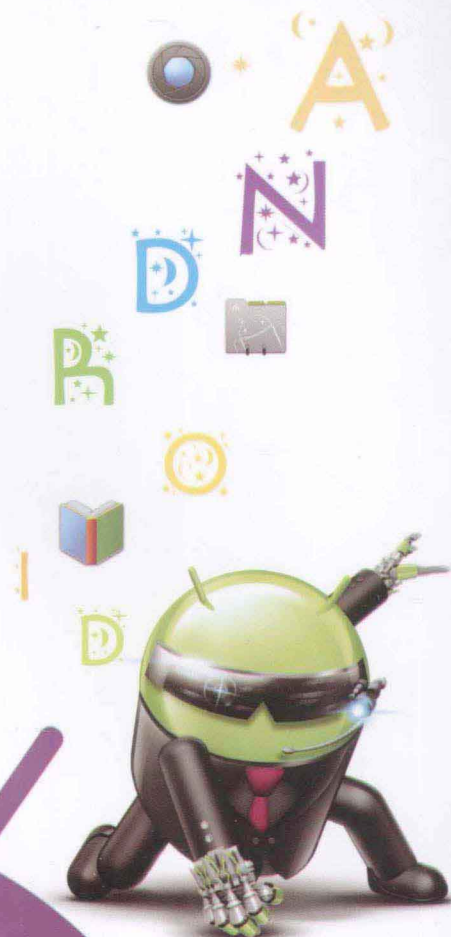


高等院校移动开发人才培养规划教材

本书特色

- 关注理论知识结构完整性，重视实践前瞻性环节
- 本书内容包含Android系统驱动开发、系统移植等内容，基于流行的Cortex-A8 ARM处理器
- 本书包含自主研发的基于Android系统的智能家居等丰富实用的项目开发案例
- 所有实验均可通过国内领先自主研发的Cortex-A8开发板测试



Android

系统移植和驱动开发

华清远见移动互联网学院 郑萌 等编著



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

高等院校移动开发人才培养规划教材

Android

系统移植和驱动开发

华清远见移动互联网学院 郑萌 等编著



电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书针对特定硬件平台从零构建 Android 系统,包括平台移植、应用开发、内核开发及驱动开发。目前企业对 Android 人才需求主要分布在应用开发和底层系统开发两个方面,企业最紧缺的是兼具二者的全才型系统工程师,和嵌入式 Linux 人才需求基本相似。本书共分 12 章,首先介绍了 Android 底层开发和嵌入式开发之间的关系;接着介绍了 Android 系统的编译和移植,并在此基础上进一步阐述了 Android 驱动开发的原理;最后给出了一个 Android 智能家居的实例。

本书适合作为大学院校嵌入式技术专业、电子信息类其他专业的教材,也可供高等及中等职业技术学院使用。

未经许可,不得以任何方式复制或抄袭本书的部分或全部内容。
版权所有,侵权必究。

图书在版编目(CIP)数据

Android 系统移植和驱动开发 / 郑萌等编著. —北京:电子工业出版社,2013.9

高等院校移动开发人才培养规划教材

ISBN 978-7-121-20699-3

I. ①A… II. ①郑… III. ①移动终端—应用程序—程序设计—高等学校—教材 IV. ①TN929.53

中国版本图书馆 CIP 数据核字(2013)第 130803 号

策划编辑:胡辛征

责任编辑:葛娜

特约编辑:赵树刚

印刷:北京中新伟业印刷有限公司

装订:河北省三河市路通装订厂

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开本:787×1092 1/16 印张:17.75 字数:455 千字

印次:2013 年 9 月第 1 次印刷

印数:3000 册 定价:49.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010)88254888。

质量投诉请发邮件至 zlls@phei.com.cn,盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010)88258888。

前 言

Android 一词的本义指“机器人”，是由 Google 公司于 2007 年 11 月正式对外发布的，作为一种以 Linux 为基础的开放源代码操作系统，主要用于便携设备。依靠 Google 的强大开发和媒体资源，凭借其开放性和优异性，Android 平台在发展的过程中得到了包括大手机厂商和著名移动运营商在内的业界的广泛支持，除手机的应用之外，目前已逐渐扩展到平板电脑及其他领域上。2011 年第一季度，Android 在全球的市场份额首次超过塞班系统，跃居全球第一。2012 年 2 月数据，Android 占据全球智能手机操作系统市场 52.5% 的份额，中国市场占有率为 68.4%。与此同时，随着行业的迅猛发展，Android 研发工程师更是日益成为 IT 职场的紧缺人才。近几年来，各大院校纷纷开设 Android 移动开发专业。但是，各院校在 Android 专业教学建设的过程中几乎都面临教材难觅、内容更新迟缓的困境。虽然目前市场上的 Android 开发相关的书籍比较多，但几乎都是针对有一定基础的行业内研发人员而编写的，并不完全符合高校的教学要求。高校教学需要一套充分考虑学生现有知识基础和接受度的、明确各门课程教学目标的、便于学校安排课时的 Android 专业系列教材。

针对高校专业教材缺乏的现状，我们以多年来在嵌入式工程技术领域及移动开发行业内人才培养、项目研发的经验为基础，汇总了近几年积累的数百家企业对 Android 研发相关岗位的真实需求，并结合行业应用技术的最新状况及未来发展趋势，调研了开设 Android 专业的大学院校的课程设置情况、学生特点和教学用书现状。通过细致的整理和分析，对专业技能和基本知识进行合理划分，我们编写了这套高等院校 Android 开发人才培养规划教材，包括以下 4 本：

- 《Android 系统下 Java 编程详解》
- 《Android 应用程序开发与典型案例》
- 《Android 游戏案例开发与关键技术》
- 《Android 系统移植和驱动开发》

本套教材按照专业整体教学要求组织编写，各自对应的主干课程之间既相对独立，又有机衔接，整套教材具有系统性。考虑到 Android 研发领域对学生 Java 语言能力要求较高，专门有针对性地编写了《Android 系统下 Java 编程详解》这本教材，可供“Java 语言基础”课程的后续提高课程使用；《Android 应用程序开发与典型案例》则结合 Android 应用开发的核心知识，重点突出了贯穿前面所学知识的实训案例及内容，可供“Android 应用程序开发”课程使用；在 Android 游戏开发方面，根据各院校的教学重点和行业实际应用情况，量身定制了《Android 游戏案例开发与关键技术》；《Android 系统移植和驱动开发》侧重介绍 Android 底层移植和驱动技术。

本书主要讲解 Android 系统移植和驱动开发技术。Android 是典型的嵌入式操作系统，和传统的基于 PC 桌面的编程有很大差别，嵌入式系统的特点在于可定制性、可移植性、硬件

相关性、功耗低，基于这些特点，本书针对特定硬件平台从零构建 Android 系统，包括平台移植、应用开发、内核开发及驱动开发。

本书由华清远见嵌入式学院资深讲师郑萌编著并统校全稿。本书的完成需要感谢华清远见嵌入式学院及华清远见移动互联网学院，教材内容参考了学院与嵌入式及移动开发企业需求无缝对接的、科学的专业人才培养体系。参与本书编写的人员有谢培良、崔浩、周志强、李宗亮、李珊珊、吴现凯、高良伟、王泽政、蒋铎、袁升、付世全、赵健乔、赵晶晶、范爱伟等，并在后期审校工作中提供了很多帮助，在此表示衷心的感谢。

由于作者水平所限，书中不妥之处在所难免，恳请读者批评指正。对于本书的批评和建议，可以发到 www.farsight.com.cn 网站的技术论坛中。

编著者
2013 年 7 月

目 录

第 1 章 Android 系统的编译和移植实例	1
1.1 移植背景与目标	1
1.2 移植涉及的主要过程	1
1.3 下载 Android Linux 内核	1
1.4 安装交叉工具链	3
1.5 移植 Android Linux 内核支持 EZ6410 平台	4
1.5.1 CS8900a 驱动移植	4
1.5.2 键盘驱动编写	4
1.5.3 液晶驱动	5
1.5.4 触摸屏驱动	6
1.5.5 USB 驱动修改	7
1.5.6 安装 Android SDK	8
1.5.7 提取 Android 根文件系统	9
1.5.8 系统环境设置	11
1.6 小结	12
1.7 思考题	12
第 2 章 Android 系统与嵌入式开发	13
2.1 Android 与传统嵌入式 Linux 系统	13
2.2 嵌入式系统概述	13
2.2.1 嵌入式系统简介	13
2.2.2 嵌入式系统的特点	14
2.2.3 嵌入式系统的发展	15
2.3 嵌入式系统的组成	17
2.3.1 嵌入式系统的硬件组成	18
2.3.2 嵌入式系统的软件组成	19
2.4 嵌入式系统开发概述	19
2.5 Android 系统概述	26
2.6 Android 系统架构	28



2.6.1	Linux 内核 (Linux Kernel)	28
2.6.2	Android 核心库 (Libraries)	29
2.6.3	Android 运行时环境 (Android Runtime)	29
2.6.4	Android 应用程序框架 (Application Framework)	29
2.6.5	Android 应用程序和小部件	30
2.7	小结	30
2.8	思考题	30
第 3 章	Android 移植平台工具介绍	31
3.1	开发板外观	31
3.2	开发板硬件资源	32
3.3	功能展示	33
3.4	目标板与主机之间的连接	35
3.5	文件传输	37
3.6	网络文件系统	38
3.7	设备安装	39
3.8	工具软件的设置	39
3.9	U-Boot 镜像的下载与烧写	42
3.10	内核镜像的下载与烧写	48
3.11	烧写文件系统镜像	50
3.12	U-Boot 启动参数设置	51
3.13	小结	52
3.14	思考题	52
第 4 章	Android 移植环境搭建	53
4.1	构建 Android 移植交叉开发环境	53
4.1.1	嵌入式交叉编译环境搭建	53
4.1.2	主机交叉开发环境配置	55
4.2	Bootloader	59
4.2.1	Bootloader 的种类	60
4.2.2	U-Boot 编译与使用	61
4.2.3	U-Boot 移植	73
4.3	Android 内核与移植	74
4.3.1	Android 移植简介	76
4.3.2	FS_S5PC100 开发平台移植环境搭建	76
4.4	U-Boot、内核、文件系统编译	78
4.4.1	U-Boot 的编译	78
4.4.2	内核镜像的编译	78

4.4.3	Android 文件系统的编译.....	79
4.5	小结.....	81
4.6	思考题.....	81
第 5 章	Android 移植与驱动	82
5.1	Android 移植和驱动的关系.....	82
5.2	设备驱动程序.....	82
5.2.1	设备驱动概念.....	82
5.2.2	内核驱动程序.....	83
5.2.3	用户空间驱动程序.....	83
5.2.4	Linux 设备驱动概述.....	83
5.3	Linux 核心与 Android 驱动.....	85
5.3.1	Android 专用驱动.....	86
5.3.2	Android 使用的设备驱动.....	87
5.4	Android 驱动的 HelloWorld.....	92
5.5	小结.....	95
5.6	思考题.....	95
第 6 章	Android 驱动编程	96
6.1	Android 内核内核模块编程.....	96
6.2	字符设备驱动编程.....	105
6.2.1	字符设备驱动编写流程.....	105
6.2.2	重要数据结构.....	106
6.2.3	设备驱动程序主要组成.....	107
6.3	LCD 控制器.....	114
6.3.1	LCD 控制器介绍.....	114
6.3.2	S5PC100 LCD 控制器介绍.....	114
6.3.3	S5PC100 LCD 控制器操作.....	116
6.3.4	LCD 控制器寄存器.....	117
6.4	驱动程序.....	123
6.4.1	驱动程序初始化和退出.....	124
6.4.2	驱动程序 Open and release 函数.....	126
6.4.3	驱动程序 ioctl 函数.....	127
6.4.4	驱动测试程序 main.c.....	128
6.5	小结.....	128
6.6	思考题.....	129



第 7 章 Android HAL 实例解析	130
7.1 概述.....	130
7.2 HAL 介绍.....	130
7.3 HAL 内容.....	131
7.3.1 知识准备: HAL 主要的存储目录.....	131
7.3.2 知识准备: 两种 HAL 架构比较.....	131
7.4 mokoid 工程代码下载与结构分析.....	132
7.4.1 知识准备: mokid 项目概述.....	132
7.4.2 知识准备: 结构分析.....	132
7.4.3 知识准备: 第一种方法 直接调用 service 方法的实现过程.....	135
7.4.4 知识准备: 第二种方法——经过 Manager 调用 service.....	140
7.4.5 FS-S5PC100 实验平台 LED 灯实验.....	143
7.4.6 FS-S5PC100 实验平台 Sensor 实验.....	144
7.5 实验中需要注意的问题.....	145
7.6 小结.....	146
7.7 思考题.....	147
第 8 章 Android 4.0 系统的下载与编译	148
8.1 准备 Android 下载与编译环境.....	148
8.1.1 硬件环境.....	148
8.1.2 软件环境.....	148
8.2 下载 Android 4.0 及 Goldfish 源码.....	149
8.2.1 下载并初始化 repo 工具.....	149
8.2.2 下载 Android 源码.....	150
8.2.3 其他源码下载源.....	150
8.2.4 下载模拟器 Goldfish 内核源码.....	150
8.3 编译 Android 及 Goldfish 内核源码.....	152
8.3.1 编译 Android 源码.....	152
8.3.2 编译 goldfish 内核源码.....	154
8.4 Android 编译过程分析.....	154
8.4.1 source build/envsetup.sh.....	155
8.4.2 执行 lunch full-eng.....	156
8.4.3 执行 make 命令.....	156
8.5 创建目标产品编译项.....	157
8.6 编译并运行 Android 4.0 系统模拟器.....	158
8.6.1 编译 Android 模拟器.....	158
8.6.2 Android 启动流程.....	161

8.7	定制手机模拟器 ROM.....	162
8.7.1	定制手机开机界面.....	162
8.7.2	定制 Android 启动字样.....	165
8.7.3	定制 Android 动画.....	166
8.8	为 Android 启动加速.....	168
8.8.1	定制本地服务.....	168
8.8.2	定制 Android 系统服务.....	169
8.9	Android 系统企业级定制.....	170
8.10	小结.....	172
8.11	思考题.....	172
第 9 章	Android 系统蜂鸣器与按键.....	173
9.1	蜂鸣器驱动的实现.....	173
9.1.1	PWM 定时器概述.....	173
9.1.2	PWM 定时器特点.....	174
9.1.3	PWM 定时器的寄存器.....	175
9.1.4	PWM 定时器操作示例.....	180
9.1.5	蜂鸣器的种类和工作原理.....	181
9.1.6	开发板上的蜂鸣器原理图分析.....	181
9.2	蜂鸣器 HAL 层的实现.....	182
9.3	Android 按键检测程序流程.....	184
9.4	驱动层实现.....	185
9.4.1	HAL 层实现.....	186
9.4.2	HAL 使用.....	187
9.4.3	HAL stub 实现.....	188
9.4.4	JNI 层实现.....	189
9.4.5	Framework 层实现.....	189
9.4.6	应用层实现.....	190
9.5	小结.....	190
9.6	思考题.....	190
第 10 章	Android 与传感器设计与实现.....	191
10.1	Android Sensor 实现原理概述.....	191
10.2	JNI 原理.....	191
10.3	Android HAL.....	192
10.4	Sensor 传感器系统.....	193
10.4.1	Sensor Stub 分析.....	194
10.4.2	Sensor Stub 实现细节.....	202



10.5	Logcat 调试.....	202
10.6	一些开发环境的问题.....	203
10.7	小结.....	203
10.8	思考题.....	203
第 11 章	Android 与摄像头.....	204
11.1	OV9650 介绍.....	204
11.1.1	芯片功能描述.....	204
11.1.2	OV9650 寄存器详解.....	205
11.2	SCCB 总线.....	206
11.2.1	SCCB 协议介绍.....	206
11.2.2	SCCB 的总线编程.....	207
11.3	CAMIF 接口详解.....	209
11.3.1	基于 S5PC100 的 CAMIF 接口介绍.....	209
11.3.2	S5PC100 CAMIF 寄存器详解.....	210
11.4	摄像头驱动.....	213
11.4.1	摄像头工作原理.....	213
11.4.2	驱动开发思路.....	214
11.5	Android Camera 架构分析.....	228
11.5.1	CameraService 介绍.....	229
11.5.2	Camera-HAL 介绍.....	229
11.5.3	CameraService 与 Camera-HAL 的联系.....	230
11.6	小结.....	232
11.7	思考题.....	232
第 12 章	Android 下综合项目介绍——智能家居.....	233
12.1	项目概述.....	233
12.1.1	项目背景.....	233
12.1.2	术语定义.....	234
12.1.3	系统描述.....	234
12.1.4	功能描述.....	235
12.1.5	项目功能图.....	235
12.1.6	所涉及的系统与工具.....	235
12.2	前端数据中心 (A8) 总体设计.....	236
12.2.1	程序设计流程图.....	236
12.2.2	线程定义.....	236
12.2.3	所用类定义.....	236
12.3	前端 A8 模块设计.....	237





12.3.1	A8-Android Application 层设计	237
12.3.2	A8-数据传递架构模块	245
12.3.3	传输协议模块设计	257
12.4	终端 M0 模块设计	259
12.4.1	终端设备方案描述	259
12.4.2	终端设备工作流程	259
12.4.3	功能模块描述	260
12.5	系统测试	267

第 1 章 Android 系统的编译和移植实例

本章主要介绍 Android 系统的编译和移植技术，作为建立在 Linux 内核基础上的 Android 操作系统，其编译和移植无论是过程还是技术都和嵌入式 Linux 非常相似。本章着重介绍一个典型的 Android 系统编译和移植的实例，该实例类似于编程学习中著名的 HelloWorld，虽然简单，但是涵盖了特定平台下 Android 系统编译和移植的整个流程。如果暂时看不懂也没有关系，随着后面章节的继续深入，就会有豁然开朗的感觉。

1.1 移植背景与目标

现有的环境是一套能够正常运行 Linux 2.6.21 的 EZ6410（基于 S3C6410）硬件系统。移植目标是在 EZ6410 系统上运行 Android 2.3 系统。

1.2 移植涉及的主要过程

- 下载 Android Linux 内核。
- 安装交叉工具链。
- 移植 Android Linux 内核支持 EZ6410 平台。
- 安装 Android SDK。
- 获得 Android 根文件系统。
- 设置系统环境，完成 Android 正常启动。

1.3 下载 Android Linux 内核

目前，支持 S3C6410 硬件的 Android 系统可以在网上找到，网址为 <http://code.google.com/hosting/>，如图 1.1 所示，可以看到有很多支持 S3C6410 的 Android 项目。

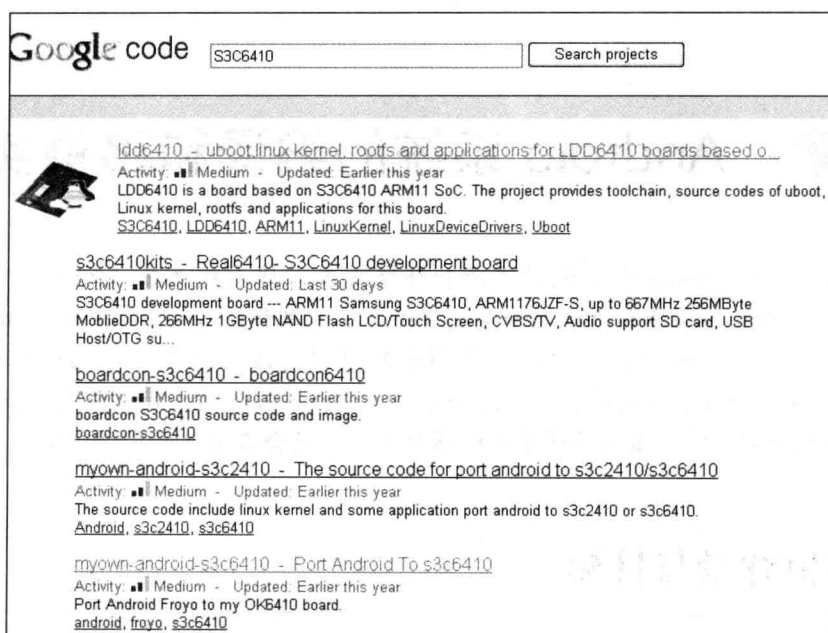


图 1.1 支持 S3C6410 的 Android 项目

单击“ldd6410”链接，打开相应网页。

LDD6410 的硬件结构如图 1.2 所示，我们需要针对其与 EZ6410 硬件结构的差异进行移植。EZ6410 的具体硬件配置请参考开发板手册。

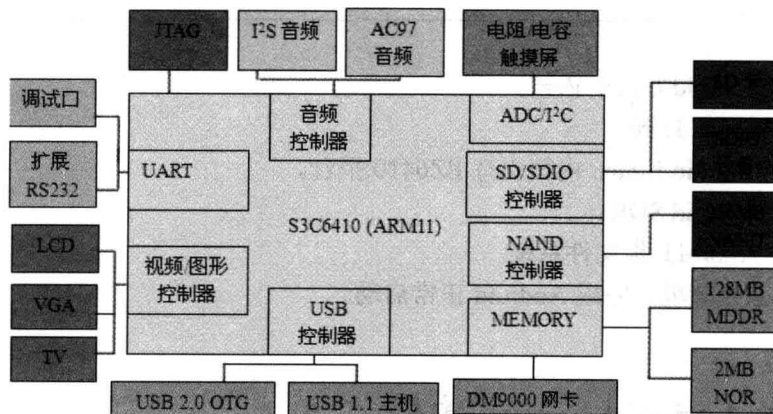


图 1.2 LDD6410 开发板结构图

LDD6410 整合了完整的 Android 驱动（位于 drivers/android 下的 binder、lowmemory killer 等）、内核电源管理（位于 kernel/power 下的 wakelock、userwakelock 等）、ashmem 补丁（mm/ashmem.c）和虚拟电池（drivers/power/fake_battery.c）等。

如图 1.3 所示为 drivers/android 下驱动的配置。

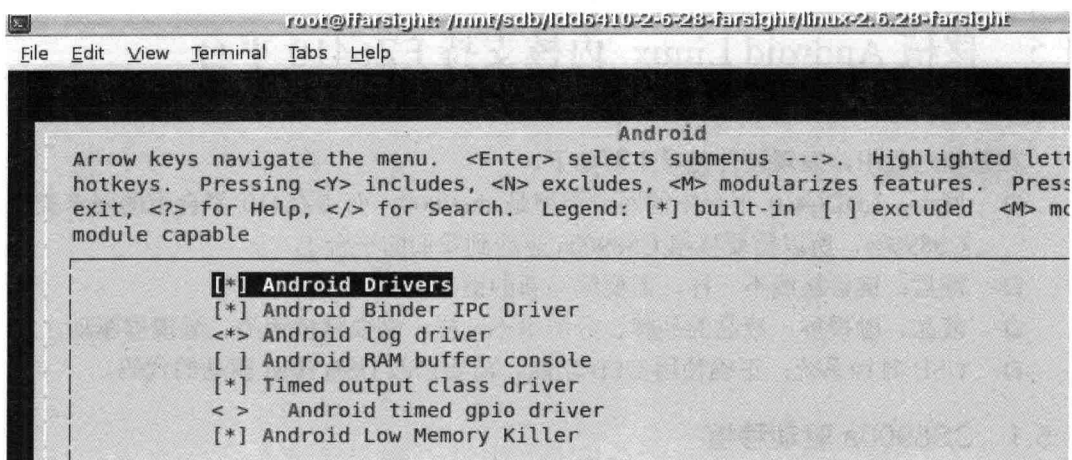


图 1.3 Android 驱动配置

如图 1.4 所示为 kernel/power 下 Android 电源管理的配置。

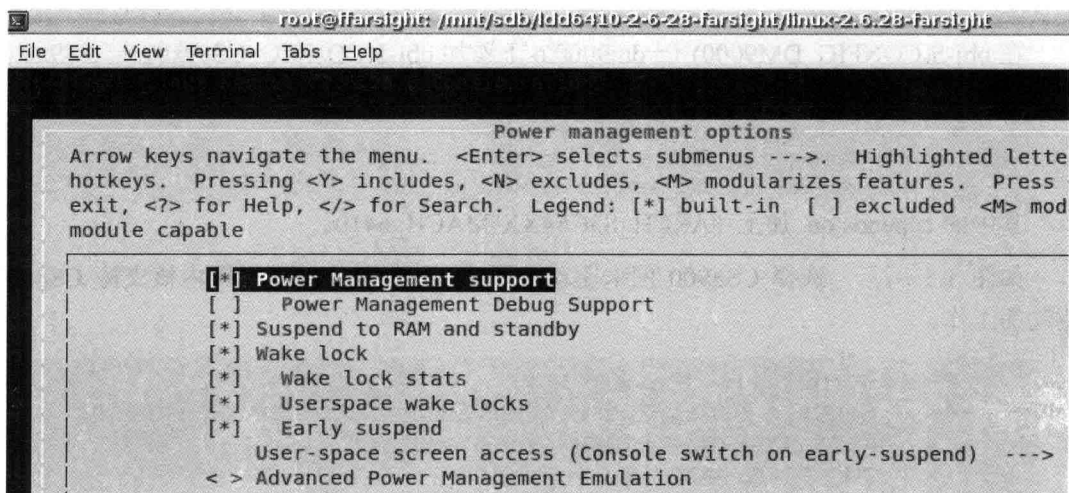


图 1.4 Android 电源配置

1.4 安装交叉工具链

在开始内核移植之前，先完成工具链的搭建。解压 EZ6410 光盘\B\Linux\cross_tools\arm-none-linux-gnueabi.tar.bz2 到 ubuntu-8.10 系统的/usr/local/arm 目录下。

在 ldd6410-read-only 目录下修改 vim .cross_compile 内容为：

```
/usr/local/arm/arm-none-linux-gnueabi/bin/arm-none-linux-gnueabi-
```



1.5 移植 Android Linux 内核支持 EZ6410 平台

在移植过程中，发现硬件差异主要如下。

- 网卡：LDD6410 平台中的网卡类型是 dm9000，而 EZ6410 平台中的网卡类型是 CS8900a，所以需要移植 CS8900a 驱动到我们的平台上。
- 键盘：键盘接线不一样，需要编写新的键盘驱动。
- 液晶、触摸屏：液晶的品牌、分辨率不一样，需要移植液晶、触摸屏驱动。
- USB 时钟系统：正确使用 USB 功能，需要修改 USB Host 驱动的代码。

1.5.1 CS8900a 驱动移植

将实验代码目录下的 cs89x0.c 复制到内核的 drivers/net/ 目录下。配置内核支持 cs8900a9A71 驱动，代码如下：

```
#vim Makefile
```

在 obj-\$(CONFIG_DM9000) += dm9000.o 下添加 obj-\$(CONFIG_CS89x0) += cs89x0.o

```
#vim Kconfig
```

```
tristate "CS89x0 support"
```

```
depends on NET_ETHERNET && (ISA || EISA || MACH_IKDP2351 \  
    || ARCH_IKDP2X01 || ARCH_PNX010X || MACH_MX31ADS)
```

其中的 depends on 加上 ||ARCH_S3C64XX||MACH_6410。

如图 1.5 所示，选择 CS8900 网卡驱动配置项，这样就完成了配置内核支持 CS8900a 驱动的工作。

```
--- Ethernet (10 or 100Mbit)
-*- Generic Media Independent Interface device support
< > ASIX AX88796 NE2000 clone support
< > SMC 91C9x/91C1xxx support
< * > DM9000 support
(4)   DM9000 maximum debug level
[ ]   Force simple NSR based PHY polling
< > SMSC LAN911[5678] support
< > Broadcom 440x/47xx ethernet support
< * > CS89x0 support
```

图 1.5 配置 CS8900 网卡驱动

1.5.2 键盘驱动编写

针对 Andorid 要求，编写键盘驱动。EZ6410 实验平台上有 8 个按键，即 K1~K8。

Linux 系统提供了 Input 子系统，按键、触摸屏、键盘、鼠标等输入都可以利用 Input 接口函数实现设备驱动，因此，按键和触摸屏设备驱动都可以作为 Input 设备驱动而实现。如图 1.6 所示为键盘原理图。

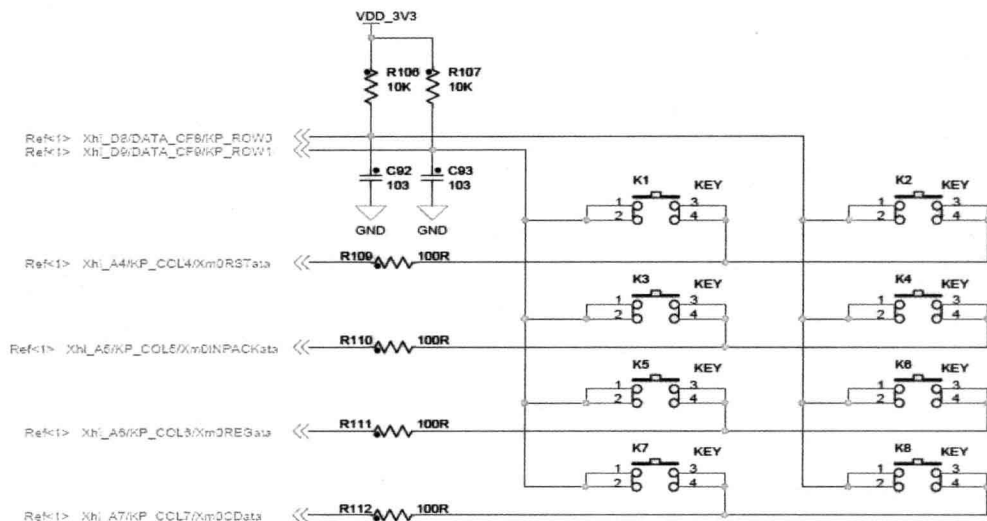


图 1.6 键盘原理图

Android 系统键值要求可以参见 Android 源码，具体的源码文件路径是 `system/usr/keylayout/qwerty.kl`，我们用 vim 打开这个文件：

```
#vim system/usr/keylayout/qwerty.kl
#define BACK          158
#define SOFT_RIGHT    60
#define MENU          229
#define SEARCH        127
#define HOME          102
#define DPAD_CENTER   232
#define DPAD_DOWN     108
#define DPAD_UP       103
#define DPAD_LEFT     105
#define DPAD_RIGHT    106
.....
```

我们实现其中的 8 个按键，对应关系如下：

K1(RIGHT)	K2(CENTER)	K3(MENU)	K4(DOWN)
K5(HOME)	K6(UP)	K7(BACK)	K8(LEFT)

代码参见“实验代码/键盘驱动代码”目录下的 `key_drv.c`。

1.5.3 液晶驱动

修改 LDD6410 中的液晶驱动支持 EZ6410 平台的 320×240 的液晶屏。

修改的文件为：`drivers/video/samsung/s3cfb_wanxin.c`。