

基于 ARM2440 平台的 Android 操作系统的移植

宋杰, 王书菊, 曹竹冬, 檀林欣

(安徽大学 计算智能与信号处理教育部重点实验室, 安徽 合肥 230039)

摘要: Android 作为 2007 年底刚刚发布的全新的开源手机操作系统, 由于其开源性与强大的应用层 API, 使得其在短时间内拥有很多开发者。结合实例阐述了 Android 操作系统(Android OS)移植到 ARM2440 平台的具体实现过程, 重点论述了移植过程中的具体方法。其中主机操作系统采用 Fedora 9(Linux-2.6.25), 目标机采用友善之臂 mini2440, Android OS 内核采用 Linux-2.6.25-android-1.0, 编译工具采用 arm-linux-gcc-4.3.2。最后, 成功将 Android OS 移植到开发板上。

关键词: Android 操作系统; 移植; Fedora 9; mini2440

中图分类号: TP316.89

文献标识码: A

文章编号: 1673-629X(2011)01-0066-04

Android Operating System Migration Based on ARM2440 Platform

SONG Jie, WANG Shu-ju, CAO Zhu-dong, TAN Lin-xin

(Ministry of Education, Key Laboratory of Intelligent Computing & Signal Processing,
AnHui University, Hefei 230039, China)

Abstract: Android platform is a new generation of smart mobile phone platform launched by Google. Android OS is open and Android provides powerful application layer API, so it has vast numbers of developers in a short time. Discuss with Android operating system (Android OS) migration to the ARM2440 platform specific implementation process. Focus on the specific method of transplantation. The host operating system using Fedora 9 (Linux-2.6.25), the target machine is a friendly arm mini2440, Android OS kernel using Linux-2.6.25-android-1.0, compilation tools using arm-linux-gcc-4.3.2. Finally, the Android OS was ported to the development board successfully.

Key words: Android operating system (Android OS); Migration; Fedora 9; mini2440

0 引言

Android 操作系统是 Google 与 OHA (Open Handset Alliance, 开放手机联盟) 合作开发的基于 Linux 2.6 平台的开源智能手机操作系统平台。Android 从开放以来, 受到业界人士追捧, 吸引了越来越多的程序员参与到手机开发的领域中, 所以将 Android OS 移植到不同的平台上有了很好的市场价值。

Android Linux Kernel 是基于 Linux 操作系统 2.6 版所更改而来, 本文所用的 Android Linux Kernel 版本为 2.6.25, 采用的 CPU 版本为嵌入式系统常用的 ARM 架构, 不是 PC 常见的 x86 架构。内核底层用来提供核心系统服务: 安全机制、内存管理、进程管理、网络堆栈、驱动模型等。由于 Android 与 ARM 的良好结合性, 所以笔者选用基于 S3C2440 处理器的 mini2440 开发板作为目标机^[1-4]。

1 Android OS 介绍

Android OS 是 Google 公司最新推出的面向下一代以移动互联网业务为核心的智能终端开源平台, 该平台由操作系统、中间件、用户界面和应用软件组成, 具体框架结构由五部分组成, 其核心为嵌入式 Linux 2.6 操作系统, 向上搭配 Google 为 Android 开发 Libraries (函数库) 及 Android Runtime (运行环境), 再配合 Application Framework (应用程序框架), 来开发各种不同的 Application (应用程序)。

Android 是基于 Linux version 2.6.25 内核开发的, 主要是添加了一个名为 Goldfish 的虚拟 CPU 以及 Android 运行所需的特定驱动代码。该层用来提供系统的底层服务, 包括安全机制、内存管理、进程管理、网络堆栈及一系列的驱动模块。作为一个虚拟的中间层, 该层位于硬件与其它软件层之间。需要注意的是, 这个内核操作系统并非类 GNU/Linux 的, 所以其系统库、系统初始化和编程接口都和标准的 Linux 系统有所不同的。

对 Android 内核结构进行分析是移植到其他平台

收稿日期: 2010-05-22; 修回日期: 2010-08-01

基金项目: 安徽省教育科研计划项目 (2008jyxm277)

作者简介: 宋杰 (1966-), 男, 副教授, 硕士生导师, 研究方向为嵌入式系统、计算机原理与接口、生物信息学。

的先决条件,由于 Android 的开源性,可以直接到官方网站下载源码,本文采用 Linux-2.6.25-android-1.0。Android 内核是基于 Linux 内核改进而来,所以做移植工作的重点是比较两种内核的异同点,通过查看源码,可以发现有以下异同点。首先,Android 增加了基于 ARM 架构增加 Gold-Fish 平台,目录如下: kernel/arch/arm/mach-goldfish、kernel/include/asm-arm/arch-goldfish;增加了 yaffs2 FLASH 文件系统,目录如下: kernel/fs/yaffs2;增加了 Android 的相关 Driver,目录如下: kernel/drivers/android;增加了 switch 处理,目录如下: kernel/drivers/switch/;增加了一种新的共享内存处理方式,文件如下: kernel/mm/ashmem.c;其次为 Linux-2.6.25 内核所做的补丁等,例如 BlueTooth,在此不做详细分析。进行比较之后在移植过程中就可以在需要修改的地方进行修^[5,6]。

2 软硬件平台搭建

所谓移植,就是使一个实时内核能在某个微处理器或微控制器上运行,或者说就是将不同平台、不同编译环境的程序代码经过修改转移到自己的系统中运行的实现过程。本文是将 Android OS 通过修改移植到 ARM2440 平台上。

Android OS 是基于 Linux 内核修改而来的,通过在 Linux 平台下,对两个操作系统的内核进行比较分析,从而修改需要裁剪的地方,顺利移植到相应的开发板之上。本文采用的时基于 Linux kernel2.6 的 fedora 9 操作系统, fedora 9 是 Linux 的一个免费发行版,它的前身是 Red Hat Linux。可以从官方网站下载安装,并可下载到 fedora 9 源码。

其次,android OS 内核提供了基于 ARM 的架构,移植到 ARM 上有了理论支持。本文选择基于 Samsung S3C2440A 处理器的 mini2440 开发板,是 ARM 9 系列芯片。mini2440 开发板主频是 400MHz,带有内存管理单元(MMU),支持 Linux 和 Windows CE 等嵌入式操作系统的移植嵌入。另外,mini2440 带有 FLASH 存储单元,其中 Nand Flash128M,Nor Flash2M,掉电非易失^[7]。

3 移植过程分析

系统移植是个庞大的工程,主要包括四个步骤:第一步是准备阶段,包括搭建平台,下载源码;第二步是建立交叉编译环境;第三步是编写引导程序 BootLoader,配置编译内核,必要的时候要根据具体的硬件平台修改源代码;第四步是制作文件系统映象;第五步就是把相应的文件烧到开发板上。

在嵌入式系统中,主要使用 FLASH 作为系统的存

储媒介,而很少使用磁盘,因此整个系统的加载启动任务就完全由引导程序来完成。通常,在嵌入式系统中,引导用 Flash 会被按顺序分为几个区域,Android OS 也不例外,Android 中的 FLASH 分区如图 1 所示。

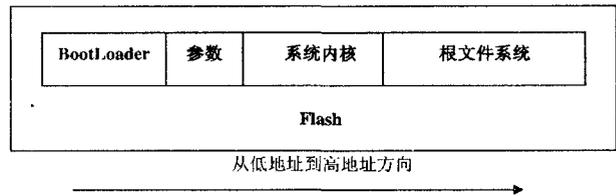


图 1 Android 的 FLASH 分区

BootLoader 是嵌入式系统最初运行的一段小程序,它在系统复位时被运行。从字面意义理解,它包括两个功能:装载(load)和启动(boot)。具体来说,BootLoader 完成的任务主要包括初始化处理器及外设的硬件资源配置、建立内存空间的映射图、装载操作系统映象到内存中去、对 FLASH 编程、运行操作系统等。BootLoader 是与操作系统相关的,主要符合操作系统的引导标准,就可以让此 BootLoader 引导操作系统。本实验采用 Supervivi,它由 Vivi(韩国 mizi 公司原创,开放源码)演变而来。它支持 yaffs 文件系统和 Linux、WinCE、Vxworks 等多种嵌入式操作系统,是目前 2440/2410 系统中功能最强大最好用的 BootLoader^[8]。

GCC(GNU Compiler Collection)是 GNU 公社的一个项目,是一个用于编程开发的自由编译器。GCC 包含众多语言的编译器,其中包括 C,C++,Ada,Object C 和 Java 等。如今的 GCC 借助于它的特性,具有了交叉编译器的功能,即在一个平台下编译另一个平台的代码。GCC 交叉编译原理如图 2 所示^[9]。

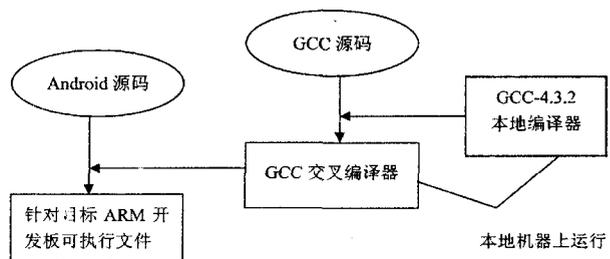


图 2 GCC 交叉编译原理图

GCC 的强大功能及开放源码,本文采用的 arm-linux-gcc-4.3.2.tgz(with EABI)的大小是 84MB,EABI(Embedded Application Binary Interface)嵌入式应用程序二进制接口,EABI 支持软件浮点和硬件实现浮点功能混用,使系统调用的效率更高,另以后的工具更兼容,是一种软件加速的方式,本文将 Android 配置成 EABI 特性。

4 移植过程实现

软硬件平台搭建好之后,我们需要做的就是具体

的编译工作了。

4.1 准备阶段

主机上采用在 aware 虚拟机上安装 fedora 9 操作系统,并到官网下载 Linux 2.6 内核和 Linux-2.6.25-android-1.0 内核,通过 Linux 操作系统提供的指令可以比较两种内核之间的差异。使用友善之臂提供的 arm-linux-gcc-4.3.2. 工具链和 mkyaffs2image. tgz 根文件系统制作工具,均可以到 <http://www.arm9.net/> 下载。确保目标板 mini2440 开发板上的 USB 接口、串口、JTAG 已经连接上,并且安装了相应的驱动^[10-12]。

4.2 交叉编译环境的建立

安装编译器的过程如下,首先以根用户身份登录系统,复制 arm-linux-gcc-4.3.2. tgz 到根目录下 tmp 文件夹里,解压命令 # tar xvzf arm-linux-gcc-4.3.2 -C /,解压的文件放到了根目录下 /usr/local/arm 下,然后配置下编译环境路径 # gedit /root/.bashrc,在文本编辑器最后一行加上 export PATH = \$PATH:/usr/local/arm/4.3.2/bin,重启系统,通过 arm-linux-gcc -v 查看 gcc 的版本信息。

4.3 内核移植

交叉编译环境成功建立以后,可以开始编译内核了,在编译内核之前,需要对内核进行相应的修改,首先建立工作目录: #/opt/FriendlyARM/android, 然后解压内核到此目录下: #tar xvzf linux-2.6.25-android-1.0. tar.gz -C /opt/FriendlyARM/android,解压出来的内核中已经包含了一个缺省的内核配置文件: config_mini2440,运行命令: #cp config_mini2440.config (注意 config 前面有个“.”),然后运行 make menuconfig,不要做任何设置,保存退出,然后 make zImage,执行完毕,将在 arch/arm/boat 目录下生产 zimage 文件^[13]。

4.4 制作 yaffs2 文件系统映象

首先需要安装 mkyaffs2image. tgz 根文件系统制作工具, # tar xvzf mkyaffs2image-128M. tgz - C /,把解压后的文件放在 /usr/sbin/mkyaffs2image-128M 中通过当前目录下执行 ./mkyaffs2image-128M 即可得到相关信息。

Android 文件采用是和 Linux 不完全相同的根文件系统,这里我们下载 busybox 工具,制作 Android 文件系统,这里主要是参考了友善之臂所提供的文件系

统,稍加修改而来,解压后放在 /opt/FriendlyARM/android/fs 里面,然后通过 #/usr/sbin/mkyaffs2image -128M fs minni2440T35_android.img,利用根文件系统制作工具得到我们需要的 Android OS 的 yaffs2 文件系统映象 minni2440T35_android.img。

4.5 烧写到开发板上

在准备阶段,已经确保 mini2440 能够正常烧写,主要是 USB 驱动和串口终端设置,可以参见 mini2440 开发板提供的用户手册。

烧写 Android OS 到开发板主要过程如下:首先是格式化 nand flash,然后烧写友善之臂提供的 supervivi-128M,主要通过 ARM 串口烧写工具 dnw 来进行传送数据,其次是将上面编译过的 zimage 文件也是通过 dnw 传送数据,最后把相应的 yaffs2 文件系统映象通过串口传送工具 dnw 下载到 nand flash 中。重启系统,即可看到相关的 Android 启动信息,如图 3 所示^[14]。



图 3 串口中 Android OS 启动信息

通过阅读启动信息可以看见,系统移植的还不够完善,有些驱动程序并没有找到,还需要针对平台编写相应的驱动模块。等待片刻即可在 mini2440 开发板上看到 Android 启动界面,正中间有个大钟,至此系统移植工作告一段落,如图 4 所示。

5 结束语

Android 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。谷歌开发 Android 操作系统的目标是让(移动通讯)不依赖于设备甚至平台。越来越多的 Google Android 爱好者们开始移植 Android 系统到现有掌上设备上。最早的成功移植在 Zairis 上,然后是在 OMAP 芯片组基础上的 N810,还有很多平台没有或者正在研究移植 Android 系统。通过本文的试

验,将 Android 移植到 ARM2440 平台上去,符合谷歌公司开发 Android 的初衷,可以为以后移植 Android 到其他新的平台上作为参考实例。

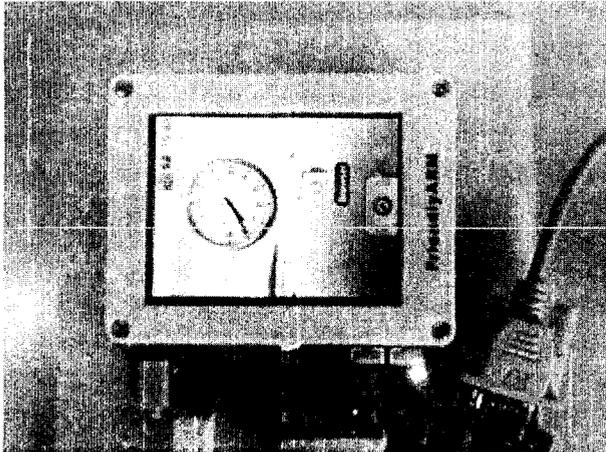


图4 mini2440 中 Android OS 启动后界面

参考文献:

- [1] 石晶翔,陈蜀宇,黄晗辉. 基于 Linux 系统调用的内核级 Rootkit 技术研究[J]. 计算机技术与发展,2010,20(4):175-178.
- [2] 姚昱旻,刘卫国. Android 的架构与应用开发研究[J]. 计算机系统应用,2008(11):118-121.
- [3] 余志龙,陈小凤,陈小凤,等. Google android SDK 开发范例

(上接第 65 页)

性检索方法自然与一般的形状匹配方法也是不同的。因而,商标图像的相似性检索过程也并不是一个完全准确的匹配过程。一般而言,商标图像的检索方法首先通过排除不相似的商标,然后在相似图像中选出与预查询的目标图像具有较高相似度的图像^[12]。这些常用的商标图像检索方法在某些方面仍有些不足^[13]。它体现在旋转不变性对于几何变形图像的检索能力、尺度、检索精度以及图像与人的视觉感受一致等方面。因此,需要研究出新型、有效的综合性方法。此外,对于图像的相似性,不同的用户存在着不同的理解。为不同的用户提供不同的特征组合方法,并提供友好的人机交互界面,使得系统能够根据用户需求做出调整输出,以及更好地满足用户需求,这些都是值得研究的问题。

参考文献:

- [1] 夏叔华. 商标法要论[M]. 北京:中国政法大学出版社,1989.
- [2] 陈世鸿,王 轩. 商标数据库存储模式及其检索算法研究[J]. 武汉大学学报,2003,49(1):63-66.
- [3] 卢章平,朱科铃. 基于形状特征和用户反馈的商标图像检索技术[J]. 农业机械学报,2007,38(5):150-152.

大全[M]. 北京:人民邮电出版社,2009.

- [4] Jacoby G, Davis N. Battery-based intrusion detection[C]// in Global Telecommunications Conference, 2004. GLOBE-COM '04. [s. l.]:IEEE,2004:2250 - 2255.
- [5] Enck W, Ongtang M, McDaniel P. Understanding Android Security[J]. IEEE Security & Privacy,2009,7(1):50 - 57.
- [6] Leavitt N. Mobile Phones: The Next Frontier for Hackers? [J]. Computer,2005,38(4):20 - 23.
- [7] 广州友善之臂公司. mini2440 用户手册[M]. 广州:广州友善之臂公司,2009:10-15.
- [8] 刘 森. 嵌入式系统接口设计与 Linux 驱动程序设计[M]. 北京:北京航空航天大学出版社,2006:350-359.
- [9] 于 明,范书瑞,曾祥焯. ARM9 嵌入式系统设计与开发教程[M]. 北京:电子工业出版社,2006:144-145.
- [10] 赵利娜,贾豫东,欧 攀. 嵌入式 Linux 系统中的多层次驱动程序[J]. 计算机工程,2009,35(9):144-146.
- [11] 冯忠岭,童英华. 嵌入式 Linux 的设备驱动程序设计及交叉编译[J]. 青海大学学报(自然科学版),2009(6):63-66.
- [12] 刘胜辉,马 嵩. 基于 Linux 内核的实时调度机制研究及应用[J]. 计算机工程与应用,2008,44(6):121-123.
- [13] 靳 岩,姚尚郎. Google android 开发入门与实战[M]. 北京:人民邮电出版社,2009:334-340
- [14] 韩艳芬,吴援明,王斯瑶,等. 一种二次 Bootloader 升级和回退的设计与实现[J]. 计算机技术与发展,2009,19(10):89-92.

- [4] 戴新亮,孙卫真,关 永. 印鉴识别中图像预处理问题的研究[J]. 微计算机信息,2007,23(27):305-306.
- [5] 娄正良,黄 磊,刘昌平. 基于内容的大数据量商标检索系统[J]. 小型微型计算机系统,2005,26(8):1397-1400.
- [6] Mehtre B M, Kankanhalli M S, Wing Foon Lee. Shape Measures for Content Based Image Retrieval a Comparison[J]. Information Processing & Management,1999,33(3):319-337.
- [7] 黄清元,谭汉松. 基于矩特征和面积特征的二值图像检索方法[J]. 企业技术开发(学术版),2007,26(6):24-26.
- [8] Robert M, Shanmugam H K. Texture features for image classification[J]. IEEE Trans. on Sys, Man, and Cyb,1973,SMC-3(6):610-621.
- [9] 杨秀娟. 基于纹理特征的图像检索研究[D]. 西安:西安科技大学计算机系,2009.
- [10] Rocchio J J. Relevance feedback in information retrieval[M]. [s. l.]:Prentice Hall,1971:313-323.
- [11] 徐 庆,杨维维,陈生潭. 基于颜色特征图像检索与相关反馈综合研究[J]. 计算机技术与发展,2007,17(12):251-254.
- [12] 闫乐林,元莱滨,蔡平胜. 一种基于内容的图像检索系统的设计与实现[J]. 计算机技术与发展,2009,19(12):205-208.
- [13] 孙强强. 基于内容的二值商标图像检索研究[D]. 扬州:扬州大学,2008.