

## <<ARM嵌入式系统移植实战开发>>

### 图书基本信息

书名：<<ARM嵌入式系统移植实战开发>>

13位ISBN编号：9787512407794

10位ISBN编号：7512407793

出版时间：2012-5

出版时间：北京航空航天大学出版社

作者：韩少云，奚海蛟，谌利 编著

页数：294

字数：421000

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

## <<ARM嵌入式系统移植实战开发>>

### 内容概要

《ARM嵌入式系统移植实战开发》以嵌入式Linux移植技术为主，以基于S3C2440的TQ2440开发板以及基于S3C6410的TQ6410开发板为硬件平台，详细讲述了嵌入式Linux中非常繁琐却又十分重要的系统移植过程，包括

UBoot、内核、文件系统、驱动程序以及应用程序的移植。

本书理论与实践相结合，配有详细的步骤，学完后读者可以在TQ2440开发板平台上搭建起自己的一套系统。

这样不仅可以使读者更好地理解所学知识，还能增加读者的学习兴趣。

《ARM嵌入式系统移植实战开发》可以作为高等院校嵌入式系统开发与应用的教材，嵌入式培训用书，以及嵌入式系统开发技术人员的参考书。

本书由韩少云、奚海蛟、谌利编著。

## <<ARM嵌入式系统移植实战开发>>

### 作者简介

韩少云，达内IT培训集团创始人，总裁 / CEO，加拿大技术移民。

曾任亚信公司软件事业部副总工程师。

在软件开发、软件培训、公司运营方面积累了丰富的实战经验，被誉为软件工程师创业的典范。

奚海蛟，博士后，北京融慧广泽科技有限公司创始人，毕业于北京航空航天大学电子工程学院，主要研究嵌入式与物联网、虚拟现实等方向。

主持过多个相关大型项目，曾在多家大学与培训机构任教。

谌利，硕士，北京融慧广泽公司合伙人。

毕业于北京航空航天大学电子工程学院，曾任飞思卡尔半导体公司硬件工程师、技术经理，有多年高端嵌入式处理器系统软硬件开发经验。

# <<ARM嵌入式系统移植实战开发>>

## 书籍目录

### 第1章 嵌入式系统概述

- 1.1 嵌入式系统介绍
    - 1.1.1 嵌入式系统概念
    - 1.1.2 嵌入式系统特点
    - 1.1.3 嵌入式产品
  - 1.2 嵌入式操作系统
    - 1.2.1 VxWorks
    - 1.2.2 WinCE
    - 1.2.3 UC/OS-
    - 1.2.4 Symbian
    - 1.2.5 Linux
  - 1.3 嵌入式系统开发流程
    - 1.3.1 嵌入式系统组成
    - 1.3.2 嵌入式系统开发流程
  - 1.4 嵌入式系统的移植
- 本章小结

### 第2章 构建嵌入式Linux开发环境

- 2.1 硬件环境构建
    - 2.1.1 主机与目标板结合的交叉开发模式
    - 2.1.2 硬件要求
  - 2.2 软件环境构建
    - 2.2.1 在虚拟机中设置Linux与Windows共享目录
    - 2.2.2 NFS的配置与启动
    - 2.2.3 嵌入式交叉编译工具的安装
    - 2.2.4 minicom和超级终端的配置及使用
    - 2.2.5 H-JTAG和DNW的安装和使用
- 本章小结

### 第3章 Bootloader移植

- 3.1 Bootloader简介
  - 3.1.1 Bootloader概念
  - 3.1.2 Bootloader启动流程分析
  - 3.1.3 常用的Bootloader介绍
- 3.2 U-Boot代码分析
  - 3.2.1 U-Boot简介
  - 3.2.2 U-Boot代码结构
  - 3.2.3 U-Boot代码编译
  - 3.2.4 U-Boot代码导读
  - 3.2.5 U-Boot命令
- 3.3 U-Boot移植
  - 3.3.1 在U-Boot中建立自己的开发板
  - 3.3.2 支持Nor Flash
  - 3.3.3 支持Nand Flash
  - 3.3.4 支持从Nand Flash启动
  - 3.3.5 支持网卡DMg000
  - 3.3.6 支持YAFFS文件系统

## <<ARM嵌入式系统移植实战开发>>

3.3.7 U-Boot引导内核

3.3.8 移植后U-Boot的使用

本章小结

### 第4章 内核移植

4.1 Linux内核结构

4.1.1 内核组成

4.1.2 内核目录

4.2 内核Makefile分析

4.2.1 内核Makefile的分类

4.2.2 Makefile的编译流程

4.2.3 Makefile主要内容解析

4.3 内核配置选项

4.3.1 通用选项

4.3.2 模块相关选项

4.3.3 块相关选项

4.3.4 系统类型、特性和启动相关选项

4.3.5 网络协议相关选项

4.3.6 设备驱动相关选项

4.3.7 文件系统类型相关选项

4.3.8 其他选项

4.4 内核在ARM上的移植

4.4.1 内核基本结构的移植

4.4.2 添加内核对YAFFS的支持

4.4.3 内核中RTC时钟驱动移植

4.4.4 内核中LCD驱动移植

4.4.5 内核中DM9000驱动移植

本章小结

### 第5章 构建Linux根文件系统

5.1 文件系统简介

5.2 嵌入式文件系统

5.2.1 嵌入式文件系统的特点

5.2.2 常见嵌入式文件系统

5.3 Linux根文件系统的结构

5.4 移植Busybox

5.4.1 Busybox简介

5.4.2 Busybox编译

5.5 安装glibc库

5.6 Linux系统的引导过程

5.6.1 启动内核

5.6.2 init进程介绍及用户程序启动

5.7 构建根文件系统

本章小结

### 第6章 Linux设备驱动移植

6.1 Linux设备驱动移植概述

6.1.1 Linux设备驱动程序的介绍

6.1.2 Linux设备驱动的分类、

6.1.3 Linux设备驱动移植步骤

## <<ARM嵌入式系统移植实战开发>>

### 6.2 简单Linux设备驱动的移植实例

#### 6.2.1 Hello World驱动的移植

#### 6.2.2 LED驱动的移植

#### 6.2.3 按键驱动的移植

### 6.3 完善已有的Linux设备驱动实例

#### 6.3.1 完善串口驱动

#### 6.3.2 配置USB设备驱动

#### 6.3.3 声卡驱动移植

#### 6.3.4 SD卡驱动移植

#### 本章小结

## 第7章 Linux下应用程序的开发和移植

### 7.1 嵌入式GUI简介

#### 7.1.1 Qt / Embedded

#### 7.1.2 MiniGUI

#### 7.1.3 MicroWindows

### 7.2 Qtopia移植

#### 7.2.1 Qt主机开发环境搭建

#### 7.2.2 交叉编译并安装Qtopia 4.5.3

#### 7.2.3 开发第一个Qt程序：Hello world!

### 7.3 MiniGUI移植

#### 7.3.1 MiniGUI开发环境搭建

#### 7.3.2 MiniGUI立用程序开发

### 7.4 音频解码器madplay移植

### 7.5 SQLite数据库移植

### 7.6 WebServer软件设计与移植

#### 7.6.1 WebServer简介

#### 7.6.2 WebServer的工作原理

#### 7.6.3 移植boa软件

#### 7.6.4 移植cgie库

#### 7.6.5 配置WebServer

#### 本章小结

## 第8章 Android在\$3C6410上的移植

### 8.1 Android简介

#### 8.1.1 初识Android

#### 8.1.2 Android的发展历程

#### 8.1.3 开发环境介绍

### 8.2 Android系统的移植

#### 8.2.1 交叉编译工具的安装

#### 8.2.2 NFS服务器的配置

#### 8.2.3 编译U-Boot

#### 8.2.4 编译内核

#### 8.2.5 编译Android文件系统

### 8.3 Android系统的烧写

#### 8.3.1 烧写SD卡的U-Boot

#### 8.3.2 烧写Nand Flash启动的U-Boot

#### 8.3.3 烧写内核和设置从NFS启动文件系统

#### 8.3.4 启动文件系统

<<ARM嵌入式系统移植实战开发>>

8, 3.5 U-Boot启动Android2.0文件系统

本章小结

参考文献

## &lt;&lt;ARM嵌入式系统移植实战开发&gt;&gt;

## 章节摘录

版权页：插图：3.网络接口网络接口是区别于字符设备和块设备的第三大标准类设备，与前两种设备不同，因为Unix世界里“一切皆是文件”的论述对于它来说并不适用。

例如，块设备可以在系统文件树的ldev目录下找到特定的文件入口标志，而网络设备则没有这种文件操作入口。

Unix式的操作系统访问网络接口的方法是给它们分配一个唯一的名字（比如eth0），而这个名字在文件系统中（比如刚刚提到的/dev目录下）不存在对应的节点项。

网络接口同时具有字符设备、块设备的部分特点，但是都有不同。

相比于字符设备，它的输入/输出是有结构的、成块的（报文、包、帧）。

相比于块设备，它的“块”又不是固定大小的，可以大到数百甚至数千字节，又可以小到几个字节。

由于网络接口并不是以文件的方式存在，这就导致应用程序、内核和网络驱动程序间的通信完全不同于字符设备和块设备，内核提供了一套push等操作来完成数据包的转换与递送，而不是open，read和write等。

6.1.3 Linux设备驱动移植步骤本节来介绍一下本章的重点-Linux设备驱动的移植。

其实，Linux设备驱动的移植就是将写好的驱动程序添加到相应的内核中去。

这里说的相应的内核而不是所有的内核，其原因是不同的内核它的目录树的结构是不同，即其中有些头文件的位置是不同的。

甚至，其中的一些函数也不相同。

所以在移植驱动程序的时候一定要注意驱动程序开发所参照的内核版本和所移植的版本要相同。

设备驱动程序移植有如下4个步骤：（1）准备内核源码准备要移植到的操作系统平台的源码，因为无论将驱动程序编译成模块还是将驱动程序添加到内核中编译内核，都需要该平台的内核源码包。

（2）准备驱动程序源码该源码可以是自己写的驱动程序，也可以是别人写好的驱动程序，但前提是该驱动和所要移植的内核版本符合。

（3）编译驱动程序到内核编译驱动程序可以分为两种形式：将准备好的驱动程序代码放到内核相应的目录下，并通过修改Makefile和Kconfig文件将其添加到内核目录树中，然后通过makemenuconfig配置该选项。

使用准备好的内核源码将驱动程序编译成内核模块，将编译好的模块移植到相应的操作系统平台上，加载模块。

（4）测试驱动程序编写简单的测试程序，测试所添加的驱动程序，查看其是否可用。





版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>