

《零点起步》

图书基本信息

书名：《零点起步》

13位ISBN编号：9787111333166

10位ISBN编号：7111333160

出版时间：2011-4

出版社：机械工业出版社

页数：356

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：www.tushu000.com

《零点起步》

内容概要

《零点起步:嵌入式Linux编程入门与开发实例》共分15章，分别介绍了嵌入式系统基础、文本编辑器的使用、构建嵌入式Linux开发环境、Linux常用命令、编译与调试、图形界面应用程序开发、嵌入式文件系统与存储技术、Linux设备驱动程序开发、进程控制、线程控制、嵌入式车载终端的设计、嵌入式BOA服务器的构建、嵌入式VNC远程控制的实现、ARM Linux指纹门禁系统和基于ARM Linux的家庭安全监控系统设计。《零点起步:嵌入式Linux编程入门与开发实例》不仅有详细的理论知识介绍，还有大量的开发案例以供参考，可读性和实用性强，适合于没有或者缺乏嵌入式Linux程序设计经验的初学者作为嵌入式Linux C语言开发的自学教材，同时也适合于已掌握C语言基础并想学习嵌入式开发的读者。《零点起步:嵌入式Linux编程入门与开发实例》可作为各类学校的教学用书，也可作为工程技术人员的参考书。

书籍目录

前言

第1章 嵌入式系统基础

- 1.1 嵌入式系统概述
- 1.2 嵌入式操作系统
- 1.3 Linux 操作系统
 - 1.3.1 Linux的发展历史
 - 1.3.2 Linux的发行版本
 - 1.3.3 Linux系统的特点和组成
- 1.4 ARM处理器平台
 - 1.4.1 ARM处理器简介
 - 1.4.2 ARM处理器的体系结构
 - 1.4.3 ARM微处理器系列
 - 1.4.4 ARM微处理器的应用领域及特点
 - 1.4.5 ARM微处理器的结构
 - 1.4.6 ARM微处理器的应用选型
- 1.5 嵌入式系统开发
 - 1.5.1 嵌入式系统的开发流程
 - 1.5.2 Linux程序设计流程
- 1.6 思考与练习

第2章 文本编辑器的使用

- 2.1 Vi编辑器
 - 2.1.1 Vi的基本模式
 - 2.1.2 Vi的基本操作
- 2.2 Vim编辑器
- 2.3 Emacs编辑器
 - 2.3.1 Emacs的启动与退出
 - 2.3.2 Emacs的基本编辑
 - 2.3.3 Emacs的C模式
 - 2.3.4 Emacs的Shell模式
- 2.4 gedit编辑器
- 2.5 思考与练习

第3章 构建嵌入式Linux开发环境

- 3.1 嵌入式系统开发环境的构建
- 3.2 移植U-Boot
 - 3.2.1 Bootloader简介
 - 3.2.2 Bootloader的启动流程
 - 3.2.3 U-Boot的分析与移植
 - 3.2.4 VIVI 分析
- 3.3 嵌入式Linux操作系统内核编译
 - 3.3.1 Linux 内核结构
 - 3.3.2 内核的配置
 - 3.3.3 内核编译的过程
 - 3.3.4 内核的移植
- 3.4 思考与练习

第4章 Linux常用命令

- 4.1 目录命令
 - 4.1.1 ls

- 4.1.2 cd
- 4.1.3 pwd
- 4.1.4 mkdir
- 4.1.5 rmdir
- 4.2 文件命令
 - 4.2.1 cp
 - 4.2.2 rm
 - 4.2.3 mv
 - 4.2.4 cat
 - 4.2.5 chmod
 - 4.2.6 find
 - 4.2.7 grep
 - 4.2.8 sort
- 4.3 思考与练习
- 第5章 编译与调试
 - 5.1 嵌入式Linux编译器GCC
 - 5.1.1 GCC编译分析
 - 5.1.2 GCC编译选项分析
 - 5.1.3 库函数的使用方法
 - 5.2 调试工具GDB
 - 5.2.1 GDB使用实例
 - 5.2.2 GDB的帮助
 - 5.2.3 设置/删除断点
 - 5.2.4 各种相关命令
 - 5.3 Make工程管理器
 - 5.3.1 Makefile文件的构成
 - 5.3.2 Makefile变量
 - 5.3.3 Make管理器的使用
 - 5.4 思考与练习
- 第6章 图形界面应用程序开发
- 第7章 嵌入式文件系统与存储技术
- 第8章 Linux设备驱动程序开发
- 第9章 进程控制
- 第10章 线程控制
- 第11章 嵌入式车载终端的设计
- 第12章 嵌入式BOA服务器的构建
- 第13章 嵌入式VNC远程控制的实现
- 第14章 ARM Linux 指纹门禁系统
- 第15章 基于ARM Linux的家庭安全监控系统设计
- 参考文献

章节摘录

版权页：插图：在必要的情况下，用户可以自己编写程序，及时为Linux打补丁，以修补系统的漏洞，这是其他操作系统没有的优势。另外，由于系统的代码是开放的，用户可了解系统的各个方面，不用担心系统会预留“后门”。当然，用户要自己阅读或修改Linux系统的源代码，必须具有相关的程序设计知识才行。对于普通的系统管理员用户，可经常关注Linux相关的网站，通过其他程序员编写的相关程序来构建自己的安全操作系统。除了系统内核外，在Linux上运行的绝大多数应用程序也是开放的，大部分可通过免费方式获取。因此，使用Linux操作系统环境可省去使用其他操作系统所必需的大笔费用。

2.多用户多任务环境所谓多用户，是指系统资源可以被不同用户使用，每个用户对自己的资源（如文件和设备等）有特定权限，互不影响。而多任务是现代计算机的主要特点，是指计算机同时执行多个程序，且各程序相互独立运行。只有很少的操作系统能提供真正的多任务能力。尽管许多操作系统声明支持多任务，但并不完全准确，如Windows等。而Linux则充分利用了x86 CPU的任务切换机制，实现了真正的多任务、多用户环境，允许多个用户同时执行不同的程序，并且可以给紧急任务以较高的优先级。

3.良好的用户界面Linux向用户提供了两种界面，即字符界面和图形界面。在配置较差的计算机中，可优先使用字符界面。此时，系统管理员通过在字符界面中输入相关的控制、配置命令对操作系统进行控制。在字符界面下进行操作，要求操作人员要熟练记住Linux的相关指令（多达上千条）。而对于配置较好的计算机，则可以使用图形界面。Linux的图形界面称为X Window系统。X Window的操作界面类似于微软的Windows界面，操作人员可以利用鼠标、菜单、窗口和滚动条等设施方便地进行操作。X Window界面给用户呈现了一个直观、易操作、交互性强、友好的图形化界面。所谓设备独立性，是指Linux操作系统将所有的外围设备都作为文件来进行处理。在使用这些外围设备之前，只要将这些设备的驱动程序安装好，以后就可以像访问系统中的文件一样去访问这些设备了，而不需要知道这些设备在系统中的具体存在形式。Linux是具有设备独立性的操作系统，其内核具有高度的适应能力。随着更多的程序员加入Linux编程，会有更多的硬件设备加入到各种Linux内核和发行版本中。这样，用户就可以与使用文件相同的方法来控制、使用这些设备。由于用户可以免费得到Linux的源代码，因此，有经验的用户也可以自己修改内核源代码，以便增加新的外围设备。

《零点起步》

编辑推荐

《零点起步:嵌入式Linux编程入门与开发实例》从零开始，轻松入门，图解案例，清晰直观，图文并茂，操作简单，实例引导，专业经典，学以致用，注重实践，实例源代码、电子，教案和相关视频。

《零点起步》

精彩短评

- 1、学校规定用书，感觉不错
- 2、这个很不错，支持了
- 3、阅读了本书，写的太笼统，太糊弄事儿了，很难从中学习到有用的知识。

《零点起步》

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:www.tushu000.com