

<<嵌入式软件开发及C语言实现>>

图书基本信息

书名：<<嵌入式软件开发及C语言实现>>

13位ISBN编号：9787121063091

10位ISBN编号：7121063093

出版时间：2008-5

出版时间：电子工业出版社

作者：魏永明

页数：400

字数：666000

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>

## <<嵌入式软件开发及C语言实现>>

### 内容概要

随着高端消费类电子产品(智能手机、便携式移动多媒体终端、可视电话、IPTV等)的广泛应用,嵌入式产品的开发越来越得到开发人员及厂商的关注。

本书以嵌入式产品开发中最热门的技术——嵌入式图形中间件技术为例,通过剖析业内广泛使用的嵌入式图形中间件MiniGUI,讲述使用C语言开发嵌入式软件过程中的常见问题及其解决办法,从而帮助读者了解嵌入式软件的开发过程及原理,掌握嵌入式软件开发的常见工具及一般性方法,并学习如何恰当采用设计技巧来提升嵌入式软件的可移植性、性能及可维护性等。

本书适合嵌入式软件开发人员、计算机相关专业高年级本科生及研究生阅读。

本书以嵌入式产品开发中最基础的技术——嵌入式图形中间件技术为例,通过剖析业内广泛使用的嵌入式图形中间件MiniGUI,讲述使用c语言开发嵌入式软件过程中的常见问题及其解决办法,从而帮助读者了解嵌入式软件的开发过程及原理,掌握嵌入式软件开发的常见工具及一般性方法,并学习如何恰当采用设计技巧来提升嵌入式软件的可移植性、性能、可维护性等。

## <<嵌入式软件开发及C语言实现>>

### 作者简介

魏永明曾就读于清华大学，是MiniGUI的主要开发者，飞漫软件的创始人。他为Linux在中国的普及和应用做了大量工作，著有《Linux实用教程》与《学用Linux与WindowsNT》，并主持翻译了《RedcHatLinux奥秘》。  
《Linux编程宝典》。  
《Linux设备驱动程序》（第二版。  
第三版）等大批

## &lt;&lt;嵌入式软件开发及C语言实现&gt;&gt;

## 书籍目录

第一篇 使用嵌入式开发工具	第1章 综述	1.1 常用嵌入式操作系统	1.1.1 实时嵌入式操作系统的一般结构	1.1.2 常用(实时)嵌入式操作系统	1.2 嵌入式Linux / uClinux	1.3 eCos操作系统	1.4 MiniGUI简介	1.4.1 嵌入式产品开发中常用的图形解决方案	1.4.2 MiniGUI的起源和发展	1.4.3 基于MiniGUI的嵌入式系统软件结构	1.4.4 MiniGUI的运行模式	1.5 嵌入式软件的开发过程及相关概念	1.5.1 一般开发过程	1.5.2 交叉编译	1.5.3 内核移植	1.5.4 驱动程序开发	1.5.5 应用软件开发及调试	1.6 小结	第2章 运行嵌入式操作系统	2.1 嵌入式Linux的体系结构	2.1.1 Linux系统的构成	2.1.2 内核	2.1.3 根文件系统	2.1.4 函数库和应用程序	2.2 嵌入式Linux的开发流程、方法和开发环境	2.2.1 基本的开发流程和方法	2.2.2 建立开发环境	2.2.3 GNU交叉开发工具链	2.2.4 终端模拟程序	2.3 系统引导	2.3.1 嵌入式Linux的启动过程	2.3.2 引导装载器	2.3.3 系统引导方式	2.4 内核的选择、编译与安装	2.4.1 选择内核	2.4.2 配置和编译内核	2.5 准备根文件系统	2.5.1 根文件系统的基本结构	2.5.2 函数库	2.5.3 内核映像和内核模块	2.5.4 设备文件	2.5.5 安装系统程序	2.5.6 系统初始化	2.6 选择和安装文件系统	2.6.1 存储设备和文件系统	2.6.2 各种类型文件系统的特性	2.6.3 使用NFS	2.6.4 CRAMFS	2.6.5 JFFS2	2.6.6 ROMFS	2.6.7 使用RAM disk	2.7 在SkyEye上运行ARM Linux	2.7.1 安装SkyEye	2.7.2 安装交叉编译工具链	2.7.3 配置、编译和运行内核	2.7.4 构建根文件系统	2.8 在Xcopilot上运行uClinux	2.8.1 安装Xcopilot	2.8.2 安装交叉编译工具链	2.8.3 配置、编译和运行uClinux	2.9 运行eCos的Linux Synthetic目标	2.9.1 获取并安装eCos及其开发工具	2.9.2 配置并编译eCos	2.9.3 编译eCos Linux Synthetic应用程序	2.10 小结	第3章 编译并运行嵌入式应用程序	3.1 交叉编译及相关工具	3.2 利用Makefile维护嵌入式应用工程	3.2.1 make和makefile的简单回顾	3.2.2 编写针对交叉编译的Makefile文件	.....	第4章 在P上运行MiniGUI	第5章 在嵌入式操作系统上运行MiniGUI	第二篇 嵌入式应用软件开发的特殊性	第6章 嵌入式软件和桌面软件的主要不同	第7章 嵌入式处理器的特殊性	第8章 嵌入式系统中的内存管理	第9章 浮点运算和定点运算	第10章 外部数据的处理	第11章 在程序中内嵌汇编	第三篇 MiniGUI剖析	第12章 MiniGUI的体系结构	第13章 灵活运用设计模式	第14章 面向对象技术的运用	第15章 可移植性及其实现	第16章 降低程序的资源占用	第17章 抽象数据源接口	附录A 安装开发用LinuxP	附录B MiniGUI-GPL版本的配置选项
---------------	--------	---------------	----------------------	---------------------	------------------------	--------------	---------------	-------------------------	---------------------	---------------------------	--------------------	---------------------	--------------	------------	------------	--------------	-----------------	--------	---------------	-------------------	------------------	----------	-------------	----------------	---------------------------	------------------	--------------	------------------	--------------	----------	---------------------	-------------	--------------	-----------------	------------	---------------	-------------	------------------	-----------	-----------------	------------	--------------	-------------	---------------	-----------------	-------------------	-------------	--------------	-------------	-------------	------------------	-------------------------	----------------	-----------------	------------------	---------------	-------------------------	------------------	-----------------	-----------------------	------------------------------	-----------------------	-----------------	----------------------------------	---------	------------------	---------------	-------------------------	--------------------------	---------------------------	-------	------------------	------------------------	-------------------	---------------------	----------------	-----------------	---------------	--------------	---------------	---------------	-------------------	---------------	----------------	---------------	----------------	--------------	-----------------	------------------------

## &lt;&lt;嵌入式软件开发及C语言实现&gt;&gt;

## 章节摘录

第一篇 使用嵌入式开发工具 第1章 综述 1.1 常用嵌入式操作系统 1.1.1 实时嵌入式操作系统的一般结构 为了达到硬件实时特性，与一般的通用操作系统不同，实时嵌入式操作系统（Real-Time Embedded Operating System, RTOS）通常都运行在直接寻址模式下，该模式是相对于虚拟内存管理机制而言的。

因为是直接寻址，实时嵌入式操作系统一般不提供类似UNIX操作系统那样具有独立地址空间的进程机制，也就是说，实时嵌入式操作系统中的所有任务都共享同一个地址空间。

在此基础上，RTOS的内核提供任务创建和调度、消息队列、信号量、互斥锁、事件标志等基本任务同步及通信机制。

仅仅有内核，我们还无法方便地开发嵌入式产品。

通常，RTOS产品提供C语言接口，内核一般也用C语言及汇编语言编写，加上符合ISO C（ANSI C）标准的基本C函数库，我们就可以基于该RTOS开发自己的嵌入式产品了。

“内核接+ANSI C库”的模式，是大多数嵌入式操作系统开发采用的编程模式。

但是，这种模式还存在一些问题。

如果我们要使用ANSI C库中的标准I/O接口，则需要操作系统提供文件系统及字符输出的支持；如果要使用ANSI C库中的内存管理函数（malloc/free函数族），就需要提供针对具体硬件的堆管理方案及实现代码。

因此，RTOS通常都设计为模块化的软件系统，需要什么样的功能，可向RTOS产品厂商购买对应的模块。

比如pSOS操作系统，它由实时多任务核心pSOS+、TCP/IP协议堆栈pNA+、远程过程调用库pRPC+、文件系统管理pHILE+、ANSI C标准库pREPC+、调试功能模块pROBE+、系统信息实时分析工具pMONT+等模块组成。

实际上，不同RTOS之间，除了在任务管理上的核心区别之外，其他的主要区别在外围模块上。图1-1给出了RTOS的一般软件结构。

## <<嵌入式软件开发及C语言实现>>

### 编辑推荐

随着高端消费类电子产品（智能手机、便携式移动多媒体终端、可视电话、IPTV等）的广泛应用，嵌入式产品的开发越来越得到开发人员及厂商的关注。  
《嵌入式软件开发及C语言实现：MiniGUI剖析》应运而生。

<<嵌入式软件开发及C语言实现>>

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>