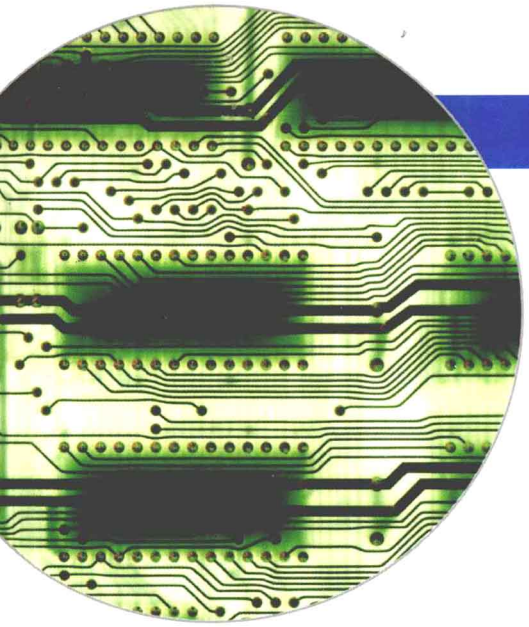
Broadview®
www.broadview.com.cn

华清远见
FARSIGHT®
嵌入式培训专家



华清远见系列图书

- 多年工程经验为基础，看懂学会为目标
- 深入解析VxWorks内核映像类型、启动方式、内核结构层次
- 详解VxWorks字符设备驱动、串口驱动、块设备驱动Flash设备驱动、网络设备驱动、USB设备驱动原理和结构
- 书中代码注释详尽，实例丰富

VxWorks
设备驱动研发详解

◎ 曹桂平 等编著　　华清远见嵌入式培训中心　审校

電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

华清远见系列图书

VxWorks
设备驱动开发详解

◎ 曹桂平 等编著　华清远见嵌入式培训中心 审校

電子工業出版社
Publishing House of Electronics Industry
北京•BEIJING

内 容 提 要

本书结合作者实际的开发经验，对 VxWorks 操作系统内部的机制及其各类设备驱动的开发进行了深入的讲解和分析。全书分 3 篇共 11 章，第 1 篇对 VxWorks 操作系统的主要组件如任务、任务调度、任务间通信、内存管理、中断处理进行了较为细致的分析；其后对很多 VxWorks 开发者不甚了解的 VxWorks 内核映像类型以及启动方式和流程进行了详细的说明和解析。第 2 篇是驱动开发的准备阶段，着重介绍了 VxWorks 系统下设备驱动的内核层次结构。第 3 篇作为本书的重点，每章对应一类驱动，结合开发实例，详细而完整地分析了 VxWorks 下普通字符设备驱动、串口驱动、普通块设备驱动、Flash 设备驱动、网络设备驱动以及 USB 设备驱动的设计和实现。

本书面向广大工程技术工作者，既可作为高等院校教师和相关专业学生的教材，又可作为各类培训班的培训教程。

图书在版编目（CIP）数据

VxWorks 设备驱动开发详解 / 曹桂平等编著.—北京：电子工业出版社，2011.3
（华清远见系列图书）
ISBN 978-7-121-12828-8

Ⅰ. ①V… Ⅱ. ①曹… Ⅲ. ①实时操作系统，VxWorks－软件开发 Ⅳ. ①TP316.2

中国版本图书馆 CIP 数据核字（2011）第 013548 号

策划编辑：胡辛征
责任编辑：李利健
印　　刷：
装　　订：北京中新伟业印刷有限公司
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：26.25　字数：672 千字
印　　次：2011 年 3 月第 1 次印刷
印　　数：4000 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 zlts@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

VxWorks 是较为常用的嵌入式硬实时操作系统，在很多领域都有其应用的身影，然而由于 VxWorks 操作系统源代码不公开，虽然文档中对各种驱动设计都有说明，但在实际应用中还是会遇到很多问题。本书根据作者工作中的一些驱动经验，结合 Wind River 提供的一些官方文档和开发环境下的源代码写成，对 VxWorks 下各种常见设备的驱动都做了比较详细的介绍和分析，可以作为 VxWorks 下设备驱动设计的指南。另外，对于各种类型的 VxWorks 启动方式以及映像文件组成进行了较为细致的分析，将澄清 VxWorks BSP 开发者具有的很多疑问。本书既可作为 VxWorks 初学者的学习材料，也可作为 VxWorks 老手的参考资料。

本书内容

本书分为 3 篇共 11 章。

第 1 篇为 VxWorks 操作系统快速入门篇，共包括 3 章内容。

第 1 章简单介绍了嵌入式系统，并对 VxWorks 操作系统的特性进行了简单说明。

第 2 章对 VxWorks 操作系统的主要组成进行了比较详细的介绍，包括任务、任务调度、任务间通信、内存管理、中断处理。书中内容不是翻译官方的文档，而是结合作者的经验有感而发。

第 3 章对 VxWorks 内核映像类型以及启动方式进行了详细分析，并对下载方式中使用的 bootrom 进行了较为深入的分析和介绍，此后对 VxWorks 操作系统的启动过程进行了梳理。本章将澄清读者对 VxWorks 启动方面的很多疑问。

第 2 篇为 VxWorks 设备驱动起步篇，共包括 2 章内容。

第 4 章讨论了驱动程序的基本功能和结构，对驱动程序中常用的一些策略以及注意事项进行了介绍。

第 5 章介绍了 VxWorks 设备驱动的内核结构层次，着重对 I/O 子系统及其维护的三张系统表进行了讨论，并对 VxWorks 下已有的几个较为常用的驱动以代码示例的方式介绍其使用方法。

第 3 篇为本书的重点，对 VxWorks 下六大核心设备驱动进行了详细的分析和介绍，六类驱动中每类驱动对应一章内容，故本篇共包括 6 章内容。

第 6 章开始进入具体设备驱动的设计，本章将从结构层次最简单的普通字符设备驱动开始讲起，以一个 SPI 接口驱动代码为例，着重讨论了普通字符设备驱动的结构、设计方式和具体实现。

第 7 章对串口驱动设计和实现进行了详细的分析。串口也是字符设备的一种，由于其常用性，VxWorks 内核专门提供了 TTY 中间层来提高串口驱动设计的效率，降低串口驱动设计的复杂度。

第 8 章进入第二大类设备——块设备驱动的设计和实现的分析。我们将从数据结构的知识出发，分析块设备驱动的基本结构，进而讨论其具体实现。VxWorks 下块设备驱动工作的方式比较特殊，其使用的阻塞读写方式不同于通用操作系统下的中断读写方式，这与 VxWorks 特殊的工作环境有关。

第 9 章将对 Flash 设备驱动进行详细介绍。Flash 设备是嵌入式平台上最常见和常用的设备，用以存储操作系统内核映像和用户数据。本章将对 VxWorks 内核提供的 TrueFFS 中间层进行展开，分析 Flash 设备驱动涉及的各个方面。

第 10 章进入第三大类设备——网络设备驱动的设计和实现。网络设备由于其独特的工作方式，其内核驱动层次不同于其他两类设备（字符设备、块设备），其不属于 I/O 子系统管理，而是直接工作在内核网络栈的实现下。为了简化网络设备驱动设计的复杂度，VxWorks 提供了 MUX 中间层，在该层次下实现的驱动通常被称为增强型网络驱动。本章同样也是从数据结构的知识出发，以实际项目中使用的网口驱动代码为例，逐步完成对网络设备驱动的设计和实现。

第 11 章分析了 USB 设备驱动的设计和实现。首先对 USB 本身进行了详细的介绍，之后对要驱动的对象进行了澄清。一般而言，USB 设备驱动指的是对 USB 主机或者目标机控制器的驱动，这个驱动由于与内核 USB 栈耦合较紧密，故必须对内核 USB 栈的实现有很清楚的了解才能成功地完成 USB 主机控制器的驱动开发。本章首先跟随一个 USB 类驱动层读数据请求，对请求在内核 USB 栈中的传递路径进行了跟踪，对路径上调用的关键函数以及使用的数据结构进行了较为详细的分析和介绍，之后以 Mass Storage 类驱动为例，介绍类驱动的初始化过程，并以 UHCI 控制器驱动为例，介绍主机控制器驱动的初始化过程，最后总结出了 USB 主机控制器的驱动结构，给出了驱动中两个中心函数的实现框架。

致谢

我要感谢我尊敬的慈祥的奶奶，您的勤劳善良将影响我的一生，是您教导我“自成人，方成人”，告诫我要严于律己。同时，我要感谢我的父母，你们鬓角的丝丝白发，早已印入我的心中，成为我奋斗不止的动力。

本书参与编写的人员有：曹桂平、张勇、王丽娜、周毅、林小峰、刘刚、马海波、李强、吴慧、马玉刚、冯浩、唐爱琴、李子龙、王明明、蒋志，全书由华清远见嵌入式培训中心审校。

鉴于作者能力有限，不可能对 VxWorks 下各类驱动中的每一个问题都理解到位，如果发现理解或者分析有误之处，恳请广大读者批评、指正，读者也可通过邮件（邮箱地址为 ingdxdy@gmail.com）与作者联系。

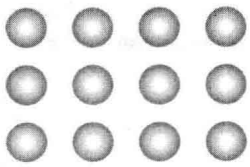
曹桂平

目　录

第 1 篇　VxWorks 操作系统快速入门篇

第 2 篇 VxWorks 设备驱动起步篇

第 3 篇 VxWorks 之六大核心设备驱动



第 1 篇

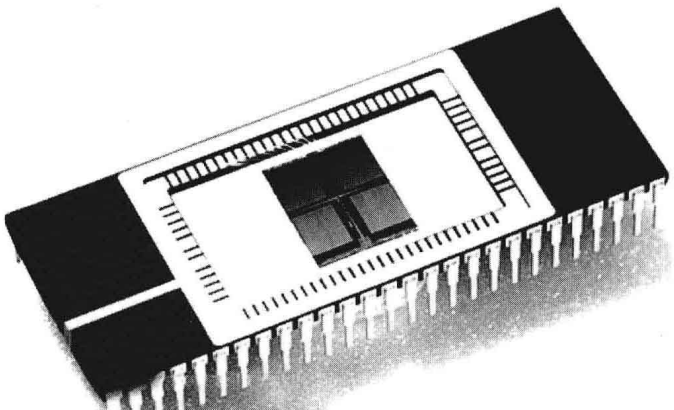
VxWorks 操作系统快速入门篇

本篇共包括 3 章内容，着重介绍 VxWorks 操作系统方面的相关知识。第 1 章从总体上介绍了嵌入式实时操作系统 VxWorks 的基本特点和应用范围。第 2 章则较为深入地对 VxWorks 操作系统的组成进行了探讨，包括 VxWorks 任务的基本结构以及任务的实现方式，基于优先级的抢占式内核调度算法，以及任务间通信常用的几种机制的内在本质；其后对 VxWorks 下的内存管理进行了介绍，主要是对虚拟地址支持下内核的一些关键结构和相关文件进行了说明；本章最后讨论了 VxWorks 下的中断处理方式，即多层次的中断转移机制，并对“中断上下文中为何不可调用可引起睡眠的函数”这个传统问题进行了探讨。第3章详细介绍了 VxWorks 操作系统的启动过程，对 VxWorks 常用的两种启动方式及各方式下操作系统的内核构成进行了较为细致的讲解；接着详细分析了下载启动方式下必不可少的 BootRom 的构成、编译生成过程及其与 VxWorks 操作系统内核之间的关系，这些内容将消除很多 VxWorks 开发人员的疑点；最后介绍了 VxWorks BSP 的文件构成。

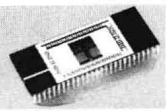
VxWorks

第1章 VxWorks嵌入式操作系统的特点与应用

本章首先从嵌入式系统的定义、组成和特点，以及发展趋势三个方面简单地对嵌入式系统进行了介绍，并对实时操作系统的特点进行了说明。接着从内核结构的角度介绍了微内核和宏内核结构的区别。最后对微内核嵌入式操作系统 VxWorks 的基本特点进行了说明。



1.1 嵌入式系统概述



1.1.1 嵌入式系统定义

嵌入式系统的定义有很多种，比较通用的定义为：以应用为中心，以计算机技术为基础，软硬件可裁减，迎合特定的应用环境，对功能、可靠性、成本、体积、功耗方面要求严格的专用计算机系统。

按照电气和电子工程师学会（IEEE）的定义，嵌入式系统是用来控制、监控或者辅助操作机器、装置、工厂等大规模系统的设备（devices used to control，monitor，or assist the operation of equipment，machinery or plants）。这个定义主要是从嵌入式系统的用途方面来进行定义的。

1.1.2 嵌入式系统组成和特点

根据以上嵌入式系统的定义，我们可以看出，嵌入式系统是由硬件和软件相结合组成的具有特定功能、用于特定场合的独立系统。其硬件主要由嵌入式微处理器、外围硬件设备组成；其软件主要包括底层系统软件和用户应用软件。嵌入式系统具有如下特点。

（1）专用、软硬件可裁减可配置

从嵌入式系统的定义可以看出，嵌入式系统是面向应用的，与通用系统最大的区别在于嵌入式系统的功能专一。根据这个特性，嵌入式系统的软、硬件可以根据需要进行精心设计、量体裁衣、去除冗余，以实现低成本、高性能。也正因如此，嵌入式系统采用的微处理器和外围设备种类繁多，系统不具通用性。

（2）低功耗、高可靠性、高稳定性

嵌入式系统大多用在特定场合，要么是环境条件恶劣，要么要求其长时间连续运转。因此，嵌入式系统应具有高可靠性、高稳定性、低功耗等性能。

（3）软件代码短小精悍

由于成本和应用场合的特殊性，嵌入式系统的硬件资源（如内存等）通常都比较少，因此，对嵌入式系统的设计也提出了较高的要求。嵌入式系统的软件设计尤其要求高质量，要在有限的资源上实现高可靠性、高性能的系统。虽然随着硬件技术的发展和成本的降低，在高端嵌入式产品上也开始采用嵌入式操作系统，但其和 PC 资源比起来还是少得可怜。所以，嵌入式系统的软件代码依然要在保证性能的情况下，占用尽量少的资源，保证产品的高性价比，使其具有更强的竞争力。

（4）代码可固化

为了提高执行速度和系统可靠性，嵌入式系统中的软件一般都固化在存储器芯片或单片机本身中，而不是存储于磁盘中。

（5）实时性

很多采用嵌入式系统的应用具有实时性要求，所以，大多数嵌入式系统采用实时性系统。

需要注意的是，嵌入式系统不等于实时系统。

（6）弱交互性

嵌入式系统不仅功能强大，而且要求使用灵活、方便，一般不需要类似键盘、鼠标等之类的工具。人机交互以简单方便为主。

（7）需要专门的开发工具和开发环境

嵌入式系统软件开发通常需要专门的开发工具和开发环境。

（8）要求开发、设计人员有较高的技能

嵌入式系统是将先进的计算机技术、半导体技术和电子技术与各个行业的具体应用相结合后的产物。这一点就决定了它必然是一个技术密集、资金密集、高度分散、不断创新的知识集成系统，从事嵌入式系统开发的人才也必须是复合型人才。

1.1.3 嵌入式系统发展趋势

未来嵌入式系统的发展趋势主要有 8 个方面。

（1）小型化、智能化、网络化、可视化

随着技术水平的提高和人们生活的需要，嵌入式设备（尤其是消费类产品）正朝着小型化、便携式和智能化的方向发展。如果你携带笔记本电脑外出办事，你肯定希望它轻薄小巧，甚至希望有一种更便携的设备来替代它，目前的上网本、MID（移动互联网设备）、便携式投影仪等都是因类似的需求而出现的。对嵌入式而言，可以说是已经进入了嵌入式互联网时代（有线网、无线网、广域网、局域网的组合），嵌入式设备和互联网的紧密结合，更为我们的日常生活带来了极大的方便和无限的想象空间。嵌入式设备功能越来越强大，未来我们的冰箱、洗衣机等家用电器都将实现网上控制；异地通信、协同工作、无人操控场所、安全监控场所等的可视化也已经成为了现实，随着网络运载能力的提升，可视化将得到进一步完善。人工智能、模式识别技术也将在嵌入式系统中得到应用，使得嵌入式系统更具人性化、智能化。

（2）多核技术的应用

人们需要处理的信息越来越多，这就要求嵌入式设备运算能力更强。因此，需要设计出更强大的嵌入式处理器，多核技术处理器在嵌入式中的应用将更普遍。

（3）低功耗（节能）、绿色环保

在嵌入式系统的硬件和软件设计中都在追求更低的功耗，以求嵌入式系统能获得更长的、可靠的工作时间，如手机的通话和待机时间、MP3 听音乐的时间等。同时，绿色环保型嵌入式产品将更受人们青睐，在嵌入式系统设计中也会更多地考虑如辐射和静电等问题。

（4）云计算、可重构、虚拟化等技术被进一步应用到嵌入式系统中

简单地讲，云计算是将计算分布在大量的分布式计算机上，这样我们只需要一个终端，就可以通过网络服务来实现我们需要的计算任务，甚至是超级计算任务。云计算（Cloud Computing）是分布式处理（Distributed Computing）、并行处理（Parallel Computing）和网格计算（Grid Computing）的发展，或者说是这些计算机科学概念的商业实现。在未来几年里，云计算将得到进一步发展与应用。

可重构性是指在一个系统中，其硬件模块或（和）软件模块均能根据变化的数据流或控

制流对系统结构和算法进行重新配置（或重新设置）。可重构系统最突出的优点就是能够根据不同的应用需求，改变自身的体系结构，以便与具体的应用需求相匹配。

虚拟化是指计算机软件在一个虚拟的平台上而不是真实的硬件上运行。虚拟化技术可以简化软件的重新配置过程，易于实现软件的标准化。其中 CPU 的虚拟化可以单 CPU 模拟多 CPU 并行运行，允许一个平台同时运行多个操作系统，并且都可以在相互独立的空间内运行而互不影响，从而提高工作效率和安全性，虚拟化技术是降低多内核处理器系统开发成本的关键。虚拟化技术是未来几年最值得期待和关注的关键技术之一。

随着各种技术的成熟与在嵌入式系统中的应用，将不断为嵌入式系统增添新的魅力和发展空间。

（5）嵌入式软件开发平台化、标准化、系统可升级，代码可复用将更受重视

嵌入式操作系统将进一步走向开放、开源、标准化和组件化。嵌入式软件开发平台化也将是今后的一个趋势，越来越多的嵌入式软硬件行业标准将出现，最终的目标是使嵌入式软件开发简单化，这也是一个必然规律。同时随着系统复杂度的提高，系统可升级和代码复用技术在嵌入式系统中得到更多的应用。另外，因为嵌入式系统采用的微处理器种类多，不够标准，所以在嵌入式软件开发中将更多地使用跨平台的软件开发语言与工具。目前，Java 语言正在被越来越多地使用到嵌入式软件开发中。

（6）嵌入式系统软件将逐渐 PC 化

需求和网络技术的发展是嵌入式系统发展的一个原动力，随着移动互联网的发展，将进一步促进嵌入式系统软件 PC 化。如前所述，结合跨平台开发语言的广泛应用，未来的嵌入式软件开发的概念将被逐渐淡化，也就是嵌入式软件开发和非嵌入式软件开发的区别将逐渐减小。

（7）融合趋势

嵌入式系统软硬件融合、产品功能融合、嵌入式设备和互联网的融合趋势加剧。嵌入式系统设计中软硬件结合将更加紧密，软件将是其核心。消费类产品将在运算能力和便携方面进一步融合。传感器网络将迅速发展，其将极大地促进嵌入式技术和互联网技术的融合。

（8）安全性

随着嵌入式技术和互联网技术的结合发展，嵌入式系统的信息安全问题日益凸显，保证信息安全也成了嵌入式系统开发的重点和难点。

1.1.4 实时操作系统

1. 实时操作系统定义

实时操作系统（RTOS）是指当外界事件或数据产生时，能够接收并以足够快的速度予以处理，其处理的结果又能在规定的时间内来控制生产过程或对处理系统作出快速响应，并控制所有实时任务协调一致运行的操作系统。因而，提供及时响应和高可靠性是其主要特点。实时操作系统有硬实时和软实时之分，硬实时要求在规定的时间内必须完成操作，这是在操作系统设计时保证的；软实时则只要按照任务的优先级，尽可能快地完成操作即可。我们通常使用的操作系统在经过一定改变之后就可以变成实时操作系统。

实时操作系统是保证在一定时间限制内完成特定功能的操作系统。例如，可以为确保生

产线上的机器人能获取某个物体而设计一个操作系统。在“硬”实时操作系统中，如果不能在允许的时间内完成使物体可达的计算，操作系统将因错误结束。在“软”实时操作系统中，生产线仍然能继续工作，但产品的输出会因产品不能在允许时间内到达而减慢，这使机器人有短暂的不生产现象。一些实时操作系统是为特定的应用设计的，另一些是通用的。一些通用目的的操作系统称为实时操作系统。但某种程度上，大部分通用目的的操作系统有实时系统的特征。这就是说，即使一个操作系统不是严格的实时系统，它们也能解决一部分实时应用的问题。

2．实时操作系统的特征

实时操作系统具有以下特点：

- 多任务。
- 有线程优先级。
- 多种中断级别。

小的嵌入式操作系统经常需要实时操作系统，内核要满足实时操作系统的要求。

3．实时操作系统的相关概念

（1）基本概念

代码临界段：指处理时不可分割的代码。一旦这部分代码开始执行，则不允许中断打扰。

资源：任何为任务所占用的实体。

共享资源：可以被一个以上任务使用的资源。

任务：也称为一个线程，是一个简单的程序。每个任务被赋予一定的优先级，有它自己的一套 CPU 寄存器和自己的栈空间。典型的是，每个任务都是一个无限的循环，每个任务都处在以下五个状态：休眠态、就绪态、运行态、挂起态、被中断态。

任务切换：将正在运行任务的当前状态（CPU 寄存器中的全部内容）保存在任务自己的栈区，然后把下一个将要运行的任务的当前状态从该任务的栈中重新装入 CPU 的寄存器，并开始下一个任务的运行。

内核：负责管理各个任务，为每个任务分配 CPU 时间，并负责任务之间的通信。它分为不可剥夺型内核和可剥夺型内核。

调度：是内核的主要职责之一，决定轮到哪个任务运行。一般基于优先级调度法。

（2）关于优先级的问题

任务优先级：分为优先级不可改变的静态优先级和优先级可改变的动态优先级。

优先级反转：优先级反转问题是实时系统中出现最多的问题。共享资源的分配可导致优先级低的任务先运行，优先级高的任务后运行。解决的办法是使用“优先级继承”算法来临时改变任务优先级，以遏制优先级反转。

（3）互斥

虽然共享数据区简化了任务之间的信息交换，但是必须保证每个任务在处理共享数据时的排他性。使之满足互斥条件的一般方法有：关中断、使用测试并置位指令（TAS）、禁止做任务切换、利用信号量。

因为采用实时操作系统的意义就在于能够及时处理各种突发事件，即处理各种中断，因而衡量嵌入式实时操作系统的最主要、最具有代表性的性能指标参数无疑应该是中断响应时