

<< μ C/OS ARM移植要点详解 >>

图书基本信息

书名：<< μ C/OS ARM移植要点详解 >>

13位ISBN编号：9787810777254

10位ISBN编号：7810777254

出版时间：2005-11

出版时间：北京航空航天大学出版社

作者：黄燕平

页数：264

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>

<< μ C/OS ARM移植要点详解 >>

内容概要

本书内容共7章，各章主题如下：第0章为嵌入式环境的选择，对嵌入式产品开发中常见的芯片、软件方案进行了简单比较分析。

第1章为OS内核概念，包括ARM微处理器特性、内核结构基础等重要概念的详细说明。

它是本书中非常重要的一章。

第2章为 μ C/OSII移植过程，是在常见ARM微处理器上移植 μ C/OSII的代码详解。

第3章为代码组织及功能设计，把嵌入式产品的设计从简单移植的角度扩展到内核整体体系结构设计及功能组件组织的角度并引入一个有益的、重要的COS组件方法。

它是本书中篇幅最长的一章，也是最重要的一章。

第4章为 μ Rtos V1.0 代码说明，介绍一种硬实时分层调度体系结构的嵌入式内核产品。

第5章为ARM开发环境，解答软件开发工具使用中的一些常见问题。

第6章为软件工程简述，对嵌入式产品开发中的软件项目管理中的要点进行了探讨，讨论了一些如何提高产品品质的技术知识。

<< μ C/OS ARM移植要点详解 >>

书籍目录

第0章 嵌入式环境的选择0.1 简介10.2 关于微处理器40.3 关于OS50.4 关于功能模块的移植60.5 关于本书7

第1章 OS内核概念1.1 嵌入式实时内核相关概念91.1.1 ARM7主要特性91.1.2 ARM特性代码121.1.3 中断与设备151.1.4 任务与调度181.1.5 临界区与保护201.2 内核结构251.2.1 硬保护泛滥问题251.2.2 硬保护泛滥问题的解决261.2.3 μ Rtos V1.0281.3 关键机制291.3.1 复位引导机制291.3.2 单层中断机制321.3.3 嵌套中断机制331.3.4 端口轮询机制361.3.5 不可屏蔽中断机制381.3.6 自保护软件FIFO391.3.7 高速处理需求综合讨论461.3.8 其他杂项481.4 关键算法逻辑501.4.1 硬保护算法501.4.2 调度器算法521.4.3 任务就绪算法571.4.4 软保护算法611.4.5 ITC算法621.4.6 OS_TCB结构631.4.7 OS_EVENT结构65

第2章 μ C/OSII移植过程2.1 头文件定义722.1.1 ARM微处理器定义732.1.2 S3C44B0微处理器定义742.1.3 LPC2214微处理器定义782.1.4 产品板定义822.2 移植代码实现842.2.1 入口代码842.2.2 C运行环境代码1002.2.3 环境切换代码102

第3章 代码组织及功能设计3.1 代码组件化技术1043.1.1 普通组件化1053.1.2 抽象组件化1123.2 设备驱动框架设计1203.2.1 ISR层设备驱动框架设计1203.2.2 高层设备驱动框架1393.3 ITC算法设计1403.3.1 软保护问题1473.3.2 ITC与任务关系1543.3.3 信号灯1613.3.4 事件1643.3.5 队列1663.4 时间片轮换调度算法1813.5 模块间衔接接口1823.5.1 套接字1853.5.2 管道1883.5.3 通用接口1913.6 状态机组件设计1923.6.1 状态机基础1933.6.2 层次化状态机特性1963.6.3 状态机组件设计2003.6.4 状态机组件的使用2033.7 杂项设计考虑2043.7.1 任务局部存储2043.7.2 循环等待死锁检查工具设计2053.7.3 内存管理设计207

第4章 μ Rtos V1.0代码说明4.1 移植目录2204.2 项目目录2224.3 内核主目录2224.4 功能目录2234.5 在 μ Rtos下开发应用产品的说明2244.6 常用设备驱动设计指南2264.6.1 人机交互串口/PPP2264.6.2 键盘2264.6.3 网口2274.7 网络协议栈设计2304.7.1 网络开发接口设计2304.7.2 TCP协议2314.7.3 TCP协议的简化实现2324.7.4 TCP协议实现的其他问题234

第5章 ARM开发环境5.1 环境的准备2365.2 ARMulator2395.2.1 中断控制器2405.2.2 时钟2415.2.3 看门狗2425.2.4 调试输出2435.2.5 堆栈跟踪器2435.3 编译器工作环境2435.3.1 汇编语言编译选项2445.3.2 C语言编译选项2465.3.3 链接器选项2465.4 代码烧写249

第6章 软件工程简述6.1 软件测试基本概念2506.2 软件工程模型2526.3 状态机的测试256

附录A 常用缩写对照表附录B 代码/伪代码目录后记参考文献

<< μ C/OS ARM移植要点详解 >>

媒体关注与评论

前言 随着国内工业化、数字化的步伐加快，嵌入式开发在IT行业中的重要性越来越显著。

中国成为“世界制造中心”甚至“设计中心”的趋势，必然导致对小型数字控制系统的需求越来越大。

在嵌入式系统开发方面，最核心的技术就是微处理器芯片和嵌入式操作系统。

其中在微处理器芯片方面，ARM已经给出了比较理想的一个答案；而在嵌入式操作系统方面，适合国内发展方向的解决方案以及系统基础结构方面并不理想。

? 风河公司的VxWorks操作系统成本高，结构复杂，不适合小型应用。

? 微软公司的WinCE操作系统更适合民用、便携式娱乐设备等。

? 开源的Linux操作系统体系结构同样复杂，产品化和商业化程度不够，即使在Linux本来的PC目标环境下，也难寻理想的技术支持，更不用说嵌入式环境下的Linux。

这方面的弱势对批量生产、大规模、长时间运行使用的工业化产品来说是致命的。

另外，在以上讨论的这3种系统中，只有VxWorks是硬实时操作系统，而WinCE和Linux是非硬实时操作系统。

在这种情况下，类似于 μ C/OSII的小型硬实时嵌入式操作系统内核具有低成本、易控制、小规模、高性能的特性，因而有相当好的发展前景。

但是这类系统的基础较为薄弱，面临产品化和商业化程度不够的局面。

采用此类系统进行产品开发需要仔细分析和设计，否则也很难真正满足工业产品生产的要求。

本书正是针对这种情况，在ARM微处理器环境下，针对商业化、产品化环境的严格要求，设计、构造了一种硬实时嵌入式内核体系结构。

当然，真正的商业化、产品化的嵌入式内核既需要这种能够满足高标准要求的体系结构设计基础，又需要严格的产品化软件开发测试过程。

只有理论基础与工程实践完整地结合，才能产生真正经得起考验的，能够满足工业化生产，能够在各种环境下稳定运行并确保达到设计目的的产品。

从这个角度考虑，仅仅拿来一个操作系统内核并开发应用产品很难完全满足这种要求。

必须要对内核的设计思路进行仔细的考虑和验证，对应用的可选开发设计方法进行审慎的评估，并配合真正工业化的项目开发管理办法，才能保障产品达到要求。

本书中提到的 μ Rtos V1.0内核，正是作者及其所在团队按照以上精神付出巨大努力严格设计、测试的产品。

该内核的体系结构设计思路在本书中有充分详细的解释和说明。

另外， μ C/OSII是读者在市场上可以方便获得的一种“半开源”的操作系统内核。

本书针对该内核在ARM下的移植以及与本书所述内核体系结构的关系及比较，进行了详细解说。

通过对比，既方便 μ C/OSII的爱好者、使用者学习掌握 μ C/OSII内核，同时又在对比分析过程中，使读者掌握 μ C/OSII和 μ Rtos V1.0内核各自的详细特征、特点，方便读者在此基础上开发设计出更好的嵌入式系统产品。

作者 2005年7月25日

<< μ C/OS ARM移植要点详解 >>

编辑推荐

本书适合的读者是对ARM微处理器有一定了解，对嵌入式内核有一定了解和对嵌入式产品开发有一定经验的读者。

对于从事嵌入式产品开发，特别是基于ARM的嵌入式产品开发的项目经理、体系结构设计师、设计师、代码开发工程师、测试工程师，解决实际问题有一定的帮助。

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>