

<<嵌入式实时操作系统C/OS-III应用>>

图书基本信息

书名：<<嵌入式实时操作系统C/OS-III应用开发-基于STM32微控制器>>

13位ISBN编号：9787512409873

10位ISBN编号：7512409877

出版时间：2012-11

出版时间：北京航空航天大学出版社

作者：拉伯罗斯

页数：178

字数：224000

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>

<<嵌入式实时操作系统C/OS-III应用>>

内容概要

原书的第1部分宽泛地讲述实时内核，把 μ C/OS- 作为实时内核的实例加以介绍。

拉伯罗斯编著的《嵌入式实时操作系统 μ C\OS- 应用开发》（原书第2部分）则看起来完全不同，它给出了流行的微控制器STM32介绍、评估板原理图和实际开发的6个范例，包括译者补充的2个范例：

嵌入式WiFi和文件系统 μ C/FS。

这些是其他书籍涉及不多的。

精彩的部分是书中的附录，详细解释了 μ C/OS- 移植到ARM

Cortex-M3的全过程， μ C/OS- 针对

ARMcort-M3架构的移植代码说明和 μ C/Probe的使用介绍，这是 μ C/OS- 书籍中涉及移植部分最严谨和最具权威性的内容。

本书参考的硬件是原书指定的STM32F107评估板，中国版略有修改已经上市，书中的应用实例指定采用IAR EW

ARM编译器、汇编器、链接器和调试器开发工具，这就使读者能够很方便地体验 μ C/OS- ，从而精通 μ C/OS- 的使用。

《嵌入式实时操作系统 μ C\OS- 应用开发》的读者是嵌入式RTOS爱好者和STM32初学者以及电子设计的工程师们，也可作为高等院校本科生和研究生嵌入式系统和单片机类课程的教材。

作者简介

Jean

Labrosse先生于1999年创立了Micri μ m公司，他经常为波士顿和硅谷的嵌入式系统大会以及其他行业的会议提供讲座和报告。

同时，他是两本嵌入式设计经典书籍的作者MicroC / OS-II，The ReakTime Kernel和Embedded Systems Building Blocks，Complete end Ready-to-Use Modules in C。

Jean

Labrosse先生在加拿大魁北克省舍布鲁克大学获得电子工程学士学位和硕士学位。

1984年和1991年毕业于北京航空航天大学，获得自动化学士和计算机科学硕士学位。

女，2003年毕业于华东地质学院电子信息工程专业，2003年9月升八本校研究生部，主要研究方向为智能仪器与测控技术。

2006年加入北京麦克泰软件技术有限公司，主要从事嵌入式实时操作系统 μ C/OS的技术支持和应用开发。

书籍目录

- 第1章 简介
- 第2章 ARM Cortex-M3和STM32
- 第3章 准备和设置
- 第4章 μ C/OS- 应用实例1
- 第5章 μ C/OS- 应用实例2
- 第6章 μ C/OS- 应用实例3
- 第7章 μ C/OS- 应用实例4
- 第8章 μ C/OS- 应用实例5
- 第9章 μ C/OS- 应用实例6
- 第10章 IAR EWARM开发工具的使用
- 附录A μ C/OS- 移植到Cortex-M3
- 附录B μ C/CPU移植到Cortex-M3
- 附录C IAR公司IAR Embedded Workbench for ARM
- 附录D Micri μ m的 μ C/Probe
- 附录E μ C/Eval-STM32F107用户指南
- 附录F 参考文献
- 附录G μ C/OS-

章节摘录

版权页：插图：8.1.1 $\mu\text{C}/\text{FS}$ 特点 POSIX兼容的文件访问接口（FOPEN、FREAD等）和目录访问（opendir、readdir等）与处理器无关 轻松地移植到新平台 RAM和ROM空间可调 支持FAT12 / 16 / 32和长文件名（VFAT）可选的日志组件，实现FAT掉电保护 支持设备格式化和创建分区 $\mu\text{C}/\text{FS}$ 的代码是用ANSI C写的，适用于所有处理器。

$\mu\text{C}/\text{FS}$ 具有如下一些特点： 支持与MS—DOS / Windows兼容的FAT12、FAT16和FAT32文件系统。

支持多种设备驱动。

$\mu\text{C}/\text{FS}$ 支持各种不同的设备驱动，从而允许用户在同一时间通过文件系统访问不同类型的硬件。

支持多种存储介质。

通过设备驱动允许用户在同一时间访问不同的介质。

支持操作系统。

其他操作系统，包括 $\mu\text{C}/\text{OS}$ — 可以很方便地与 $\mu\text{C}/\text{FS}$ 结合，这样用户就可以在多线程环境下进行文件操作。

为用户的应用程序提供类似于stdio.h的API，它是用ANSI C写的，所以一个用标准C I / O库的应用程序可以方便地移植以使用 $\mu\text{C}/\text{FS}$ 。

非常简单的设备驱动结构。

$\mu\text{C}/\text{FS}$ 只需要读写分区的底层函数，所以要支持用户定制的硬件也很简单。

提供以下设备的驱动：SMC、SD、MMC、CF、IDE、RAMdisk和Windows（允许用户在Windows环境下使用仿真软件）。

8.1.2 $\mu\text{C}/\text{FS}$ 文件系统结构 $\mu\text{C}/\text{FS}$ 由API层、文件系统层、逻辑块层及设备驱动层组成，文件系统结构如图8—1所示。

API层（API Layer）API层是 $\mu\text{C}/\text{FS}$ 与用户应用程序之间的接口，包含了一个与文件函数相关的ANSI C库，如FS_Fopen（），FS—Fwrite（）等。

API层把这些调用传递给文件系统层。

目前在 $\mu\text{C}/\text{FS}$ 下只有FAT型的文件系统可以获取，但是API层可以同时处理不同类型的文件系统层，所以在 $\mu\text{C}/\text{FS}$ 下可以同时使用FAT和其他文件系统。

文件系统层（File System Layer）文件系统层把文件操作请求传递给逻辑块操作，通过这种传递文件系统调用逻辑块操作来为设备指定相应的设备驱动。

系统驱动层（System Driver Layer）系统驱动层的主要功能是使对设备驱动的访问同步，并为文件系统层提供一个便捷的接口。

设备驱动层（Device Driver）设备驱动层是处于系统底层的例程，用以访问存储硬件。

设备驱动的结构简单，易于与用户自己的存储设备进行整合。

编辑推荐

《嵌入式实时操作系统 μ C/OS- 应用开发:基于STM32微控制器》的读者是嵌入式RTOS爱好者和STM32初学者以及电子设计的工程师们,也可作为高等院校本科生和研究生嵌入式系统和单片机类课程的教材。

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>