

嵌入式实时操作系统 μ C/OS-III 应用开发

——基于STM32微控制器

μ C/OS-III, The Real-Time Kernel For the STM32 ARM Cortex-M3

 **IAR**
SYSTEMS

[美] Jean J. Labrosse 著
何小庆 张爱华 译



北京航空航天大学出版社
BEIHANG UNIVERSITY PRESS

嵌入式实时操作系统 $\mu\text{C}/\text{OS} - \text{III}$ 应用开发 ——基于 STM32 微控制器

$\mu\text{C}/\text{OS} - \text{III}$, The Real - Time Kernel for the STM32 ARM Cortex - M3

[美] Jean J. Labrosse 著

何小庆 张爱华 译

北京航空航天大学出版社

内 容 简 介

原书的第1部分宽泛地讲述实时内核,把 $\mu\text{C}/\text{OS}-\text{III}$ 作为实时内核的实例加以介绍。本书(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 and 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 $\mu\text{C}/\text{FS}$ 。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 $\mu\text{C}/\text{OS}-\text{III}$ 移植到ARM Cortex-M3的全过程, $\mu\text{C}/\text{OS}-\text{III}$ 针对ARM Cortex-M3架构的移植代码说明和 $\mu\text{C}/\text{Probe}$ 的使用介绍,这是 $\mu\text{C}/\text{OS}-\text{III}$ 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 $\mu\text{C}/\text{OS}-\text{III}$ 的使用。

本书的读者是嵌入式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。

图书在版编目(CIP)数据

嵌入式实时操作系统 $\mu\text{C}/\text{OS}-\text{III}$ 应用开发:基于STM32微控制器/(美)拉伯罗斯著;何小庆,张爱华译.
--北京:北京航空航天大学出版社,2012.11

书名: $\mu\text{C}/\text{OS}-\text{III}$, The Real-Time Kernel for the STM32 ARM Cortex-M3
ISBN 978-7-5124-0987-3

I. ①嵌… II. ①拉… ②何… ③张… III. ①实时操作系统 IV. ①TP316.2

中国版本图书馆CIP数据核字(2012)第242235号

英文原名: $\mu\text{C}/\text{OS}-\text{III}$, The Real-Time Kernel for the STM32 ARM Cortex-M3

Copyright © 2011 by Micrium Press.

Translation Copyright © 2012 by Beijing University of Aeronautics and Astronautics Press.

All rights reserved.

本书中文简体字版由美国Micrium出版社授权北京航空航天大学出版社在中华人民共和国境内独家出版发行。版权所有。

北京市版权局著作权合同登记号 图字:01-2012-0408号

嵌入式实时操作系统 $\mu\text{C}/\text{OS}-\text{III}$ 应用开发 ——基于STM32微控制器

$\mu\text{C}/\text{OS}-\text{III}$, The Real-Time Kernel for the STM32 ARM Cortex-M3

[美] Jean J. Labrosse 著

何小庆 张爱华 译

责任编辑 王静竞

*

北京航空航天大学出版社出版发行

北京市海淀区学院路37号(邮编100191) <http://www.buaapress.com.cn>

发行部电话:(010)82317024 传真:(010)82328026

读者信箱:emsbook@gmail.com 邮购电话:(010)82316936

涿州市新华印刷有限公司印装 各地书店经销

*

开本:710×1000 1/16 印张:11.75 字数:224千字

2012年11月第1版 2012年11月第1次印刷 印数:5000册

ISBN 978-7-5124-0987-3 定价:29.00元

若本书有倒页、脱页、缺页等印装质量问题,请与本社发行部联系调换。联系电话:(010)82317024

译者序

本译作的原书《 μ C/OS - III, The Real - Time Kernel for the STM32 ARM Cortex - M3》大约在 2009 年底的时候在美国出版。出版不久,清华大学邵贝贝教授把这个信息给了我和北京航空航天大学出版社胡晓柏主任。结果我们被两个问题难住了,一是作者 Jean J. Labrosse 希望书和板子在中国也是和他们在美国一样捆绑销售,这个要求是我们国内的出版社无法做到的;二是我们自己对于市场的顾虑,因为在那个时候, μ C/OS - III 不是像以前的 μ C/OS - II 一样,虽然商业应用是要得到授权和收费的(见附录 G),但还是开放源代码的软件,而且对于教育是完全免费的。针对《 μ C/OS - III, The Real - Time Kernel for the STM32 ARM Cortex - M3》和其他 MCU 的图书,Micrium 公司和作者只是提供 μ C/OS - III 目标代码的版本。我们担心,中国的用户是否能够接受这个变化。

一晃到了 2011 年 7 月,我看到了 μ C/OS - III 已经开放源代码的信息,再查看 www.amazon.com,发现 μ C/OS - III 图书已经分成单独发行的书和与 Micrium μ C/Eval - STM32F107 评估板绑定的套装两个版本销售。我感觉在中国出版的机会基本成熟了。我与 Jean 谈了两个月,他同意在中国出版。因为源代码开放之后,原书的部分章节需要重新修改,新的修订版本原书《 μ C/OS - III, The Real - Time Kernel for the STM32 ARM Cortex - M3》到了 10 月底我才拿到。新版增加了 50 多页,主要增加了源代码开放后移植部分的说明,而且 Jean 也没有坚持绑定开发板的要求。在 Micrium、意法半导体有限公司大华区和上海庆科信息技术公司的配合下,Micrium μ C/Eval - STM32F107 评估板的中國版本由北京麦克泰软件技术公司顺利发布。

原书分为 2 个部分,第 1 部分是 μ C/OS - III—The Real - Time Kernel,第 2 部分是 μ C/OS - III and the STMicroelectronics STM32F107。经过多方面协调确定中文版分为两本书,也就是现在的《嵌入式实时操作系统 μ C/OS - III》(由

官辉、曾鸣、龚光华、杜强和黄土琛译,邵贝贝审校)和本书《嵌入式操作系统 $\mu\text{C}/\text{OS} - \text{III}$ 应用开发——基于 STM32 微控制器》。前者主要是介绍 $\mu\text{C}/\text{OS} - \text{III}$ 的原理、编程接口和配置,与具体的 MCU 无关;后者是介绍 $\mu\text{C}/\text{OS} - \text{III}$ 在 Cortex-M3 和 STM32 上的移植,应用实例, $\mu\text{C}/\text{OS} - \text{III}$ 的开发,基于 $\mu\text{C}/\text{Probe}$ 的测试过程和 $\mu\text{C}/\text{Eval} - \text{STM32F107}$ 评估板的介绍。这样安排为未来几本基于其他几家公司 MCU 的 $\mu\text{C}/\text{OS} - \text{III}$ 图书出版做好了铺垫准备。

回想起来,我认识 Jean 是在 2000 年左右。2003 年 1 月,我们在加拿大的蒙特利尔见了面。当时我正好在纽约参加 Linuxworld 大会,从纽瓦克机场直接飞到蒙特利尔。那时的蒙特利尔一片冰天雪地, Jean 和 Chrisian Legare 很热情地邀请我到他们的办公室,介绍他们公司(Micrium)和 $\mu\text{C}/\text{OS} - \text{II}$ 产品,并请我在当地特色饭店吃了晚餐。Jean 是 RTOS 的专家,Christan 是通信方面的专家,后来写了《嵌入式协议栈 $\mu\text{C}/\text{TCP} - \text{IP}$ 》一书,我也有幸参加中文版编辑工作。此次,这本书也会与我们的两本 $\mu\text{C}/\text{OS} - \text{III}$ 图书一同出版。借此,我很感谢北京邮电大学计算机学院邝坚副院长,他很爽快地接受了书本的翻译工作。因为我感觉他是最合适的人选,对于嵌入式系统和通信协议都很熟悉。与 Jean 的见面和交流对于我认识了解 RTOS、MCU 的技术和市场有很大帮助,这些经历在我的《我与单片机和嵌入式系统 20 年》一文中已经做了详细的叙述,这里就不赘言了。

本书的第 1、2 章和附录 C、D、F、G 由何小庆翻译;第 3~7 章和附录 A、B 和 E 由张爱华翻译,她有多年应用及支持 $\mu\text{C}/\text{OS} - \text{II}$ 和 III 的相关经验;第 8 章由徐炜撰写;第 9 章和第 10 章由张爱华和姜桥撰写。第 8~10 章是应出版社要求针对原书的补充和丰富。全书由何小庆审阅。本书编写中得到了北京麦克泰软件技术公司尹立杰先生的帮助。与本书配套的中国版 $\mu\text{C}/\text{Eval} - \text{STM32F107}$ 评估板的设计和生,得到了意法半导体中国研究中心梁平经理、意法半导体大中华区市场经理曹锦东、上海庆科信息公司王永虹先生和华东师范大学沈建华教授的支持和帮助,在此一并致谢。过去几年我虽然写了不少文章,也有 20 多年的嵌入式操作系统应用经验,但是总共只有 2 个多月的翻译、校对和编辑时间,的确非常仓促,如遇错误之处欢迎指正,我的电子邮件地址是 allan.hexq@gmail.com。

何小庆

2012 年 8 月于北京

目 录

第 1 章 简 介	1
1.1 μ C/Eval - STM32F107 评估板	1
1.2 本书的章节内容	3
1.3 作者致谢	6
第 2 章 ARM Cortex - M3 和 STM32	7
2.1 Cortex CPU	8
2.1.1 程序状态寄存器	8
2.1.2 堆栈和中断	10
2.2 嵌套向量的中断控制器	10
2.3 异常向量表	11
2.4 SysTick(系统节拍)	12
2.5 存储器映像	13
2.6 指令集	14
2.7 调试特性	15
2.8 意法半导体的 STM32	16
第 3 章 准备和设置	18
3.1 下载针对本书的 μ C/OS - III 项目	19
3.1.1 \EvalBoards	19
3.1.2 \uC - CPU	22
3.1.3 \uC - LIB	22
3.1.4 \uCOS - III	23
3.2 下载 μ C/Probe	24
3.3 下载 IAR Embedded Workbench for ARM	24

3.4	下载 STM32F107 文档	25
第 4 章	$\mu\text{C}/\text{OS} - \text{III}$ 应用实例 1	26
4.1	运行这个项目	28
4.2	实例项目是如何工作的	30
4.3	使用 $\mu\text{C}/\text{Probe}$ 观测变量	33
4.4	总 结	38
第 5 章	$\mu\text{C}/\text{OS} - \text{III}$ 应用实例 2	39
5.1	运行这个项目	40
5.2	使用 $\mu\text{C}/\text{Probe}$ 观测温度传感器	43
第 6 章	$\mu\text{C}/\text{OS} - \text{III}$ 应用实例 3	45
6.1	运行这个项目	45
6.2	使用 $\mu\text{C}/\text{Probe}$ 进行性能测试	46
6.3	测试代码是如何工作的	51
6.4	其他性能测试	56
6.5	总 结	58
第 7 章	$\mu\text{C}/\text{OS} - \text{III}$ 应用实例 4	59
7.1	运行这个项目	60
7.2	显示旋转轮数据	60
7.3	RPM 测量仿真执行	62
7.4	代码是如何工作的	65
7.5	观 测	68
7.6	总 结	70
第 8 章	$\mu\text{C}/\text{OS} - \text{III}$ 应用实例 5	71
8.1	$\mu\text{C}/\text{FS}$ 文件系统	71
8.1.1	$\mu\text{C}/\text{FS}$ 特点	72
8.1.2	$\mu\text{C}/\text{FS}$ 文件系统结构	72
8.2	基于 $\mu\text{C}/\text{FS}$ 的 SD 卡文件系统的实现	73
8.3	$\mu\text{C}/\text{FS}$ 应用	78
第 9 章	$\mu\text{C}/\text{OS} - \text{III}$ 应用实例 6	80
9.1	EMW 系列 WiFi 模块简介	80
9.2	准备软硬件环境	82
9.2.1	硬件设备	82
9.2.2	工具软件	82
9.3	硬件设计	84

9.4 软件设计	85
9.5 实验例程下载验证	89
第 10 章 IAR EWARM 开发工具的使用	91
10.1 创建工程	91
10.1.1 生成新的工作区(Workspace)	91
10.1.2 生成新项目	91
10.1.3 给项目添加文件	93
10.1.4 设置项目选件	93
10.2 编译和链接应用程序	95
10.2.1 编译源文件	96
10.2.2 链接应用程序	97
10.2.3 查看 MAP 文件	101
10.3 IAR C-SPY 调试器应用	102
10.3.1 开始调试	102
10.3.2 组织窗口	103
10.3.3 检查源代码语句	103
10.3.4 检查变量	104
10.3.5 设置和监视断点	105
10.3.6 在反汇编界面上调试	106
10.3.7 监视寄存器	107
10.3.8 查看存储器	108
10.3.9 观察 Terminal I/O	109
10.3.10 执行程序到结束	109
附录 A μC/OS-III 移植到 Cortex-M3	111
A-1 os_cpu.h	111
A-2 os_cpu.c.c	113
A-2-1 os_cpu.c.c-OSIdleTaskHook()	113
A-2-2 os_cpu.c.c-OSInitHook()	114
A-2-3 os_cpu.c.c-OSStatTaskHook()	115
A-2-4 os_cpu.c.c-OSTaskCreateHook()	115
A-2-5 os_cpu.c.c-OSTaskDelHook()	116
A-2-6 os_cpu.c.c-OSTaskReturnHook()	117
A-2-7 os_cpu.c.c-OSTaskStkInit()	118
A-2-8 os_cpu.c.c-OSTaskSwHook()	120

A-2-9	os_cpu_c.c - OSTimeTickHook()	122
A-2-10	os_cpu_c.c - OS_CPU_SysTickHandler()	123
A-2-11	os_cpu_c.c - OS_CPU_SysTickInit()	124
A-3	os_cpu_a.asm	125
A-3-1	os_cpu_a.asm - OSStartHighRdy()	125
A-3-2	os_cpu_a.asm - OSCtxSw() and OSIntCtxSw()	126
A-3-3	os_cpu_a.asm - OS_CPU_PendSVHandler()	126
附录 B	$\mu\text{C}/\text{CPU}$ 移植到 Cortex - M3	129
B-1	cpu_core.c	129
B-2	cpu_core.h	130
B-3	cpu_def.h	130
B-4	cpu_cfg.h	130
B-5	bsp.c 中的 $\mu\text{C}/\text{CPU}$ 函数	131
B-5-1	bsp.c, CPU_TS_TmrInit() 中的 $\mu\text{C}/\text{CPU}$ 函数	131
B-5-2	bsp.c 中的 $\mu\text{C}/\text{CPU}$ 函数, CPU_TS_TmrRd()	132
B-6	cpu.h	132
B-6-1	cpu.h - #defines	133
B-6-2	cpu.h - 数据类型	134
B-6-3	cpu.h - 函数原型	136
B-7	cpu_a.asm	136
附录 C	IAR 公司 IAR Embedded Workbench for ARM	138
C-1	IAR Embedded Workbench for ARM(IAR EWARM)的特点	139
C-2	模块和可扩展的 IDE	140
C-3	深度优化的 C/C++ 编译器	141
C-4	设备支持	142
C-5	先进的 C-SPY 调试器	143
C-6	C-SPY 调试器和目标系统支持	143
C-7	IAR 汇编	144
C-8	IAR I-LINK 连接器	144
C-9	IAR 库和库工具	145
C-10	文档	145
C-11	技术支持	145
附录 D	Micrium 的 $\mu\text{C}/\text{Probe}$	146

D-1	μ C/Probe 是一个 Windows 应用	148
D-2	给一个变量指定一个对象	150
附录 E	μ C/Eval - STM32F107 用户指南	152
E-1	特 性	153
E-2	硬件的布局 and 配置	153
E-3	电 源	155
E-4	启 动	156
E-5	复 位	157
E-6	CAN	157
E-7	RS-232	157
E-8	SD/MMC	157
E-9	USB-OTG	158
E-10	LM75 温度传感器	158
E-11	调试接口	158
E-12	以太网	159
E-13	时 钟	159
E-14	连接器	160
E-14-1	扩展连接器(CN3).....	160
E-14-2	电源连接器(CN1&CN9).....	161
E-14-3	WiFi 模块 EMW3280 连接器	161
E-14-4	RS-232 连接器(CN7).....	162
E-15	I/O 分配	162
E-16	原理图	166
附录 F	参考文献	176
附录 G	μ C/OS-III 许可政策	177

第 1 章

简 介

本书是《嵌入式实时操作系统 $\mu\text{C}/\text{OS} - \text{III}$ 》的第二部分,通过使用国际流行的开放工具,一步步指导读者学习。这里你还会看到 $\mu\text{C}/\text{Eval} - \text{STM32F107}$ 评估板上的例子,这块板子包含了 STMicroelectronics(简称 ST,中文是意法半导体)STM32F107 可联网微控制器。

为了能够构建本书提供的代码和应用,读者需要从 IAR 网站下载一份 IAR Embedded Workench kickstart(初学者)版本,它允许构建一个代码最大是 32 KB 的应用(见本书第 3 章:准备和设置)。同时,还可以从 Micrium 网站下载一份我们获得大奖的 $\mu\text{C}/\text{Probe}$,它可以让你在代码运行的时候监视和改变变量(见本书的第 3 章)。

STM32F107 的核心是 ARM Cortex - M3 CPU,是目前市场上流行的 CPU 核之一,运行非常高效的 ARM v7 指令集。STM32F107 运行的时钟频率最高可到达 72 MHz,包含了高性能外设,如 10/100M 以太网 MAC、全速的 USB OTG(On - The - Go)、CAN 控制器、TIMER 和 UART 等。STM32F107 还有内置的 256 KB Flash 和 64 KB 高速静态 RAM。

1.1 $\mu\text{C}/\text{Eval} - \text{STM32F107}$ 评估板

$\mu\text{C}/\text{Eval} - \text{STM32F107}$ 评估板的结构图如图 1-1 所示,主要的特性如下:

- 低成本
- 基于 STM32F107 的微控制器
- 以太网控制权

- 板上的 J-Tag 调试
- SD 卡插座
- USB-OTG 连接器
- RS-232C 母连接器
- 温度传感器
- LED
- 扩展连接器
- 原型机区域

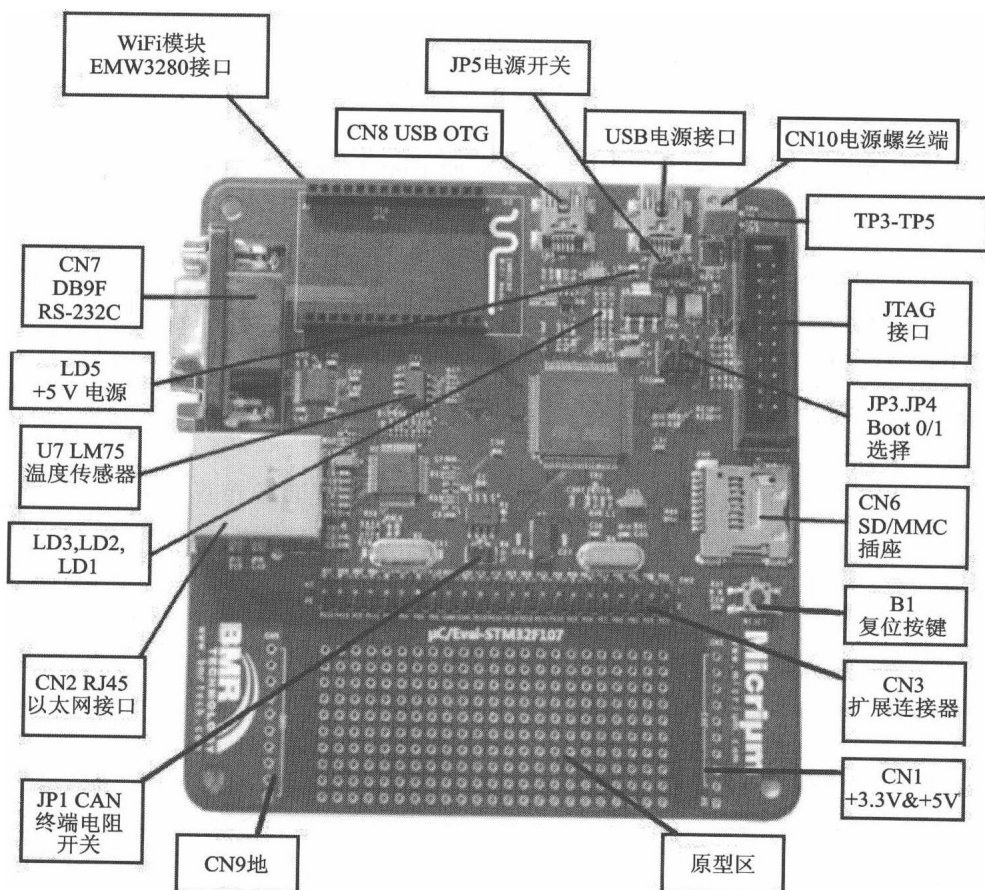


图 1-1 $\mu\text{C}/\text{Eval} - \text{STM32F107}$ 评估板

【译者注】 中国版的 $\mu\text{C}/\text{Eval} - \text{STM32F107}$ 评估板只有 JTAG 连接线接口, 原版是板上内置 SWB 接口。

STM32107 能够运行一个完整的 TCP/IP 协议, 比如 Micrium $\mu\text{C}/\text{TCP} - \text{IP}$, 这是一个商用的高质量协议软件。这样应用可以与其他设备联网, 同样还可以接入互联网。 $\mu\text{C}/\text{TCP} - \text{IP}$ 已经在 $\mu\text{C}/\text{Eval} - \text{STM32F107}$ 评估板进行了测试, 实测的数据是: TCP 的传输率, 接收 25 Mbps, 发送 30 Mbps; UDP 的传输率, 接收 50 Mbps, 发送 50 Mbps。 $\mu\text{C}/\text{TCP} - \text{IP}$ 应用层协议包括: DHCP client (功能是获得一个 IP 地址)、FTP client/server、HTTP server (比如一个 web server)、SMTP client (功能是发送 E-mails) 和 POP3 client (功能是接收 E-mails) 等。

STM32F107 还可以运行 USB On - The - Go 协议, 比如 Micrium $\mu\text{C}/\text{USB} - \text{Devices}$, $\mu\text{C}/\text{USB} - \text{HOST}$, $\mu\text{C}/\text{USB} - \text{OTG}$ 。STM32F107 USB 控制器是一个全速的设备, 传输速率可达到 12 Mbps。 $\mu\text{C}/\text{Eval} - \text{STM32F107}$ 评估板既可以用作 USB device 还可以用作 USB host。做为 USB device, 评估板可以是人机接口设备 (HID), 或者主存设备 (MSD)。尤其是加上板上的 SD 卡, $\mu\text{C}/\text{Eval} - \text{STM32F107}$ 评估板可以作为 PC 的一个磁盘外设。作为 USB Host, $\mu\text{C}/\text{Eval} - \text{STM32F107}$ 评估板支持 U 盘的数据储存和恢复。

板上的 SD 连接器, 让用户可以运行文件系统, 比如 $\mu\text{C}/\text{FS}$, 用以在 SD 上保存内容和读取 SD 上的内容。SD 卡还可以用来保存数据记录, 这些记录可以通过 PC 的 USB device MSD 类, 或者 FTP Client (FTP 用户端) 访问。

RS - 232C 连接器可以让一个应用输出信息到一个终端上 (或者终端仿真)。RS - 232C 接口还可以支持 $\mu\text{C}/\text{Probe}$, 比板子的 J - link 连接更快地传输数据。当然这个时候, 板子上的目标代码处于运行状态。

附录 E 包含了 $\mu\text{C}/\text{Eval} - \text{STM32F107}$ 评估板完整的电路图。

1.2 本书的章节内容

图 1 - 2 所示为本书的布局和流程, 这对于理解本书的各章节和附录之间的关系非常有帮助。左边第一列是 $\mu\text{C}/\text{OS} - \text{III}$ 结构和例子, 中间是 $\mu\text{C}/\text{OS} - \text{III}$

移植到 ARM Cortex - M3 CPU 的信息部分,右边是各种附录。

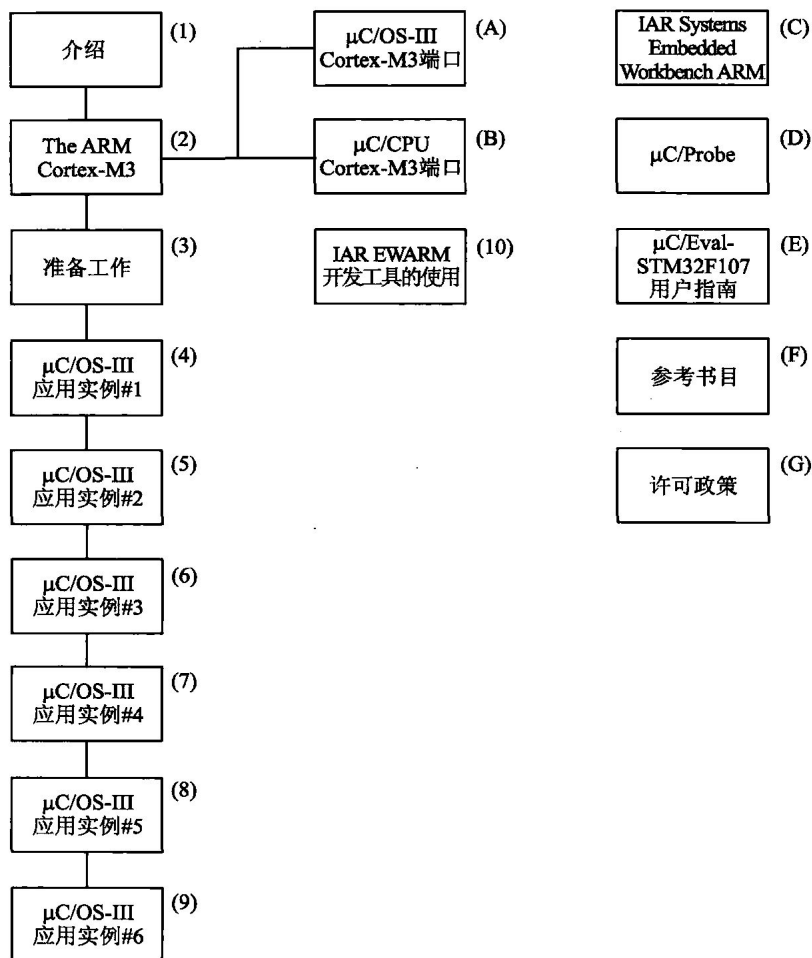


图 1-2 本书内容布局安排

第 1 章:介绍本章。

第 2 章:简要介绍 ARM Cortex - M3 CPU。

第 3 章:准备工作。本章叙述了从准备测试环境到运行 $\mu\text{C}/\text{OS} - \text{III}$ 例子的过程。叙述了下载一个 32 KB Kickstart 版本 IAR Embedded Workbench ARM 工具链,如何得到本书附带的例子,以及如何将 PC 与 $\mu\text{C}/\text{Eval} - \text{STM32F107}$ 评估板连接起来的过程。

第 4 章:应用实例 1。本章叙述了如何让 $\mu\text{C}/\text{OS} - \text{III}$ 启动和运行起来。例

子简单地让 $\mu\text{C}/\text{Eval} - \text{STM32F107}$ 评估板的 LED 灯闪动, 还可以看到, 使用 $\mu\text{C}/\text{Probe}$ 观察目标板的数据变化是很方便和容易的。

第 5 章: 应用实例 2。本章叙述了如何读板上的 LM75 温度传感器, 并且使用 $\mu\text{C}/\text{Probe}$ 显示当前的数据。

第 6 章: 应用实例 3。本章叙述了如何测量选定的 $\mu\text{C}/\text{OS} - \text{III}$ 性能参数, 并使用 $\mu\text{C}/\text{Probe}$ 观察实时信息。

第 7 章: 应用实例 4。本章叙述了如何使用 $\mu\text{C}/\text{OS} - \text{III}$ 仿真测量一个旋转轮的 RPM。

第 8 章: 应用实例 5。本章叙述了 $\mu\text{C}/\text{FS}$ 文件系统的结构和驱动, 以及如何在 $\mu\text{C}/\text{OS} - \text{III}$ 环境下构建支持 SD 卡读写的 $\mu\text{C}/\text{FS}$ 应用。

第 9 章: 应用实例 6。本章叙述了在 $\mu\text{C}/\text{OS} - \text{III}$ 环境下, 配合 EMW WiFi 模块实现 WiFi 应用。

第 10 章: IAR EWARM 开发工具的使用。本章详细介绍了如何使用 EWARM 集成开发环境调试应用代码。

附录 A: $\mu\text{C}/\text{OS} - \text{III}$ 移植到 Cortex - M3。本附录解释了 $\mu\text{C}/\text{OS} - \text{III}$ 如何适配到 Cortex - M3 CPU 上。Cortex - M3 包含了针对实时内核非常有意义的特性, $\mu\text{C}/\text{OS} - \text{III}$ 非常很好地使用了这些特性。

附录 B: $\mu\text{C}/\text{CPU}$ 移植到 Cortex - M3。本附录叙述了 $\mu\text{C}/\text{CPU}$ 如何适配到 Cortex - M3 CPU 上。

附录 C: 本附录提供了 IAR Embedded Workbench ARM 的简单介绍。

附录 D: 本附录提供了 Micrium 获得大奖的 $\mu\text{C}/\text{Probe}$ 产品的简单介绍, 它让用户可以很方便地修改和显示目标系统变量的实时变化。

附录 E: $\mu\text{C}/\text{Eval} - \text{STM32F107}$ 评估板用户指南。本附录提供了 $\mu\text{C}/\text{Eval} - \text{STM32F107}$ 评估板的简单介绍, 还有一份完整的电子线路图。

附录 F: 参考书目。

附录 G: 授权政策。

1.3 作者致谢

我要感谢 ST 公司 Mr. Dominique Jugnon 和 Mr. Olivier Brun。他们的优秀团队设计了 $\mu\text{C}/\text{Eval} - \text{STM32F107}$ 评估板,并撰写了用户指南。

我要感谢 IAR 公司的支持,他们让我们可以使用 32 KB Kickstart 版本的 IAR Embedded Workbench ARM (EWARM)。我非常肯定的是,EWARM 是一个很酷的工具,读者一定会很喜欢在 $\mu\text{C}/\text{Eval} - \text{STM32F107}$ 评估板上使用 EWARM 运行 $\mu\text{C}/\text{OS} - \text{III}$ 。

特别感谢 Mr. Rolf Segger,他提供了板上的 J-link,使得调试和通过 $\mu\text{C}/\text{Probe}$ 访问变量变得轻而易举。

还有感谢 Hitex 提供第 2 章 ARM Cortex - M3 和 STM32 大部分文本。

最后,我还要感谢 Micrium 公司,感谢这个伟大的团队对这个项目的支持和帮助。

第 2 章

ARM Cortex - M3 和 STM32

ARM Cortex - M3 系列是新一代处理器,提供一个适应广泛技术需求的、标准的架构。与其他的 ARM CPU 不一样,Cortex 是一个完整处理器核,提供标准的 CPU 和系统架构,面向低成本和微控制器(MCU)应用。

本章提供一个简要介绍,其他的参考资料见参考书目部分。

ARM7 和 ARM9 已经成功集成到微控制器里面,这显示出它们成功的 SoC 遗传基因。每个特定的芯片制造商都会设计自己的中断处理。然而,Cortex - M3 提供一个标准的微控制器核,其功能远远超过一般的 CPU,有更加完整的微处理器核心(包括中断、系统时钟节拍、调试和内存映像)。Cortex - M3 4 GB 的地址空间分成代码、SRAM、外设和系统外设 4 个部分。与 ARM7 不同,Cortex - M3 是哈佛架构,具有多个总线,允许执行并行操作,提高整体性能。不同于早期的 ARM 架构,Cortex 家族允许未对齐的数据访问,这保证了对内部 SRAM 更高效率的使用。Cortex 家族还支持使用一种称为位绑定(bit banding)的方法,在 2 个 1 MB 的存储器里对位进行设置和清除操作。这样的好处是可以更有效地访问位于 SDRAM 存储器的外设寄存器和标识,而不需要一个完整的布尔处理器。

Cortex - M3 内核的一个关键部件是嵌套向量中断控制器(NVIC)。NVIC 为所有的 Cortex 微控制器提供一个标准的中断架构和异常中断处理。NVIC 可以为最多 240 个外设源提供唯一的中断向量,这样每个外设源都可以单独优先。在背对背(back - to - back)中断情况下,NVIC 使用尾链(Tail chaining)的方法让连续中断服务的开销降至最小。在中断堆栈的阶段,高优先级中断可以抢占低优先级中断,不需要额外的 CPU 时钟周期。在 Cortex - M3 核心里,这样的中断结构与低功耗模式是耦合在一起的。

尽管 Cortex - M3 是按照一款低功耗内核设计的,它依然是一个 32 位的