从0到1教你写FreeRTOS内核，详解FreeRTOS源码如何使用。
由浅入深，基于野火STM32全系列开发板，提供完整的源代码，极具可操作性。

华章科技

野火

野火嵌入式系列

RTOS

FreeRTOS内核实现与应用开发实战指南

基于STM32

刘火良 杨森 编著

机械工业出版社
China Machine Press

野火

野火嵌入式系列

RTOS

FreeRTOS内核实现与应用开发实战指南

基于STM32

刘火良 杨森 编著

机械工业出版社
China Machine Press

图书在版编目（CIP）数据

FreeRTOS 内核实现与应用开发实战指南：基于 STM32 / 刘火良，杨森编著 . —北京：机械工业出版社，2019.2
（电子与嵌入式系统设计丛书）

ISBN 978-7-111-61825-6

I. F… II. ①刘… ②杨… III. 微控制器 – 系统开发 – 指南 IV. TP332.3-62

中国版本图书馆 CIP 数据核字（2019）第 010054 号

FreeRTOS 内核实现与应用开发实战指南
基于 STM32

出版发行：机械工业出版社（北京市西城区百万庄大街 22 号　邮政编码：100037）
责任编辑：赵亮宇　　责任校对：李秋荣
印　　刷：北京诚信伟业印刷有限公司　　版　　次：2019 年 3 月第 1 版第 1 次印刷
开　　本：186mm × 240mm　1/16　　印　　张：31
书　　号：ISBN 978-7-111-61825-6　　定　　价：99.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换
客服热线：（010）88379426　88361066　　投稿热线：（010）88379604
购书热线：（010）68326294　88379649　68995259　　读者信箱：hzit@hzbook.com

前　言

如何学习本书

本书系统讲解 FreeRTOS，共分为两个部分。第一部分重点讲解 FreeRTOS 的原理实现，从 0 开始，不断迭代，教你把 FreeRTOS 的内核写出来，让你彻底学会任务是如何定义的、系统是如何调度的（包括底层的汇编代码讲解）、多优先级是如何实现的等操作系统最深层次的知识。当你拿到本书开始学习的时候，你一定会惊讶，原来 RTOS（Real Time Operating System，实时操作系统）的学习并没有那么复杂，反而是那么有趣，原来自己也可以写 RTOS，成就感瞬间爆棚。

当彻底掌握第一部分的知识之后，再学习其他 RTOS，可以说十分轻松。纵观现在市面上流行的几种 RTOS，它们的内核实现差异不大，只需要深入研究其中一种即可，没有必要对每一种 RTOS 都深入地研究源码。但如果时间允许，看一看也并无坏处。第二部分重点讲解 FreeRTOS 的移植、内核中每个组件的应用，比起第一部分，这部分内容掌握起来应该比较容易。

全书内容循序渐进，不断迭代，尤其在第一部分，前一章多是后一章的基础，建议从头开始阅读，不要进行跳跃式阅读。在学习时务必做到两点：一是不能一味地看书，要把代码和书本结合起来学习，一边看书，一边调试代码。如何调试代码呢？即单步执行每一条程序，看程序的执行流程和执行的效果与自己所想的是否一致。二是在每学完一章之后，必须将配套的例程重写一遍（切记不要复制，哪怕是一个分号。但可以照书录入），做到举一反三，确保真正理解。在自己写的时候肯定会错漏百出，要认真纠错，好好调试，这是你提高编程能力的最好机会。记住，编写程序不是一气呵成的，而是要一步一步地调试。

本书的编写风格

本书第一部分以 FreeRTOS V9.0.0 官方源码为蓝本，抽丝剥茧，不断迭代，教你如何从 0 开始把 FreeRTOS 内核写出来。书中涉及的数据类型、变量名称、函数名称、文件名称、

文件存放的位置都完全按照 FreeRTOS 官方的方式来实现。学完这本书之后，你可以无缝地切换到原版的 FreeRTOS 中使用。要注意的是，在实现的过程中，某些函数中会去掉一些形参和冗余的代码，只保留核心的功能，但这并不会影响学习。

本书第二部分主要介绍 FreeRTOS 的移植和内核组件的使用，不会再去深入讲解源码，而是着重讲解如何应用，如果对第一部分不感兴趣，可以跳过第一部分，直接进入第二部分的学习。

本书还有姊妹篇——《RT-Thread 内核实现与应用开发实战指南：基于 STM32》[⊖]，两本书的编写风格、内容框架和章节命名与排序基本一致，语言阐述类似，且涉及 RTOS 抽象层的理论部分也相同，不同之处在于 RTOS 的实现原理、内核源码的讲解和上层 API 的使用，这些内容才是重点部分，是读者学习的核心。例如，虽然两本书的第一部分的章节名称基本类似，但内容不同，因为针对的 RTOS 不一样。其中，关于新建 FreeRTOS 工程和裸机系统与多任务（线程）系统的描述属于 RTOS 抽象层的理论部分，不具体针对某个 RTOS，所以基本一样。第二部分中，对于什么是任务（线程）、阻塞延时和信号量的应用等 RTOS 抽象层的理论讲解也基本类似，但是具体涉及这两个 RTOS 的原理实现和代码讲解时则完全不同。

如果读者已经学习了其中一本书，再学习另外一本的话，那么涉及 RTOS 抽象层的理论部分可跳过，只需要把精力放在 RTOS 内核的实现和源码 API 的应用方面。因为现有的 RTOS 在理论层基本都是相通的，但在具体的代码实现上各有特点，所以可以用这两本书进行互补学习，掌握了其中一本书的知识，再学习另外一本书定会得心应手，事半功倍。

本书的参考资料及配套硬件

关于本书的参考资料和配套硬件的信息，请参考本书附录部分。

本书的技术论坛

如果在学习过程中遇到问题，可以到野火电子论坛 www.firebbs.cn 发帖交流，开源共享，共同进步。

鉴于作者水平有限，书中难免有错漏之处，热心的读者也可把勘误发送到论坛上以便改进。祝你学习愉快，FreeRTOS 的世界，野火与你同行。

⊖ 此书由机械工业出版社出版，书号为 978-7-111-61366-4。——编辑注

引　言

为什么要学习 RTOS

当我们进入嵌入式这个领域时，首先接触的往往是单片机编程，单片机编程又首选 51 单片机来入门。这里面说的单片机编程通常都是指裸机编程，即不加入任何 RTOS 的编程。常用的 RTOS 有国外的 FreeRTOS、μC/OS、RTX 和国内的 FreeRTOS、Huawei LiteOS 和 AliOS-Things 等，其中，开源且免费的 FreeRTOS 的市场占有率最高。

在裸机系统中，所有的程序基本都是自己写的，所有的操作都是在一个无限的大循环中实现。现实生活中的很多中小型电子产品中用的都是裸机系统，而且能够满足需求。但是为什么还要学习 RTOS 编程，要涉及一个操作系统呢？一是因为项目需求，随着产品要实现的功能越来越多，单纯的裸机系统已经不能完美地解决问题，反而会使编程变得更加复杂，如果想降低编程的难度，可以考虑引入 RTOS 实现多任务管理，这是使用 RTOS 的最大优势；二是出于学习的需要，必须学习更高级的技术，实现更好的职业规划，为将来能有更好的职业发展做准备，而不是一味拘泥于裸机编程。作为一个合格的嵌入式软件工程师，学习是永远不能停歇的，时刻都得为将来做准备。书到用时方恨少，希望当机会来临时，你不要有这种感觉。

为了帮大家厘清 RTOS 编程的思路，本书会在第 3 章简单地分析这两种编程方式的区别，我们将这个区别称为“学习 RTOS 的命门”，只要掌握这一关键内容，以后的 RTOS 学习可以说是易如反掌。在讲解这两种编程方式的区别时，我们主要讲解方法论，不会涉及具体的代码，即主要还是通过伪代码来讲解。

如何学习 RTOS

裸机编程和 RTOS 编程的风格有些不一样，而且有很多人说学习 RTOS 很难，这就导致想要学习的人一听到 RTOS 编程就在心里忌惮三分，结果就是“出师未捷身先死”。

那么到底如何学习 RTOS 呢？最简单的方法就是在别人移植好的系统上，先看看 RTOS

中 API 的使用说明，然后调用这些 API 实现自己想要的功能，完全不用关心底层的移植，这是最简单、快速的入门方法。这种方法有利有弊。如果是做产品，好处是可以快速地实现功能，将产品推向市场，赢得先机；弊端是当程序出现问题时，因对 RTOS 不够了解，会导致调试困难。如果想系统地学习 RTOS，那么只会简单地调用 API 是不可取的，我们应该深入学习其中一款 RTOS。

目前市场上的 RTOS，其内核实现方式差异不大，我们只需要深入学习其中一款即可。万变不离其宗，只要掌握了一款 RTOS，以后换到其他型号的 RTOS，使用起来自然也是得心应手。那么如何深入地学习一款 RTOS 呢？这里有一个非常有效但也十分难的方法，就是阅读 RTOS 的源码，深入研究内核和每个组件的实现方式。这个过程枯燥且痛苦。但为了能够学到 RTOS 的精华，还是很值得一试的。

市面上虽然有一些讲解相关 RTOS 源码的图书，但如果基础知识掌握得不够，且先前没有使用过该款 RTOS，那么只看源码还是会非常枯燥，并且不能从全局掌握整个 RTOS 的构成和实现。

现在，我们采用一种全新的方法来教大家学习一款 RTOS，既不是单纯地介绍其中的 API 如何使用，也不是单纯地拿里面的源码一句句地讲解，而是从 0 开始，层层叠加，不断完善，教大家如何把一个 RTOS 从 0 到 1 写出来，让你在每一个阶段都能享受到成功的喜悦。在这个 RTOS 实现的过程中，只需要具备 C 语言基础即可，然后就是跟着本书笃定前行，最后定有所成。

选择什么 RTOS

用来教学的 RTOS，我们不会完全从头写一个，而是选取目前国内外市场占有率很高的 FreeRTOS 为蓝本，将其抽丝剥茧，从 0 到 1 写出来。在实现的过程中，数据类型、变量名、函数名称、文件类型等都完全按照 FreeRTOS 里面的写法，不会再重新命名。这样学完本书之后，就可以无缝地过渡到 FreeRTOS 了。

目　录

第一部分

从 0 到 1 教你写 FreeRTOS 内核

本部分以 FreeRTOS Nano 为蓝本，抽丝剥茧，不断迭代，教大家如何从 0 开始把 FreeRTOS 写出来。这一部分着重讲解 FreeRTOS 实现的过程，当你学完这部分之后，再来重新使用 FreeRTOS 或者其他 RTOS，将会得心应手，不仅知其然，而且知其所以然。在源码实现的过程中，涉及的数据类型、变量名称、函数名称、文件名称以及文件的存放目录都会完全按照 FreeRTOS 的来实现，一些不必要的代码将会剔除，但并不会影响我们理解整个操作系统的功能。

本部分几乎每一章都是前一章的基础，环环相扣，逐渐揭开 FreeRTOS 的神秘面纱，读起来会有一种豁然开朗的感觉。如果把代码都敲一遍，仿真时得出的效果与书中给出的一样，那从心里油然而生的成就感简直就要爆棚，恨不得一下子把本书读完，真是让人看了还想看，读了还想读。

第 1 章
初识 FreeRTOS

1.1 FreeRTOS 版权

FreeRTOS 由美国的 Richard Barry 于 2003 年发布，Richard Barry 是 FreeRTOS 的拥有者和维护者，在过去的十多年中 FreeRTOS 历经了 9 个版本，与众多半导体厂商合作密切，有数百万开发者，是目前市场占有率最高的 RTOS。

FreeRTOS 于 2018 年被亚马逊收购，改名为 AWS FreeRTOS，版本号升级为 V10，且开源协议也由原来的 GPLv2+ 修改为 MIT，与 GPLv2+ 相比，MIT 更加开放，你完全可以理解为完全免费。V9 以前的版本还是维持原样，V10 版本相比于 V9 就是加入了一些物联网相关的组件，内核基本不变。亚马逊收购 FreeRTOS 也是为了进军物联网和人工智能领域。本书还是以 V9 版本来讲解。

1.2 FreeRTOS 收费问题

1.2.1 FreeRTOS

FreeRTOS 是一款“开源免费”的实时操作系统，遵循的是 GPLv2+ 的许可协议。这里说到的开源，指的是可以免费获取 FreeRTOS 的源代码，且当你的产品使用了 FreeRTOS 而没有修改 FreeRTOS 内核源码时，你的产品的全部代码都可以闭源，不用开源，但是当你修改了 FreeRTOS 内核源码时，就必须将修改的这部分开源，反馈给社区，其他应用部分不用开源。免费的意思是无论你是个人还是公司，都可以免费地使用，不需要花费一分钱。

1.2.2 OpenRTOS

FreeRTOS 和 OpenRTOS 拥有的代码是一样的，但是可从官方获取的服务却是不一样的。FreeRTOS 号称免费，OpenRTOS 号称收费，它们的具体区别如表 1-1 所示。

表 1-1　FreeRTOS 开源授权与 OpenRTOS 商业授权的区别

比较的项目	FreeRTOS	OpenRTOS
是否免费	是	否
可否商业使用	是	是
是否需要版权费	否	否
是否提供技术支持	否	是
是否被法律保护	否	是
是否需要开源工程代码	否	否
是否需要开源修改的内核源码	是	否
是否需要声明产品使用了 FreeRTOS	如果发布源码，则需要声明	否
是否需要提供 FreeRTOS 的整个工程代码	如果发布源码，则需要提供	否

1.2.3　SaveRTOS

SaveRTOS 也基于 FreeRTOS，但是 SaveRTOS 为某些特定的领域做了安全相关的设计，有关 SaveRTOS 获得的安全验证具体如表 1-2 所示。SaveRTOS 需要收费。

表 1-2　SaveRTOS 获得的安全方面的验证

行 业 类 目	验 证 编 号
工控	IEC 61508
铁路	EN 50128
医疗	IEC 62304/FDA 510K
核工业	IEC 61513、IEC62138、ASME NQA-1
汽车电子	ISO 26262
加工业	IEC 61511
航空	DO178B

1.3　FreeRTOS 资料获取

FreeRTOS 的源码和相应的官方书籍均可从官网 www.freertos.org 获得，如图 1-1 所示为官网首页。

1.3.1　获取源码

单击图 1-1 中的 Download Source 按钮，可以下载 FreeRTOS 最新版本的源码。如果想下载以往版本，可从托管网址 https://sourceforge.net/projects/freertos/files/FreeRTOS/ 下载。截至本章编写时，FreeRTOS 已经更新到 V10.0.1，具体如图 1-2 所示。