



Making Embedded Systems

嵌入式系统设计领域的经典之作，资深嵌入式系统设计专家撰写
系统总结嵌入式系统设计的通用原型和原则



嵌入式系统 设计与实践

O'REILLY®

机械工业出版社
China Machine Press

Elecia White 著
余水清 译

. 013044045

TP360.21
73

嵌入式系统设计与实践



Elecia White 著
余水清 译

TP360.21
73



O'REILLY®

Beijing • Cambridge • Farnham • Köln • Sebastopol • Tokyo

O'Reilly Media, Inc. 授权机械工业出版社出版

机械工业出版社

图书在版编目 (CIP) 数据

嵌入式系统设计与实践/ (美) 怀特 (White, E.) 著; 余水清译. —北京: 机械工业出版社, 2013.3

(O'Reilly精品图书系列)

书名原文: Making Embedded Systems

ISBN 978-7-111-41584-8

I. 嵌… II. ①怀… ②余… III. 微型计算机—系统设计 IV. TP360.21
中国版本图书馆CIP数据核字 (2013) 第034483号

北京市版权局著作权合同登记

图字: 01-2012-4850号

Copyright © 2012 Elecia White.

Simplified Chinese Edition, jointly published by O'Reilly Media, Inc. and China Machine Press, 2013. Authorized translation of the English edition, 2012 O'Reilly Media, Inc., the owner of all rights to publish and sell the same.

All rights reserved including the rights of reproduction in whole or in part in any form.

英文原版由O'Reilly Media, Inc. 出版2012。

简体中文版由机械工业出版社出版 2013。英文原版的翻译得到O'Reilly Media, Inc.的授权。此简体中文版的出版和销售得到出版权和销售权的所有者——O'Reilly Media, Inc.的许可。

版权所有, 未得书面许可, 本书的任何部分和全部不得以任何形式重制。

封底无防伪标均为盗版

本书法律顾问

北京市展达律师事务所

书 名/ 嵌入式系统设计与实践

书 号/ ISBN 978-7-111-41584-8

责任编辑/ 盛思源

封面设计/ Karen Montgomery, 张健

出版发行/ 机械工业出版社

地 址/ 北京市西城区百万庄大街22号 (邮政编码 100037)

印 刷/ 藁城市京瑞印刷有限公司

开 本/ 178毫米×233毫米 16开本 18.75印张

版 次/ 2013年5月第1次印刷

定 价/ 69.00元 (册)

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)88378991; 88361066

购书热线: (010)68326294; 88379649; 68995259

投稿热线: (010)88379604

读者信箱: hzjsj@hzbook.com

O'Reilly Media, Inc.介绍

O'Reilly Media通过图书、杂志、在线服务、调查研究和会议等方式传播创新知识。自1978年开始，O'Reilly一直都是前沿发展的见证者和推动者。超级极客们正在开创着未来，而我们关注真正重要的技术趋势——通过放大那些“细微的信号”来刺激社会对新科技的应用。作为技术社区中活跃的参与者，O'Reilly的发展充满了对创新的倡导、创造和发扬光大。

O'Reilly为软件开发人员带来革命性的“动物书”；创建第一个商业网站（GNN）；组织了影响深远的开放源代码峰会，以至于开源软件运动以此命名；创立了Make杂志，从而成为DIY革命的主要先锋；公司一如既往地通过多种形式缔结信息与人的纽带。O'Reilly的会议和峰会集聚了众多超级极客和高瞻远瞩的商业领袖，共同描绘出开创新产业的革命性思想。作为技术人士获取信息的选择，O'Reilly现在还将先锋专家的知识传递给普通的计算机用户。无论是通过书籍出版，在线服务或者面授课程，每一项O'Reilly的产品都反映了公司不可动摇的理念——信息是激发创新的力量。

业界评论

“O'Reilly Radar博客有口皆碑。”

——Wired

“O'Reilly凭借一系列（真希望当初我也想到了）非凡想法建立了数百万美元的业务。”

——Business 2.0

“O'Reilly Conference是聚集关键思想领袖的绝对典范。”

——CRN

“一本O'Reilly的书就代表一个有用、有前途、需要学习的主题。”

——Irish Times

“Tim是位特立独行的商人，他不光放眼于最长远、最广阔的视野并且切实地按照Yogi Berra的建议去做了：‘如果你在路上遇到岔路口，走小路（岔路）。’回顾过去Tim似乎每一次都选择了小路，而且有几次都是一闪即逝的机会，尽管大路也不错。”

——Linux Journal

译者序

谈到设计模式，不得不提“四人帮”所著的《设计模式：可复用面向对象软件的基础》，在那本经典著作中，作者总结了设计灵活、易扩展的面向对象软件的23种设计模式，使软件开发人员在面对类似应用场景、要解决类似问题的时候可以应用这些优秀的解决方案。

在此之后，“设计模式”称为软件开发领域的研究热点之一，诞生了一系列著作，涵盖了从需求到分析，再到设计、实现、测试，甚至软件界面设计等主题。然而，在嵌入式系统开发领域里是否可以应用设计模式，如果可以的话，又该如何应用？作为热衷软件开发以及模式应用的我来说，一直在思考和探索这个问题。直到看到White的这本书，这个问题才算有了一个比较完美的解答。

作者以高屋建瓴的方式，从嵌入式系统设计的重中之重——架构设计入手，介绍了嵌入式系统的架构设计步骤以及架构设计模式，在此基础上转向如何运行代码，如何与外设通信，如何进行代码更新，如何使用定时器和中断，以及如何在小型处理器上实现复杂的数学运算，同时也介绍了如何降低系统功耗的一些有效方法。对本书内容的组织，也是一个非常优秀的架构设计示例，体现了作者构思的巧妙。

本书中，作者不仅结合具体的例子讨论了如何在嵌入式软件中使用经典的设计模式，如门面模式、命令模式等，而且也深入讨论了在嵌入式系统中独有的设计模式，如环形缓冲区、中断和定时器、硬件调试、PID控制、系统优化等。

也许很多人会质疑，用C语言很难实现C++的面向对象设计方法。但在本书中，可以看到，用C语言的指针很容易就可以做到。而且，掌握了这点之后，就可以发现面向对象设计思想在嵌入式系统开发中也可以得心应手地运用，与语言没有关系。

从事过嵌入式开发的技术人员，对如何调试电路板可能都非常熟悉，这其中不仅仅是技术，更多的是要面对交付的压力。要学会如何沟通，遇到问题时如何寻求帮助。在本书中，作者给出了具体的例子，读完之后，就会发现其中某些部分也许就是自己开发过程中的某段过程的缩影，让人回味无穷。

本书主要介绍没有使用操作系统的嵌入式系统的开发，因此必然涉及实现一些简单操作系统的功能，比如任务调度、状态机等。本书也给出了非常简明的例子，讨论了状态机的不同实现机制以及在什么情况下使用中断，如何不使用中断实现相应的功能。

不夸张地说，软件模式的介绍只是本书的一小部分，但本书将模式的思想发挥得更加自如，任何部分都可以称为一个模式。贯穿这些软件设计、硬件设计、调试模式中的一个核心思想就是权衡，而这一点恰恰是最难掌握的设计艺术之一，值得用心去体会。

本书字字珠玑，行文生动有趣，每章还附上了一个面试问题，并且给出了回答这个面试问题的独到思路。这也是本书与众不同的地方，作者将面试也“模式”化了。同时这也体现了作者写作本书的目的之一，帮大家解决问题。

有些书教给人们某个工具的使用，有些书教给人们某个方法，而本书则既教给人们工具、方法，又启发大家如何去思考和权衡。通读本书，就像经历过一个完整的嵌入式开发项目，让人难以忘怀。

在本书的翻译过程，得到了机械工业出版社吴怡编辑的精心指导和帮助，华章公司的编辑们对技术的熟练掌握以及对语言的精准把握让人受益匪浅。

同时，感谢好友韩丽萍的热心帮助，感谢妻子和家人的宽容和耐心，你们的关心和鼓励让我可以安心地专注于技术、翻译和文字所带来的乐趣。

时间有限，错误之处在所难免。因此，怀着忐忑的心情，向广大读者呈上这本译著，希望大家不吝指正。

译者介绍

余水清 国家认证系统分析师，信息系统项目管理师，系统架构师，中国系统分析员协会（CSAI）专业顾问。曾参与过多项大型软件开发项目，担任软件设计师、系统架构设计师、项目经理等职位，涉及多个行业，包括银行、医疗、制造业、国家单位等。热衷于软件设计模式和架构模式的研究和应用。

作者介绍

Elecia White曾经设计过DNA扫描仪、飞机和赛车的惯性测量单元、学龄前儿童玩具、抓捕罪犯的射击定位系统以及其他种类繁多的医疗和消费类设备。她是Logical Elegance的创办人，这是一家位于圣何塞（San Jose）的嵌入式系统咨询公司。艾丽西亚在信号处理、硬件集成、复杂系统和性能设计等方面有着很强的背景和经验。在经过美国联邦航空局（FAA）和美国食品药品监督管理局（FDA）认证过程后，她理解了高质量设计的重要性，以及高质量的设计如何引领高质量的实现。

作者拥有多年的管理经验，但是她仍然很享受动手实践的工程设计工作以及交付优秀产品的兴奋和快感。在继续领导和指导工作的同时，她更倾向于专注在项目的技术方面。作为一名哈佛穆德学院（位于加利福尼亚州克莱尔蒙特市）的毕业生，艾丽西亚热衷于分享她对于科学、工程以及有趣小发明的激情，特别是如何利用这一切让世界变得更美好。

封面简介

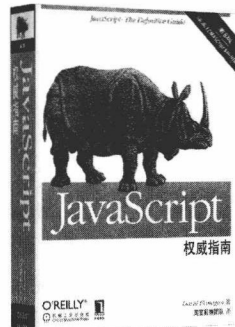
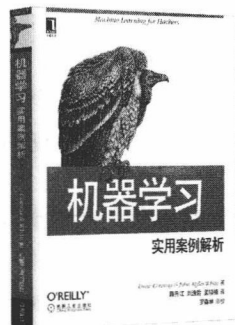
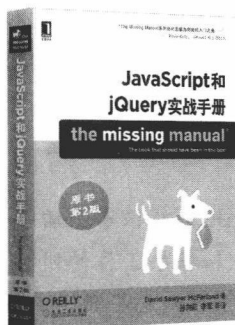
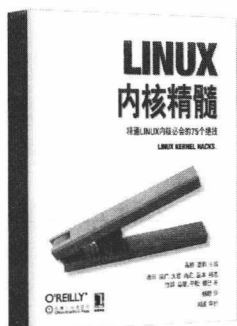
本书封面上的动物是一只大耳朵夜鹰。

大耳朵夜鹰是夜鹰科家族中的成员，俗称欧夜鹰。夜鹰的称谓来自于一段不那么确切的传说——这种鸟喝山羊的奶。这不应该和“卓柏卡布拉”（Chupacabra，一种传说中存在于美洲专门吸食山羊血的野兽）混为一谈。因此，现在人们通常叫它“大耳欧夜鹰”。

大耳欧夜鹰生活在东南亚的亚热带和热带潮湿的低湿林地区，包括印度、孟加拉国、缅甸、泰国、马来西亚、印度尼西亚、老挝、柬埔寨、越南、菲律宾以及中国。它们喜欢在光线昏暗的时候活动，通常是在黄昏或者晚上。这时，它们就会捕食一些小昆虫和蛾子，同时发出自己独特的叫声：一声尖锐的“啾”，跟着两个节拍的“吧呢”。

大耳夜鹰以突出的耳毛著称，相对于夜鹰平均16英寸的长度来说，这让它们在整个夜鹰家族中更加与众不同。它们灰褐相间的羽毛与其偏爱的栖息场所、落叶和欧洲蕨十分相似。它们常常把蛋下在光秃秃的地上或者由落叶做成的窝里。据观察，欧夜鹰的雏鸟非常安静，这让它们在落叶色的羽毛之外又多了一层保护，可以在哺乳的时候避免危险。

推荐阅读



- **Linux内核精髓：精通Linux内核必会的75个绝技**
作者：Hirokazu Takahashi 等
ISBN: 978-7-111-41049-2
定价：79.00元
- **JavaScript和jQuery实战手册（原书第2版）**
作者：David Sawyer McFarland
ISBN: 978-7-111-39903-2
定价：99.00元
- **机器学习：实用案例战**
作者：Drew Conway 等
ISBN: 978-7-111-41731-6
定价：69.00元
- **iOS 程序设计（原书第2版）**
作者：Matt Neuburg
ISBN: 978-7-111-41847-4
定价：89.00元
- **社交网站的数据挖掘与分析**
作者：Matthew A. Russell
ISBN: 978-7-111-36960-8
定价：59.00元
- **JavaScript权威指南（原书第6版）**
作者：David Flanagan
ISBN: 978-7-111-37661-3
定价：139.00元



北航

C1647094

目录

前言	1
第1章 导论	7
1.1 编译器、编程语言以及面向对象编程	7
1.2 嵌入式系统开发	8
1.2.1 调试	8
1.2.2 更多挑战	10
1.2.3 解决问题的原则	11
1.3 延伸阅读	12
第2章 创建系统架构	15
2.1 构建系统框图	16
2.1.1 架构框图	16
2.1.2 控制层级图	18
2.1.3 层次图	20
2.2 从框图到架构	21
2.2.1 封装模块	22
2.2.2 分派任务	22
2.2.3 驱动程序接口：打开（Open）、关闭（Close）、 读（Read）、写（Write）和输入输出控制（IOCTL）	23
2.2.4 适配器模式	24
2.2.5 开始设计其他接口	26
2.2.6 例子：一个日志接口	27

2.3 一个可以实践的沙盒	33
2.4 延伸阅读	37
第3章 检视硬件	39
3.1 硬件/软件集成	39
3.1.1 理想的项目流程	40
3.1.2 电路板调试	41
3.2 阅读数据表	42
3.2.1 遇到问题时需要查找的数据表部分	44
3.2.2 对软件开发者比较重要的内容	46
3.2.3 用数据表评估元器件	48
3.3 处理器是一门语言	51
3.4 阅读原理图	53
3.5 调试工具箱（和灭火器）	56
3.5.1 保管好电路板	56
3.5.2 工具箱	57
3.5.3 数字万用表	58
3.5.4 示波器和逻辑分析仪	58
3.6 测试硬件（和软件）	61
3.6.1 构建测试	62
3.6.2 闪存测试范例	63
3.6.3 命令和响应	66
3.6.4 命令模式	70
3.7 处理错误	71
3.7.1 一致的方法	72
3.7.2 错误处理库	73
3.7.3 调试时序错误	73
3.8 延伸阅读	74
第4章 输入、输出和定时器	76
4.1 触发输出	76
4.1.1 从寄存器开始	77
4.1.2 将引脚设置为输出	78

4.1.3 点亮LED.....	80
4.1.4 LED闪烁.....	81
4.1.5 故障诊断.....	81
4.2 将硬件和动作分离.....	82
4.2.1 特定于电路板的头文件.....	83
4.2.2 I/O处理代码.....	84
4.2.3 主循环.....	86
4.2.4 门面模式.....	86
4.3 I/O中的输入.....	87
4.3.1 按钮的简单接口.....	88
4.4 瞬间按钮动作.....	90
4.4.1 按钮按下时的中断.....	90
4.4.2 配置中断.....	91
4.4.3 去抖动开关.....	92
4.5 运行时不确定性.....	94
4.5.1 依赖注入.....	95
4.6 使用定时器.....	96
4.6.1 时间片.....	97
4.6.2 数学计算.....	99
4.6.3 定时器节拍之间的漫长等待.....	103
4.6.4 使用定时器.....	104
4.7 使用脉冲宽度调制.....	104
4.8 产品发货.....	106
4.9 延伸阅读.....	108
第5章 任务管理.....	109
5.1 调度和操作系统基础.....	109
5.1.1 任务.....	109
5.1.2 任务间通信.....	110
5.1.3 避免竞争条件.....	110
5.1.4 优先级倒置.....	112
5.2 状态机.....	113

5.2.1 状态机范例：信号灯控制器	114
5.2.2 以状态为中心的状态机	114
5.2.3 具有隐式迁移的以状态为中心的状态机	115
5.2.4 以事件为中心的状态机	116
5.2.5 状态模式	117
5.2.6 表格驱动的状态机	118
5.2.7 选择状态机的实现	120
5.3 中断	120
5.3.1 中断请求	121
5.3.2 保存上下文	128
5.3.3 从向量表中获得ISR	130
5.3.4 调用ISR	132
5.3.5 恢复上下文	134
5.3.6 何时使用中断	135
5.4 怎样不使用中断	135
5.4.1 轮询	136
5.4.2 系统节拍	136
5.4.3 基于时间的事件	138
5.4.4 一个极小型调度器	138
5.5 监控程序（看门狗）	140
5.6 延伸阅读	142

第6章 外设通信 145

6.1 应用广泛的外设	145
6.1.1 外部存储器	145
6.1.2 按钮和按键矩阵	146
6.1.3 传感器	147
6.1.4 执行器	150
6.1.5 显示器	155
6.2 种类繁多的通信方式	160
6.2.1 串行通信	161
6.2.2 并行	170

6.2.3 以太网和WiFi	171
6.3 将外设和通信结合	172
6.3.1 数据处理	172
6.3.2 增强通信的健壮性	182
6.3.3 修改数据	185
6.3.4 修改算法	187
6.4 延伸阅读	188
第7章 更新代码	191
7.1 板载启动装载器	192
7.2 构建自己的更新程序	193
7.2.1 修改驻留的更新程序	194
7.3 砖块装载器	195
7.3.1 将装载器复制到内存	196
7.3.2 运行装载器	198
7.3.3 复制新代码到暂存空间	198
7.3.4 危险时刻：擦除和编程	199
7.3.5 重置到新代码	199
7.4 安全	200
7.5 链接器脚本	201
7.6 总结	204
第8章 事半功倍	206
8.1 代码空间	207
8.1.1 读映射文件（第1部分）	207
8.1.2 简化的过程	210
8.1.3 库	212
8.1.4 函数和宏	212
8.1.5 常量和字符串	214
8.2 RAM	214
8.2.1 释放和分配	215
8.2.2 读映射文件（第2部分）	216

8.2.3 寄存器和局部变量	217
8.2.4 函数链	219
8.2.5 全局变量的优点和缺点	221
8.2.6 内存覆盖	221
8.3 速度	222
8.3.1 程序性能分析	223
8.3.2 优化	227
8.4 总结	236
8.5 延伸阅读	237
第9章 数学运算	240
9.1 区别快、慢运算	241
9.1.1 求平均值	242
9.2 使用已知的算法	245
9.3 设计与修改算法	248
9.3.1 多项式因式分解	248
9.3.2 泰勒级数	248
9.3.3 除以常数	250
9.3.4 放大输入	251
9.3.5 查找表	253
9.4 伪浮点数	259
9.4.1 有理数	260
9.4.2 精度	261
9.4.3 加法（和减法）	262
9.4.4 乘法（和除法）	263
9.4.5 决定误差	264
9.5 延伸阅读	268
第10章 降低功耗	270
10.1 理解功耗	271
10.2 离开房间的时候请关灯	273
10.2.1 关掉外围设备	273

10.2.2 关掉不用的I/O设备	274
10.2.3 关掉处理器子系统.....	274
10.2.4 降速节能.....	274
10.3 让处理器休眠	276
10.3.1 基于中断的代码流模型	277
10.3.2 深入研究主循环	279
10.3.3 处理器看门狗	280
10.3.4 避免频繁唤醒	280
10.3.5 处理器链.....	281
10.4 延伸阅读.....	281

前言

我热爱嵌入式系统。当我第一次让马达转动时，我着迷了。我很快从单纯软件开发进入到我可以触摸真实物体的一个领域。在我离开软件开发的时候，设计模式^{注1}开创性的工作已经完成。我和我的团队通读了这本书，讨论这些模式以及可以在哪儿应用这些模式。随着我进一步深入研究嵌入式系统，我发现不能处理C++继承的编译器，在处理器极其有限的内存中去实现模式，以及一系列新的问题使设计模式看起来难以应用。但是我从来没有忘记在工程化方法中存在着模式。通过学习识别模式，我们可以反复地使用这些健壮的解决方案。本书大部分内容着重于阐述在嵌入式系统开发中的标准设计模式，同时也给出了一些新的模式。而且，我还增加了一些章节，这些内容在绝大部分书里都找不到。

关于这本书

在研究了医疗设备、赛车、飞机、儿童玩具以及射击瞄准系统中的嵌入式系统之后，我发现了很多共性。如何为嵌入式系统设计和实现软件，要是那时候懂得很多就好了。本书包含了我所学到的知识。这是一本教授大家如何在资源受限的环境下设计优秀软件的书。

本书告诉你在应聘嵌入式系统开发工作时，了解面试官希望你身上看到什么。每章的最后都会有一道面试题。这些题通常与具体的语言不相关，而是希望能启发你的思路。好的面试题一般都没有唯一正确的答案，因此我在每道面试题下面给出了提示，没有列

注1: Gamma, Erich, Richard Helm, Ralph Johnson和John Vlissides (1995), 《设计模式: 可复用面向对象软件的基础》, 这本书已由机械工业出版社引进出版。

举所有可能的答案，而是给出了面试官期望从你的回答中看到什么的一些提示。毕竟，你得凭借你自身的优势去获得工作（和答案）。

有一点需要说明，这里提到的嵌入式系统是没有操作系统的、运行在裸机上的软件。当软件发出打开灯的指令时，它直接向处理器发出而没有经过任何中间层。本书不讨论嵌入式操作系统，但是相关概念可以运用到运行操作系统的处理机上。因此，如果你深入研究，就可以了解操作系统底层的原理。运行在没有操作系统的环境下会让人真正体会并感谢操作系统为我们所做的一切。

本书给出了在设计嵌入式软件时通用的原型和原则。我并没有基于任何特定的平台、处理器、编译器或者编程语言，因为通过本书打下扎实的基础之后，转向特定的平台自然水到渠成。

关于作者

在嵌入式系统领域中，我设计过DNA扫描器、飞机惯性测量单元、赛车、儿童玩具、抓捕罪犯的射击瞄准系统以及各种医疗和消费类设备。

我曾专门从事信号处理、硬件集成、复杂系统设计和性能优化工作。在经过美国联邦航空局（FAA）和美国食品药品监督管理局（FDA）认证过程后，我理解了高质量设计的重要性，以及高质量的设计如何引领高质量的实现。

我从事过多年的管理工作，但我更享受动手实践的工程设计工作以及交付优秀产品的兴奋和快感。我可以很高兴地说，离开管理工作并没有减少我领导和指导别人的机会。

致谢

本书并不是诞生于真空之中，而是始于同事的这样一句话：“嗨，你是否知道有哪本书我可以拿去给我的一个新手工程师阅读？”这句话埋下了后面数月写作的种子。我学会了真诚地感谢善解人意并且富于勇气的朋友，其中有花费了不少时间来评阅技术材料的工程师（当然，任何遗留的问题都是我的疏忽）。最后，在整个过程中，O'Reilly出版社提供了巨大的支持。

对每一个人表达恰如其分的感谢，可能需要好几页纸的篇幅，因此，这里我只是简单地把他们罗列一下，没有按照什么特别的顺序：Phillip King、Ken Brown、Jerry Ryle、Matthew Hughes、Eric Angell、Scott Fitzgerald、John Catsoulis、Robert P.J.Day、Rebecca Demarest以及Jen Costillo。这些同事为本书提供了与众不同的素材。此外，在本书写作的过程中，我还对那些在独特的领域给我提供帮助的人们表达了特别的感谢。因此，这些名字（或者这里没有提到的名字）还会出现在本书的其他地方。