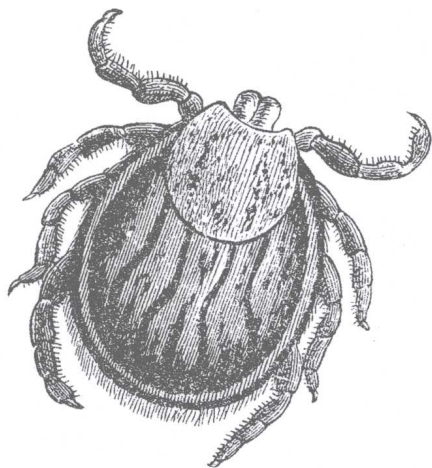


*Programming Embedded Systems  
with C and GNU Development Tools*

第二版  
Includes Real-Time and  
Linux Examples

# 嵌入式 系统编程

——使用C和GNU开发工具



O'REILLY®  
中国电力出版社



*Michael Barr & Anthony Massa 著*

*Jack Ganssle 序*

王映辉 王琼芳 李军怀 等译

第二版

---

# 嵌入式系统编程

—— 使用 C 和 GNU 开发工具

*Michael Barr & Anthony Massa* 著  
*Jack Ganssle* 序

王映辉 王琼芳 李军怀等 译

O'REILLY®

Beijing • Cambridge • Farnham • Köln • Sebastopol • Taipei • Tokyo

O'Reilly Media, Inc. 授权中国电力出版社

中国电力出版社

## 图书在版编目 (CIP) 数据

嵌入式系统编程：使用 C 和 GNU 开发工具：第二版 / (美) 巴尔安 (Barr, M.)，(美) 马萨 (Massa, A.) 著；王映辉等译. —北京：中国电力出版社，2009

书名原文：Programming Embedded Systems with C and GNU Development Tools, Second Edition

ISBN 978-7-5083-8376-7

I. 嵌… II. ①巴… ②马… ③王… 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 (2009) 第 008048 号

北京市版权局著作权合同登记

图字：01-2006-7452 号

©2006 by O'Reilly Media, Inc.

Simplified Chinese Edition, jointly published by O'Reilly Media, Inc. and China Electric Power Press, 2007. Authorized translation of the English edition, 2006 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 2006。

简体中文版由中国电力出版社出版 2007。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。

书 名 / 嵌入式系统编程——使用 C 和 GNU 开发工具 (第二版)

书 号 / ISBN 978-7-5083-8376-7

责任编辑 / 马首鳌

封面设计 / Karen Montgomery, 张健

出版发行 / 中国电力出版社 (www.infopower.com.cn)

地 址 / 北京三里河路 6 号 (邮政编码 100044)

印 刷 / 北京市同江印刷厂

开 本 / 787 毫米 × 980 毫米 16 开本 18 印张 348 千字

版 次 / 2009 年 5 月第 1 版 2009 年 5 月第 1 次印刷

印 数 / 0001 – 3000 册

定 价 / 36.00 元 (册)

## 敬告读者

本书封面贴有防伪标签，加热后中心图案消失。

本书如有印装质量问题，我社发行部负责退换。

版权所有 翻印必究

## 作者简介

---

**Michael Barr** 是电子设备软件设计的权威专家。他在美国的某区法庭提供了与这方面相关的专家证词，出现在 PBS 节目“American Business Review”上，并且被报纸文章所转载。Michael 是 40 多篇科技文章的作者，同时还是《嵌入式系统字典》这本书的共同执笔者。他还担任过“嵌入式系统编程”杂志三年半的主编工作。

由 Michael 在其职业生涯早期设计和编写的嵌入式软件已经在数以百万计的系统（从消费电子设备到医疗设备）中得到应用。现在 Michael 开始为商业构建嵌入式系统，而不是再针对个人了。他是 Quantum Leaps 公司的 CEO，还是 Netrino 公司的创始人。这两个公司在不同的方面帮助嵌入式开发人员编写出更好的嵌入式软件。

**Anthony Massa** 在嵌入式软件开发方面已经拥有了超过 10 年的开发经验。现在使用的几款软件的体系结构和开发环境都凝聚了他的心血，其中包括卫星和光缆调制解调器、无线设备、机顶盒和顶端传输设备。

Anthony 在顶级的软件开发杂志中已经写了几篇有关嵌入式软件开发方面的文章，同时也是《用 eCos 进行嵌入式软件开发》这本书的作者。Anthony 是 Elintrix 的奠基人之一 (<http://www.elintrix.com>)，同时也是它的主要软件工程师，Elintrix 是无线网络和信号处理产品的提供者。Anthony 拥有圣迭戈大学电子工程文学士和理学士的双学士学位。

## 封面介绍

---

本书封面上的昆虫是壁虱。全世界各地分布着大约有 850 多种这样的吸血微观寄生生物，它们尤其生存在那些热带和亚热带地区。有两种主要的壁虱种类：硬壁虱，它们的嘴可以看得得到；还有一种是软壁虱，它们的嘴部是隐藏着的。

无论是硬壁虱还是软壁虱，它们的嘴部都是由三个主要部分构成：唇须、前肢，还有垂唇。在壁虱吸血的时候，正是垂唇这部分深入到宿主的皮肤里。垂唇上一系列的反向投射性质的东西使得壁虱在吸血时很难从皮肤上滑落下来。壁虱还会分泌一些分泌物将它们粘贴在固定的位置，这种物质在壁虱吸完血之后就会失去吸附的作用。它们可以吸相当于自己体表 2~6 倍面积内所含的血液。

壁虱的一生经历三个阶段：幼虫、若虫和成虫。每个阶段它们分别以哺乳动物、爬行动物和鸟类宿主为食。壁虱靠前端的两个向外扩张的脚栖息在树叶上或是其他物体的表面等待宿主的出现。当宿主出现以后，它们便行动，吸附在宿主的身上。成年的雌性硬壁虱会在产一批数以千计的卵之后死去。成年的雄性壁虱也会在一次交配之后死去。

正像其它寄生生物一样，壁虱非常讨厌。它们与其他的吸血寄生生物相比，能传播更多的疾病，包括：莱姆病、洛基山斑疹热和回归热，它们也可以导致失血过多。一些根本无毒的壁虱也可能导致死亡。可以通过使用镊子或是尽可能靠近皮肤的清除设备将其处理掉。不要挤压壁虱，马上将它们用马桶冲掉。或者将其放入密封的容器内放上一个星期，以防引发疾病。

# O'Reilly Media, Inc.介绍

为了满足读者对网络 and 软件技术知识的迫切需求，世界著名计算机图书出版机构 O'Reilly Media, Inc. 授权中国电力出版社，翻译出版一批该公司久负盛名的英文经典技术专著。

O'Reilly Media, Inc. 是世界上在 Unix、X、Internet 和其他开放系统图书领域具有领导地位的出版公司，同时也是联机出版的先锋。

从最畅销的 *The Whole Internet User's Guide & Catalog*（被纽约公共图书馆评为 20 世纪最重要的 50 本书之一）到 GNN（最早的 Internet 门户和商业网站），再到 WebSite（第一个桌面 PC 的 Web 服务器软件），O'Reilly Media, Inc. 一直处于 Internet 发展的最前沿。

许多书店的反馈表明，O'Reilly Media, Inc. 是最稳定的计算机图书出版商——每一本书都一版再版。与大多数计算机图书出版商相比，O'Reilly Media, Inc. 具有深厚的计算机专业背景，这使得 O'Reilly Media, Inc. 形成了一个非常不同于其他出版商的出版方针。O'Reilly Media, Inc. 所有的编辑人员以前都是程序员，或者是顶尖级的技术专家。O'Reilly Media, Inc. 还有许多固定的作者群体——他们本身是相关领域的技术专家、咨询专家，而现在编写著作，O'Reilly Media, Inc. 依靠他们及时地推出图书。因为 O'Reilly Media, Inc. 紧密地与计算机业界联系着，所以 O'Reilly Media, Inc. 知道市场上真正需要什么图书。

## 译者序

不管您是了解个人电脑发展史还是紧跟着当今时尚潮流,苹果公司的产品都应该具有与众不同的地位。不只是它的产品开创了个人电脑的时代,苹果公司也作为重要角色不断书写着个人电脑的发展历史。历经几十年风雨起伏它又凭借着自己先进的技术、令人称赞的创新精神、优秀的产品和独到的市场运作,重新回到主流市场之中。作为现存不多的老牌个人电脑公司之一,苹果公司如今几乎成为时尚创新科技的代名词,市值超过思科、IBM、惠普、戴尔等行业巨头。

随着近几年来苹果电脑在国内市场的逐渐发展,现在越来越多的开发人员开始意识到应该在更具活力和创新精神的平台上开发优异的产品。《Cocoa入门—使用Objective-C》中文版的诞生也正是希望填补目前国内市场这方面的空白,通过它引领那些想在苹果平台有所作为的开发人员接触到这个平台独特而又强大的开发技术。秉承了O'Reilly一贯的优秀品质和同样独特的出版风格,本书的英文版在海外市场大受好评。

早在几年前苹果公司就开始和中国高校合作培养苹果平台未来的开发人员。四川大学作为第一批建立苹果开发技术实验室的高校还创建了苹果俱乐部,开设苹果开发课程,与其他合作高校一起培养出了一批苹果平台的爱好者。而与此对应的是,关于苹果电脑技术特别是开发技术的中文书籍相当匮乏。这抬高了苹果开发的入门难度,也把一批苹果平台的爱好者拦在开发者群体之外。译者很荣幸能将此书翻译介绍给国内读者,为苹果平台上开发技术在国内的普及贡献一份力量。

本书原版由James Duncan Davidson撰写,对于准备在Mac OS X平台上开发应用程序的人来说这是一部难得的好书。它条理明晰、层层深入,先以概述和基本介绍开始,通过各个主题逐渐深入,使读者对如何在Mac OS X平台上开发应用程序得到具体的认识。作者的用语虽然平实易懂却不乏精彩之处,尤其是各种形象的举例。最为称道的是每章都会有一个示例程序,手把手地指导您利用所学的知识创建具有相应功能的Cocoa应用程序。

由于 Mac OS X 已发布最新的版本 Mac OS 10.5 Leopard，而本书的原版是基于 Mac OS 10.2 编著。为此，译者做了相应修改，采用了 Xcode 而非早先的 ProjectBuilder 作为开发工具，并将所有图片在新的环境下进行了重新的截取和更换，所有的示例代码也在 Leopard 下进行了编译调试以验证其有效性，并做了必要修改。

所以说本书凝聚了多人的智慧和努力。彭舰翻译了本书的序言和第 1~10 章，张磊翻译了本书第 11~15 章，李征、琚生根翻译了本书的剩余章节，王俊和黄忍冬也参与了很多翻译和技术工作。彭舰负责全书的统稿工作。在书籍的翻译过程中有很多人参与其中并给出了诸多建议，在此特别表示感谢，他们是谢刚、简晓渝、谢川、廖朝辉、钟虹、伍敏、郭翔、张达平、熊奇、吴佳欣、刘唐、王红伟、李亭升和四川大学苹果俱乐部的同学们。苹果公司的端木恒先生完美地完成了本书的终稿技术审校工作，将本书的质量提升到新的高度。

感谢苹果公司夏鹏先生对本书的出版给予的极大帮助。感谢四川大学计算机学院院长周激流教授、唐宁九教授、谢汶副教授、朱敏教授、邵明松老师、卢莉老师和中国电力出版社，感谢他们对本书提供了很多有益的建议和热情的支持。

译者

2007 年 10 月

# 目录

|                         |           |
|-------------------------|-----------|
| 序 .....                 | 1         |
| 前言 .....                | 3         |
| <b>第 1 章 简介 .....</b>   | <b>11</b> |
| 什么是嵌入式系统? .....         | 11        |
| 设计模式的多样性 .....          | 14        |
| 嵌入式设计实例 .....           | 17        |
| 一位嵌入式软件开发者的生活 .....     | 20        |
| C 语言: 最基本的必需品 .....     | 21        |
| 硬件简介 .....              | 25        |
| <b>第 2 章 了解硬件 .....</b> | <b>27</b> |
| 了解概况 .....              | 27        |
| 硬件基础 .....              | 30        |
| 审视 .....                | 35        |
| 学会怎么通信 .....            | 39        |

---

|                                 |           |
|---------------------------------|-----------|
| 开始了解处理器 .....                   | 40        |
| 学习外部外围设备 .....                  | 44        |
| 初始化硬件 .....                     | 45        |
| <br><b>第 3 章 第一个嵌入式程序 .....</b> | <b>48</b> |
| “Hello, World!” .....           | 48        |
| 闪灯程序 .....                      | 49        |
| 无限循环的作用 .....                   | 56        |
| <br><b>第 4 章 编译、链接和定位 .....</b> | <b>57</b> |
| 构建过程 .....                      | 57        |
| 构建闪灯程序 .....                    | 65        |
| 快速浏览 makefiles 文件 .....         | 69        |
| <br><b>第 5 章 下载和调试 .....</b>    | <b>73</b> |
| 下载闪灯程序 .....                    | 73        |
| 远程调试器 .....                     | 79        |
| 仿真器 .....                       | 86        |
| 其他有用的工具 .....                   | 87        |
| 深入理解硬件 .....                    | 93        |
| <br><b>第 6 章 内存 .....</b>       | <b>94</b> |
| 存储类型 .....                      | 94        |
| 直接存储器存取 .....                   | 98        |
| 字节序问题 .....                     | 99        |
| 存储检测 .....                      | 102       |
| 确认存储内容 .....                    | 114       |
| 闪存的使用 .....                     | 117       |

|                              |                |
|------------------------------|----------------|
| <b>第 7 章 外围设备 .....</b>      | <b>120</b>     |
| 控制寄存器和状态寄存器 .....            | 120            |
| 设备驱动思想 .....                 | 128            |
| 设备驱动设计 .....                 | 137            |
| <br><b>第 8 章 中断 .....</b>    | <br><b>139</b> |
| 总 述 .....                    | 139            |
| 中断映射 .....                   | 144            |
| 中断服务程序 .....                 | 146            |
| 改进的闪灯程序 .....                | 152            |
| 中断问题总结 .....                 | 158            |
| <br><b>第 9 章 整体理解 .....</b>  | <br><b>160</b> |
| 应用总述 .....                   | 160            |
| 使用串行端口 .....                 | 162            |
| 命令行接口处理 .....                | 163            |
| <br><b>第 10 章 操作系统 .....</b> | <br><b>168</b> |
| 历史和目的 .....                  | 168            |
| 调度程序 .....                   | 169            |
| 任 务 .....                    | 174            |
| 任务同步 .....                   | 180            |
| 消息传递 .....                   | 184            |
| 其他功能 .....                   | 184            |
| 中断处理 .....                   | 185            |
| 实时特性 .....                   | 186            |
| 使用或不使用 RTOS .....            | 187            |
| 更多资源 .....                   | 189            |

|                                      |                |
|--------------------------------------|----------------|
| <b>第 11 章 eCos 实例 .....</b>          | <b>191</b>     |
| 简介 .....                             | 191            |
| 任务机制 .....                           | 192            |
| 互斥量任务同步 .....                        | 194            |
| 信号量任务同步 .....                        | 198            |
| 消息传递 .....                           | 202            |
| eCos 中断处理 .....                      | 205            |
| <br><b>第 12 章 嵌入式 Linux 实例 .....</b> | <br><b>210</b> |
| 简介 .....                             | 210            |
| 在 Linux 中访问硬件 .....                  | 211            |
| 任务机制 .....                           | 211            |
| 互斥量任务同步 .....                        | 213            |
| 信号量同步任务 .....                        | 215            |
| 消息传递 .....                           | 218            |
| <br><b>第 13 章 扩充功能 .....</b>         | <br><b>223</b> |
| 通用外围设备 .....                         | 223            |
| 适用于大型和小型设备的网络 .....                  | 233            |
| <br><b>第 14 章 优化技术 .....</b>         | <br><b>239</b> |
| 提高代码效率 .....                         | 240            |
| 减少代码量 .....                          | 243            |
| 最优编译器问题 .....                        | 244            |
| 减少存储使用 .....                         | 245            |
| 节省能源的技术 .....                        | 247            |
| 限制 C++ 的影响 .....                     | 250            |

---

|                                   |     |
|-----------------------------------|-----|
| 附录 A Arcom VIPER-Lite 开发工具包 ..... | 253 |
| 附录 B 安装你的软件开发环境 .....             | 256 |
| 附录 C 实现 GNU 软件工具 .....            | 261 |
| 附录 D 安装 eCos 开发环境 .....           | 264 |
| 附录 E 安装嵌入式 Linux 开发环境 .....       | 267 |

---

# 序

如果谈到“嵌入式”这个词，很多人会认为是在谈论战区记者的一些事情。包括正规的牛津英语辞典在内，很少有辞典将“嵌入式”这个词和计算机系统联系起来。然而，从蜂窝式移动电话到车库自动门开关，再到医疗器械设备，嵌入式系统却几乎已成为现今所有电子产品的最基础、最低层的支持。直到现在，要想在不添加任何微处理器和相关软件的条件下来开发出一款电子产品设备，那几乎是不可能的事情。

制造商们每年大约要生产出90亿个微处理器。其中可能只有1亿或是1.5亿的微处理器用到了PC机的生产制造上。也就是说只有1%的微处理器芯片用到了PC机的生产上，其余的99%都用到了嵌入式系统中。显然，这种商业行为表明了当今高科技社会上的一种细微的结构形态。

同时，这些技术的使用只会越来越多。环境问题的解决方案也将会依靠灵巧嵌入式系统资源的参与。你必须了解有关Toyota混合型Prius的32位处理器网络，以便了解其将来的发展趋势。

尽管现在很难预言，但是消费者将来对那些包括多处理器和相关复杂软件的智能化产品的需求必然会增加的。据估计，大多数产品的固件的使用量每10~24个月就会翻一番。然而在代码需求量不断增加的同时，我们的生产率却增长缓慢。所以很显然，在工业生产上会需要很一大批嵌入式人才来满足日益增长的需求。

那么这些嵌入式人才到底需要什么样的技术呢？在PC行业技术人员必须是一个能力很强的C/C++程序员。但是作为一个嵌入式开发人员，你必须对编程语言和硬件本身都有较深入的认识和理解。例如，如果不知道中断是从哪里来的，也不清楚硬件针对它们所具有的优先权，还不知道有关保存系统环境的硬件和机制细节的实际原理的话，你根本

就不能设计、编码、测试中断服务程序。作为一个固件设计者，你必须在写每一行驱动代码之前，就能够观察到系统外设硬件执行时的细节问题。

在PC环境中，有关硬件的秘密则被隐藏在大量的API后面。在一个嵌入式系统中，API经常是由开发产品的工程师来编写的。

在本书中，Michael Barr和Anthony Massa两位作者讲解了软件和硬件是怎样结合成为一个协调的整体的。他们没有回避有关中断和I/O或是优先权转换和互斥量方面的细节问题。

作者恰到好处地演示了使用各种开放源工具来构建嵌入式系统，其中的工具包括GNU编译器组件，它是在业界广泛使用的一种标准工具。eCos和Linux也都是即免费又开源的产品，它们也是用来演示小型或大型操作系统的。

本书的原始版本使用的是x86电路板，但是在本版中已经用基于ARM的产品取代了。巧合的是，正当本版图书在制作的时候，Intel公司宣称所有本公司的嵌入式x86处理器生产终止。读者可以确定的是：因为ARM得到很多基础零售商的支持，所以它在市场上会占据相当长的时间。

硬件通常价格不贵，而且很容易获得，而软件是免费的。这样也就使嵌入式系统开发成为了主流。读者将来一定会使用到这些工具的。

购买开发工具包，阅读本书，然后实现书中的实例。你就会拥有职业人员所需的来自有经验人员的指点：构建和利用嵌入式应用程序。

—— Jack Ganssle

---

# 前言

首先要指明你为什么要求学生学习这门课程，  
以及你希望他们从中学到什么，  
他们可以或多或少获得一些常识的途径。

—— Richard Feynman

目前，几乎每个电子设备都要使用嵌入式软件。这些软件隐藏在我们的手表里、DVD播放器里、移动电话里、防锁刹车里，甚至是一些烤箱里。军事上用嵌入式软件来为导弹导航、侦测敌方飞行器，以及用于无人驾驶的飞机里。人造通信卫星、外太空探测器和许多医疗器械中没有它简直就不可能造得出来。

总得有人写出所有这些嵌入式软件。实际上有成千上万的电子工程师、计算机科学家，以及其他的专业人士都在这样做，我们只是他们其中的一员。而且通过我们的个人经历，我们知道学习这些技术是很困难的。

每个嵌入式系统都是独特的，而硬件是特别针对应用领域的。因此，嵌入式系统编程具有多样性，要想熟练掌握往往就要花费许多年的时间。不过，几乎所有的嵌入软件开发有一个共同的特征：它们都是使用C程序语言编写的。本书会教给你在嵌入式系统中如何使用C语言。

即使你已经知道怎样写嵌入式软件，你仍然能从本书中学到很多。在学习如何有效使用C语言之余，你也会从详细的解说和从嵌入式软件常见问题有关的源代码中获益。本书的高级知识有内存测试和验证、设备驱动设计和实现、实时操作系统内部机理，以及代码优化技术。

## 我们为什么写这本书

每年，全球大约平均每人造出一个新的处理器。每年都有超过60亿新的处理器，这个数目低于所有个人电脑新机中奔腾和Power PC的2%。你可能会疑惑在我们身边有没有那么多电脑。但是我可以打赌不到五分钟你就可能发现自己家中有几打的商品中都含有处

理器，诸如电视、音响、MP3播放器、咖啡炉、闹钟、录像机、DVD播放机、微波炉、洗碗机、遥控器、面包机、数字手表等。而这些只是家用物品——这类设备中更多的是用于工作。实际上每个这样的产品不仅包括处理器，也包括软件，这正是促成本书的动力。

关于本书主题最难的是要知道什么时候应该停笔，不要再写。每个嵌入系统都是唯一的，并且我们知道，任何法则都会有例外。不过，对于嵌入式系统中程序员必需的相关知识，我们尝试提取其中的精髓并将之呈现出来。

## 针对的读者群

这是一本使用C语言编写嵌入式系统的书。同样地，它假定读者已经有一些编程经验，至少熟悉C语言的语法。如果你熟悉基本数据结构，例如链表等则会更有益。这本书不要求你有很多电脑硬件的知识，但是希望你通过本书的学习之后，能够掌握一些硬件方面的知识。毕竟，这是嵌入式程序员应该做到的。

本书写作时，我们预期有两类读者。第一类是初学者——和我们当初一样。他应该有计算机或者工程专业背景，有几年编程经验。初学者感兴趣的是写嵌入式软件来谋生但不了解怎样开始才好。读过前几章之后，他就能应用自己的编程技巧来开发简单的嵌入式程序了。余下的章节类似于参考资料，对于程序员将来的职业生涯中会遇到的高级课题具有指导作用。

第二类读者已经是嵌入式系统的程序员了。这类读者熟悉嵌入式硬件，也知道怎样来为之写软件，但是他们正在寻找能够解释那些关键问题的参考书。也许嵌入式系统程序员仅仅只有汇编语言编程的经验，相比较而言，对C语言来说还是一个新手。这种情况下，本书将教会他们怎样在嵌入式系统中高效地使用C语言，并在稍后的章节里提供实时操作系统、外围设备和代码优化的更进一步的资料。

无论你是不是所提到的上述几种类型之一，我们希望本书提供的信息正是你在寻找的，而且形式明了，容易理解。

## 结构

本书包括14章和5个附录。这些章节能很好地分成两部分。第一部分由第1章～第5章组成，主要针对嵌入式系统的新手。这些章节应该按顺序全部看完。这会使你快速提高，也会让你为嵌入式软件开发打好基础。完成第5章后，你就为独立地开发各种小型的嵌入式软件做好了准备。