HZ BOOKS
华章IT

Pea

Linux/Unix
技术丛书

Linux

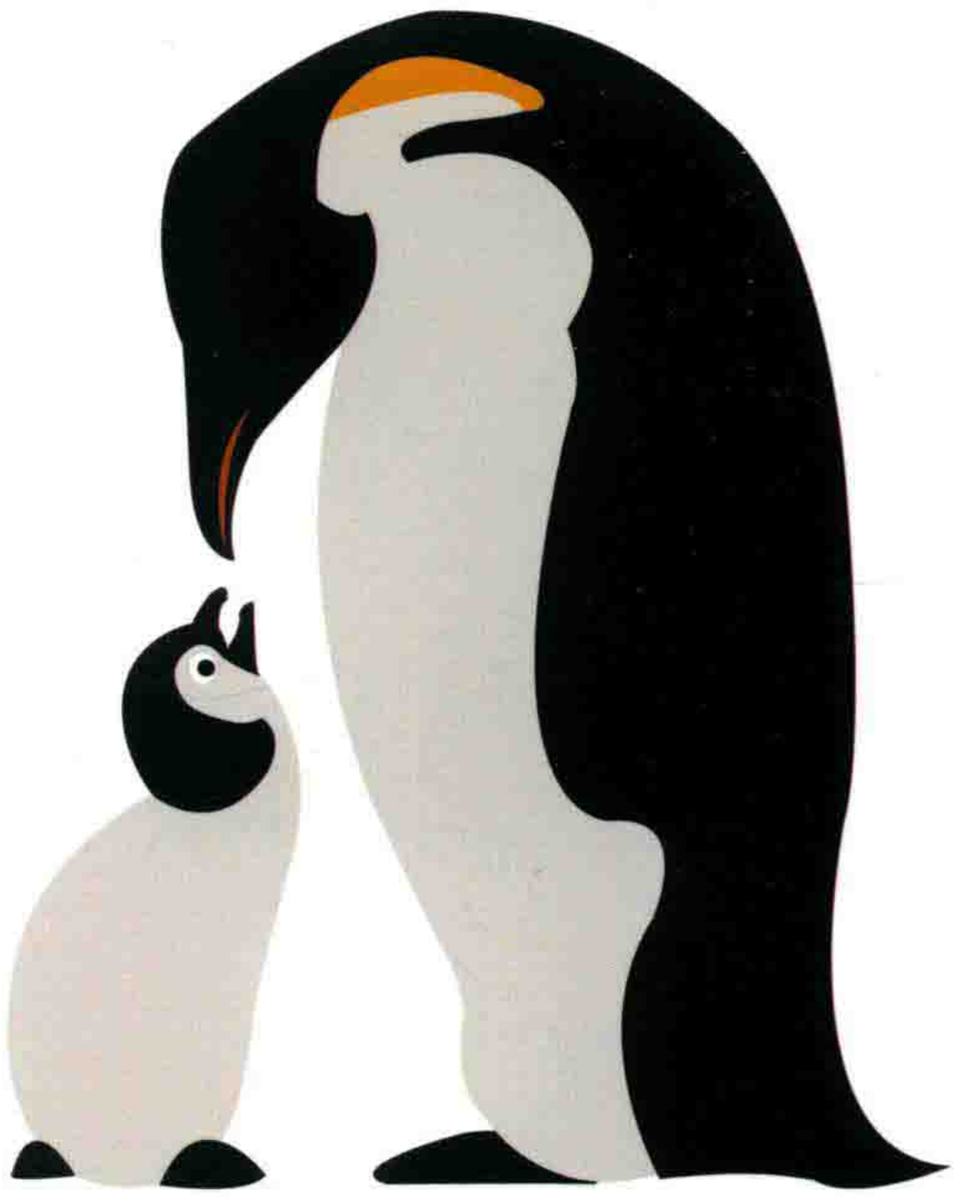
Embedded Linux Systems with the Yocto Project

嵌入式Linux系统开发

基于Yocto Project

[美] 鲁道夫·J. 斯特雷夫（Rudolf J. Streif） 著

胥峰 译



机械工业出版社
China Machine Press

嵌入式Linux系统开发

基于Yocto Project

[美] 鲁道夫·J. 斯特雷夫（Rudolf J. Streif） 著

胥峰 译

机械工业出版社
China Machine Press

图书在版编目（CIP）数据

嵌入式Linux系统开发：基于Yocto Project/（美）鲁道夫·J. 斯特雷夫（Rudolf J. Streif）著；胥峰译 . —北京：机械工业出版社，2018.6
（Linux/Unix技术丛书）
书名原文：Embedded Linux Systems with the Yocto Project

ISBN 978-7-111-60382-5

I.嵌… II.①鲁… ②胥… III. Linux操作系统－程序设计 IV. TP316.85

中国版本图书馆CIP数据核字（2018）第146528号

本书版权登记号：图字 01-2016-8663

Authorized translation from the English language edition, entitled *Embedded Linux Systems with the Yocto Project*, ISBN:9780133443240 by Rudolf J. Streif published by Pearson Education, Inc, Copyright © 2016 Pearson Education, Inc.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc.

Chinese simplified language edition published by China Machine Press, Copyright © 2018.

本书中文简体字版由Pearson Education（培生教育出版集团）授权机械工业出版社在中华人民共和国境内（不包括香港、澳门特别行政区及台湾地区）独家出版发行。未经出版者书面许可，不得以任何方式抄袭、复制或节录本书中的任何部分。

本书封底贴有Pearson Education（培生教育出版集团）激光防伪标签，无标签者不得销售。

嵌入式Linux系统开发：基于Yocto Project

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：张志铭 责任校对：李秋荣
印 刷：北京市兆成印刷有限责任公司 版 次：2018年7月第1版第1次印刷
开 本：186mm×240mm 1/16 印 张：23
书 号：ISBN 978-7-111-60382-5 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

版权所有·侵权必究
封底无防伪标均为盗版
本书法律顾问：北京大成律师事务所 韩光/邹晓东

The Translator's Words 译 者 序

为什么要翻译这本书

物联网的兴起推动着越来越多的嵌入式设备投入使用，从普通的家用设备到用于提供治安监控、交通流控等公共服务的设备，其中的核心之一就是嵌入式操作系统。而 Linux 正成为越来越多嵌入式开发者的首选。

市面上不缺乏适用于各种开发语言的开发、编译和打包工具，但在 Yocto 项目出现前，这些零散的工作需要嵌入式开发者自己串起来以交付最终的嵌入式系统。Yocto 项目恰好解决了这些问题，它是很多工具的有机整合，可以让嵌入式开发者更容易且更高效地交付嵌入式系统。

本书就是专注于讲解 Yocto 项目的实战手册。书中从一开始就强调“动手实验”的重要性，鼓励读者跟着里面的例子动手做起来，最终完全了解整个过程并知道每个步骤需要注意什么。

另外，本书也讲解了如何在嵌入式系统开发和交付过程中管理许可及合规性。在强调知识产权的今天，这一点的重要性也不言而喻。

Yocto 项目社区经理、本书英文原版技术审稿人 Jeffrey Osier-Mixon 在亚马逊网站（https://www.amazon.com/gp/customer-reviews/R20UE23XPGTB7Y/ref=cm_cr_arp_d_rvw_ttl?ie=UTF8&ASIN=0133443248）对本书评论说：“该书非常细致地解释了系统的各个工具和组件是如何相互适应的，从整体构建工具 BitBake 及其操作的菜谱（元数据）开始，然后深入系统经过的各个过程。最有用的是，它从开发者的角度讲解构建系统，涵盖了开发者在构建过程的各个环节需要解决的问题……总之，本书是了解使用 Yocto 项目工具构建用于嵌入式系统的 Linux 之极佳资源。我非常推荐。”简而言之，本书是一位有着超过 20 年软件工程实践经验的专家的扛鼎之作，它务实、细致、逻辑清晰。它不仅仅是一本参考书，更是一本实践指南。

本书中文版可能是专注于 Yocto 项目的先驱性图书，希望能对广大嵌入式系统开发者有所帮助。

读者对象

本书适合以下几类读者阅读：

- 嵌入式开发工程师
- 嵌入式测试工程师
- 嵌入式产品经理
- 嵌入式产品合规工程师
- 计算机相关专业学生

勘误

虽然译者努力保证本书翻译中不出现错误，但鉴于译者的知识和能力，书中难免出现用词错误、适用性等问题。在此，恳请读者不吝指教，指出错误。请读者发送邮件到 xufengnju@163.com 帮助译者修正错误。勘误将发布在 https://github.com/xufengnju 上。

致谢

感谢机械工业出版社华章公司引进了原书的中译本版权，这是本书中文版得以面市的最重要条件。感谢华章公司张志铭和关敏老师，你们专业的编辑能力为本书提供了重要的质量保证。

感谢我的妻子吕宁和可爱的女儿胥欣，谢谢你们的支持和理解。

胥峰

2018 年 3 月

此为试读，需要完整PDF请访问：www.ertongbook.com

Foreword 序

嵌入式 Linux 的形势有点像美国老西部（the Old West）：不同的技术前哨分散在各处，在这些技术前哨之间是贫瘠和往往十分危险的形势。如果想去那里旅行，则需要有充足的储备，要熟悉地形，并且要有可靠的向导。

正如在 18 世纪中期淘金热期间人们迁移到美国老西部一样，伴随着物联网（Internet of Things，IoT）的热潮，开发者正在进入嵌入式 Linux 的世界。正如增加的人口为老西部带来法律、秩序和文明一样，重要的新开源软件项目正在为嵌入式 Linux 带来秩序。

Yocto 项目是十分重要的秩序缔造者。它的工具能帮你主要关注设计项目（你想要构建的东西）并且仅仅投入最少的必要时间和精力来把它们整合在一起（如何构建你想要的东西）。

这本书是一个可靠的向导。通过按逻辑排序并带有清晰和完整说明的章节，它将帮助你完成工作并把物联网项目带向市场。如果运气不错，你将会在一路上收获快乐。

享受冒险吧！

Arnold Robbins

Prentice Hall 开源软件开发系列编辑

前　言 *Preface*

智能家居、智能汽车、智能手机、智能电视、智能恒温器、智能电灯、智能手表、智能洗衣机、智能烘干机、智能冰箱以及智能篮球。欢迎来到智能万物的勇敢新世界。

在日常生活中，我们触及和交互的几乎所有东西中都有嵌入式计算机，它的剧增，使得嵌入式系统工程和嵌入式软件开发受到了瞩目。嵌入式系统隐藏在其用户的直接视线之外，它们缺少那些带有浮华用户界面的网络应用所具有的吸引力，也缺少那些带有动画和逼真图像的计算机游戏所具有的酷酷的感觉。毫不奇怪的是，计算机科学专业的学生和软件开发者几乎不会把嵌入式软件工程当成他们的第一职业选择。然而，“智能万物革命”和物联网正在推动人们对那些能够桥接硬件和软件世界的专业人士的需求。那些使用电路原理图语言和编程语言的专家将受到雇主的追捧。

对于爆炸式增多的嵌入式应用来说，Linux 已经成为第一选择。对于这种选择，有很多恰当的理由。我们将在接下来的章节中详细阐述这些理由。经历了作为服务于各种行业的嵌入式软件开发者的旅程，我通过艰辛的方式学习了嵌入式系统的 Linux。对几乎任何编程语言来说，都不缺少优秀的开发工具。绝大部分用于 Linux 的库和应用都会由于它们的工具化而被简单原生地构建。使用内核自有的构建系统，即使从头开始构建 Linux 内核几乎也是一件轻而易举的事。然而，当需要把所有的东西整合进可启动的系统时，选择就很少了。

通过提供一套全面的、以 OpenEmbedded 构建系统为中心的集成化工具，Yocto 项目弥补了这种鸿沟。在大约几个小时内实现从源代码到可启动的系统——我希望在我开始使用嵌入式 Linux 的时候会有这样的奢侈。

本书定位

集成多个不同的必要步骤来从头创建完全可用的 Linux 操作系统栈的构建系统是相当复

杂的。本书专注于构建系统本身以及如何有效使用它来构建定制的 Linux 发行版。本书不是关于嵌入式 Linux 的教程。虽然第 6 章解释了 Linux 系统架构的基础（因为该信息对于理解构建系统如何组合多个不同的组件成为可运行的系统是必要的），但是就嵌入式 Linux 本身而言，不会深入它的细节。如果你是嵌入式 Linux 开发新手，那么强烈推荐 Christopher Hallinan 的好书《 Embedded Linux Primer 》，该书与本书是同一系列丛书（ Prentice Hall 开源软件开发系列）。

在本书中，你将会学习 OpenEmbedded 构建系统如何工作、如何编写菜谱（ recipe）来构建你自己的软件组件、如何使用和创建 Yocto 项目板支持包（Board Support Package，BSP）来支持不同的硬件平台以及如何调试构建失败。你还将学习如何为应用开发构建软件开发包，以及如何为了无缝的往返开发（ round-trip development）而把这些开发包和流行的 Eclipse 集成开发环境（Integrated Development Environment，IDE）集成起来。

本书读者

本书面向那些具有 Linux 实践知识的软件开发者和程序员。假设你熟悉 Linux 命令行，可以使用典型的工具（例如 Make 和 C/C++ 编译器）在 Linux 系统上构建程序，还可以阅读和理解基本的 shell 脚本。

构建系统是完全用 Python 编写的。即使你不是 Python 专家也可以使用它，并理解它如何工作，但是具备一些核心的 Python 知识一定是有利的。

本书内容

第 1 章对采用嵌入式系统的 Linux 进行简要介绍，为嵌入式 Linux 的形势和创建定制嵌入式发行版的挑战的概览奠定了基础。

第 2 章通过用构建系统启动 Linux 操作系统栈的首次构建来介绍 Yocto 项目，同时也给出了 Yocto 项目系列和 Yocto 项目历史的概览。

第 3 章解释了构建系统的基础、工作流和架构。

第 4 章解析了 BitBake——处于 OpenEmbedded 构建系统核心的构建引擎。其中解释了菜谱（ recipe）的元数据（ metadata）概念、类、配置文件及其语法。BitBake 方式的“ Hello World”项目说明了构建工作流。通过提供的信息，你会获得用于理解所提供的菜谱和编写自己的菜谱的必要知识。

第 5 章介绍了可用于对构建问题进行故障排除的工具和机制，并且提供了如何有效使用

这些工具的实用建议。

第 6 章提供了 Linux 操作系统栈的基础，并解释了不同组件是如何分层的。另外还讨论了内核空间和用户空间的概念以及应用程序如何通过由标准 C 库提供的系统调用来和 Linux 内核交互。

第 7 章细致讲解了如何使用 Yocto 项目创建定制化的 Linux 发行版。它以构建系统可用的 Linux 发行版蓝图以及如何定制它们的简要介绍作为开始。然后演示了如何使用构建系统工具完全从头创建 Linux 发行版。读完本章以后，你将知道如何构建自己的操作系统镜像。

第 8 章解释了 BitBake 菜谱以及如何编写它们来用构建系统构建自己的软件包。这一章提供了你可以尝试的各种真实世界的菜谱例子。

第 9 章仔细研究了用 OpenEmbedded 构建系统来构建 Linux 内核的细节，解释了构建系统工具如何与内核的构建环境交互以设置内核配置和应用补丁。本章以关于构建系统如何处理树外（out-of-tree）内核模块以及合并构建设备树和构建过程的讨论结束。

第 10 章介绍了构建系统如何支持针对不同硬件的构建——也就是说，为不同 CPU 架构和系统构建。在介绍了 Yocto 项目板支持包的概念以后，本章细致讲解了如何使用板支持包构建项目。然后研究了 Yocto 项目板支持包的内部原理，并用实际的例子来解释如何构建自己的板支持包，你可以将这个例子使用在真实硬件上。本章以为不同硬件配置创建可启动介质镜像结束。

第 11 章描述了 Yocto 项目对开发应用的支持，这些应用被用在通过构建系统创建的 Linux 操作系统栈上。其中提供了关于如何构建那些包含用于往返应用开发的全部必需工具的应用开发工具集（Application Development Toolkit，ADT）的实际操作指南。示例说明了如何通过命令行工具和 Eclipse 集成开发环境来将应用开发工具集用于应用开发。指南会一步步教你如何远程运行和调试真实硬件目标上的应用。

第 12 章讨论了对于开源许可合规性的要求以及 Yocto 项目提供的那些有助于满足要求的工具。

第 13 章介绍了帮助你把 Yocto 项目扩展到团队的一些工具。Toaster 是基于 Web 的图形用户界面，它可用于创建能从 Web 浏览器远程控制的构建系统。构建历史是提供追踪和审计能力的工具。使用源镜像，你可以共享源包来避免重复的下载，并为产品交付控制源版本。最后但并非最不重要的是，Autobuilder 为自动化构建、质量保证和发布过程提供了开箱即用的持续构建和集成框架。掌握了本章的知识后，你可以为 Yocto 项目有效地设置团队环境。

附录包括流行的开源许可，以及以字母排序的构建系统元数据层和机器的参考。

动手操作经验

本书的撰写目的是提供 Yocto 项目的实际动手操作经验。如果依照并且尝试了相关示例，那么你会收获良多。它们中的大部分可以依赖运行最新 Linux 发行版的基于 x86 的工作站来轻松完成（详细的需求会在第 2 章提供）。为了获得更好的经验，要利用流行的开发板，例如 BeagleBone、MinnowBoard Max 或者 Wandboard。BeagleBone 是优秀的低成本的实验平台。另外两个板提供更好的性能，也让你获得关于多核系统的经验。

分析代码并且尝试理解本书的代码。遵循步骤，然后通过改变设置、应用你自己的配置等来完成自己的设计。这是最好的学习方式，并且我可以告诉你，这也相当有趣。让你自己的第一个 Linux 发行版工作在你选择的一个硬件上是很棒的体验。

致　谢 *Acknowledgements*

本书是我第一次尝试撰写的技术书籍。好吧，就此而言，是我人生第一本书。我必须谦逊地承认，我大大低估了开始这样一个项目所需要的精力——花费在实验上的时间、找到最好的让它们工作的方法以及以简洁和可理解的方式文档化所有事物。在这个过程中，要真心感谢很多作者和技术撰稿人的工作——我已经阅读过很多并且正在持续阅读他们的书籍和手册。

首先，对我的家庭，包括我挚爱的妻子 Janan 和 3 个非常棒的儿子 Dominic、Daniel 和 Jonas 表示感谢。如果没有他们的支持和理解，我是不可能在写作上花费很多时间的。

特别感谢 Yocto 项目团队。当我接触彼时还是 Yocto 项目项目经理的 Dave Stewart 和 Yocto 项目社区经理的 Jeffrey Osier-Mixon 时，他们立刻对写作本书的主意表示欢迎并且提供了支持。这个团队的一些人在给予建议和问题解答方面特别有帮助：Beth Flanagan 对 Autobuilder 方面、Belen Barros Pena 和 Ed Bartosh 对 Toaster 方面以及 Paul Eggleton 和 Khem Raj 对我提交到 Yocto 项目邮件列表里面的问题提供的参考。

特别感谢撰写了《Embedded Linux Primer：A Practical Real-World Approach》（Prentice Hall, 2006）的 Christopher Hallinan。他写的这本书激发了我创作关于 Yocto 项目的书的欲望。

特别感谢总策划编辑 Debra Williams Cauley，感谢她的指导，特别是在本书写作过程中付出的耐心。本书花费了远超预期的时间，而我是唯一需要为本书错过交稿期限负责的人。

怎么感谢和表扬本书专注的审核团队都不为过，他们是 Chris Zahn、Jeffrey Osier-Mixon、Robert Berger 和 Bryan Smith。感谢他们的纠错及改进建议，这保证了本书的质量。

我也想感谢 Prentice Hall 的生产团队——Julie Nahil 和 Anna Popick，感谢他们的协调和指导工作，特别是 Carol Lallier 在编辑原稿过程中的勤勉。

还要感谢 Linux 基金会和 Jerry Cooperstein，他们给了我开发 Linux 基金会的关于 Yocto 项目的培训课程的机会。没有什么方法比教别人更利于学习。感谢我教过的学生。通过他们的宝贵提问和反馈，我获得了大量对于学生用嵌入式 Linux 开发产品时遇到的很多不同问题的理解。他们问得最多的问题是：“有没有一本关于 Yocto 项目的书?”终于，我可以回答：“有的。”

此为试读，需要完整PDF请访问：www.ertongbook.com

Contents 目　录

第 1 章 Chapter 1

用于嵌入式系统的 Linux

物联网（Internet of Things）正激发着梦想家们的想象灵感，也激发着工程师们的创造性。作为由巨大数量的实时收集、分析和传递数据的互联设备组成的统一计算网络，物联网承载着新的信息技术时代的希望。

组成物联网的设备需要满足一套全新的需求并且提供原来嵌入式系统中不存在的功能性。连通性，包括通过蜂窝数据网络连通，是一个明显的需求，其他还有远程管理、软件配置和更新、能耗效率以及长的寿命，当然还有安全性，仅举几例。

正在改变的嵌入式系统的形势需要新的方法来构建操作这种新的互联硬件的软件栈。

1.1 为什么为嵌入式系统选择 Linux

Linux 首次亮相是作为英特尔 x86 架构的 PC 硬件所用的通用操作系统（General-Purpose Operating System，GPOS）。正如 Linux 创造者 Linus Torvalds 在 news:comp.os.minix 上现在非常有名的帖子中所写的，他明确说道，“我正在做一个（免费）操作系统……它是不可移植的（使用 386 任务切换等），并且它可能永远不会支持除了 AT 硬盘以外的硬盘……”

被互联网的兴起所驱动，因为很多理由充分的原因，Linux 迅速进化成为网络服务器和联网服务提供基础设施的服务器操作系统。

然而，在三个主要方面，Linux 保持了它的通用操作系统的初始特点。这三个方面使得它在初期并没有成为工程师为嵌入式操作系统所定的首要选择：

- **文件系统**：Linux 是基于文件的操作系统，它需要一个在有读和写访问的面向块的大容量存储设备上的文件系统。面向块的大容量存储通常意味着硬盘驱动器具有旋转的盘片，而这对于大部分嵌入式的用例都是不实际的。

此为试读，需要完整PDF请访问：www.ertongbook.com