

- ★ 深入浅出，介绍 CPU 设计技巧和新兴 RISC-V 指令集架构。
- ★ 内容全面，讲解开源蜂鸟 E200 系列处理器代码与设计精髓。
- ★ 画龙点睛，涵盖全套 SoC、软件工具链和 FPGA 原型平台的搭建和使用。



手把手

教你设计 CPU—— RISC-V 处理器 篇

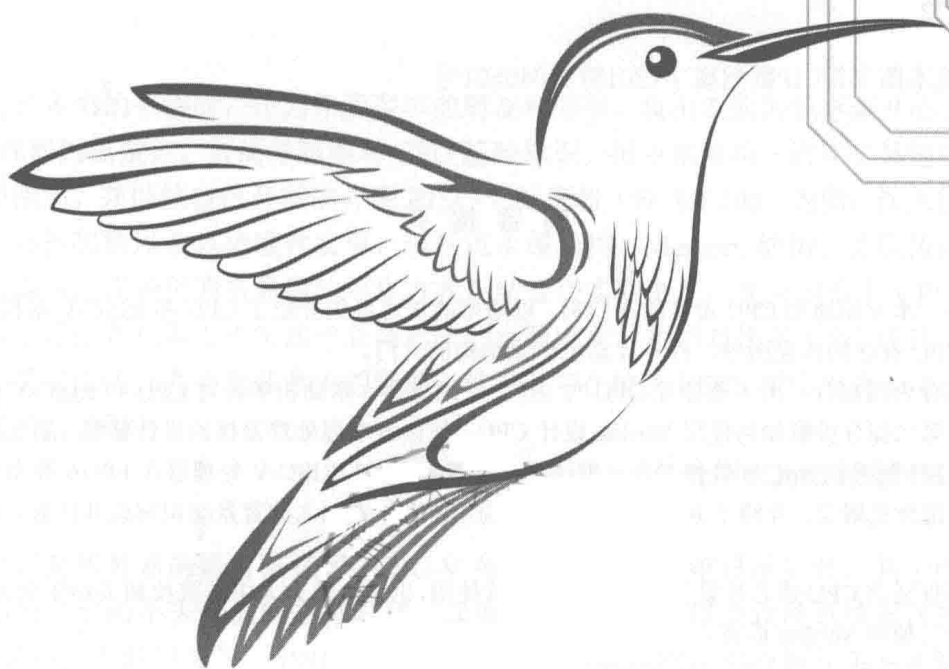
胡振波◎著



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS



手把手

教你设计 CPU—— RISC-V 处理器^篇

胡振波◎著

人民邮电出版社

北京

图书在版编目 (C I P) 数据

手把手教你设计CPU. RISC-V处理器篇 / 胡振波著

— 北京 : 人民邮电出版社, 2018.6

ISBN 978-7-115-48052-1

I. ①手… II. ①胡… III. ①微处理器—系统设计
IV. ①TP332

中国版本图书馆CIP数据核字(2018)第046501号

内 容 提 要

本书是一本介绍通用 CPU 设计的入门书,以通俗的语言系统介绍了 CPU 和 RISC-V 架构,力求为读者揭开 CPU 设计的神秘面纱,打开计算机体系结构的大门。

本书共分为四部分。第一部分是 CPU 与 RISC-V 的综述,帮助初学者对 CPU 和 RISC-V 快速地建立起认识。第二部分讲解如何使用 Verilog 设计 CPU,使读者掌握处理器核的设计精髓。第三部分主要介绍蜂鸟 E203 配套的 SoC 和软件平台,使读者实现蜂鸟 E203 RISC-V 处理器在 FPGA 原型平台上的运行。第四部分是附录,介绍了 RISC-V 指令集架构,辅以作者加入的背景知识解读和注解,以便于读者理解。

本书不仅适合 CPU 或芯片设计相关从业者阅读使用,也适合作为大中专院校相关师生学习 RISC-V 处理器设计(使用 Verilog 语言)和 CPU 设计的指导用书。

◆ 著 胡振波

责任编辑 张 爽

责任印制 焦志炜

◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号

邮编 100164 电子邮件 315@ptpress.com.cn

网址 <http://www.ptpress.com.cn>

北京市艺辉印刷有限公司印刷

◆ 开本: 800×1000 1/16

印张: 26.75

字数: 598 千字

印数: 1—3 000 册

2018 年 6 月第 1 版

2018 年 6 月北京第 1 次印刷

定价: 99.00 元

读者服务热线: (010)81055410 印装质量热线: (010)81055316

反盗版热线: (010)81055315

广告经营许可证: 京东工商广登字 20170147 号

精彩书评

当拿到这本书的书稿时，作为作者多年的朋友和同事，我由衷地为他感到开心。作者以 RISC-V 处理器为出发点，言简意赅地对 CPU 基础知识、指令集架构，软件工具链等核心技术做了原理阐述，并以他自己开发的开源 RISC-V 处理器（蜂鸟 E200）为例，深入浅出地介绍了该处理器微架构以及具体硬件实现，包括流水线结构、Memory 结构、总线协议、中断异常和调试器等。作者用通俗易懂的写作方式，结合切实的例子，充分剖析了 CPU 设计的全过程。本书是作者长期工作实践中总结出来的经验宝典，相信对很多 CPU 设计爱好者以及想从事相关工作的读者会有非常大的帮助。对于很多在校的 EECS 学生来说，这也将是一本不可多得的指导用书。

——Synopsys 公司 ARC 处理器高级研发经理 彭剑英

这本书以简练朴素的语言描述了 RISC-V 处理器架构的完整设计实践，其中既有对处理器体系结构简练而不失全面的总结对比，又有 RISC-V 处理器及软硬系统设计的诸多细节。“小小麻雀，五脏俱全”，书中内容反映着作者对本领域的深刻理解和亲身实验，值得学习借鉴。

初识本书作者是在处理器设计架构的研讨会中，他勤勉务实的作风给人留下了深刻的印象，我想也影响着本书的行文和风格，值得学习。

——国防科技大学 教授 姜晶菲

RISC-V 处理器已经在工业界大放异彩，预计不久之后在“微机原理”和“计算机体系结构”课程中讲解 RISC-V 将像在“操作系统”课程中讲解 Linux 一样成为主流。本书作者领导的团队开源了“蜂鸟 E203”处理器，并提供了完整的开发环境，为读者搭建了从 RISC-V 理论学习过渡到工程实践的桥梁。本书不仅深入地诠释了项目代码，而且凝聚了作者多年从事 CPU 设计工作的经验与感悟。对于电子类和计算机类专业的师生、CPU 技术爱好者和从业者来说，本书极具参考价值！

——天津大学电气自动化与信息工程学院 副教授 吕卫

本书介绍的蜂鸟 E203 RISC-V 处理器核，“蜂鸟虽小，五脏俱全”。本书不仅适合作为大

中专院校师生学习 RISC-V 处理器设计（使用 Verilog 语言）的教学或自学案例，而且在 IoT 领域也会大有可为。结合该处理器实例与作者多年的 CPU 设计经验与心得，本书用通俗易懂的语言，为读者轻松揭开了 CPU 设计的神秘面纱，非常适合作为大中专院校师生学习 RISC-V 和 CPU 设计的教学书籍。

——华中科技大学微电子工程系 副教授 郑朝霞

本书非常全面地介绍了 RISC-V 开发所需的知识点，内容丰富，实用性非常强，并且详尽地介绍了 RISC-V 的架构设计和性能优化方面的内容，必将成为 RISC-V 开发者的必备。我打算用本书和蜂鸟 E200 作为学生的课程学习资料，相信大家会有所收获。

——西安邮电大学 高工、博士 焦继业

长期以来，由于商用处理器 IP 核高昂的授权费用和商业保密因素的影响，商用处理器微体系结构总是“秘不示人”。这导致大量学习者只能从框图和体系结构仿真软件上去学习“概念化”的处理器微体系结构，与实战差距较大。而本书作者研发的蜂鸟 E200 开源处理器是中国较早的基于 RISC-V 指令集开发的开源处理器。在本书中，作者以蜂鸟 E200 处理器为例，介绍了从处理器微体系结构到片上系统设计的相关知识。更为可贵的是，作者是以“集成电路设计视角”，而非“计算机系统结构视角”来介绍这些知识的，这样的知识组织结构更加符合集成电路设计从业人员的学习习惯和思考方式。因此，这本书对于从事数字集成电路设计的工程师、高校教师以及相关专业的学生是一本难得的参考书籍。

——电子科技大学电子科学与工程学院 副教授 黄乐天

有幸同本书作者共事过一年，每每被他的知识面之广博所折服。今日读到此书，更加佩服！RISC-V 在需要低功耗和可配置性的场合有强大的生命力，在 IoT 领域以及高性能定制（例如 AI 加速）芯片领域即将大放异彩，此书来得非常及时！如果您是硬件背景的工程师，此书可以让您快速上手 RISC-V，增强自身的竞争力。如果您是软件背景的工程师，此书轻松易懂，可以替代 *Computer Systems: A Programmer's Perspective*（《深入理解计算机系统》）一书，让您了解计算机系统的底层是如何工作的。

——北京比特大陆 资深芯片和 CPU 设计专家 王逵

2018 年，在 meltdown 和 spectre 两大芯片设计的漏洞爆发之后，我愈加感觉到一个程序员了解和理解处理器的设计原理和工作机制是多么的必要。这是一本能让你学习到上述知识的好书，所有程序员都应该读一读。

——《奔跑吧 Linux 内核》作者 笨叔叔

序 一

芯片，是整个电子信息产业的基石。目前，全球半导体市场规模达 3200 亿美元，全球 54% 的芯片都出口到了中国，但国产芯片的市场份额只占 10%。中国芯片产业每年进口需要消耗 2000 多亿美元外汇，超过了石油和大宗商品，在进口商品中占有相当大的比重。

CPU 作为芯片的“心脏”，可谓“芯中之心”，国内的产业实力在此方面一直比较薄弱。CPU 实现国产自主化对我国的发展至关重要，但是 CPU 的主流指令集架构（譬如 x86 和 ARM）一直为国外公司所垄断，国内公司需要支付高昂的专利费用且受制于人。CPU 作为一种特殊的芯片，其要求指令集架构具有普世的通用性且能够共享生态系统，因此囿于一国范围内发明一套封闭的指令集并不具备实用性，必须走与世界主流架构接轨的道路。在这种背景下，开放的 RISC-V 架构给中国 CPU 芯片产业的发展带来了巨大的战略机遇，有希望彻底实现 CPU 的国产自主化和架构主流化。

目前，我国正处于大力发展芯片设计行业的关键时期，实现中华民族伟大复兴的重任需要广大科研和工程工作者孜孜不倦地努力与拼搏，需要很多像作者这样求真务实的技术中坚力量来担负起国产芯片振兴的重任。而国内 CPU 领域人才的奇缺是长期制约行业发展的主要因素，本书作者作为一名长期工作在一线的资深 CPU 设计专家，将其经验撰写成书，资料翔实，文字生动。配合作者所在公司开发的蜂鸟 E200 系列处理器核作为实例，非常适合用于教学领域以及爱好者学习，对于普及 CPU 的设计技术具有十分积极的意义。

新兴的 RISC-V 架构在全球范围内已经掀起了一场热潮，在国内也引起了广泛的关注，但是由于没有很好的中文普及书籍，很多人对于 RISC-V 仍然是“只闻其声，未见其形”。作者作为国内第一批接触 RISC-V 架构，并最早研发成功 RISC-V 处理器的技术专家，在工作之余将其自研的处理器核开源，并著书详细解读其实现细节，体现了作者极高的专业水准和推进国产 CPU 产业发展的炽热情怀。

本书以极为通俗易懂的语言对 RISC-V 架构进行了系统而全面地介绍，并且结合蜂鸟 E200 系列开源处理器核对 CPU 设计技术进行了深入浅出的讲解，图文并茂，生动活泼，体现了作者深厚的专业技能以及将专业知识进行通俗化表述的优秀能力。令人印象深刻的是，本书作者在对 RISC-V 架构进行介绍的过程中，加入了大量的背景知识解读以及个人注解，使得枯燥的专业知识变得非常易于理解，可以说是难能可贵。这是一本凝聚了作者多年所学之精心之作，非常值得一读，对于 RISC-V 架构在国内的传播也将具有巨大的推动作用。本书作为国内不可多得的介绍 RISC-V 的中文书籍，相信一定会成为该领域的经典之作。

邹雪城

华中科技大学武汉国际微电子学院执行院长、教授
武汉集成电路设计工程技术研究中心主任

2018 年初 于武汉

序 二

近 40 年来，通用计算的两次跨时代的飞跃都来自硬件系统的逐步开放。以 PC 为终端的有线互联网技术完成了地址与地址之间的通信，让数据真正实现了大量传输与交互。而这一切源于美国 Intel 公司发明 x86 芯片，并将芯片提供给各大计算机制造厂商按照开放硬件标准组装，使 PC 大批量制造成为可能。PC 的普及开创了计算机互联网的时代，互联网时代开始之初的用户数量是 5 亿（2003 年）。以手机为终端的移动互联网实现了人与人之间的通信，让终端个人直接连接到网络。这一切都是因为英国 ARM 公司将 CPU 的 IP 授权给手机芯片厂商以 SoC 片上系统的形式研制自己的芯片，使智能手机得以大量普及，同时开创了移动互联网时代，移动互联网时代开始之初的用户数量是 25 亿（2009 年）。

未来是物联网的时代，那时的用户数量会是多少呢？有分析称，2020 年将会达到 300 亿，2050 年极保守估算将是 1000 亿以上。美国公司通过完成硬件开放标准化，研发芯片并将其出售，完成了 5 亿用户普及；英国公司通过完成芯片标准化，并将其芯片 IP 有限的授权，完成了 25 亿用户的推广。而面对一个上百亿用户推广的目标，数以千万计的不同应用需求，必须要有更开放的形式吸引更多的力量参与，而这个答案显而易见就是——开源。只有开源，才能降低进入门槛，加快技术迭代速度，实现技术普及，满足市场需求。开源、共享、共赢，可以说在物联网时代，开源的硬件是一种必需，没有基础芯片技术的进一步开源，就没有物联网应用的未来。

而且，在摩尔定律减缓的今天，一味比拼硬件性能的技术竞赛变得越发艰难。然而性能提升的最终目的是满足应用，如何在现有的能力下最大程度地满足数以千万计的不同应用的需求，就成了当今处理器行业要面对的重要问题。RISC-V 站在了时代的风口，其作为一种开源架构的出现必将对芯片产业产生深远的影响。硬件芯片的开源不再是一个噱头，而是变成了一种刚需。蜂鸟 E200 作为中国本土较早开源的 RISC-V 芯片，也将对国内的相关产业发展起到巨大的推动作用，为国内公司抓住物联网风口的大发展而助力。开源即透明，透明即可控。自主可控也是国家信息安全的保障，由国内公司自主研发并开源的 RISC-V 处理器，也为国家信息安全领域的应用提供了多一种选择。

这本书基于蜂鸟 E200，全面而详细地介绍了 RISC-V 处理器，分四大部分讲解了处理器和指令集架构的基本知识与背景、处理器设计要诀、软件运行实例以及 RISC-V 附录资料，是国内 RISC-V 开源处理器社区极其稀缺的教材，可以说是 RISC-V 芯片版的《鸟哥的 Linux 私房菜》。更难能可贵的是，书中作者不但深入浅出地讲解了处理器相关的几乎全部的基本

2 | 手把手教你设计 CPU——RISC-V 处理器篇

概念和功能设计要点，而且辅以大量实例与解析，是初学者在 RISC-V 和处理器设计领域推门入室的不二之选。

我推荐这本书，并不仅仅因为它是一本好的技术教材，更是因为它可以为更多人开启开源硬件、芯片世界的大门。开源的本质是技术合作和知识的自由传播。知识只有传播才有价值，而开源就是让知识能在广泛群体中传播，并凝聚这个群体的力量来推动相应技术的发展。开源精神可能是一种理想主义，而这种理想主义也是有其现实意义的，尤其是在著作权和专利构成技术进步壁垒下的今天，创新需要这样的理想主义来推动，而理想主义的世界需要优秀的作品来开启。

期待这本优秀的教材能为读者打开理想主义世界的大门！

郑云龙

中国科学院博士后

九天微星技术总监

2018 年初 于北京

前言

永恒的热点——CPU

灯，等灯等灯……

——Intel

如果要评选过去十数年间经典的科技广告音乐，想必 Intel 的广告音乐“灯，等灯等灯……”会榜上有名。而熟悉的蓝色贴标“Intel Inside”，也必是辨识度很高的广告图标。

中央处理单元（Center Processor Unit, CPU），虽然早已被大众所熟知，但在相当长的时间内，它一直是高端大气的代名词。也许很多读者都和作者有着相同的印象，在早期的电脑杂志上，必然有着浓墨重彩的篇幅与专栏，详细介绍 Intel 或 AMD 公司推出的 CPU 芯片的详细参数，抑或 MIPS 与 ARM 的优劣之处，甚至是令人津津乐道的各家 CPU 的小道消息与幕后桥段。

而 CPU 代表的高端技术，也一直笼罩在神秘的面纱之下。当作者还是一名学生的时候，每当浏览 CPU 的论坛与探讨文章，总能看到其作为热门版块吸引着众多的拥护者，并拥有极高的下载量。当我结束了多年的校园生活，开始选择人生第一份工作时，正逢某知名外企到中国设立 CPU 研发部门。经过激烈的竞争，我有幸成为了一名 CPU 逻辑设计工程师。而后多年间，我先后供职于多家国际一流公司的 CPU 设计部门，参与过多次校园招聘和社会招聘，CPU 设计部门总能以其光环吸引众多的应聘者前来竞争角逐。就像每个男孩心中都有一个军人梦一样，似乎每一个芯片设计人员都有一个 CPU 梦。

当今世界，科学技术正在以令人惊异的速度飞快发展，从 IoT（物联网）到大数据再到人工智能，新技术与新领域如雨后春笋般层出不穷。而 CPU 这门诞生于 20 世纪 60~70 年代的技术，是否已经垂垂老矣，应该退出历史舞台了呢？其实不然，从 IoT 的超低功耗微控制器到大数据的高性能计算，再到人工智能的异构计算，CPU 均扮演着核心的角色。可以说，在未来相当长的时间内，从指尖的超低功耗处理器到云端的超级计算机，CPU 技术都将持续站在前沿科技的风口浪尖，散发着它历久弥新的魅力。

然而长期以来，CPU 架构主要由以 Intel（x86 架构）和 ARM（ARM 架构）为代表的商业巨头公司所掌控，成为普通公司与个人无法逾越的天堑。2016 年，RISC-V 基金会成立，开放免费的 RISC-V 架构。这一举动具有划时代的意义，任何公司与个人均可依据开放的 RISC-V 架构设计自己所需的处理器，极大地降低了 CPU 的准入门槛。因此，RISC-V 架构在极短的时间内便引起了业界的高度关注，从众多反应快速的小公司到实力雄厚的巨头公司（如 NVIDIA、三星等）均开始使用 RISC-V 架构开发产品。“旧时王谢堂前燕，飞入寻常百姓家”，在摩尔定

律逐步逼近极限的今天，开放且免费的 RISC-V 架构的诞生，将催生出新一轮的创新热潮。

当前国内 CPU 产业方兴未艾，x86、ARM 和 MIPS 等传统商用处理器架构呈现“百花齐放”之势。龙芯、兆芯、飞腾等资深专业 CPU 公司在不断突破，华为、展讯等一线大公司相继开始研发自主的处理器核心，海光、华芯通等新锐也开始摩拳擦掌。此时，开放的 RISC-V 架构的诞生，更是锦上添花。可以说，学习 CPU 设计正当时，学习 RISC-V 正当时！

作者在培训新入门工程师，或与爱好者、学生交流的过程中，能够感到 CPU 设计这门要求计算机体系结构和软硬件皆通的技术令初学者难以学习和掌握，作者时常遗憾于没有很好的通俗读本。正所谓“曲高和寡，妙伎难工”，CPU 设计过于专业，相关的书籍或卷帙浩繁，或晦涩难懂，令初学者不知从何下手，且难以理解。在实际的练习中，也难以找到易于学习和上手的例程，更别说完整地设计一款处理器了。

如上种种，正是促使作者撰写此书的原因。作者希望本书能够作为一本通俗读本，帮助初学者和爱好者顺利越过初期的陡峭学习曲线，进入 CPU 设计的坦途。谨以此书献给曾经帮助过作者的良师益友、合作伙伴和默默工作的工程师们！

本书内容

Stay hungry, Stay foolish.
(求知似饥，虑心若愚。)

—— Steven Jobs (史蒂夫·乔布斯)

- 您是否想学习工业级 Verilog RTL 数字 IC 设计的精髓与技巧？
- 您是否阅读了众多计算机体系结构的书籍仍不明就里？
- 您是否想揭开 CPU 设计神秘的面纱，并亲自设计一款处理器？
- 您是否想学习国际一流公司真实的 CPU 设计案例？
- 您是否想用最短的时间熟悉并掌握 RISC-V 架构？
- 您是否想深入理解并使用一款免费可靠的开源 RISC-V 处理器和完整的 SoC 平台？

如果您对上述任意一个问题感兴趣，本书都将是您很好的选择。

作者所在公司的团队，总结各国际一流公司多年从事 CPU 设计工作的丰富经验，开发了一款超低功耗 RISC-V 处理器（蜂鸟 E200），也是一款开源的 RISC-V 处理器。

结合该处理器实例与作者多年的 CPU 设计经验与心得，本书将用通俗易懂的语言，深入浅出地剖析 RISC-V 处理器的微架构以及代码实现，为读者揭开 CPU 设计的神秘面纱，打开计算机体系结构的大门。

本书旨在成为国内第一本系统介绍 RISC-V 指令集架构的通俗读本，以及第一本结合实际 RISC-V 开源实例进行教学的技术图书。相信通过对本书的学习，读者能够快速掌握并轻松使用 RISC-V 架构处理器。

通过学习实例蜂鸟 E200 的 Verilog 代码，您将能成为一名合格的数字 IC 设计工程师；通过

学习本书推荐的完整开源 SoC 平台,您也可以快速搭建 FPGA 原型平台,运行完整的软件实例。

希望本书能够为科普 RISC-V 指令集架构起到推动作用,同时通过对“蜂鸟 E200 处理器”的开源与解析,为 RISC-V 处理器在国内的普及贡献绵薄之力。本书共分四部分,各部分主要内容如下。

第一部分是 CPU 与 RISC-V 综述,包括第 1~4 章。该部分将介绍 CPU 的一些基础背景知识、RISC-V 架构的诞生和特点。

第 1 章主要介绍 CPU 的基础知识、指令集架构的历史、国产 CPU 的发展现状及原因、CPU 的应用领域、各领域的主流架构、RISC-V 的诞生背景等。

第 2 章主要介绍 RISC-V 架构和特点,着重分析其大道至简的设计哲学,并阐述 RISC-V 和以往曾经出现过的开放架构有何不同。

第 3 章主要对当前全球范围内的商业或者开源 RISC-V 处理器进行盘点,分析其优缺点,并引出开源的 RISC-V 处理器——蜂鸟 E200 系列处理器核和 SoC。

第 4 章主要对蜂鸟 E200 系列处理器核和 SoC 的特性进行整体介绍。

第二部分主要讲解如何使用 Verilog 设计 CPU,包括第 5~16 章。该部分将对蜂鸟 E200 处理器核的微架构和源代码进行深度剖析,结合该处理器核进行处理器设计案例分析。

第 5 章主要先从宏观的角度着手,介绍若干处理器设计的技巧、蜂鸟 E200 处理器核的总体设计思想和顶层接口。帮助读者整体认识蜂鸟 E200 处理器的设计要诀,为后续各章针对不同部分展开详述奠定基础。

第 6 章主要介绍处理器的一些常见流水线结构,并介绍蜂鸟 E200 处理器核的流水线结构。

第 7 章主要介绍处理器的取指功能,并介绍蜂鸟 E200 处理器核取指单元的微架构和源码分析。

第 8 章主要介绍处理器的执行功能,并介绍蜂鸟 E200 处理器核执行单元的微架构和源码分析。

第 9 章主要介绍处理器交付的功能和常见策略,并介绍蜂鸟 E200 处理器核交付单元的微架构和源码分析。

第 10 章主要介绍处理器的写回功能和常见策略,并介绍蜂鸟 E200 处理器核的写回硬件实现微架构和源码分析。

第 11 章主要介绍处理器的存储器架构,并介绍蜂鸟 E200 处理器核存储器子系统的微架构和源码分析。

第 12 章主要介绍蜂鸟 E200 处理器核的总线接口模块,介绍其使用的总线协议,以及该模块的微架构和源码分析。

第 13 章主要介绍 RISC-V 架构定义的中断和异常机制,蜂鸟 E200 处理器核中断和异常的硬件微架构及其源码分析。

第 14 章主要介绍处理器的调试机制,介绍 RISC-V 架构定义的调试方案、蜂鸟 E200 处理器调试机制的硬件实现微架构和源码分析。

第 15 章主要介绍处理器的低功耗技术，并以蜂鸟 E200 处理器为例阐述其低功耗设计的诀窍。

第 16 章主要介绍如何利用 RISC-V 的可扩展性，并以蜂鸟 E200 的协处理器接口为例详细阐述如何定制一款协处理器。

第三部分是使用 Verilog 进行仿真和在 FPGA SoC 原型上运行软件，包括第 17~20 章。该部分将对蜂鸟 E200 配套 SoC 的软硬件平台进行剖析，并详细讲解如何进行 Verilog 仿真测试，如何在 FPGA 原型上运行软件示例程序和使用 GDB 进行调试。想快速使用蜂鸟 E200 的读者可以跳过第二部分，直接阅读第三部分内容。

第 17 章主要介绍蜂鸟 E200 开源平台如何运行 Verilog 仿真测试。

第 18 章主要介绍蜂鸟 E200 处理器配套的 SoC 如何在 FPGA 上实现该 SoC 原型。

第 19 章主要介绍如何使用 SoC 的 FPGA 原型平台运行真正的软件示例、如何使用 GDB 对程序进行调试。

第 20 章主要介绍如何使用 SoC 的 FPGA 原型平台运行跑分程序，对蜂鸟 E200 处理器核的性能进行量化评估。

第四部分是附录部分，包括附录 A~附录 G。该部分将对 RISC-V 指令集架构进行详细介绍，对 RISC-V 指令集架构细节感兴趣的读者可以先行阅读附录部分。

附录 A 主要介绍 RISC-V 架构的指令集。该附录翻译自 RISC-V 的“指令集文档”，并对相关内容进行了重新组织，以求通俗易懂。

附录 B 主要介绍 RISC-V 架构的 CSR 寄存器。该附录对于 CSR 寄存器的介绍翻译自 RISC-V 的“特权架构文档”，同时还介绍了蜂鸟 E200 处理器核自定义的 CSR 寄存器。

附录 C 主要介绍 RISC-V 架构定义的系统平台中断控制器（Platform Level Interrupt Controller, PLIC）。该附录对于 PLIC 的介绍翻译自 RISC-V 的“特权架构文档”。

附录 D 主要介绍存储器模型（Memory Model）的相关背景知识，帮助读者更深入地理解 RISC-V 架构的存储器模型。

附录 E 主要结合多线程“锁”的示例对存储器原子操作指令的应用背景进行简介。

附录 F 和附录 G 分别是 RISC-V 指令的编码列表和 RISC-V 伪指令的列表。附录均节取自 RISC-V 的“指令集文档”，供读者快速查阅。

建议与反馈

由于时间仓促且作者水平有限，书中难免存在不足之处，敬请各位读者批评指正。相关问题请与本书编辑（zhangshuang@ptpress.com.cn）联系交流，或到异步社区（<https://www.epubit.com/>）中提交勘误。

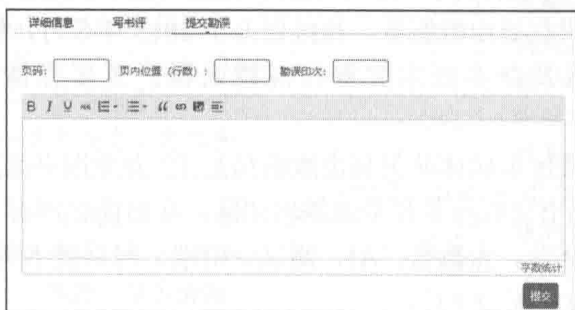
资源与支持

本书为异步社区出品的图书，在社区（<https://www.epubit.com/>）上为您提供以下服务。

提交勘误

作者和编辑尽最大努力来确保书中内容的准确性，但难免还会存在差错。欢迎您将发现的问题告诉我们，帮助我们提升图书的质量。

当您发现错误时，请登录异步社区主页 <https://www.epubit.com/>，搜索到本书页面，点击“提交勘误”，输入相应信息，最后点击“提交”按钮即可。之后本书的作者和编辑会对您提交的勘误进行审核。确认并接受后，您将获赠异步社区的 100 积分。积分可用于在社区兑换优惠券，以及兑换样书或奖品之用。



扫码关注本书

请扫描下方二维码关注本书，即可在异步社区微信服务号中看到本书和进一步的服务信息。



与我们联系

我们的联系邮箱是 contact@epubit.com.cn。

如果您对本书有任何疑问或建议，请发邮件到此邮箱，邮件标题中请注明本书书名。

如果您有兴趣出版图书、录制教学视频，或参与图书翻译、技术审校等工作，可以发邮件，或者到异步社区在线提交投稿：

<https://www.epubit.com/selfpublish/submission>

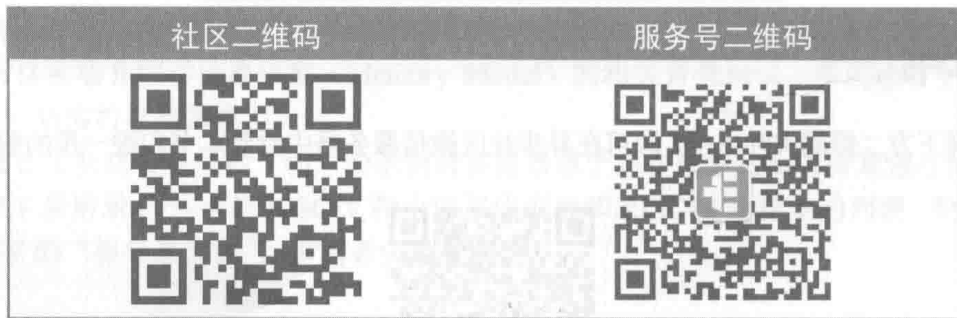
如果您是学校、培训机构或企业，想批量购买本书或异步社区出版的其他图书，请发邮件联系我们。

如果您在网上发现针对异步图书的各种形式的盗版行为，包括图书或部分内容的非授权传播，请您将怀疑有侵权行为的链接发邮件给我们。您的举动是对作者权益的保护，我们由此才能继续为您带来有价值的内容。

关于异步社区和异步图书

异步社区是人民邮电出版社旗下 IT 专业图书社区，致力于出版精品 IT 技术图书和相关学习产品，为作译者提供优质出版服务，社区创办于 2015 年 8 月，提供超过 1000 种图书、近 1000 种电子书，以及众多技术文章和视频课程。更多详情请访问异步社区官网 <https://www.epubit.com>。

异步图书是由异步社区编辑团队策划出版的精品 IT 专业图书品牌，依托于人民邮电出版社近 30 年的计算机图书出版积累和专业编辑团队，在封面上印有异步图书的 LOGO。我们的出版领域包括软件开发、大数据、AI、测试、前端、网络技术等。



目 录

第一部分 CPU 与 RISC-V 综述

第 1 章 一文读懂 CPU 之三生三世·····2

- 1.1 眼看他起高楼，眼看他宴宾客，眼看他楼塌了——CPU 众生相·····3
 - 1.1.1 ISA——CPU 的灵魂·····4
 - 1.1.2 CISC 与 RISC·····5
 - 1.1.3 32 位与 64 位架构·····6
 - 1.1.4 ISA 众生相·····6
 - 1.1.5 CPU 的领域之分·····10
- 1.2 ISA 请扛起这口锅——为什么国产 CPU 尚未足够成功·····12
 - 1.2.1 MIPS 系——龙芯和君正·····12
 - 1.2.2 x86 系——北大众志、兆芯和海光·····13
 - 1.2.3 Power 系——中晟宏芯·····13
 - 1.2.4 Alpha 系——申威·····14
 - 1.2.5 ARM 系——飞腾、华为海思、展讯和华芯通·····14
 - 1.2.6 背锅侠 ISA·····15
- 1.3 人生已是如此艰难，你又何必拆穿——CPU 从业者的无奈·····17
- 1.4 无敌是多么寂寞——ARM 统治着的世界·····18
 - 1.4.1 独乐乐与众乐乐——ARM 公司的盈利模式·····18
 - 1.4.2 小个子有大力量——无处不在的 Cortex-M 系列·····21
 - 1.4.3 移动王者——Cortex-A 系列在手持设备领域的巨大成功·····23
 - 1.4.4 进击的巨人——ARM 进军 PC 与服务器领域的雄心·····25
- 1.5 东边日出西边雨，道是无晴却有晴——RISC-V 登场·····25

- 1.6 原来你是这样的“薯片”——ARM 的免费计划·····28
- 1.7 旧时王谢堂前燕，飞入寻常百姓家——你也可以设计自己的处理器·····28

第 2 章 大道至简——RISC-V 架构之魂·····29

- 2.1 简单就是美——RISC-V 架构的设计哲学·····30
 - 2.1.1 无病一身轻——架构的篇幅·····30
 - 2.1.2 能屈能伸——模块化的指令集·····32
 - 2.1.3 浓缩的都是精华——指令的数量·····32
- 2.2 RISC-V 指令集架构简介·····33
 - 2.2.1 模块化的指令子集·····33
 - 2.2.2 可配置的通用寄存器组·····34
 - 2.2.3 规整的指令编码·····34
 - 2.2.4 简洁的存储器访问指令·····34
 - 2.2.5 高效的分支跳转指令·····35
 - 2.2.6 简洁的子程序调用·····36
 - 2.2.7 无条件码执行·····37
 - 2.2.8 无分支延迟槽·····37
 - 2.2.9 零开销硬件循环·····38
 - 2.2.10 简洁的运算指令·····38
 - 2.2.11 优雅的压缩指令子集·····39
 - 2.2.12 特权模式·····40
 - 2.2.13 CSR 寄存器·····40
 - 2.2.14 中断和异常·····40
 - 2.2.15 矢量指令子集·····40
 - 2.2.16 自定义指令扩展·····41
 - 2.2.17 总结与比较·····41
- 2.3 RISC-V 软件工具链·····42
- 2.4 RISC-V 和其他开放架构有何不同·····44

2 | 手把手教你设计 CPU——RISC-V 处理器篇

2.4.1 平民英雄——OpenRISC	44	3.1.9 Andes Core (商业 IP)	52
2.4.2 豪门显贵——SPARC	44	3.1.10 Microsemi Core (商业 IP) ...	52
2.4.3 名校优生——RISC-V	45	3.1.11 CodaSip Core (商业 IP) ...	53
第 3 章 乱花渐欲迷人眼——盘点 RISC-V		3.1.12 蜂鸟 E200 Core 与 SoC	
商业版本与开源版本	46	(开源)	53
3.1 各商业版本与开源版本综述	47	3.2 总结	53
3.1.1 Rocket Core (开源)	47	第 4 章 开源 RISC-V——蜂鸟 E200 系列	
3.1.2 BOOM Core (开源)	49	超低功耗 Core 与 SoC	54
3.1.3 Freedom SoC (开源)	50	4.1 与众不同的蜂鸟 E200 处理器	55
3.1.4 LowRISC SoC (开源)	50	4.2 蜂鸟 E200 简介——蜂鸟虽小，	
3.1.5 PULPino Core and SoC		五脏俱全	56
(开源)	50	4.3 蜂鸟 E200 型号系列	57
3.1.6 PicoRV32 Core (开源)	51	4.4 蜂鸟 E200 性能指标	58
3.1.7 SCR1 Core (开源)	51	4.5 蜂鸟 E200 配套 SoC	59
3.1.8 ORCA Core (开源)	51	4.6 蜂鸟 E200 配置选项	60

第二部分 手把手教你使用 Verilog 设计 CPU

第 5 章 先见森林,后观树木——蜂鸟 E200		5.8 蜂鸟 E200 处理器流水线结构	74
设计总览和顶层介绍	65	5.9 蜂鸟 E200 处理器核顶层接口介绍 ...	74
5.1 处理器硬件设计概述	66	5.10 总结	77
5.1.1 架构和微架构	66	第 6 章 流水线不是流水账——蜂鸟 E200	
5.1.2 CPU、处理器、Core 和		流水线介绍	78
处理器核	66	6.1 处理器流水线概述	79
5.1.3 处理器设计和验证的特点 ...	66	6.1.1 从经典的五级流水线说起 ...	79
5.2 蜂鸟 E200 处理器核设计哲学	67	6.1.2 可否不要流水线——流水线和	
5.3 蜂鸟 E200 处理器核 RTL 代码风格		状态机的关系	81
介绍	68	6.1.3 深处种菱浅种稻,不深不浅种	
5.3.1 使用标准 DFF 模块例化生成		荷花——流水线的深度	81
寄存器	68	6.1.4 向上生长——越来越深的	
5.3.2 推荐使用 assign 语法替代		流水线	82
if-else 和 case 语法	70	6.1.5 向下生长——越来越浅的	
5.3.3 其他若干注意事项	71	流水线	83
5.3.4 小结	72	6.1.6 总结	83
5.4 蜂鸟 E200 模块层次划分	72	6.2 处理器流水线中的乱序	83
5.5 蜂鸟 E200 处理器核源代码	73	6.3 处理器流水线中的反压	84
5.6 蜂鸟 E200 处理器核配置选项	73	6.4 处理器流水线中的冲突	84
5.7 蜂鸟 E200 处理器核支持的 RISC-V		6.4.1 流水线中的资源冲突	84
指令子集	74	6.4.2 流水线中的数据冲突	85

6.5	蜂鸟 E200 处理器的流水线	86	8.1.7	小结	121
6.5.1	流水线总体结构	86	8.2	RISC-V 架构特点对于执行的简化	121
6.5.2	流水线中的冲突	87	8.2.1	规整的指令编码格式	122
6.6	总结	87	8.2.2	优雅的 16 位指令	122
第 7 章	万事开头难吗——一切从取指令开始	88	8.2.3	精简的指令个数	122
7.1	取指概述	89	8.2.4	整数指令都是两操作数	122
7.1.1	取指特点	89	8.3	蜂鸟 E200 处理器的执行实现	123
7.1.2	如何快速取指	90	8.3.1	执行指令列表	123
7.1.3	如何处理非对齐指令	91	8.3.2	EXU 总体设计思路	123
7.1.4	如何处理分支指令	92	8.3.3	译码	124
7.2	RISC-V 架构特点对于取指的简化	97	8.3.4	整数通用寄存器组	130
7.2.1	规整的指令编码格式	97	8.3.5	CSR 寄存器	133
7.2.2	指令长度指示码放于低位	97	8.3.6	指令发射派遣	134
7.2.3	简单的分支跳转指令	98	8.3.7	流水线冲突、长指令和 OITF	139
7.2.4	没有分支延迟槽指令	100	8.3.8	ALU	145
7.2.5	提供明确的静态分支预测依据	100	8.3.9	高性能乘除法	157
7.2.6	提供明确的 RAS 依据	101	8.3.10	浮点单元	158
7.3	蜂鸟 E200 处理器的取指实现	101	8.3.11	交付	159
7.3.1	IFU 总体设计思路	102	8.3.12	写回	159
7.3.2	Mini-Decode	103	8.3.13	协处理器扩展	160
7.3.3	Simple-BPU 分支预测	105	8.3.14	小结	160
7.3.4	PC 生成	109	第 9 章	善始者实繁，克终者盖寡——交付	161
7.3.5	访问 ITCM 和 BIU	111	9.1	处理器交付、取消、冲刷	162
7.3.6	ITCM	115	9.1.1	处理器交付、取消、冲刷简介	162
7.3.7	BIU	116	9.1.2	处理器交付常见实现策略	163
7.4	总结	116	9.2	RISC-V 架构特点对于交付的简化	164
第 8 章	一鼓作气，执行力是关键——执行	117	9.3	蜂鸟 E200 处理器交付硬件实现	164
8.1	执行概述	118	9.3.1	分支预测指令的处理	165
8.1.1	指令译码	118	9.3.2	中断和异常的处理	168
8.1.2	指令执行	118	9.3.3	多周期执行指令的交付	169
8.1.3	流水线的冲突	119	9.3.4	小结	169
8.1.4	指令的交付	119	第 10 章	让子弹飞一会儿——写回	170
8.1.5	指令发射、派遣、执行、写回的顺序	119	10.1	处理器的写回	171
8.1.6	分支解析	121	10.1.1	处理器写回功能简介	171