

原书第3版

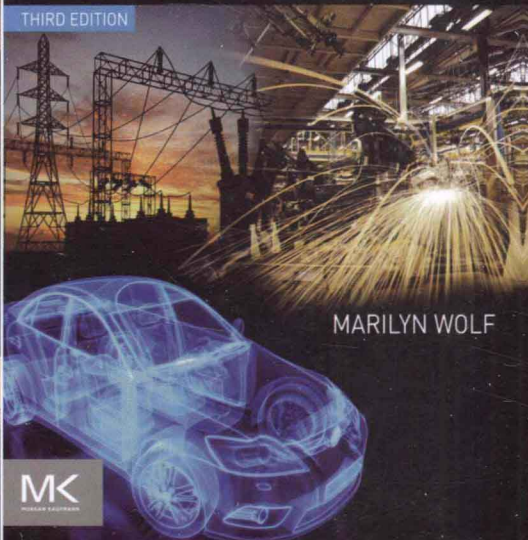
嵌入式计算系统设计原理

(美) Marilyn Wolf 著 李仁发 等译
佐治亚理工学院 湖南大学

Computers as Components
Principles of Embedded Computing System Design Third Edition

COMPUTERS
AS COMPONENTS
PRINCIPLES OF EMBEDDED COMPUTING SYSTEM DESIGN

THIRD EDITION



MK
Morgan Kaufmann



机械工业出版社
China Machine Press

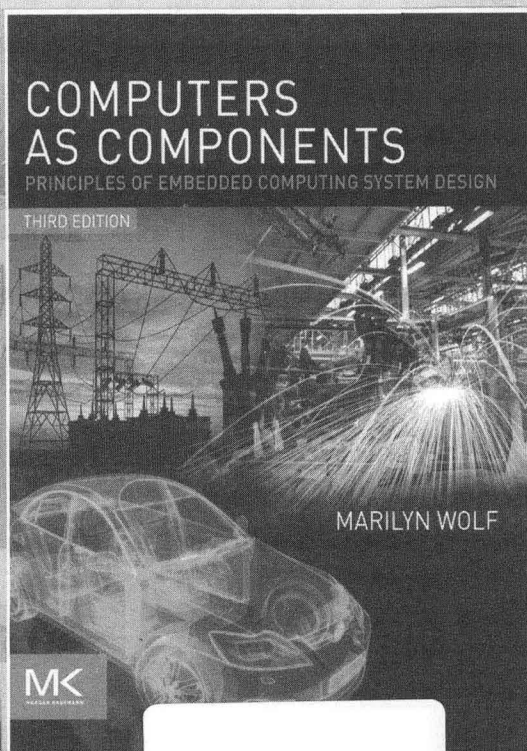
计 算 机 科 学 丛

原书第3版

嵌入式计算系统设计原理

(美) **Marilyn Wolf** 著 李仁发 等译
佐治亚理工学院 湖南大学

Computers as Components
Principles of Embedded Computing System Design Third Edition



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

嵌入式计算系统设计原理 (原书第3版) / (美) 沃尔夫 (Wolf, M.) 著; 李仁发等译. —北京: 机械工业出版社, 2013. 7

(计算机科学丛书)

书名原文: Computers as Components: Principles of Embedded Computing System Design, Third Edition

ISBN 978-7-111-44075-8

I. 嵌… II. ①沃… ②李… III. 微型计算机—系统设计 IV. TP36

中国版本图书馆 CIP 数据核字 (2013) 第 221700 号

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问 北京市展达律师事务所

本书版权登记号: 图字: 01-2012-7898

Marilyn Wolf: Computers as Components: Principles of Embedded Computing System Design, Third Edition (ISBN 978-0-12-388436-7).

Copyright © 2012 by Elsevier Inc. All rights reserved.

Authorized Simplified Chinese translation edition published by the Proprietor.

Copyright © 2014 by Elsevier (Singapore) Pte Ltd. All rights reserved.

Printed in China by China Machine Press under special arrangement with Elsevier (Singapore) Pte Ltd. This edition is authorized for sale in China only, excluding Hong Kong SAR, Macau SAR and Taiwan. Unauthorized export of this edition is a violation of the Copyright Act. Violation of this Law is subject to Civil and Criminal Penalties.

本书简体中文版由 Elsevier (Singapore) Pte Ltd. 授权机械工业出版社在中国大陆境内独家出版和发行。本版仅限在中国境内 (不包括香港特别行政区、澳门特别行政区及台湾地区) 出版及标价销售。未经许可之出口, 视为违反著作权法, 将受法律之制裁。

本书封底贴有 Elsevier 防伪标签, 无标签者不得销售。

本书从组件技术的视角出发, 讲述了嵌入式计算的基本原理和技术。全书每一章涵盖一个专题, 包括与嵌入式系统设计相关的若干主要内容: 指令系统、CPU、计算平台、程序设计与分析、进程和操作系统、系统设计技术以及多处理器和网络等。

本书特别适合作为计算机、电子信息、通信工程、自动化、机电一体化、仪器仪表及相关专业高年级本科生和研究生的教材, 也适合相关的工程技术人员参考。

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: 姚 蕾 刘立卿

北京诚信伟业印刷有限公司印刷

2014 年 1 月第 1 版第 1 次印刷

185mm × 260mm · 19.75 印张

标准书号: ISBN 978-7-111-44075-8

定 价: 69.00 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) 88378991 88361066

投稿热线: (010) 88379604

购书热线: (010) 68326294 88379649 68995259

读者信箱: hzsj@hzbook.com

文艺复兴以降，源远流长的科学精神和逐步形成的学术规范，使西方国家在自然科学的各个领域中取得了垄断性的优势；也正是这样的传统，使美国在信息技术发展的六十多年间名家辈出、独领风骚。在商业化的进程中，美国的产业界与教育界越来越紧密地结合，计算机学科中的许多泰山北斗同时身处科研和教学的最前线，由此而产生的经典科学著作，不仅擘划了研究的范畴，还揭示了学术的源变，既遵循学术规范，又自有学者个性，其价值并不会因年月的流逝而减退。

近年，在全球信息化大潮的推动下，我国的计算机产业发展迅猛，对专业人才的需求日益迫切。这对计算机教育界和出版界都既是机遇，也是挑战；而专业教材的建设在教育战略上显得举足轻重。在我国信息技术发展时间较短的现状下，美国等发达国家在其计算机科学发展的几十年间积淀和发展的经典教材仍有许多值得借鉴之处。因此，引进一批国外优秀计算机教材将对我国计算机教育事业的发展起到积极的推动作用，也是与世界接轨、建设真正的世界一流大学的必由之路。

机械工业出版社华章公司较早意识到“出版要为教育服务”。自1998年开始，我们就将工作重点放在了遴选、移译国外优秀教材上。经过多年的不懈努力，我们与Pearson, McGraw-Hill, Elsevier, MIT, John Wiley & Sons, Cengage等世界著名出版公司建立了良好的合作关系，从他们现有的数百种教材中甄选出Andrew S. Tanenbaum, Bjarne Stroustrup, Brian W. Kernighan, Dennis Ritchie, Jim Gray, Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, Abraham Silberschatz, William Stallings, Donald E. Knuth, John L. Hennessy, Larry L. Peterson等大师名家的一批经典作品，以“计算机科学丛书”为总称出版，供读者学习、研究及珍藏。大理石纹理的封面，也正体现了这套丛书的品位和格调。

“计算机科学丛书”的出版工作得到了国内外学者的鼎力襄助，国内的专家不仅提供了中肯的选题指导，还不辞劳苦地担任了翻译和审校的工作；而原书的作者也相当关注其作品在中国的传播，有的还专程为其书的中译本作序。迄今，“计算机科学丛书”已经出版了近两百个品种，这些书籍在读者中树立了良好的口碑，并被许多高校采用为正式教材和参考书籍。其影印版“经典原版书库”作为姊妹篇也被越来越多实施双语教学的学校所采用。

权威的作者、经典的教材、一流的译者、严格的审校、精细的编辑，这些因素使我们的图书有了质量的保证。随着计算机科学与技术专业学科建设的不断完善和教材改革的逐渐深化，教育界对国外计算机教材的需求和应用都将步入一个新的阶段，我们的目标是尽善尽美，而反馈的意见正是我们达到这一终极目标的重要帮助。华章公司欢迎老师和读者对我们的工作提出建议或给予指正，我们的联系方式如下：

华章网站：www.hzbook.com

电子邮件：hzjsj@hzbook.com

联系电话：(010) 88379604

联系地址：北京市西城区百万庄南街1号

邮政编码：100037



华章教育

华章科技图书出版中心

译者序

Computers as Components: Principles of Embedded Computing System Design, Third Edition

嵌入式计算系统的应用在日常生活中已无所不在，嵌入式系统的开发者遍布世界，嵌入式计算系统的研究在广度和深度上日益增加。

《Computers as Components: Principles of Embedded Computing System Design》是第一本系统讲述嵌入式计算这门学科的书，其第1版和第2版已经被全球多所高校的嵌入式系统相关课程采用。

在第3版中，作者针对嵌入式计算应用大幅扩展的现状，相较于前一版，对内容进行了重新组织和更新：增加了数码相机和汽车应用相关的全新嵌入式设计示例；新增了 TI C64X 和先进的 ARM 芯片等处理器相关内容，尤其在阐释小型 RISC 嵌入式处理器时介绍了 PIC16F；重组了网络和多处理器等与主题紧密相关的内容。全书每一章涵盖一个专题，包括与嵌入式系统设计相关的若干主要内容：指令系统、CPU、计算平台、程序设计与分析、进程和操作系统、系统设计技术以及多处理器和网络等。

本书内容丰富，文字通俗流畅，讲述深入浅出，配合了丰富的设计示例与编程示例，使得读者在进行理论学习的同时，能够较容易地联系实际，加深对嵌入式计算设计思想的理解，并获得先进的技术实践指导。

本书特别适合作为计算机、电子信息、通信工程、自动化、机电一体化、仪器仪表及相关专业高年级本科生和研究生的教材，也适合相关的工程技术人员参考。

参与本书翻译工作的有：连佩佩（第1章）、张欢（第2章）、晏巍（第3章）、凌纯清（第4章）、王欢欢（第5章）、黄丽达（第6章）、马萌（第7章）、肖玲（第8章），术语由凌纯清、黄丽达和肖玲共同翻译完成，许莹、肖竹、何晶、蒋斌和杨超等参与了校对工作。李仁发教授统筹了全书的翻译，组织、汇总与校订由黄丽达负责。徐成教授给予了大量的专业指导，并审阅了全书。

由于时间与水平所限，译稿中难免存在不妥或错误之处，敬请读者指正。

译者

2013年7月

数字系统设计已经进入一个新的时代。当微处理器设计转向一种典型的优化工作时，只把微处理器作为其部件的嵌入式计算系统设计已经成为一种广阔的科技前沿。无线系统、可穿戴系统、网络系统、智能家电、工业过程系统、先进汽车系统以及生物接口系统由于与这个新领域产生了交叉而提供了一些例子。

受传感器、转换器、微电子学、处理器性能、操作系统、通信技术、用户接口和封装技术进步的推动以及对于人类的需要和市场潜能更深入的理解，正在涌现出大量新的系统和应用。现在系统设计师和嵌入式系统设计人员的责任是把这些可能变成现实。

然而，在现阶段，嵌入式系统设计仍处于一种手工阶段。虽然有关硬件组件和软件子系统的知识是很清楚的，但是还没有协调整个设计过程的通用系统设计方法，在大多数项目中嵌入式系统的设计仍然采用某种特定方法。

嵌入式系统设计所面临的某些挑战源于基础技术的改变以及系统各部件如何能全部正确地混合和集成在一起的细微之处，另一些挑战源于新的并且常常是不熟悉的系统需求。此外，用于通信和协作的基础设施和技术的改善已经使快速设计以响应市场需求呈现出前所未有的可能性。但是，还没有有效的设计方法和相关的设计工具足以迅速应付这些挑战。

在 VLSI 时代的开始阶段，晶体管和导线是基本部件，快速设计基于芯片的计算机是一种梦想。今天，CPU 和各种特殊处理器以及子系统仅是一些基本部件，但快速、高效地设计极复杂的嵌入式系统却仍是一个梦想。现在不但系统规格说明极其复杂，而且它们遇到了实时期限、低功耗、有效地支持复杂实时用户界面、强劲的成本竞争以及设计的系统必须可升级等问题。

Wayne Wolf 教授编写了系统地处理大量新的系统设计需求和挑战的第一本教科书。他提出了嵌入式系统设计的形式体系和方法学，那些真正要理解跨越大量部件技术的系统设计基础、构造新型“超薄”系统的设计师可以运用这些方法。

从分析每一种技术基础出发，Wolf 教授为规范和建造系统结构及行为提供了形式化方法，然后通过一系列示例解析这些思想。他仔细研究了所涉及的复杂性以及如何系统化地处理这些复杂问题。你会提前清楚地理解这些设计问题的本质，并知道如何攻克这些难关的关键方法和工具。

作为嵌入式系统设计的第一本教科书，本书将被证明是在这个重要和最新出现的领域里获得知识的一个无价的工具。本书亦可以作为实际设计工作的参考，并在设计过程中作为可信的指南。我强烈地向读者推荐本书。

Lynn Conway

密歇根大学电子工程和计算机科学系名誉教授

第3版前言

Computers as Components: Principles of Embedded Computing System Design, Third Edition

本书第3版反映了我对嵌入式计算的深入思考以及对本书读者的若干建议，其中一个重要的目标是扩大嵌入式计算应用的范围。学习有关数码相机和汽车之类的主题需要付出很多努力。我希望本书可以为这些系统中最直接影响嵌入式计算设计者决策的部分提供一些有用的见解。我也扩大了示例处理器的范围，包括尖端的处理器，如 TI C64x 和高级 ARM 扩展（advanced ARM extension），同时还包括 PIC16F，并通过它描述小型 RISC 嵌入式处理器的特性。最后重新组织了网络和多处理器章节，使这些紧密相关的主题看起来更统一。读者可以在课程网站 <http://www.marilynwolf.us> 寻找附加的材料，这个网站包含了上面所说的所有内容，实验样例以及获取附加信息的提示。

我要感谢 Nate McFadden、Todd Green 和 Andre Cuello，感谢他们耐心的编辑以及在本书修订过程中对我的关心。我还要感谢匿名的评论者和科罗拉多大学的 Andrew Pleszkun 教授，他们对本书草稿富有洞察力的评论让我受益匪浅。特别感谢 David Anderson、Phil Koopman 和 Bruce Jacob 帮助我理解了一些内容。同样要感谢 Atlanta Snowpocalypse of 2011 给了我大量不受打扰的写作时间。

最重要的是，现在正是表达我对父亲的无比感激的时候。他教会我如何去工作：不仅仅是教会我怎样做具体的事情，更重要的是教会我如何处理问题，开拓思路，然后把它们转化为成果。一直以来，他教我如何体贴关心他人。感谢您，父亲。

Marilyn Wolf

相对 2000 年本书第 1 版出版的时候,如今嵌入式计算更为重要。更多的产品中使用了嵌入式处理器,从玩具到飞机都有应用。片上系统现在使用几百个 CPU。手机朝着新的标准计算平台方向发展。就像 2006 年 9 月《IEEE Computer》杂志上我的专栏中所指出的那样,当今世界至少有 50 万嵌入式系统程序员,可能接近 80 万。

在这一版中,我尽力做了更新和补充。本书的一个主要改变是使用 TI TMS320 C55x™ (C55x) DSP。我慎重地重写了关于实时调度的讨论。尝试将性能分析主题在尽可能多的抽象层次上扩展。指出多处理器在甚至最平凡的嵌入式系统中的重要性,此版对硬/软件协同设计和多处理器也进行了更普遍化的介绍。

计算机教学领域内的一个改变是,本教材成为了越来越低年级的课本。过去用于研究生的教材现在用于高年级本科生;在可预见的未来,本书的部分内容将可作为大学二年级教材。我认为可以选取本书的部分内容去覆盖更先进和更基础的课程。一些高年级学生可能不需要前面章节的背景知识,这样可以花更多时间在软件性能分析、调度和多处理器上。当开设介绍性课程时,软件性能分析可作为探索微处理器体系结构和软件体系结构的一个可选方案;这样的课程可以关注前几章的内容。

本书和我的其他书的新网站是 <http://www.waynewolf.us>。在这个网站里,可以找到本书相关材料的概括、实验建议,还可找到关于嵌入式系统的更多信息的网站链接。

感谢

感谢许多帮助我完成第 2 版的人。德州仪器的 Cathy Wicks 和 Naser Salameh 在理解 C55x 上给了我非常有价值的帮助。FreeRTOS.org 的 Richard Barry 不仅慷慨地允许我引用他的操作系统中的源码,还帮我澄清代码的解释。本书的编辑是 Morgan Kaufmann 出版社的 Chuck Glaser,他知道何时需要耐心,何时需要鼓励,何时需要诱导。当然,还要感谢 Nancy 和 Alec 耐心为我录入。本书的任何问题,不管是大小,自然都是我个人的责任。

Wayne Wolf

微处理器早已成为我们生活的一部分，然而，微处理器足够强大以致能执行真正复杂的功能还是近几年的事。根据摩尔定律，微处理器飞速发展的结果是嵌入式计算作为一门学科出现。在微处理器的早期阶段，所有组件相对较小也较简单，需要且期望把一些单独的指令和逻辑门集中在一起。今天，当系统包含了几千万个晶体管和数万行高级语言代码时，我们必须使用有助于我们处理复杂性的设计技术。

本书试图捕捉嵌入式计算这一新学科的某些基本原理和技术。嵌入式计算的某些挑战在台式机计算世界中是众所周知的。例如，为从带流水线的高速缓存体系结构中获得最高性能，经常需要仔细分析程序轨迹。类似地，随着嵌入式系统的复杂性不断增长，在软件工程中针对特定复杂系统开发的技术变得十分重要。另外一个例子是设计多进程系统。对于台式机，通用操作系统的需求与实时操作系统的需求是截然不同的；过去 30 年针对大型实时系统开发的实时技术如今已普遍用于基于微处理器的嵌入式系统中。

嵌入式计算还面临一些新的挑战。一个较好的例子是功耗问题。在传统计算机系统中功耗已经不是一个主要考虑因素，但是对于用电池供电的嵌入式计算机，这是一个基本考虑因素，而且在功耗容量受重量、成本或噪声等限制的情况下是十分重要的。另外一种挑战是截止时限驱动的程序设计。嵌入式计算机常常对程序完成的期限做硬性限制；这种形式的限制在台式机世界里是罕见的。随着嵌入式处理器越来越快，高速缓存和其他 CPU 单元也使得执行时间越来越难以预测。然而，通过仔细分析和巧妙编程，我们可以设计可预测执行时间的嵌入式程序，甚至面对高速缓存等不可预测的系统部件也如此。

幸运的是，有许多工具可用来处理复杂嵌入式系统所面临的挑战：高级语言、程序性能分析工具、进程和实时操作系统，等等。但是理解这些工具如何协调地一起工作本身就是一项很复杂的任务。本书提供了一种自底向上的方法来理解嵌入式系统设计技术。通过先理解微处理器硬件和软件的基础知识，我们就能获得有助于创建复杂系统的强有力的抽象能力。

对于嵌入式系统专业人员的忠告

本书不是一本用来理解某种特定微处理器的手册。为什么在这里呈现的技术你会感兴趣呢？有两个理由：第一，诸如高级语言编程和实时操作系统这样的技术对于构造大型实际的复杂嵌入式系统是非常重要的。生产会因为不能工作的错误系统设计而被弄得杂乱无章，系统之所以不能工作是因为它们的设计者试图从出现的问题中寻求解决方法而不是从问题中走出来换个更大的视角研究问题。第二，用于建造嵌入式系统的组件是经常变化的，但其原理不变。一旦你掌握了创建复杂嵌入式系统所涉及的基本原理，就可以迅速地学习一种新的微处理器（或编程语言），并且把同样的基本原理用于新的组件。

对于教师的忠告

传统的微处理器系统设计起源于 20 世纪 70 年代，当时微处理器的种类相对有限。传统课程强调定制硬件和软件来构建一个完整系统。因此，它只强调某一特定微处理器的特性，包括其指令系统、总线接口，等等。

本书采用更抽象的途径研究嵌入式系统。本书利用一切机会讨论实际组件和应用，但本质上它

不是一本微处理器数据手册，因此它的论述方法初看起来是新奇的。本书不是集中于某种特定微处理器，而是试图用一般性的例子来推出更一般的应用原理。本人认为这种方法对于教学更有利，而从长远角度来看对于学生也更有用。对于教学更有利是因为不必太过于依赖复杂的实验室装置，而只需花费一些时间在纸上练习、模拟和编程练习。对于学生更有用是因为他们在这一领域的最终工作所使用的组件和设施与学校的肯定是不同的。一旦学生掌握了基础知识，他们学习新组件的细节就会容易得多。

对于获得有关嵌入式系统的物理直觉，实践经验特别重要。某些硬件设计经验是非常宝贵的；我认为每一个学生都应该知道烧制塑封集成电路块的气味。但我强烈建议你避免专注于硬件设计。如果你花费太多的时间去构建硬件平台，你将没有足够的时间去编写有趣的程序。一个实际问题是，大多数课程没有时间让学生用高性能 I/O 设备和可能的多处理器来建造复杂的硬件平台。多数学生可以通过测量和评价一个现有的硬件平台来学习硬件知识。编制复杂嵌入式系统程序的实践也可以教给学生相当多的硬件知识，调试中断驱动代码是学生基本不会忘记的一种经验。

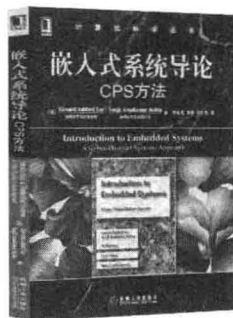
本书的主页 (www.mkp.com/embed) 中包括本书相关材料的概括、教师手册、实验材料、到相关 Web 站点的链接以及包含习题解答的有口令保护的 ftp 站点的链接。

感谢

感谢许多帮助我准备这本书的人。一些人给了我关于本书各个方面的建议：关于规格说明的 Steve Johnson (印第安纳大学)，关于程序跟踪的 Louise Trevillyan 和 Mark Charney (均在 IBM 研究所)，关于高速缓存未命中的 Margaret Martonosi (普林斯顿大学)，关于低功耗的 Randy Harr (Synopsys)，关于低功耗的 Phil Koopman (卡内基 - 梅隆大学)，关于低功耗计算与累加器的 Joerg Henkel (NEC C&C 实验室)，关于实时操作系统的 Lui Sha (伊利诺伊大学)，关于 ARM 体系结构的 John Rayfield (ARM)，关于编译器和 SHARC 的 David Levine (Analog Devices)，以及关于 SHARC 的 Con Korikis (Analog Devices)。许多人员在各阶段对本书进行了审阅：David Harris (哈佛姆德学院)；Jan Rabaey (加州大学伯克利分校)；David Nagle (卡内基 - 梅隆大学)；Randy Harr (Synopsys)；Rajesh Gupta、Nikil Dutt、Frederic Doucet 和 Vivek Sinha (加州大学欧文分校)；Ronald D. Williams (弗吉尼亚大学)；Steve Sapiro (SC 协会)；Paul Chow (多伦多大学)；Bernd G. Wenzel (Eurostep)；Steve Johnson (印第安纳大学)；H. Alan Mantooth (阿肯色大学)；Margarida Jacome (得克萨斯大学奥斯汀分校)；John Rayfield (ARM)；David Levine (Analog Devices)；Ardsher Ahmed (马萨诸塞大学/达特茅斯大学)；Vijay Madisetti (佐治亚理工学院)。还要特别感谢编辑 Denise Penrose，Denise 费了很大精力寻找本书的潜在用户并和他们交流，帮助我们了解读者想要学什么。特别感谢她的直觉和坚持。Cheri Palmer 和她的出版团队在无法想象的紧迫日程内作了出色的工作。当然，所有的错误和失误都是我的。

Wayne Wolf

推荐阅读



■ 嵌入式系统导论：CPS方法

作者：Edward Ashford Lee 等
ISBN：978-7-111-36021-6
定价：55.00元

■ 嵌入式软件设计基础——基于ARM Cortex-M3（原书第2版）

作者：Daniel W. Lewis
ISBN：978-7-111-44176-2
定价：45.00元

■ 现代嵌入式计算（英文版）

作者：Peter Barry 等
ISBN：978-7-111-41235-9
定价：79.00元

■ 嵌入式系统设计与实践

作者：Elecia White
ISBN：978-7-111-41584-8
定价：69.00元

■ 嵌入式系统软硬件协同设计实战指南：基于Xilinx Zynq

作者：陆佳华 等
ISBN：978-7-111-41107-9
定价：69.00元

■ STM32嵌入式系统开发实战指南：FreeRTOS与LwIP联合移植

作者：李志明 等
ISBN：978-7-111-41716-3
定价：69.00元

目 录

Computers as Components: Principles of Embedded Computing System Design, Third Edition

出版者的话

译者序

序言

第3版前言

第2版前言

第1版前言

第1章 嵌入式计算 1

1.1 引言 1

1.2 复杂系统与微处理器 1

1.2.1 嵌入式计算机 1

1.2.2 嵌入式计算应用的特点 3

1.2.3 为什么要使用微处理器 4

1.2.4 网络物理系统 5

1.2.5 嵌入式计算系统设计所面临的挑战 5

1.2.6 嵌入式计算系统的性能 6

1.3 嵌入式系统的设计过程 7

1.3.1 需求 7

1.3.2 规格说明 10

1.3.3 体系结构设计 11

1.3.4 设计硬件组件和软件组件 12

1.3.5 系统集成 12

1.3.6 系统设计的形式化方法 12

1.3.7 结构描述 13

1.3.8 行为描述 15

1.4 设计示例：模型火车控制器 17

1.4.1 需求 17

1.4.2 DCC 18

1.4.3 概念性规格说明 19

1.4.4 详细的规格说明 20

1.4.5 本节所学内容 24

1.5 本书概览 24

1.5.1 第2章：指令系统 25

1.5.2 第3章：CPU 25

1.5.3 第4章：计算平台 25

1.5.4 第5章：程序设计和 分析 26

1.5.5 第6章：进程和操作 系统 26

1.5.6 第7章：系统设计技术 27

1.5.7 第8章：网络和多处理器 27

1.6 总结 27

本章学习要点 28

推荐读物 28

习题 28

实验练习 29

第2章 指令系统 30

2.1 引言 30

2.2 预备知识 30

2.2.1 计算机体系结构分类 30

2.2.2 汇编语言 32

2.2.3 超长指令字处理器 32

2.3 ARM 处理器 33

2.3.1 处理器和存储器组织 34

2.3.2 数据操作 34

2.3.3 控制流 39

2.3.4 高级 ARM 特性 43

2.4 PICmicro 中档系列 44

2.4.1 处理器和存储器组织 44

2.4.2 数据操作 44

2.4.3 控制流 46

2.5 TI C55x DSP 46

2.5.1 处理器和存储器组织 47

2.5.2 寻址方式 49

2.5.3 数据操作 50

2.5.4 流程控制 51

2.5.5 C 编码准则 52

2.6 TI C64x 53

2.7 总结 55

本章学习要点	55	4.2.2 平台软件组件	99
推荐读物	55	4.3 CPU 总线	99
习题	55	4.3.1 总线结构和协议	99
实验练习	57	4.3.2 DMA	104
第3章 CPU	58	4.3.3 系统总线配置	105
3.1 引言	58	4.4 存储设备和系统	106
3.2 输入/输出编程	58	4.5 计算平台的设计	108
3.2.1 输入/输出设备	58	4.5.1 示例平台	108
3.2.2 输入/输出原语	60	4.5.2 平台选择	109
3.2.3 忙等 I/O	61	4.5.3 知识产权	110
3.2.4 中断	62	4.5.4 开发环境	110
3.3 管态、异常和陷阱	70	4.5.5 调试技术	111
3.3.1 管态	71	4.5.6 调试难点	113
3.3.2 异常	71	4.6 消费类电子设备的体系结构	114
3.3.3 陷阱	71	4.6.1 消费类电子设备的用例和 需求	114
3.4 协处理器	71	4.6.2 文件系统	115
3.5 存储系统机制	72	4.7 平台级性能分析	115
3.5.1 高速缓存	72	4.8 设计示例：闹钟	118
3.5.2 存储管理单元和地址 转换	76	4.8.1 需求	118
3.6 CPU 的性能	78	4.8.2 规格说明	119
3.6.1 流水线技术	78	4.8.3 系统体系结构	120
3.6.2 高速缓存的性能	81	4.8.4 组件设计与测试	122
3.7 CPU 的功耗	82	4.8.5 系统集成与测试	122
3.8 设计示例：数据压缩器	85	4.9 设计示例：音频播放器	123
3.8.1 需求和算法	85	4.9.1 工作原理和需求	123
3.8.2 规格说明	86	4.9.2 规格说明	124
3.8.3 程序设计	88	4.9.3 系统体系结构	124
3.8.4 测试	93	4.9.4 组件设计与测试	125
3.9 总结	93	4.9.5 系统集成与调试	126
本章学习要点	94	4.10 总结	126
推荐读物	94	本章学习要点	126
习题	94	推荐读物	126
实验练习	96	习题	126
第4章 计算平台	97	实验练习	128
4.1 引言	97	第5章 程序设计与分析	129
4.2 基本计算平台	97	5.1 引言	129
4.2.1 平台硬件组件	97	5.2 嵌入式程序组件	129

5.2.1 状态机	129	5.12.2 规格说明	177
5.2.2 循环缓冲区和面向流的 程序设计	131	5.12.3 系统体系结构	178
5.2.3 队列和生产者/消费者 系统	134	5.12.4 组件设计与测试	179
5.3 程序模型	136	5.12.5 系统集成与测试	179
5.3.1 数据流图	136	5.13 总结	180
5.3.2 控制/数据流图	137	本章学习要点	180
5.4 汇编、连接和装载	138	推荐读物	181
5.4.1 汇编程序	139	习题	181
5.4.2 连接	141	实验练习	187
5.4.3 目标代码设计	142	第6章 进程和操作系统	188
5.5 编译技术	143	6.1 引言	188
5.5.1 编译处理	143	6.2 多任务和多进程	188
5.5.2 基本编译方法	144	6.3 多速率系统	189
5.5.3 编译器优化	149	6.3.1 进程的时限要求	191
5.6 程序级性能分析	154	6.3.2 CPU 使用率度量	193
5.6.1 程序性能的要素	156	6.3.3 进程状态与调度	193
5.6.2 测量驱动的性能分析	158	6.3.4 运行周期性进程	194
5.7 软件性能优化	160	6.4 可抢占的实时操作系统	195
5.7.1 循环优化	160	6.4.1 两个基本概念	196
5.7.2 cache 优化	161	6.4.2 进程和上下文	196
5.7.3 性能优化策略	162	6.4.3 进程和面向对象的设计	199
5.8 程序级功耗分析与优化	162	6.5 基于优先级的调度	199
5.9 程序大小的分析与优化	165	6.5.1 单一速率调度	200
5.10 程序验证与测试	165	6.5.2 共享资源	202
5.10.1 白盒测试	166	6.5.3 优先级反转	204
5.10.2 黑盒测试	170	6.5.4 最早截止时限优先调度	204
5.10.3 评估功能测试	171	6.5.5 RMS 和 EDF 的比较	206
5.11 设计示例：软件调制解 调器	171	6.5.6 模型化的假设的详细 介绍	207
5.11.1 工作原理和需求	171	6.6 进程间通信机制	208
5.11.2 规格说明	172	6.6.1 共享内存通信	208
5.11.3 系统体系结构	173	6.6.2 消息传递	209
5.11.4 组件设计与测试	173	6.6.3 信号	210
5.11.5 系统集成与测试	173	6.6.4 邮箱	210
5.12 设计示例：数码相机	174	6.7 评估操作系统性能	210
5.12.1 工作原理和需求	174	6.8 进程的功耗优化策略	213
		6.9 实时操作系统举例	215
		6.9.1 POSIX	215

6.9.2 Windows CE	219	习题	248
6.10 设计示例:电话应答机	221	实验练习	248
6.10.1 工作原理和需求	221	第8章 网络和多处理器	249
6.10.2 规格说明	222	8.1 引言	249
6.10.3 系统体系结构	224	8.2 为什么要使用网络和多处 理器	249
6.10.4 组件设计和测试	225	8.3 多处理器分类	250
6.10.5 系统集成和测试	225	8.4 分布式嵌入式系统	251
6.11 设计示例:发动机控制 单元	225	8.4.1 网络抽象模型	252
6.11.1 工作原理和需求	225	8.4.2 CAN 总线	252
6.11.2 规格说明	226	8.4.3 汽车和飞机上的分布式 计算	254
6.11.3 系统体系结构	226	8.4.4 I ² C 总线	255
6.11.4 组件设计与测试	227	8.4.5 以太网	258
6.11.5 系统集成与测试	227	8.4.6 因特网	259
6.12 总结	227	8.5 多处理器片上系统和共享存储 多处理器	260
本章学习要点	227	8.5.1 异构共享存储多处理器	260
推荐读物	228	8.5.2 加速器	261
习题	228	8.5.3 加速器性能分析	262
实验练习	231	8.5.4 调度和分配	264
第7章 系统设计技术	233	8.6 设计示例:视频加速器	265
7.1 引言	233	8.6.1 视频压缩	265
7.2 设计方法学	233	8.6.2 算法和需求	266
7.2.1 为什么需要设计方法学	233	8.6.3 规格说明	267
7.2.2 设计流	234	8.6.4 体系结构	268
7.3 需求分析	237	8.6.5 组件设计	269
7.4 规格说明	238	8.6.6 系统测试	269
7.4.1 面向控制的规格说明 语言	238	8.7 应用示例:CD	270
7.4.2 改进的规格说明	239	8.8 总结	272
7.5 系统分析和结构设计	241	本章学习要点	272
7.6 质量保证	243	推荐读物	272
7.6.1 质量保证技术	245	习题	272
7.6.2 检验规格说明	246	实验练习	273
7.6.3 设计评审	247	术语	274
7.7 总结	247	索引	285
本章学习要点	248		
推荐读物	248		

嵌入式计算

本章要点

- 为什么在系统中嵌入微处理器？
- 嵌入式计算和网络物理系统设计的困难和特点。
- 设计方法学。
- 系统规范。
- 本书导读。

1.1 引言

在这一章，我们开始进入对嵌入式计算系统设计的研 究。为了解设计的流程，我们首先需要知道微处理器为什么以及怎样被用于控制、用户界面、信号处理以及其他一些任务。微处理已经如此普遍，我们几乎忘记了如果没有微处理器许多事情将会多么困难。

我们首先回顾微处理器的不同用途，然后回顾微处理器为什么被用于系统设计，实现传递复杂行为、快速设计周期等任务。接下来，在 1.2 节中，我们通过一个完整的系统设计示例，理解系统设计的主要步骤。1.3 节深入剖析了嵌入式系统的具体技术，这些技术将贯穿整本书的内容。在 1.4 节，我们使用一个火车控制器模型作为例子来应用 1.3 节中介绍的技术。1.5 节提供了本书的逐章导读。

1.2 复杂系统与微处理器

我们通常认为我们的笔记本电脑是计算机，但实际上它只是许多计算机系统类型中的一种。一台计算机就是一个从存储器中获取和执行指令的一个存储程序的机器。我们能把不同类型的设备连接到计算机，用不同种类的软件进行装载，并且创建许多不同种类的系统。

1

那么，什么是**嵌入式计算系统**？从广义上来定义，它可以是包含可编程计算机的任何设备，但其本身并非通常意义上的计算机。因此，一台 PC 本身并不是一个嵌入式计算系统。但是，通过微处理器构成的一台传真机或时钟是一个嵌入式计算系统。

这意味着，对很多类型的产品设计来说，嵌入式计算系统设计是一种有用的技术。汽车、手机甚至家用电器都广泛地使用了微处理器。在许多领域中，设计者都必须能判定哪些地方可以使用微处理器，为所需任务设计带有 I/O 设备的硬件平台，并实现能够执行所需任务的软件。计算机工程，例如机械设计、热力学，是能够广泛应用于许多不同领域的基础学科。但是，嵌入式计算系统设计并不是孤立的学科。许多嵌入式计算系统设计中遇到的挑战并不是计算机工程的问题，例如，它们有可能是机械或模拟电路方面的问题。我们在本书中仅关注嵌入式计算机本身，因此，将重点集中在使最终产品能实现预期功能的硬件和软件上。

1.2.1 嵌入式计算机

在计算早期，计算机就已经被嵌入到应用中。Whirlwind 就是其中的一个例子，它是一台在 20 世纪 40 年代末到 50 年代初由 MIT 设计的计算机。Whirlwind 也是第一台支持实时操作的计算机，最初的设想是作为一种控制飞机模拟器的装置。尽管与现代的计算机相比它的体积非常庞大（包含超过 4000 个电子管），但其从构件到系统的完整设计是适应实时嵌入式计算需求的。计算机在取代机械或人为控制方面的作用在它刚开始发展时就很明显，比如，在 20 世纪 40 年代末，就用计算机来

控制化学药品的生产过程 [Sto95]。

微处理器是一种单芯片 CPU。20 世纪 70 年代开始, 超大规模集成电路技术使得我们能够将一个完整的 CPU 集成到单芯片上, 但是这些 CPU 都非常简单。第一个微处理器是 Intel 4004, 它是为嵌入式应用设计的, 用于计算器中。这台计算器并不是通用计算机, 它仅提供基本的算术功能。然而, Intel 的 Ted Hoff 实现了可正确执行所需功能的可编程通用计算机, 并且这种单片计算机经过重新编程就可用于其他产品中。因为集成电路设计是一个昂贵且费时的过程, 所以通过修改软件实现硬件设计复用是一个关键的突破。HP-35 是第一台用来计算超越函数的手持计算器 [Whi72], 1972 年面世。它使用多块芯片实现了 CPU, 而不同于以往单芯片的微处理器。特别是, 它通过编程而不是数字电路设计来执行诸如三角函数之类的数学计算, 这是这台计算器成功设计的关键。

在单芯片 CPU 问世后, 汽车设计者很快开始使用微处理器。汽车中最重要同时也是最复杂的微处理器应用就是控制引擎, 如控制火花塞点火、燃料与气体混合等等。汽车电子的通用发展趋势, 就是电子设备将会取代机械装置。而推动基于微处理引擎控制的巨大力量却来自于两个几乎同时发展起来的领域, 一是 20 世纪 70 年代的石油危机使得消费者在燃料上的花费更多, 二是人们担心大气污染会促成限制汽车引擎排放的法律出台。要同时实现低油耗和低排放是相当困难的, 为满足这些目标而又不损害发动机性能, 汽车制造商求助于只有微处理器才能执行的复杂控制算法。

微处理器具有不同层次的复杂性, 通常按它们的字长来分类。8 位微控制器是为低开销应用设计的, 它包括板载存储器和 I/O 设备。16 位微控制器通常用于更复杂的应用, 比如更长的字长或者片外 I/O 和存储器。32 位 RISC (精简指令集计算机) 微处理器为更复杂的计算应用提供非常高的性能。

各种类型微处理器的出现, 使微处理器毫不意外地有了更广泛的应用范围。现在已经出现了许多家庭使用的微处理器。典型的微波炉至少有一个微处理器控制微波炉的操作。许多房子拥有先进的自动温控系统, 可以在一天的不同时间调节室内温度。现代照相机也是通过嵌入微处理器控制实现其强大功能的一个很好的例子。

数字电视广泛地使用了嵌入式处理器。在一些情况下, 需要设计专用的 CPU 来执行重要的算法, 例如, 用于 DirecTV 的 SGS Thomson 芯片集中就有进行音频处理 [Lie98] 的 CPU, 它可以高效地实现数字音频解码的程序。使用可编程 CPU 而不是硬件单元, 主要有两个原因: 第一, 它使系统更容易设计和调试; 第二, 它为升级和将 CPU 用于其他应用提供可能。

现在, 高端的汽车可能有 100 多个微处理器, 即便是便宜的轿车也有 40 多个微处理器。其中一些微处理器只做很简单的事情, 例如检测安全带是否在使用。其他的微处理器控制关键的功能, 如点火和制动系统。

设计示例 1-1 介绍了 BMW 850i 中使用的一些微处理器。

设计示例 1-1 BMW 850i 制动与稳定控制系统

此例介绍了 BMW 850i 的一套复杂的车轮控制系统。防抱死制动系统 (ABS) 通过反复制动减少打滑。自动稳定控制系统 (ASC + T) 辅助引擎以改善汽车行驶的稳定性。这些系统积极的控制汽车的关键部分, 作为控制系统, 它们需要和汽车交互操作。

我们首先看 ABS。ABS 的目的就是当车轮停止转动, 车辆开始打滑变得难以控制时, 临时释放制动力。它介于助力泵 (提供刹车动力) 和制动器之间, 如下图所示, 这个连接允许 ABS 系统来调节刹车, 以防止车轮被锁死。ABS 系统使用车轮上的传感器测量车轮的转速。根据车轮的转速, ABS 系统决定如何改变液压泵的压力, 以防止车轮打滑。