

PEARSON
Prentice
Hall

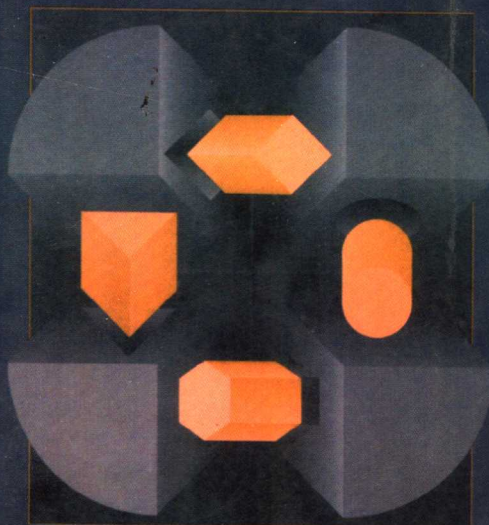
计 算 机 科 学 丛 书

嵌入式系统的 描述与设计

(美) Daniel D. Gajski Frank Vahid Sanjiv Narayan Jie Gong 著

边计年 吴为民 等译
清华大学

SPECIFICATION
AND DESIGN
OF EMBEDDED
SYSTEMS



DANIEL D. GAJSKI / FRANK VAHID
SANJIV NARAYAN / JIE GONG

Specification and Design of
Embedded Systems



机械工业出版社
China Machine Press



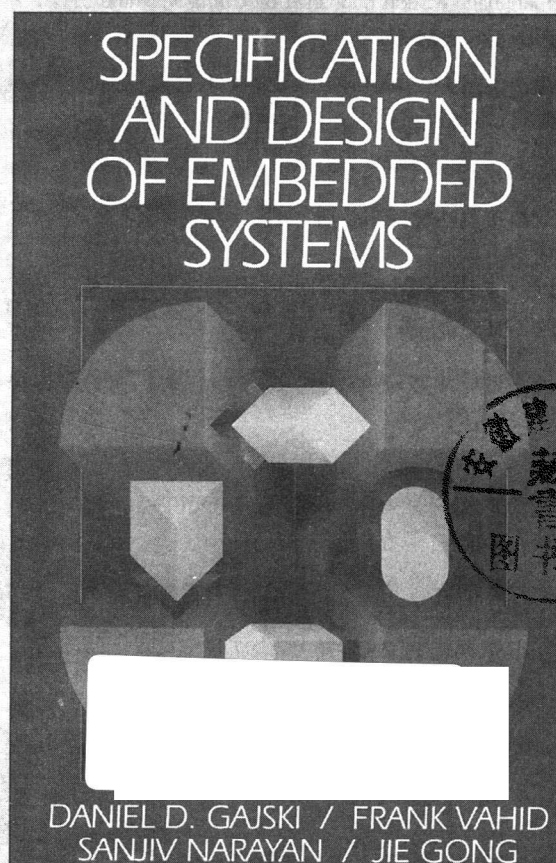
计 算 机 科 学 丛 书

嵌入式系统的 描述与设计

TP360.21
14

(美) Daniel D. Gajski Frank Vahid Sanjiv Narayan Jie Gong 著

边计年 吴为民 等译
清华大学



Specification and Design of Embedded Systems



机械工业出版社
China Machine Press

本书介绍嵌入式系统领域的基本概念以及实际的描述和 design 方法,包括嵌入式系统设计的模型和体系结构、描述语言、系统划分、设计质量评估、描述细化以及系统级方法学等方面。讲解详细,实例丰富,有针对性地介绍了若干著名算法或解法,并解释每种方法的优缺点,还包括对该领域其他工作的综述,并提出尚未解决的一些问题。本书适合从事嵌入式系统设计和研究的工程技术人员、科研人员、高等院校计算机和电子信息工程专业的本科生和研究生。

Authorized translation from the English language edition entitled *Specification and Design of Embedded Systems* by Daniel D. Gajski, Frank Vahid, Sanjiv Narayan, Jie Gong, published by Pearson Education, Inc., publishing as Prentice-Hall (ISBN 0-13-150731-1), Copyright © 1994 by Daniel D. Gajski, Frank Vahid, Sanjiv Narayan, Jie Gong.

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

Chinese Simplified language edition published by China Machine Press.

Copyright © 2005 by China Machine Press.

本书中文简体字版由美国 Pearson Education 培生教育出版集团授权机械工业出版社独家出版。未经出版者书面许可,不得以任何方式复制或抄袭本书内容。

版权所有,侵权必究。

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:01-2003-8708

图书在版编目(CIP)数据

嵌入式系统的描述与设计/(美)盖斯基(Gajski D. D)等著;边计年等译. -北京:机械工业出版社,2005.7

(计算机科学丛书)

书名原文:Specification and Design of Embedded Systems

ISBN 7-111-16422-9

I. 嵌… II. ①盖… ②边… 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(2005)第 066897 号

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:隋 曦

北京昌平奔腾印刷厂印刷·新华书店北京发行所发行

2005 年 7 月第 1 版第 1 次印刷

787mm×1092mm 1/16·16.75 印张

印数:0 001-4 000 册

定价:33.00 元

凡购本书,如有倒页、脱页、缺页,由本社发行部调换

本社购书热线:(010)68326294

出版者的话

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，还揭橥了学术的源变，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、从业人员较少的现状下，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图文信息有限公司较早意识到“出版要为教育服务”。自1998年开始，华章公司就将工作重点放在了遴选、移译国外优秀教材上。经过几年的不懈努力，我们与Prentice Hall, Addison-Wesley, McGraw-Hill, Morgan Kaufmann等世界著名出版公司建立了良好的合作关系，从它们现有的数百种教材中甄选出Tanenbaum, Stroustrup, Kernighan, Jim Gray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、研究及度藏。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，有的还专诚为其书的中译本作序。迄今，“计算机科学丛书”已经出版了近百个品种，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，为进一步推广与发展打下了坚实的基础。

随着学科建设的初步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步入一个新的阶段。为此，华章公司将加大引进教材的力度，在“华章教育”的总体规划之下出版三个系列的计算机教材：除“计算机科学丛书”之外，对影印版的教材，则单独开辟出“经典原版书库”；同时，引进全美通行的教学辅导书“Schaum's Outlines”系列组成“全美经典学习指导系列”。为了保证这三套丛书的权威性，同时也为了更好地为学校和老师服务，华章公司聘请了中国科学院、北京大学、清华大学、国防科技大学、复旦大学、上海交通大学、南京大学、浙江大学、中国科技大学、哈尔滨工业大学、西安交通大学、中国人民大学、北京航空航天大学、北京邮电大学、中山大学、解放军理工大学、郑州大学、湖北工学院、中国国家信息安全测评认证中心等国内重点大学和科研机构在计算机的各个领域的著名学者组成“专家指导委员会”，为我们提供选题意见和出版监督。

这三套丛书是响应教育部提出的使用外版教材的号召，为国内高校的计算机及相关专业

的教学度身订造的。其中许多教材均已为M. I. T., Stanford, U.C. Berkeley, C. M. U. 等世界名牌大学所采用。不仅涵盖了程序设计、数据结构、操作系统、计算机体系结构、数据库、编译原理、软件工程、图形学、通信与网络、离散数学等国内大学计算机专业普遍开设的核心课程,而且各具特色——有的出自语言设计者之手、有的历经三十年而不衰、有的已被全世界的几百所高校采用。在这些圆熟通博的名师大作的指引之下,读者必将在计算机科学的宫殿中由登堂而入室。

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,但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。教材的出版只是我们的后续服务的起点。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:

电子邮件: hzedu@hzbook.com

联系电话: (010) 68995264

联系地址: 北京市西城区百万庄南街1号

邮政编码: 100037

专家指导委员会

(按姓氏笔画顺序)

尤晋元
石教英
张立昂
邵维忠
周立柱
范明
袁崇义
谢希仁

王珊
吕建
李伟琴
陆丽娜
周克定
郑国梁
高传善
裘宗燕

冯博琴
孙玉芳
李师贤
陆鑫达
周傲英
施伯乐
梅宏
戴葵

史忠植
吴世忠
李建中
陈向群
孟小峰
钟玉琢
程旭

史美林
吴时霖
杨冬青
周伯生
岳丽华
唐世渭
程时端

译者序

多年来,半导体制造技术持续迅速发展,芯片规模的增大使得集成电路的应用范围也在迅速扩大,已经渗透到各行各业,在机械系统和其他系统中广泛使用,形成所谓嵌入式系统。嵌入式系统深入到人类活动的几乎所有方面。从和我们日常生活密切相关的手机、个人数字助理(PDA)、音响设备、高清晰度电视、游戏机,到公共场所常见的汽车电子设备、通信设备、电子医疗设备,再到特殊领域,如军事、航天、水下等使用的电子设备,嵌入式系统无时无刻不在发挥着重要作用并将继续扮演越来越重要的角色。同时,片上系统(SOC)应运而生,在一个芯片上可以同时集成微处理器、存储器、专用电路、输入输出电路、模拟电路,甚至射频电路和微机械。

我们已经有了相当多有关逻辑综合、布局、布线等方面的工具。在目标系统的规模不断扩大、功能日益复杂的情况下,传统的设计方法学已经越来越不适应这种变化,迫切需要从更高的层次上重新审视和研究数字系统的设计方法学问题。本书作者提出的设计理念是,在设计早期的更抽象的高层次上开始,进行系统级设计。首先,在高层次上的设计有助于更好地理解系统的功能,保证设计的正确性。其次,在高层次上的决策对整个设计的影响更大,因而,早期的正确决策将得到更大的回报。由于在系统级引入优化策略,可以在各种设计选择中寻求最好的解决方案,因而可以预期基于这种新的方法学会设计出更加有竞争力的产品。

本书是一本很好的介绍嵌入式系统描述和设计技术的教科书。作者根据多年的研究经验和成果,既系统地介绍基本概念,又介绍实际的描述和设计方法。本书几乎论述了嵌入式系统设计的所有方面,包括模型和体系结构、描述语言、系统划分、设计质量评估、描述细化以及系统级方法学等。而且,对于每章涉及的每个问题,作者都予以详细地讲解并给出大量的实例。本书对于初学者和相关领域的从业人员来说,是一本很好的入门书,而对于从事嵌入式系统设计和研究的工程技术人员、科研人员和高等院校的本科生和研究生又是一本很详尽的参考书。据我们所知,本书已在全世界很多大学中作为教材或参考书,具有最高的采用率。

本书虽出版于1994年,但其基本的思想和方法仍然是嵌入式系统设计的指导思想和理论基础。本书介绍的描述语言,作者在后来又有所改进,并改称为SpecC。除美国外,还有日本和加拿大等许多国家都在研究和推广这种系统描述语言。

进入2000年以来,在国家的大力支持和推动下,我国在嵌入式系统和片上系统SOC的研究与开发领域出现了前所未有的好势头。我们向读者介绍这本书,相信对我国集成电路工业和技术的发展会发挥一定作用。

本书是许多老师同学共同努力的结晶。吴为民译第1章,刘志鹏译第2章,王海力译第3章,王迪译第4章和附录A,芦琰琰译第5章,童琨译第6、9章,赵康、林锋译第7章,王云峰、赵建洲译第8章。边计年和吴为民分别对所有翻译初稿进行了仔细的审校和修改,最后由边计年统稿定稿。对于有些术语,国内出现不同的说法,我们根据全国科学技术名词审定委员会公布的《计算机科学技术名词》和自己的理解决定取舍。虽然我们的宗旨是准确表达原文的意思,但鉴于译者水平和时间所限,错误和不足在所难免,望读者不吝指正。

译者

2005年1月

英文版序言

基本原理

在过去的十年里,VLSI 的设计技术特别是 CAD 产业取得了很大成功,与 IC 制造技术同步以异常迅猛的速度发展。与高抽象层次相比,低层次的设计问题更早地变得难解和费时。因此,学术界和产业界都被迫将注意力首先投向诸如电路模拟、布局、布线及布图规划等问题。当这些问题变得易于处理时,便成功地开发出用于逻辑模拟和综合的 CAD 工具,并将其引入到设计过程中。随着设计复杂性的大幅度提高和产品进入市场时间要求的大幅度缩短,产业界和学术界都开始关注比逻辑级和布局级更高级别的设计问题。由于高层次的抽象使设计者需要考虑的对象数目降低了一个数量级,也就允许产业界能够在更短的时间内设计和制造出复杂的专用集成电路(ASIC)。

继逻辑综合之后,行为综合也在设计方法学中发挥了提高抽象级别的作用。行为综合是用于设计单个专用集成电路的。当设计方法学要求更高的抽象级别时,这些专用集成电路和标准处理器及内存一起被用做系统的组件。系统级的设计方法学关注系统描述以及从描述到一组互联组件的变换和细化。系统描述依据执行于抽象数据类型上的计算,而变换和细化则包括针对标准处理器进行软件编译和根据定制的组件进行硬件综合。但在这一点上,虽然已有若干年系统制造的历史,尽管有明确的需求,产业界和学术界并未对系统级设计方法学的发展和形成给予充分关注。为解决复杂性问题和缩短设计周期,产业界最近已开始关注于建立一个一致性的系统级设计方法学。

强调更抽象化和系统级方法学的主要原因是基于这样一个事实:高层次抽象更接近设计者的日常思维习惯。比如,难以想像设计者仅靠电路原理图如何能够对一个由 10 万个门或 10 万个布尔表达式组成的系统设计进行描述、建立文档及进行交流。系统越复杂,当设计者用电路级、逻辑级、寄存器级原理图对其进行描述时,就越会产生功能理解上的困难。另一方面,当系统被描述成操作于抽象数据类型上的一系列复杂计算,并且通过抽象通道进行通信时,设计者将发现这更易于描述和验证特定的功能,更易于采用不同技术对各种实现进行评估。

必须承认,对系统级设计的研究虽已有很多年,但目前仍在相当程度上局限于特定领域和团体。例如,计算机体系结构界考虑的是将计算和算法映射到不同体系结构的方法。体系结构可以是脉动阵列、超立方体、多处理器以及大规模并行处理器。软件工程界则一直在研究软件代码的描述和设计方法。CAD 界集中于接口综合、内存管理、系统描述捕捉以及设计空间探索等系统问题。但是,很多问题仍悬而未决,其中最重要的是缺乏一个得到普遍接受的理论框架和支持系统设计方法学的 CAD 环境。尽管有这些悬而未决的问题,系统设计技术也已成熟到一定程度。因此,用一本书对发展至今的基本概念和成果进行总结,希望会对系统设计领域的学生和从业者有所帮助。在本书中,我们力图涵盖多种研究项目的思想和成果。由于这个领域还比较年轻,我们仍可能忽略了某些很有趣和有益的项目,为此我们在此表示歉意,并希望得到这些技术的信息,以便将它们结合到将来的版本中。同样,由于各种原因,我们也没

能在本书中详细地论及系统级的若干重要问题,包括形式验证、测试设计以及协同模拟。尽管如此,我们相信,一本有关系统级描述和设计的书将有助于电子系统设计自动化(ESDA)在未来的发展和繁荣。

读者

本书面向计算机科学和工程界的三类读者。首先,它会吸引系统设计者和工程管理者,因为他们会对 ASIC 和系统设计方法学、软硬件协同设计以及设计过程管理感兴趣。其次,该书也可以被 CAD 工具的开发所利用,他们会采用该书中所给出的现存或未来工具中的一些概念进行系统描述捕捉、设计空间探索以及系统建模和细化。最后,由于该书综述了系统设计的基本概念,提出了包括软件和硬件在内的系统设计方法学的原理,因此,对于一门高年级本科或研究生课程来说,该书也是有价值的。该课程可针对有志于计算机体系结构、设计自动化和/或软件工程的学生。

本书的组织

本书组织成 9 章,分为四个部分。第 1、2 章提出系统设计的一些基本问题,讨论了可用于系统行为捕获和实现的各种概念模型。第 3、4、5 章论及用于描述系统功能性的语言,还涉及通过模拟验证系统功能性的各种问题。第 6、7、8 章提供了一个有关系统划分、评估和模型细化的算法和技术的综述。第 9 章将所有这些主题结合成一个一致性的设计方法学,包括对系统设计通用环境的讨论。

在对第 1、2 章所定义概念理解的前提下,每一章都是独立的,可以单独阅读。本书每一章都采用了相同的写作风格和组织方法。典型的一章包含了介绍性的例子,定义了基本概念,阐述了要解决的主要问题。针对所提出的问题,每一章还描述了若干著名的算法或解法,并解释每种方法的优缺点。每一章还包括对该领域其他工作的小综述,以及对发展方向的讨论。

在每章的末尾,我们还给出若干练习,分为三类:家庭作业类问题,项目类问题,以及课题类问题。家庭作业类问题旨在检测读者对每章基本内容的理解。项目类问题用一个星标记。为解决项目类问题,读者需要在对一些文献进行研究的基础上才能对题目有更深入的理解。这类问题可能需要学生用若干星期时间来完成。课题类问题用两个星标记,是尚未解决的问题,若深入研究,将是不错的硕士或博士学位论文课题。

本书可用于两种不同的课程。一种可侧重于系统描述、建立文档以及验证方面,跳过第 6、7、8 章中的算法部分。另一种课程,侧重设计方法学和设计空间探索技术,跳过语言和模拟方面的内容。无论采用哪种课程,我们认为本书都会有助于填补计算机科学与工程课程中的一个空白,即除了涵盖电路设计、逻辑设计以及计算机体系结构的内容,还讲述了系统设计技术。

我们希望本书的内容选择和写作风格接近您的期望。我们欢迎您的意见和建议。

Daniel Gajski, Frank Vahid, Sanjiv Narayan, Jie Gong
加州大学艾尔温分校

致 谢

在过去六年中,我们同许多同事和学生讨论过系统设计的基本问题。感谢他们,因为没有这些讨论,很多问题就不会得到澄清,很多思想也不会得到尝试。

感谢帮助我们形成本书的选题和确定写作材料的人们。特别要感谢 SRC 的 Peter Verhofstadt 和 NSF 的 Bob Grafton 鼓励我们在此领域进行研究,以及 Rockwell 国际公司的 Bob Larsen 多年来帮助我们开发质量度量方法和用于比较分析的客观方法。

我们对加州大学艾尔温分校的 Nikil Dutt 教授、Fadi Kurdahi 教授和 Queensland 大学的 Sri Parameswaran 教授致以谢意。在几次会议期间,他们提供了有价值的意见,指出了系统设计的相关工作。我们也感谢松下研究和开发实验室的 Peter Fung 为我们的设计方法学所做的证实工作。我们同时感谢加州大学艾尔温分校 CAD 实验室的以下人员: Smita Bakshi、Viraphol Chaiyakul、Tedd Hadley、Nancy Holmes、Pradip Jha、Erica Juan、Raghava Kondepudy 以及 Loganath Ramachandran,他们的审校工作和提出的有价值的建议使本书更易于理解。感谢 Sarah Wilde 和 Judy Olson 的编辑工作和在理解复杂技术材料时所付出的耐心。我们对 Jon Kleinsmith 在协调作者和审稿者过程中所做的工作表示感谢。

该项工作受到美国自然科学基金(批准号 MIP-8922851)和半导体研究公司(批准号 93-DJ-146)的部分资助。作者对他们的支持表示感谢。

目 录

出版者的话		2.9.1 控制器体系结构	26
专家指导委员会		2.9.2 数据通路体系结构	26
译者序		2.9.3 带数据通路的有限状态机	28
英文版序言		2.10 处理器	28
致谢		2.10.1 复杂指令集计算机	28
第1章 引言	1	2.10.2 精简指令集计算机	29
1.1 设计表示	1	2.10.3 向量机	30
1.2 抽象级别	2	2.10.4 超长指令字计算机	31
1.3 当前的设计方法学	3	2.11 并行处理器	31
1.4 系统级方法学	5	2.12 结论	33
1.5 系统描述和设计	6	2.13 练习	33
第2章 模型与体系结构	9	第3章 系统描述语言	35
2.1 引言	9	3.1 引言	35
2.2 模型分类	11	3.2 概念模型的特性	36
2.3 面向状态的模型	11	3.2.1 并发性	36
2.3.1 有限状态机	11	3.2.2 状态迁移	38
2.3.2 Petri 网	13	3.2.3 层次化	39
2.3.3 层次化并发有限状态机	15	3.2.4 程序结构	40
2.4 面向活动的模型	16	3.2.5 行为完成	41
2.4.1 数据流图	16	3.2.6 通信	42
2.4.2 流程图	17	3.2.7 同步	43
2.5 面向结构的模型	18	3.2.8 异常处理	45
2.6 面向数据的模型	18	3.2.9 非确定性	46
2.6.1 实体-关系图	18	3.2.10 时序	46
2.6.2 Jackson 图	19	3.3 嵌入式系统的描述要求	47
2.7 异构模型	20	3.4 描述语言综述	48
2.7.1 控制/数据流图	20	3.4.1 VHDL	48
2.7.2 结构图	20	3.4.2 Verilog	50
2.7.3 程序设计语言模式	22	3.4.3 HardwareC	51
2.7.4 面向对象的模型	23	3.4.4 CSP	52
2.7.5 程序状态机	24	3.4.5 Statecharts	53
2.7.6 队列模型	25	3.4.6 Argos	55
2.8 体系结构分类	26	3.4.7 SDL	55
2.9 专用体系结构	26	3.4.8 Silage	56

3.4.9 Esterel	56	5.4.2 分叉	88
3.5 SpecCharts	57	5.5 异常处理	89
3.5.1 语言描述	57	5.6 从程序状态机到任务	90
3.5.2 用 SpecCharts 描述嵌入式系统	59	5.6.1 概述	90
3.5.3 等价图图形化表示	61	5.6.2 算法	91
3.5.4 语言的可扩展性	61	5.6.3 时间调整	92
3.6 结论和发展方向	62	5.6.4 综合	93
3.7 练习	63	5.7 结论和发展方向	93
第 4 章 系统描述举例	65	5.8 练习	93
4.1 引言	65	第 6 章 系统划分	95
4.2 电话应答机	65	6.1 引言	95
4.3 用 SpecCharts 进行系统描述	67	6.2 结构划分和功能划分	95
4.4 测试用例举例	72	6.2.1 结构划分	95
4.5 可执行系统描述的优点	73	6.2.2 功能划分	96
4.6 PSM 模型的优势	74	6.3 划分中的问题	97
4.6.1 层次性	74	6.3.1 系统描述抽象级别	98
4.6.2 状态迁移	74	6.3.2 粒度	98
4.6.3 程序设计结构	74	6.3.3 系统组件的分配	99
4.6.4 并行性	75	6.3.4 度量和评估	99
4.6.5 异常处理	75	6.3.5 目标函数和接近函数	100
4.6.6 完成	75	6.3.6 划分算法	101
4.6.7 状态分解和编码的等价性	75	6.3.7 输出	101
4.7 实验	75	6.3.8 控制流程和设计者的参与	101
4.7.1 系统描述的获取比较	75	6.3.9 典型系统配置	102
4.7.2 系统描述的理解比较	76	6.4 基本划分算法	102
4.7.3 系统描述的量化比较	76	6.4.1 随机映射	102
4.7.4 设计质量比较	77	6.4.2 层次化结群	102
4.8 结论	78	6.4.3 多级结群	104
4.9 练习	78	6.4.4 成组移动	105
第 5 章 转换成 VHDL	81	6.4.5 比率切割	107
5.1 引言	81	6.4.6 模拟退火	108
5.2 状态迁移	82	6.4.7 遗传进化	109
5.3 消息传递通信	84	6.4.8 整数线性规划	110
5.3.1 阻塞式消息传递	84	6.5 硬件功能划分	110
5.3.2 非阻塞式消息传递	86	6.5.1 Yorktown 硅编译器	111
5.4 并发	86	6.5.2 BUD	113
5.4.1 数据流	87	6.5.3 Aparty	115

6.5.4 其他技术	117	7.5.2 Aparty	162
6.6 软硬件划分算法	118	7.5.3 Vulcan	162
6.6.1 贪心算法	118	7.5.4 SpecSyn	162
6.6.2 爬山算法	119	7.6 结论和发展方向	164
6.6.3 二分约束搜索算法	120	7.7 练习	164
6.7 系统功能划分	120	第8章 设计描述细化	167
6.7.1 Vulcan	120	8.1 引言	167
6.7.2 Cosyma	121	8.2 细化变量群组	167
6.7.3 SpecSyn	122	8.2.1 变量折叠	167
6.7.4 其他技术	123	8.2.2 存储地址转换	168
6.8 折中的探索	124	8.3 通道细化	169
6.9 结论和发展方向	125	8.3.1 通道和总线的表征	169
6.10 练习	125	8.3.2 问题的定义	170
第7章 设计质量评估	127	8.3.3 总线生成	170
7.1 引言	127	8.3.4 协议生成	176
7.1.1 精确性与速度	128	8.4 解决访问冲突	178
7.1.2 评估的保真度	128	8.4.1 仲裁模型	178
7.2 质量度量	129	8.4.2 仲裁方案	179
7.2.1 硬件成本度量	129	8.4.3 仲裁器生成	180
7.2.2 软件成本度量	130	8.5 细化不兼容接口	181
7.2.3 性能度量	130	8.5.1 问题的定义	182
7.2.4 其他度量	134	8.5.2 通信协议描述	182
7.3 硬件评估	135	8.5.3 接口进程生成	184
7.3.1 硬件评估模型	135	8.5.4 协议兼容的其他方法	189
7.3.2 时钟周期评估	136	8.6 细化软件/硬件接口	191
7.3.3 控制步评估	141	8.6.1 目标体系结构	192
7.3.4 执行时间评估	145	8.6.2 变量分配	192
7.3.5 通信速率评估	147	8.6.3 接口生成	194
7.3.6 面积评估	148	8.6.4 数据访问细化	195
7.3.7 引脚评估	156	8.6.5 控制访问细化	197
7.4 软件评估	156	8.7 结论和发展方向	198
7.4.1 软件评估模型	156	8.8 练习	199
7.4.2 程序执行时间	159	第9章 系统设计方法学	201
7.4.3 程序存储大小	160	9.1 引言	201
7.4.4 数据存储大小	160	9.2 基本概念	201
7.5 系统级工具的评估技术	161	9.3 设计方法学举例	201
7.5.1 BUD	161	9.3.1 当前的惯例	204

9.3.2 系统级方法学	205	9.5 系统设计的概念化环境	213
9.4 通用综合系统	206	9.6 结论和发展方向	215
9.4.1 系统综合	208	9.7 练习	216
9.4.2 ASIC 综合	209	附录 A 应答机的自然语言描述	217
9.4.3 逻辑综合和时序综合	211	附录 B 应答机的 SpecCharts 描述	219
9.4.4 物理设计	212	参考文献	233
9.4.5 软件综合	212	术语解释	247
9.4.6 系统数据库	213	索引	249

第1章 引言

在过去的20年里,设计一个系统所需经过的实际步骤没有实质性的变化。另一方面,各设计步骤的侧重点却有了显著变化:由于较后的阶段已或多或少地实现了自动化,设计者越来越关注于系统设计过程较早期的更抽象阶段。这种关注点的转换使设计者能够在更短的时间内设计出愈加复杂的系统。设计这样的复杂系统,达到功能正确性要比达到芯片面积或程序所占内存最小化要重要且复杂得多。系统的功能性在早期的设计步骤中能得到最好的理解,此时很多的实现细节还没有被加入。这就是为什么在系统设计过程中早期的阶段至关重要。

在本章中,我们将从多种角度着重探究这种关注点的转变。我们将描述系统设计的相关步骤、设计表示的分类、各步骤的抽象层次,给出曾有的关注点及其如何转变的历史,描述下一次可能会向系统级的转变,以及在系统级所需的工具和方法学。

1

1.1 设计表示

对任何特定的产品,设计过程通常开始于对产品功能的概念化,结束于对产品制造蓝图的制订。在设计完成之前,很多不同的人员会参与其中。

例如,需要市场部门研究市场需求和确定对新产品的要求。需要一个首席设计师将这些要求转换为产品的体系结构。技术专家介入选择采用的工艺、可能需要的组件和供应者,而负责计算机辅助设计和计算机辅助软件工程的小组必须获得或开发出支持产品各部分设计的工具。负责设计的小组将制订出蓝图,指明如何从可获得的并采用了所选工艺的组件来制造出产品。软件工程师将为产品所采用的处理器编写代码。测试工程师需要制定测试策略和测试向量以确定产品的可靠性,而制造工程师则需要为产品的实际制造定义机器操作和制定工厂生产进度表。

每个小组都是从各自的角度看待产品,需要特定的信息支持其特定的工作。这样,每个产品、进而每个设计,必然有若干个不同的表示或视角,侧重于不同类型的信息。对于单一的表示也是这样,这时随着设计的周期进展可获得不同层次的细节。

3种最常采用的表示法分别针对产品的行为、结构和物理方面。

行为表示 将设计简单地看做黑盒,并将其行为描述为输入值和终止时间的函数。换句话说,行为表示描述系统的功能性,但不涉及任何实现。行为表示定义了黑盒如何对输入值的任何组合做出响应,但不提供关于如何设计这个黑盒的任何指示。

2

结构表示 与行为表示比较而言,开始解答一些设计问题,因为结构表示将黑盒定义成一组组件及其互联关系。换句话说,结构表示着重于描述产品的实现。虽然黑盒的功能性可从互联的组件中得出,但并不明确描述功能性。

物理表示 承载了设计的更进一步的实现,以结构化表示描述了组件的特点。例如,物理表示会提供每个组件的尺寸和位置,以及组件之间互联的物理特性。这样,结构表示提供了设计的互联性,而物理表示则描述了互联组件之间的空间关系,描述了被制造设计系统的重量、大小、散热、功耗以及每个输入输出引脚的位置。

一般来说,设计系统的过程应遵循从行为表示到结构表示再到物理表示的路线,并沿此路

线逐渐获得实现细节。如上所述,虽然我们需要这些实现细节来制造产品,但这些细节也会引起系统功能性的模糊,阻碍了设计者确保系统正常工作的努力。举一个例子:考虑一个能做 32 位数字加减的简单系统。行为表示仅包含两个等式: $a := a + b$ 和 $a := a \times b$ 。而结构表示则可能包含若干互联的寄存器、算术单元以及多路选择器,使得系统功能性难以辨别,尤其组件数量很多或者某组件的功能仅被部分使用时更是如此。因此,如果要强调功能正确性这个正在不断变成至关重要的问题,我们就应该承认:当工作于行为表示而不是工作于结构或物理表示时,设计者将会更成功从而设计出更好的产品。

3

1.2 抽象级别

在前一部分中,我们描述了最常见的几种设计表示类型,即行为、结构和物理表示。到此,我们可以进入下一个阶段,并注意到在电子系统的设计中,每种表示类型适合于若干不同的抽象级别或者粒度。不同的抽象级别基于它们所使用的对象类型进行区分,归入四类:晶体管级、门级、寄存器级和处理器级。在图 1-1 中总结了这些不同的抽象级别与各种表示类型的关系。

级 别	行 为 表 示	结 构 组 件	物 理 对 象
晶体管	微分方程,电流-电压关系图	晶体管,电阻,电容	模拟和数字单元
门	布尔方程,有限状态机	门,触发器	模块,元件
寄存器	算法,流程图,指令集,广义有限状态机	加法器,比较器,寄存器,计数器,寄存器文件,队列	微芯片,专用集成电路
处理器	可执行描述程序	处理器,控制器,存储器,专用集成电路	印刷电路板,多芯片模块

图 1-1 设计表示和抽象级别

根据图 1-1,在晶体管级的主要组件是晶体管、电阻和电容。这些对象可组合起来,形成满足给定功能的模拟电路和数字电路。在该级别上,功能通常用一组微分方程或者某类电流-电压关系描述。最后,这样一种电路的物理表示,称为单元,将包含晶体管级元件和它们之间的连线。这种单元通常根据它们所包含元件的版图来定义。

4

在门级,主要的元件是逻辑门和触发器。逻辑门是用于执行布尔运算的特殊电路,如“或”和“与”。触发器是基本存储单元,每个仅能存储一位信息。这些门和触发器表示典型的数字单元。可将这些独立组件结合起来,安置在硅片表面,以构成算术和存储模块,其行为可通过逻辑方程和有限状态机图描述。

在寄存器级的主要元件是由门和触发器,如加法器、比较器、多路选择器、计数器、寄存器、寄存器文件、数据缓冲器和队列设计而成的算术和存储单元。每个寄存器级元件是一个物理对象,具有固定尺寸、固定传播时间,并且在模块边界上其输入和输出有固定位置。寄存器级元件可用于微芯片设计,该设计可用流程图、指令集、广义有限状态机或状态表描述。

最后,最高抽象级别称为处理器级,因为该级别的基本组件是处理器、存储器、控制器、接口,以及称为专用集成电路(application-specific integrated circuit, ASIC)的定制微芯片。一个或多个这样的组件可安置于一个印刷电路板(printed-circuit board, PCB)上,微芯片焊接于其上,并通过印制于板上的导线连接起来。为降低板的尺寸,可用硅基片代替 PCB 用于微芯片

的连接,这种情况下的封装称为多芯片模块(multi-chip module,MCM)。由处理器级组件组合而成的系统可用若干种不同的方式描述其行为:如采用自然语言,采用以硬件描述语言方式的可执行描述,或者采用算法或程序设计语言程序。

现在,重要的是记住这样一个事实,设计者只能将他们的努力关注于他们所能理解的系统级别上,这个级别很大程度上决定于需要确保对象数量相对小。例如,一个设计者可能理解一个包含 10 个布尔方程的系统,但肯定不能理解一个包含 1 万个方程的系统。在后一种情况下,他将不得不向高抽象级别转移直至达到系统能表示成可管理的对象数量,如表示成 10 个算法。在较低抽象层次上,只有当我们将系统划分为小块并在许多设计者中分配时,或者当我们采用自动工具建造它时,系统才是可管理的。幸运的是,随着新的在较低层次上的设计工具出现,设计者能自由地关注于高级别,而在高级别做出决策要比在较低级别对质量的影响大得多。

5

每个电子系统在经历从概念化到制造的设计过程中,不可避免地要经过大部分这些抽象级别。在设计过程中的这组特定任务,它们执行的特定顺序和每个任务执行过程中要使用的 CAD 工具,称为设计方法学。在下一部分中,我们将简要讨论过去和现在在工业环境中曾占支配地位的设计方法学。

1.3 当前的设计方法学

在过去的 25 年中,大多数 ASIC 和系统生产厂家采用一种基于“捕获-模拟”(capture-and-simulate)的设计方法学。这种设计方法学开始于对产品的一组特定需求,该需求通常由市场部门提供。由于这些需求不能囊括有关产品实现的任何信息,故由首席设计师组成的小组提出一个芯片体系结构的粗略结构图,作为虽不完备但初步的系统描述。在某些情况下,这个初步结构图在提交给逻辑和版图设计者小组之前还要被进一步细化,而逻辑和版图设计者的任务是将每个功能模块转换为逻辑或电路原理图(最终被原理图捕获工具所捕获),通过模拟方法对功能、时延及故障覆盖率进行验证。这个被捕获的原理图,在门阵列技术中可用于驱动物理设计工具进行门的布局和布线,或在定制技术中用于在布局和布线前将门映射到标准和定制单元。

仅仅在最近几年,逻辑综合才被公认为是设计过程构成整体所必需的部分,而这种认可导致了在设计方法学上的革命性变化,因为“捕获-模拟”方法正逐步让位于“描述-综合”(describe-and-synthesize)方法学。这个新方法学的优势是它允许我们用纯行为形式描述一个设计,而不包括实现细节。特别地,我们能采用布尔方程和有限状态机图描述设计。在这个方法学中,设计结构采用 CAD 工具自动综合生成,而不是用手工综合而成,因为除非对于简单电路,手工综合是一项非常乏味的工作。

6

“描述-综合”方法学可应用于若干抽象层次。在门级,功能和控制单元可采用逻辑综合(logic synthesis)方法进行综合。例如,功能单元如 ALU、比较器及多路选择器等,可用布尔方程描述,然后通过两个阶段进行综合。第一个阶段,称为逻辑最小化(logic minimization),使布尔方程中“与”和“或”运算符的数量(或者在相等意义上,文字的数量)达到最小化,同时满足代价和时间约束。第 2 个阶段,称为工艺映射(technology mapping),将这些已最小化的布尔方程采用逻辑门实现,这些逻辑门来自于选定工艺技术的门库。

另一方面,控制单元采用有限状态机图定义,然后通过两个阶段综合。第 1 个阶段,称为