

计算机科学丛书

嵌入式计算系统设计原理

(美) Wayne Wolf 著

(普林斯顿大学)

孙玉芳 梁 彬 罗保国 谢 谦 等译

孙玉芳 梁 彬 校

本书附盘可从本馆主页 <http://lib.szu.edu.cn/>
上由“馆藏检索”该书详细信息后下载,
也可到视听部复制



机械工业出版社
China Machine Press

Wayne Wolf: Computers as Components: Principles of Embedded Computing System Design.

ISBN: 1-55860-541-X

Copyright © 2001 by Morgan Kaufmann. All rights reserved.

Copyright © 2001 by Harcourt Asia Pte Ltd. All rights reserved.

Printed in China by Harcourt Asia Pte Ltd. under special arrangement with China Machine Press. This edition is authorized for sale in China only, excluding Hong Kong 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.

本书中文简体字版由Harcourt Asia Pte Ltd. 授权机械工业出版社在中国大陆境内独家发行。本版仅限在中国境内（不包括香港特别行政区及台湾）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。

Harcourt Asia Pte Ltd.

583 Orchard Road, #09-01 Forum, Singapore 238884

Tel: (65) 737-3593 Fax: (65) 734-1874

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the publisher.

本书任何部分之文字及图片，如未获得本公司之书面同意不得用任何方式抄袭、节录或翻印。

本书版权登记号：图字：01-2001-2211

图书在版编目（CIP）数据

嵌入式计算系统设计原理 /（美）沃尔夫（Wolf, W.）著；孙玉芳等译. —北京：机械工业出版社，2002.2

（计算机科学丛书）

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

ISBN 7-111-09640-1

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

中国版本图书馆CIP数据核字（2001）第088579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：杨海玲

北京第二外国语学院印刷厂印刷·新华书店北京发行所发行

2002年2月第1版第1次印刷

787mm×1092mm1/16·28.75印张

印数：0 001—5 000册

定价：65.00元（附两张光盘）

凡购本书，如有倒页、脱页、缺页，由本社发行部调换

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”的总规划之下出版三个系列的计算机教材：针对本科生的核心课程，剔抉外版菁华而成“国外经典教材”系列；对影印版的教材，则单独开辟出“经典原版书库”；定位在高级教程和专业参考的“计算机科学丛书”还将保持原来的风格，继续出版新的品种。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。

“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。

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) 68995265

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
周克定
郑国梁
高传善
裘宗燕

王 珊
吕 建
李伟琴
陆丽娜
周傲英
施伯乐
梅 宏
戴 葵

冯博琴
孙玉芳
李师贤
陆鑫达
孟小峰
钟玉琢
程 旭

史忠植
吴世忠
李建中
陈向群
岳丽华
唐世渭
程时端

史美林
吴时霖
杨冬青
周伯生
范 明
袁崇义
谢希仁

译者序

今天,嵌入式系统带来的工业年产值已超过了1万亿美元,它正在成为信息技术(IT)产业争夺的重点之一。据统计,每年只有10%~20%的计算机芯片用于台式机或笔记本电脑等,这就意味着每年有10~20亿块CPU是为嵌入式系统设计和制造的;每年有超过1万个新的嵌入式系统计划产生,它们被广泛地应用到工业控制系统、仿真系统、医疗仪器、信息家电、通信设备等众多领域中。尤其是在最近几年,嵌入式设备不断地进入到新的应用领域,比如PDA、手持设备、AutoPC、智能电话和机顶盒(STB)等。在中国有8000万有线电视网用户,1亿以上的有线电话网用户,为其发展提供了巨大市场。机顶盒不仅可使模拟电视接收数字电视节目,而且可以上网,还可能成为未来家庭的控制中心。掌上电脑在未来几年内也将在我国得到快速发展,估计5年内总销量将达到3000万台。从功能上看,掌上电脑(或PDA)将扩充通信功能,甚至会具有手机和寻呼机的功能。另外,手机和寻呼机上网将成为新的趋势,车载盒随着全球定位技术的成熟也会得到广泛的应用。

针对如此巨大的市场,围绕嵌入式系统展开研究和开发也就成了计算机软硬件技术发展最活跃的方向之一。嵌入式系统的研发和生产也给我国信息产业实现跨越式发展提供了良好的机遇。中国科学院软件研究所和以其技术为依托的中科红旗软件技术有限公司的同仁们几年前就已涉足嵌入式系统的研究和开发,尽管我们已经基于Linux研制出了一组嵌入式系统及其产品,但是在研究和开发过程中我们感到经验缺乏,有许多问题迟迟得不到很好解决,延缓了研发工作的进展。正是在这种情况下,我们学习了美国普林斯顿大学Wayne Wolf教授所著的《*Computers as Components: Principles of Embedded Computing System Design*》一书,它给我们提供了许多有益的指导。

我们在学习过程中,感到本书对于嵌入式这一新领域的研发者有巨大帮助,所以在机械工业出版社的大力支持下,我们把本书译成中文版,希望与大家分享。

本书重点讲述嵌入式计算这一新兴学科的基本原理和技术。它不是详细讨论某一种微处理器,尽管本书以ARM和SHARC作为嵌入式系统微处理器重点讲述的对象;也不是讲述某种语言和编程方法,尽管在本书中讲述了相关的汇编语言和高级语言(如C)的编程工作。本书寻求更抽象的途径来研究嵌入式系统,但是这种研究既是系统的也是实际的,特别是讲述了大量实际例子,以利于读者掌握。

本书的另一特点是,在讨论原理的同时,讲述原理的具体应用。本书的主要章节都是一章一个重点,遍及构造嵌入式系统相关的指令系统、CPU、嵌入式计算平台、程序设计和分析、进程和操作系统、硬件加速器、网络、系统设计技术等各主要方面。每章后列出了供进一步学习的推荐读物,以及习题和实习题,有利于读者进一步领会该章陈述的内容。

由于嵌入式计算系统是一个新的领域,相关的书籍即使在信息技术发达的美国也不多见。正如密歇根大学电子工程和计算机科学系教授Lynn Conway在序中所说的本书是“嵌入式计算系统设计的第一本教科书”。它涉及的范围较广,许多术语是新的。加上我们以前的学习是

为了“致用”，在某种程度上是为了“救急”，难免有理解不够深入和系统之处。虽然我们一再努力，但是译文中难免有疏漏和错误之处。敬请读者指正。

最后感谢参与本书翻译和讨论的其他同仁，恕不在此一一列出。

孙玉芳

2001年8月



孙玉芳 男，1947年生，江苏省张家港人。中国科学院软件研究所副所长、博士生导师、研究员。1970年毕业于北京大学数学力学系，1978年考入中科院研究生院，1981年获计算机软件硕士学位。社会兼职中国中文信息学会副理事长，中国信息技术标准化委员会委员，北京市政府顾问，中国软件行业协会UNIX分会秘书长，国际开放系统互连亚洲大洋洲AOW技术协调组多国语言处理专委主席，香港特区大学研究资助局特邀评委，厦门大学、云南大学兼职教授。先后多次在新加坡、日本、美国等国家和香港地区的大学、科研单位和公司作访问学者、讲学和合作研究开发。在计算机软件特别是操作系统、中文信息处理、大型数据库和网络工程方面有精深的研究。在国内率先引进、推广和开发UNIX系统，实现该环境下的中文处理，并主持或参与主持完成了十余项国家和中科院科技奖攻关项目及数十项大型应用工程，取得了多项具有国际和国内先进水平的成果，先后获得国家科技进步奖（二等、三等各2项）、部委科技进步奖（电子工业部特等1项、中科院一等1项、二等2项、国家技术监督局一等1项）计九项。科技专著《操作系统原理与应用》、《实用C语言程序设计教程》等22本；“文件系统的设计与实现”、“UNIX系统中文应用环境的开发”等论文280余篇；《UNIX操作系统》等编译著122本。指导博士后、博士生、硕士生60余名。享受国务院颁发的政府特殊津贴。

序

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、耐耗系统、网络系统、智能家电、工业过程系统、高级自动化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们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的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。虽然有关硬件构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到的复杂性以及如何去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

前言

微处理器很长时间以来已经是我们生活的一部分，然而，微处理器对于真实复杂功能所产生的强有力影响还只是近几年的事。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取得最高性能，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。另外一个例子是多进程的系统设计。对于台式机的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现在在基于微处理器的嵌入式系统中能得到很普遍的应用。

其他的挑战对于嵌入式计算来说是全新的。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许多情况下是十分重要的。另外一种挑战是期限驱动的程序设计。嵌入式计算机常常对程序完成的期限做硬性限制；这种形式的约束在台式机世界里是罕见的。当嵌入式处理器变得更快时，高速缓存和其他CPU元素亦使得执行时间难于预测。然而，通过仔细分析和巧妙编程，我们可以把嵌入式程序设计得具有可预测的执行时间，甚至面对不可预测的系统部件如高速缓存也如此。

幸运的是，有许多处理复杂嵌入式系统所面临的挑战的工具：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那样的技术在构造大型实际的复杂的嵌入式系统中是非常重要的。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在错误处往回走再从更大的视角研究问题。第二，用于建造嵌入式系统的构件是经常变化的，但其原理不变。一旦你理解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构件。

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。这种传

统方法的重点是教授如何构建一个完整系统的硬件和软件。结果导致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。本书利用每次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，为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上，而是试图用一般性的例子来推出更一般性的应用原理。本人认为这种方法对于教学更为容易，而从长远角度来看对于学生也更有用。教学更容易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用于纸笔上的练习、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将大多使用不同于那些在学校里的组件和设施。一旦学生学会了基础知识，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。某些硬件构造经验是非常有价值的；我相信每一个学生都应该知道烧制塑封集成电路块的气味。但我极力让你避免构建硬件。如果你花费太多的时间去构建一个硬件平台，那么你就没有足够时间去编写有趣的程序。一个实际问题是，大多数班级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存硬件平台来学习硬件。编制复杂嵌入式系统的程序的实践将教会学生相当多的有关硬件的知识，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）中包括辅助操作、导师手册、实验材料、到相关Web站点的链接以及与包含习题解答的口令保护的ftp站点的连接。

目 录

出版者的话	
专家指导委员会	
译者序	
序	
前言	
第1章 嵌入式计算	1
1.1 引言	1
1.2 复杂系统和微处理器	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 嵌入式计算系统设计所面临的挑战	5
1.3 嵌入式系统的设计过程	6
1.3.1 需求	8
1.3.2 规格说明	11
1.3.3 体系结构设计	11
1.3.4 设计硬件构件和软件构件	13
1.3.5 系统集成	13
1.4 系统设计的形式化方法	13
1.4.1 结构描述	14
1.4.2 行为描述	19
1.5 设计示例：模型火车控制器	21
1.5.1 需求	22
1.5.2 概念性规格说明	22
1.5.3 详细的规格说明	25
1.5.4 本节所学内容	31
1.6 本书导读	31
1.6.1 第2章：指令系统	31
1.6.2 第3章：CPU	32
1.6.3 第4章：嵌入式计算平台	32
1.6.4 第5章：程序设计和分析	32
1.6.5 第6章：进程和操作系统	33
1.6.6 第7章：硬件加速器	33
1.6.7 第8章：网络	34
1.6.8 第9章：系统设计技术	34
1.7 小结	35
推荐读物	35
习题	35
实习题	36
第2章 指令系统	37
2.1 引言	37
2.2 预备知识	37
2.2.1 计算机体系结构分类	37
2.2.2 汇编语言	39
2.3 ARM处理器	40
2.3.1 处理器和存储器组织	40
2.3.2 数据操作	41
2.3.3 控制流	47
2.4 SHARC处理器	53
2.4.1 存储器组织	53
2.4.2 数据操作	54
2.4.3 控制流	60
2.4.4 指令的并行性	64
2.5 小结	64
推荐读物	65
习题	65
实习题	67
第3章 CPU	68
3.1 引言	68
3.2 输入输出编程	68
3.2.1 输入输出设备	68
3.2.2 输入输出原语	69
3.2.3 忙等I/O	70
3.2.4 中断	71
3.3 管态、异常和陷阱	82
3.3.1 管态	82
3.3.2 异常	82
3.3.3 陷阱	83

3.4 协处理器	83	4.5.2 设备接口	136
3.5 存储系统机制	83	4.6 用微处理器设计	137
3.5.1 高速缓存	84	4.6.1 系统的体系结构	137
3.5.2 存储管理单元和地址转换	89	4.6.2 硬件设计	138
3.6 CPU的性能	93	4.6.3 PC作为平台	139
3.6.1 流水线技术	93	4.7 开发与调试	142
3.6.2 超标量执行	96	4.7.1 开发环境	142
3.6.3 高速缓存	97	4.7.2 调试方法	143
3.7 CPU的功耗	97	4.7.3 调试难点	146
3.8 设计示例: 数据压缩器	101	4.8 制造测试	147
3.8.1 需求和算法	101	4.9 设计示例: 闹钟	150
3.8.2 规格说明	103	4.9.1 需求	150
3.8.3 程序设计	105	4.9.2 规格说明	151
3.8.4 测试	109	4.9.3 系统体系结构	154
3.9 小结	110	4.9.4 构件设计与测试	154
推荐读物	111	4.9.5 系统集成与测试	155
习题	111	4.10 小结	155
实习题	113	推荐读物	155
第4章 嵌入式计算平台	114	习题	155
4.1 引言	114	实习题	157
4.2 CPU总线	114	第5章 程序设计和分析	158
4.2.1 总线协议	114	5.1 引言	158
4.2.2 DMA	119	5.2 程序设计	158
4.2.3 系统总线配置	121	5.2.1 设计范型	158
4.2.4 ARM总线	123	5.2.2 嵌入式系统设计范型	159
4.2.5 SHARC总线	123	5.3 编程模型	162
4.3 存储设备	125	5.3.1 数据流图	162
4.3.1 存储设备组织	125	5.3.2 控制/数据流图	164
4.3.2 随机存储器	125	5.4 汇编和链接	166
4.3.3 只读存储器	129	5.4.1 汇编程序	166
4.4 I/O设备	130	5.4.2 链接	169
4.4.1 定时器和计数器	130	5.5 基本编译技术	170
4.4.2 A/D和D/A转换器	131	5.5.1 语句翻译	171
4.4.3 键盘	131	5.5.2 过程	175
4.4.4 发光二极管	133	5.5.3 数据结构	176
4.4.5 显示器	133	5.5.4 表达式简化	177
4.4.6 触摸屏	134	5.5.5 死代码清除	177
4.5 组件接口	135	5.5.6 过程内嵌	178
4.5.1 存储器接口	135	5.5.7 循环变换	178

5.5.8 寄存器分配	180	6.5.3 进程中的计时需求	231
5.5.9 调度	183	6.5.4 进程间通信	233
5.5.10 指令选择	184	6.5.5 操作系统的其他功能	236
5.5.11 理解并使用你的编译程序	185	6.6 调度策略	237
5.5.12 解释程序与JIT编译程序	185	6.6.1 单一速率调度	237
5.6 执行时间的分析与优化	186	6.6.2 期限最近者优先调度	241
5.6.1 程序性能的元素	187	6.6.3 RMS和EDF比较	244
5.6.2 跟踪驱动的性能分析	189	6.6.4 对建模假设的进一步分析	244
5.6.3 优化执行速度	194	6.6.5 其他POSIX调度策略	246
5.7 能量和功率的分析和优化	195	6.7 进程间通信机制	246
5.8 程序长度的分析及优化	198	6.7.1 信号	246
5.9 程序验证和测试	200	6.7.2 UML中的信号	248
5.9.1 白盒测试	200	6.7.3 共享内存通信	248
5.9.2 黑盒测试	206	6.7.4 基于消息的通信	250
5.9.3 评估功能测试	206	6.8 评估操作系统的性能	251
5.9.4 性能测试	207	6.9 进程的功率优化策略	253
5.10 设计示例: 软件调制解调器	207	6.10 设计示例: 电话应答机	255
5.10.1 工作原理和需求	208	6.10.1 工作原理和需求	255
5.10.2 规格说明	209	6.10.2 规格说明	257
5.10.3 系统体系结构	210	6.10.3 系统体系结构	259
5.10.4 构件设计和测试	210	6.10.4 构件设计和测试	261
5.10.5 系统集成和测试	210	6.10.5 系统集成和测试	261
5.11 小结	211	6.11 小结	261
推荐读物	211	推荐读物	262
习题	211	习题	262
实习题	217	实习题	266
第6章 进程和操作系统	218	第7章 硬件加速器	267
6.1 引言	218	7.1 引言	267
6.2 多任务和多进程	219	7.2 CPU和加速器	267
6.2.1 多速率系统	220	7.2.1 为什么使用加速器	268
6.2.2 早期多任务技术: 协同例程	221	7.2.2 加速器的设计	269
6.3 进程	222	7.3 加速系统的设计	270
6.4 文境切换	224	7.3.1 性能分析	270
6.4.1 协作多任务处理	224	7.3.2 系统体系结构框架	273
6.4.2 抢先式多任务处理	227	7.3.3 划分	275
6.4.3 进程和面向对象设计	228	7.3.4 调度和分配	276
6.5 操作系统	228	7.3.5 系统集成和调试	278
6.5.1 进程状态和进程调度	229	7.4 设计示例: 视频加速器	278
6.5.2 操作系统结构	230	7.4.1 算法和需求	278

7.4.2 规格说明	281	实习题	317
7.4.3 体系结构	282	第9章 系统设计技术	318
7.4.4 构件设计	283	9.1 引言	318
7.4.5 系统测试	284	9.2 设计方法学	318
7.5 小结	284	9.2.1 为什么研究设计方法学	318
推荐读物	284	9.2.2 设计流程	319
习题	284	9.3 需求分析	324
实习题	285	9.4 规格说明	325
第8章 网络	286	9.4.1 面向控制的规格说明语言	325
8.1 引言	286	9.4.2 高级规格说明	327
8.2 分布式嵌入体系结构	286	9.5 系统分析和体系结构设计	329
8.2.1 为什么使用分布式系统	287	9.6 质量保证	332
8.2.2 网络抽象模型	287	9.6.1 质量保证技术	334
8.2.3 硬件和软件体系结构	288	9.6.2 验证规格说明	335
8.2.4 报文传递编程	291	9.6.3 设计评审	337
8.3 嵌入式系统网络	292	9.6.4 测量驱动的质量保证	338
8.3.1 PC总线	292	9.7 设计示例: 电话用户交换机	341
8.3.2 CAN总线	296	9.7.1 工作原理	341
8.3.3 SHARC链接端口	298	9.7.2 系统体系结构	343
8.3.4 以太网	300	9.8 设计示例: 喷墨打印机	344
8.3.5 Myrinet	301	9.8.1 硬件设计	346
8.3.6 因特网	301	9.8.2 软件设计	349
8.4 基于网络的设计	303	9.9 设计示例: 个人数字助理	351
8.4.1 通信分析	303	9.10 设计示例: 机顶盒	353
8.4.2 系统性能分析	306	9.11 片上系统	355
8.4.3 硬件平台设计、分配和调度	308	9.12 小结	357
8.5 可达Internet系统	309	推荐读物	357
8.6 设计示例: 电梯控制器	310	习题	357
8.6.1 工作原理和需求	310	实习题	358
8.6.2 规格说明	312	附录A UML符号	359
8.6.3 体系结构	314	附录B 硬件设计附注	363
8.6.4 测试	314	术语表	371
8.7 小结	315	参考文献	385
推荐读物	315	索引	395
习题	315		

第1章 嵌入式计算

- 我们为什么要在系统中嵌入微处理器。
- 嵌入微处理器的困难所在。
- 设计方法学。
- 用UML描述设计。
- 本书导读。

1.1 引言

本章我们确立学习嵌入式计算系统设计的层面。为了更好地理解设计过程，我们首先必须理解微处理器怎样以及为什么被用于控制、用户界面、信号处理和其他许多方面。现在，微处理器运用得十分广泛，以至于我们忽略了在某些领域如果没有它们工作将多么困难。

首先，我们要回顾微处理器在不同领域内的用途。接着，我们将讨论微处理器被应用于系统设计的主要原因——提供复杂的操作、快速的开发周期等等。接下来，在1.3节，我们将会通过对一个系统设计示例的详细分析来使大家理解设计一个系统的主要步骤。1.4节包含了对于详细说明嵌入式系统的技术的深入考察，我们将在本书通篇使用这些规格说明技术。在1.5节中，我们将设计一个模型火车控制器，我们试图通过这个模型来让大家了解如何运用我们在1.4节中介绍并在本书后续章节中不断用到的规格说明技术。1.6节我们将提供本书每一章的简介。

1.2 复杂系统和微处理器

什么是嵌入式计算机系统？不严格地说：它是任意包含一个可编程计算机的设备，但是这个设备不是作为通用计算机而设计的。因此，一台个人电脑并不能称之为嵌入式计算系统，尽管个人电脑经常被用于搭建嵌入式计算系统，就像在9.7节中将介绍的以个人计算机为基础的电话系统一样。但是，一台包含了微处理器的传真机或时钟就可以算是一种嵌入式计算系统。

这就意味着，嵌入式计算系统的设计是一种十分有用的技术，它可以应用于多种类型的产品设计。微处理器在汽车、个人数字助理（PDA）甚至是家用电器领域都获得了广泛的运用。在许多领域，设计者必须做出判断可以在哪些部分使用微处理器，设计出一套带有输入输出设备的硬件平台以满足所要完成的任务，并设计出相应的软件来执行相关的操作。同机械设计和热力学一样，计算机工程是一门基础学科，它可以被运用到其他许多领域。当然，嵌入式计算系统并不是独立的，在嵌入式计算系统设计过程中遇到的挑战并不是计算机工程中的问题——例如，可能是机械或模拟电信号的问题。本书中我们所关心的是嵌入式计算机本身，所以我们将专注于我们所涉及的硬件和软件，这些硬件和软件能够使我们在最终产品上实现所期望的功能。

1.2.1 嵌入式计算机

在计算任务早期，计算机就已经嵌入到应用中了。一个例子就是Whirlwind，它是MIT

(美国麻省理工学院)在20世纪40年代到50年代之间设计的一台计算机。Whirlwind是第一台支持实时操作的计算机,并且它被设计用于控制一台飞行仿真设备。虽然Whirlwind在物理尺寸上要远大于今天的计算机(比如,它包含了4000多个真空管),但它无论从组成部分还是从系统设计方面来说都完全是为了满足实时嵌入式计算的要求而设计的。从计算机发展的早期开始,就被用来替代机械控制器或人工控制器了,比如,在20世纪40年代末,就有人计划用计算机控制化学反应过程[sto95]。

微处理器是一种单芯片的中央处理器。从20世纪70年代起,VLSI(超大规模集成电路)技术的运用就使我们可以将整个中央处理器集成到一块芯片上,但是当时的中央处理器所能做的还十分有限。世界上第一台微处理器Intel 4004就是为嵌入式应用而设计的,即计算器。该计算器并不是通用计算机,它仅提供基本的算术运算功能。然而,Intel公司的Ted Hoff在当时就意识到既然一台通用计算机可以通过编程使它实现所需要的功能,那么微处理器也可以通过编程运用到别的产品上。因为集成电路的设计在过去和现在都是一项昂贵和费时的过程,所以通过重新设计软件而重用硬件的能力就成了关键。HP-35是第一种手持计算器[Whi72],它可以计算超越函数。它于1972年问世,它使用了几块芯片来完成中央处理器的功能,而不是采用集成在单一芯片上的微处理器。但是,它能够通过编写软件完成数学运算而不是通过设计数字电路来实现三角数学函数一类的计算,这是它成为成功的计算器设计的主要因素。

汽车设计师们在单片机问世后不久就开始使用微处理器了。微处理器在汽车设计中主要被用于控制发动机:决定什么时候打开火花塞、控制燃料和空气成分的比例等等。在汽车制造业流行着一种趋势,即用电子设备取代机械设备。对微处理器控制的发动机的大的冲击来自于两种几乎同时产生的发展:一是20世纪70年代的石油危机使消费者不得不在燃油上付出比以前高得多的费用;二是担心污染,各个国家都推出了更加严格的汽车尾气排放标准。要同时达到低油耗、低废气排放是一件十分困难的事情,并且要不降低发动机工作效率,为此,汽车制造商都倾向于采用微处理器控制发动机,因为只有在此基础上才能执行精密的控制算法。

微处理器按它们的精度可以分成几个不同的等级,这种等级的划分是根据它们的字长来确定的。8位的微控制器通常是低价应用设计的,它们通常由集成的内存和输入/输出设备组成;16位微控制器被用于比较精密的应用,这些应用通常或是需要较长的字长或是独立的输入/输出设备和内存;32位的RISC微处理器提供很高的性能,它们被用于运算强度很大的应用中。

在知道微处理器可以分成这么多种类以后,我们就不再会为它们如此广阔的用途而感到奇怪了。在日常生活中,我们周围就有许多微处理器。一般的微波炉就至少内置了一块以上的微处理器以控制微波炉的工作;许多家庭现在都安装了比较先进的空调系统,这些空调系统可以根据一天当中不同时间动态地调节温度;现代照相机是一个微处理器应用的主要范例,佳能EOS 3型照相机内置了3块微处理器,其中一块32位RISC中央处理器主要负责自动聚焦以及眼控系统的处理[Pho 99]。

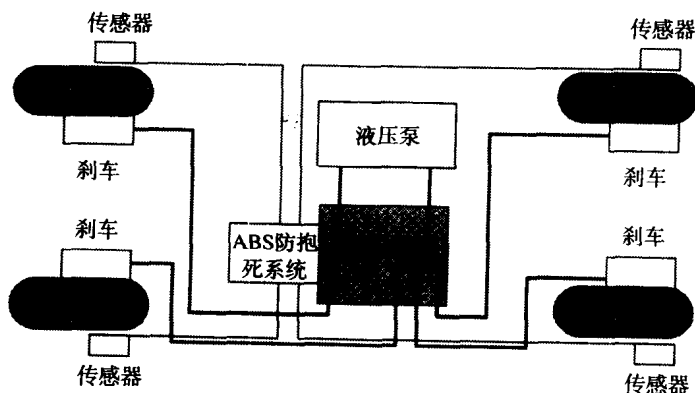
数字电视是嵌入式处理器的又一大应用领域。在某些情况下,专门设计特定的中央处理器来执行重要的算法,用在DirectTV上的SGS Thomson芯片组中专门为处理音频信号而设计的中央处理器就是这方面应用的一个很好的例子,该处理器被设计用于执行数字音频信号的

解码程序。人们愿意采用可编程中央处理器来代替硬件有两个原因：第一，可编程中央处理器的使用简化了系统的设计和调试，第二：用户可以对中央处理器进行升级，也可以将它们用于其他目的。

现在，一辆高级汽车上可能有上百块微处理器，即使是便宜的汽车上也采用了多块微处理器。这些微处理器中有些负责十分简单的工作，比如检测安全带是否系上；而有些则负责诸如点火或刹车之类十分关键的功能。应用示例1-1描述了在BMW 850i上使用的一些微处理器。

应用示例1-1 BMW 850i刹车和稳定控制系统 BMW 850i带有控制车轮的精密系统，一套防抱死系统（ABS）用于在刹车时避免车辆打滑；一套自动稳定控制（ASC+T）系统用于在车辆运行过程中控制发动机以保持车身的稳定。这些系统主动地控制着车辆的关键部分；作为控制系统，它们需要从汽车得到输入，同时向汽车输出控制信息。

让我们先来看看ABS防抱死系统。设计ABS的目的是在汽车制动的过程中，当车轮转动得太慢时，临时脱开刹车——因为当汽车车轮停止转动时，它就会开始打滑，从而变得十分难以控制。该系统在液压泵和刹车之间，液压泵为刹车提供能量，刹车系统如下图所示。这种连接方式使ABS可以调节刹车系统，使车轮不至于锁死。ABS系统通过安装在每个车轮上的传感器去探测它们的速度，然后由ABS系统去决定如何改变液压泵的压力以防止车轮打滑。



ASC+T系统的作用是在车辆运行过程中控制发动机的动力和刹车系统以保持车辆的稳定性。ASC+T系统控制着四个不同的系统：油门、定时点火、差速刹车以及（对于无级变速汽车而言）档位变换。ASC+T系统可以由驾驶员关闭，当汽车在令人讨厌的雪地上行驶时这就显得十分重要了。

ABS系统和ASC+T系统必须紧密地联系，因为ASC+T系统也和刹车系统之间相互关联。由于ABS系统比ASC+T系统要早几年产生，因此ASC+T系统和ABS系统之间的衔接是十分重要的，同其他电子设备的衔接也存在同样的问题。发动机和控制管理单元包括电子控制油门、数字发动机管理以及电子传输控制。ASC+T控制单元包含在两块印刷电路版上的两块微处理器，其中一块集成于逻辑相关构件上；另外一块集成于特定性能构件上。 ■

1.2.2 嵌入式计算应用的特点

与为个人电脑或工作站写的程序相比，嵌入式计算应用程序在许多方面要满足更多的要