

SEI 软件工程译丛

卡内基·梅隆大学软件工程研究所推荐·软件学院教材



并发与实时系统 软件设计

Software Design Methods
for Concurrent and Real-Time Systems

[美] 哈桑·戈玛 [Hassan Gomaa] 著
姜昊 周靖 译



清华大学出版社



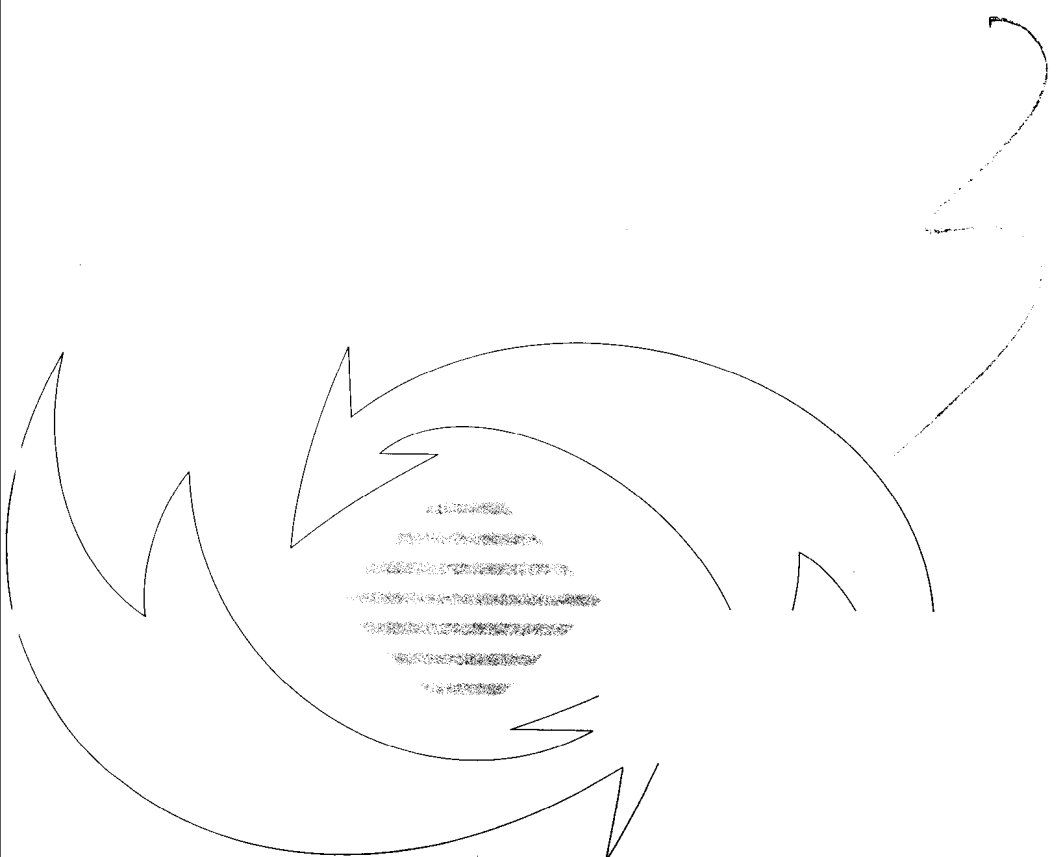
SEI 软件工程译丛
卡内基·梅隆大学软件工程研究所 著 · 软件学院教材



并发与实时系统 软件设计

Software Design Methods
for Concurrent and Real-Time Systems

[美] 哈桑·戈玛 [Hassan Gomaa] 著
姜 昊 周 靖 译



清华大学出版社
北 京

内 容 简 介

随着微处理器价格的大幅下降和性能的显著提高, 并发、实时系统和分布式实时系统已成为一种成本低、效率高的解决方案。本书介绍了设计这些系统所涉及的重要概念, 概括了并发和实时系统的特征, 通过经典的案例对并发和实时系统进行了深入的剖析。

本书是卡内基·梅隆大学软件工程研究所推荐教材, 可供软件学院及高校相关专业本科生和研究生使用, 也适合业内人士参考。

Simplified Chinese edition copyright © 2003 by **PEARSON EDUCATION ASIA LIMITED and TSINGHUA UNIVERSITY PRESS.**

Original English language title from Proprietor's edition of the Work.

Original English language title: *Software Design Methods for Concurrent and Real-Time Systems*,
1 edition by Hassan Gomaa Copyright © 1993

EISBN: 0-201-52577-1

All Rights Reserved.

Published by arrangement with the original publisher, Pearson Education, Inc., publishing as Addison-Wesley Publishing Company, Inc.

This edition is authorized for sale only in the People's Republic of China (excluding the Special Administrative Region of Hong Kong and Macau).

本书中文简体翻译版由 Addison-Wesley Publishing Company, Inc. 授权给清华大学出版社在中国境内(不包括中国香港、澳门特别行政区)出版发行。

北京市版权局著作权合同登记号: 图字 01-2002-4499 号

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

图书在版编目(CIP)数据

并发与实时系统的软件设计 / [美] 戈玛著; 姜昊, 周靖译. —北京: 清华大学出版社, 2003
(SEI 软件工程译丛)

书名原文: *Software Design Methods for Concurrent and Real-Time Systems*

ISBN 7-302-06647-7

I. 并... II. ①戈...②姜...③周... III. 实时操作系统-软件设计 IV. TP311.5

中国版本图书馆 CIP 数据核字 (2003) 第 037063 号

出 版 者: 清华大学出版社

<http://www.tup.com.cn>

社总机: (010) 6277 0175

地 址: 北京清华大学学研大厦

邮 编: 100084

客户服务: (010) 6277 6969

文稿编辑: 姜汉鲁

印 刷 者: 北京牛山世兴印刷厂

发 行 者: 新华书店总店北京发行所

开 本: 787×960 1/16 印 张: 28 插 页: 1 字 数: 620 千字

版 次: 2003 年 7 月第 1 版 2003 年 7 月第 1 次印刷

书 号: ISBN 7-302-06647-7/TP·4975

印 数: 0001~4000

定 价: 59.00 元

出版说明

1984年,美国国防部出资在卡内基·梅隆大学设立软件工程研究所(Software Engineering Institute,简称SEI)。SEI于1986年开始研究软件过程能力成熟度模型(Capability Maturity Model,简称CMM),1991年正式推出了CMM 1.0版,1993年推出CMM 1.1版。此后,SEI还完成了能力成熟度模型集成(Capability Maturity Model Integration,简称CMMI)。目前,CMM 2.0版已经推出。

CMM自问世以来备受关注,在一些发达国家和地区得到了广泛应用,成为衡量软件公司软件开发管理水平的重要参考因素,并成为软件过程改进的事实标准。CMM目前代表着软件发展的一种思路,一种提高软件过程能力的途径。它为软件行业的发展提供了一个良好的框架,是软件过程能力提高的有用工具。

SEI十几年的研究过程和成果,都浓缩在由SEI资深专家亲自撰写的SEI软件工程丛书(SEI Series In Software Engineering)中。为增强我国软件企业的竞争力,提高国产软件的水平,经清华大学出版社和三联四方工作室共同策划,全面引进了这套丛书,分批影印和翻译出版。这套丛书采取开放式出版,不断改进,不断出版,旨在满足国内软件界人士学习国外经典软件工程高级教程的愿望。

“SEI 软件工程译丛” 编 委 会

主 任 周伯生

副 主 任 郑人杰

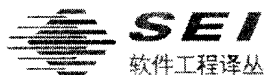
委 员 (按姓名拼音顺序排列)

董士海 顾毓清 王 伟

吴超英 尤晓东

执行委员 尤晓东

秘 书 廖彬山



总 序

——为清华大学出版社出版“SEI 软件工程译丛”而作

美国卡内基·梅隆大学软件工程研究所(CMU/SEI)是美国联邦政府资助构建的研究单位,由美国国防部主管。他们确认,为了保证软件开发工作的成功,由软件开发人员、软件采办人员和软件用户组成的集成化团队必须具有必要的软件工程知识和技能,以保证能按时向用户交付正确的软件。所谓“正确的”就是指在功能、性能和成本几个方面都能满足用户要求且无缺陷;所谓“无缺陷”就是指在编码后对软件系统进行了彻底的穷举测试并修复了所有的缺陷,或保证所编写的代码本身不存在缺陷。

CMU/SEI 为了达到这个目的,提出了创造、应用和推广的战略。这里的“创造”是指与软件工程研究社团一起,共同创造新的实践或改进原有的实践,而不墨守成规。这里的“应用”是指与一线开发人员共同工作,以应用、改进和确认这些新的或改进的实践,强调理论联系实际。这里的“推广”是指与整个社团一起,共同鼓励和支持这些经过验证和确认的、新的或改进的实践在世界范围内的应用,通过实践进行进一步的检验和提高。如此循环,往复无穷。

他们把所获得的成就归纳为两个主要领域。一个是倡导软件工程的实践,使软件组织在采办、构建和改进软件系统时,具有预测的能力与控制质量、进度、成本、开发周期和生产效率的能力。另一个是改进软件工程技术的实践,使软件工程师具有分析、预测和控制软件系统属性的能力,其中包括在采办、构建和改进软件系统时,能进行恰当的权衡,作出正确的判断和决策。CMU/SEI 通过出版软件工程丛书,总结他们的研究成果和实践经验,是推广这两个领域经验的重大举措。

SEI 软件工程丛书由 CMU/SEI 和 Addison-Wesley 公司共同组织出版,共分 4 个部分:计算机和网络安全(已出版了 2 本著作),工程实践(已出版了 8 本著作),过程改进和过程管理(已出版了 11 本著作),团队软件过程和个体软件

过程(已出版了3本著作)。前两者属于软件工程技术实践,后两者属于软件工程管理实践。目前这4个部分共出版了24本著作,以向软件工程实践人员和学生方便地提供最新的软件工程信息。这些著作凝聚了全世界软件工程界上百位开拓者和成千上万实践者的创造性劳动,蕴含了大量的宝贵经验和沉痛教训,很值得我们学习。

清华大学出版社邀请我和郑人杰教授共同组织 SEI 软件工程译丛编委会。清华大学出版社计划首先影印 6 本著作,翻译出版 15 本著作。据我所知,在 Addison-Wesley 公司出版的 SEI 软件工程丛书中,人民邮电出版社已经翻译出版了《个体软件过程》和《团队软件过程》,还拟影印出版《个体软件过程》和《软件工程规范》;电子工业出版社已经翻译出版了《净室软件工程的技术与过程》、《能力成熟度模型 CMM 1.1 指南》、《能力成熟度模型集成 CMMI》和《软件项目管理》;北京航空航天大学出版社已经翻译出版了《统计过程控制》。这些出版社共计影印 2 本著作,翻译出版 7 本著作。这样,可以预期我国在今年年底共可影印 8 本著作,翻译出版 22 本著作。各个出版社的有远见的辛勤劳动,为我们创造了“引进、消化、吸收、创新”的机遇。我们应该结合各自的实践,认真学习国外的先进经验,以大大提高我国软件工程的理论和实践水平。

在这套丛书中,特别值得一提的是,在过程工程领域被誉为软件过程之父的 Humphrey 先生所撰写的《软件过程管理》、《技术人员管理》、《软件工程规范》、《个体软件过程》、《团队软件过程》和《软件制胜之道》等 6 本著作,将于今年年内全部翻译出版,其中《软件过程管理》、《技术人员管理》、《软件工程规范》、《个体软件过程》和《软件制胜之道》这 5 本著作亦已经或将于今年年内影印出版。

《软件过程管理》是软件过程领域的开创性著作,是为软件公司经理和软件项目经理撰写的。用这本书提出的原理来指导软件开发,可以有效地按照预定进度得到高质量的软件,同时还可了解如何持续进行过程改进。美国 CMU/SEI 按照这本书提出的原理开发了能力成熟度模型,在国际上得到绝大多数国家的认可和广泛采用,是改进软件过程能力的有力武器。在信息技术迅速发展和企业激烈竞争的今天,能否持续改进过程往往决定企业的命运。

作为一个软件经理,在改进组织的能力之前,首先必须明确绝大多数软件问题是由管理不善所引起的。因此,要改进组织的性能,首先需要改进自己的管理模式。同时还要认识到软件开发是一项智力劳动,需要拥有掌握高技能和忘我工作的技术人员。因此,有效的软件管理需要充分注意技术人员的管理。

《技术人员管理》这本著作就是为达到这个目的而撰写的。高质量的技术

工作要求没有差错,这就要求人们高度专心和高度献身。因此要求人们对他所从事的工作不仅具有高度的责任感,而且具有浓厚的兴趣和高度的热忱。在当前知识经济群龙相争的今天,一个能激励人们进行创造性工作的领导群体,是众多竞争因素中最重要的因素。本书提供了大量的实用指南,可用来有效地改进工程人员、经理和组织的性能。

Humphrey 先生还认为这本书特别适合于在我国工作的软件经理。我国是一个人口大国,拥有大量能干的知识分子,而且信息领域的劳动力价格比国际市场的价格要低,因此吸引了许多国家到我国来投资。但若不提高人员的素质,不在产品质量和进度方面也狠下功夫,就不能在这方面持续保持优势。

《软件工程规范》是为编程人员撰写的。它精辟地阐述了个体软件过程(PSP)的基本原理,详尽地描述了人们如何来控制自己的工作,如何与管理方协商各项安排。在软件工程界,这本著作被誉为是软件工程由定性进入定量的标志。目前在世界范围内,有成千上万的软件工程技术人员在接受有关 PSP 的培训,以便正确地遵循 PSP 的实践、开发和管理工作计划,在他们承诺的进度范围内,交付高质量的产品。

《软件制胜之道》这本著作描述了团队软件过程的基本原理,详尽地阐述了在软件组织中如何应用 PSP 和 TSP 的原理以及它所能带来的效益。此外,虽然 CMM 同样适用于小型组织,但在其他著作中都没有描述如何应用 CMM 于个体或小型团队,这本书填补了这个空白。应该指出,如果一个组织正在按照 CMM 改进过程,则 PSP 和 TSP 是和 CMM 完全相容的。如果一个组织还没有按照 CMM 改进过程,则有关 PSP 和 TSP 的训练,可以为未来的 CMM 实践奠定坚实的基础。

在软件工程技术实践方面目前共出版了 10 本著作,其中《用商业组件构建系统》、《软件构架实践》和《软件构架评估——方法和案例研究》等 3 本著作详尽地阐述了软件构架的构建、实践和评估。鉴于是否有一个稳定的软件构架,对软件的质量和成本影响很大,因此如何获得一个良好的构架就成为当今软件界研究的重点。我相信这几本著作的出版,将对我国软件构架领域的研究与实践有重要的参考价值。此外,众所周知,计算机与网络的安全问题对信息系统的可靠使用关系极大,《CERT 安全指南——系统与网络安全实践》的出版将会对我国在这一领域的研究和实践起积极的促进作用。《风险管理——软件系统开发方法》、《软件采办管理——开放系统和 COTS 产品》、《项目管理原理》、《软件产品线——实践和模式》和《系统工程——基于信息的设计方法》等 5 本著作,分别从风险管理、软件采办、项目管理、软件产品线以及信息系统设计方法

等几个方面阐述了大型、复杂软件系统的开发问题,是有关发展软件产业的重要领域,很值得我国软件产业界借鉴。目前我们所处的时代是信息化时代,是人类进入能够综合利用物质、能量和信息三种资源的时代。千百年来以传统的物质产品的生产、流通、消费为基本特征的物质型经济,将逐步进入以信息产品的生产、流通、利用和消费为基本特征的知识型经济。在这个历史任务中,建造和广泛应用各类计算机应用系统是其公共特征。计算机软件是计算机应用系统的灵魂,没有先进的软件产业,不可能有先进的信息产业,从而也不可能建成现代化的知识型经济。

我们应该看到,在软件领域中我国在总体上离世界先进水平还有相当大的差距。但是,我们不能跟随他国的脚印,走他人的老路。我们应该抓住机遇,直接针对未来的目标,在软件工程技术和软件工程管理两个方面,注意研究 SEI 软件工程丛中倡导的原理和方法,联系实际,认真实践,并充分利用我国丰富优秀的人力资源和尊重教育的优良传统,大力培养各个层次的高质量的软件工程人员,使其具有开发各类大型、复杂软件系统的能力。我衷心地预祝清华大学出版社影印和翻译出版这套丛书,在把我国建设成为一个真正现代化的软件产业大国的历史任务中起到推波助澜的作用,并请读者在阅读这些译著时,对这套丛书的选题、译文和编排等方面都提出批评和建议。

周伯生

于北京

2002 年 8 月 18 日

前 言

本书介绍了在并发与实时系统的软件设计过程中所使用的概念和方法,概括了并发与实时系统的特征,描述了这些系统设计中的重要概念,讨论了软件设计在软件开发过程中的作用,并对在并发与实时系统所使用的许多软件设计方法进行了研究和比较。然后以更详尽的细节介绍了两种用于设计这类系统的相关软件的设计方法,并使用一些案例研究进行了说明。

在过去的几年中,随着微处理器价格的大幅度降低和性能上的显著提高,对于许多问题来说,并发、实时以及基于微型计算机的分布式实时系统已经成为一种经济高效的解决方案。目前,越来越多的商品、工业产品、消费品、医疗产品和军用品都是基于微型计算机的,这些产品或使用软件来控制,或在其中加入了关键的软件组件。这种系统的范围很广,从微波炉到录像机,从电话到电视,从汽车到飞机,从探索大洋深处的潜艇到遨游太空深处的宇宙飞船,从自动售货机到自动柜员机,从病人监护系统到工厂监测系统,从机器人控制器到电梯控制器,从城市交通控制到空中交通控制、从“智能的”交通运输高速公路到“智能的”信息高速公路——这类系统囊括的范围在不断扩大。这些系统都是并发的、也是实时的。许多系统同时还是分布式的系统。

介绍大规模并发和实时软件系统的书籍很少。作者曾经开发过工业中使用的并发与实时系统,开发并应用了软件设计方法、教授过通用的软件工程以及详细的软件设计课程(讲解理论及其在实际中的应用),本书凝聚了作者在这些方面的经验。本书面向软件专业人员,以及想要设计、实施、评估或了解如何为这些并发、实时和分布式应用设计软件的学生。

本书分为4个部分。第Ⅰ部分介绍了并发与实时系统设计的整体概念。第2章研究针对并发与实时系统的软件生命周期问题。第3章着重介绍对并发与实时系统设计来说特别重要的设计概念,如并发任务、信息隐藏、面向对象的概念以及有限状态机。它还介绍了并发处理所使用的环境,以及对并发处理的操作系统支持。

在第Ⅱ部分中,第4章首先对软件设计和软件设计方法进行了概述。然后

研究了一些用于并发与实时系统的几种软件设计方法。在这里介绍的设计方法有实时结构化分析与设计(第5章),实时系统设计方法—DARTS(第6章),Jackson系统开发(第7章),海军研究实验室软件成本降低方法(第8章),以及面向对象的设计(第9章)。在这几章中,首先都会介绍各种方法所依据的概念,以表明这种方法所要实现的目标是什么。在对方法所使用概念进行了简要介绍之后,就会介绍使用这种方法所需的步骤,以使读者可以完全了解这种方法。书中在介绍每种方法时还列举了使用该方法的一系列产品。因为通过研究实例来学习方法是最有效的方式,所以在介绍每种设计方法时,都会举出一个应用每种设计方法的例子来解决同一个实时问题,即汽车行驶监控系统。每个例子后还有对这种方法的评价,以及这种方法扩展形式和/或变化形式的总结。

在第10章中,从如何应用重要设计概念(在第3章中介绍过)的角度上对各种方法作了比较。第11章介绍了并发与实时系统软件设计的性能分析,并引入了实时调度的思想,这是在实时设计中需要考虑的一个重要问题。

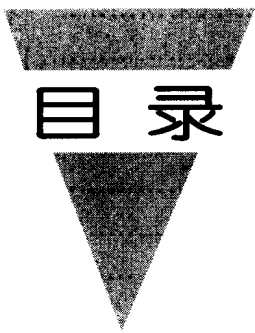
第Ⅲ部分给出了对两种相关软件设计方法的详细介绍,这两种方法都可以用于设计并发与实时系统,它们分别是ADARTS(基于Ada的实时系统设计方法)和CODARTS(实时系统并发设计方法),这些方法建立在第Ⅱ部分介绍的方法的基础之上。ADARTS方法经证明非常有效,多家航空公司都在积极使用这种方法。CODARTS以ADARTS为基础,其中包含了作者最新的软件分析和设计思想。

第Ⅲ部分的结构如下:第12章给出了对ADARTS和CODARTS的概述,并介绍了这些方法所使用的表示法。第13章介绍了在CODARTS中使用的基于并发对象的实时分析(COBRA)方法,这是在开发行为模型过程中用于替代实时结构化分析(ADARTS所使用的方法)的另一种方法。第14~18章给出了ADARTS方法和CODARTS方法中随后步骤的详细介绍。第19章介绍了如何对ADARTS方法或CODARTS方法的设计进行性能分析,以及如何应用实时调度理论和事件序列分析。第20章介绍了处理分布式并发应用程序和分布式实时应用程序设计的CODARTS方面。

第Ⅳ部分给出了一些ADARTS和CODARTS案例研究。其中包括汽车行驶监控系统(第22章)、机器人控制器(第23章)、电梯控制系统(第24章)和分布式工厂自动化系统(第25章)。

熟悉软件生命周期问题和软件设计概念的读者可以略过第2章和第3章。熟悉软件设计方法的读者可以略过第Ⅱ部分中的研究部分,直接开始阅读第

10 章中的比较部分。对 ADARTS 和 CODARTS 特别感兴趣的读者可以直接阅读第Ⅲ部分和第Ⅴ部分。对实时调度及其应用部分感兴趣的读者应该阅读第 11 章、第 19 章以及从第 22~25 章中的案例研究部分。



目录

第 I 部分 概 念

第 1 章 总述	3
1.1 广泛的应用程序	3
1.2 并发系统	3
1.3 实时系统	5
1.4 分布式应用程序	6
第 2 章 并发与实时系统的软件生命周期问题	8
2.1 概述	8
2.2 瀑布生命周期模型	8
2.3 瀑布模型的限制	11
2.4 抛弃式原型法	12
2.5 使用增量式开发的演化原型法	12
2.6 将抛弃式原型法与增量式开发结合起来	13
2.7 螺旋模型	13

2.8 设计确认和验证	14
第3章 软件设计概念	15
3.1 并发处理	15
3.2 并发处理的环境	18
3.3 并发处理的运行时支持	20
3.4 信息隐藏	23
3.5 面向对象的概念	25
3.6 有限状态机	28

第Ⅱ部分 软件设计方法的研究

第4章 软件设计方法概述	33
4.1 概述	33
4.2 软件设计术语	33
4.3 软件设计方法的演化	34
4.4 需求与设计之间的界限	36
4.5 选择软件设计方法的标准	37
4.6 选择方法的基本原理	37
4.7 并发与实时系统的软件设计策略	38
4.8 行驶监控系统的问题描述	39
第5章 实时系统的结构化分析与设计	42
5.1 概述	42
5.2 基本概念	43
5.3 表示法	43
5.4 使用方法的步骤	44
5.5 设计过程的成果	47
5.6 行驶监控系统案例研究	47
5.7 方法评估	63
5.8 扩展与变化	64
第6章 实时系统设计方法	65
6.1 概述	65
6.2 基本概念	65
6.3 表示法	66
6.4 使用方法的步骤	67

6.5	设计过程的成果	68
6.6	行驶监控系统案例研究	68
6.7	方法评估	73
6.8	扩展与变化	73
第7章	Jackson 系统开发	75
7.1	概述	75
7.2	基本概念	75
7.3	表示法	76
7.4	使用方法的步骤	77
7.5	设计过程的成果	78
7.6	行驶监控系统案例研究	78
7.7	方法评估	85
7.8	扩展与变化	86
第8章	海军研究实验室软件成本降低方法	87
8.1	概述	87
8.2	基本概念	87
8.3	表示法	88
8.4	使用方法的步骤	88
8.5	设计过程的成果	90
8.6	行驶监控系统案例研究	90
8.7	方法评估	98
第9章	面向对象设计	100
9.1	概述	100
9.2	基本概念	100
9.3	表示法	101
9.4	使用方法的步骤	101
9.5	设计过程的成果	103
9.6	行驶监控系统案例研究	103
9.7	方法评估	109
9.8	扩展与变化	110
第10章	并发和实时软件设计方法的比较	111
10.1	概述	111
10.2	并发任务比较	111

10.3	信息隐藏和对象比较	113
10.4	有限状态机比较	114
10.5	时间约束	115
第 11 章	并发与实时系统软件设计性能分析	117
11.1	概述	117
11.2	性能模型	117
11.3	Petri 网	120
11.4	实时调度理论	121
11.5	使用事件序列分析进行性能分析	133
11.6	使用实时调度理论和事件序列分析进行性能分析	133

第 III 部分 ADARTS 和 CODARTS

第 12 章	ADARTS 和 CODARTS 的概览	137
12.1	概述	137
12.2	ADARTS 和 CODARTS 的起源	137
12.3	ADARTS 和 CODARTS 的特性	138
12.4	ADARTS 和 CODARTS 的概念基础	139
12.5	概念的集成	141
12.6	使用 ADARTS 的步骤	142
12.7	使用 CODARTS 的步骤	142
12.8	ADARTS 和 CODARTS 表示法	143
第 13 章	并发与实时系统的分析与建模	149
13.1	概述	149
13.2	开发环境模型	150
13.3	系统分解为子系统	152
13.4	行为模型的表示法	159
13.5	问题域中的建模对象	164
13.6	问题域中的建模功能	168
13.7	行为分析	172
13.8	状态依赖行为分析的例子	174
13.9	非状态依赖行为分析	180
13.10	非状态依赖行为分析的例子	180
13.11	与其他方法的比较	182

第 14 章 任务结构化	184
14.1 概述	184
14.2 并发任务结构问题	185
14.3 任务结构分类	185
14.4 I/O 任务结构标准	186
14.5 内部任务结构标准	190
14.6 任务内聚标准	194
14.7 任务优先级标准	202
14.8 开发任务构架	203
14.9 任务通信与同步	205
14.10 使用任务反向进行设计重构	214
14.11 任务行为规范	217
第 15 章 信息隐藏模块结构化	221
15.1 概述	221
15.2 信息隐藏模块结构分类	221
15.3 设备接口模块	222
15.4 行为隐藏模块	224
15.5 软件决策模块	228
15.6 信息隐藏模块聚合层次	228
15.7 设计中的继承	230
15.8 信息隐藏模块的非正式规范	231
第 16 章 任务和信息隐藏模块视图的集成	234
16.1 概述	234
16.2 异步设备 I/O	234
16.3 轮询 I/O	236
16.4 内部任务和模块	237
16.5 对模块的并发访问	239
16.6 对模块同步访问的例子	239
16.7 资源监视任务和设备接口模块的例子	243
16.8 软件构架图的任务行为规范	244
第 17 章 基于 Ada 的构架设计	246
17.1 将设计映射到目标系统环境	246
17.2 Ada 支持任务	246