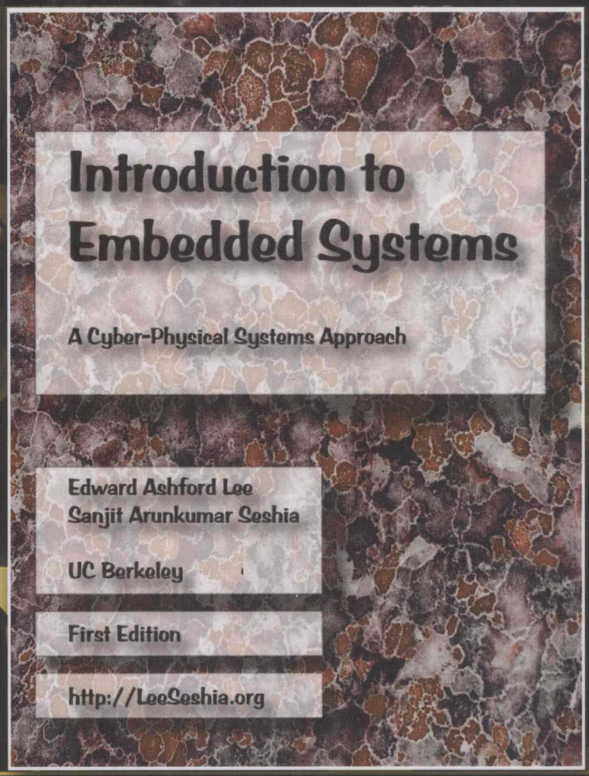
HZ BOOKS
华章教育

计 算 机 科 学 丛 书

嵌入式系统导论
CPS方法

(美) Edward Ashford Lee
加州大学伯克利分校
Sanjit Arunkumar Seshia
加州大学伯克利分校
著 李实英 贺蓉 李仁发 译

Introduction to Embedded Systems
A Cyber-Physical Systems Approach



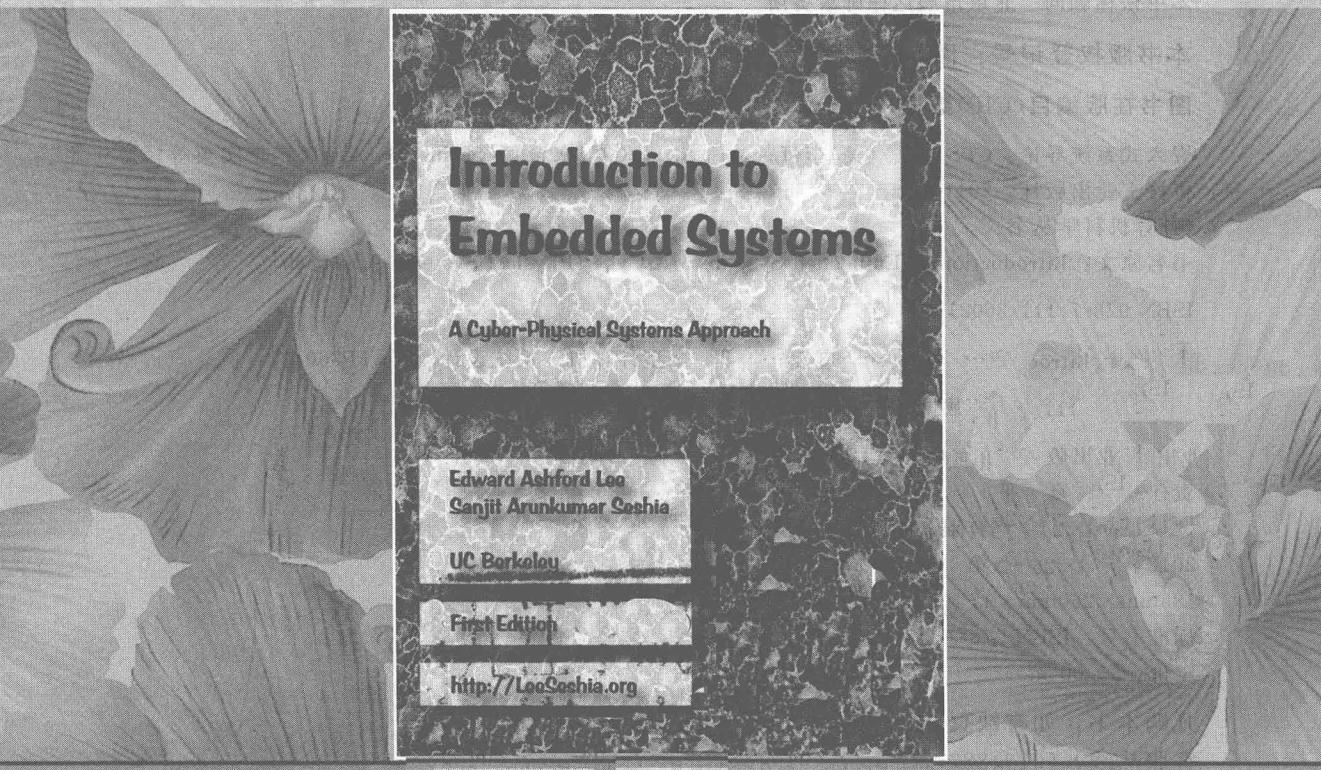
机械工业出版社
China Machine Press

计 算 机 科 学 丛 书

嵌入式系统导论
CPS方法

(美) Edward Ashford Lee 加州大学伯克利分校 Sanjit Arunkumar Seshia 加州大学伯克利分校 著 李实英 贺蓉 李仁发 译

Introduction to Embedded Systems
A Cyber-Physical Systems Approach



机械工业出版社
China Machine Press

本书是业界第一本关于CPS的专著，重点论述系统模型与系统实现的关系，以及软件和硬件与物理环境的相互作用。

从CPS的视角，围绕系统的建模、设计和分析这三个方面，本书分成四大部分。第一部分(第2～6章)分别讲述动态建模、离散建模和混合建模，以及状态机的并发组合与并发计算模型。第二部分(第7～11章)强调嵌入式系统中处理器、存储器架构、输入和输出、多任务处理和实时调度的算法与设计，以及这些设计在CPS中的主要作用。第三部分(第12～15章)重点介绍一些系统特性的精确规格、规格之间的比较方法、规格与产品设计的分析方法以及嵌入式软件特性的定量分析方法。第四部分包括两个附录，提供了一些数学和计算机科学的背景知识，有助于读者加深对文中所介绍知识的理解。本书通过大量实例深入浅出地介绍了设计和实现CPS的整体过程及各阶段的细节。

本书适合作为高等院校相关专业"嵌入式系统"课程的教材或教学参考书。

本书版权登记号：图字：01-2011-3604

图书在版编目(CIP)数据

嵌入式系统导论：CPS方法/(美)李(Lee, E. A.)，(美)塞希阿(Seshia, S. A.)著；李实英等译．—北京：机械工业出版社，2011.12

(计算机科学丛书)

书名原文：Introduction to Embedded Systems—A Cyber-Physical Systems Approach

ISBN 978-7-111-36021-6

Ⅰ.嵌…　Ⅱ.①李…　②塞…　③李…　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(2011)第200947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：夏　平

北京诚信伟业印刷有限公司印刷

2012年1月第1版第1次印刷

185mm×260mm·17印张

标准书号：ISBN 978-7-111-36021-6

定价：55.00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

购书热线：(010)68326294；88379649；68995259

投稿热线：(010)88379604

读者信箱：hzjsj@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，Brain W. Kernighan，Dennis Ritchie，Jim Gray，Afred 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，我们的联系方法如下：

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

华章科技图书出版中心

译者序

Introduction to Embedded Systems—A Cyber-Physical Systems Approach

不同于大多数嵌入式系统的书籍着重于计算机技术在嵌入式系统中的应用，本书的重点是论述系统模型与系统实现的关系，以及软件和硬件与物理环境的相互作用。本书是业界第一本关于CPS(Cyber-Physical System，信息物理系统)的专著。

原书作者美国加州大学伯克利分校Edward Ashford Lee教授是世界上嵌入式系统领域的著名学者，也是CPS研究的倡导者和引领者之一。Lee教授领导的团队还开发了一个叫做Ptolemy Ⅱ的系统，这是一个非常优秀的开源嵌入式系统研究与开发平台，可惜国内了解的人不多。

CPS将计算、网络和物理过程集成在一起，CPS的建模、设计和分析成为本书的重点。

从CPS的视角，围绕系统的建模、设计和分析这三个方面，本书分成四大部分。第一部分包括第2～6章，分别讲述动态建模、离散建模和混合建模，以及状态机的并发组合与并发计算模型。第二部分包括第7～11章，强调嵌入式系统中处理器、存储器架构、输入和输出、多任务处理和实时调度的算法与设计，以及这些设计在CPS中的主要作用。第三部分包括第12～15章，重点介绍一些系统特性的精确规格、规格之间的比较方法、规格与产品设计的分析方法以及嵌入式软件特性的定量分析方法。第四部分包括两个附录，提供了一些数学和计算机科学的背景知识，有助于加深对文中所介绍知识的理解(可以登录网站http：//LeeSeshia.org获取本书的英文版和更多信息)。

李仁发教授负责本书的整体翻译工作。李实英博士主译了前言、第1章以及第一部分和第二部分，贺蓉博士主译了第三部分和第四部分。此外，参与本书翻译工作的还有谢勇、黄鑫、杜家宜、王震、高楠、宋倩、刘琳、周权、吴武飞、谷连军、何翔、许文龙、吴文康、黄晶、李志灿和胡环。

把一种语言表达转换成另一种是一件困难的事情。看似很直白的一个词，虽然理解其词义，但要换一种语言表达时往往煞费苦心，况且CPS的研究在国内尚处于起步阶段，许多名词还很难给出确切的中文译名。译者力求忠实地表达书中所介绍的技术，保持原作者的行文风格。此外，本书的翻译以Leesheshia _ DigitalV1 _ 03为基础，并根据作者更新的版本(Leesheshia _ DigitalV1 _ 05)进行了校对，修正了原稿中的一些改动。限于时间(急于想把此书介绍给国内读者)以及译者水平和经验的不足，译文中难免存在许多不当之处，恳请读者提出宝贵的意见。

本书在翻译过程中得到了Edward Ashford Lee教授本人的直接帮助，同时还得到湖南大学嵌入式系统与网络实验室同仁及机械工业出版社许多人士的帮助。对此，译者深表感谢。

关于此书

计算机和软件最显著的用途是处理人们所使用的信息。我们用计算机和软件写教材(如本书)、在网上搜索信息、通过电子邮件进行通信以及跟踪财务数据。然而，绝大多数应用中的计算机并非如此显而易见。这些计算机运行于汽车上的引擎、刹车、安全带、安全气囊和音响系统中。它们将声音进行数字编码并转换成无线电信号，然后从手机发送到基站。它们控制微波炉、冰箱和洗碗机。它们运行各种打印机，从台式喷墨打印机到大型工业用的高容量打印机。它们指挥车间里的机器人、发电厂的电力生产、化工厂的各种工序，以及城市的交通灯。它们在生物样本中搜寻细菌，构建人体内部图像，以及测量生命体征。它们处理来自太空的无线电信号，寻找超新星和外星智慧生物。它们给人类生活带来各种玩具，而且让这些玩具能对人的触摸和声音有所反应。它们控制飞机和火车。这些不显眼的计算机称为**嵌入式系统**(embedded system)，而在嵌入式系统上运行的软件称为**嵌入式软件**(embedded software)。

尽管嵌入式系统得到如此广泛的应用，但是从计算机科学相对短暂的发展历史来看，它主要致力于信息处理。直到最近，嵌入式系统才受到研究人员的更多关注。研究界认识到，设计和分析嵌入式系统所需的工程技术是与通用计算机系统不同的。尽管嵌入式系统从20世纪70年代就开始应用，但是长期以来，这些系统被简单地视做小型计算机。最主要的工程问题被理解为如何运用有限资源(有限的处理能力、有限的电源、小型存储器等)的问题。这样一来，工程挑战成为优化设计。由于所有设计都得益于优化，所以该学科与计算机科学的其他方面相比并没有什么独特的地方。它只在应用同样的优化方法方面必须更加积极投入。

目前，研究界已经认识到，嵌入式系统中最主要的挑战来自于它们与物理过程的相互作用，而不是它们的有限资源。“Cyber-Physical System”(CPS)这个术语由美国国家科学基金会的Helen Gill提出，用于描述计算与物理过程的整合。在CPS中，嵌入式的计算机和网络通常采用反馈回路来监视和控制物理过程，在反馈回路中物理过程和计算相互影响。因而，这种系统设计需要理解计算机、软件、网络和物理过程的动态融合。正是对于动态融合(joint dynamics)的研究将这个学科与其他学科分离开来。

在研究CPS的过程中，会遇到一些在通用计算中很少出现的关键问题。例如，在通用软件中，执行一个任务的时间与性能相关，但不与正确性相关。执行一个任务耗费更长时间并不是不正确的，只是不太方便，因而不那么有价值。但是在CPS中，执行一个任务所需的时间可能对系统的正确功能实现至关重要。与信息世界相反，在物理世界中时间的流逝是必然的。

而且，在CPS中许多事情会同时发生。物理过程是许多同时发生事情的组合，这与深深植根于顺序步骤中的软件进程不同。Abelson and Sussman(1996)将计算机科学描述为“程序化认识论”(procedural epistemology)，即知识贯穿于整个程序。相对而言，在物理世界中，过程很少是程序化的。物理过程是许多并行过程的组合。通过调和影响这些过程的行为来对这些过程的动态状况进行测量和控制是嵌入式系统的主要任务。因此，并发性是CPS固有的。在设计和分析嵌入式软件时，许多技术挑战源于必须建立原本顺序的语义与本质上并发的物理世界之间的桥梁。

写作目的

当今，使计算机与物理过程协作需要复杂的技术和底层的设计。因此嵌入式软件设计人员不得不与中断控制器、存储器架构、汇编级编程(以开发专用指令或进行精确的时间控制)、设备驱动程序设

计、网络接口和调度策略打交道，而不是只关注所要实现的行为。这些技术的高质量和复杂性需要我们重点掌握一些入门知识。这些知识着重于如何对软件、网络和物理过程的动态融合进行建模和设计，介绍实现这些动态融合的最新(而不是过去的)方法。本书正是力求成为这样一本教材。

大多数关于嵌入式系统的书籍侧重于使计算机与物理系统交互所需的技术(Barr and Massa, 2006; Berger, 2002; Burns and Wellings, 2001; Kamal, 2008; Noergaard, 2005; Parab et al., 2007; Simon, 2006; Valvano, 2007; Wolf, 2000)。其他书籍则着重于将计算机技术(如编程语言、操作系统、网络等)应用于嵌入式系统中的技术问题(Buttazzo, 2005a; Edwards, 2000; Pottie and Kaiser, 2005)。虽然这些实现技术可以满足设计人员使嵌入式系统正常工作的需求，但并不是构成该学科的智能核心。其智能核心是将计算与物理动态结合在一起的模型和抽象化。

有些教科书在该方向上付出了很多努力。Jantsch (2003)着重于并发计算模型，Marwedel(2003)侧重于软件和硬件的行为模式，Sriram and Bhattacharyya (2009)关注信号处理行为的数据流模型及其在可编程 DSP 上的映射。以上都是非常良好的开端。并发模型(如数据流)以及软件的抽象模型(如状态图)可以作为比命令式编程语言(如 C)、中断和线程以及设计人员必须解决的架构问题(比如缓存)更好的开端。然而，这些教材并不适合入门课程。它们要么太专业要么太高级，或者两者兼而有之。本书力求成为一本入门式的教材，侧重于介绍系统模型及其与系统实现之间的关系。

本书的主题是关于系统模型以及它们与系统实现的关系。这些模型主要是动态的，即系统状态随时间演化。我们不介绍关于组成系统静态信息的结构模型，尽管这些模型对于嵌入式系统设计也非常重要。

应用模型有一个很大的优势。模型具有形式上的特性。可以利用模型定义事件。例如，可以断言一个模型是确定性的(deterministic)，也就是说，给定相同的输入，它总是产生相同的输出。系统的任何物理实现都不能进行这样绝对的断言。如果模型是对一个物理系统的良好抽象(在此，“良好抽象”是指它只忽略了一些无关紧要的细节)，那么模型的决定性断言可以增加系统物理实现的可靠性。这种可靠性是极具价值的，尤其对于嵌入式系统而言，其出现的故障可能危及人的生命。通过学习系统模型可以了解这些系统在物理世界中是如何运行的。

我们的重点是软件和硬件与所处物理环境的相互作用。这就要求对软件和网络的时序动态进行明确建模并精确描述应用中固有的并发特性。事实上，实现技术还不能满足这一要求，这当然不能成为我们传授错误的工程方法的理由，而应该按照设计和建模应该有的形式进行教学，并以如何通过最新技术(部分)实现这些目标的关键表述来丰富其内容。因此，当今的嵌入式系统技术，不应该如上述许多教材那样，被描述成一大堆事实和技巧，而应该脚踏实地逐步实现合理的设计实践，重点应该放在什么是合理的设计实践，以及如今的技术如何阻碍和实现它。

Stankovic et al. (2005)支持这一观点，指出“现有的 RTES(实时嵌入式系统)设计技术并不能有效地支持可靠而鲁棒的嵌入式系统开发”。他们提出需要“提高编程抽象化的水平”。我们认为，只提高抽象化水平是不够的，应该从根本上改变所使用的抽象。例如，如果没有建立于其上的底层抽象，软件的时间特性则不能有效地利用更高层次的抽象。

我们需要采用鲁棒性好、可预见且具有可重复时序动态的设计(Lee, 2009a)，这必须通过建立能适当反映信息物理系统实际情况的抽象来实现。其结果将使 CPS 的设计更加复杂，包括更多的自适应控制逻辑、时间演化，以及安全性和可靠性的提高，而避免当今设计的不稳定，很小的改变都成为大问题。

除了处理时序动态，CPS 设计总是面临并发问题的挑战。由于软件是植根于顺序抽象，并发机制(如中断和多任务处理，利用信号和互斥锁)变得非常重要。因此，我们在本书中特别强调对线程、消息传递、死锁避免、竞争条件和数据决策等关键内容的理解。

本书未尽事宜

本书的版本并不完整。实际上，要完全涵盖 CPS 意义上的嵌入式系统是不可能的。其中涵盖了

Berkeley(伯克利)本科嵌入式系统课程(http: //LeeSeshia. org)的具体内容，并希望在将来的版本中包括：传感器和制动器、网络、容错、安全性、仿真技术、控制系统以及硬件/软件协同设计。

如何使用本书

本书分为三个主要部分，分别关注建模、设计和分析，如图 1 所示。这三个部分彼此相对独立，在很大程度上是为了能同时学习。利用虚线框出的七个分区可以实现系统性的学习。每个分区包括两章，假设每个分区需要花两周时间，并且绪论和结束部分各花一周时间，则本书可以在 15 周内学完。

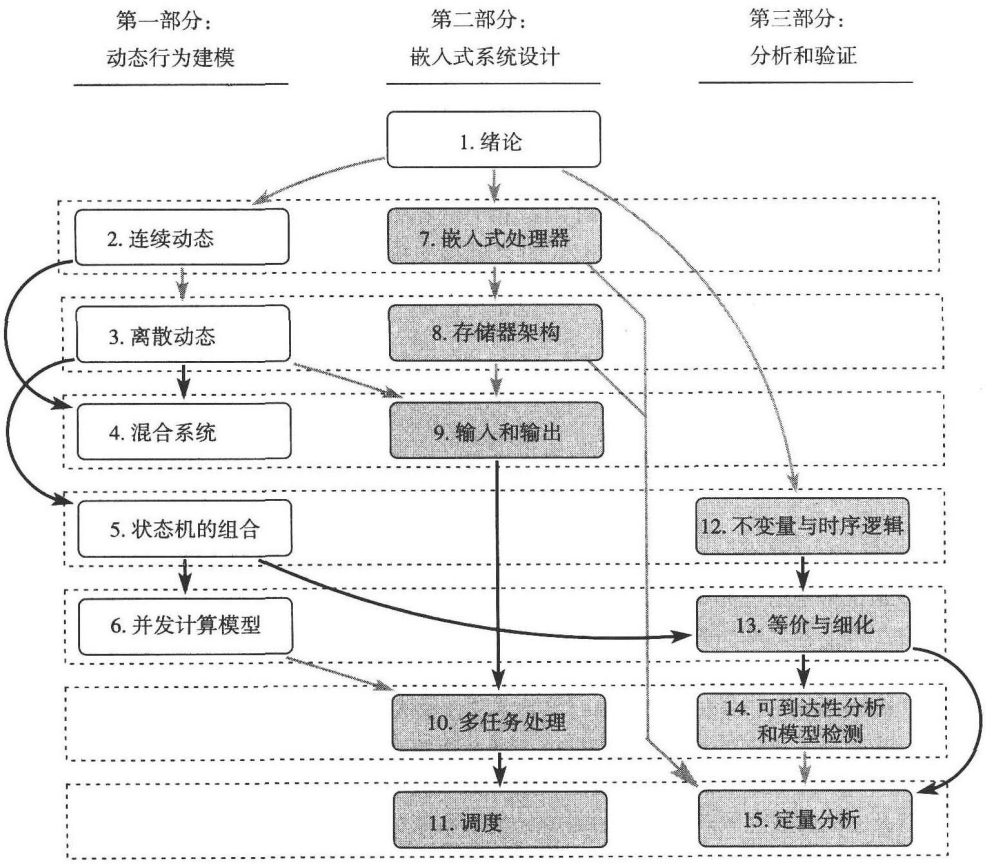


图 1　本书各章节之间依赖性强弱分布图。黑色箭头表示章节之间有强依赖关系，灰色箭头表示章节之间有弱依赖关系。如果章节 i 和章节 j 之间有弱依赖关系，则可以只学习章节 j，而不学习章节 i，或者略去一些实例或具体的分析方法

附录提供的背景资料在其他教材中也很详尽，但是收入本书对学习可以提供很大帮助。附录 A 介绍的是集合和函数的概念，这些概念可以提高嵌入式系统学习中常见的精确度。附录 B 介绍的是复杂度和可计算性理论的基本成果，这些可以加深理解系统建模和分析中的问题。值得注意的是，附录 B 依赖于第 3 章讲述的状态机理论，因此应该在读完第 3 章之后进行学习。

由于技术的最新发展从根本上改变了出版业的技术，本书英文原版以非传统方式出版。至少当前的版本可以以 PDF 文件方式在线阅读，可从网站 http: //LeeSeshia. org 上免费获取。布局适合于中型屏幕，尤其是笔记本电脑、iPad 以及其他平板电脑。此外，大量使用超链接和色彩以加强在线阅读体验。

我们力图使英文书适用于电子书形式，理论上可以在不同大小的屏幕上阅读，从而充分利用现有的显示屏。然而，类似于 HTML 文档，电子书的格式采用回流技术，使页面布局即时重新绘制。其结

果非常依赖于屏幕的大小，在很多屏幕上显得有些怪异，不是最优的显示。因此，我们选择性地对布局进行了控制，不建议在 iPhone 上阅读。

读者对象

本书适用于高年级本科生或低年级研究生，以及想要了解嵌入式系统工程原理的实践工程师和计算机专家。读者需要懂得计算机结构(如知道什么是 ALU)、计算机编程(本书采用 C 语言)、基本的离散数学和算法知识，最好对信号和系统也有所了解(比如何为对连续信号进行采样)。

致谢

感谢以下人员在过去三年中对本书的贡献和提出的宝贵建议：Murat Arcak, Janette Cardoso, Gage Eads, Stephen Edwards, Suhaib Fahmy, Shanna-Shaye Forbes, Jeff C. Jensen, Jonathan Kotker, Wenchao Li, Isaac Liu, Slobodan Matic, Le Ngoc Minh, Steve Neuendorffer, Minxue Pan, Hiren Patel, Jan Reineke, Rhonda Righter, Chris Shaver, Stavros Tripakis, Pravin Varaiya, Maarten Wiggers, Qi Zhu, 以及 UC Berkeley 的 EECS 149 班的学生，尤其是 Ned Bass 和 Dan Lynch。特别感谢 Elaine Cheong 认真审阅了大多数章节并提出了有益的建议。特别感谢家人的忍耐和支持，尤其是 Helen、Katalina、Rhonda (来自 Edward)，以及 Appa、Amma、Ashwin 和 Bharathi (来自 Sanjit)。

错误反馈

如果发现本书中的错误或印刷错误，或者如果有改进或其他建议，请发送电子邮件到 authors@leeseshia.org。

请注明该书的版本号和相关的页码，无论是电子版或纸质版。非常感谢！

扩展阅读

最近几年出现了很多嵌入式系统方面的教材。这些教材以各种不同的方式介绍这一主题，往往反映一些与嵌入式系统紧密结合的成熟领域的发展前景，如 VLSI 设计、控制系统、信号处理、机器人、实时系统或软件工程。其中有些教材可以作为本书的有益补充，强烈推荐给那些想要深入理解嵌入式系统的读者。

具体而言，Patterson and Hennessy (1996)尽管不是以嵌入式处理器作为重点，但它是一本计算机架构的经典教材，是对嵌入式处理器架构感兴趣的读者必读的。Sriram and Bhattacharyya (2009)侧重于信号处理应用，如无线通信和数字媒体，并全面介绍了数据流编程方法。Wolf (2000)对硬件设计技术和微处理器架构以及它们对嵌入式软件设计的影响给出了精彩的概述。Mishra and Dutt (2005) 介绍了基于结构描述语言(ADL)的嵌入式架构。Oshana (2006) 专门介绍了 Texas Instruments(德州仪器)公司(TI)的 DSP 处理器，综述了架构方法和汇编级编程。

在软件上，Buttazzo(2005a)对实时软件的调度技术进行了全面综述。Liu (2000)对处理软件中的突发实时事件提供了一个目前为止最好的方法。Edwards (2000)介绍了一种用于嵌入式系统设计的域专用高级编程语言。Pottie and Kaiser (2005) 综合介绍了网络技术，特别是嵌入式系统中的无线技术。Koopman (2010)侧重于嵌入式软件的设计过程，包括需求管理、项目管理、测试计划和安全计划。

没有任何一本教材可以全面涵盖所有技术，提供给嵌入式系统工程师。很多教材提供了关于当前设计技术的有益信息(Barr and Massa, 2006; Berger, 2002; Burns and Wellings, 2001; Gajski et al., 2009; Kamal, 2008; Noergaard, 2005; Parab et al., 2007; Simon, 2006)。

致教师

在 Berkeley，本书用于高年级本科生课程“嵌入式系统导论”。通过本书的网站主页 http://LeeSeshia.org 可以获得大量讲座和实验的资料。另外，还可以获得解题手册。请联系 authors@leeseshia.org。

$x\|_{t\leqslant\tau}$	时间约束
$\neg$	否定
$\wedge$	逻辑与
$\vee$	逻辑或
$L(M)$	语言
$:=$	赋值
V_{CC}	电源电压
$\Longrightarrow$	蕴涵
$\mathbf{G}\phi$	全局
$\mathbf{F}\phi$	最终
$\mathbf{U}\phi$	直到
$\mathbf{X}\phi$	下一个状态
$L_a(M)$	FSM可接受语言
λ	空序列
$\mathbb{B}=\{0,1\}$	二进制数
$\mathbb{N}=\{0,1,2,\cdots\}$	自然数
$\mathbb{Z}=\{\cdots,-1,0,1,2,\cdots\}$	整数
$\mathbb{R}$	实数
$\mathbb{R}_+$	非负实数
$A\subseteq B$	子集
2^A	幂集
$\varnothing$	空集
$A\backslash B$	差集
$A\times B$	笛卡儿积
$(a,b)\in A\times B$	元组
A^0	单元素集
$f:A\rightarrow B$	函数
$f:A\rightharpoonup B$	部分函数
$g\circ f$	复合函数
$f^n:A\rightarrow A$	函数到幂
$f^0(a)$	恒等函数
$\hat{f}:2^A\rightarrow 2^B$	象函数
$(A\rightharpoonup B)$	A到B的所有函数的集合
B^A	A到B的所有函数的集合
π_l	投影
$\hat{\pi}_l$	提升版投影
$f\|_C$	限制
A^*	有限序列
$A^{\mathbb{N}}$	无限序列
$\omega=\{\varnothing,\{\varnothing\},\{\varnothing,\{\varnothing\}\},\cdots\}$	冯·诺依曼数
A^ω	无限序列
A^{**}	有限和无限序列
$\square$	空格

目 录

Introduction to Embedded Systems—A Cyber-Physical Systems Approach

绪　论

CPS(Cyber-Physical System)是计算与物理过程的整合。嵌入式计算机和网络通过反馈回路监视和控制物理过程，在反馈回路中物理过程与计算相互影响。作为一项智力挑战，CPS是物理层和信息层的交集，而不是并集。物理组件和计算组件不能分开理解。相反，必须理解它们之间的相互作用。

本章通过一些CPS应用阐述这种系统的工程原理和设计流程。

1.1　应用

CPS应用不可争辩地具有使20世纪IT技术革新黯然失色的潜力。考虑以下实例。

例1.1：心脏手术通常需要停止心脏跳动来实施手术，然后再使心脏重新跳动。这种手术是极其有风险的，而且会带来很多有害的副作用。很多研究团队一直在研究另一种方法——可以在心脏跳动时做手术，而不需要使之停止跳动。有两个重要想法可以使之成为可能。首先，手术工具可以由机器人控制，使它们能跟随心脏的跳动而移动(Kremen，2008)。因此，外科医生可以在心脏持续跳动的同时使用工具给心脏上的一点施加定量的压力。其次，立体视频系统可以向外科医生展现一个静止心脏的假象视频(Rice，2008)。对于外科医生而言，心脏看起来停止了跳动，而事实上心脏是在持续跳动的。要实现这样的外科手术系统，需要大量的心脏模型、工具、计算硬件以及软件。它需要精心设计的软件以保证精确的计时和处理故障的安全备用行为，而且还需要对模型和设计进行详细分析来提供高可靠性。

例1.2：考虑城市中的交通灯和车辆共同配合来保证有效的交通流。特别地，假设只有在实际的十字路口才需要因为红灯而停车。这样的系统可以通过检测路上车辆等昂贵基础设施来实现。但是一种更好的方法是让车辆之间进行协作。它们记录自己的位置，并进行通信以互相配合共享资源，比如十字路口。当然，系统的可靠性是其可行性的关键。故障是灾难性的。

例1.3：设想一架能够防止坠落的飞机。虽然不能杜绝所有可能导致坠落的因素，但是一个精心设计的飞行控制系统可以防止很多因素。这样的系统是很好的信息物理系统实例。

在传统的飞机中，飞行员通过连接驾驶舱控制器的机械和液压装置以及机翼的可移动水平面和飞机尾部来控制飞机。在电控式(fly-by-wire)飞机中，飞行员发出的指令经由飞行计算机进行调节，并通过网络将电子信号传送到机翼和机尾的制动器。电控式飞机比传统飞机轻很多，因而更加省燃料。而且已证实电控式飞机更加可靠。事实上，所有新型飞机都被设计成电控系统。

在电控式飞机中，由于计算机调节来自飞行员的指令，所以计算机可以修正这些指令。很多现代飞行控制系统都在某些情况下修正飞行员的指令。例如，Airbus公司制造的商业飞机采用一种称为飞行状态防护(flight envelope protection)的技术防止飞机超出安全工作范围。例如，可以防止飞行员引起飞机失速。

飞行状态防护的概念可以得到扩展来帮助防止一些其他坠落因素。例如，Lee(2001)提出一种软壁(soft wall)系统，这种系统一旦实现，就可以对安装该系统的飞机进行跟踪定位，并防止飞机撞到障碍物上，如山脉和建筑物。利用 Lee 方案，当飞机接近障碍物边界时，电控式飞行控制系统会产生一个虚拟推动力迫使飞机远离障碍物。飞行员感觉飞机好像碰撞到一堵软壁而转移方向。设计和部署这种系统在技术和非技术上还有很多挑战。这些问题的讨论可参见 Lee(2003)。

尽管上例中的软壁系统是非常超前的，但是已经有一些不那么超前的自动安全系统得到部署或处于最高级研发阶段。例如，当今很多车辆能够检测无意间的车道变化并提醒驾驶员。考虑一些更具挑战性的自动纠正驾驶员行为的问题。显而易见，这比仅仅提醒驾驶员要困难得多。如何能保证系统在需要时进行响应和接管，并在一定程度上进行实际干预呢？

显然可以设想很多其他应用，比如老年人辅助系统，能让外科医生进行远距离手术的远程外科手术系统，以及需要在稳定电压下工作的家用电器。此外，还很容易设想利用 CPS 改进很多现有的系统，比如机器人制造系统，发电和配电，化工厂的过程控制，分布式计算机游戏，产品的运输，建筑物的供暖、制冷和照明，电梯等客运工具，以及能够监控自身健康状态的桥梁。这些改进对安全、能源消耗和经济具有极其深远的影响。

以上很多实例可以采用如图 1-1 所示的结构进行部署。图中有三个主要部分。首先，**物理设备**是信息物理系统中的“物理”部分。这只是系统中没有利用计算机或数字网络实现的部分。它包括机械零件、生物或化学过程，以及人工操作。第二，有一个或多个计算**平台**，由传感器、驱动器、一台或多台计算机，以及(可能的)一个或多个操作系统组成。第三，有一个**网络结构**，它提供计算机进行通信的机制。平台和网络结构共同构成信息物理系统的“信息”部分。

如图 1-1 所示的是两个网络连接平台，每个都有各自的传感器和/或驱动器。驱动器的行为影响传感器从物理设备接收到的数据。图中，平台 2 通过驱动器 1 控制物理设备。它通过传感器 2 测量物理设备的工作进程。图中标为计算 2 的方框实现一个**控制规则**，根据传感器数据决定发出什么样的命令到驱动器。这种回路称为**反馈控制**回路。平台 1 利用传感器 1 进行其他测量，并通过网络结构将消息送给平台 2。计算 3 实现另外一个控制规则，与计算 2 融合在一起，可能抢占计算 2。

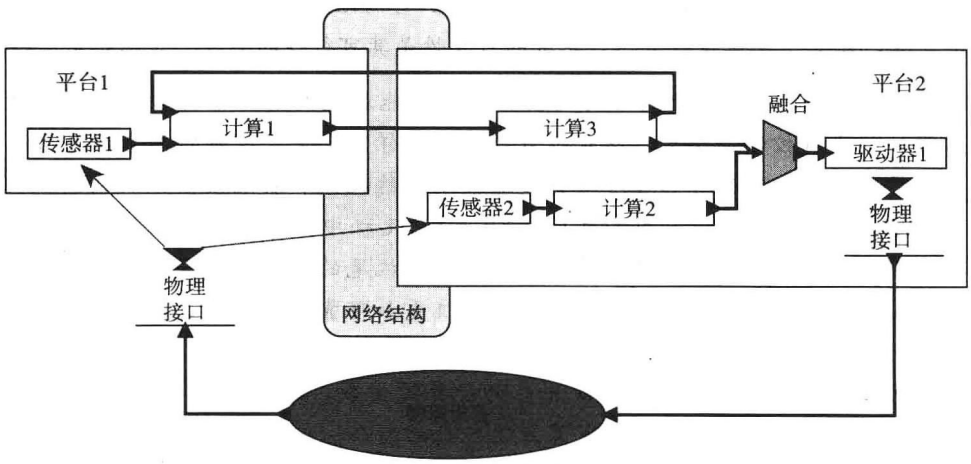


图 1-1　CPS 的结构实例

例 1.4：考虑一个提供按需打印服务的高速打印机。它的结构大概与图 1-1 类似，但是具有更多的平台、传感器和驱动器。驱动器控制电动机推动纸张通过打印机，将墨水印在纸上。控制规则包括纸张拉伸补偿策略，通常取决于纸张类型、温度以及湿度。图 1-1 所示的网络结构也许用于卡纸的情况下快速关机以防止损坏设备。这种关机要求整个系统紧密协调以防止崩溃。类似的情况也可见于高端仪器系统以及能源生产和调度(Eidson et al.，2009)。

关于术语 CPS

CPS 这个术语大约出现于 2006 年，由美国国家科学基金会的 Helen Gill 提出。虽然大家都熟悉“信息空间”(cyberspace)这个术语，也许会将它与 CPS 联系起来，但是 CPS 的根源更加久远。认为“信息空间”和 CPS 源于相同的根源——“控制论”(cybernetics)应该是对的，而不应该认为一个衍生于另外一个。

“控制论”这个术语是由 Norbert Wiener (Wiener，1948)提出的，他是一名对控制系统理论的发展具有深远影响的美国数学家。在二战期间，Wiener 首创了高射炮自动瞄准和射击技术。虽然他采用的机制没有涉及数字计算机，但其中的理论与现今用于各类基于计算机的反馈控制系统的理论相类似。控制论这个术语是 Wiener 从希腊文中的 κυβερνητης (kybernetes)一词衍生来的，意指舵手、管理者、领航员或方向舵。这个比喻非常适合控制系统。

Wiener 将控制论视为控制和通信的结合。他的控制概念深深植根于闭环反馈，其中控制逻辑是由物理过程的测量值驱动，并反过来驱动物理过程。Wiener 虽然没有使用数字计算机，但控制逻辑实际上是一个计算，因此控制论是物理过程、计算和通信的结合。

Wiener 没有预料到数字计算和网络的巨大影响力。CPS 可以粗略地解释为信息空间与物理过程的结合，这一事实有助于强调 CPS 将有巨大的影响力。CPS 充分发挥了惊人的信息技术，甚至远远超过 Wiener 所处时代最疯狂的梦想。

1.2 一个实例

本节主要讲述 CPS 的一个实例，目的是借用这个实例说明本书所述内容的重要性。具体应用如斯坦福大学多代理控制的自主旋翼飞行器实验平台(STARMAC)，由 Claire Tomlin 和同事在斯坦福和伯克利合作开发(Hoffmann et al.，2004)。STARMAC 是一个小型四旋翼飞行器，如图 1-2 所示。它的主要目的是提供一个多飞行器自主控制技术的实验平台。目标是使多个飞行器合作完成一个通常的任务。

使这样的系统正常工作将面临巨大的挑战。第一，飞行器的控制至关重要。主要的驱动装置是四个旋翼，它们产生可变的向下推力。通过平衡来自四个旋翼的推力，飞行器可以起飞、降落、转向，甚至在空中翻转。如何确定需要怎样的推力呢？这就需要复杂的控制算法。

第二，飞行器的重量是另一个重要问题。它越重，就需要携带越多的能量，这无疑又会增加飞行器的重量。飞行器越重就需要越大的推力来飞行，这意味着需要更大更强的马达和旋翼。当飞行器达到一定重量，旋翼会对人类造成威胁，这是设计要跨越的主要门槛。即便