

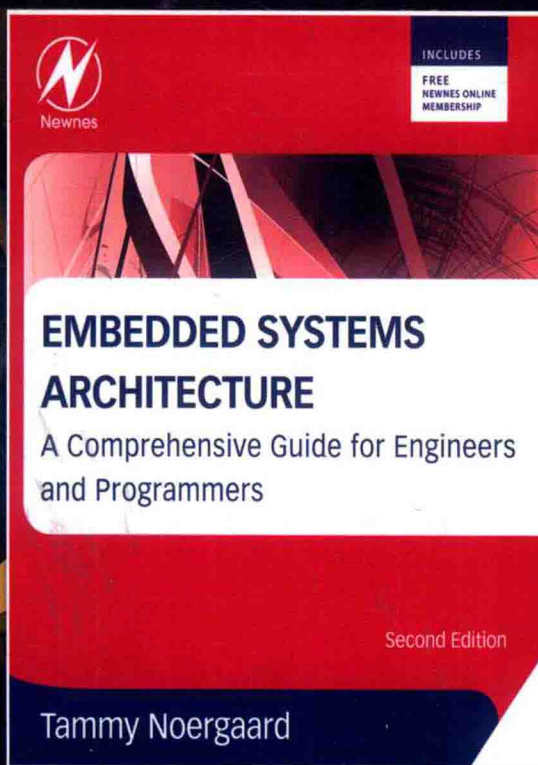
# 嵌入式系统

## 硬件、软件及软硬件协同

[美] 塔米·诺尔加德 (Tammy Noergaard) 著  
马志欣 苏锐丹 付少锋 译

### Embedded Systems Architecture

A Comprehensive Guide for Engineers and Programmers Second Edition



计 算 机 科 学 丛 书

原书第2版

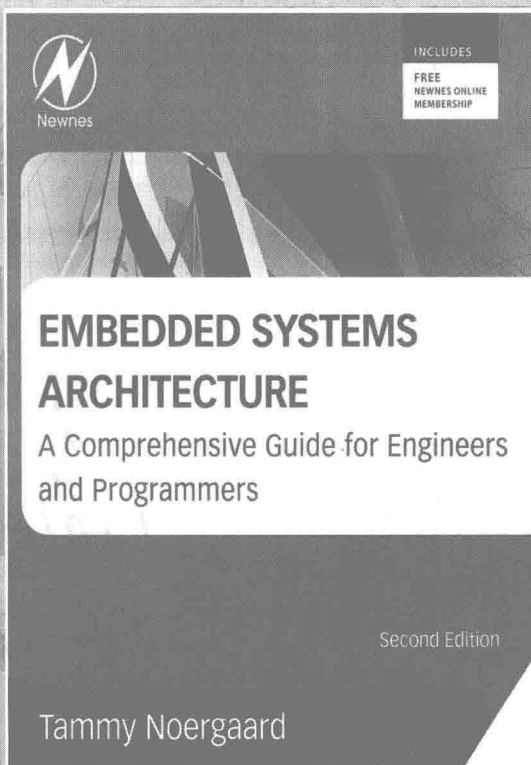
# 嵌入式系统

## 硬件、软件及软硬件协同

[美] 塔米·诺尔加德 (Tammy Noergaard) 著  
马志欣 苏锐丹 付少锋 译

### Embedded Systems Architecture

A Comprehensive Guide for Engineers and Programmers Second Edition



机械工业出版社  
China Machine Press

## 图书在版编目 (CIP) 数据

嵌入式系统: 硬件、软件及软硬件协同 (原书第 2 版) / (美) 塔米·诺尔加德 (Tammy Noergaard) 著; 马志欣, 苏锐丹, 付少锋译. —北京: 机械工业出版社, 2018.2  
(计算机科学丛书)

书名原文: Embedded Systems Architecture: A Comprehensive Guide for Engineers and Programmers, Second Edition

ISBN 978-7-111-58887-0

I. 嵌… II. ①塔… ②马… ③苏… ④付… 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 (2018) 第 003100 号

本书版权登记号: 图字 01-2013-5995

ELSEVIER

Elsevier (Singapore) Pte Ltd.

3 Killiney Road, #08-01 Winsland House I, Singapore 239519

Tel: (65) 6349-0200; Fax: (65) 6733-1817

Embedded Systems Architecture: A Comprehensive Guide for Engineers and Programmers, Second Edition  
Tammy Noergaard  
Copyright © 2013, 2005 by Elsevier Inc. All rights reserved  
ISBN-13: 9780123821966

This translation of Embedded Systems Architecture: A Comprehensive Guide for Engineers and Programmers, Second Edition by Tammy Noergaard was undertaken by China Machine Press and is published by arrangement with Elsevier (Singapore) Pte Ltd.

Embedded Systems Architecture: A Comprehensive Guide for Engineers and Programmers, Second Edition by Tammy Noergaard 由机械工业出版社进行翻译, 并根据机械工业出版社与爱思唯尔(新加坡)私人有限公司的协议约定出版。

《嵌入式系统: 硬件、软件及软硬件协同 (原书第 2 版)》(马志欣 苏锐丹 付少锋 译)

ISBN: 9787111588870

Copyright © 2018 by Elsevier (Singapore) Pte Ltd.

All rights reserved. No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or any information storage and retrieval system, without permission in writing from Elsevier (Singapore) Pte Ltd. Details on how to seek permission, further information about the Elsevier's permissions policies and arrangements with organizations such as the Copyright Clearance Center and the Copyright Licensing Agency, can be found at our website: [www.elsevier.com/permissions](http://www.elsevier.com/permissions).

This book and the individual contributions contained in it are protected under copyright by Elsevier (Singapore) Pte Ltd. and China Machine Press (other than as may be noted herein).

### 注意

本译本由 Elsevier (Singapore) Pte Ltd. 和机械工业出版社完成。相关从业及研究人员必须凭借其自身经验和知识对文中描述的信息数据、方法策略、搭配组合、实验操作进行评估和使用。由于医学科学发展迅速, 临床诊断和给药剂量尤其需要经过独立验证。在法律允许的最大范围内, 爱思唯尔、译文的原文作者、原文编辑及原文内容提供者均不对译文或因产品责任、疏忽或其他操作造成的人身及/或财产伤害及/或损失承担责任, 亦不对由于使用文中提到的方法、产品、说明或思想而导致的人身及/或财产伤害及/或损失承担责任。

Printed in China by China Machine Press under special arrangement with Elsevier (Singapore) Pte Ltd. This edition is authorized for sale in the People's Republic of China only, excluding Hong Kong SAR, Macau SAR and Taiwan. Unauthorized export of this edition is a violation of the contract.

本书封底贴有 Elsevier 防伪标签, 无标签者不得销售。

## 嵌入式系统: 硬件、软件及软硬件协同 (原书第 2 版)

出版发行: 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: 缪 杰

责任校对: 李秋荣

印 刷: 北京瑞德印刷有限公司

版 次: 2018 年 2 月第 1 版第 1 次印刷

开 本: 185mm × 260mm 1/16

印 张: 27.5

书 号: ISBN 978-7-111-58887-0

定 价: 119.00 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) 88379426 88361066

投稿热线: (010) 88379604

购书热线: (010) 68326294 88379649 68995259

读者信箱: [hzit@hzbook.com](mailto:hzit@hzbook.com)

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光 / 邹晓东



文艺复兴以来，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；也正是这样的优势，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相助，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，有的还专门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](http://www.hzbook.com)

电子邮件：[hzjsj@hzbook.com](mailto: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



当 Tammy Noergaard 第一次告诉我她想写一本全方位介绍如何建立嵌入式系统的书时，我试图劝阻过她。该领域如此庞大，需要了解电子技术、逻辑电路、计算机设计、软件工程、C 语言、汇编语言以及其他种种。但是在我们讨论的时候，她向我描述了这个行业在这方面的文献如何缺乏一本权威著作。我提醒她这个项目太大了。

经过一年多的讨论之后，联邦快递送来了这本书的审校副本。英文原书共有大约 700 页，几乎有关于这个方面的任何其他著作的两倍厚。你现在拿到的这本书实实在在的“工程师和程序员的综合指南”。当然，其中没有介绍 PIC 定时器的编程细节，但本书涉及的范围非常广泛而重要。

Tammy 从电子技术的基本原理开始，进一步到软件再到昂贵的最终维护阶段。她将硬件和软件作为一个集成的整体来对待，这是符合嵌入式系统的本质的。然而，讽刺的是，开发者却越来越专业化了。没有几个软件人员会了解晶体管，而太多的电子工程师无法准确地详细描述出中间件是什么。我恐怕读者可能会跳过那些与眼前项目不相关的章节。

抵制住任何这样的诱惑！要通过拓宽你的视野，全方位了解这个迷人的领域，从而成为一名真正的大师，一个嵌入式专家。工程师是专业人士，你我都深知这一点。然而，真正的专业人士是那些学习新事物、应用新技术来解决问题的人。想想医生：20 世纪 40 年代，青霉素的发现和生产完全改变了医药行业。任何忽视这种新技术的医生（他们在实践中依旧只使用在大学里学到的技能）突然仿佛变成了屠夫。软件和硬件开发人员面临着同样的情况。我上学的时候还没有 C 语言。FPGA 也尚未发明。使用 GOTO 语句还很正常。我们学习的是借助原始工具链用机器代码为微处理器编程。而今天，我们知道变化有多么大。

变化总是越来越多，而变化的一阶导数是一个不断升值的正数。专业开发人员应该把这本书完整地读完，并不断寻求其他信息来源。如果你做不到每个月至少浏览几本技术杂志，每年读几本这样的书，那么不需要白垩纪小行星你都要变成恐龙了。

本书有些内容可能会让你感到惊讶。数据手册写了 10 页？事实上，数据手册是“合同式”承诺材料浓缩版的正式汇编。供应商承诺，只要我们以约定的方式使用这个产品，则它必然能够按照承诺工作。如果违反了任何可能的众多的指标规范，部件会不起作用或不能可靠工作。如果某些部件耗散功率不低于 100W，那么即使像热特性那样的参数指标也与设备的指令集一样重要。

Tammy 不惜使用大量的示例来阐明那些不易理解的知识点。工程（无论是硬件还是软件）是构建事物和解决问题的艺术。学者可以只研究纯理论，而我们实践开发人员通常是要通过了解事物的工作原理来得到最好的解决方案。因此，关于设备驱动程序章节解释了构建这些通常较复杂的代码的繁杂程度，但是使用了与大量的实际应用示例相结合的方式来解决这些问题。

最后，Tammy 关于嵌入式系统的体系结构业务周期的论点与我有强烈的共鸣。我们不会只是为了让自己心情愉快来建立这些东西（尽管我们当然希望如此），我们的目的是要解

决重要的业务问题。我们做出的每一个决定都有其业务含义。功率要求过低和开发成本飙升，有时会导致项目不可行。对问题的糟糕分析（导致你不计成本添加过多闪存）可能会导致成本高到令人无法接受。如果从一个失败的公司选择了某些组件（硬件或软件），你将会因此遭受重大损失。

从这本书中获益，让自己在事业发展中永不过时吧。

Jack Ganssle

## 致 谢 |

Embedded Systems Architecture: A Comprehensive Guide for Engineers and Programmers, Second Edition

我创作本书第 2 版来回馈读者和审校者，我希望他们能够惊喜地看到他们的众多建议已经纳入了这本书中。他们包括 Al M. Zied 博士，我的两个兄弟（尤其是我的弟弟，他给了我最初撰写本书的灵感），Jack Ganssle 和 Steve Bailey。

感谢 Elsevier 出版社，特别感谢 Elsevier 团队为此书顺利出版所做的努力和付出。

我也要感谢我的 Terma 团队的成员，他们是最有天赋的人，我很幸运能够和他们以及我在索尼电子公司时的导师 Kazuhisa Maruoka 一起工作。Maruoka 先生耐心地培训我设计电视机，建立了我成长的强大基础，还有索尼电子公司的经理 Satoshi Ishiguro，是他决定雇用了我。引领我写这本书的是我在嵌入式系统领域的经历，这些经历从我在日本和圣地亚哥的索尼公司与这些伟大人物的合作开始。

感谢我家人的支持。我美丽的女儿 Mia 和 Sarah，谢谢她们总是给我无尽的笑容、拥抱和亲吻。感谢在南加州的妈妈和兄弟姐妹他们的支持鼓励着我前行。

最后，要特别感谢 Kenneth Knudsen 和他热情友好的家人。此外，感谢美国驻丹麦大使馆的支持，感谢在丹麦的其他美国同胞，以及可爱的丹麦人，很幸运与你们相识！我在丹麦努力完成这个版本的时候，是你们照亮了我最艰难的日子。感谢在丹麦的 Mia、Sarah 和我的所有其他朋友。我会永远感谢。

Tammy Noergaard 特别有资格撰写全方位介绍嵌入式系统的书籍。她从业以来，在产品开发、系统设计和整合、运营、销售、市场营销以及培训方面都获得了丰富的经验。她拥有使用多种硬件平台、操作系统、中间件以及编程语言的设计经验。Tammy 曾经是索尼公司开发和测试用于模拟电视机嵌入式软件的首席软件工程师，并且曾负责管理和培训新的嵌入式工程师和程序员。她在日本以及加利福尼亚州帮助开发的电视机曾在《消费者报道》杂志上被评为第一名。她已经在国际上从事咨询顾问工作多年，服务的公司包括 Esmertec 和 WindRiver 等，并曾在加州大学伯克利分校、斯坦福大学担任工程课程的客座讲师，曾应丹麦奥尔胡斯大学的邀请，为专业人士和学生做技术讲座。Tammy 多年来多次在嵌入式互联网会议以及圣何塞的 Java 用户小组做专业报告。最近，她以其专业经验在丹麦帮助其他团队成员和组织成功建立一流的嵌入式系统。



# 目 录

Embedded Systems Architecture: A Comprehensive Guide for Engineers and Programmers, Second Edition

出版者的话  
第1版推荐序  
致谢  
关于作者

## 第一部分 嵌入式系统简介

第1章 嵌入式系统设计的系统化方法	2
1.1 什么是嵌入式系统	2
1.2 嵌入式系统的体系结构简介	3
1.3 嵌入式系统模型	5
1.4 为什么使用整体化、体系结构化的系统工程方法? 黄金法则是什么	6
1.5 小结	11
习题	11
尾注	13
第2章 了解设计标准	14
2.1 编程语言概述及其标准示例	20
2.1.1 影响嵌入式系统体系结构的编程语言示例: 脚本语言、Java 和 .NET	22
2.1.2 垃圾回收机制	25
2.1.3 处理 Java 字节码	27
2.2 标准与网络	30
2.2.1 互连设备间的距离	30
2.2.2 物理介质	31
2.2.3 网络架构	33
2.2.4 OSI 模型	33
2.3 基于多种标准的设备示例: 数字电视 (DTV)	43
2.4 小结	55
习题	56
尾注	57

## 第二部分 嵌入式硬件

第3章 嵌入式硬件的组件和嵌入式系统板	60
3.1 学习硬件的第一节课: 学习读懂原理图	60
3.2 嵌入式系统板和冯·诺依曼模型	63
3.3 给硬件供电	65
3.4 基础硬件材料: 导体、绝缘体和半导体	67
3.5 电路板上和芯片中常见的无源元件: 电阻器、电容器和电感器	70
3.5.1 电阻器	70
3.5.2 电容器	73
3.5.3 电感器	74
3.6 半导体以及处理器和存储器的有源构建模块	75
3.6.1 二极管	75
3.6.2 晶体管	76
3.6.3 从基本的门电路构建出更复杂的电路	80
3.6.4 时序逻辑和时钟	81
3.7 全部整合到一起: 集成电路	83
3.8 小结	85
习题	85
尾注	89
第4章 嵌入式处理器	90
4.1 ISA 体系结构模型	92
4.1.1 功能	92
4.1.2 ISA 模型	97
4.2 处理器内部设计	102
4.2.1 中央处理单元	104
4.2.2 片上存储器	115
4.2.3 处理器 I/O	124

4.2.4 处理器总线 .....	141
4.3 处理器性能 .....	142
4.4 阅读处理器的数据手册 .....	144
4.5 小结 .....	153
习题 .....	153
尾注 .....	155
<b>第 5 章 板载存储器 .....</b>	<b>156</b>
5.1 ROM .....	159
5.2 RAM .....	162
5.3 辅助存储器 .....	168
5.4 外部存储器的存储管理 .....	171
5.5 板载存储器及其性能 .....	172
5.6 小结 .....	173
习题 .....	173
尾注 .....	174
<b>第 6 章 板载 I/O .....</b>	<b>175</b>
6.1 数据管理: 串行 I/O 与并行 I/O .....	177
6.1.1 串行 I/O .....	177
6.1.2 并行 I/O .....	184
6.2 互连 I/O 组件 .....	191
6.2.1 互连 I/O 设备和嵌入式系统板 .....	191
6.2.2 互连 I/O 控制器和主 CPU .....	193
6.3 I/O 与性能 .....	193
6.4 小结 .....	194
习题 .....	194
尾注 .....	197
<b>第 7 章 板载总线 .....</b>	<b>198</b>
7.1 总线仲裁和时序 .....	199
7.2 将总线与其他板载组件集成 .....	206
7.3 总线性能 .....	207
7.4 小结 .....	207
习题 .....	208
尾注 .....	209

### 第三部分 嵌入式软件介绍

<b>第 8 章 设备驱动程序 .....</b>	<b>213</b>
8.1 示例 1: 中断处理的设备驱动程序 .....	216
8.1.1 中断优先级 .....	218
8.1.2 上下文切换 .....	221
8.1.3 中断设备驱动程序伪代码示例 .....	222
8.1.4 中断处理及其性能 .....	228
8.2 示例 2: 存储器设备驱动程序 .....	229
8.3 示例 3: 板载总线设备驱动程序 .....	246
8.4 板载 I/O 驱动程序示例 .....	251
8.4.1 示例 4: 初始化以太网驱动程序 .....	251
8.4.2 示例 5: 初始化 RS-232 驱动程序 .....	264
8.5 小结 .....	268
习题 .....	268
尾注 .....	270
<b>第 9 章 嵌入式操作系统 .....</b>	<b>271</b>
9.1 什么是进程 .....	274
9.2 多任务和进程管理 .....	276
9.2.1 进程实现 .....	276
9.2.2 进程调度 .....	285
9.2.3 任务间通信和同步 .....	294
9.3 内存管理 .....	299
9.3.1 用户存储空间 .....	300
9.3.2 内核存储空间 .....	308
9.4 I/O 和文件系统管理 .....	309
9.5 操作系统标准示例: POSIX .....	310
9.6 操作系统性能指南 .....	312
9.7 选择正确的嵌入式操作系统以及 BSP .....	312
9.8 小结 .....	313
习题 .....	314
尾注 .....	316

<b>第 10 章 中间件和应用软件</b> .....	317
10.1 什么是中间件 .....	317
10.2 什么是应用程序 .....	319
10.3 中间件示例 .....	319
10.4 应用层软件示例 .....	345
10.4.1 FTP 客户端应用软件示例 .....	346
10.4.2 SMTP 和电子邮件示例 .....	348
10.4.3 HTTP 客户端和服务 器示例 .....	352
10.4.4 对应用整合的简要说明 .....	354
10.5 小结 .....	358
习题 .....	358
尾注 .....	360
 <b>第四部分 系统整合：设计与开发</b>	
<b>第 11 章 定义系统：创建体系结构和 设计文档化</b> .....	362
11.1 创建嵌入式系统体系结构 .....	363
11.1.1 阶段 1：打好坚实的技术 基础 .....	364
11.1.2 阶段 2：了解嵌入式系统的 体系结构业务周期 .....	364
11.1.3 阶段 3：详细描述体系结构 的模式和参考模型 .....	372
11.1.4 阶段 4：创建体系结构的 框架 .....	378
11.1.5 阶段 5：体系结构的 文档化 .....	380
11.1.6 阶段 6：对体系结构进行 分析和评估 .....	381
11.2 小结 .....	382
习题 .....	383
尾注 .....	384
 <b>第 12 章 嵌入式系统设计的最后 阶段：实现和测试</b> .....	385
12.1 设计的实现 .....	385
12.1.1 主要的软件实用工具：在 编辑器或 IDE 中编写代码 .....	386
12.1.2 CAD 和硬件 .....	387
12.1.3 翻译工具：预处理器、 解释器、编译器和链接器 .....	388
12.1.4 调试工具 .....	390
12.1.5 系统启动 .....	396
12.2 对设计的质量保证和测试 .....	403
12.3 结论：维护与嵌入式系统及 其他 .....	406
习题 .....	406
尾注 .....	407
 <b>附录 A 项目和练习</b> .....	408
 <b>附录 B 原理图符号</b> .....	423

# 嵌入式系统简介

嵌入式系统涉及领域广泛而多样，难以给出确定的定义或描述。不过，在第1章中介绍了一个可应用于任何嵌入式系统的实用模型。引入该模型来帮助读者了解构成不同类型电子设备的主要组件，无论其复杂性或差异。第2章介绍并详细描述了构建嵌入式系统时所遵循的通用标准。因为这本书是对嵌入式体系结构的概述，不可能涵盖所有可能实现的基于标准的组件。因此，只选择了当前基于标准的组件的典型示例，例如网络和Java，以展示标准如何定义嵌入式系统中的主要组件。其目的是让读者能够使用模型、标准和实际应用示例背后的方法来了解各种嵌入式系统，并且能够将任何其他标准应用于嵌入式系统的设计。

# 嵌入式系统设计的系统化方法

## 本章内容

- 嵌入式系统的定义
- 嵌入式系统的设计过程
- 嵌入式系统的体系结构
- 体系结构的作用
- 本书其他章节的概述

## 1.1 什么是嵌入式系统

嵌入式系统是一个特定的计算机系统，区别于诸如个人电脑或超级计算机等其他类型的计算机系统。不过，你会发现对“嵌入式系统”很难给出一个固定不变的定义，因为它一直在发展：技术不断进步的同时，实现各种硬件和软件组件的成本大幅度降低。国际上本领域的很多传统定义描述都已经产生了很大的变化。因为读者很可能会看过这些描述和定义，因此了解产生上述问题背后的原因以及为什么这些描述的准确性如今会发生变化是很有必要的，这样能够帮助我们在了解掌握相关知识的基础上来讨论这个问题。下面就是一些比较常见的对嵌入式系统的描述：

- 嵌入式系统在硬件和软件功能上的局限性比 PC 大得多。这个说法对于多数嵌入式系统类型来说是无误的。拿硬件局限性来说，可以指处理器性能、功耗、存储器、硬件功能等方面。对于软件，通常指的是相对于 PC 软件的局限性——应用软件的数量更少，功能更弱，没有或者仅有一个能力有限的操作系统，或者更低抽象层次的代码。然而，这个说法在当前只能说部分正确，之前甚至现在通常在 PC 中才有的板卡和软件已经被整合应用于更复杂的嵌入式系统设计中了。
- 一个嵌入式系统是被设计用于某一个特定功能的。多数嵌入式设备主要是被设计用于单一功能的。然而如今我们会看到诸如 PDA 与手机功能合为一体的智能手机，这样的嵌入式系统设计可以完成多种主要功能。同样，最新的数字电视机可以通过交互式的应用完成多种与传统“电视”不相关的通用功能，比如收发 email，上网浏览，玩游戏等。
- 嵌入式系统比其他计算机系统质量和可靠性要求更高的计算机系统。某些类别的嵌入式设备具备极高的质量和可靠性需求指标。例如，如果一辆在繁忙的高速路上行驶的汽车的引擎控制器失效，或者实施急救的关键医疗设备功能紊乱，都会产生严重的后果。不过，也有很多嵌入式设备（比如电视机、游戏机、手机等）在使用中出现个别问题会导致用户不便但不至于造成致命的后果。
- 某些设备被称为嵌入式系统，比如 PDA 或者上网本，但这并非真正的嵌入式系统。有一些关于一个部分满足传统的嵌入式系统定义的计算机系统是否属于嵌入式系统

或是其他类别的讨论。有些人认为：把这些更复杂的设计（如 PDA）当作是嵌入式系统的，是非技术的市场和销售人士，而非工程师。事实上，那些设计此类系统的工程师对此类系统是否应该归类于嵌入式系统也意见不一。传统的嵌入式的定义是否应该继续发展或指定一个计算机系统的新领域来包含这些更复杂的系统，应由行业内人士来最终确定。目前，由于在传统的嵌入式系统和通用 PC 系统之间并没有新的行业支持指定的计算机系统定义，本书支持嵌入式系统定义进化的观点，即嵌入式系统包含这些更复杂的计算机系统设计。

- 几乎所有工程市场领域的电子设备都归类于嵌入式系统（见表 1-1）。总之，在属于“某类计算机系统”之外，对于如此广泛的嵌入式系统设备唯一保持准确的特征描述就是，没有单一的定义可以适用于所有的系统。

表 1-1 嵌入式系统及其应用领域示例<sup>[1]</sup>

应用领域	嵌入式设备
汽车	点火系统、发动机控制、刹车系统（如防抱死刹车系统）
消费电子	数字和模拟电视、机顶盒（DVD、VCR 等）、个人数字助理（PDA）、厨房电器（电冰箱、面包机、微波炉等）、汽车电子、玩具/游戏机、电话/手机/传呼机、照相机、全球定位系统（GPS）
工业控制	机器人技术和控制系统（制造业）
医疗	输液泵、透析机、义肢、心脏监护
网络	路由器、集线器、网关
办公自动化	传真机、复印机、打印机、显示器、扫描仪

## 1.2 嵌入式系统的体系结构简介

为了让产品设计具备坚实的技术基础，所有的团队成员首先必须要充分理解其准备制造的设备的**体系结构**。嵌入式系统的体系结构相当于对这个嵌入式设备的抽象化表示，指的是对一个系统的概括性描述，通常不包括具体实现的细节，比如软件源代码或硬件电路设计<sup>[3]</sup>。在体系结构级别上，嵌入式系统中的硬件和软件组件被替代性地表示为一些可交互的元素的组合。元素是指对实现细节已经被抽取出去的硬件和软件的表述，只保留了关于其行为及相互关系的信息。体系结构的元素可以集成在嵌入式设备内部，或者置于嵌入式设备外部与系统内部元素进行交互。总之，一个嵌入式体系结构包含了嵌入式系统的基本元素、元素与嵌入式系统的交互、每个独立元素的属性以及元素之间的交互关系。

体系结构级别的信息使用结构的形式来物理地表示。结构就是对体系结构的一个可能的表述，包含它对其自身元素、属性以及相互关系信息表达的集合。因此一个结构可以看作是在设计时和运行时对系统硬件和软件的一个“快照”，给出了一个特定的环境以及给定的元素集合。由于很难用“快照”来获取一个系统的所有复杂特性，故一个体系结构通常会由多个结构所组成。一个体系结构中所包含的所有结构本质上都是相互关联的，所有的这些结构一起构成了设备的嵌入式体系结构。表 1-2 总结了一些最常用的可以用来构建嵌入式体系结构的结构类型，并且指出了某个结构的元素通常代表什么，以及这些元素如何互相关联。表 1-2 介绍的内容将在后文中定义和讨论，其中同时也展示出了可用于表示一个嵌入式系统体系结构中的结构的多样性。体系结构以及它们的结构（它们之间如何互相关联，如何构建出一个体系结构，等等）将在第 11 章中详细讨论。



表 1-2 体系结构中的结构示例<sup>[4]</sup>

结构类型 <sup>①</sup>		详细描述
模块		元素（称为模块）被定义为嵌入式设备中实现不同功能的组件（系统实现正常功能的基本硬件/软件）。市场营销和销售架构图通常是用模块化结构来表示的，因为软件或硬件一般会打包成各种模块（操作系统、处理器、Java 虚拟机等）来销售的
	使用关系（也称为子系统结构和组件）	一种模块化结构描述系统，用来表示系统运行时模块之间的相互使用关系（如一个模块使用了哪些其他模块）
	层	一种使用关系的结构，模块按层组织（层次化），上层的模块使用（依赖）下层的模块
	内核	该结构用来表示使用操作系统内核模块（服务）的模块或由内核进行操作的那些模块
	管道架构	该结构顺序地表示模块，描述模块通过其使用产生的转换过程
	虚拟机	该结构表示使用虚拟机模块的那些模块
	分解	一种模块化结构，其中某些模块实际上是其他模块的子单元（分解后的单元）以及它们之间的相互关系的说明。通常用来确定资源分配、项目管理（规划）、数据管理（比如封装、私有化）等
	类（也称为泛化）	一种表示软件的模块化结构，其中的模块被称为类，其相互关系是根据面向对象方法定义的，其中类是其他类的继承或父类的实例。在有类似基础的系统设计中很实用
组件与连接器		这些结构是由组件（主要的硬件/软件处理单元，如处理器、JVM）或者连接器（组件间互联的通信机制，如硬件总线、软件操作系统消息等）这些元素组成的
	客户端/服务器（也称为分布式）	用于系统运行时的结构，其中组件是客户端或服务器（或对象），连接器是用于在客户端和服务器（或对象）之间互相通信的机制（协议、消息、数据包等）
	进程（也称为通信进程）	此结构是一种软件结构，用于包含有操作系统的系统。组件是进程或线程（见第 9 章），它们的连接器为进程间通信机制（共享数据、管道等）。这种结构对于调度和性能分析十分有用
	并发性和资源	此结构是包含有操作系统的系统的一个运行时快照，其中各个组件通过并行运行的线程连接（见第 9 章）。实质上，此结构被用于资源管理和判断共享资源是否存在问题，以及确定哪些软件可以并行执行
	中断	表示系统的中断处理机制的结构
	调度（最早截止时间优先 EDF、按优先级、轮询）	此结构用来表示线程的任务调度机制，来说明操作系统调度器的公平性
	内存	用来表示内存和数据组件的运行结构表述，包括内存的分配和释放（连接器）机制——本质上是系统的内存管理方案
	垃圾回收	此结构表示垃圾回收机制（详见第 2 章）
	分配	此结构用来表示系统的内存分配机制（静态或动态、大小等）
	安全性和可靠性	此运行时结构中，冗余组件（硬件和软件元素）和它们互相通信的机制描述了系统在出现问题时的可靠性和安全性（从多种问题中恢复的能力）
分配		此结构表示不同环境中的软件和硬件元素与外部元素之间的关系
	工作分派	此结构将模块功能任务分派给不同的开发和设计小组。通常用于项目管理
	实现	这是一个软件结构，它指明了软件在开发系统的文件系统的位置
	部署	此运行时结构的元素是硬件和软件，元素之间的关系为软件映射到硬件中的位置（归属于、迁移至，等等）

① 请注意，本书中在许多情况下术语“体系结构”和“结构”（一个快照）有时可互换使用。

总之，嵌入式系统的体系结构可以用于解决在项目初期遇到的这些类型的问题。无须定义或了解任何内部实现细节，嵌入式设备的体系结构可以作为最初的工具，用来分析并作为高级蓝图来定义设计的基础框架、可能的设计选项以及设计的约束条件。使得体系结构方法如此强大的原因是它提供的通俗、快速与不同类型人员交流某个设计的能力，无论其是否具备相关技术背景，这种方法甚至可以作为项目规划或是实际设计一个设备的基础依据。由于它清晰地刻画出了系统的需求，体系结构可以作为一个坚实的基础用于分析和测试设备在不同环境条件下的特性及性能。此外，如果被正确地理解、创建和利用，体系结构可以通过对实现各类元素的风险的论证来精确评估并降低设计开销，进而降低此类风险。最终，体系结构中的各种结构可以调整并持续用于设计未来的具有类似特性的产品，由此实现设计知识的复用，从而减少未来设计开发的开销。

通过使用本书描述的基于体系结构的方法，希望能够让读者明白：**详细描述并理解一个嵌入式系统的体系结构是实现一个良好的系统设计的关键所在。**这是因为除了上述理由之外还有以下好处：

1. 每一个嵌入式系统都有其体系结构（无论是否明文定义），因为每一个嵌入式系统都是由交互的元素（包括硬件和软件）构成的。体系结构的定义即这些元素及其相互关系的表述的集合。应该通过事先详细描述体系结构从而掌控整个设计过程，而不是因未事先定义好体系结构花更大代价被动地接受一个不良的系统。

2. 因为一个嵌入式体系结构是从多视角对系统表现的描述，因此这是理解所有主要系统元素的有效工具：各个部件如何部署以及每个元素具备什么样的行为。嵌入式系统中没有与其他部分隔绝的元素，设备中的每个元素都会以各种方式与其他元素实现交互。此外，元素的外部可见特性在其与不同的元素一起工作时可能表现不尽相同。如果不能理解隐藏在各种元素的功能及性能背后的所以然，就会难以判断系统在不同的实际运行条件下会有何种行为。

即使是仅有粗略的不规范的体系结构，也比没有强。只要体系结构以某种方式将一个设计的各个核心部件及其相互关系表达清楚，就能为项目成员提供设备是否满足其需求以及如何成功构建出此系统的关键信息。

### 1.3 嵌入式系统模型

在本书中，使用了一系列体系结构的结构设计来介绍嵌入式系统的技术概念和基本原理。同时也介绍了新的体系结构工具（即参考模型）作为这些体系结构的设计基础。在最高层次上，如图 1-1 所示，首要的架构工具简称为嵌入式系统模型，它用来引入位于嵌入式系统设计中的主要元素。

嵌入式系统模型意味着所有的嵌入式系统在最高层次上具有一个相似之处：也就是说它们都有至少一层（硬件层）或者容纳所有部件的所有层（硬件层、系统软件层和应用软件层）。硬件层包括所有位于嵌入式板卡上的主要物理元件，而系统软件层和应用软件层包括所有包含于嵌入式系统并在系统中被处理执行的软件。

这个参考模型本质上是一个嵌入式体系结构的层次化（模块化）表达，从中可得到一个模块化的体系结构设计。不考虑表 1-1 中所示的设备间的差异，通过形象地将这些设备中的

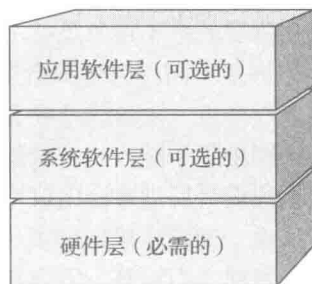


图 1-1 嵌入式系统模型

部件分组为不同的层，就有可能理解所有这些系统的体系结构。分层的概念不仅仅用于嵌入式系统设计中（体系结构与所有计算机系统相关，嵌入式系统是其中一种类型），它是一种可用于设计嵌入式系统设计的将数以百计的软、硬件组件组合可视化的有益工具。大体来说，本书选择这种嵌入式系统体系结构的模块化表示方法作为主要结构是基于两个原因：

1. 主要元素的可视化表示及其相关的功能。分层的方法使得读者能够直观地了解嵌入式系统中的各种部件及其相互关联。

2. 模块化的体系结构的表示方法通常是用来表示整个嵌入式项目结构的方法。这主要是因为在这种结构中，各个模块（元素）通常都是独立工作的。这些元素之间又具有高度的交互性，这样将这类元素分层表达既改善了系统的结构化组织，又避免了将复杂的交互过分简单化，也没有忽略必需的功能。

本书的第二和第三部分详细描述了嵌入式系统模型层次结构中的主要模块，实质上描述了存在于大多数嵌入式系统中的主要部件。第四部分从设计和开发的角度将不同的层组合在一起，向读者展示如何应用前面章节所介绍的技术概念以及本章介绍的体系结构化过程。整本书中，均通过一些真实应用的建议和例子来体现技术理论的实用意义，并且作为嵌入式概念的主要教学工具。在你阅读这些实例的时候，为了能够从本书中获得最大的收益，并且能够将所学到的知识应用于未来的嵌入式项目中，我建议读者注意以下几点：

- 所有这些例子所遵循的模式，不仅对应于局部内容所介绍的技术概念，也对应于更高层次的体系结构的表达。这些模式可以普遍用于理解或者设计任何嵌入式系统，而不仅限于文中所分析的系统。
- 这些信息是从哪里得来的。因为读者可以从各种各样的来源获取嵌入式系统设计方面有价值的信息，包括互联网、嵌入式相关杂志中的文章、嵌入式系统相关会议、产品说明书、用户手册、编程手册和设计图纸等。

## 1.4 为什么使用整体化、体系结构化的系统工程方法？黄金法则是什么

本书针对嵌入式系统设计使用了一种整体化、体系结构化的系统工程方法，来阐明嵌入式系统本身和构成其内部设计的不同类型的组件。这是因为确保一个工程团队能够成功的最有效方法之一就是通过对系统化方法来详细描述体系结构并实现其设计。

系统化的工程方法解决了一个现实问题，比仅使用单纯的嵌入式系统技术能够更有利于产品设计的成功。换句话说，要记住的**第1法则**就是：构建一个嵌入式系统并能够成功地产品化需要的绝不仅仅是技术！

许多不同的因素都会影响一个嵌入式设计的构建及其产品化进程。这包括金融、技术、商业导向性、政治以及社会资源等各方面的影响。这些不同类型的因素产生需求，而需求反过来则决定了嵌入式系统的体系结构，然后这个体系结构成为设备制造的基础，得到的嵌入式系统设计反过来会向设计团队提供系统需求和功能的反馈信息。因此，对嵌入式系统设计者来说，理解和规划好项目的非技术要素与技术要素同等重要，比如社会、政治、法律以及金融方面的影响等。因为导致一个嵌入式系统项目失败最常见的错误通常不是某一个特定的因素本身，比如：

- 定义及获得系统设计的过程；
- 成本的限制；
- 系统完整性的判定，比如可靠性和安全性；