

计算机系列教材

嵌入式系统体系、 原理与设计

张凯龙 编著

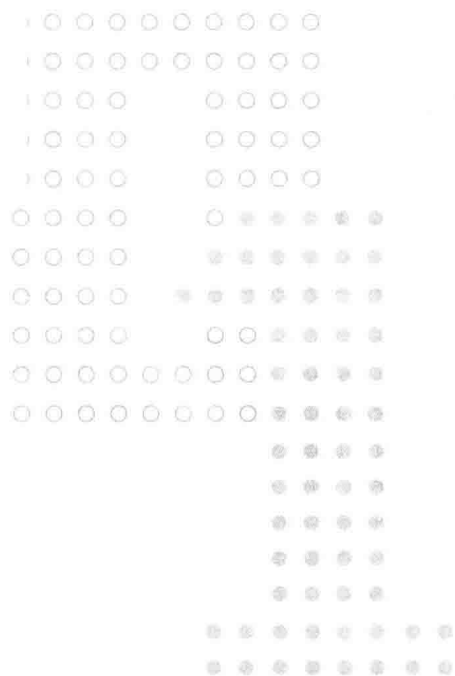


清华大学出版社

计算机系列教材

张凯龙 编著

嵌入式系统体系、 原理与设计



清华大学出版社
北京

内 容 简 介

在梳理不断演化和丰富的嵌入式系统理论、方法、技术及应用脉络的基础上,本书致力于构建“自架构、原理到设计,由宏观特性到微观原理,硬件与软件、理论与实践相融合”的多维知识体系。开篇首先以经典的嵌入式系统概念、内涵以及必要的基础知识作为铺垫进行导入,进而从嵌入式硬件、嵌入式软件以及设计与验证方法三个方面展开论述。其中,硬件部分深入地阐述了嵌入式处理器、存储器等子系统以及最小系统外围电路和 I/O 接口扩展等硬件组件的体系结构、工作原理、设计方法与示例。软件部分重点阐述了嵌入式软件系统的组成结构与基础组件,嵌入式(实时)操作系统共性的功能模型、服务机制、设计方法和典型系统示例,以及嵌入式图形库、文件系统、数据库等典型的嵌入式软件组件。设计与验证方法部分则分析和阐述了嵌入式软件与系统的开发机制以及常见调试、测试、仿真方法的技术原理。全书写作采用了整体体系贯穿与局部深度剖析相结合、共性机理归纳与设计示例分析相呼应的思路,力求系统、生动又深入浅出地阐述嵌入式系统技术涉及的体系结构、技术原理与设计机制,并在突出技术性、新颖性的同时又注重内容的易读性和实用性。

本书适合供高等院校计算机科学与技术、物联网、自动化、电子信息及电气工程等相关专业的高年级本科生、研究生学习,尤其适合作为“嵌入式(计算机)系统”类课程的授课教材或参考教材,同时也适合作为专业设计人员的技术参考用书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

嵌入式系统体系、原理与设计/张凯龙编著. —北京:清华大学出版社,2017

(计算机系列教材)

ISBN 978-7-302-46721-2

I. ①嵌… II. ①张… III. ①微型计算机—系统设计—教材 IV. ①TP360.21

中国版本图书馆 CIP 数据核字(2017)第 036670 号

责任编辑:张 民 李 晔

封面设计:常雪影

责任校对:梁 毅

责任印制:沈 露

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社 总 机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课件下载: <http://www.tup.com.cn>, 010-62795954

印 装 者:北京国马印刷厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:33

字 数:802千字

版 次:2017年5月第1版

印 次:2017年5月第1次印刷

印 数:1~2000

定 价:59.50元

产品编号:072570-01

嵌入式系统作为信息技术领域的重要技术发展方向,已广泛融入到军事国防、工业装备、智能交通、医疗电子、信息通信等诸多领域。在经历网络化发展阶段之后,嵌入式系统技术的发展已经开始步入微型化、信息物理融合以及智能化的时代,是互联网+、物联网、智能制造等众多领域进行技术变革和跨越式发展的重要支撑。随着相关技术的飞速发展,嵌入式系统技术完备、庞大的知识体系也在不断演化和丰富,对嵌入式系统技术的学习随之迈入更为深化和高级的阶段。

在总结嵌入式系统知识特点的基础上,本书全面地梳理了嵌入式系统技术的知识结构,并以构建“自架构、原理到设计,由宏观特性到微观原理,以及硬件与软件、理论与实践相融合”的多维、立体知识体系为总目标,以整体体系贯穿与局部深度剖析相结合、共性机理归纳与设计示例分析相呼应的方法展开相关内容的阐述和讨论。全书内容总体分为嵌入式系统概述、硬件、软件及设计方法四个大的部分,共 11 章,知识结构如图 1 所示。其中,第 1 章为概述,第 2 章至第 6 章为嵌入式硬件部分,第 7 章至第 10 章为嵌入式软件相关内容,第 11 章为开发与验证方法,具体内容安排如下。

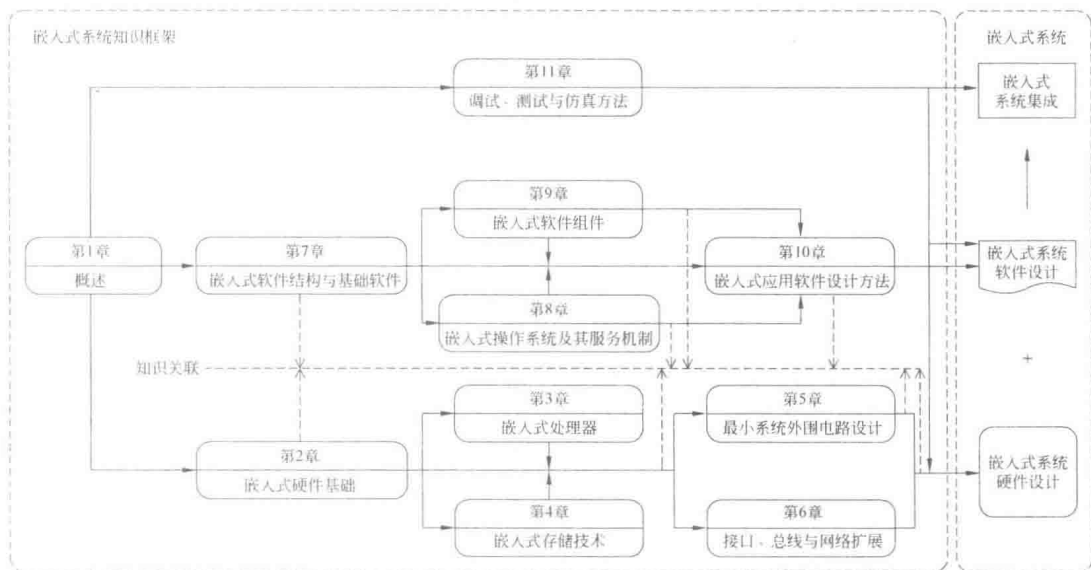


图 1 本书的知识架构

第 1 章,从计算技术、通信与网络技术的发展导入,分析了计算装置的“可嵌入”发展历程,并重点讨论了嵌入式系统的内涵、组成、特点及其趋势。

第2章,总结、论述了与本书后续内容密切相关的硬件电路知识;进而阐述了嵌入式硬件的架构模型与子系统组成、典型嵌入式系统的硬件形式以及基本的硬件设计方法。

第3章,由嵌入式处理器的组成模型和典型逻辑架构出发,总结、阐述了不同类型嵌入式处理器的体系、组成、特点及其差异,在此基础上分析了几种典型嵌入式处理器体系的机制和特性。

第4章,归纳、讨论了嵌入式存储器子系统的体系与模型,重点分析了不同类型存储器的架构、原理、特性以及微观的电路机制,并简要阐述了存储器的测试与验证方法。

第5章,围绕最小系统硬件设计,阐述了电源、复位、时钟等外围电路的基本工作原理与设计机制,并对相关知识进行了延伸讨论。

第6章,面向完整的嵌入式系统硬件设计需要,分析、讨论了常见和典型 I/O 接口与总线的原理、特性及应用机制。

第7章,在分析嵌入式系统启动与运行过程的基础上,总结了嵌入式软件系统的典型架构,重点对系统软件中基础软件组件的机制与特性进行了分析和讨论。

第8章,系统地讨论了嵌入式(实时)操作系统的相关内容。本章首先归纳了嵌入式操作系统的架构与功能模型,进而分析了不同嵌入式操作系统中的共性服务机制和方法,最后给出了几个典型嵌入式操作系统的分析实例。

第9章,结合嵌入式软件系统的设计需要,分析、阐述了嵌入式图形库、文件系统、数据库、协议栈等嵌入式软(硬)件组件的原理与机制。

第10章,综合阐述了嵌入式应用软件的相关设计方法,包括嵌入式应用软件结构、中断程序与数据共享问题、典型的嵌入式软件设计机制以及软件工程方法等。

第11章,根据嵌入式系统特点,总结了嵌入式软件与系统开发过程中调试、测试及仿真等验证技术的基本原理、工作机制和实施方法。

在本书内容的组织和编写过程中,作者以构建多维度的知识体系为目标,力求实现模型到实现、宏观到微观、硬件到软件、高级语言到低级语言的单点纵向穿透和多点横向融合。这一方面将有助于读者系统、深入地理解和掌握相关的知识原理、机制和方法,另一方面也有助于消除知识盲点,促进整个知识体系的融会贯通。阅读过程中,读者将发现本书的知识体系还实现了嵌入式系统技术到电子、物理、材料等学科知识以及经典示例的延伸与融合,这也使得本书内容更为丰富和生动。按照上述方式组织本书内容的另一个目的在于,尝试引导读者建立从多个角度来分析嵌入式系统设计问题的综合能力以及多维度的思维方式,在阅读本书内容的过程中读者将会有深刻体会。

嵌入式系统知识体系的体量较大且软硬件高度融合。作者虽致力于在本书中构建一个完整的知识体系,但因篇幅有限,对部分相关内容的细节未能进行深入讨论,请读者根据兴趣和实际工作需要延伸阅读与学习。同时,作者本人知识水平有限,虽在近一年半的写作中已尽全力,但书中谬误之处在所难免,虚心接受各位读者的意见和建议。

张凯龙

2016年1月13日于西安

F O R E W O R D

| | |
|------------------------------|-----|
| 第 1 章 概述 | /1 |
| 1.1 计算技术延伸 | /1 |
| 1.1.1 从开尔文设想到现代计算技术出现 | /1 |
| 1.1.2 计算机技术的发展 | /2 |
| 1.1.3 通信与网络技术发展 | /4 |
| 1.2 理解嵌入式(计算)系统 | /6 |
| 1.2.1 计算装置的“可嵌入”发展过程 | /6 |
| 1.2.2 嵌入式(计算)系统的内涵 | /8 |
| 1.2.3 相关术语辨析 | /10 |
| 1.3 嵌入式系统的组成与特点 | /12 |
| 1.3.1 体系结构与组成 | /12 |
| 1.3.2 应用及技术特点 | /13 |
| 1.4 发展趋势 | /16 |
| 1.5 知识体系小结 | /18 |
| 习题 | /19 |
| 第 2 章 嵌入式硬件基础 | /20 |
| 2.1 硬件基础 | /20 |
| 2.1.1 器件术语 | /20 |
| 2.1.2 电路术语与符号 | /23 |
| 2.1.3 基本元件与逻辑 | /25 |
| 2.2 嵌入式系统硬件组成 | /30 |
| 2.2.1 硬件基本架构模型 | /30 |
| 2.2.2 核心逻辑与子系统 | /32 |
| 2.3 典型嵌入式系统硬件形式 | /38 |
| 2.3.1 基于通用计算机扩展的嵌入式硬件 | /38 |
| 2.3.2 基于领域标准模块集成的嵌入式硬件 | /39 |
| 2.3.3 针对具体应用设计的嵌入式硬件 | /41 |
| 2.4 硬件设计基本方法 | /42 |
| 2.4.1 Protel EDA 软件 | /42 |
| 2.4.2 电路设计基本流程 | /45 |
| 2.4.3 电磁兼容性问题 | /47 |
| 2.4.4 低功耗问题 | /51 |

| | | |
|------------|----------------------|-------------|
| 2.4.5 | 嵌入式硬件设计示例 | /52 |
| 2.5 | 小结 | /60 |
| | 习题 | /61 |
| 第3章 | 嵌入式处理器 | /62 |
| 3.1 | 处理器模型与逻辑体系 | /62 |
| 3.1.1 | 处理器基本组成模型 | /62 |
| 3.1.2 | 处理器的基本逻辑体系 | /65 |
| 3.2 | 嵌入式处理器类型及特点 | /70 |
| 3.2.1 | 嵌入式微控制器 | /70 |
| 3.2.2 | 嵌入式微处理器 | /72 |
| 3.2.3 | 数字信号处理器 | /73 |
| 3.2.4 | 可编程逻辑器件 | /77 |
| 3.2.5 | 片上系统 | /84 |
| 3.2.6 | 多核、众核处理器 | /88 |
| 3.3 | 典型嵌入式处理器体系 | /95 |
| 3.3.1 | ARM 架构 | /95 |
| 3.3.2 | MIPS 架构 | /110 |
| 3.3.3 | PowerPC 架构 | /115 |
| 3.4 | 性能评价指标 | /124 |
| 3.5 | 小结 | /126 |
| | 习题 | /126 |
| 第4章 | 嵌入式存储技术 | /127 |
| 4.1 | 基本存储体系与模型 | /127 |
| 4.1.1 | 嵌入式系统存储体系 | /127 |
| 4.1.2 | 存储器结构模型 | /128 |
| 4.1.3 | 存储器基本操作流程 | /129 |
| 4.1.4 | 存储器技术指标 | /130 |
| 4.2 | 存储器分类及特性 | /131 |
| 4.2.1 | 随机访问存储器 | /132 |
| 4.2.2 | 只读存储器 | /145 |
| 4.2.3 | 混合存储器 | /150 |

| | | |
|------------|----------------------------|-------------|
| 4.3 | 存储器测试与验证 | /165 |
| 4.3.1 | 可读写存储器的测试 | /166 |
| 4.3.2 | 只读存储器测试 | /167 |
| 4.4 | 小结 | /169 |
| | 习题 | /169 |
| 第5章 | 最小系统外围电路设计 | /170 |
| 5.1 | 电源电路 | /170 |
| 5.1.1 | 电源电路设计方法 | /170 |
| 5.1.2 | 延伸：电源管理与低功耗设计 | /174 |
| 5.2 | 复位电路 | /179 |
| 5.2.1 | 上电复位 | /179 |
| 5.2.2 | 手动复位 | /184 |
| 5.2.3 | 看门狗复位 | /186 |
| 5.2.4 | 软件复位 | /187 |
| 5.3 | 时钟电路 | /191 |
| 5.3.1 | 信号源：振荡电路原理 | /191 |
| 5.3.2 | 多时钟管理 | /196 |
| 5.3.3 | 延伸：时钟管理与低功耗设计 | /201 |
| 5.4 | 电路抖动与消抖 | /202 |
| 5.4.1 | 抖动现象与危害 | /202 |
| 5.4.2 | 优化硬件消除抖动 | /203 |
| 5.4.3 | 软件消抖方式 | /205 |
| 5.5 | 小结 | /208 |
| | 习题 | /208 |
| 第6章 | 接口、总线与网络扩展 | /209 |
| 6.1 | 通用 I/O 与串行总线 | /209 |
| 6.1.1 | GPIO | /209 |
| 6.1.2 | SPI | /213 |
| 6.1.3 | SDIO | /222 |
| 6.1.4 | I ² S/IIS | /226 |
| 6.1.5 | I ² C/IIC | /232 |

| | | |
|--------------|----------------------------|-------------|
| 6.1.6 | UART 与 USART | /237 |
| 6.2 | 典型工业总线、背板总线及网络..... | /249 |
| 6.2.1 | CAN 总线 | /249 |
| 6.2.2 | 实时工业以太网..... | /259 |
| 6.2.3 | VPX 总线 | /264 |
| 6.3 | 无线接口与网络 | /274 |
| 6.3.1 | ZigBee | /274 |
| 6.3.2 | 蓝牙..... | /280 |
| 6.3.3 | Wi-Fi | /292 |
| 6.3.4 | NFC | /295 |
| 6.4 | 小结 | /300 |
| | 习题..... | /300 |
| 第 7 章 | 嵌入式软件结构与基础软件 | /302 |
| 7.1 | 计算系统的启动与运行过程 | /302 |
| 7.1.1 | 通用计算机的引导和启动过程..... | /302 |
| 7.1.2 | 嵌入式系统的引导与启动过程..... | /303 |
| 7.2 | 嵌入式软件体系与结构 | /305 |
| 7.2.1 | 基于裸机的嵌入式软件..... | /306 |
| 7.2.2 | 基于嵌入式操作系统的软件结构..... | /307 |
| 7.3 | 基础软件组件 | /308 |
| 7.3.1 | ROM Monitor | /308 |
| 7.3.2 | Boot Loader | /313 |
| 7.3.3 | BSP | /320 |
| 7.3.4 | 虚拟机监控程序..... | /332 |
| 7.4 | 小结 | /339 |
| | 习题..... | /339 |
| 第 8 章 | 嵌入式操作系统及其服务机制 | /341 |
| 8.1 | 架构与模型 | /341 |
| 8.1.1 | 微内核、超微内核架构 | /341 |
| 8.1.2 | 功能模型..... | /343 |
| 8.2 | 服务机制与特性 | /346 |

| | | |
|--------------|---------------------|-------------|
| 8.2.1 | 基本概念 | /346 |
| 8.2.2 | 关于实时 | /357 |
| 8.2.3 | 内核服务机制 | /362 |
| 8.3 | 典型嵌入式操作系统 | /379 |
| 8.3.1 | RTX51 | /379 |
| 8.3.2 | μ C/OS | /380 |
| 8.3.3 | 嵌入式 Linux 系列 | /382 |
| 8.3.4 | VxWorks | /387 |
| 8.3.5 | Windows 嵌入式操作系统 | /389 |
| 8.3.6 | 其他嵌入式操作系统 | /396 |
| 8.4 | 小结 | /401 |
| | 习题 | /402 |
| 第 9 章 | 嵌入式软件组件 | /403 |
| 9.1 | 嵌入式图形库与图形组件 | /403 |
| 9.1.1 | μ C/GUI 与 emWin | /403 |
| 9.1.2 | Nano-X Window | /405 |
| 9.1.3 | MiniGUI | /407 |
| 9.1.4 | QT | /413 |
| 9.1.5 | 可编程图形界面硬件模块 | /418 |
| 9.2 | 嵌入式文件系统 | /419 |
| 9.2.1 | JFFS/JFFS2 | /420 |
| 9.2.2 | Yaffs/Yaffs2 | /422 |
| 9.2.3 | UBIFS | /425 |
| 9.2.4 | 只读文件系统 | /428 |
| 9.2.5 | RAM 文件系统 | /430 |
| 9.3 | 嵌入式数据库 | /432 |
| 9.3.1 | SQLite | /432 |
| 9.3.2 | Berkeley DB | /434 |
| 9.3.3 | eXtremeDB | /435 |
| 9.4 | 嵌入式网络协议栈 | /437 |
| 9.5 | 小结 | /438 |
| | 习题 | /438 |

| | | | |
|---------------|------------------------|-------|------|
| 第 10 章 | 嵌入式应用软件设计方法 | | /439 |
| 10.1 | 嵌入式软件典型结构 | | /439 |
| 10.1.1 | 轮转结构 | | /439 |
| 10.1.2 | 前后台软件结构 | | /442 |
| 10.1.3 | 函数队列调度结构 | | /444 |
| 10.1.4 | 基于嵌入式操作系统的软件结构 | | /445 |
| 10.2 | 中断与数据共享问题 | | /447 |
| 10.2.1 | 中断服务程序设计 | | /447 |
| 10.2.2 | 消除数据共享问题 | | /451 |
| 10.3 | 嵌入式软件设计机制 | | /458 |
| 10.3.1 | 可重入代码设计 | | /458 |
| 10.3.2 | 软件看门狗方法 | | /462 |
| 10.3.3 | 避免频繁地创建、取消任务 | | /465 |
| 10.3.4 | 基于状态机方法设计软件逻辑 | | /466 |
| 10.3.5 | FPGA IP 核开发流程 | | /466 |
| 10.3.6 | 生成可执行的嵌入式软件 | | /469 |
| 10.4 | 软件工程方法 | | /477 |
| 10.4.1 | 模型驱动的开发方法：从 V 模型到 Y 模型 | | /477 |
| 10.4.2 | 软硬件协同设计 | | /479 |
| 10.4.3 | 制定编码规范 | | /480 |
| 10.5 | 小结 | | /482 |
| | 习题 | | /482 |
| 第 11 章 | 调试、测试与仿真方法 | | /483 |
| 11.1 | 嵌入式软件调试 | | /483 |
| 11.1.1 | 基于宿主机的调试 | | /484 |
| 11.1.2 | ROM Monitor 软件调试 | | /486 |
| 11.1.3 | ROM 仿真器 | | /487 |
| 11.1.4 | ICE 在线仿真器 | | /488 |
| 11.1.5 | JTAG 调试 | | /490 |
| 11.2 | 嵌入式软件测试 | | /492 |

| | | |
|--------|------------------|------|
| 11.2.1 | 基本测试方法与指标····· | /493 |
| 11.2.2 | 测试工具的使用····· | /498 |
| 11.3 | 系统仿真验证技术····· | /502 |
| 11.3.1 | 计算机仿真方法及其原理····· | /503 |
| 11.3.2 | 可视化技术····· | /505 |
| 11.4 | 小结····· | /507 |
| | 习题····· | /507 |
| | 参考文献····· | /508 |
| | 后记····· | /513 |

第 1 章 概 述

*Go, wondrous creature! Mount where science guides,
Go, measure earth, weigh air, and state the tides;
Instruct the planets in what orbs to run,
Correct old time, and regulate the sun;*

—Alexander Pope^①

1.1 计算技术延伸

1.1.1 从开尔文设想到现代计算技术出现

现代计算技术的出现源于解决复杂计算问题的需要,也蕴含着人们创造新型生产工具来突破人类极限并提高计算效能和精度的期望。19 世纪中期,面对潮汐预报这一高度复杂的计算问题,英国剑桥大学毕业的开尔文^②提出了自己的设想,并尝试“可以找到一种装置,其可以代替人来完成艰苦的计算工作”(原文: *The work hitherto done has been accomplished by laborious arithmetical calculation, but calculation of so methodical a kind that a machine ought to be found to do it*)。从开尔文的笔记中,人们相信他“采用科学的力量和方法来理解自然世界”的想法在一定程度上受到了 18 世纪英国诗人亚历山大·蒲柏上述诗句的启发。1876 年,开尔文设计出图 1.1 中所示的潮汐预报装置。这是一种由圆形刻度盘和齿轮等组件所构成的,专门用于潮汐时间和高度等复杂数据计算的模拟计算

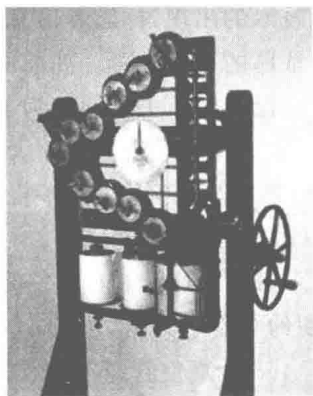


图 1.1 开尔文和他的潮汐预报装置

^① 亚历山大·蒲柏(Alexander Pope, 1688—1744),18 世纪英国最伟大的诗人,开篇的这个片段出自其诗作 *An Essay on Man*。

^② 开尔文男爵威廉·汤姆森(William Thomson, 1st Baron Kelvin, 1824—1907),英国数学家、物理学家、电学家、发明家,热力学第二定律奠基人之一,创立了热力学温度,用莱顿瓶的振荡实验推导出了电磁振荡频率的公式,设计了静电计、镜式电流计、双肩电桥等电磁学测量仪器。

装置。操作时,摇动手柄就可以完成模拟计算和输出,计算效率是一个专业计算人员的 10 倍以上。1882 年,开尔文在一场学术报告中首先使用了 Calculator 一词。

20 世纪以来,开尔文设想的另一个典型实现当属范内瓦·布什^①于 1931 年研制的微分分析仪(Differential Analyzer),如图 1.2 所示。这是第一台被用来解算微分方程的机械式计算机,被认为是电子计算机的先驱。这台装置与现代计算机的结构完全不同,将几百根平行的钢轴安放在金属框架上形成了这个机械式的计算装置,占地几十平方米。在应用过程中,操作人员只有用改锥和铁锤改变机械特性,才能实现参数的编程输入。进而,多个电动机通过齿轮使这些钢轴转动,钢轴的转动则模拟了数的运算。后来,布什进一步采用电子元件来替代某些机械零件,对计算装置进行优化,实现了洛克菲勒微分分析仪 2 号。

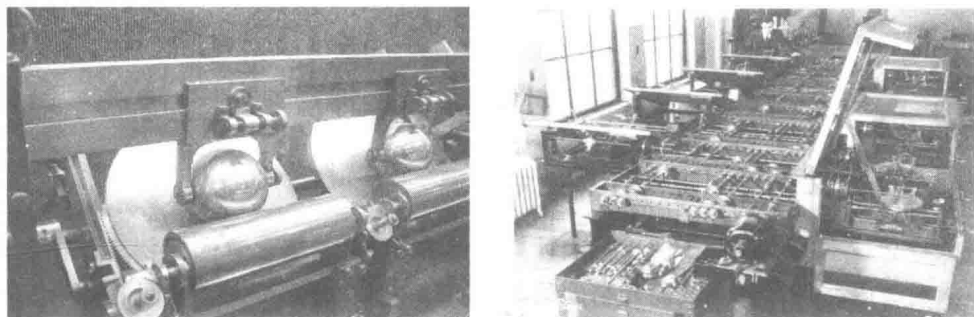


图 1.2 范内瓦·布什的微分分析仪

第二次世界大战中,美军曾广泛用微分分析仪来计算弹道射击表,大大提高了计算和作战效率。例如,对于一张有 3000 多个参数的射击表、60 秒射程的弹道,单人手工计算时需要大约 20 小时,而采用微分分析仪后仅需要 20 多分钟,且有更高的计算精度与可靠性。

1.1.2 计算机技术的发展

电子计算机技术的出现,标志着信息技术时代的到来。究其历史,电子计算机技术的诞生和发展都具有浓厚的军事色彩。如前所述,提高弹道和火力表参数的计算速度和精度,对于提高炮弹的打击命中率至关重要。为了进一步快速、精确地计算炮弹弹道,第二次世界大战期间,美国军方委派宾夕法尼亚州立大学的 Mauchly 博士和 24 岁的 Eckert 开始设计以电子管^②替代继电器的“电子化”计算装置。1946 年 2 月 14 日,全世界第一台电子计算机——ENIAC^③(埃尼阿克)研制成功。

如图 1.3 所示,ENIAC 计算机是一个昂贵的庞然大物,长 30.48 米、宽 1 米,具有 30 个操作台,占地面积约 170 平方米,重约 30 吨,耗电量 150 千瓦/小时,造价达到 48 万美元。其内部包含了 17 468 个真空管、7200 个水晶二极管、70 000 个电阻、10 000 个电容、1500 个继电器和 6000 多个开关。当然,最令人叹为观止的是 ENIAC 的计算性能,如每秒执行

① 范内瓦·布什(Vannevar Bush,1890—1974),美国杰出科学家与工程师,麻省理工学院教授,著有信息科学经典之作 *As We May Think*。

② 电子管:1904 年问世,一种最早期的电信号放大器件,利用电场对真空中的控制栅极注入电子调制信号,并在阳极获得对信号放大或反馈振荡后的不同参数信号。

③ ENIAC: Electronic Numerical Integrator and Calculator,电子数字积分器与计算器。

5000次加法或400次乘法,是当时继电器计算机的1000倍、手工计算的20万倍。ENIAC的应用,使弹道参数的计算时间降至30秒。这一阶段,电子计算机不但可以替代人的计算功能,还完完全全地超越了人的计算效率和精度。

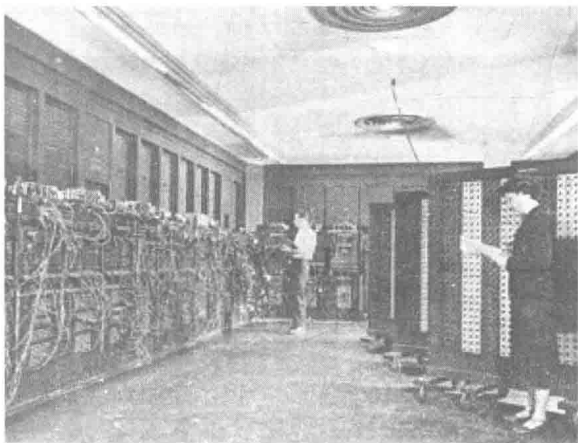


图 1.3 ENIAC 电子计算机

随着电子器件技术的不断发展和演化,基于半导体材料的晶体管^①技术、集成电路^②(Integrated Circuit, IC)技术出现,电子计算机硬件的计算性能不断提升、体积日益缩小。尤其是在大规模集成电路(LSI)技术诞生后,单一半导体芯片中可以集成更多的电子元件,计算机硬件的体积、功耗、成本显著降低,处理速度大幅提升。从此,开发专用的微型计算装置并将其从形式上完全“嵌入”到应用对象就成为可能。可以说,微型机的出现是嵌入式系统诞生的重要基础,是嵌入式系统技术发展历程中的一个重要里程碑。图 1.4 描述了相关技术的发展脉络。

(1) 1904年,电子管技术问世→1946年,美国宾夕法尼亚州立大学研制成功电子计算机——ENIAC。

(2) 1947年,基于半导体材料的晶体管问世→1954年美国贝尔实验室研制成功晶体管计算机——TRADIC(催迪克)。

(3) 1958年,集成电路技术诞生,可以把很大数量的微晶体管集成到一个小的半导体芯片上→1962年,IBM推出IC计算机IBM360。

(4) 半导体工艺发展迅速,20世纪70年代的大规模集成电路,20世纪80年代后期的超大规模集成电路(VLSI)、甚大规模集成电路(ULSI)→1981年IBM推出PC,各种微型、小型的计算装置出现。

与此同时,软件结构和编程语言也在很大程度上影响着计算技术的发展和應用。简单的软件结构无法有效发挥硬件的计算能力,而机器语言、汇编语言严重制约了复杂软件的设计与开发。工欲善其事,必先利其器。20世纪70年代初,贝尔实验室的Dennis Ritchie与

^① 晶体管:泛指一切以半导体材料为基础的单一元件。1947年12月,第一个晶体管问世。1956年,肖克利、巴丁和布拉顿,因发明晶体管同获诺贝尔物理学奖。

^② 集成电路:将多个电子元件集成到一个半导体器件。1958年,德州仪器的工程师Jack Kilby发明了集成电路,将三种电子元件结合到一片小的硅片上。



图 1.4 电子器件与计算机技术的发展关系

Kenneth Thompson^① 承担了为 PDP-11 计算机重新开发可移植 UNIX 作业系统的任务。为了提高系统软件的设计效率和质量,他们首先将目标转向了编程语言的设计。1972 年,基于 B 语言的 C 程序设计语言诞生。作为结构化的高级编程语言,C 语言可支持复杂高效、跨平台可移植软件的设计。毫无疑问,C 语言的出现是计算机发展史中的一次重大飞跃,其也成为至今仍然经典的嵌入式软件开发语言。现代操作系统的诞生和应用,是计算机史上的又一个重要里程碑,典型的如磁盘启动型操作系统 CP/M、磁盘操作系统 DOS、Mac OS 等。这些操作系统软件使计算机的资源管理、服务功能更为强大和丰富。在嵌入式系统领域,Ready System 公司于 1981 年面向控制应用推出了 VTRX32。该操作系统被公认为第一款商业嵌入式实时内核。20 世纪 90 年代至今,采用微内核、超微内核及模块化设计思想的嵌入式操作系统(Embedded OS, EOS)不断涌现,代表性的如 VxWorks、 μ CLinux、 μ C/OS、ThreadX、Windows Embedded 系列及国内自主研发的天脉嵌入式操作系统等。

1.1.3 通信与网络技术发展

随着电磁学领域相关原理的发现和证明,电子通信技术诞生并于 19 世纪 30 年代后期开始应用于远程数据传输等领域。这些伟大的基础研究中包括了法拉第^②的电磁感应(1831 年)、麦克斯韦^③的电磁场理论(1873 年)、奈奎斯特准则和采样定理(1928 年)以及香农定理(1948 年)等一批经典电磁学原理。基于这些发现和研究工作,1888 年赫兹^④在实验

^① Kenneth Lane Thompson(1943—)与 Dennis MacAlistair Ritchie(1941—2011)均为著名的美国计算机科学家和软件工程师,共同设计了 B 语言、C 语言,创建了 UNIX 和 Plan 9 操作系统,1983 年同获图灵奖。

^② 迈克尔·法拉第(Michael Faraday, 1791—1867),英国物理学家,在电磁学及电化学领域做出了巨大贡献,包括电磁感应、抗磁性、电解特性等。

^③ 詹姆斯·克拉克·麦克斯韦(James Clerk Maxwell, 1831—1879),苏格兰数学物理学家,提出了将电、磁、光统归为电磁场现象的麦克斯韦方程组等。

^④ 海因里希·赫兹(Heinrich Hertz, 1857—1894),德国物理学家,于 1887 年首先用实验证实了电磁波的存在,鉴于其对电磁学的伟大贡献,频率的国际制单位以他的名字命名。

中首次证实了电磁波的存在,意大利的马可尼^①和俄罗斯的波波夫^②于1895年同时发明了无线电等,都为有线/无线、模拟/数字数据通信系统的研制奠定了坚实的基础。图1.5列举了早期电磁学及通信技术研究的代表性节点。



图 1.5 通信理论与技术发展历程

循序渐进,水到渠成。有了远程数据传输的技术基础,研制更为复杂的数据通信网络就开始成为可能。1962年,MIT的利克莱德^③教授发表一篇学术论文,提出了“星云网络”概念,核心思想是将全球的计算机连结以便从任一节点获取资料。可以说,“星云网络”的理念是现代互联网及各种通信网络的思想之源。而在这一时期,全世界正处于美国和苏联两个超级大国所主导的冷战状态。为了提高安全防范能力,美国军方正在制定新的北约信息协作战略,“将北约的全部计算机系统连成一个网络,即使其中任何一个节点遭到苏联的核弹攻击,连成一体的信息系统也不会瘫痪,信息都可以在如图1.6所示的网络中绕道传送”,蕴含了具有高度鲁棒性的端到端数据传输理念。在美国高级研究计划署(ARPA)的支持下,为美国国防需要开发的ARPANET网络于1969年建成。ARPANET网络最初只有四个节点并采用分组交换技术,奠定了网络的基本形态,被公认为现代计算机网络的鼻祖。



图 1.6 ARPANET 的初衷

① 古列尔莫·马可尼(Guglielmo Marconi, 1874—1937),意大利工程师,从事无线电设备研制,1899年首次实现英法无线通信,1909年获诺贝尔物理学奖。

② 亚历山大·波波夫(Александр Попов, 1859—1906),原俄国物理学家和电气工程师,电磁波研究先驱。

③ 利克莱德(J. C. R. Licklider, 1915—1990),美国心理学家、计算机科学家,计算机网络理念先驱。