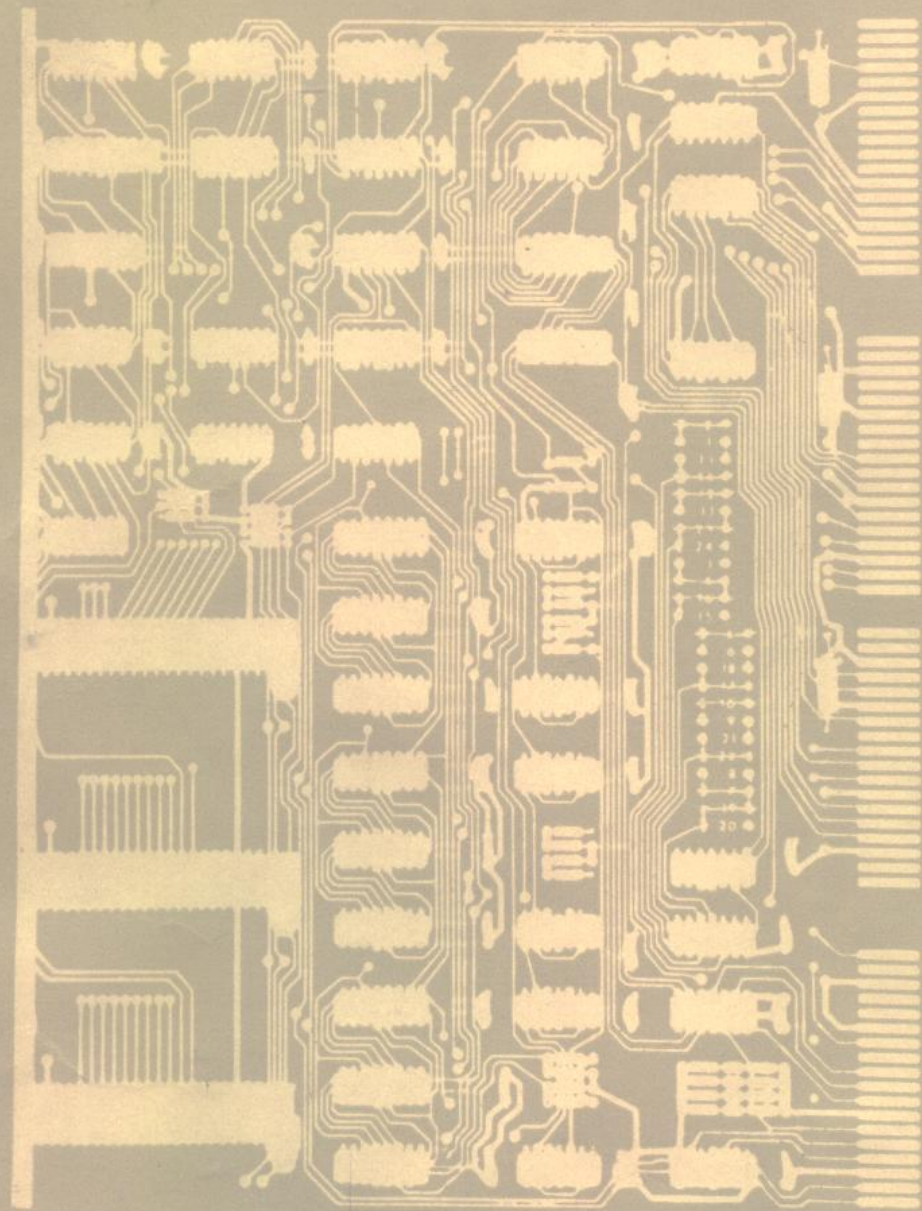


计算机工程



〔美〕 C.G. 贝尔 J.C. 马奇 J.E. 麦克纳马拉 著

科学出版社

13.81
291

计 算 机 工 程

〔美〕 C.G. 贝尔 J.C. 马奇 J.E. 麦克纳马拉 著
王祖永 石履超 苏中杰 译

(JS44 / 8)

JSPP/B

科 [第 1 版] 社

内 容 简 介

本书是根据美国数字设备公司(简称DEC)出版的《计算机工程》一书摘译的,这是一本全面论述计算机硬件系统设计的专著。DEC以其PDP-11和VAX-11等小型计算机系列闻名于世界,这些机器的设计思想对当代小型和微型计算机的发展影响很大。本书以“实例研究”的方法全面地讨论和总结了上述小型计算机系列硬件系统设计的经验。

本书对从事计算机设计和应用的工程技术人员及大专院校计算机专业的师生都是一本难得的参考书。

C. Gordon Bell J. Craig Mudge John E. McNamara
COMPUTER ENGINEERING
A DEC View of Hardware Systems Design
Digital Press, 1978

计 算 机 工 程

【美】 C.G. 贝尔 J.C. 马奇 J.E. 麦克纳马拉 著

王祖永 石履超 苏中杰 译

责任编辑 李淑兰 孙月湘

科学出版社出版

北京朝阳门内大街 137 号

中国科学院印刷厂印刷

新华书店北京发行所发行 各地新华书店经售

1984年7月第 一 版 开本: 787×1092 1/16
1984年7月第一次印刷 印张: 21 3/4
印数: 0001—16,800 字数: 494,000

统一书号: 15031·576

本社书号: 3582·15—8

定价: 3.35 元

译者前言

本书是美国数字设备公司（简称 DEC）出版的《计算机工程》的选译本。由 C. G. 贝尔（C. Gordon Bell）等主编的《计算机工程》原计划在 1982 年 DEC 成立二十五周年之际出版，实际提前于 1978 年出版。编者在序言中对提前出版的原因作了说明。

《计算机工程》一书在取材和编排上不同于一般计算机教科书，其特点之一是采用了“实例研究”的方法。所谓“实例研究”就是对已经设计和生产的实际机器进行分析研究。

计算机发展的历史表明，计算机设计技术的进步总是表现为局部的改进而不是全面创新。可以说，任何一台新计算机的设计都是在以前计算机的基础上改进的。其原因是迄今为止还没有系统地解决计算机设计问题的理论。在这种情况下，实例研究就成了学习计算机设计的一种好方法。国内外设计计算机的经验表明，如果不借鉴一台或几台已有计算机的结构和设计，很难设计出新的有生命力的计算机。因此可以说，每个计算机设计师都曾经自觉不自觉地进行过计算机的实例研究。同时，真正从事计算机设计的人盼望看到对一些有代表性的计算机和计算机系列的实例研究。遗憾的是一般著者很少采用这种方法。原因是这种方法可能使自己的著作遭到“片面性”或“局限性”的指责。《计算机工程》是少数采用这种方法的专著之一，因而受到了普遍的欢迎。

《计算机工程》一书分析了 DEC 创建以来生产的四个计算机系列：18 位小型计算机，12 位小型计算机 PDP-8，16 位小型计算机 PDP-11（包括 32 位的 VAX-11）和大型计算机 PDP-10。其中 PDP-11 系列是全书的重点，占总篇幅的 2/3。

PDP-11 是目前世界上占主导地位的小型计算机系列。我国发展计算机的方针是“中小为主、推广应用”，因此 PDP-11 系列的结构不可避免地会对我国发展的计算机有一定的影响。这本以 PDP-11 系列的实例研究为主要内容的计算机教科书必将对计算机的设计师、用户以及计算机科学或工程系的学生有所裨益。

我们以选译本的形式推荐给读者的是《计算机工程》的三章引论和有关 PDP-11 的全部章节。原书中未收入本书的章节有：原书第四章林肯 TX-2 的晶体管线路，第五章计算机的基础：数字模块，第六章 PDP-1 和其它 18 位计算机，第七章 PDP-8 和其它 12 位计算机，第八章 PDP-8 的结构层次，第十八章寄存器传输模块（RTM）的描述和使用及第二十一章 DEC 系统 10 的评价。这些章节约占原书篇幅的 1/4。

本书共分十五章和三个附录。其中第一章由苏中杰翻译，第二至第九章由石履超翻译，其余部分由王祖永翻译。全部译文由王祖永校订整理。杨天行、童颖同志曾对译稿进行了审阅，特此致谢。由于译者水平有限，书中难免有欠妥处，敬请读者批评指正。

译者

1983 年 2 月

序

世界上正在使用的计算机的数量在一代人的时间内已从几十台发展到几百万台，这种进步不是一个发明家或一个公司的功劳。在半导体物理和机械工程等各个不同的领域中，人们进行了长期的研究，从不同的角度进行了大量的工作，发现和发展了必要的技术把计算机技术推进到目前的状态。

计算机技术的发展已使得与之有关的人们得到莫大的鼓舞和利益，这种发展是多方面的。首先，在我们的有生之年已经发生了许多重大的进步，例如半导体晶体管的发明就是其中之一。其次是有机会联络许多优秀的同事，他们的明智、胆识、孜孜不倦的工作给予我们巨大的启发和鼓舞。最后，计算机获得了巨大的成功——它们能把人们从重复烦琐的脑力劳动中解放出来，它们能降低产品的成本，它们能从各个方面改善人们的生活。和计算机一起工作是有趣和令人兴奋的。

在本书中，各位著者根据他们过去二十多年的经验说明计算机技术怎样一步一步地发展到目前的水平，同时也描绘了他们已看到的未来的发展趋势。虽然一本书不可能容纳这些年来所有的成就和问题，但是他们完成了一项令人钦佩和值得称赞的工作。我们希望本书将鼓励未来的学生们带着他们的理想、雄心和胆略进入计算机工程这一领域。

数字设备公司总经理

K. H. 奥尔森

前 言

本书的对象有三类，第一类是直接从事计算机设计的人员——微型计算机、小型计算机或大型计算机的设计师。第二类是与计算机有关，对计算机设计有兴趣的人——用户、程序员、外围设备和存贮器的设计师。第三类是学习计算机工程及科学的学生。

计算机工程是不同的领域和学科的总和，而一般的教科书仅仅涉及它的一个方面——技术方面（例如乘法器的设计、同步线路的特性、串行方案与并行方案优缺点比较等等）。本书以实例研究的方法表明所有各种不同的因素（工艺、市场、制造）怎样构成计算机工程的约束条件和成功的机会。

计算机工程可以看作是一个多变量的数学问题，工程师们正是在一定的约束条件之内寻求该问题的最佳值。遗憾的是，对于一个变量的最佳值很少是对另一变量的最佳值。因此，计算机工程的主要部分就是寻求合理的折衷。有助于找到合理折衷的一种有效方法是给系统的各个变量加权，然后求出加权后的最佳值。各个变量的权随着应用的不同而有所变化。例如在一种情况下，速度被指定为最大的权；在另一种情况下，指令系统的兼容性可能最为重要；而在第三种情况下，可靠性可能居于首位。问题涉及的面很广，但对付它们的有效措施却不多。例如，成本是一个多维的变量，它包括制造、系统开发和现场维护成本。此外，在各个变量之间存在着大量相互依赖的关系。例如，指令系统，机器的组织，逻辑设计以及线路设计之间都存在相互依赖的关系。这些关系以及影响各变量权的大小的各种约束条件又是随着时间的推移而变化的。例如当不同的子系统采用不同的工艺时，成本函数随之变化，这种变化对相互关系必然有所影响。此外，象可维护性和兼容性等约束条件的重要程度也是逐年变化的。最后要指出的一点是，虽然我们对有些关系，例如加法器设计中的时间和空间的折衷，已了解得比较清楚，但对其它一些关系，特别对牵涉到市场因素的那些关系还不甚了了。

由于当前还没有一种理论从根本上解决这种多维设计问题，因此我们认为还没有一种方法可代替对已经设计出来并在市场上出售过的系统的实例进行广泛深入的研究。正是由于这个原因，本书采用了实例研究的方法。例如，我们在本书中利用了 DEC 公司创建后二十多年中制造的三十多种 DEC 计算机以及卡内基-梅隆（Carnegie-Mellon）大学在 PDP-11 的基础上研制的几种机器。卡内基-梅隆大学的机器探索了我们认为将构成未来几代计算机的基础的互连结构。

DEC 公司和卡内基-梅隆大学的合作不仅产生了几种值得研究的机器，而且为本书提供了一些书面材料。在大学工作的人有条件写文章也经常写文章；反之，很少有人要求或鼓励直接参与设计工作的工程师发表他们的工作成果。

这次 DEC 公司的著者所提供的材料大部分是历史性的。我们坚信这些历史性材料很值得花点功夫写一写、读一读、学一学。理由是计算机的设计原理和设计技术是缓慢变化的。事实上，目前正在设计的计算机所采用的原理都是好多年前就了解并用过了的。我们经常扪心自问：“我们是否正在耗尽设计的源头？”是的，工艺只是促进新的设

计, 并未促成新的原理。

但是学会设计显然是很重要的。虽然新的设计似乎是老的原理的重新应用, 但重新应用的过程改变并超过了以前的应用。学习设计的方法是结合一般原理、新的应用和新的工艺, 研究、模仿并改进以前的设计。例如, 微型计算机的发展从小型计算机中广泛地汲取了营养。当我们设计新的结构时, 应能避免上一次设计的缺陷。

在本书中我们有意把范围限制在 DEC 公司的计算机。其理由是显而易见的: 我们能用第一手材料进行讨论。如果采用其它公司的设计, 我们的数据可能会不够精确, 某些因素, 例如设计风格可能会被忽略。这个主要理由正是本书宗旨之关键。为了了解计算机设计的发展, 基础工艺变革所产生的影响以及不随时间变化的一般原理, 我们必须从头至尾研究一个系列, 从这个系列的诞生开始分析, 一直跟踪它, 看它怎样经历几代不同的工艺。DEC 公司有四个系列可以进行这样的分析。DEC 的计算机也为研究计算机工程的另一个方面——相互补充的 (有时是相互竞争的) 产品的共存提供了机会。几项设计工作必然要相互竞争资源 (设计力量、生产能力、软件支持、市场调查和销售支持)。一般来说, DEC 的计算机是设计成相互补充的, 并且避免了重复或冗余的产品。由此, 还可以看到在设计领域中起作用的另一组约束条件。

本书所讨论的是企图在商业上广泛应用的通用计算机。对特别专用的只制造了几台样机的计算机工程不作讨论。此外, DEC 的计算机并未囊括计算机体系结构和计算机工程的全部主要原理, 因此读者还应当研究其它的设计。例如, 读者不能从本书中学到描述符体系结构, 阵列处理机、表格处理机和通用仿真器。

曾经一度考虑把本书的出版时间推迟到 1982 年, 到那时, DEC 将庆祝本公司成立 25 周年。后来改变了主意, 因为五年以后收集初期计算机的资料将更加困难。更重要的是, DEC 的插件和计算机在 20 年里已经从早期的第二代发展到第四代。现在有几种 DEC 计算机的处理机是包括几千个晶体管的单片大规模集成电路。然而在 1957 年, 在单片锗上只能制造一个晶体管。在今后五年里, 计算机的设计、制造和分配将是完全不同的, 值得出版一本新书。

我们预期会有越来越多的人将与计算机工程有关而成为本书的读者, 同时我们还预计象我们目前所了解的计算机十年之内将会消失。随着单片处理机的应用, 计算机系统设计人员 (用户) 的数量已增加了几个数量级。

在大型计算机系统的领域中, 计算机的主顾和用户显然也是计算机设计师: 他们从现有的成套部件中选择一些部件, 把它们相互连接起来构成特定的结构。对大家来说, 主要是要有价格、性能和可靠性参数的模型以及表明它们怎样随着时间变化的模型。前几代计算机, 先是着眼于计算机的发明, 接下来着眼于了解计算机的价格/性能折衷。最近着眼于计算机的制造, 尤其是半导体器件的制造, 因为当前正是半导体器件推动着计算机的发展。在今后的五年中, 设计的重点将在应用: 普通的应用将更加有效, 同时将扩展新的应用, 而且将更注意使用期费用。对于计算机工程师来说, DEC 计算机的发展使他们能特别清楚地看到应用对设计的影响。为了提高可靠性、可用性和可维护性, 我们必须研究设计目标、约束条件和目标函数。为此, 迫切需要先弄清以前的设计问题。

对于那些不仅使用计算机, 而且参与部分计算机设计过程的程序员来说, 如果要知道决策的规则, 也必须了解本书的内容。在本书中除了讨论了软件对硬件设计的影响之

外,对软件本身讲得很少。在硬件的范畴之内,软件的作用越来越大,这是一个明显的动向。这一动向已促使计算机的体系结构发生变革。我们准备专门出几本书来讨论DEC计算机的软件工程,也许其中有一本讨论语言翻译程序,有一本讨论操作系统。我们也希望将来有几本书专门讨论大容量存贮器、终端设备和应用。

ISP和PMS这两种符号的表示方法是在《计算机结构》^[22]一书中首次提出的。我们在本书中使用这两种表示方法是因为它们已超出了符号表示法的范畴,而成为设计工作的工具。ISP开始时用于描述计算机的指令系统处理机——由程序(和程序员)所看到的机器,现在已用于机器描述、模拟、诊断验证、微程序设计、汇编程序自动生成以及计算机体系结构的比较。ISP的发展和改进主要归因于美国陆军和海军的计算机系列体系结构(CFA)计划的需要和马里奥·巴巴西(Mario Barbacci)的工作。ISP的最新版本ISPS正在DEC公司内用于实现处理器、模拟器等。现有的DEC计算机(PDP-8, PDP-10, PDP-11, VAX-11)和几个终端设备的ISPS语言描述已经完成。我们希望它们将被广泛使用,从而进一步促进机器描述语言的应用。广泛应用合适的语言将有助于解决目前存在的两个设计问题:第一,手工设计工具跟不上新工艺的推广速度;第二,计算机结构日益增加的复杂性的管理问题。本书附录I给出的PDP-8计算机的语言描述不同于一般的描述,它已经过机器诊断程序的验证。

PMS(处理机-存贮器-开关)符号表示法(见附录I)还未广泛用作辅助设计的正式方法。但是它已经广泛地用于描述计算机结构。努森(Knudsen)^[125]设计了一个识别PMS描述并实现几个性能分析函数的样机系统。目前由于ISPS正在扩展到包括计算机部件之间的互连,PMS和ISPS正在结合起来构成描述结构和特性的单一系统。在本书中我们用PMS描述功能块。但是所采用的符号不同于最初的条形符号,书中所有的PMS元件都用实线框了起来,形成一个方框。

本书一开始是三章引论。第一章提出了本书需要阐明的主题。这一章指出计算机的发展主要以半导体工艺和磁记录技术为基础。这些工艺技术决定了计算机的成本(从而也决定了价格)、性能、可靠性、体积、重量、电源以及构成机器物理特性的其它方面。本书的主要论题是,工艺的发展使我们能制造三种类型的计算机:

1. 具有不变的性能和降低成本的计算机;
2. 具有不变的成本和提高性能的计算机;
3. 完全新型的(大的或小的)结构,往往是用于研究的计算机,这种计算机可能创造全新的超越一般扩展可能性的计算机系列。

第二章回顾了存贮器工艺和逻辑线路工艺的发展。工程学既扎根于经济性,又有固有的实践性。组装(包括元件的互连)被编为第三章是出于实用的原因——组装和电源占小型计算机系统总生产成本的50%,而且这类成本还在上升。在第三章,为了进一步强调工程学的实践性,还包括了“大批量生产”一节;设计人员创造性的成果不仅必须行得通,而且还必须能在生产线上生产。

三章引论之后的内容分作五个部分:

- 一、起点;
- 二、小型计算机的起源;
- 三、PDP-11系列;

四、计算机积木块的发展;

五、PDP-10 系列。

每一部分有一段引言概括说明每种计算机发展过程中的主要因素: 计算机的设计人员与工艺和应用(市场)的相互作用。更重要的是我们力图指出一些经典的(长期有效的)设计原理。本书还包括了自从原始论文发表以后所收集到的数据。

第一部分讲插件。这是 DEC 公司创造初期的产品。第五章说明为了制造出计算机, 怎样研制插件和吸收半导体技术。

如第二部分所述, PDP-1 和其它 18 位计算机以及 PDP-8 是小型计算机的开始。虽然构成 18 位计算机系列的计算机有六种, 但只用了一章来进行讨论, 主要是因为缺乏现成的论文。这一章是专为本书撰写的。第七章叙述 12 位计算机的发展历史。第八章详细阐述 PDP-8 计算机的结构。

第三部分建立在 PDP-11 计算机的基础之上, 约占本书总篇幅的三分之二。DEC 在一定的时间内采用多种工艺和多种设计目标实现了 PDP-11 系列。PDP-11 系列是覆盖一个性能范围的一组计算机。由于成本和性能的要求, 必须在解决大量问题之后才能实现向下兼容 (LSI-11) 和向上兼容 (存储器更大的 PDP-11/70 和 VAX-11)。

第四部分专门叙述插件的发展。第十八章介绍了寄存器传输插件 (RTM, 也称 PDP-16), 这是一套可用于组成数字系统的通用插件。虽然这套插件在市场上并不成功, 但它们是现在广泛用来实现中型处理机和专用数字系统的位片方法的先驱。第二十章介绍了以 PDP-11 计算机为基础的一组插件, 这组插件又称为“计算机模块”。计算机模块是从初期的寄存器传输插件的研究中发展起来的, 它曾用于制造 C₃ 多微处理机系统。

第五部分讨论 PDP-10 计算机。在本书第二十一章那篇论文发表之前, 在工程学一级发表的关于 PDP-10 的论文很少。已经出版的文献着重于它的操作系统、语言、网络和应用。

《计算机工程》以《计算机结构》^[22]为楷模, 并企图为其补充题材。《计算机结构》讨论了指令系统体系结构的设计; 《计算机工程》则讨论实现指令系统的机器的设计。《计算机结构》论述了大量的 ISP 结构和 PMS 结构, 从早期的堆栈机和位串行机, 表格处理机和高级语言机, 直到巨型计算机。只要给读者提供伯克斯 (Burks)、戈德斯坦 (Goldstine) 和冯·纽曼 (Von Neumann) 等人的最基本的论文以及关于“旋风”计算机的论文, 读者可以利用《计算机结构》一书追溯计算机的发展历史。另一方面, 《计算机工程》选取的 ISP 范围较小 (仅四种), 但详细地分析了它们的实现方法。对于指令系统只有涉及到它们与实现方法的相互作用时才予以说明。本书偏重于从设计师的观点和历史的观点来讨论这四个计算机系列。书中特别强调影响 DEC 计算机系列发展的技术、经济、组织和环境因素。

本书主要是为设计师和学生编写的, 但是作为历史记录, 它对于曾经参加设计、制造、分配和维修的 DEC 工作人员来说也是有价值的。

我们推荐本书作为大学教材, 其依据是高等院校的教学经验, 对设计课程教材的要求以及我们参与课程编制的实践。电气与电子工程师协会计算机学会“计算机体系结构”工作小组的基本观点是: “为要正确评价计算机系统体系结构的发展, 必须分析完整的系统”。本书直接赞助这种观点。因此, 对于“计算机体系结构”课程 (例如, 电气与电

子工程师协会的 CO-3¹⁾ 课程), 本书是补充教材, 起着补充下列著者著作的作用: 巴克霍尔兹 (Buchholz)⁽⁸⁸⁾、贝尔 (Bell) 和纽厄尔 (Newell)⁽²²⁾、布莱伍 (Blaauw) 和布鲁克斯 (Brooks)。

对于“计算机组织”课程 (例如电气与电子工程师协会 CO-1¹⁾ 课程和 ACM 课程 I3 与 A2²⁾) 来说, 本书可作为辅助教材。至于采用 CO-2 (输入、输出和存贮器系统) 作为教学大纲的“计算机工程”课程, 本书是一本主要的教科书, 但是应利用其它制造商的计算机作为素材阐明不同的观点。

C. G. 贝尔

J. C. 马奇

J. E. 麦克纳马拉

1978年8月

1) “委员会报告: 计算机科学和工程课程” 电气与电子工程师协会标准课程分委员会, EH0119-8, 1977年1月。

2) “课程68” ACM通信11卷3期151~197, 1968年3月。

目 录

译者前言	i
序	ii
前言	iii
第一章 计算机系统的七种观点	
C. G. 贝尔, J. C. 马奇, J. E. 麦克纳马拉	1
第二章 逻辑电路工艺和存储器工艺的进展	
C. G. 贝尔, J. C. 马奇, J. E. 麦克纳马拉	25
第三章 组装和制造	
C. G. 贝尔, J. C. 马奇, J. E. 麦克纳马拉	57
第四章 PDP-11 文献综述	
C. G. 贝尔, J. C. 马奇, J. E. 麦克纳马拉	81
第五章 小型计算机的一种新型体系结构——DEC PDP-11	
C. G. 贝尔等	88
第六章 PDP-11 系列计算机高速缓冲存储器	
W. D. 斯特雷克	110
第七章 总线——计算机结构的骨架	
J. V. 利维	115
第八章 与小型计算机兼容的微型计算机系统: DEC LSI-11	
M. J. 赛朋	143
第九章 PDP-11/60 中档小型计算机的设计决定	
J. C. 马奇	156
第十章 实例研究: PDP-11 设计折衷对性能的影响	
E. A. 斯诺, D. P. 西维沃克	167
第十一章 用软件掩盖硬件差别的实例	
R. F. 布伦特	203
第十二章 对 PDP-11 的评价	
C. G. 贝尔, J. C. 马奇	218
第十三章 VAX-11/780: DEC PDP-11 系列的一种虚地址扩充	
W. D. 斯特雷克	246
第十四章 微型计算机设计实例研究: 用位片 LSI 处理机装配一台 PDP-11	
T. M. 马克威廉姆斯, S. H. 富勒, W. H. 舍伍德	266
第十五章 多微处理机: 综述和实例	
S. H. 富勒, J. K. 奥斯脱豪, L. 雷斯金, P. I. 鲁宾费尔德, P. S. 辛特休, R. J. 斯旺	279

附录 I	ISPS 初阶	
	M. 巴巴西	297
附录 II	PMS 表示法	
	J. C. 马奇	310
附录 III	性能	
	C. G. 贝尔, J. C. 马奇, J. E. 马克纳马拉	312
参考文献		320
汉英对照索引		329

第一章 计算机系统的七种观点

C. G. 贝尔 (C. Gordon Bell), J. C. 马奇 (J. Craig Mudge),
J. E. 麦克纳马拉 (John E. McNamara)

计算机决定于许多因素,其中包括体系结构、结构性能、工艺环境条件以及设计和制造计算机的环境条件中人的因素。在本书中,各位作者在 DEC 计算机的广泛范围内考虑了这些因素——它们的目标,体系结构,各种不同的实现方法和实际构造,偶尔也涉及设计计算机的人员。

计算机工程是各种活动的总和,其中包括与计算机设计和制造有关的分类学、理论、模型和试探法的应用。计算机工程同其它工程一样, Richard Hamming (那时在贝尔实验室)对它们提出了特别恰当的定义:工程师首先从科学中寻求解答和帮助,然后依靠数学获得模型和感性知识,最后转向他们的研究实践。

在几十年里,由于设计和制造了大量的计算机,计算机工程已经从主要依靠以研究实践为基础的一系列设计进入新的阶段:对某些零部件已十分了解,并使之建立在良好的模型和经验公式(例如工艺模型)的基础之上;对其它的零部件已充分了解,并采用了如电路最简化一类的有用的理论。

在这一章中提出了七种观点,作者发现它们对研究计算机及其结构和功能的形成过程是很有用的。七种观点的意图是互不关联的,每种观点从不同角度观察计算机。计算机科学家或数学家把计算机看作是一层层的解释器,工程师从结构的基础上看待计算机,特别注重结构的逻辑设计,用户所持的观点常常是一种市场观点。虽然上述每种人都倾向于计算机的某种观点,但每种人一般都了解其它观点的某些方面。本章的目的在于加深对其它观点的了解。增加用来描述研究目标的各种模型的数目,从而改进对这类观点的阐述。因此,“计算机系统的七种观点”一章构成了本书其余各章的背景。其余各章论述过去、现在和将来的计算机。

1.1 第一种观点:计算机系统的结构层次

在《计算机结构》^[22]一书中,针对计算机系统的描述、了解、分析、设计和使用提出了一系列概念性的结构层次,这个层次结构模型已经经受了工艺(例如把一台完整的计算机制造在一块硅片上)及体系结构上的重大变革(例如矢量和数组等数据类型的增加)的考验。

如图 1.1 所示,至少有五个系统描述级(层)可以用于描述一台计算机。每级以一种不同的语言为特征,这种语言用于表达与该级有关的元件、它们的组合方程式以及它们的行为规律。在每级内存在系统和子系统的完整的分级体系,但只要它们是用同一种语言描述的,它们就不构成独立的级。根据这个总的观点,人们可以从底部开始一级一级研究计算机系统。

图 1.1 中最低层是器件级。该级的元件是 p 型和 n 型半导体材料、介电材料以及用不同方法加工成各种形状的金属材料。元件的特性用半导体物理学和材料科学的语言来描述。

第二层是电路级。在电路级中，元件为电阻、电感、电容、电压源和非线性器件。系统的工作特性利用电压、电流和磁通量进行测量。这些都是与不同元件有关的连续变量。因此，在所有的时间内其工作特性是连续的，并且可写出描述变量行为的方程式（其中包括微分方程）。元件具有若干分立的端子，借此可以将它们与其它元件相连接。

在电路级上面的是开关线路或逻辑级。虽然数字技术的线路级非常类似于电气工程的其余部分，但逻辑级是一个分叉点，数字技术正是在该点从电气工程中分离出来。现在系统的行为可以通过只取称为 0 和 1（或 + 和 -，实和虚，高和低）的两个值的分立变量来加以描述。元件执行称为“与”、“或”、“与非”、“或非”和“非”的逻辑功能。系统构成的方法与电路级相同，把工作特性值不同的元件的端子连接起来构成系统。一个系统经过这样构成之后，可利用布尔代数定理从元件的工作特性和性能计算出系统的工作特性。

组合逻辑电路的输出在任何情况下都与输入直接有关，此外还有时序逻辑线路。时序逻辑线路有能力保持数值，因此它可以存贮信息。组合级分析解决的问题是在时间 t 产生一组输出作为同一时间 t 的若干输入的函数。虽然人们需要添加存储器元件，但时序开关电路的表示法基本上与组合开关电路的表示法相同。详细说明时序逻辑电路结构的方程必须是与时间有关的差分方程，而不是简单的布尔代数方程。布尔方程只能单纯描述组合逻辑电路。

在开关线路级上面的一层称为寄存器传输（RT）级。寄存器传输级的元件是寄存器和寄存器之间的功能传输器。当系统进行分立操作时出现功能传输，通过这种方法各寄存器的数值可根据某一法则加以组合，然后存入（传输到）另一寄存器中。组合的法则或定律几乎可以是任意的，从简单的不修改的传输（ $A \leftarrow B$ ）到逻辑组合（ $A \leftarrow B \wedge C$ ）或算术组合（ $A \leftarrow B + C$ ）。因此，如同时序电路用布尔方程描述，电路级用微分方程描述，开关线路级的工作特性用一组表达式（通常称为乘积）描述。表达式给出将进

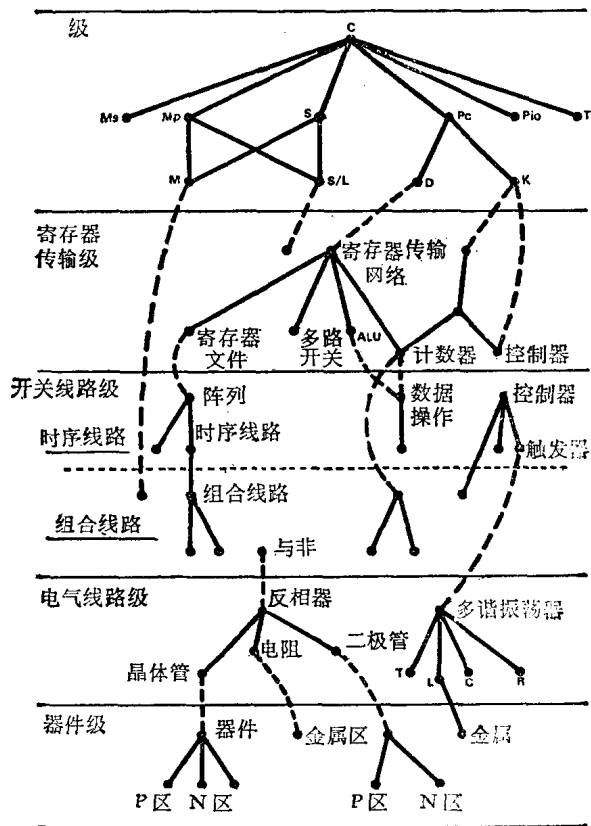


图 1.1 计算机层次的分级体系 (22)

行的这种传输条件。

图 1.1 中的第五层，即最后一层称为处理器-存贮器-开关 (PMS) 级。这一级给出计算机系统最集中的特性，它包括中央处理器、磁心存贮器、磁带、磁盘、输入/输出处理器、通信线路、打印机、磁带控制器、总线、电传打字机、显示器等。计算机系统被看作是一种处理信息的介质。这里的信息可用二进制数位（或十进制数位、字符、字等）进行度量。因此，这一级的元件以容量和流动速率作为它们的工作特性。

在 Bell 和 Newell 的文章^[22]中原来还有一层程序级，因为它的功能不是一层结构级，所以未在此讨论。

五种结构层次的每一级使用了许多表示法。两种特别的表示法是处理器-存贮器-开关 (PMS) 和指令系统处理器 (ISP) 表示法。Bell 和 Newell^[22, 第2*]给出了这些表示法的全部使用说明。本书所用的处理器-存贮器-开关中的有关问题将在附录 I 中说明。指令系统处理器的表示法已经发展成 ISPS 语言，这将在附录 I 中介绍。

1.2 第二种观点：解释器的层次

同结构观点相比较，这种观点是功能性的。按照这种观点，计算机系统由一层层解释器组成，其结构就象葱头的层次。

解释器是一种由指令驱动并依靠状态信息操作的处理系统。图 1.2 所示的基本解释周期在计算机语言级中是最熟悉的，但其它几级也存在这样的解释周期。

为了使解释器的层次形式化，人们可以利用图 1.3 中的图解表示一种处理系统。

由解释器处理的状态信息分为内部状态或外部状态。表 1.1 所示的典型班机预定系统的“葱头式”五级处理系统的层次结构清楚地表明了这一点。表 1.1 中列出了这些层次。

第 0 级系统是安排第 1 级微机器时序的逻辑。第 1 级系统是一种用实际硬件实现的微程序设计的处理器。这是逻辑设计师所看到的计算机。第 2 级系统是中央处理机 (CPU)。这是计算机语言程序设计员所看到的计算机。这里所表示的第 3 级系统是 FORTRAN 语言处理系统。第 4 级系统是一个班机座位预定系统。上述五级系统中的四级构成

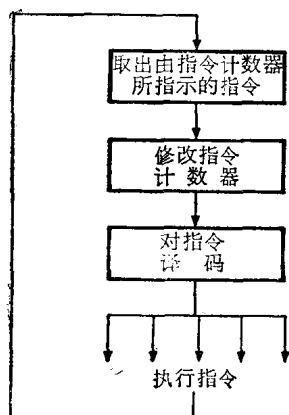


图 1.2 基本解释周期^[131]

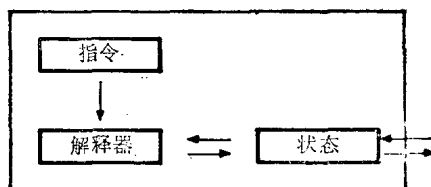


图 1.3 处理系统^[131]

了图 1.4 所表示的分级体系。在这里，每个系统是一个解释器，每个解释器为了完成高级解释器的一步操作需要顺序通过好几步。最高一级系统——班机座位预定系统是处理从系统外部接受的信息的解释器。它测试和修改状态，并产生送回系统外部的电文，

表1.1 班机预定系统用的五级解释器

第4级	指令:	预定座位的电文.
	解释器:	班机座位预定系统.
	内部状态:	在此之前已预定的座位数; 乘客表在磁盘文件上的位置; 与系统连接的线路的数量.
	外部状态:	在给定的飞机上预定的座位数量; 给定飞机的航线名.
第3级	指令:	FORTRAN解释程序语句代码.
	解释器:	FORTRAN解释程序执行系统.
	内部状态:	存储器管理参数; 用户名称; 主存储器容量; 磁盘文件位置; 中断允许位; 表达式计算栈; 数组维数.
	外部状态:	子程序名称; 数组数据值; 语句数; 程序容量; 表达式的值; DO 循环变量值; 宽行打印机上印制的字符.
第2级	指令:	机器语言指令.
	解释器:	处理器.
	内部状态:	程序寄存器; 条件码; 程序计数器.
	外部状态:	主存储器中的数据; 磁盘控制器寄存器.
第1级	指令:	微指令.
	解释器:	微程序机器.
	内部状态:	指令寄存器; 错误状态保存触发器; 微子程序链接堆栈.
	外部状态:	程序寄存器; 条件码; 程序计数器.
第0级	指令:	硬接线组合网络.
	解释器:	控制微程序机器的时序机.
	内部状态:	控制微程序机器时序的时钟、计数器等.
	外部状态:	微机器控制台.

因而完成了最外层解释器的一步操作。

实际上,几乎没有几种系统是纯解释器层次结构,尽管层次是存在的。由于硬件和软件的原因,已经出现了与模型的不一致。在硬件不一致情况下,第1级所表示的微程序机器往往并不存在。第2级中央处理机是直接使用第0级的时序控制器实现的。跳过第1级的这种实际应用,最初是由于缺乏合适的只读存储器,但现在一般只限于极高速

计算机，例如 Cray 1 和 Amdahl V6。因为从控制存储器取出和执行微指令的周期时间，对这些极高速计算机来说太慢了。

使真实系统与上述纯解释器模型不一致的软件原因有二：（1）高级语言通常是由编译程序或者是由解释程序实现的；（2）当更理想的原语存在更深的层次时，某些中间层次都可以加以旁路。图 1.5 用图解说明了这种旁路的过程。FORTRAN 纯解释器的实现方法可能是让图中指定的所有 FORTRAN C 的操作使用一个目标程序运行系统（OTS）。目标程序运行系统可能要求一个操作系统解释它的某些操作，操作系统又可能由指令系统解释器（ISP 解释器）进行解释。然而，图中的 A 操作可能由指令系统解释器直接进行解释。

总之，层次的数目正好是另一种折衷的结果。性能上的考虑导致层次的取消，复杂性又导致层次的增加。由于提出了纯解释器模型，使人们现在能够回到葱头式分层模型，以便更好地了解不同的层次是如何联系起来的。

宏机器的硬件可以看作是一级基本的解释器。它通常利用操作系统向上扩展。可能有若干种操作系统层次，因此计算机可以按有序的方式构成。核心机器可以管理和诊断硬件元件（磁盘、终端）和提供同步操作，这样，控制硬件的多个进程能并行操作。其次，增加文件系统和基本实用服务程序之类的更复杂的操作，在此基础之上又附加资源管理和统计之类的政策因素。从操作系统来观察，人们看到的机器与基本指令系统体系

结构提供的完全不同。事实上，在最终的机器上几乎看不到一般由符号汇编程序给出的体系结构。最终的机器包括基本计算机，但具有更大的输入/输出能力以及由许多程序（或任务）所共享的功能。

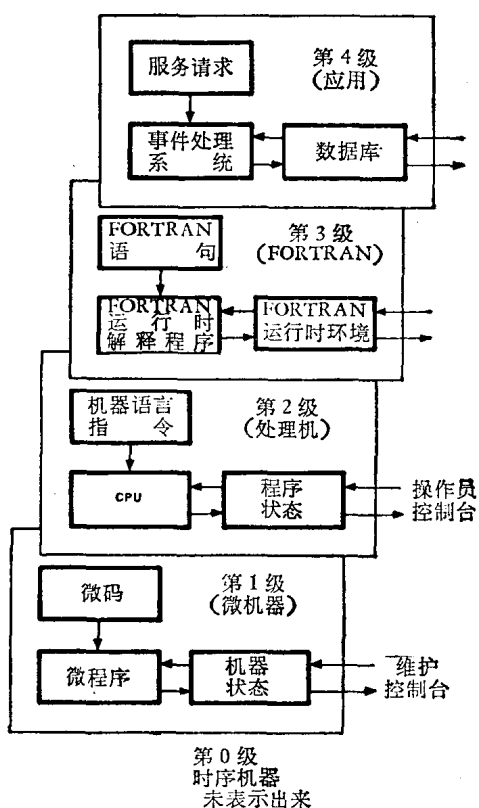


图 1.4 解释器的分级体系 (131)

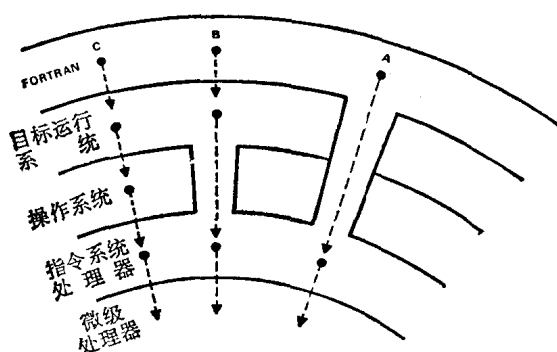


图 1.5 具有旁路“导管”的解释器层次。
FORTRAN C 操作由目标程序运行系统进行解释，而目标程序运行系统的操作又借助操作系统加以解释，操作系统又由指令系统解释器予以解释。FORTRAN A 具有直接通向指令系统解释的导管

操作系统设计师认为，为了实现更高一层的解释器——标准语言，所有这些设置都