

信息与电子学科百本精品教材工程

新编计算机类本科规划教材

计算机系统结构教程

陈建铎 编著 李学干 主审

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



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

TP303
118

新编计算机类本科规划教材

计算机系统结构教程

陈建铎 编著

李学干 主审

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

“计算机系统结构”是计算机科学与技术专业的一门主干课程。其目的在于向学生讲授计算机的各种系统结构,并使学生了解和掌握设计新型计算机的思想、方法和步骤。

全书共分为9章。第1章讲述计算机系统的层次结构、类型、性能分析方法及发展过程;第2章讲述数据的类型与表示、指令系统的设计思想,以及CISC与RISC计算机的比较;第3章讲述存储器体系结构的基本原理、虚拟存储器、Cache及其管理模式;第4章讲述标量流水线工作原理、流水线的实现与控制,以及流水线中的并行处理;第5章讲述向量处理机的体系结构与类型、向量处理方式、快速向量处理技术及其性能分析;第6章讲述互连网络的作用、设计准则、类型、结构、互连函数,以及消息传递方式;第7章讲述并行处理机的基本概念、开发策略,SIMD阵列机的基本结构,以及并行存储器的无冲突访问等;第8章讲述多处理机的基本结构、类型,多处理机中的并行处理技术、操作系统、算法、程序设计语言、Cache的一致性,以及并行处理技术的新发展;第9章讲述脉动阵列机、数据流计算机、归约机,以及人工智能计算机的基本原理、实现方式及体系结构。

本书可以作为大学计算机科学与技术专业的本科教材,也可以供电类研究生使用。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

计算机系统结构教程/陈建铎编著. —北京:电子工业出版社,2006.2

新编计算机类本科规划教材

ISBN 7-121-02202-8

I. 计… II. 陈… III. 计算机体系结构—高等学校—教材 IV. TP303

中国版本图书馆CIP数据核字(2006)第001000号

责任编辑:冉哲 特约编辑:李云霞

印刷:北京牛山世兴印刷厂

出版发行:电子工业出版社

北京市海淀区万寿路173信箱 邮编 100036

经销:各地新华书店

开本:787×1092 1/16 印张:16.25 字数:410千字

印次:2006年2月第1次印刷

印数:5000册 定价:21.80元

凡购买电子工业出版社的图书,如有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系。联系电话:(010)68279077。质量投诉请发邮件至 zltz@phei.com.cn,盗版侵权举报请发邮件至 dbqq@phei.com.cn。

前 言

“计算机系统结构”是计算机专业的学生学完“数字逻辑与电路”、“计算机组成原理”、“操作系统”、“汇编语言程序设计”、“编译原理”和一门“高级语言程序设计”课之后开设的一门课程。过去属于专业课，现在逐步被认为是专业基础课。不管是专业课还是专业基础课，它都是计算机科学与技术专业的一门主干课程。它不仅涉及硬件，还涉及软件；不仅涉及基础理论，还涉及新型计算机系统结构的研究与发展。因此，通过该门课程的学习，可把学生的水平提升到一个更高的层面。本书是针对计算机科学与技术专业本科生的教学要求而编写的，也可以供电类非计算机专业的研究生使用。

全书共分为9章，其中第1章主要讲述计算机系统的层次结构、类型、性能分析方法及发展过程；第2章主要讲述数据的类型与表示、指令系统的设计思想，以及CISC与RISC计算机的性能比较；第3章主要讲述存储器体系结构的基本原理、虚拟存储器、高速缓冲存储器(Cache)；以及管理模式；第4章主要讲述标量流水线工作原理、流水线中的主要障碍与处理方法、流水线的实现与控制，以及流水线中的并行处理；第5章主要讲述向量处理机的体系结构与类型、向量处理方式、快速向量处理技术、向量处理机实例及其性能分析；第6章主要讲述互连网络的作用、设计准则、类型、结构、互连函数，以及互连网络中的消息传递方式；第7章主要讲述并行处理机的基本概念、开发策略，SIMD阵列机的基本结构、并行算法、并行存储器的无冲突访问、典型SIMD阵列机举例与应用；第8章主要讲述多处理机的基本结构、类型，多处理机中的并行处理技术、操作系统、算法、程序设计语言、Cache的一致性，以及并行处理技术的发展趋势与实例；第9章主要讲述脉动阵列机、数据流计算机、归约机，以及人工智能计算机的基本原理、实现方式及体系结构。参考学时为64学时。

在编写时，作者力求知识结构完备，层次分明，概念清楚，内容简练，既能讲清基本理论，又能反映最新技术。通过增加应用实例，试图把理论教学与实际应用结合起来，以加深学生对理论知识的理解。在文字方面，尽量做到语言简练，通俗易懂。

由于作者水平有限，难免存在一些谬误之处，诚请同行教师与读者批评指正。

电子邮件：ch-jd@163.com

作 者

2006年1月

目 录

第 1 章 计算机系统结构的基本概念	1
1.1 概述	1
1.2 计算机系统结构的定义与内涵	3
1.2.1 计算机系统的层次结构	3
1.2.2 计算机系统结构的定义与内涵	4
1.2.3 计算机系统的组成与实现	5
1.3 计算机系统结构的类型	6
1.3.1 按“流”分类	7
1.3.2 按“并行级”和“流水线”分类	8
1.3.3 冯氏分类法	9
1.4 计算机系统性能分析	10
1.4.1 定量分析法	10
1.4.2 CPU 性能分析	11
1.4.3 MIPS 与 MFLOPS 指标	12
1.4.4 性能测试结果的统计与比较	13
1.5 计算机系统结构的发展过程	15
1.5.1 计算机系统结构的演变过程	15
1.5.2 软件对计算机系统结构发展的影响	17
1.5.3 应用需求对计算机系统结构发展的影响	21
1.5.4 器件对计算机系统结构发展的影响	22
1.5.5 算法对计算机系统结构发展的影响	23
1.6 计算机系统的成本、性能与价格	24
1.6.1 计算机系统的成本与价格	24
1.6.2 计算机系统的性能与价格	25
习题与思考题	26
第 2 章 数据的表示与指令系统设计	28
2.1 数据的类型与表示	28
2.1.1 数据类型	28
2.1.2 数据表示	29
2.1.3 定点数与浮点数	30
2.1.4 向量数据表示	33
2.1.5 自定义数据表示	34
2.2 指令系统设计	36
2.2.1 指令系统的发展及其风格	36
2.2.2 指令系统的设计要求与分类	38

2.2.3	地址分配与寻址方式	40
2.2.4	按内容寻址方式	43
2.2.5	指令格式与优化设计	45
2.2.6	指令系统结构举例	48
2.3	CISC 计算机和 RISC 计算机	50
2.3.1	CISC 计算机的特点与风格	50
2.3.2	RISC 计算机的特点与风格	52
2.3.3	CISC 计算机和 RISC 计算机性能比较	53
2.3.4	RISC 计算机中的新技术	55
2.3.5	RISC 计算机指令系统实例分析	58
	习题与思考题	59
第 3 章	存储器体系结构	61
3.1	概述	61
3.1.1	存储器的层次结构	61
3.1.2	基本概念与性能参数	62
3.1.3	多体并行存储器	63
3.1.4	磁盘阵列	65
3.2	高速缓冲存储器 Cache	69
3.2.1	Cache 的基本概念	69
3.2.2	Cache 的组成与工作原理	69
3.2.3	地址映像与变换	70
3.2.4	Cache 置换策略与主存更新方法	74
3.2.5	Cache 性能分析	78
3.3	虚拟存储器	80
3.3.1	虚拟存储器的基本概念	80
3.3.2	虚拟存储器的管理方式	82
3.3.3	虚拟地址快速变换法	86
3.3.4	置换策略	87
3.3.5	虚拟存储器管理举例	88
3.4	存储器保护措施	91
	习题与思考题	93
第 4 章	标量流水线技术	95
4.1	概述	95
4.1.1	控制流及其改变	95
4.1.2	程序执行过程中的重叠操作与先行控制	96
4.2	标量流水线工作原理	98
4.2.1	标量流水线工作原理	98
4.2.2	标量流水线分类	99
4.2.3	流水线性能分析	100
4.2.4	流水线中的主要障碍	103

4.2.5	流水线的实现与控制	109
4.2.6	流水线的动态调度	111
4.3	指令级流水线	115
4.3.1	指令级流水线概述	115
4.3.2	超级标量流水线	115
4.3.3	超长指令字	117
4.3.4	展开循环体后调度	119
4.3.5	软件流水法	122
4.3.6	超级流水机举例	123
	习题与思考题	123
第 5 章	向量流水与向量处理机	126
5.1	向量流水的概念与工作原理	126
5.1.1	向量流水的概念与特点	126
5.1.2	向量处理机的基本组成	127
5.1.3	向量启动时间与结果流出时间	128
5.1.4	向量操作长度控制与向量访问步长	129
5.2	向量处理与增强向量处理性能的方法	130
5.2.1	向量处理方法	130
5.2.2	增强向量处理性能的方法	132
5.3	向量处理性能的评价参数与评价方法	136
5.3.1	机器向量长度与向量流水处理时间	136
5.3.2	向量流水操作中处理时间及速率与向量长度的关系	138
5.3.3	向量流水处理中与向量长度有关的参数	138
5.4	向量化编译技术	139
5.5	向量处理机举例	140
5.5.1	多向量多处理机 CRAY Y-MP 816	140
5.5.2	C-90	141
	习题与思考题	142
第 6 章	互连网络	144
6.1	互连网络的基本概念	144
6.1.1	互连网络的功能与特征	144
6.1.2	互连函数	147
6.2	静态互连网络	150
6.2.1	静态互连网络结构	150
6.2.2	静态互连网络特性	153
6.3	动态互连网络	154
6.3.1	总线互连方式	154
6.3.2	交叉开关互连方式	155
6.3.3	多级网络互连方式	156
6.4	消息传送与控制	162

6.4.1	消息寻径方式	162
6.4.2	虚拟通道与死锁	164
6.4.3	流控制策略	165
	习题与思考题	167
第7章	并行处理技术与 SIMD 阵列机	169
7.1	并行处理技术	169
7.1.1	并行处理的基本概念	169
7.1.2	并行性的开发途径	170
7.2	SIMD 阵列机	171
7.2.1	SIMD 阵列机的基本结构	171
7.2.2	阵列机并行算法	173
7.2.3	SIMD 阵列机的特点	176
7.2.4	并行存储器无冲突访问	177
7.3	典型 SIMD 阵列机举例	178
7.3.1	ILLIAC-IV阵列机	178
7.3.2	BSP 阵列机	181
	习题与思考题	181
第8章	多处理机	183
8.1	多处理机的基本结构与类型	183
8.1.1	紧耦合多处理机	183
8.1.2	松耦合多处理机	185
8.1.3	多处理机中 Cache 的一致性	187
8.2	并行多处理机技术	188
8.2.1	程序并行性分析	188
8.2.2	并行程序设计语言	190
8.2.3	多处理机操作系统	192
8.2.4	多处理机调度策略	193
8.3	并行多处理机的发展趋势	195
8.3.1	大规模并行处理机	195
8.3.2	MPP 机举例	196
8.3.3	共享存储器型多处理机	200
8.3.4	并行向量多处理机	205
8.3.5	机群	206
	习题与思考题	211
第9章	计算机体系结构的发展	212
9.1	脉动阵列机	212
9.1.1	脉动阵列机的组成原理	212
9.1.2	面向特定算法的脉动阵列机的结构形式	214
9.1.3	通用脉动阵列机的结构	216
9.2	数据流计算机	218

9.2.1	数据驱动原理	218
9.2.2	数据流计算机模型	219
9.2.3	数据流程图与数据流语言	221
9.2.4	数据流计算机举例	224
9.3	归约机	228
9.3.1	归约方式	228
9.3.2	函数式程序设计语言	229
9.3.3	面向函数式程序设计语言的归约机	231
9.4	人工智能计算机的研究与发展	233
9.4.1	人工智能计算特征	234
9.4.2	并行推理机	235
9.4.3	数据库机与知识库机	236
9.4.4	模糊控制器与人工神经网络	241
	习题与思考题	247
	参考文献	249

第 1 章 计算机系统结构的基本概念



内容提要

本章首先简要介绍计算机系统结构的含义、层次结构与类型，然后讲述计算机系统结构性能分析的方法，计算机系统结构的发展，以及计算机系统的成本、性能与价格。重点是计算机系统的层次结构、按“流”分类、定量分析、Amdahl 定律、影响计算机系统结构的要素。难点是计算机系统的层次结构、按“流”分类、定量分析及 Amdahl 定律。

1.1 概述

自从 20 世纪 40 年代世界上第一台电子数字计算机诞生以来，在短短 60 年的发展历程中，其结构、规模、类型、性能、应用领域都发生了翻天覆地的变化。这种变化是其他任何一种科学技术所无法比拟的。回顾数十年的发展历程，在促使计算机飞速发展的诸多因素中，起决定因素的主要是器件和系统结构的发展与变化。根据计算机所使用的器件与系统结构的构成方式，人们把计算机的发展过程归结为四个历史时期，也称为四代。现在，正在向第五代发展。各历史时期跨越的时间范围及其典型机器的主要特征见表 1.1。

表 1.1 计算机的发展历史

年代	特征	器件	结构	软件	典型机器
第一代 1945—1954 年		电子管和继电器	存储程序、计算机程序控制 I/O	机器语言与汇编语言	ENIAC、IBM 701
第二代 1955—1964 年		晶体管、磁芯、印制电路	浮点数据表示、多寻址方式、中断技术、I/O 处理机	高级语言与编译、批处理技术与监控系统	Univac LARC CDC 1604 IBM 7030
第三代 1965—1974 年		SSI、MSI、多层印制电路	微程序控制器、流水线技术、Cache、先行处理、系列机	多道程序处理分时操作系统	IBM 360/370 CDC 6600/7600 DEC PDP-8
第四代 1975—1990 年		LSI、VLSI、半导体存储器	向量处理、并行计算、分布式存储器	并行与分布式处理	Cray-1 IBM 3090 DEC VAX 9000 Convax-1
第五代 1991 至现在		高性能微处理器、高密度器件	超标量、超级流水线 SMP、MP、MPP	大规模可扩展并行与分布式处理	SGI Cray T3E IBM SP2 DEC Alpha Server 8400

从表 1.1 可以看出，在器件方面所采用的依次是电子管、晶体管、小规模集成电路、中规模集成电路、大规模集成电路，以至超大规模集成电路。其功能在不断地增强，速度和可

靠性在不断地提高，成本在不断地下降。由此可以看出，在计算机发展的过程中，器件是其物质基础，是其更新换代的重要标志。另外还可以看出，在器件更新的同时，计算机的系统结构也在不断地改进。许多新的概念不断产生，新的组成方式不断得以实现。例如，采用大量的通用寄存器，以减少访问存储器的次数；使用浮点数据表示，增设浮点运算器，以提高浮点运算能力；在存储器管理方面，采用变址寄存器、间接寻址方式、虚拟存储器管理、Cache 等技术，以提高存储器的性能；设立中断控制系统，采用输入/输出通道技术，以提高数据输入/输出的能力；此外，还有微程序控制器、并行处理技术、多处理机技术及系列化产品等概念。正是器件的更新与系统结构改进的结合，促使了计算机的不断发展和更新换代。

为了验证在计算机的发展过程中各种因素所起的作用，计算机科学家恩斯洛 (P.H.Enslow) 曾对 1965—1975 年间计算机发展的情况进行了研究，比较了器件延迟时间与指令执行时间的变化。结果表明，这 10 年间器件延迟时间降至原来的十分之一，而指令的平均执行时间降至原来的百分之一。也就是说，计算机系统的性能提高了 100 倍，其中 10 倍归功于器件延迟时间的降低，而另外 10 倍则应归功于系统结构的变化。因此可以说，系统结构的变化也是计算机发展和更新换代的重要标志。

在计算机进入第四代以后，有人认为计算机的档次不应再以器件为标准来进行衡量，因此提出，第五代计算机应当是智能化计算机。但是，智能化计算机既涉及了硬件体系结构，又涉及了智能化软件的发展。虽然经过了多年的努力，一时还是难以实现智能化计算机的目标。而在 20 世纪 90 年代以后，计算机的硬件体系结构就有了新的突破性发展，于是人们将其归结为第五代发展时期。其中重要的标志是采用了 VLSI 工艺，构成更加完善的高密度、高速度的处理机和存储器芯片，实现了大规模并行处理技术和规模可扩展性。典型机种除了表 1.1 中所列举的几种之外，还有 Fujitsu 的 VPP500、Cray Research 的 MPP、Thinking Machines 公司的 CM-5、Intel 超级计算机系统 Paragon、SGI 的 Origin2000 及 SUN 10000 服务器等。

目前，就计算机的规模而言，有微型机、小型机、中型机、大型机和巨型机；就计算机的体系结构而言，有标量流水机、向量流水机、阵列机、并行多处理机、数据流机和归约机等；另外，人工智能计算机仍然在发展之中。

在计算机系统结构的发展中，并行处理技术起到了重要的推动作用。早在 1953 年，IBM 701 机率先实现了位并行运算。在同一时期，其他机器也实现了 CPU 与 I/O 的并行工作，其结果是使计算机的性能有了明显的提高。到了 20 世纪 60 年代，并行机进一步发展，出现了单 CPU 流水线。例如，1964 年完成的 CDC6600 设有加法、移位、乘法、除法和布尔运算等 10 个独立操作的功能部件，可分时复用，并行工作；1967 年完成的 IBM360/91 机实现了取指令、译码和操作数地址计算的流水线重叠工作。在 1970—1980 年间，大规模集成电路快速发展，导致各种并行处理机，例如，向量机、阵列机等，大量涌现。其中，Cray-1 就是比较成功的向量流水处理机，其浮点运算达到 130MFLOPS。

在进入 20 世纪 80 年代以后，软件技术的发展，对计算机系统结构的发展也起到了一定的促进作用。一方面是对汇编语言的要求有所降低，放宽了对目标代码兼容性的苛刻要求；另一方面，与厂商无关、类似于 UNIX 操作系统那样的标准化的建立，使新的系统结构的设计成本与风险有所降低，这就促使许多厂商相继推出了许多新型计算机，例如小巨型机 (minisupercomputer)、高性能的微处理机 (high-performance microprocessor)、工程工作站 (engineering workstation)，以及各种多处理机 (multiprocessor) 等。

在微型计算机的发展过程中，随着芯片集成度的提高，一方面使微型机的性能向工程工

作站、小型机靠拢；另一方面，各种高性能的微控制器、嵌入式芯片大量涌现出来，广泛地用于自动控制、电信和智能化仪器仪表中。

进入 20 世纪 80 年代以后，出现了另一个突破性的进展，这就是精简指令系统计算机（RISC, Reduced Instruction Set Computer）。它是对传统的复杂指令系统计算机（CISC, Complex Instruction Set Computer）的一种挑战，使计算机的系统结构再一次发生了质的改变。

1.2 计算机系统结构的定义与内涵

1.2.1 计算机系统的层次结构

早期的计算机是使用机器语言或汇编语言进行程序设计的。后来，随着软件技术的发展，人们开始使用不同的高级语言编程，从不同的层面使用计算机。于是，产生了计算机系统的多层次结构，如图 1.1 所示。

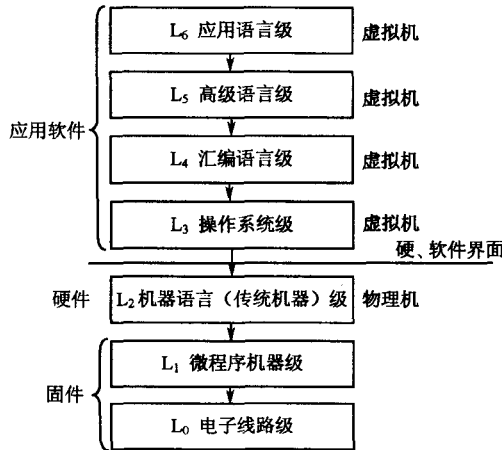


图 1.1 计算机系统的层次结构

图 1.1 中每一层对应一种“机器”，为该层的操作者所使用。操作者通过该层的语言与“机器”对话或者交互信息，而不必关心其内层的结构和工作过程。这体现的是一种称之为透明性的特性。

L_0 和 L_1 是计算机的低层硬件，主要是指中央控制器。其中 L_1 是微程序控制器，也称为固件。在这一级，程序员面对的是微指令集，通过微指令设计的微程序来实现机器语言指令的功能。 L_0 由电子线路组成，用来实现微指令的功能。

在早期的计算机中没有微程序控制器，在现在主流的 RISC 机中也不再设置微程序控制器，每一条机器语言指令直接由电子线路译码执行。

L_2 是传统的机器语言级机器，也称为物理机。在这一级，程序员面对的是机器语言指令系统，通过对机器语言指令的编程，来实现外部的功能要求。向下进入 L_1 ，每一条机器语言指令对应于一段微程序，通过译码识别，执行其微程序，实现该指令的功能。

L_3 是操作系统虚拟机。在这一级，多数指令是传统的机器语言指令，如算术/逻辑运算指令、移位指令等。另外，操作系统也提供了自己的命令集，如打开/关闭文件、读/写文件等。在这一级，程序员主要使用操作系统的命令对文件进行操作，其功能由软件来实现，即由操

作系统进行解释，因此称为操作系统虚拟机。

L₄ 是汇编语言虚拟机。在这一级，程序员面对的是汇编语言指令系统，通过对汇编语言指令的编程，来实现外部的功能要求。用汇编语言编写的程序首先需要翻译成 L₃ 和 L₂ 语言，然后再由相应的机器执行。这种完成翻译工作的程序称为汇编程序。

L₅ 是高级语言虚拟机。在这一级，程序员可使用各种高级语言进行程序设计，然后再由编译或解释程序将其翻译成 L₄ 或 L₃ 上的语言，然后由下一级机器执行。

L₆ 是应用语言虚拟机。这一级是为某些专门的应用而设计的，使用的是面向问题或对象的应用语言，可以设计出用于不同领域的虚拟机。应用语言编写的程序一般由应用程序包翻到 L₅ 上，然后再一级一级地翻译或解释执行。

1.2.2 计算机系统结构的定义与内涵

计算机系统结构 (computer architecture) 这一概念，是 1962 年 Buchholz 在描述 IBM 7030 的扩展性 (stretch) 时首先提出来的，主要是指计算机的外貌，也称为外特性。1964 年，IBM 370 的主设计师 Amdahl 给出了计算机系统结构的定义：计算机系统结构就是程序员所看到的计算机的基本属性，即概念性结构与功能特性。

对于不同层次上的程序员来说，由于使用的程序设计语言不同，可能看到的概念性结构与功能特性也会有所不同。例如，使用机器语言或汇编语言分别在 PDP-11 和 80x86 机上进行程序设计，就会发现两种机器的基本属性截然不同；如果使用高级语言，例如 FORTRAN 或 C/C++ 进行程序设计，经编译后再在这两种机器上运行，就会发现它们之间不再有什么太大的区别。

那么，导致这种情况出现的原因是什么呢？主要是编译程序起的作用。为了能够运行高级语言程序，各种档次的计算机都配有针对某种语言的编译程序，由编译程序对高级语言程序进行编译，然后再运行。这样，站在高级语言的角度来看，两种机器似乎没有什么区别。其实质是编译程序填补了两种机器之间的差异。像这种本来存在差异或区别的事物或特性，站在某一角度来看，似乎不再有什么差异或者区别，这种现象或概念称为透明性 (transparency)。

由此可见，在各级层次结构的机器上，都有相应的系统结构。一般而言，底层机器的属性对高层机器的程序员来说往往是透明的。对于非计算机专业的高层次用户来说，总是希望透明性越多越好。但是，对于从事计算机系统结构研究的人们来说，则要透过这种透明性，去发现或研究不同层次机器之间的差别。在这里，Amdahl 提出的系统结构是指传统机器语言级的系统结构，也就是机器语言程序员所看到的计算机的外特性，是在传统的硬件层面上看到的特性。这种外特性主要包括：

- ① 指令系统，包括操作类型、指令格式、指令间的排序和控制机构等；
- ② 数据表示，也就是硬件能够直接识别和处理的数据类型；
- ③ 寻址规则，它反映机器中最小的寻址单元、寻址方式及其表示方法；
- ④ 寄存器结构，包括寄存器的定义、类型、数量及使用方式；
- ⑤ 中断系统，包括中断类型，硬件响应机构、条件与方式；
- ⑥ 工作状态的定义与切换，例如系统态和管态等，以及它们之间的切换方式；
- ⑦ 存储系统，包括主存容量、程序员编程可寻址的最大存储空间等；
- ⑧ 信息保护，包括信息保护的方式和硬件对信息保护的支持；
- ⑨ I/O 结构，包括数据输入/输出方式，CPU/存储器与 I/O 设备的连接方式，数据传送时

的格式及控制方式等。

这些特性在计算机系统中是由硬件或固件完成的功能，程序员只有在了解了这些特性之后，才能编写出在传统机器上正确运行的程序。由图 1.1 可以看出，计算机系统的概念性结构和功能特性实质上是由计算机系统中软件与硬件之间的界面所确定的，界面之上主要是软件，界面之下则主要是硬件。由此可见，研究计算机系统结构的一个重要的目的就是划分硬件与软件之间的界面。因为有许多功能，既可以由硬件来实现，也可以由软件来实现。

在以上的讨论中，仅仅是针对经典的计算机系统结构的概念而言的。随着计算机技术的发展，计算机系统结构的概念和它所包含的内容也正处在不断地发展与变化之中。如果要进行更深入的研究与探讨，就要涉及广义计算机系统结构的概念，包括新的计算机组成与实现技术。希望广大读者在掌握了经典的计算机系统结构之后，大胆地向广义的系统结构进军。

1.2.3 计算机系统的组成与实现

1. 计算机的组成与实现

在了解了计算机系统结构的含义之后，就可以讨论计算机的组成与实现了。所谓计算机组成，其任务是在明确计算机系统结构分配给硬件的功能与概念性结构之后，研究如何组成其硬件体系，并建立相互之间的联系，以实现机器语言级指令的功能与特性。其中包括各功能部件的配置、组成，相互之间的连接与相互作用等。各功能部件的性能、参数，以及相互之间的合理匹配是计算机组成的重要指标。因而，计算机组成涉及计算机的许多组成方法与技术，例如，运算器的组成、功能、运算速度，存储器的层次结构、容量、存取周期，以及虚拟存储技术等；为了使输入/输出设备与处理机之间的信息流量相平衡，需要采取什么样的通道技术或 I/O 处理机等；为了进一步提高整个计算机系统的速度，是否需要采用先行控制、流水线技术、多功能部件、阵列机及并行多处理机技术等。

由此可见，计算机组成是计算机系统结构的逻辑实现，包括机器内部数据流和控制流的组成及其逻辑设计。在研究计算机组成时，还涉及各种功能部件的组成逻辑、时序电路，以及时钟信号的设计与选用。例如，在设计运算器时，既涉及时序电路，又涉及算法；在设计存储器时，既涉及采用什么样的器件，又涉及它的容量、外围电路和速度；在设计输入/输出通道时，同样也涉及采用什么样的器件及外围电路。另外，在构成计算机时还涉及信号的传输方式、模块的划分、插件与底板的连接，以及电源、冷却方式、组装工艺和技术等。这些，统称为计算机组成的物理实现。

计算机系统结构、计算机组成和计算机实现是三个不同的概念。计算机系统结构是指计算机系统的软件与硬件之间的界面，计算机组成是指计算机系统结构的逻辑实现，计算机实现是指计算机组成的物理实现。它们各自包含不同的内容，相互之间又有着密切的联系。

但是还应当指出，这三种概念所包含的内容根据机器的不同可能有所变化。在一些机器中属于系统结构的内容，而在另一种机器中可能成为计算机组成或实现的内容。例如，高速缓冲存储器 Cache 一般是作为计算机组成提出来的，其中存储的信息全部由硬件管理，对程序员来说是透明的。然而在有的机器中为了提高效率，配置了 Cache 操作指令，使程序员可以通过指令参与 Cache 的管理。这样，Cache 就成为系统结构的一部分，对程序员不再透明了。

按照 Amdahl 对计算机系统结构的定义，计算机系统结构主要是指机器语言指令系统及其

执行模型。这样，在一个系列机中不同档次的机器具有相同的系统结构。

2. 外特性、内特性与微外特性

通过以上分析可以知道，对于机器语言程序员所看到的计算机的概念性结构与功能特性，我们称之为计算机的系统结构，也就是计算机系统的“外特性”。这种外特性是由硬件和固件来实现的，而硬件和固件本身则是由电子线路、逻辑门、寄存器等器件组成的。对这些器件和电子线路的抽象，称为计算机系统的“内特性”，对应于计算机组成，其本身实际上就是计算机系统的实现。

计算机系统的内特性所涉及的内容主要包括机器语言级内的数据流和控制流的组成，CPU与主存等部件的逻辑设计，各事件的排序方式与控制机构，以及各部件之间的联系等。它所关心的问题主要是如何合理地实现分配给硬件的功能及各项技术指标。

在设有微程序控制器的计算机系统中，执行指令的过程实际上就是执行微程序的过程。对于微程序设计人员来说，他们所看到的外特性实际上就是微程序控制器的外特性，也称为“微外特性”。如果把它作为一个抽象级，可以视为硬件与固件之间的界面，也称为微系统结构。外特性、内特性及微系统结构之间的关系如图 1.2 所示。

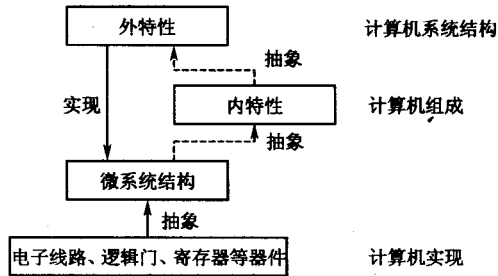


图 1.2 计算机的外特性、内特性与微系统结构的关系

由于计算机系统的外特性是对内特性的抽象，因此二者之间的映射关系是一对多的关系，即某一种外特性可以映射多种内特性，而某一种内特性也可以映射多种物理实现。因此，计算机组成和它的具体实现可以由设计者根据所需要的性能价格比及实现的难易程度，在较大的范围内进行选择。

3. 计算机系统的组成与实现

计算机系统的组成既包括硬件系统，又包括软件系统。它的组成与实现如图 1.1 所示，即分为下层的物理机和上层的虚拟机。其系统的组成实际上就是各层的硬件和软件构成的统一体，由其硬件和软件实现各层的功能。各层的组成与作用已在 1.2.1 节介绍，这里不再赘述。

1.3 计算机系统结构的类型

对于计算机的系统结构，可以根据不同的需求进行分类。人们常用的分类方式主要有三种，一种是按“流”分类，一种是按“并行级”和“流水线”分类，另一种是按照计算机系统结构的最大并行度分类。

1.3.1 按“流”分类

按“流”分类法是 Flynn 教授在 1966 年提出的一种分类方法，他是按照计算机中指令流（Instruction stream）和数据流（Data stream）的多倍性进行分类的。

所谓指令流，是指机器执行的指令序列；所谓数据流，是指指令流调用的数据序列。多倍性是指在机器中最受限制（瓶颈最严重）的部件上，在同一时间单位内，最多可并行执行的指令条数或处理的数据个数。这样，计算机系统结构可分成表 1.2 中的 4 种类型，即单指令流单数据流（SISD）、单指令流多数据流（SIMD）、多指令流单数据流（MISD）和多指令流多数据流（MIMD）。4 种类型的逻辑结构如图 1.3 所示，其中 CU 表示控制部件，PU 表示处理部件，MM 是主存模块，CS 表示控制流，DS 表示数据流，IS 表示指令流，SM 表示共享主存。

表 1.2 计算机系统结构的 Flynn 分类

指令流 \ 数据流	单	多
	单	SISD
多	MISD	MIMD

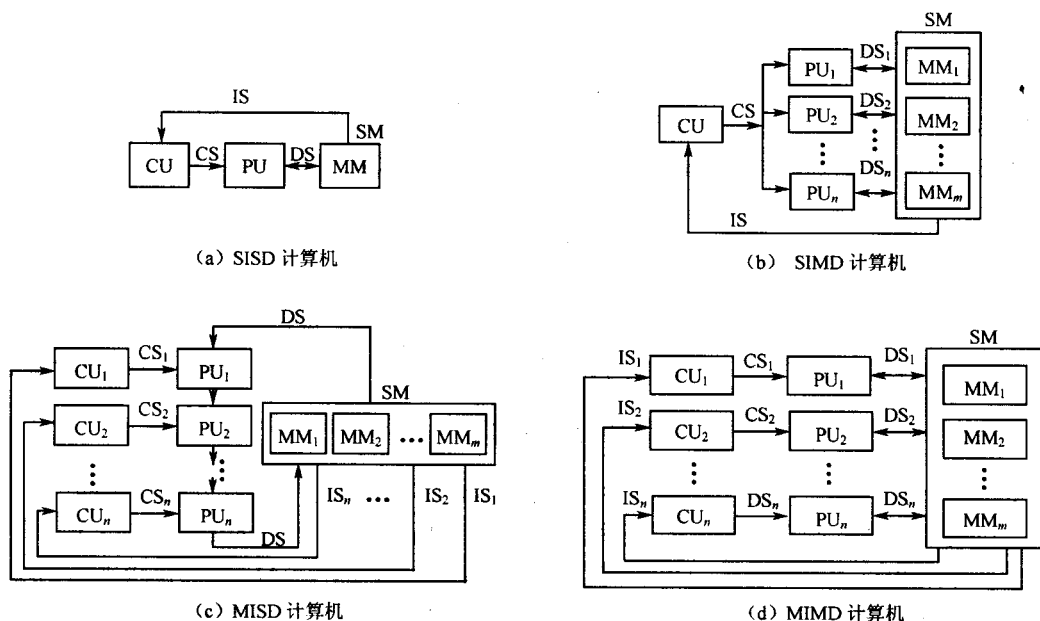


图 1.3 Flynn 分类法中 4 种类型的逻辑结构

1. SISD (Single Instruction Stream Single Data Stream) 计算机

单指令流单数据流计算机的逻辑结构如图 1.3 (a) 所示。其 CU 只有一个，PU 也只有一个。因此，在同一时间单元内只能对一条指令进行译码和执行。由于 MM 也只有一个，因此一次所能处理的数据也只有一个。

在计算机发展的初期，其系统结构一般属于这种类型，现在一些简单的微程序控制器也采取这种结构，例如 IBM 370/1680P, VAX 11/780 等。

2. SIMD (Single Instruction Stream Multiple Data Stream) 计算机

单指令流多数据流计算机的逻辑结构如图 1.3 (b) 所示。其 CU 只有一个，因此在同一

时间单元内只能对一条指令进行译码和执行；但是它有多个 PU，可同时执行控制单元分配给各自的任务，对多个数据进行处理，即多数据流。并行处理机是这种结构的典型代表，例如 ILLIAC-IV，ICL-DAP，CRAY-1，YH-1，以及 CM-2 等都属于这种结构。

3. MISD (Multiple Instruction Stream Single Data Stream) 计算机

多指令流单数据流计算机的逻辑结构如图 1.3 (c) 所示。其 CU 有多个，PU 也有多个，因此在同一时间单元内可对多条指令进行译码和执行，即多个指令流。但是主存模块 (MM) 只有一个数据通路，由多个处理模块对一个数据流进行处理。

就目前来看，有人认为这种结构的计算机尚不存在，但是也有学者认为，超级标量机、超长指令字计算机、退耦 (Decoupled) 计算机及脉动阵列机可以归类为 SIMD 计算机。尤其是在退耦计算机中，有两个独立的程序计数器、两个指令流，但是只有一个数据流。

4. MIMD (Multiple Instruction Stream Multiple Data Stream) 计算机

多指令流多数据流计算机的逻辑结构如图 1.3 (d) 所示，它多个 CU 和多个 PU，可以对多个数据流进行处理。目前的大多数多处理机系统和多计算机系统都属于这种类型。其中，多个处理机共享一个主存空间，并由共享主存交换或传递数据信息的 MIMD 称为紧耦合 MIMD 系统；不直接共享一个主存空间的称为松耦合 MIMD 系统。实际上，MIMD 是多个独立的 SISD 计算机系统的集合。例如，IBM 370/168MP，CRAY X-MP，YH-2 等都属于这种结构。

实际上，使用 Flynn 分类法也有一些缺点。例如，对于广泛流行的标量流水机的向量流水操作应当属于哪一类，不很明确。若严格按照 Flynn 分类法的定义，标量流水机的向量流水操作不包含多个处理部件，应属于 SISD 类型；但是若从一条指令执行的全过程来理解，每一个向量元素都可以视为一个分离的数据流，即有多个数据流，可归属于 SIMD 计算机。

1.3.2 按“并行级”和“流水线”分类

按“并行级”和“流水线”分类的方法是 Wolfgang Händler 在 1977 年提出来的一种分类方法。他是在计算机系统三个子系统集上，按照并行级和流水线的处理程度进行分类的。这三个子系统也称为三个层次，即处理控制器 (PCU)、运算部件 (ALU) 和位级电路 (BLC)。其中，处理控制器相当于一个处理机或 CPU，运算部件相当于一个功能部件或阵列机中的一个处理单元 (PE)，位级电路相当于 ALU 中进行一位运算的组合逻辑电路。这样，一个计算机系统可以用三对整数来描述：

$$T(C) = \langle k \times k', d \times d', w \times w' \rangle$$

式中，C 为计算机系统型号；

k 为计算机系统中 PCU 的个数；

k' 为可组成流水线的 PCU 的个数；

d 为每一个 PCU 中运算部件 (ALU 或 PE) 的个数；

d' 为可组成流水线的 ALU 或 PE 的个数；

w 为 ALU 或 PE 的字长；

w' 为所有 ALU 或一个 PE 中的流水段数。

如果任意一对参数的第二个元素值为 1，则可以省略。