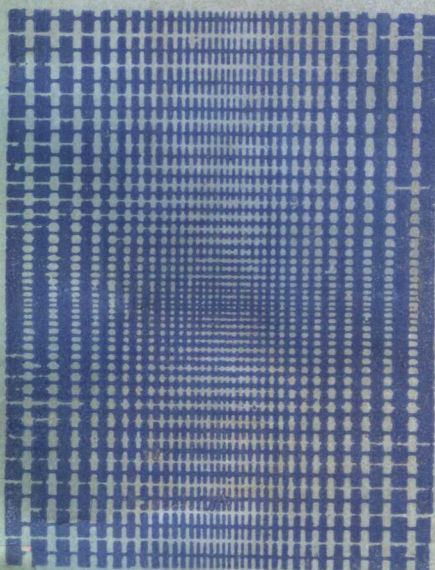


王仁庆 主编 ●

计算机系统结构

北京航空航天大学出版社 ●



王仁庆 主编 ●

计算机系统结构

北京航空航天大学出版社 ●

计 算 机 系 统 结 构

王仁庆 主编

北京航空航天大学出版社

(京)新登字166号

内 容 简 介

本书着重讲述计算机系统结构的基本概念、结构原理、设计方法和性能分析。其中介绍了现代计算机系统的新概念和新技术,并用典型结构举例,以加深理解。尽量反映出现代计算机系统结构的特点和水平。

全书分为九章:概论、计算机系统描述语言、数据表示和指令系统、信息加工与控制结构、存贮体系、输入输出系统、并行处理机与多处理机系统、新型计算机系统、计算机系统可靠性与性能评价。

本书适用于计算机专业本科高年级教材,并可供有关专业的研究生和专业技术人员参考。

计 算 机 系 统 结 构

JISHUANJI XITONGJIEGOU

王 仁 庆 主 编

责任编辑 樊毅

北京航空航天大学出版社出版 通县觅子店印刷厂印装

新华书店总店科技发行所发行 各地新华书店经销

开本 850×1168 1/32 印张: 11.5 字数: 309千字

1992年8月 第一版 1992年8月第一次印刷 印数: 4000册

ISBN 7-81012-305-X/TP·064 定价: 3.55 元

前 言

本书是由航空航天工业部教材编审室规划和审定的计算机专业教材。是在多年教学实践基础上，经过认真修改后编写而成的。

本书内容着重讲述计算机系统结构的基本概念、结构原理、设计方法、性能分析。全书以提高计算机系统性能出发，强调从并行性观点去讨论各种并行措施和并行结构，力求能反映出现代计算机系统的特点和发展水平。

计算机系统结构的迅速发展，其特点为：过去用于巨型机、大型机的许多先进技术逐渐被小型机以至微型机采用；硬件结构与算法、程序设计语言、系统软件更加紧密的结合。在本书中也力求反映出这二个特点。

全书包括九章：第一章论述系统结构的基本概念；第二章介绍系统不同层次的描述语言；第三章系统地介绍了各种数据表示、编址技术和指令系统；第四章讨论重迭控制、流水技术和向量处理机；第五章讨论存贮体系；第六章论述I/O系统的发展和中断、通道、I/O处理机等；第七章论述并行处理技术和并行处理机、多机处理技术和多处理机系统；第八章为新型计算机系统，主要介绍了RISC技术和数据流计算机；第九章简要介绍了计算机系统可靠性和性能评价。系统结构的有些内容，如纠错和容错技术，可靠性技术以及新型的归约计算机、高级语言计算机、数据库计算机、神经网络计算机、智能计算机等等，因单设课程或因篇幅和学时数等因素，没有包括在内。

本书在阐明原理之后，都用具有代表性的典型结构为例，进行分析说明，使内容充实、具体。在每章后面都附有习题。在编写上力求概念清楚，含义明确，语言简练。

本书第三、四、五章由葛本修编写，第六章由曾能浚编写，其余各章由王仁庆编写，全书由王仁庆负责统稿。

本书出版得到了北京航空航天大学教材科和航空工业部教材编审室的支持和帮助，北京理工大学张绍诚教授对书稿认真审阅，提出了宝贵的意见，在此深表感谢。

由于时间仓促，水平有限，在选材和编写中难免有谬漏之处，敬请读者指正。

编 著 者

目 录

第一章 概 论

§1.1 计算机系统结构含义	(1)
§1.1.1 硬件、软件及功能分担	(1)
§1.1.2 系统结构、组织与实现	(2)
§1.1.3 计算机系统分层结构	(7)
§1.2 系统结构发展历程	(10)
§1.2.1 诺依曼型结构及其演变	(10)
§1.2.2 系统结构的并行性及其发展历程	(13)
§1.2.3 单机和多机系统并行性发展	(25)
§1.3 计算机系统分类与分型	(26)
§1.3.1 计算机系统分类	(26)
§1.3.2 计算机系统分型	(30)
习题	(32)

第二章 计算机系统描述语言

§2.1 计算机系统的硬件层次	(34)
§2.2 系统总体结构描述	(35)
§2.2.1 方框图和PMS表示法	(35)
§2.2.2 PMS应用	(39)
§2.3 处理器层描述——ISP语言	(40)
§2.4 寄存器传送层描述——CDL语言	(46)
§2.4.1 说明语句	(47)
§2.4.2 执行语句	(50)
§2.5 系统的动态模型	(50)
习题	(57)

第三章 数据表示与指令系统

§3.1 数据的表示	(58)
§ 3.1.1 数据表示与软硬件界面的确定	(58)
§ 3.1.2 数据表示的确定	(60)
§ 3.1.3 向量数据表示	(69)
§ 3.1.4 自定义数据	(72)
§3.2 编址技术与寻址方式	(78)
§ 3.2.1 编址技术	(79)
§ 3.2.2 寻址方式	(82)
§3.3 指令系统	(83)
§ 3.3.1 指令格式	(84)
§ 3.3.2 指令种类	(89)
§ 3.3.3 VAX-11系列指令系统举例	(93)
习题	(99)

第四章 信息加工与控制结构

§4.1 单处理机系统的并行性	(102)
§ 4.1.1 处理机与I/O的重迭操作	(103)
§ 4.1.2 处理机与存储器的重迭操作	(103)
§ 4.1.3 存储器内部的重迭操作	(103)
§ 4.1.4 处理机内部的重迭操作	(104)
§ 4.1.5 按字或字节并行操作	(104)
§ 4.1.6 按位片并行操作	(105)
§ 4.1.7 按向量并行操作	(105)
§4.2 高性能运算部件	(106)
§ 4.2.1 高速乘法部件	(106)
§ 4.2.2 高速浮点运算部件	(111)
§4.3 重迭控制方式	(111)
§ 4.3.1 最简单的重迭	(113)
§ 4.3.2 一次重迭	(114)
§4.4 先行控制方式	(116)

§4.4.1 基本原理	(116)
§4.4.2 先行控制中的问题	(118)
§4.5 流水处理技术	(126)
§4.5.1 流水处理的原理	(126)
§4.5.2 流水线的分类	(129)
§4.5.3 流水线性能分析	(132)
§4.5.4 流水线设计中的几个技术问题	(134)
§4.5.5 举例——IBM360/91	(143)
§4.6 向量处理机	(148)
§4.6.1 向量处理机的特点	(148)
§4.6.2 向量处理机的处理方式	(148)
§4.6.3 向量处理机的指令系统	(150)
§4.6.4 向量处理机结构	(151)
§4.6.5 举例——YH-1系列	(153)
习题	(160)

第五章 存贮体系

§5.1 概述	(164)
§5.1.1 存贮器系统的发展	(164)
§5.1.2 层次结构	(165)
§5.2 多体并行存贮器	(170)
§5.2.1 单体并行访问多字	(170)
§5.2.2 多体并行存取	(171)
§5.2.3 多体并行交叉存取	(172)
§5.2.4 对速度的估算	(174)
§5.2.5 举例——YH-1系列	(175)
§5.3 虚拟存贮器	(178)
§5.3.1 多道程序运行与程序再定位	(178)
§5.3.2 虚拟存贮器的地址映象	(179)
§5.3.3 替换算法及其实现	(185)
§5.3.4 虚拟存贮器的工作过程	(189)
§5.3.5 虚拟存贮器性能分析	(193)

§5.3.6 举例—VAX11系列	(194)
§5.4 高速缓冲存储器	(196)
§5.4.1 高速缓冲存储器的结构与工作过程	(196)
§5.4.2 地址转换方式	(197)
§5.4.3 高速缓冲存储器性能分析	(202)
§5.4.4 高速缓冲存储器举例	(204)
§5.5 主存存储器保护	(206)
习题	(208)

第六章 输入输出系统

§6.1 输入输出系统的发展与一般结构	(211)
§6.1.1 概述	(211)
§6.1.2 输入输出控制方式	(211)
§6.1.3 主机与输入输出设备的连接	(212)
§6.2 总线结构	(215)
§6.2.1 总线的类型	(215)
§6.2.2 总线控制	(216)
§6.2.3 总线通讯方式	(221)
§6.2.4 I/O总线的标准化	(222)
§6.3 中断系统	(224)
§6.3.1 中断的定义及分类	(224)
§6.3.2 中断结构	(226)
§6.4 通道与输入输出处理机	(230)
§6.4.1 通道的工作过程	(230)
§6.4.2 输入输出处理机 (IOP) 结构	(233)
习题	(250)

第七章 并行处理机与多处理机

§7.1 并行处理机结构特点	(252)
§7.1.1 两种结构类型	(252)
§7.1.2 处理单元PE基本组成	(254)
§7.1.3 处理单元之间的通信	(257)

§7.2 互连网络	(258)
§7.2.1 单级互连网络	(258)
§7.2.2 多级互连网络	(261)
§7.2.3 介绍几种互连网络	(263)
§7.3 并行处理机典型结构	(271)
§7.3.1 ILLIACIV阵列处理机	(271)
§7.3.2 BSP科学处理机	(278)
§7.4 并行处理机性能分析	(282)
§7.5 多处理机特征与结构	(285)
§7.5.1 多处理机系统特征	(285)
§7.5.2 两种结构类型	(285)
§7.5.3 多处理机互连方式	(289)
§7.6 多处理机操作系统	(295)
§7.6.1 多处理机操作系统的特点和要求	(295)
§7.6.2 多处理机操作系统类型	(295)
§7.6.3 多处理机调度	(296)
§7.6.4 进程通信	(299)
§7.6.5 系统死锁	(301)
§7.7 并行性检测与并行程序设计	(302)
§7.7.1 数据相关性分析	(302)
§7.7.2 程序并行性检测	(304)
§7.7.3 并行程序设计语言	(305)
习题	(312)

第八章 新型计算机系统

§8.1 精简指令集计算机 (RISC)	(315)
§8.1.1 RISC技术形成与发展	(315)
§8.1.2 RISC设计原则与技术措施	(318)
§8.2 数据流计算机	(326)
§8.2.1 数据驱动计算	(326)
§8.2.2 数据流程图	(328)
§8.2.3 数据流语言	(331)

§ 8.2.4 数据流机系统结构.....	(332)
§ 8.2.5 数据流机的优点和存在的问题.....	(338)
习题	(340)

第九章 计算机系统可靠性评价与性能评价

§9.1 计算机系统可靠性评价	(341)
§ 9.1.1 可靠性评价指标.....	(341)
§ 9.1.2 可靠性评价方法.....	(343)
§ 9.1.3 提高系统可靠性措施.....	(344)
§9.2 计算机系统性能评价	(345)
§ 9.2.1 性能评价目的.....	(345)
§ 9.2.2 性能评价指标.....	(345)
§ 9.2.3 性能评价方法.....	(346)
习题	(358)
参考文献	(359)

第一章 概 论

§ 1.1 计算机系统结构含义

§ 1.1.1 硬件、软件及功能分担

计算机问世以来,迄今已近半个世纪。在计算机设计、工程和应用等方面都取得巨大的成就。现在人们所说的“计算机”无论巨型机、大型机、中型机、小型机以至微型机,实际上是指“计算机系统”,与早期的计算机及其它计算工具的区别,就在于“计算机系统”是由“硬件”和“软件”两者不可缺少的部分组成的。随着计算机的发展和广泛应用,“硬件”和“软件”已超出了计算机领域,成为当前人们普遍乐于使用的术语。因此,今天我们说“硬件”和“软件”,有必要在前面加上“计算机”,以别于其它领域。

计算机硬件是指计算机系统物理上存在的装置或设备,是计算机系统的物质基础。它包括:直接执行机器指令、处理数据的中央处理器;保存大量数据信息的存贮体系;与外界(人或其它设备)接口并交换信息的输入输出系统。

计算机软件指的是计算机系统运行所必要的程序及有关资料。它包括:系统自动管理的操作系统;翻译和加工程序设计语言的汇编程序、编译程序;检测系统正确性的诊断程序、检测程序;调试程序、数据库管理程序和各种各样的应用程序包等等。

软件不仅是保证现代计算机系统运行所必需的组成部分,也是扩展硬件功能、提高系统效率、增加运行可靠性的工具。然而,软件必须依附于硬件才能起作用。

“硬件”和“软件”在计算机系统中,各自承担着所分配的功能,但是它们却具有逻辑功能的等效性。某些由软件实现的功能

能,在原理上可以由硬件来实现;同样,某些由硬件实现的操作,可以由软件来实现。因此,具有同样功能的计算机系统,其硬件和软件的功能分担可以在很大范围内变化。随着计算机系统的不断发展,两者的功能分担,也不断地相互转化和渗透,它们之间的界面也在动态地变化。例如,早期的计算机有复杂的控制面板,由人工操作和监控运行状态。自从有了操作系统后,这些装置就大大减少了,代之由操作员输入几条简单的命令。又如,早期机器没有乘、除法指令,乘、除法运算是由软件(子程序)循环执行加法和移位指令来实现的,后来的机器则用乘、除法指令,直接由硬件实现。其它如字符运算、浮点运算等在目前大、中型机器里也都由硬件取代了原来的软件。软件和硬件在逻辑功能上的等效性还可以扩展到编译、检测、中断处理、存贮管理等。

硬、软件功能分担在不同时期的相互转化,是与计算机技术、元器件的发展、硬软件成本相对变化有密切的关系。一般来说,硬件分担的比重加大,对提高处理速度是有利的,但系统的成本会相应增加;而软件比重加大,则可降低系统成本,并有利于提高系统灵活性和适应性,但运算时间可能加长。然而,随着微电子技术的发展,硬件成本正在日益降低,尤其是超大规模集成电路(VLSI)技术的发展和运用,各种高密度的存贮芯片和门阵列电路的商品化,使得原先必须用软件实现的功能,可以用硬件或固件(软件固化)来实现,提高了计算机系统的性能价格比。现代的计算机系统,在设计和应用上将是硬件和软件更加密切配合,在功能上将会更多的相互转化和渗透。

§ 1.1.2 系统结构、组织与实现

计算机系统结构(Computer Architecture),目前还没有统一的确切定义。这个概念是在六十年代初期,由IBM公司的一些设计人员提出来的。1962年,布洛克(F.P.Brook)就提出:“计算机系统结构,是一门技艺,它确定用户对某些结构的要求,然后进行设计,在经济和技术的条件内,尽可能地有效地满

足这些要求。还应包括工程方面的考虑，从而使设计是经济的和可行的，但它的重点在于满足用户的要求，而工程方面的考虑则在于满足制造者的要求。”1964年阿姆达尔（G.M.Amdahl）在介绍IBM360系统时，曾为这个概念定义如下：“系统结构这个词表示从程序员角度看系统的属性……概念性地论述其结构，对功能特性加以说明。”之后，又有许多专家对系统结构作过这样那样的解释和定义，但是似乎一致认为，计算机系统结构是从整体出发，提供程序员有用的工具，或者说，从程序员的角度上来看计算机的属性。然而，“程序员”这个名称又可以有多种解释，如系统程序员、应用程序员、程序分析员等。同时，程序设计又有不同的语言层次，他们所看到的或者所关心的计算机属性也是不同的。早期的计算机用户，只用机器语言或汇编语言编写程序，他们不仅要熟悉机器的指令系统，而且需要了解更多的计算机内部结构、操作方式及输入输出方式等。但随着普遍使用高级语言编写程序之后，机器的指令系统、存贮体系及其它内部结构不那么重要和被关心了。比如，对于使用FORTRAN77编写程序的用户来说，IBM机和VAX机几乎没有多少差别，几乎有相同的属性，而实质上，这两类机器在指令系统、寻址模式、输入输出系统等方面都有显著的不同。这种本来存在的、有差异的事物和属性，从某种角度看又好像是不存在的现象，在计算机技术中通常称为“透明性”。两种不同的机器在指令系统等方面本来存在的不同属性，使用高级语言编程的程序员却看不见。也就是说，这些属性对高级语言程序员来说是透明的。“透明性”这个术语，在计算机系统开发和应用的许多场合中经常使用。

计算机系统总是在不断发展，概念总应该是明确的和有限的。计算机系统结构这个概念究竟是限定在哪一层次语言编程的程序员看到的机器属性。一般认为，应该是指使用机器语言或汇编语言编写程序的程序员。这样，才能反映系统内部结构、各主要部件之间的关系以及系统运行状态。本章下一节介绍了计算机系统的多级分层结构之后，这个概念会更清楚了。

但是,也有人认为:计算机系统结构的研究范围仅仅限于程序员看到的机器属性这一方面还是不够的,因为机器的许多内部功能结构,对于即使是使用机器语言的程序员也是透明的。例如,高速缓冲存储器、重叠和流水线结构、虚拟存储体系等,在指令系统中往往反映不出来,这就要深入了解计算机的逻辑组成和数据流程。因此,计算机系统结构研究还应该包括:从设计者的角度上,以最合理的概念性结构与功能特性组合起来,才能得出较完整的计算机系统结构概念,才能设计出高性能的计算机系统,以满足用户所需要的系统环境。

总的来说,计算机系统结构就是要在整体上提供程序员以有用工具,并满足算法、功能和性能需要,确定最合理的概念性结构,包括系统基本组织之间的关系。为此,需要确定软硬件的界面和系统中软硬件功能的分配。为达到系统的设计目标和具体应用需求,在当前器件发展的水平上,使软硬件紧密配合,提供合理的硬件支持和最佳的组合,从系统的整体出发,使软硬件都能发挥最大的功效。

对于目前通用计算机系统来说,概念性结构与功能特性,大致包括:

数据表示;

寻址方式,包括最小寻址单元和地址运算;

寄存器定义、数量和使用方式;

指令系统的类型、格式、指令间排序和控制机构;

中断方式和机构;

机器工作状态的定义和变换;

存储体系和管理;

I/O结构和数据传送方式;

信息保护的硬件支持,等等。

计算机组织(Computer Organization),是在系统结构所确定的机器概念性结构和功能特性范围内,研究机器内数据流和控制流的组织及其逻辑实现。它着重于机器内部事件的时序和

控制机构。例如，由系统结构所确定的指令系统，在指令执行过程中，取指、译码、取操作数、运算、送结果，这些事件的操作和时序则属于计算机组织。因此，同样的指令系统，因速度、成本等要求的不同，可以有不同的机器组织。

对于一般的计算机系统来说，其组织设计大致包括：

数据通路宽度；

功能部件的设置、共享程度以及并行执行；

专用功能部件的设置；

缓冲、排队技术和预测技术；

可靠性技术；

控制机构组成，等等。

计算机实现 (Computer Implementation)，指的是计算机组织的物理实现。包括的主要内容是：处理机、主存贮器等物理结构；器件、模块、插件、底板的划分与连接；器件的集成度、速度和体积的考虑；专用器件和线路的设计；还应包括电源、冷却、整机装配等等。

例如，在指令系统中确定用乘法指令，这是属于系统结构的内容，但乘法指令采用何种算法及逻辑结构，则属于计算机组织的内容，而乘法器的物理设计，即电路设计、微组装技术等，则属于计算机实现的内容。又如，主存贮器的容量和寻址方式的确定，是系统结构的内容，主存的存取速度应多快，逻辑结构应采取什么措施属于计算机组织，而存贮器采用什么器件、具体电路设计和组装技术，则属于计算机实现的内容。

同一种计算机系统结构可以有多种逻辑组成方案；同一种计算机组织也可以采用多种物理实现方法。后者是前者的基础，先进的实现技术，尤其是器件发展将对系统结构和组织产生巨大的影响和积极的推动作用。例如，超大规模集成电路 (VLSI) 的发展，使门阵列逻辑电路得以普遍使用，有利于多处理机和流水线系统结构的实现。微程序技术是计算机组织的一种方式，它的应用能使软件固化和硬件软化，使硬软件之间界面发生变化，也引

起了系统结构的深刻变化。反过来系统结构也要影响到计算机组织技术和实现方法。例如，如果指令系统是寄存器-寄存器型二地址的指令格式，为了提高速度，可采取寄存器并行操作的计算机组织技术；而主存-主存型三地址指令格式，可采取交叉存贮体系的并行访问方式，或采用高速缓冲存贮器的多级存贮体系的组织方式。

为进一步说明系统结构、组织和实现这三者之间关系，“系列机”是一个很好的例子。所谓“系列机”，是指在同一厂家生产的具有基本相同的系统结构，但有不同的计算机组织和实现，形成一系列不同型号的机器。IBM公司在IBM360系列机问世后，又相继推出IBM370、303X、43XX、308X、309X等系列；DEC公司在PDP11系列之后，又推出VAX-11、8000、6200、6300、6400等系列。以VAX-11系列为例，该系列不同型号的机器都是32位字长、有相同的指令格式、数据类型、寻址模式、虚存空间，都采取I/O设备寄存器与主存统一编址方式，以及它们都能运行VAX/VMS操作系统等等。也就是说，它们都属于VAX-11系统结构。但是它们的组织与实现却不完全相同。VAX-11/730采用位片和可编程的阵列逻辑技术（PLA），有一个单总线（VNIBUS）适配器与大容量存贮器或外围设备相连接。VAX-11/750采用双极型LSI肖特基逻辑，有一个单总线和多至三个多总线（MASSBUS）适配器，而VAX-11/780则采用多达四个单总线适配器和四个多总线适配器。在VAX-11/750、780中都有高速缓冲存贮器（Cache），而VAX-11/730中没有。这些不同的总线结构与存贮体系对程序员来说是透明的。图1-1示出VAX-11的三种硬软件组成。

可见，一种系统结构可以有多种组织和实现，以满足不同用户的要求。然而，正因为系列机对程序员所看到的机器属性相同，他们所编写的机器语言程序以及编译语言程序都能通用于同系列的各种型号，我们称这种情况下的各型号机器是“软件兼容”的。系列机的软件兼容，指的是同一个软件可以不加修改地