



J 刁宏志 刘亚秋 赵更寅 编著
ISUANJI XITONG JIEGOU

计算机系统结构



东北林业大学出版社

TP303

143

2007

计算机系统结构

刁宏志 刘亚秋 赵更寅 编著

东北林业大学出版社

图书在版编目 (CIP) 数据

计算机系统结构/刁宏志, 刘亚秋, 赵更寅编著. —哈尔滨: 东北林业大学出版社, 2007.2

ISBN 978-7-81076-957-0

I. 计… II. ①刁… ②刘… ③赵… III. 计算机体系结构
IV. TP303

中国版本图书馆 CIP 数据核字 (2007) 第 020668 号

责任编辑: 张红梅

封面设计: 彭 宇



NEFUP

计算机系统结构

Jisuanji Xitong Jiegou

刁宏志 刘亚秋 赵更寅 编著

东北林业大学出版社出版发行

(哈尔滨市和兴路 26 号)

东北林业大学印刷厂印装

开本 787 × 960 1/16 印张 14.75 字数 264 千字

2007 年 2 月第 1 版 2007 年 2 月第 1 次印刷

印数 1—1 000 册

ISBN 978-7-81076-957-0

TP·76 定价: 25.50 元

前 言

本书是高等学校计算机专业本科生及研究生计算机系统结构课程的通用教材。本教材通过介绍目前通用的和先进的计算机系统的具体实例,来阐述计算结构原理的各个侧面,讲述各种通用结构的基本原理以及应用场合,并力求从系统分析和设计的角度深入浅出地进行介绍,使读者能够比较全面地掌握计算机系统结构的基本概念、基本原理、基本结构和基本分析法。并且着重强调量化分析能力和分析方法,通过大量的实际例题,说明如何进行量化分析,使读者能够更具体、更实际地了解计算机系统结构的分析、设计过程。

计算机系统结构是一门从外部来研究计算机系统的科学,是研究计算机系统结构设计、性能综合评价和提高计算机系统结构性能方法的科学。在学习本课程之前,我们已经掌握“计算机组成原理”、“计算机操作系统”、“汇编语言程序设计”、“数据结构”等计算机硬件和软件方面的诸多计算机知识。而计算机系统结构是从设计和分析的角度讲解计算机的各个侧面,对全面、深入理解计算机结构有着重要意义。本书的主要内容包括:计算机系统的性能评价、数据表示、指令系统、流水线技术、存储系统、输入输出系统、多处理机等几大部分。

本书共分六章,第一章为计算机系统结构的基本概念,讨论了计算机系统结构的概念、发展过程、定量分析技术基础及并行性概念。第二章论述了计算机指令集结构设计,讨论了计算机指令集结构的分类、功能设计、数据表示及指令集格式的设计。第三章为流水线处理机,分别探讨了计算先行控制技术、流水线处理机、超标量处理机、超流水线处理机和向量处理机。第四章为存储系统,论述了计算机存储系统的原理、高速缓冲存储器(Cache),及提高Cache性能的方法。第五章为输入输出系统,讨论了计算机输入输出原理、通道处理机及I/O处理机。第六章为多处理机,论述了多处理机的存储体系结构、互连网络及并行化技术。

本书由刁宏志副教授撰写第一章和第四章,刘亚秋博士撰写第六章,赵更寅撰写第三章和第五章,刁宏志和刘亚秋联合撰写第三章。

由于作者水平有限,书中难免有错误和不妥之处,敬请读者批评指正。

作 者
2007年1月

目 录

1 计算机系统结构的基本概念	(1)
1.1 计算机系统结构的概念	(1)
1.2 计算机体系结构的发展	(10)
1.3 计算机系统结构中并行性的发展	(18)
1.4 计算机系统结构的设计	(21)
1.5 定量分析技术基础	(26)
习题	(33)
2 数据表示与指令系统	(36)
2.1 数据表示	(36)
2.2 指令集结构的分类	(51)
2.3 寻址技术	(55)
2.4 指令格式的优化设计	(58)
2.5 指令集结构的功能设计	(65)
习题	(72)
3 流水线技术	(75)
3.1 先行控制技术	(75)
3.2 流水线的基本概念	(80)
3.3 流水操作中的相关	(101)
3.4 向量流水处理	(113)
3.5 超标量处理机与超流水处理机	(120)
习题	(124)
4 存储系统	(127)
4.1 存储系统的基本概念	(127)
4.2 Cache 的基本知识	(132)
4.3 降低 Cache 失效率的方法	(148)
4.4 减少 Cache 失效开销	(160)
4.5 减少命中时间	(167)
习题	(172)

2 计算机系统结构

5 输入输出系统	(175)
5.1 引 言	(175)
5.2 输入输出基本概念	(176)
5.3 通道处理机	(183)
5.4 输入输出处理机	(195)
习题.....	(199)
6 多处理机概述	(200)
6.1 引 言	(200)
6.2 多处理机的存储器体系结构	(205)
6.3 互连网络	(214)
6.4 并行化技术	(221)
习题.....	(229)
参考文献	(230)

1 计算机系统结构的基本概念

自从第一台计算机 1945 年诞生至今半个多世纪以来, 计算机技术一直处于发展和变革之中。计算机已从开始的单纯应用与科学计算, 扩展到生产、学习和日常生活的各个方面。如今, 我们用几千元人民币购置的个人计算机, 其性能已经大大超过了在 20 世纪 60 年代用 100 万美元购置的计算机; 互联网络将文字、图片和影像等信息, 带到世界的每个角落; 微处理器、计算机、数字化等对人们的生活、工作、学习等各方面均产生了重大深远的影响。计算机性能如此高速增长, 始终受益于制造技术和计算机体系结构技术的发展。在计算机诞生的前 25 年中, 计算机性能增长相对缓慢。在这个过程中, 制造技术和体系结构同时发挥着作用, 其间充满了尝试和创新。20 世纪 70 年代计算机性能的增长速度达到每年 25% ~ 30%, 这种增长主要归功于以集成电路为代表的电路技术的发展。70 年代, 出现了微处理器, 这是一次计算机设计和制造技术的革命。从 70 年代末到 80 年代中期, 采用微处理器的计算机性能增长达到每年 35%, 与当时广泛使用的大、中、小型计算机相比, 微处理器的性能增长, 更多地依赖于集成电路技术的发展。进入 80 年代以后, 计算机体系结构产生了一次重大变革, 出现了我们现在称之为精简指令集计算机 (reduced instruction set computer, 简称 RISC) 的处理器设计技术。此后, 计算机体系结构不断变革, 在计算机体系结构技术发展的促进下, 集成电路技术为计算机设计提供的技术空间得到了充分的发挥, 计算机系统的性能以每年 50% 以上的速度增长。20 世纪 90 年代中期以来, 尤其是 Intel 公司的 Pentium Pro 的出现, 基于 RISC 设计技术的微处理器大批量上市, 极大地推动了计算机产业的发展, 计算机的系统性能得到全面提升。目前, 计算机性能增长达到每年 50% 以上, 其中包括器件技术在内的计算机、制造技术提供其中约 8%, 其余约 42% 的部分主要依靠计算机体系结构发展的支持。

1.1 计算机系统结构的概念

设计一种新型计算机系统首先必须面临的问题是什么呢? 我们会列举出很多因素, 其中最主要的有新型计算机的主要特点和性能, 它们具体包括指

令集设计、功能组织、逻辑设计、实现技术等。实现技术包括集成电路设计、制造和封装设计、系统制造和组装技术、供电、冷却技术等。另外，我们往往要求在限定的造价范围内，使新型计算机具有最高的性能。如采用先进的计算机体系结构和生产技术，制造出具有高性能价格比（performance cost rate）的计算机系统，是所有通用计算机设计的共同目标。

1.1.1 计算机系统的层次结构

现代计算机系统是由软件和硬/固件组成的十分复杂的系统。为了对这个系统进行描述、分析、设计和使用，人们从不同的角度提出了观察计算机的观点和方法。本节从计算机语言的角度，把计算机系统按功能划分成多级层次结构。

计算机语言主要是指计算机程序设计语言，其基本构成元素是语句，是描述计算机系统可以理解并能够按规定执行的操作。随着计算机系统及应用领域的不断发展，人们已设计出一系列的计算机语言。从面向机器的语言（如机器语言、汇编语言），到各种高级程序设计语言（如 Java、C/C++、Pascal 语言），各种面向问题的语言或者叫应用语言（如面向数据库查询的 SQL 语言、面向数字系统设计的 VHDL 语言、面向人工智能的 PROLOG 语言）。计算机语言就是这样由低级向高级发展，高一级语言的语句相对于低级语言功能更强，更便于应用，但又都是以低级语言为基础的。

计算机语言可分成一系列的层次（level）或级，最低层语言的功能最简单，最高层语言的功能最强。对于用某一层语言编写程序的程序员来说，他一般不管其程序在机器中是如何执行的，只要程序正确，就能得到预期的结果。这样，对这层语言的程序员来说，他似乎有了一种新的机器，这层语言就是这种机器的机器语言，该机器能执行相应层语言编写的全部程序。因此计算机系统就可以按语言的功能划分成多级层次结构，每一层以一种不同的语言为特征。这样，可以把现代计算机系统画成如图 1-1 所示的层次结构。图 1-1 中第 4 级以上完全由软件实现。我们称由软件实现的机器为虚拟机器（virtual machine），以区别于由硬件或固件实现的实际机器。

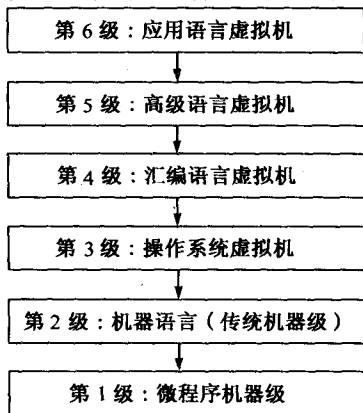


图 1-1 计算机系统的多级层次结构

第1级是微程序机器级。这一级的机器语言是微指令集，工作于该级的程序员实际上是计算机系统的设计人员，程序员用微指令编写的微程序一般是直接由硬件解释实现的。

第2级是传统机器级。这一级的机器语言是该机的指令集，程序员用机器指令集编写的程序可以由微程序进行解释执行，这个解释程序运行在第1级上。

计算机系统中也可以没有微程序机器级。在这些计算机系统中是用硬件直接实现传统机器的指令集，而不必由任何解释程序进行干预。我们目前使用的 RISC 技术就是采用这样的设计思想，处理器的指令集全部用硬件直接实现以提高指令的执行速度。

第3级是操作系统虚拟机。从操作系统的基本功能来看，一方面它要直接管理传统机器中的软硬件资源，另一方面它又是传统机器的引申。它提供了传统机器所没有的某些基本操作和数据结构，如文件结构与文件管理的基本操作、存储体系和多道程序以及多重处理所用的某些操作、设备管理等。

第4级是汇编语言虚拟机。这级的机器语言是汇编语言，用汇编语言编写的程序，首先翻译成第3级和第2级语言，然后再由相应的机器执行。完成汇编语言翻译的程序就称做汇编程序。

第4级上出现了一个重要变化。通常的第1、2和3级是用解释（interpretation）方法实现的，而第4级或更高级则经常是用翻译（translation）方法实现。

翻译和解释是语言实现的两种基本技术。它们都是以执行一串 N 级指令来实现 $N+1$ 级指令，但二者仍存在着差别：翻译技术是先把 $N+1$ 级程序全部变换成 N 级程序后，再去执行新产生的 N 级程序，在执行过程中 $N+1$ 级程序不再被访问。而解释技术是每当一条 $N+1$ 级指令被译码后，就直接去执行一串等效的 N 级指令，然后再去取下一条 $N+1$ 级的指令，依此重复进行。在这个过程中不产生翻译出来的程序，因此解释过程是边变换边执行的过程。一般来说，解释执行比翻译花的时间多，但存储空间占用较少。

第5级是高级语言虚拟机。这级的机器语言就是各种高级语言，目前高级语言已达数百种。用这些语言所编写的程序一般由称为编译（compile）程序的翻译程序翻译到第4级或第3级上，如 C/C++、Pascal 等，个别的高级语言也用解释的方法实现，如 BASIC 语言系统。

第6级是应用语言虚拟机。这一级是为使计算机满足某种用途而专门设计的，因此这一级语言就是各种面向问题的应用语言。可以设计专门用于人

工智能、教育、行政管理和计算机设计等方面的虚拟机，这些虚拟机也是当代计算机应用领域的重要研究课题。应用语言编写的程序一般是由应用程序包翻译到第5级上，甚至越级翻译到级别更低的机器上。

1.1.2 计算机系统结构

计算机体系结构 (computer architecture) 这个词目前已被广泛使用。Architecture 本来用在建筑方面，译为“建筑学”、“建筑术”、“建筑样式”、“构造”、“结构”等。这个词被引入计算机领域后，最初的译法也各有不同，以后趋向译为“体系结构”或“系统结构”，但关于它的定义仍未统一。

经典的“计算机体系结构”定义是1964年C.M.Amdahl在介绍IBM360系统时提出的：“计算机体系结构是程序员所看到的计算机的属性，即概念性结构与功能特性。”

按照计算机系统的多级层次结构，不同级程序员所看到的计算机具有不同的属性。例如，传统机器程序员所看到计算机的主要属性是该机指令集的功能特性；而高级语言虚拟机程序员所看到计算机的主要属性是该机所配置的高级语言所具有的功能特性。显然，不同的计算机系统，从传统机器级程序员或汇编语言程序员的角度来看，是具有不同属性的。但是，从高级语言（如Pascal）程序员看，它们就几乎没有什么差别，是具有相同属性的。或者说，这些传统机器级所存在的差别是高级语言程序员所“看不见”的，也是不需要他们知道的。在计算机技术中，对这种本来是存在的事物或属性，但从某种角度看又好像不存在的现象称为透明性 (transparency)。通常，在一个计算机系统中，低层机器的属性往往对高层机器的程序员是透明的，如传统机器级的概念性结构和功能特性，对高级语言程序员来说是透明的。由此看出，在层次结构的各个级上都有它的体系结构。

这些属性是机器语言程序设计者（或编译程序生成系统）为使其所设计的程序能在机器上正常运行所需遵循的计算机属性，包含概念性结构和功能特性两个方面。目前，对于通用寄存器型机器来说，这些属性主要是指：

- (1) 数据表示（硬件能直接辨认和处理的数据类型）；
- (2) 寻址规则（包括最小寻址单元、寻址方式及其表示）；
- (3) 寄存器定义（包括各种寄存器的定义、数量和使用方式）；
- (4) 指令集（包括机器指令的操作类型和格式、指令间的排序和控制机构等）；
- (5) 中断系统（中断的类型和中断响应硬件的功能等）；
- (6) 机器工作状态的定义和切换（如管态和目态等）；

(7) 存储系统 (主存容量、程序员可用的最大存储容量等);

(8) 信息保护 (包括信息保护方式和硬件对信息保护的支持);

(9) I/O 结构 (包括 I/O 连接方式、处理机/存储器与 I/O 设备间数据传送的方式和格式以及 I/O 操作的状态等)。

这些属性是计算机系统中由硬件或固件完成的功能,程序员在了解这些属性后才能编出在传统机器上正确运行的程序。因此,经典计算机体系结构概念的实质是计算机系统中软硬件界面的确定,该界面之上是软件的功能,界面之下是硬件和固件的功能。计算机体系结构研究任务:主要研究软件、硬件功能分配和对软硬件界面的确定。

综上所述,计算机体系结构定义为:计算机体系结构是程序员所看到的计算机的属性,即概念性结构与功能特性,是计算机软硬件的交界面。

这里比较全面地讨论了经典的计算机体系结构概念。随着计算机技术的发展,计算机体系结构所包含的内容是不断变化和发展的。目前经常使用的是广义的计算机体系结构概念,它既包括经典的计算机体系结构的概念范畴,也包括对计算机组成和计算机实现技术的研究。

1.1.3 系统结构、计算机组成和计算机实现的关系

计算机组成的任务是在计算机系统结构确定分配给硬件子系统的功能及其概念结构之后,研究各组成部分的内部构造和相互联系,以实现机器指令级的各种功能和特性。这种相互联系包括各功能部件的配置、相互连接和相互作用。各功能部件的性能参数相互匹配,是计算机组成合理的重要标志,因而相应的就有许多计算机组织方法。例如,为了使存储器的容量大、速度快,人们研究出层次存储系统和虚拟存储技术。在层次存储系统中,又有高速缓存、多模块交叉工作、多寄存器组和堆栈等技术。为了使输入输出设备与处理机间的信息流量达到平衡,人们研究出通道、外围处理机等方式。为了提高处理机速度,人们研究出先行控制、流水线、多执行部件等方式。在各功能部件的内部结构研究方面,产生了许多组合逻辑、时序逻辑的高效设计方法和结构。例如,在运算器方面,出现了多种自动调度算法和结构等。

由此可见,计算机组成是计算机系统结构的逻辑实现,包括机器内部的数据流和控制流的组成以及逻辑设计等。计算机组成的设计是按所希望达到的性能价格比,最佳、最合理地把各种设备和部件组成计算机,以实现所确定的计算机系统结构。一般计算机组成设计包括数据通路宽度的确定、各种操作对功能部件的共享程度的确定、专用功能部件的确定、功能部件的并行性确定、缓冲器和排队的确定、控制机构的设计、可靠性技术的确定等。对

传统机器程序员来说,计算机组成的设计内容一般是透明的。

计算机实现是指计算机组成的物理实现。它包括处理机、主存等部件的物理结构,器件的集成度和速度,信号传输,器件、模块、插件、底板的划分与连接,专用器件的设计,电源、冷却、装配等技术以及有关的制造技术和工艺等。

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。系统结构是计算机系统的软、硬件的界面;计算机组成是计算机系统结构的逻辑实现;计算机实现是计算机组成的物理实现。它们各自包含不同的内容,但又有紧密的关系。

我们还应看到系统结构、组成和实现所包含的具体内容是随不同机器而变化的。有些计算机系统是作为系统结构的内容,其他计算机系统可能是作为组成和实现的内容。开始是作为组成和实现提出来的设计思想,到后来就可能被引入系统结构中。例如高速缓冲存储器一般是作为组成提出来的,其中存放的信息全部由硬件自动管理,对程序员来说是透明的。然而,有的机器为了提高其使用效率,设置了高速缓冲存储器的管理指令,使程序员能参与高速缓冲存储器的管理。

1.1.4 计算机体系结构分类

计算机系统结构可以按照应用领域不同而分类。一般来说,计算机都是按照通用系统进行设计的,但用户在编写应用程序的时候,都带有专用性质。为解决这个问题,一般采用下列方法:灵活改变系统配置,按照特殊环境的要求采用不同的物理安装,提供多种不同的操作系统,以适应批处理、分时、实时等不同需要。按用途分类可分为科学计算、事物处理、实时控制、家用等计算机。

按处理机的个数和种类分,计算机系统又可以分为单处理机、多处理机、关联处理机、超标量处理机、超流水线处理机、多机等。

下面介绍三种常用的分类方法。

1.1.4.1 Flynn 分类法 (按指令流和数据流的不同组织方式分类)

指令流 (instruction stream) —— 机器执行的指令序列。

数据流 (data stream) —— 由指令流调动的数据序列,包括输入数据和中间结果。

多倍性 (multiplicity) —— 在系统最受限制的部件上处于同一执行阶段的指令或数据的最大可能个数。

按照指令流和数据流不同的组织方式,计算机系统可分为以下四类 (如

图 1-2 所示)。

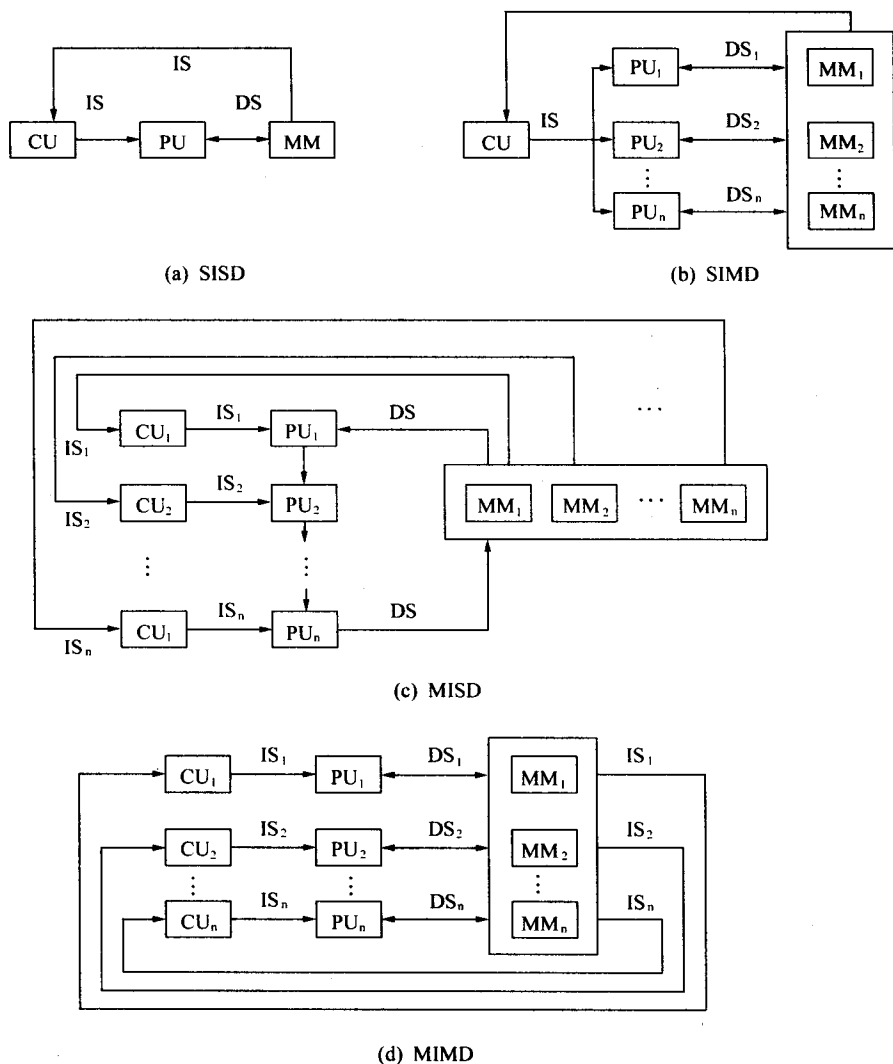


图 1-2 Flynn 分类法各类机器结构示意图

CU. 控制部件; PU. 处理部件; MM. 存储模块器; IS. 指令流; DS. 数据流

(1) 单指令流单数据流 SISD——传统的顺序处理机（串行机）传统的单处理机。

(2) 多指令流单数据流 SIMD——阵列处理机或并行处理机。

(3) 单指令流多数据流 MISD——采用流水结构的计算机。

(4) 多指令流多数据流 MIMD——多处理机。

1.1.4.2 冯氏分类法 (按最大并行度进行分类)

冯泽云于 1972 年提出用最大并行度对计算机系统结构进行分类。最大并行度 P_m 定义为: 计算机系统在单位时间内能够处理的最大的二进制位数。假定每个时钟周期 Δt_i 内能同时处理的二进制数为 P_i , 则 T 个时钟周期内平均并行度为

$$P_a = \frac{\sum_{i=1}^T P_i \Delta t_i}{T}$$

平均并行度不同于最大并行度, 它取决于系统的运用程度, 与应用程序有关。因此, 定义系统在周期 T 内的平均利用率为

$$\mu = \frac{P_a}{P_m} = \frac{\sum_{i=1}^T P_i \Delta t_i}{TP_m}$$

图 1-3 给出用最大并行度对计算机系统结构进行分类的方法。用平面

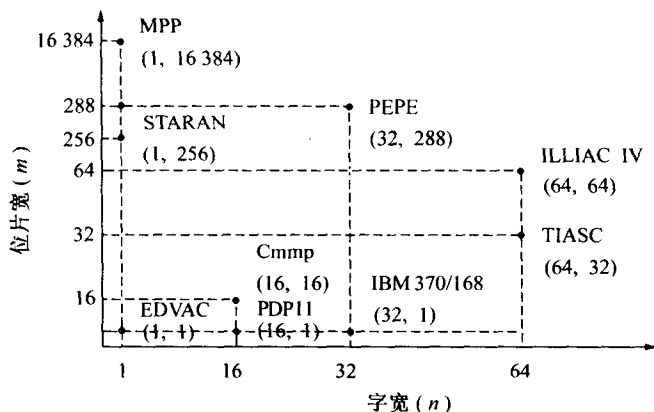


图 1-3 按最大并行度的冯氏分类法

直角坐标系中的一点代表一个计算机系统, 横坐标代表字宽 (n 位), 即在一个字中同时处理的二进位的位数; 纵坐标代表位片宽度 (m 位), 即在一个位片中能同时处理的字数。于是, 一个系统的最大并行度就可以用这两个量的乘积, 即用通过该点的水平线和垂直线与两坐标轴围成的矩形面积来表示。

由图 1-3 可得出四类不同处理方法的计算机系统结构:

(1) 字串位串 WSBS (word serial and bit serial), 其中 $n = 1$, $m = 1$ 。这是

第一代——计算机发展初期的纯串行计算机。

(2) 字并位串 WPBS (word parallel and bit serial), 其中, $n > 1$, $m = 1$ 。这是传统并行单处理机。

(3) 字串位并 WSBP (word serial and bit parallel), 其中, $n = 1$, $m > 1$ 。STARAN、MPP 属于这种结构。

(4) 字并位并 WPBP (word parallel and bit parallel), 其中, $n > 1$, $m > 1$ 。PEPE、ILLIAC IV、Cmmp 属于这种结构。

1.1.4.3 Handler 分类法 (根据并行度和流水线进行分类)

Wolfgan Handler 在 1977 年根据并行度和流水线提出了另一种分类法。这种分类方法把计算机的硬件结构分成三个层次, 并分别考虑它们的可并行度、流水处理程度。这三个层次为:

- (1) 程序控制部件 (PCU) 的个数 k ;
- (2) 算术逻辑部件 (ALU) 或处理部件 (PE) 的个数 d ;
- (3) 每个算术逻辑部件包含基本逻辑线路 (ELC) 的套数 w 。

这样我们可以把一个计算机系统的结构用如下公式表示: t (系统型号) = (k, d, w) 。为了进一步揭示流水线的特殊性, 一个计算机系统的结构可用如下公式表示:

$$t \text{ (系统型号)} = (k \times k', d \times d', w \times w')$$

式中: k' ——宏流水线中程序控制部件的个数;

d' ——指令流水线中算术逻辑部件的个数;

w' ——操作流水线中基本逻辑线路的套数。

例如: Cray1 有 1 个 CPU, 12 个相当于 ALU 或 PE 的处理部件, 可以最多实现 8 级流水线。字长为 64 位, 可以实现 1~14 位流水线处理。所以 Cray1 的系统结构可表示为

$$t \text{ (Cray1)} = (1, 12 \times 8, 64 (1 \sim 14))$$

下面是用这种分类法的例子:

$$t \text{ (PDP11)} = (1, 1, 16)$$

$$t \text{ (ILLIAC IV)} = (1, 64, 64)$$

$$t \text{ (STARAN)} = (1, 8192, 1)$$

$$t \text{ (Cmmp)} = (16, 1, 16)$$

$$t \text{ (PEPE)} = (1 \times 3, 288, 32)$$

$$t \text{ (TIASC)} = (1, 4, 64 \times 8)$$

1.2 计算机体系结构的发展

计算机体系结构研究计算机系统中软、硬件的界面，即研究哪些功能由软件完成，哪些功能由硬件完成。实际上，软硬件的功能在逻辑上是等效的，即绝大多数硬件的功能都可用软件来实现；反之亦然。但是显然，软件和硬件在性能和成本上是不等效的。因此，计算机系统结构就是在保证系统功能的基础上，合理分配软硬件资源，充分利用硬件和器件技术的发展，使系统达到较高的性价比。

计算机系统结构在存储程序计算机体系结构的研究上不断取得新成果，并且由于技术和应用的需要，人们不断探索和实践了各种非存储程序计算机体系结构。

1.2.1 存储程序计算机体系结构及其发展

最早存储程序计算机是由出生于匈牙利的美国数学家的冯·诺依曼等人在 1946 年总结并明确提出来的，因此人们常称其为冯·诺依曼结构计算机，其结构如图 1-4 所示。

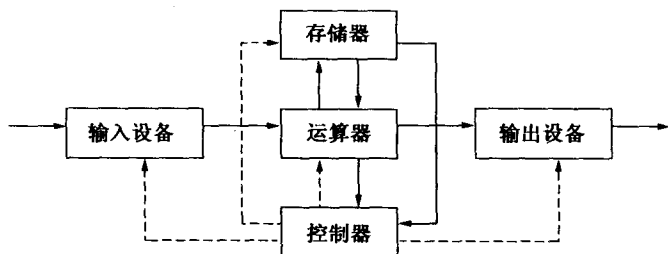


图 1-4 存储程序计算机的结构

存储程序计算机在体系结构上的主要特点如下：

(1) 机器以运算器为中心。输入/输出设备与存储器之间的数据传送都经过运算器，存储器、输入/输出设备的操作以及它们之间的联系都由控制器集中控制。

(2) 采用存储程序原理。程序（指令）和数据放在同一存储器中，并且没有对两者加以区分。指令和数据一样可以送到运算器进行运算，即由指令组成的程序自身是可以修改的。

(3) 存储器是按地址访问的、线性编址的空间。每个单元的位数是相同且固定的，称为存储器编址单位。

(4) 控制流由指令流产生。指令在存储器中按其执行顺序存储, 由程序计数器指明每条指令所在单元的地址。一般情况下每执行完一条指令, 程序计数器顺序递增。虽然执行顺序可以根据运算结果改变, 但是解题算法仍然是也只能是顺序型的。

(5) 指令由操作码和地址码组成。操作码指明本指令的操作类型, 地址码指明操作数和操作结果的地址。操作数的类型(定点、浮点或十进制数)由操作码决定, 操作数本身不能判定其数据类型。

(6) 数据以二进制编码表示, 采用二进制运算。它主要面向数值计算和数据处理。

随着计算机应用领域的扩大, 高级语言和操作系统的出现, 这种功能分配的状况引起了愈来愈多的矛盾, 迫使人们不断地对这种体系结构进行改进。下面结合上述 6 个特点进行简要地分析。

1.2.1.1 分布的 I/O 处理能力

存储程序计算机以运算器为中心, 所有部件的操作都由控制器集中控制, 这一特点带来了慢速输入/输出操作占用快速运算器的问题。此时的输入/输出操作和运算操作只能串行, 互相等待, 大大影响了运算器效率的发挥。为了克服这一缺点, 人们提出各种输入/输出方式, 如图 1-5 所示, 从而使计算机概念性结构发生了相应的变化。

在程序等待方式之后, 很快出现了程序中中断的概念, 并被应用于输入/输出操作。程序中中断概念的引入可以使 CPU 与外部设备在一定程度上并行工作, 提高了计算机的效率, 并且可以实现多种外部设备同时工作。中断技术已经成为现代计算机操作系统的技术基础。

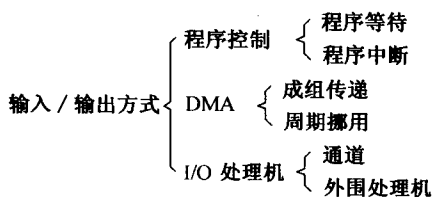


图 1-5 各种输入/输出方式

对于成块(组)地进行传送的输入/输出信息, 出现了所谓 DMA(直接存储器访问)方式。为了实现这种方式, 需要在主存和设备之间增加 DMA 控制器(数据通道), 从而形成了以主存为中心的结构。

采用 DMA 方式, 每传送完一组数据就要中断 CPU 一次。如果该部件能自己控制完成输入/输出的大部分工作, 则可使 CPU 进一步摆脱用于管理、控制 I/O 系统的沉重负担, 这就出现了 I/O 处理机方式。I/O 处理机几乎把控制输入/输出操作和信息传送的所有功能都从 CPU 那里接管过来。