



普通高等教育“十一五”国家级规划教材教学参考书

高等学校计算机科学与技术系列教材

计算机系统结构学习指导与题解

张晨曦 刘 依 沈 立 孙太一 李江峰



高等教育出版社
Higher Education Press

普通高等教育“十一五”国家级规划教材教学参考书

高等学校计算机科学与技术系列教材

计算机系统结构 学习指导与题解

Jisuanji Xitong Jiegou Xuexi Zhidao yu Tijie



高等教育出版社·北京
HIGHER EDUCATION PRESS BEIJING

ISBN 7-03-01992-1/TP
定价：14.75元
730.000

内容简介

本书是一本关于计算机系统结构课程的学习辅导书。全书共分为9章:计算机系统结构的基本概念,计算机指令集结构,流水线技术,指令级并行,存储层次,输入/输出系统,互连网络,多处理机,机群计算机。每一章都由4节组成,分别是基本要求与难点、知识要点、习题以及题解。“基本要求与难点”列出了各章教学的基本要求及难点,“知识要点”详细给出了各章内容的精华和要点。“习题”包括概念题、选择题、填空题、问答题、应用题等。“题解”给出了各题答案和详细的求解过程。

本书概念清晰,重点难点突出,题型多样,是一本很有用的学习辅导书。本书是普通高等教育“十一五”国家级规划教材《计算机系统结构》(高等教育出版社)的配套教学参考书。当然,对于采用其他系统结构教材的读者来说,也有较高的参考价值。

本书可作为计算机系统结构课程(上课或自学)的学习参考书,也可作为计算机专业硕士研究生入学考试的复习指导书。

图书在版编目(CIP)数据

计算机系统结构学习指导与题解 / 张晨曦等编. —

北京:高等教育出版社,2010.5

ISBN 978-7-04-029575-7

I. ①计… II. ①张… III. ①计算机体系结构—高等学校—教学参考资料 IV. ①TP303

中国版本图书馆 CIP 数据核字 (2010) 第 055462 号

出版发行 高等教育出版社
社 址 北京市西城区德外大街4号
邮政编码 100120

经 销 蓝色畅想图书发行有限公司
印 刷 国防工业出版社印刷厂

开 本 787×1092 1/16
印 张 14.75
字 数 330 000

购书热线 010-58581118
咨询电话 400-810-0598
网 址 <http://www.hep.edu.cn>
<http://www.hep.com.cn>
网上订购 <http://www.landaco.com>
<http://www.landaco.com.cn>
畅想教育 <http://www.widedu.com>

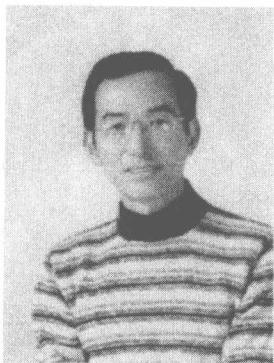
版 次 2010年5月第1版
印 次 2010年5月第1次印刷
定 价 23.50元

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换。

版权所有 侵权必究

物料号 29575-00

第一作者简介



张晨曦,男,1960年9月生,汉族,福建龙岩人。现任同济大学软件学院教授,博士生导师。国家级“中青年有突出贡献专家”,国家杰出青年基金获得者。先后主持了4项国家自然科学基金项目。1988年获博士学位,后一直在国防科技大学计算机学院工作,2005年9月调入同济大学。

作为课程负责人,张晨曦建设的“计算机系统结构”课程于2007年被评为上海市精品课程,2008年被评为国家级精品课程。他还先后获得了“教育部-微软精品课程”和“教育部-SUN精品课程”。他主讲“计算机系统结构”课程和从事系统结构的研究20余年,进行了一系列的教学改革和课程建设,取得了突出的成绩。1992年开发出了国内第一套计算机系统结构CAI课件(含30个动画),在清华大学、北京大学等全国10多所高校获得应用。2003年完成教育部的新世纪网络课程建设工程项目“计算机体系结构网络课程”。2009年开发出了国内第一套计算机系统结构实验模拟器。

他负责编写的《计算机系统结构》(高等教育出版社)是普通高等教育“十五”和“十一五”国家级规划教材,该教材2002年获全国普通高等学校优秀教材二等奖。全国至少有60所大学采用了该教材。至2009年,他编写出版了五本“十一五”国家级规划教材。撰写专著两部(第二作者),其中专著《新一代计算机》由荷兰North-Holland出版社出版,另一部于1992年获“国家教委优秀专著特等奖”,1993年获“全国优秀科技图书一等奖”。发表学术论文90多篇,其中在《中国科学》、《计算机学报》等一级刊物上发表8篇,国外发表20多篇。有20多篇被国际著名八大检索工具收录。

张晨曦获部委级科技进步一等奖两项(排名第二),二等奖一项(排名第一);获部委级教学成果二、三等奖各一项。

2007年获宝钢优秀教师奖和上海市育才奖,2008年获上海高校教学名师奖。1991年被国家教委授予“做出突出贡献的中国博士”光荣称号,被评为湖南省科技青年“十佳”之一;1993年被评为“全军优秀教师”,1993年和1995年两次获“霍英东青年教师奖”;1995年获第4届“中国青年科技奖”。从15岁起当中学教师,对教学方法和现代教育技术有深入的研究,提出了面向远程教育和CAI的动画解析教学法。

业余爱好:摄影

通信地址:上海市曹安公路4800号同济大学软件学院

邮编:201804

前 言

计算机系统结构是计算机专业及相关专业的一门重要的专业课程。本书是专门为该课程编写的辅导书,既可作为该课程的学习参考书,也可作为自学该课程的辅导书,还可作为计算机专业硕士研究生入学考试的复习指导书。

全书共包括9章。

第1章讲述计算机系统结构的基本概念以及系统结构和并行性的发展,并介绍定量分析基础。

第2章讲述计算机指令集结构设计中的一些问题,包括寻址技术、指令集的功能设计、操作数的类型和大小、指令格式的设计,并且介绍 RISC 技术以及一个指令集结构的实例——MIPS。

第3章为“流水线技术”,讲述流水线的基本概念、分类和性能分析,介绍流水线中的相关和冲突问题及其解决方法等,并对向量处理机的结构及特点、关键技术以及性能评价进行讨论。

第4章为“指令级并行”,讲述利用硬、软件技术开发程序中指令间并行性的技术和方法,包括指令动态调度、指令分支预测、超标量技术、超流水技术以及超长指令字技术等。

第5章为“存储层次”,讲述 Cache 的基本知识、降低 Cache 失效率的方法、减少 Cache 失效开销的方法以及减少命中时间的方法,并对虚拟存储器做简要的讨论。

第6章为“输入/输出系统”,讲述总线及其与 CPU/存储器的连接、通道处理机及其流量分析、廉价磁盘冗余阵列 RAID。

第7章为“互连网络”,讲述互连函数、互连网络的特性参数、静态互连网络以及动态互连网络等。

第8章为“多处理机”,讲述对称式共享存储器系统结构、分布式共享存储器系统结构、多处理机 Cache 一致性、同步以及同时多线程技术,并介绍了一个多处理机实例。

第9章为“机群计算机”,讲述机群的结构、软件模型以及机群的分类,并介绍了典型的机群系统。

每一章都由4节组成,分别是基本要求与难点、知识要点、习题以及题解。“基本要点与难点”列出了各章教学的基本要求和难点,“知识要点”详细给出了各章内容的精华和要点。“习题”包括概念题、选择题、填空题、问答题、应用题等。“题解”给出了答案和详细的求解过程。

本书主要由张晨曦、刘依、沈立、孙太一、李江峰编写,张硕、程志强、王伟、江崧等也编写了部分内容。

本书概念清晰,重点难点突出,题型多样,是一本很有用的学习辅导书。本书是普通高等教育“十一五”国家级规划教材《计算机系统结构》(高等教育出版社)的配套教学参考书。当然,对于采用其他系统结构教材的读者来说,也有较高的参考价值。

关于新的信息,请关注本网站: www.GotoSchool.net。

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,请把相关的信息发送到 xzhang2000@sohu.com。

张晨曦

2009年11月于上海

目 录

第 1 章 计算机系统结构的基本概念 1

1.1 基本要求与难点 1

1.1.1 基本要求 1

1.1.2 难点 1

1.2 知识要点 2

1.2.1 计算机系统结构的概念 2

1.2.2 定量分析技术 3

1.2.3 计算机系统结构的发展 6

1.2.4 计算机系统结构中并行性的 发展 8

1.3 习题 10

1.3.1 概念题 10

1.3.2 选择题 10

1.3.3 填空题 11

1.3.4 问答题 11

1.3.5 应用题 12

1.4 题解 14

1.4.1 概念题 14

1.4.2 选择题 16

1.4.3 填空题 16

1.4.4 问答题 16

1.4.5 应用题 18

第 2 章 计算机指令集结构 23

2.1 基本要求与难点 23

2.1.1 基本要求 23

2.1.2 难点 23

2.2 知识要点 23

2.2.1 指令集结构的分类 23

2.2.2 寻址方式 25

2.2.3 指令集结构的功能设计 25

2.2.4 操作数的类型和大小 28

2.2.5 指令格式的设计 28

2.2.6 MIPS 指令集结构 29

2.3 习题 30

2.3.1 概念题 30

2.3.2 选择题 31

2.3.3 填空题 31

2.3.4 问答题 31

2.4 题解 32

2.4.1 概念题 32

2.4.2 选择题 32

2.4.3 填空题 33

2.4.4 问答题 33

第 3 章 流水线技术 36

3.1 基本要求与难点 36

3.1.1 基本要求 36

3.1.2 难点 36

3.2 知识要点 36

3.2.1 重叠执行和先行控制 36

3.2.2 流水线的基本概念 39

3.2.3 流水线的性能指标 40

3.2.4 流水线的相关与冲突 42

3.2.5 流水线的实现 47

3.2.6 向量处理机 50

3.3 习题 54

3.3.1 概念题 54

3.3.2 选择题 55

3.3.3 填空题 56

3.3.4 问答题 57

3.3.5 应用题 57

3.4 题解	62	5.2.5 减少命中时间	133
3.4.1 概念题	62	5.2.6 主存	136
3.4.2 选择题	64	5.2.7 虚拟存储器	139
3.4.3 填空题	64	5.2.8 进程保护和虚存实例	140
3.4.4 问答题	65	5.2.9 Alpha AXP 21064 存储层次	140
3.4.5 应用题	67	5.3 习题	140
第4章 指令级并行	79	5.3.1 概念题	140
4.1 基本要求与难点	79	5.3.2 选择题	140
4.1.1 基本要求	79	5.3.3 填空题	141
4.1.2 难点	79	5.3.4 问答题	142
4.2 知识要点	80	5.3.5 应用题	143
4.2.1 指令级并行的概念	80	5.4 题解	145
4.2.2 指令的动态调度	81	5.4.1 概念题	145
4.2.3 动态分支预测技术	86	5.4.2 选择题	146
4.2.4 多指令流出技术	89	5.4.3 填空题	146
4.2.5 循环展开和指令调度	92	5.4.4 问答题	147
4.3 习题	98	5.4.5 应用题	150
4.3.1 概念题	98	第6章 输入/输出系统	155
4.3.2 填空题	98	6.1 基本要求与难点	155
4.3.3 问答题	98	6.1.1 基本要求	155
4.3.4 应用题	99	6.1.2 难点	155
4.4 题解	103	6.2 知识要点	155
4.4.1 概念题	103	6.2.1 I/O 系统性能与 CPU 性能	155
4.4.2 填空题	104	6.2.2 I/O 系统的可靠性、可用性和可信性	156
4.4.3 问答题	104	6.2.3 廉价磁盘冗余阵列 RAID	156
4.4.4 应用题	107	6.2.4 总线	160
第5章 存储层次	120	6.2.5 通道处理机	161
5.1 基本要求与难点	120	6.2.6 I/O 与操作系统	166
5.1.1 基本要求	120	6.3 习题	167
5.1.2 难点	120	6.3.1 概念题	167
5.2 知识要点	121	6.3.2 选择题	167
5.2.1 存储器的层次结构	121	6.3.3 填空题	167
5.2.2 Cache 基本知识	123	6.3.4 问答题	168
5.2.3 降低 Cache 失效率	128		
5.2.4 减少 Cache 失效开销	131		

6.3.5 应用题	168	结构	197
6.4 题解	170	8.2.3 分布式共享存储器系统 结构	201
6.4.1 概念题	170	8.2.4 同步	203
6.4.2 选择题	171	8.2.5 同时多线程	207
6.4.3 填空题	171	8.2.6 多处理机实例	210
6.4.4 问答题	172	8.3 习题	210
6.4.5 应用题	174	8.3.1 概念题	210
第7章 互连网络	178	8.3.2 填空题	210
7.1 基本要求与难点	178	8.3.3 问答题	211
7.1.1 基本要求	178	8.3.4 应用题	211
7.1.2 难点	178	8.4 题解	212
7.2 知识要点	178	8.4.1 概念题	212
7.2.1 互连网络的基本概念	178	8.4.2 填空题	213
7.2.2 互连网络的结构	181	8.4.3 问答题	213
7.3 习题	184	8.4.4 应用题	215
7.3.1 概念题	184	第9章 机群计算机	218
7.3.2 选择题	184	9.1 基本要求与难点	218
7.3.3 填空题	185	9.1.1 基本要求	218
7.3.4 问答题	186	9.1.2 难点	218
7.3.5 应用题	186	9.2 知识要点	218
7.4 题解	187	9.2.1 机群的基本概念和结构	218
7.4.1 概念题	187	9.2.2 机群的特点	220
7.4.2 选择题	188	9.2.3 机群的分类	221
7.4.3 填空题	188	9.2.4 典型机群系统简介	222
7.4.4 问答题	188	9.3 习题	222
7.4.5 应用题	189	9.3.1 概念题	222
第8章 多处理机	194	9.3.2 问答题	222
8.1 基本要求与难点	194	9.4 题解	222
8.1.1 基本要求	194	9.4.1 概念题	222
8.1.2 难点	194	9.4.2 问答题	223
8.2 知识要点	195	参考文献	225
8.2.1 引言	195		
8.2.2 对称式共享存储器系统			

第1章 计算机系统结构的基本概念

1.1 基本要求与难点

1.1.1 基本要求

- ① 掌握计算机系统结构的基本概念,如计算机系统的多级层次结构、虚拟机、计算机系统结构的定义、透明性、系列机等。
- ② 理解计算机系统的多级层次结构。
- ③ 掌握计算机系统结构、计算机组成和计算机实现的定义,了解其关系。
- ④ 了解计算机系统结构的分类方法。
- ⑤ 掌握计算机系统设计的定量原理,能熟练应用 Amdahl 定律和 CPU 性能公式进行定量分析。
- ⑥ 理解进行计算机系统性能评测的基准测试程序法,了解如何进行计算机系统性能的比较。
- ⑦ 理解冯·诺依曼结构的特点及其改进。
- ⑧ 掌握实现软件移植的 3 种方法,理解系列机的概念及其根本特征。
- ⑨ 理解并行性的定义以及并行性的不同等级,掌握提高并行性的 3 种技术途径。
- ⑩ 了解单机系统和多机系统中并行性的发展。

1.1.2 难点

- ① 计算机系统结构、计算机组成和计算机实现 3 者的定义与关系。如何判断某种技术是属于哪个方面的。如何判断是否为透明的。
- ② Amdahl 定律和 CPU 性能公式,如何应用它们进行定量分析。
- ③ 并行性及其发展。

1.2 知识要点

1.2.1 计算机系统结构的概念

1. 计算机系统的层次结构

可以把计算机系统按功能划分成**多级层次结构**:微程序机器级,机器语言(传统机器级),操作系统虚拟机,汇编语言虚拟机,高级语言虚拟机,应用语言虚拟机等。在这6级层次中,最下面的两级是用硬件/固件实现的,称为**物理机**。而上面4级一般是由软件实现的。用软件实现的机器称为**虚拟机**。

各虚拟机的实现主要靠翻译或解释,或两者的结合。一般来说,上述6级层次中的下面3级用解释方法实现,而上面3级则用翻译方法实现。**翻译**是先用转换程序把高一级机器上的程序转换为低一级机器上等效的程序,然后在这低一级机器上运行,实现程序的功能。**解释**则是对于高一级机器上的程序中的每一条语句或指令,都转去执行低一级机器上的一段等效程序。执行完后,再回到高一级机器取下一条语句或指令,再进行解释执行,如此反复,直到解释执行完整个程序。

2. 计算机系统结构的定义

计算机系统结构(computer architecture)的经典定义是1964年Amdahl在介绍IBM 360系统时提出的:计算机系统结构是指机器语言程序员所看到的计算机属性,即概念性结构与功能特性。计算机系统结构的实质是确定计算机系统中软硬件的界面,界面之上是软件实现的功能,界面之下是硬件和固件实现的功能。

在J. L. Hennessy和D. A. Patterson编写的Computer Architecture: A Quantitative Approach一书中,把系统结构定义为囊括计算机设计的3个方面:指令系统结构,组成,硬件。我们不妨将之理解为广义的系统结构定义。

3. 计算机组成和计算机实现

计算机组成指的是计算机系统结构的逻辑实现,包含物理机器级中的数据流和控制流的组成以及逻辑设计等。它着眼于物理机器级内各事件的排序方式与控制方式、各部件的功能以及各部件之间的联系。

计算机实现指的是计算机组成的物理实现,包括处理机、主存等部件的物理结构,器件的集成度和速度,模块、插件、底板的划分与连接,信号传输,电源、冷却及整机装配技术等。它着眼于器件技术和微组装技术,其中器件技术在实现技术中起主导作用。

传统机器级所存在的差别对于高级语言程序员来讲是“看不见”的。在计算机技术中,把这种本来存在的事物或属性,但从某种角度看又好像不存在的特性称为**透明性**。

具有相同系统结构的计算机因为速度、价格等方面要求的不同,可以采用不同的计算机组

成。而同一种计算机组成又可以采用多种不同的计算机实现。系列机的出现充分反映了这一点。系列机是指由同一厂家生产的具有相同系统结构、但具有不同组成和实现的一系列不同型号的计算机。

4. 计算机系统结构的分类

(1) Flynn 分类法

Flynn 分类法是按照指令流和数据流的多倍性进行分类的。

指令流：计算机执行的指令序列。

数据流：由指令流调用的数据序列。

多倍性：在系统最受限的部件上，同时处于同一执行阶段的指令或数据的最大数目。

Flynn 分类法把计算机系统的结构分为以下 4 类：

- ① 单指令流单数据流 (Single Instruction stream Single Data stream, SISD)。
- ② 单指令流多数据流 (Single Instruction stream Multiple Data stream, SIMD)。
- ③ 多指令流单数据流 (Multiple Instruction stream Single Data stream, MISD)。
- ④ 多指令流多数据流 (Multiple Instruction stream Multiple Data stream, MIMD)。

SISD 是传统的顺序处理计算机。SIMD 以阵列处理机为代表。MISD 只是一种人为的划分，目前没有实际成形的机器。多处理机属于 MIMD 结构。

(2) 冯氏分类法

冯氏分类法是以系统的最大并行度为依据对计算机进行分类的。最大并行度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。

按照这种分类法，可以把计算机分成以下 4 类具有不同最大并行度的计算机系统结构：字串位串，字串位并，字并位串，字并位并。

1.2.2 定量分析技术

1. 计算机系统设计的定量原理

(1) 以经常性事件为重点

这是计算机设计中最重要、应用最广泛的设计原则。它是指对经常发生的情况采用优化方法的原则，因为这样能得到更多总体上的改进。这里优化是指分配更多的资源、达到更高的性能等。

(2) Amdahl 定律

Amdahl 定律指出，加快某部件执行速度所能获得的系统性能加速比，受限于该部件的执行时间占系统中总执行时间的百分比。它可以用来具体地计算当对计算机系统中的一个部分进行改进后，能获得多少总体性能上的提高。

假设对某台计算机进行了某种改进，那么所获得的加速比为

$$\text{加速比} = \frac{\text{系统性能}_{\text{改进后}}}{\text{系统性能}_{\text{改进前}}} = \frac{\text{总执行时间}_{\text{改进前}}}{\text{总执行时间}_{\text{改进后}}}$$

这个加速比依赖于下面两个因素:

① 在改进前的系统中,可改进部分的执行时间在总的执行时间中所占的比例,称其为“可改进比例”,它总是小于等于1的。

② 可改进部分改进以后性能提高的倍数,称其为“部件加速比”。一般情况下,它总是大于1的。

改进后程序的总执行时间为

$$\text{总执行时间}_{\text{改进后}} = \text{总执行时间}_{\text{改进前}} \times \left[(1 - \text{可改进比例}) + \frac{\text{可改进比例}}{\text{部件加速比}} \right]$$

系统加速比为

$$\text{加速比} = \frac{\text{总执行时间}_{\text{改进前}}}{\text{总执行时间}_{\text{改进后}}} = \frac{1}{(1 - \text{可改进比例}) + \frac{\text{可改进比例}}{\text{部件加速比}}}$$

(3) CPU 性能公式

执行一个程序所需的 CPU 时间可以按下式计算

$$\text{CPU 时间} = \text{IC} \times \text{CPI} \times \text{时钟周期时间}$$

其中,IC 为所执行的指令条数。**CPI**(Cycles Per Instruction)为每条指令的平均时钟周期数,它可以根据下式计算得出

$$\text{CPI} = \text{执行程序所需的时钟周期数} / \text{所执行的指令条数}$$

CPU 的性能取决于以下 3 个参数:

① 时钟周期时间:取决于硬件实现技术和计算机组成。

② CPI:取决于计算机组成和指令系统的结构。

③ IC:取决于指令系统的结构和编译技术。

CPU 设计中还经常用到下面计算 CPU 时钟周期数的方法

$$\text{CPU 时钟周期数} = \sum_{i=1}^n (\text{CPI}_i \times \text{IC}_i)$$

其中, IC_i 为程序执行的过程中第 i 种指令出现的次数, CPI_i 为执行第 i 种指令所需的平均时钟周期数, n 为指令的种数。这时的 CPU 性能公式为

$$\text{CPI} = \frac{\text{时钟周期数}}{\text{IC}} = \frac{\sum_{i=1}^n (\text{CPI}_i \times \text{IC}_i)}{\text{IC}} = \sum_{i=1}^n \left(\text{CPI}_i \times \frac{\text{IC}_i}{\text{IC}} \right)$$

$$\text{CPU 时间} = \text{CPU 时钟周期数} \times \text{时钟周期时间} = \sum_{i=1}^n (\text{CPI}_i \times \text{IC}_i) \times \text{时钟周期时间}$$

其中, $(\text{IC}_i / \text{IC})$ 反映了第 i 种指令在所执行的指令总数中所占的比例。

(4) 程序的局部性原理

程序的局部性原理是指程序执行时所访问的存储器地址不是随机分布的,而是相对地簇聚。现在常用的一个经验规则是,程序执行时间的 90% 都是在执行程序中 10% 的代码。

局部性包括时间局部性和空间局部性。**时间局部性**是指程序即将用到的信息很可能就是目前正在使用的信息。程序的**空间局部性**是指程序即将用到的信息很可能与目前正在使用的信息在空间上相邻或者临近。

2. 计算机系统的性能评测

(1) 执行时间和吞吐率

如何评测一台计算机的性能,与测试者看问题的角度有关。对于台式计算机来说,用户关心的是单个程序的执行时间,而对于数据处理中心的管理员来说,他所关心的则是吞吐率。

在比较不同的设计方案时,经常需要对两台计算机的性能进行比较。假设这两台计算机为 X 和 Y,通常“X 的性能是 Y 的 n 倍”是指:

$$n = \frac{\text{执行时间}_Y}{\text{执行时间}_X} = \frac{\frac{1}{\text{性能}_Y}}{\frac{1}{\text{性能}_X}} = \frac{\text{性能}_X}{\text{性能}_Y}$$

目前广泛采用的一致和可靠的性能评价方法,是使用真实程序的执行时间来衡量,人们常采用“CPU 时间”这个指标。它是指 CPU 执行给定程序所花费的时间,不包含 I/O 等待时间以及运行其他程序的时间。很明显,用户看到的执行时间是程序完成任务所花费的全部时间,而不仅仅是 CPU 时间。

CPU 时间还可细分为用户 CPU 时间及系统 CPU 时间,前者表示用户程序所耗费的 CPU 时间,后者表示用户程序运行期间操作系统耗费的 CPU 时间。

(2) 基准测试程序

用于测试和比较性能的基准测试程序的最佳选择是真实应用程序。

为了比较全面地反映计算机在各个方面的处理性能,通常采用整套测试程序进行测试。这组程序称为基准测试程序套件,它是由各种不同的真实应用程序构成的。以基准测试程序套件进行测试的目标是尽可能全面地反映两台计算机的相对性能。

目前最成功和最常见的测试程序套件是 SPEC 系列,它是由美国标准性能测试公司于 20 世纪 80 年代末开发的。台式计算机的基准测试程序套件可以分为两大类:处理器性能测试程序和图形性能测试程序。SPEC 最早开发的 SPEC89 主要用于测试处理器的性能。SPEC89 后来演化出了 4 个版本:SPEC92、SPEC95、SPEC2000 和 SPEC CPU2006。

(3) 性能比较

为了能更好地比较不同计算机的性能,可以采用以下两个指标。

① **总执行时间**: 直接用计算机执行所有测试程序的总时间进行比较。也可以采用平均执行时间来代替总执行时间。平均执行时间是各测试程序执行时间的算术平均值,即

$$S_m = \frac{1}{n} \sum_{i=1}^n T_i$$

其中, T_i 是第 i 个测试程序的执行时间, n 是测试程序组中程序的个数。

② 加权执行时间: 如果各程序在测试程序组中所占的比重不同, 就可以用加权执行时间来比较。加权执行时间是各测试程序执行时间的加权平均值, 即

$$A_m = \sum_{i=1}^n W_i T_i \quad (1.1)$$

其中, W_i 是第 i 个测试程序在测试程序组中所占的比重, $\sum_{i=1}^n W_i = 1$ 。 T_i 是该程序的执行时间。

1.2.3 计算机系统结构的发展

1. 冯·诺依曼结构及其改进

最早的存储程序式计算机是美国数学家冯·诺依曼(von Neumann)等人于1946年总结并提出来的, 它由运算器、控制器、存储器、输入设备和输出设备5部分构成。人们经常称之为冯·诺依曼结构计算机。虽然与冯·诺依曼结构相比, 现代的计算机系统结构已经发生了很大变化, 但就其结构原理来说, 占主流地位的仍是改进了的冯·诺依曼结构计算机。

冯·诺依曼结构的主要特点如下:

- ① 计算机以运算器为中心, 采用集中控制。
- ② 在存储器中, 指令和数据同等对待。
- ③ 存储器是按地址访问、按顺序线性编址的一维结构。
- ④ 指令是顺序执行的, 即一般是按照指令在存储器中存储的顺序来执行的。
- 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, 地址码指明操作数地址和存放运算结果的地址。操作数的类型由操作码决定。
- ⑥ 指令和数据均以二进制编码表示, 采用二进制运算。

后来的计算机针对冯·诺依曼结构的不足之处进行了不断的改进, 在系统结构方面有了很大的进展。主要包括以下几个方面。

(1) 对输入/输出方式的改进

人们先后提出了多种输入/输出方式, 包括程序中断、DMA、通道、外围处理机等。这是把越来越多的输入/输出管理工作从CPU中分离出来, “下放”给新设置的硬件去完成。

(2) 采用并行处理技术

在不同的级别采用并行处理技术, 例如微操作级、指令级、线程级、进程级、任务级等。先后出现了向量计算机、阵列处理机、多处理机、MPP(大规模并行处理机)等各种并行处理计算机。

(3) 存储器组织结构的发展

在CPU中设置通用寄存器组, 并在CPU和主存之间设置高速缓冲存储器(Cache)。

(4) 指令系统的发展

发展方向有两个,一个是朝复杂指令集计算机(Complex Instruction Set Computer, CISC)的方向发展,把越来越多的功能交由硬件实现。另一个方向是朝精简指令系统的方向发展,只设置使用频度高、功能简单的指令。1979年, D. A. Patterson 等人提出了精简指令集计算机(Reduced Instruction Set Computer, RISC)的思想。

2. 软件对系统结构的影响

软件对系统结构的影响是多方面的。下面只讨论系统结构设计中要注意解决的可移植性问题。可移植性是指一个软件可以不经修改或者只需少量修改就可以由一台计算机移植到另一台计算机上运行,差别只是执行时间不同。在这种情况下,人们称这两台计算机为软件兼容的。实现可移植性的常用方法有3种:系列机,模拟和仿真,统一高级语言。

(1) 系列机

系列机能较好地解决软件开发中要求系统结构相对稳定与器件、硬件技术迅速发展的矛盾。系列机的软件兼容有4种:向上兼容,向下兼容,向前兼容,向后兼容。向上(下)兼容指的是按某档计算机编制的程序,不加修改就能运行于比它高(低)档的计算机。向后(前)兼容是指按某个时期投入市场的某种型号计算机编制的程序,不加修改地就能运行于在它之后(前)投入市场的计算机。向后兼容是一定要保证的,它是系列机的根本特征。

兼容机是指由不同制造商生产的具有相同系统结构的计算机。

(2) 模拟和仿真

模拟和仿真是实现软件(二进制代码)可移植性的两种常用方法。

模拟是指用软件的方法在一台现有的计算机(称为宿主机, host)上实现另一台计算机(称为虚拟机)的指令系统。通常用解释的方法来实现模拟。除了模拟虚拟机的指令系统外,还要模拟其存储系统、I/O系统、操作系统等。这种方法的缺点是运行速度较慢,性能较差。

仿真是指用一台现有计算机(称为宿主机)上的微程序去解释实现另一台计算机(称为目标机)的指令系统。这个微程序称为仿真微程序。

仿真和模拟的主要区别在于解释执行所用的语言。仿真是用微程序解释执行,而模拟则是用机器语言程序解释执行。因此,仿真的运行速度比模拟方法快,但仿真只能在系统结构差距不大的计算机之间使用。为了取长补短,可以将这两种方法混合使用。

(3) 统一高级语言

如果各计算机能采用同一种高级语言,那么用这种语言编写的应用软件和系统软件的可移植问题就解决了。然而,到目前为止,还没有一种高级语言对各种应用是真正通用的。

3. 器件发展对系统结构的影响

摩尔定律指出,集成电路芯片上所集成的晶体管数目每隔18个月就翻一番。2003年以前CPU在速度上的提高也是如此,即主频每隔18个月翻一番。这使得越来越多的功能可以在一块芯片上实现,而且芯片的性能价格比也越来越高。