



全国高技术重点图书·计算机技术领域

现代 处理器的核心技术 与基本结构

● 李晓明 程 旭 编著



电子工业出版社

全国高技术重点图书·计算机技术领域

现代处理器的核心技术 与基本结构

李晓明 程旭 编著

电子工业出版社

(京)新登字 055 号

内 容 简 介

近十年来,围绕 RISC 技术的发展,在计算机系统结构领域出现了一种生机勃勃的局面。不仅形成了若干经得起市场考验的技术,而且在设计方法上也逐渐走向科学化、工程化。本书不仅综合地介绍了当前计算机处理器设计的主流技术和基本结构,包括 RISC、超级标量、超级流水线、VLIW、编译优化等;还强调了当代计算机设计的实验性研究方法。全书以基本技术为主线,注重突出基本要点和其发生的背景。通过介绍典型的处理器,展示具体实现特定技术的可能途径。本书内容充实、取材新颖,融合了许多计算机工作者和作者本人的最新研究成果。书中各章相对独立,便于读者有选择性地阅读。

本书可作为高等院校计算机专业高年级学生或研究生的教科书或参考书,也可供有关工程技术人员参考。

全国高技术重点图书 计算机技术领域
现代处理器的核心技术与基本结构

李晓明 程旭 编著

责任编辑:龚兰方

*

电子工业出版社出版

北京市海淀区万寿路 173 信箱(100036)

电子工业出版社发行 各地新华书店经销

新世纪印刷厂印刷

*

开本:850×1168 毫米 1/32 印张:8.375 字数:210 千字

1995 年 5 月第一版 1995 年 5 月北京第一次印刷

印数:2,000 册 定价:16.00 元

ISBN 7-5053-2865-4/TP·940

《全国高技术重点图书》出版指导委员会名单

主任： 朱丽兰

副主任： 刘 杲

卢鸣谷

委员： (以姓氏笔划为序)

王大中 王为珍 王守武 牛田佳 刘 仁

刘 杲 卢鸣谷 叶培大 朱丽兰 孙宝寅

师昌绪 任新民 杨牧之 杨嘉墀 陈芳允

陈能宽 罗见龙 周炳琨 欧阳莲 张钰珍

张效祥 赵忠贤 顾孝诚 谈德颜 龚 刚

梁祥丰

总干事： 罗见龙 梁祥丰

《全国高技术重点图书·计算机技术领域》 编审委员会

主任委员： 张效祥

委 员： 王鼎兴 刘帧权 刘锦德 李三立

何成武 徐培忠 梁祥丰 董韫美

作者简介

李晓明 1957年生,湖北沙市人。1982年本科毕业于哈尔滨工业大学计算机科学与工程系。1986年毕业于美国史蒂文斯理工学院计算机科学系,获博士学位。现为哈尔滨工业大学计算机系教授、博士生导师。近年来,除教学和研究生指导工作外,致力计算机系统结构的研究工作,尤其关心适合于大规模并行处理系统的微处理器结构的研究。他现在是中国计算机学会理事、国家教委计算机教学指导委员会委员,以及 ETA KAPPA NU 和 IEEE CS 会员。

程旭 1967年生,湖北襄樊人。1984~1994年就读于哈尔滨工业大学计算机科学与工程系,分别获得工学学士、硕士、博士学位。其间,曾任哈工大首届学生科协主席、校博士生会主席。从事过数据库、语音识别、专家系统、计算机系统结构等方面的研究工作。近年来,致力于指令级并行处理、优化编译和高级微处理器结构的研究。现在北京大学计算机科学与技术系从事博士后研究。

前 言

电子计算机已有了近五十年的历史。五十年,与诸如机械、电子、动力等领域的历史比起来,是微不足道的。但这五十年却给人类的文明和进步带来了巨大的变化,这种变化是其它技术领域在同样长的时期里望尘莫及的;同时,在这五十年里,计算机技术本身所历经的发展和变化也是其它学科在同样长时期内所不能比拟的。变化快是计算机技术发展迄今为止的一个基本特征。

处理器是计算机系统的核心。处理器的概念随着计算机技术的发展也在不断变化。本书所论及的处理器,泛指以 RISC 工作站为代表的现代通用计算机中的 CPU 部分,它包含指令部件、算逻部件、浮点部件,以及关系很密切的、常常集成在同一个芯片上的高速小容量存储部件——CACHE。当前广泛使用的处理器和我们传统的计算机原理教材中介绍的,在其结构和控制方式上,已有明显的区别。例如传统的算逻部件典型采用三寄存器结构,而现在几乎都用寄存器堆;通常的计算机原理书最多只介绍到指令的重叠,而现在的处理器几乎无一不采用指令流水...。另外,处理器的研究是面向实际的,因而,只有在了解了一些真实机器及其性能的定量分析结果的基础之上,我们才可能正确理解处理器设计中的许多基本原则和抽象概念;而且,在有关计算机的新名词、新术语纷呈的今天,我们只有做到“知其然,更知其所以然”才能把握它们的来龙去脉。鉴于此,我们感到应该有一本书,综合地介绍当前处理器设计的主流技术和结构,并使读者对这些技术和结构赖以成功的内在原因和外部动力有一定的认识。

全书共分十章。第一章从处理器设计技术变化的角度简单回顾了计算机的发展历史,以及人们从实践中认识到的计算机设计

的几个基本观念。我们看到,一些现在看来是理所当然的特征,如变址寄存器等,在当初也是有它的被认识过程的;而曾经被认为是很有用的想法,例如使指令能和数据一样被修改,随着历史的发展,也有其被淘汰的过程。我们以为,计算机专业工作者应该对这些演变有所了解,尤其应了解演变发生的技术背景。

RISC 是自从 80 年代以来计算机界的热门话题。RISC 不仅代表着一类计算机,它们的特征,所涉及的关键技术,而且还代表着一种设计哲学。第二章一般性地介绍有关的内容;其中,对处理器设计中一些问题的定量分析有助于消除人们对 RISC 的简单化理解和神秘感。RISC 技术的核心:指令流水和片载 CACHE,则分别安排在第 七、第八章专门论述。

第三、四、五章分别讨论超级标量、超长指令字、超级流水线的处理器结构。由于它们差不多都是在近十几年里发展起来的,一些内容尚未定论,现在仍然是前沿的研究和开发领域。因此我们主要介绍其基本概念和被公认的主要相关技术。

编译技术和体系结构技术密不可分。这种认识现在已被广泛接受。可以说没有编译技术的配合,RISC 就不会有现在这样的成就。从事处理器设计工作的人必须对编译技术,尤其是优化编译技术,有比较透彻的了解。出于这种考虑,我们在第六章介绍优化编译的一些主要策略。

正如计算机目前被广泛用作设计其它机器的辅助工具一样,应用计算机来研究计算机的设计和评价其性能也成为一种时髦且有效的手段。实验性研究是计算机设计步入工程化阶段的基础。因而,正确地进行模拟和性能评价是从事计算机系统设计所必须具备的专业技能。第九章介绍以踪迹驱动模拟为核心的有关内容。

第十章是我们最近的一点研究心得。通过对程序内部潜在指令级并行性的分析,考虑到转移指令在程序处理过程中不可回避的副作用,我们试图展示推测式执行与多控制流并行相结合对进一步开发指令级并行性的前景。

本书是作为哈尔滨工业大学计算机专业本科生选修课“现代计算技术”教材,经两遍教学试用,又经认真修改和充实后定稿出版。刘宏伟同志曾用过一次,崔光佐同志也详细阅读了原教材各章,他们都提出了一些宝贵意见。本书所涉及的研究工作直接受到国家教委资助优秀年轻教师资金项目“大规模并行系统中的微处理器结构”的支持。我们在此谨表示诚挚的谢意。在成稿中,作者力求概念清晰、内容完整。尽管如此,由于作者对现代处理器的概念悟得还不够深透,书中仍难免有不少错误和疏漏,谨盼读者不吝指正。

作 者

1994 年 8 月,哈尔滨

目 录

第一章 引论	1
1.1 历史回顾	1
1.1.1 通用电子计算机的起源	1
1.1.2 计算机产业的形成	3
1.1.3 计算机系统——软、硬件的融合与权衡	5
1.1.4 计算机普及、高级语言机器的尝试	8
1.1.5 计算机设计的工程化	10
1.2 计算机设计人员的任务.....	12
1.2.1 功能需求分析	12
1.2.2 软件和硬件的权衡	14
1.2.3 注意机器的生命力	16
1.3 性能和价格.....	17
1.3.1 CPU 性能	19
1.3.2 成本	21
1.4 计算机设计中的若干原则.....	22
1.4.1 大概率事件原则	23
1.4.2 Amdahl 定律	23
1.4.3 引用的局部性	25
1.5 评论.....	26
第二章 RISC 技术	28
2.1 概述.....	28
2.1.1 RISC 的基本原则和特点	30
2.1.2 RISC 的优势	32
2.2 指令系统的设计原理与分析.....	34
2.2.1 指令的分类	34
2.2.2 存储器寻址方式	37

2.2.3	指令系统中的操作类型	41
2.2.4	高级语言和编译的作用	42
2.3	处理器的基本实现技术	43
2.3.1	指令执行的一般过程	43
2.3.2	硬连线控制	45
2.3.3	中断处理的概念	46
2.4	寄存器技术	47
2.4.1	寄存器	48
2.4.2	寄存器窗口	51
2.4.3	寄存器的其它一些问题	53
2.5	评论	56
第三章	超级标量技术	58
3.1	基本概念	58
3.2	IBM 的 POWER 体系结构	60
3.2.1	转移处理器	62
3.2.2	定点处理器	70
3.2.3	浮点处理器	71
3.3	多功能部件并行和指令动态调度策略	72
3.3.1	多功能部件并行	72
3.3.2	动态分析、调度的基本问题	73
3.4	评论	78
第四章	超长指令字结构	80
4.1	基本概念	80
4.2	循环展开与过程嵌入	82
4.3	软件流水	84
4.4	路径调度技术	88
4.5	评论	91
第五章	超级流水线技术	94
5.1	基本概念	94
5.2	实际机器分析	97

5.2.1	MIPS R/4000——超级流水线处理器	97
5.2.2	DEC Alpha 结构——超级流水线和超级标量技术的 结合	106
5.3	评论	109
第六章	优化编译	110
6.1	概论	110
6.2	控制相关分析	112
6.2.1	基本块和控制流图	113
6.2.2	控制相关分析	115
6.3	数据相关分析	119
6.4	优化编译技术	127
6.4.1	局部优化策略	128
6.4.2	循环优化	129
6.4.3	全局优化策略	133
6.4.4	窥孔优化	135
6.4.5	寄存器分配	141
6.5	评论	143
第七章	流水线	145
7.1	流水线设计中的难点	145
7.1.1	流水线中的相关	145
7.1.2	中断处理	150
7.1.3	指令系统复杂性	154
7.2	转移指令的处理	157
7.2.1	程序中转移指令的情况	157
7.2.2	减小转移指令的不良影响	158
7.3	处理多周期操作的基本策略	165
7.4	流水线分级的考虑	171
7.4.1	三级流水线结构	172
7.4.2	四级指令流水线结构	174
7.4.3	五级指令流水线结构	176

7.5 记分板技术和 Tomasulo 算法	178
7.5.1 记分板技术	179
7.5.2 Tomasulo 算法	183
7.6 评论	189
第八章 高速缓冲存储器	191
8.1 存储体系的基本概念	191
8.2 CACHE 设计中的基本问题	194
8.2.1 映射方式	195
8.2.2 映射机构	196
8.2.3 替换策略	198
8.2.4 存储器写操作的处理	200
8.3 CACHE 的性能	202
8.4 CACHE 失效的原因	205
8.5 CACHE 设计中的一些结论	208
8.5.1 CACHE 中存储块大小的选择	208
8.5.2 指令和数据 CACHE 的分离与统一	210
8.6 评论	212
第九章 模拟和性能评价	214
9.1 概述	214
9.2 实验性研究的方式	215
9.2.1 在实际机器上研究	215
9.2.2 模拟方法	220
9.3 基准程序	223
9.4 评论	227
第十章 推测式执行与多控制流并行	232
10.1 概述	232
10.2 推测式执行和多控制流并行的基本概念	234
10.2.1 推测式执行	234
10.2.2 多控制流并行	236

10.3 不同控制相关处理策略下的指令级并行性.....	238
10.3.1 基本块内的潜在指令级并行性	239
10.3.2 推测式执行策略对开发指令并行性的影响	240
10.3.3 多控制流并行策略对开发指令级并行性的影响 ...	243
10.3.4 推测式执行与多控制流并行的结合是充分挖掘指令 级并行的关键	244
10.4 实现推测式执行策略中的一些问题.....	245
10.5 小结.....	249
参考文献.....	251

第一章 引 论

本章首先通过回顾计算机的发展历史,介绍处理器设计中的一些技术的演变过程。我们可以看到,诸如转移指令、变址寄存器、体系结构等概念的引入,都有其特定的技术背景。然后,论述在系统设计中通常需要考虑的一些问题和处理原则。象大概率事件原则、Amdahl 定律、引用的局部性等 in 计算机科学中有普适意义的原理,是每个计算机工作者应该熟知并自觉应用的。

1.1 历史回顾

1937 年,当 A. M. Turing 提出图灵机概念时,也许人们并没有感到计算机对人类社会的真正意义;然而,在此之后的半个多世纪,人们却在科学、技术,甚至哲学、法律等社会的各个领域都品味到计算机的强大震撼力。有些我们现在看来理所当然的处理器特点,在计算机不长的历史上,也的确经历了一个不短的认识过程。了解这个过程,包括其技术背景和器件工艺背景,对我们研究将来的计算机、以及用好现在的计算机都是有益的。

1.1.1 通用电子计算机的起源

1946 年,美国宾州大学摩尔分校的 J. Presper Eckert 和 John Mauchly 等人研制出了世界上第一台通用电子计算机 ENIAC(Electronic Numerical Integrator and Calculator)。这台机器体积非常庞大:100 英尺长、8.5 英尺高、数英尺宽。但是,从现在的标准来看,这台机器的性能却很不理想;例如,ENIAC 进行一次加法就需 200 微秒。即使如此,这台机器的出现仍是计算技术发展史

中一个最重要的里程碑。ENIAC 和早期的计算器有着显著区别,它在指令系统中提供了条件转移指令,而且这台机器是可编程的。ENIAC 的编程方式是通过手动的插拔头和设置开关来实现的;在程序执行中,由穿孔卡片提供计算中所需要的数据。

1948 年,曼彻斯特大学研制成功了一台小型计算机样机——Mark-I。它是第一台可存储程序的计算机;它的存储器大小为 32 字(可扩充到 8K 字);机器字长为 32 位;指令系统中共有 6 条指令:跳转、装入累加器并取反、减法、存累加器、测零和停机。另外,同一时期,在 ENIAC 的影响下,剑桥大学的 Maurice Wilkes 也设计了一台存储程序机器,称为 EDSAC (Electronic Delay Storage Automatic Calculator)。1949 年,EDSAC 开始运行,从而成为第一台全规模的、可操作的、存储程序计算机。在这台机器中采用的基于累加器的结构和其指令系统设计特点,对以后一段时期的机器设计有着重要影响。

在这里值得一提的是,受 ENIAC 设计的启迪,1945 年,John von Neumann 首次提出了存储程序的思想,即构成计算机程序的指令可同数据一样事先存放到存储器中;然后,由计算机自己一条条取出执行。这种思想很自然地引出了转移指令和可对指令的地址部分进行修改的概念,从而使一段程序的指令可以自动地、被有意义地多次执行。1946 年,von Neumann 在普林斯顿大学和 Goldstone 等人合作提出了 IAS 机器的设计方案。这台机器采用基于累加器的结构,指令系统中有五种指令:数据传送、无条件转移、条件转移、算术运算和地址修改。IAS 在取指方面的一大特色是:可以从主存中一次取出两条独立指令,并且把不立即执行的指令存放在指令缓冲寄存器中。IAS 计算机的设计,是对前人工作的一个总结,它使存储程序思想和计算机的设计紧密联系在一起,从而开辟了计算机技术发展的广阔前景。IAS 的设计思想在计算机发展史上起到了承前启后的作用。

这一时期的机器主要利用电子管来实现。一般,主存采用静电

管和延迟线实现,辅存则用纸带、穿孔卡和延迟线实现。由于当时硬件非常昂贵,所以当时的机器大都采用基于累加器的结构,而且仅能实现极少一些必须的机器指令。在这一时期,大多数机器主要是为进一步研究而研制的。

1.1.2 计算机产业的形成

作为帮助人们进行脑力劳动的工具,计算机只有在形成产业之后,才可能在社会上得以广泛应用。

1951年6月,Eckert—Mauchly 计算机公司开始交付 UNIVAC I 机器。当时,这种机器售价为 250,000 美元,共生产了 48 套系统,从而使之成为在商业意义上取得成功的第一种计算机系统。1952年,IBM 公司推出了它的第一台机器——IBM 701,在这台机器中使用了威廉管存储器。这种机器一共售出了 19 套。1953年,麻省理工学院的林肯实验室研制成第一台晶体管计算机。从此,晶体管器件开始逐步代替电子管器件,而成为计算机实现的主要器件。1955年,IBM 704 投入市场,在这台机器中首次使用了变址寄存器。利用变址寄存器,可以有效地实现动态修改指令的地址部分所要完成的功能。并且,在这台机器上首次出现了操作系统的原始模型——控制程序。1956年,Ferranti 有限公司建造成功第一台含通用寄存器的机器——Pegasus;它有八个通用寄存器,其中,R0 恒为“0”,在今天的很多处理器中,我们仍能看到这一特点。

为了进一步提高 IBM 704 的性能,1960年 IBM 推出了世界上第一台通用流水线机器——IBM 7030。它采用四级流水线结构,取指、译码和执行阶段是重叠的,这对提高速度非常有益。另外,在 1960年前后,IBM 公司推出了它的晶体管计算机 IBM 7094,在这台机器中第一次采用逻辑指令来进行非数值计算。该机器在许多方面进行的成功改进对后来计算机的发展都有着很大的影响。例如,它有一组变址寄存器,以及能够处理定点、浮点算术运算的硬件;所有的 I/O 操作均由一台可对主存储器直接进行存

取的 I/O 处理机来控制;存储部分分成两个模块,一个模块只有偶地址单元,而另一模块则只有奇地址单元,这样在一个存储周期内,就可对独立的模块进行交叉存取,从而为 CPU 提供两个连续的字;它的指令中有三个特征位,用以指示是否需要变址,以及使用哪个变址寄存器;另外,它采用了间接寻址技术,在这种技术中,间接寻址指令所形成的地址,不是存放操作数的地址,而是用来形成操作数地址的地址。间接寻址技术在排序等应用中是非常有用的。IBM 7094 采用了专用的程序控制指令,这种指令对应于不同程序间的控制转移,例如,它的 LINK/TRA 指令对,即后来人们常称的调用子程序/子程序返回指令对,对调用子过程等的处理是十分有效的;I/O 处理机与 CPU 间的通讯采用了中断控制,并且在 I/O 处理机中采用字的拆装技术,利用这种技术可以保证 CPU 和 I/O 设备间字长的匹配;采用了 DMA 技术,即当 I/O 处理机请求访问主存储器时,可能使 CPU 的主存访问请求延迟一个存储周期,但是并不中断 CPU 的操作,只是使 CPU 操作推迟一个存储周期。IBM 7094 的指令系统是非常完备的,共有 200 多条指令。这些指令分为七类:数据传送指令、定点算术指令、浮点算术指令、逻辑指令、变址指令、转移类指令和 I/O 指令。

由于计算机的不断普及,仅采用机器语言进行编程已不能满足广大用户的要求。例如,在 ENIAC 机器上,对一个典型的积分运算进行编程就需要三十分钟到一天的时间,而且要求编程人员对机器的许多具体细节情况都非常熟悉。这种情况促成了高级语言和操作系统的出现。1954 年人们提出了最早的 Fortran 语言,从而使 Fortran 成为出现较早、且第一个真正得到广泛使用的高级语言。在此之后,该语言得到了不断完善,1958 年,出现了 Fortran II;1962 年出现了 Fortran IV;直到 1991 年,国际标准化组织 ISO 还发布了最新的 Fortran 标准——Fortran 90。另外,1961 年前后,在曼彻斯特大学研制成功的 Atlas 计算机上,第一次使用了操作系统。我们知道,操作系统是一组用于对计算机系统进行全面