



高等学校教材

现代并行 计算机原理

康继昌 朱怡安 主编



西北工业大学出版社

TP338.6
K20

411551

高等学校教材

现代并行计算机原理

康继昌 朱怡安

主 编

康继昌 朱怡安

洪远麟 赵政文 张廷园

编 著

西北工业大学出版社

1997年6月 西安

(陕)新登字 009 号

JS/34/18

【内容简介】 现代并行计算机已发展到以 MIMD 型为主,节点机普遍采用大规模生产的高性能微处理器。本书较系统地全面地介绍现代并行计算机的基本原理,重点放在近几年发展起来的新技术,力求使读者尽快掌握现代并行计算机的基本原理和应用方法。

本书在并行机硬件(高性能微处理器结构、互连通信技术、并行机系统结构)的基础上,重点介绍了并行软件和并行算法,最后归宿到并行计算机的性能评价和应用方法。使读者从原理到应用建立起关于并行机的总体概念。

本书可作为高等院校计算机专业的高年级学生和研究生教材,也可供有关科技人员阅读和参考。

高等学校教材

现代并行计算机原理

主 编 康继昌

副 编 朱怡安

责任编辑 季 强

责任校对 钱伟峰

*

©1997 西北工业大学出版社出版

(710072 西安市友谊西路 127 号 电话 8493844)

陕西省新华书店发行

咸阳市印刷厂印装

ISBN 7-5612-0915-O/TP·111(课)

*

开本:787×1092 毫米 1/16 印张: 11.75 字数:281 千字

1997 年 6 月第 1 版

1997 年 6 月第 1 次印刷

印数:1—4 000 册

定价:12.50 元

购买本社出版的图书,如有缺页、错页的,本社发行部负责调换。

前 言

80年代还有争论的并行计算机,现在已经被公认是发展超高速计算机的唯一途径了。这主要是由于微电子技术的迅猛发展,为并行机提供了越来越便宜的高性能微处理器。过去1亿次每秒运算速度的巨型机,如银河-1号,是普通用户不敢问津的。现在1万元左右的奔腾PC机即可达到这个速度。多个微处理器组成的并行机,普通用户也能买得起。包括航空航天领域在内的需要高速计算的用户,对并行机的需求正在兴起。针对这种需求,1988年我们编写了《并行处理计算机系统》讲义,先后为研究生、本科生开设了该课程,收效很好。近10年来,我们还先后研制成功了MPSC并行仿真机、PD-100并行仿真机、EP-860全互连多机系统。在积累教学和科研经验的基础上,我们编写了这本教材。

并行机的发展极为迅速。以Cray为代表的向量流水机技术、以CM为代表的SIMD型并行机技术业已过时。现代的并行机已很明显地以MIMD型为主。本书虽然系统地介绍了各种并行机技术及其发展简史,但重点放在近年发展起来的新技术,力求以较短的时间,使读者尽快掌握现代并行机的基本原理和应用方法。

目前并行机在推广应用方面并不如人意,出现了“叫座”而不“卖座”的现象。究其原因是并行编程困难,使用不便。要解决这个问题,必须综合并行机软件和硬件两方面的先进技术来解决。黄铠教授在他的《高等计算机系统结构》专著中强调“要制造性能价格比高的计算机,必须强调要对硬件和软件作综合设计”。我们完全赞同这个意见。本书的编写就是从并行机硬件入手,突出并行软件和并行算法,最后归宿到并行计算机的性能评价方法和应用实例。全书共分八章。其中康继昌编写了第一、二、四章;朱怡安编写了第三、六、七章;赵政文、张延园编写了第五章;洪远麟编写了第八章。

第一章介绍了有关并行机的基本概念及其发展简史,引出了发展并行机中的技术难题。第二章给出了并行机的器件基础,即微电子技术和微处理器,特别是微处理器的新系统结构,包括RISC、多发射、超标量、超流水线和VLIW结构以及Cache技术等。第三章说明了总线、开关网络、消息传递等各种计算机的通信互连技术,重点放在最新的蛀孔通信技术。第四章给出了并行机系统结构。在紧耦合系统中,介绍了基于总线、基于交叉开关、基于开关网络和基于多端口存储器的各种类型;在松耦合系统中,则介绍了超立方体、超环、超节点等类型的网络结构。重点讨论了各种结构中的Cache一致性技术及其对并行机系统性能的影响。第五章列举了FORTRAN, occam, Linda三种并行编程语言。在并行编译中介绍了程序和数组相关性的概念及其分析方法。更重要的是程序并行化技术,它对推广并行机的应用有重要的意义。此外,还介绍了并行编程环境。在程序自动并行化没有实现之前,可为用户提供重要的编程工具。第六章给出了并行算法的复杂性概念和分类、评价方法,并介绍了SIMD和MIMD两类并行算法,以及任务分配、负载均衡的概念。第七章简要介绍了并行机性能评价方法。其中Gustafson定律是对Amdahl定律的发展,为发展并行机奠定了理论基础。最后,第八章结合EP-860全互连多机系统,介绍了多种应用的并行编程方法,使读者从原理到应用,建立起关于并行机的一个总体概念,从而激发他们进一步研究并行机的兴趣。

在本书编写过程中,参考了不少有关资料。其中最重要的有:黄铠教授著、王鼎兴教授等译的《高等计算机系统结构》;蔡希尧教授的《多处理机系统的逻辑分析与设计》;李学干、徐甲同教授的《并行处理技术》;李三立教授的《RISC 单发射与多发射体系结构》;张德富教授的《并行处理技术》。我们还参考了王敬文、龙卫红合编的《并行处理计算机系统》讲义。在此,我们对以上的作者们深表谢意。

西北大学郝克刚教授审阅了全书,提出了宝贵的修改和改进意见,我们表示衷心感谢。

博士生肖骊、戚德虎、况正谦,硕士生李贵山,本科生赵辉等录入了全部手稿,同时提出了不少改进意见,对本书的出版起了重要的作用。

另外,还要感谢国家自然科学基金,国家教委博士点基金,航空科学基金,国防科工委八五预研项目等多年来对我们科研工作的资助和支持。

由于并行机的技术发展极快,许多有价值的先进技术来不及收集、消化,希望读者见谅。更祈有关专家不吝指教。

作 者

1996年7月

目 录

第一章 绪论	1
1.1 为什么需要并行处理	1
1.2 并行计算机的发展简史	2
1.3 并行处理的定义和分类	3
1.3.1 按并行处理的粒度分类	4
1.3.2 按并行处理的规模分类	4
1.3.3 按编程级分类	4
1.4 并行机系统结构分类	5
1.4.1 弗林分类法	5
1.4.2 按耦合程度分类	5
1.4.3 非冯·诺依曼系统结构	6
1.4.4 并行机系统结构分类图	8
1.5 并行计算机的用途	8
1.6 关于发展并行机的争论	9
1.6.1 Grosch 定律	9
1.6.2 Minsky 猜想	10
1.6.3 计算机发展迅速论	10
1.6.4 Amdahl 定律	11
1.6.5 小结	11
1.7 并行处理中的难题	11
习题	12
第二章 VLSI 微处理器	13
2.1 VLSI 工艺的发展	13
2.2 VLSI 微处理器的新系统结构	14
2.2.1 RISC 技术	14
2.2.2 编译优化	15
2.3 RISC 技术的新发展	17
2.4 多发射结构	19
2.4.1 超标量结构	19
2.4.2 超流水线结构	19
2.4.3 超长指令字结构	19
2.5 i860 微处理器	20

2.6	T9000 微处理器	23
2.7	Alpha 21064 微处理器	27
	习题	29
第三章 互连通信技术		30
3.1	概述	30
3.2	总线通信	30
3.2.1	单总线系统	30
3.2.2	总线仲裁	31
3.2.3	多层总线	33
3.3	开关网络	34
3.3.1	交叉开关网络	34
3.3.2	单级开关网络	34
3.3.3	多级互连网络	39
3.4	多端口存储器通信	43
3.5	消息传送	44
3.6	蛀孔式通信	46
3.6.1	DS-link 通信协议	47
3.6.2	信包传送过程	48
3.6.3	T9000 的蛀孔式通信	48
	习题	50
第四章 并行机系统结构		51
4.1	概述	51
4.2	联结机	51
4.3	紧耦合多机系统	52
4.4	基于总线的紧耦合多机系统	52
4.4.1	Cache 一致性技术	53
4.4.2	监视与 Cache 控制器	55
4.4.3	有 Cache 多机系统的性能分析	58
4.4.4	多机系统总线	61
4.5	基于交叉开关的紧耦合多机系统	62
4.6	基于多级开关网络的紧耦合多机系统	64
4.7	基于多端口存储器的紧耦合多机系统	65
4.8	松耦合多机系统	65
4.9	超立方体网络	67
4.10	超环网络	68
4.11	超节点网络	70
4.12	松耦合多机系统的 Cache 一致性	71

4.13	脉动阵列	73
4.13.1	一维线性脉动阵列	74
4.13.2	六角形脉动阵列	75
4.13.3	脉动阵列的优缺点	77
4.14	波前阵列	78
	习题	79
第五章	并行软件	80
5.1	概述	80
5.2	并程序的构造	81
5.3	并行语言	81
5.3.1	并行 FORTRAN 语言	81
5.3.2	occam 语言	85
5.3.3	Linda 语言	87
5.4	并行编译	89
5.4.1	表达式并行化	89
5.4.2	程序相关性	90
5.4.3	几种并行程序结构的表示	92
5.4.4	数组相关性	93
5.4.5	程序并行化转换	97
5.4.6	并行化编译器	98
5.5	并行程序开发环境	100
5.5.1	并行程序开发环境的概念及构成	100
5.5.2	并行程序调试	101
5.5.3	并行程序性能分析	104
5.5.4	集成式用户界面	105
5.5.5	一些实际并行程序设计环境简介	106
5.6	并行操作系统	109
5.6.1	早期的并行操作系统	109
5.6.2	基于微核心的并行操作系统	109
	习题	110
第六章	并行算法	111
6.1	基本概念	111
6.2	并行算法的分类	111
6.3	并行算法的评价	113
6.4	SIMD 并行算法	114
6.4.1	向量求和	114
6.4.2	SIMD 矩阵乘	117

6.4.3 SIMD 递推问题的并行算法	120
6.4.4 SIMD 快速傅里叶变换	124
6.4.5 偏微分方程的并行求解算法	125
6.5 MIMD 并行算法	128
6.5.1 宏流水线并行算法	128
6.5.2 同步并行算法	129
6.5.3 异步并行算法	133
6.6 任务分配	136
6.6.1 静态任务分配	136
6.6.2 动态任务分配	143
6.6.3 动态负载均衡	144
习题	145
第七章 性能评价方法简介	146
7.1 Gustafson 定律	146
7.2 建模和分析	147
7.2.1 排队系统模型	147
7.2.2 排队系统中的性能指标及其关系	148
7.2.3 到达间隔时间的分布和服务时间的分布	149
7.2.4 排队模型的分析	151
7.2.5 Petri 网模型	154
7.3 基准测试	160
7.4 模拟方法	161
习题	164
第八章 并行机的应用	165
8.1 EP-860 全互连多机系统	165
8.2 并行程序设计模型	166
8.3 大型矩阵乘法的并行计算	168
8.3.1 任务的并行分割	168
8.3.2 数据通信	168
8.3.3 同步	169
8.3.4 矩阵相乘并行计算程序流程图	170
8.4 线性代数方程组并行计算	171
8.5 动态系统仿真并行计算	172
8.6 流场并行计算	174
8.7 人工神经网络并行计算	176
习题	178
参考文献	179

第一章 绪 论

并行计算机(以下简称并行机)是一种具有并行结构的高性能计算机。当单机的计算速度不能满足应用要求时,即可采用并行机来提高计算速度。早在 70 年代,美国研制的由 64 个处理器组成的阵列机 ILLIAC-N,于 1975 年投入运行,这就是一种早期的并行机。由于当时硬件成本过于昂贵,ILLIAC-N 未能实现商品化。超级机 Cray-1 采用向量流水结构,具有较高的“性能价格比”,1976 年研制成功后,迅速实现了商品化。Cray 系列超级机在超级计算机市场中称雄了近 20 年,获得了很大成功。然而当时,向量流水结构的超级机价格终究还是十分昂贵,只有少数大公司或军事研究部门才用得起。80 年代出现了超大规模集成电路的微处理器芯片,大幅度降低了计算机的硬件成本。这不仅使大量的个人计算机得到普及应用,而且也使成千上万个微处理器组成的大规模并行机成为现实。许多并行机已经出现于市场,例如,美国 Cray 公司的大规模并行机 CrayT3D 含有 2 048 个微处理器。更多的公司生产各种中小规模的并行机。有人称,90 年代是并行机的时代,这是不无道理的。我国计算机学会组织研究的“未来十年计算机科学技术发展战略设想”中,把并行与分布处理技术列为 12 个专题之首。这也足以说明发展并行机的重要性。本书将主要反映近年来并行机发展的最新成就,力求以现代的观点,介绍并行处理的基本方法和并行机的基本原理。我们不想罗列许多过时的研究成果,而是旨在以较短的时间,使读者很快地掌握应用与研究并行机所必需的基本知识和理论,并了解并行机的发展动向。

1.1 为什么需要并行处理

并行处理的目的是为了提高计算机的处理速度和处理能力。自 1946 年发明世界上第一台计算机以来,计算机的速度已经提高了成千上万倍。可是新的应用还是对计算机的速度不断提出更高的要求。例如,在气象预报、流场计算、石油勘探、核物理等方面,需要有 1 百亿次每秒,甚至 1 万亿次每秒以上的浮点运算速度(10 GFLOPS~1 TFLOPS)。美国计划在本世纪末以前造出 1 万亿次每秒的计算机。单机的速度受到了材料物理的限制。根据电子在硅材料中运动的极限速度估算,单机的极限速度是 1 GFLOPS,离上述 1 TFLOPS 的速度指标还差得很远。因此,只有用成百上千个微处理器组成并行机才能达到 1 万亿次每秒的速度。

同时,计算机的应用日趋复杂,从数据处理(如科学计算)、信息处理(如企业管理)、知识处理(如专家系统)、直到智能处理(如定理证明,自然语言理解,逻辑推理等),计算机应用的范围越来越广,处理问题的规模越来越大,对并行处理的要求越来越迫切。

此外,并行处理不仅用于大型问题的计算,而且也用于实时问题的求解。例如,机器人实时控制,高速武器的实时控制或实时仿真等,计算的工作量虽然不是很大,但要求在极短的时间内完成计算,也得求助于并行处理来提高计算速度。又如现代高性能的工作站,在复杂的图形计算中,往往也采用并行处理技术来提高速度。通常工作站并无严格的实时要求,然而在人机交互环境下,处理复杂问题时,人会等得不耐烦。响应速度快的工作站肯定更能受到用户的

欢迎。

总之,过去计算机的运行速度达到1亿次每秒以上,即被认为是巨型机。它是衡量一个国家实力的一项重要指标。并行机使巨型机的计算速度成数量级地增长,其对国民经济和国防力量的重要意义也就可想而知了。

1.2 并行计算机的发展简史

在计算机的发展历史中,并行技术一直是提高计算速度的一条重要途径。

早期的计算机中,运算器曾是串行工作的。它只有一位加法器,两数相加时,从最低位开始逐位相加,正如手算的过程一样。后来发展的并行运算器,一拍即可完成两数相加的运算,当时就称之为并行计算机。这样的并行计算机逐渐地普及,也就没有必要再称它为并行计算机了。

1966年美国Illinois大学开始研制早期的并行机ILLIAC-IV,它由一个控制部件和64个处理单元组成。每个处理单元能做64位的浮点运算,设计速度为400万次每秒。64个处理单元连成 8×8 阵列,峰值运算速度可达2.5亿次每秒。由于受当时元器件条件的限制,设计几经修改,直到1975年才勉强提供使用,测试到的最大处理速度只是接近5000万次每秒。ILLIAC-IV的研制可以认为是失败的,它花费了四倍合同规定的金额,但性能甚至没有达到它最初提出的十分之一。但是,它对并行机的发展,影响是深远的。

比较成功的并行机是Cray-1。它从1972年开始研制,1976年第一台样机即交付使用。该机有12个流水功能部件,时钟周期12.5 ns。其主要的新颖特点是设有8个向量寄存器组,每组能保存64个字长为64位的浮点数。由于它采用向量流水结构,有很高的性能价格比。实测运算速度达到1.3亿次每秒浮点运算,因而成功地实现批量商品化生产。接着1983年以后,推出了它的改进型Cray X-MP,它实际上由2~4台Cray-1,通过共享存储器互连而成,属于并行的流水结构,速度达到4亿次每秒浮点运算。1985年又推出Cray-2,运算速度提高到20亿次每秒浮点运算。

自从VLSI微处理器问世以后,采用大量微处理器来组成并行机,成为并行机发展的一个新方向,而且取得了很大成功。这是由于VLSI微处理器可以大规模生产,就使得用成千上万个微处理器来组装并行机成为可能。又由于VLSI微处理器比较便宜,同样的性能指标,它比传统的巨型机可以便宜10~100倍,从而使并行机的性能指标扶摇直上。几百亿以至上千亿次每秒运算速度的并行机已有商品。1万亿次每秒的并行机也已在研制之中。

并行机诱人的发展前景,使许多国家,许多公司投入研究和竞争。比较有名的是美国Intel公司1985年推出的个人超级计算机iPSC(Personal Super-Computer)。它以Intel 80286作为节点机,采用超立方体互连结构,节点机数为8、16、32、64、128等可选。到了80年代后期,用i860来更新原来的节点机,使运算速度最高可达到50亿次每秒。90年代初,又推出了Touchstone Delta并行机,由500个i860组成,采用二维阵列结构,峰值速度达到400亿次每秒。1986年美国思维公司推出的Connection Machine CM-1,由65536个1位微处理器组成,运算速度为10亿次每秒,问世后第一年内即生产了16台。

美国的工业界和科教界都大力发展并行机,并作出了很大贡献,但他们采取的风格有所不同。工业界比较重视近期的经济效益,力求较高的“性能价格比”,采用比较成熟的技术,风险较小,面向针对性的用户,尽量方便用户编程,有一定的批量生产。例如,CDC公司原来已有二维

数据处理的用户,了解到用户有三维数据处理的需求,内存和计算速度都要提高 100 倍,于是投资研制更高档的 CYBER-250 计算机,取得很大成功。

科教界比较重视远期的发展,尽量采用新技术、新思想、新结构,敢于冒风险,追求从根本上提高系统性能,降低生产成本。他们往往生产一台样机,就此结束。例如美国 Illinois 大学研制的 ILLIAC-IV 和 Cedar 并行机。也有工业界和科教界结合较好的。如美国 Intel 公司和加州理工学院研制的 iPSC 和 Touchstone Delta 并行机,创造了超立方体结构、蛀孔式通信等新技术,将科研成果很快地转化为商品,值得我们借鉴。

英国 inmos 公司在 80 年代相继推出 Transputer T212、T414、T800 等一系列 VLSI 微处理器。其设计颇有特色,每个单片微处理器带有 4 对通信链路,极易构成二维阵列结构。到 90 年代初又推出具有超标量流水结构的 T9000,峰值速度达到 200 MIPS+20 MFLOPS。西德 Parsys tech 公司宣称用 16 000 个 T9000 组成 GC 系列并行机,峰值速度达到 4 000 亿次每秒运算(400 GFLOPS)。

1.3 并行处理的定义和分类

计算机是人类脑力劳动的工具,俗称“电脑”。俗话说,“一心不能两用”,说明人的思维活动是串行的。但是人的感觉器官往往是并行工作。例如,“看”电视的同时可以“听”,“看”和“听”并行工作。单从视觉来研究,视网膜大量的视神经细胞并行接受外界光线的刺激,传到大脑,一眨眼的功夫就可理解所看到的景物。可见,大脑的视觉是大规模并行处理的。每个人可以从切身的感受来理解“串行”和“并行”的概念。

黄铠和 Briggs 对并行处理的定义是:并行处理是一种信息处理的有效方式,着重开发计算过程中的并发事件。并发性(concurrency)含有同时性(simultaneity),并行性(parallelism)和流水线(pipelining)三种意思。同时事件是在同一瞬间发生的;并发事件可以在同样的时间间隔内发生;流水线事件则在重迭的时间间隔内发生。

同时性通过资源重复来实现并行。两个以上处理机组成的系统就符合同同时性的定义,这是最严格定义的并行性。过去硬件价格昂贵,资源重复的并行性应用较少。现在由于 VLSI 迅猛发展,价格便宜的微处理器大量涌现,从而使资源重复成为实现并行机的主要途径。

广义的并行性可通过资源共享来实现。典型的例子是计算中心广泛采用的分时系统。表面看来,许多用户在同时使用同一台计算机。实际上,他们是在轮流使用,在一台机器中,实现多道程序或多进程的并行运行。它对并发性软件的发展起了很大作用。

流水线属于时间重迭的一类。假如一个流水线加法器设计成对阶、尾数相加和规格化等三级,如图 1.1 所示。一对数相加需 3 拍时间,但 100 对数依次流水相加并不需要 300 拍,而是只需 102 拍,平均每对数相加只需 1 拍多一点时间。我们也可以把加法器的 3 级看成是 3 个部件。并行性就体现在这 3 个部件在并行工作,只是 3 级加法过程在时间上依次错开。这是一种效率很高的并行方式。Cray-1 的成功就是由于采用了这种技术。

按照上述定义,所有高性能的计算机都采用了各种各样的并行技术,都可称为并行计算机。现在由于 VLSI 微处理器的出现,关于并行处理有了新的定义,它仅指多个资源的同時工作。并行计算机也仅指多个节点机组成的计算机系统。本书重点介绍的就是这种现代的并行计算机系统,简称并行机。

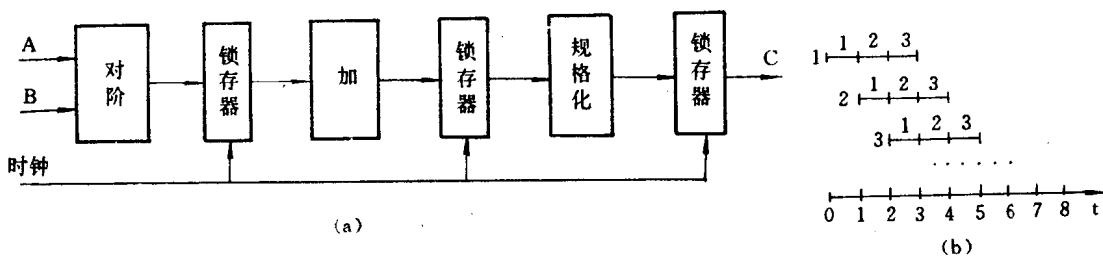


图 1.1 流水线加法

1.3.1 按并行处理的粒度分类

粒度(granularity)是并行处理的一个基本特性。粒度是指并行处理的基本单位的大小,大致可分为粗(coarse)粒度、中(medium)粒度和细(fine)粒度三类。粗粒度是指以大块的程序为并行处理的单位;细粒度则是指并行处理单位是一条语句、一个表达式、甚至是一个简单的算术或逻辑操作;介于上述两个极端之间的就是中粒度并行性。

Cray-2 和 Cray X-MP 都是由 4 个大型处理机组成,每个处理机都要运行大块的程序,属粗粒度并行的例子。细粒度并行的例子如 CM-1,由 65 536 个处理器组成,它们在统一的控制器指挥下并行工作,每次只做一个简单的操作。

大多数的商品并行机系统,如 Intel 公司的 iPSC, Alliant 公司的 FX/8, 英国的 Transputer, 均属于中粒度并行一类。

粒度大小对并行机系统的性能有很重要的影响。节点机数目较少的并行机,一般采用粗粒度并行;反之,节点机数目较大时,一般采用细粒度并行。只有在计算问题的并行算法很适合硬件结构时,才能较充分地发挥并行机系统的潜力。

1.3.2 按并行处理的规模分类

并行处理的规模由节点机数的多少来衡量,通常分为小规模、中规模和大规模三类。节点数在几十以下的称为小规模并行机,也称多机系统。节点机数为几百的,称为中规模并行机。节点机数达到成千上万以上的,称为大规模并行机(massively parallel computer),也有称高度并行机或超并行机的。

1.3.3 按编程级分类

从编程的角度看,并行处理可以在不同的级上实现。一般分为以下三级:

- (1) 作业级或程序级 多个作业(或程序)并行处理;
- (2) 任务级或过程级 多个任务(或过程)并行处理;
- (3) 指令级 多个指令并行执行。

并行处理的最高级是作业级的并行处理。多个用户的作业并行处理比较简单,因为各个作业相互独立,各算各的,互不干扰。编程也是各编各的,都是串行程序,不存在并行编程的问题。通常计算中心配置的中、大型通用机由许多用户共享,多属这种类型,即分时(time-sharing)系统。现在有了多机系统,一个用户可以分配在一个节点机中去运行。这就成为多处理(multi

-processing)系统。

其次是任务级的并行处理。一个用户的作业分为多个任务,或者一个程序分为多个过程(即程序段),以便并行处理,分工合作地完成一个计算任务。这需要利用并行算法才能高效地把一个问题分解成多个子问题,分别编程(即编成任务或过程)。由于计算的是一个完整的问题,各个任务间有相互联系,编出的是并程序,有别于传统的串程序。为了便于编制并程序,相应地需要有并行语言、并行软件、并行软件工具等。这是当前并行计算机能否推广应用的

关键。

第三是指令级并行。上述的作业级和任务级并行都属粗粒度(或中粒度)并行,指令级则是细粒度并行。为了实现指令级并行必须事先分析数据相关性。没有数据相关性的指令,才能并行执行。否则,会造成错误的计算结果。

1.4 并行机系统结构分类

并行机系统结构可以从不同的角度进行分类。

1.4.1 弗林分类法

1964年,弗林(Flynn)提出根据指令流和数据流的多少来对计算机系统结构进行分类。所谓指令流是计算机在计算过程中执行的一串指令;而数据流是指令执行过程中的一串数据。于是弗林将计算机系统结构分为以下四类:

- 单指令流单数据流(SISD) 通常的串行计算机均属此类;
- 单指令流多数据流(SIMD) 如ILLIAC-N有多个处理器而只有一个控制器;
- 多指令流单数据流(MISD) 如脉动阵列式(SYSTOLIC ARRAY)计算机;
- 多指令流多数据流(MIMD) 由多个节点机组成的并行机则属此类。

弗林分类法不是很严格的,例如专家们对向量流水机属于何类有不同看法。有的认为是SISD,有的却认为是SIMD。这无关紧要。弗林分类法比较直观,应用极为广泛。

由大量微处理器组成的现代并行机均属MIMD这一类。

1.4.2 按耦合程度分类

并行机的各个节点机通过互连网络耦合成一个整体。根据耦合的紧密程度可分为紧耦合和松耦合两大类系统。

1. 紧耦合系统

紧耦合系统中,各节点机P通过开关网络和共享存储器模块M互连,如图1.2所示。由于每个节点机离开共享存储器都不能独立工作,运行过程中,各节点机和共享存储器之间通信频繁,故称紧耦合系统,也称多处理机系统。优点是整个系统共享一个存储空间,可以沿用传统的高级语言,编程方便,故易于推广使用。缺点是当多个节点机访问同一存储器模块时,

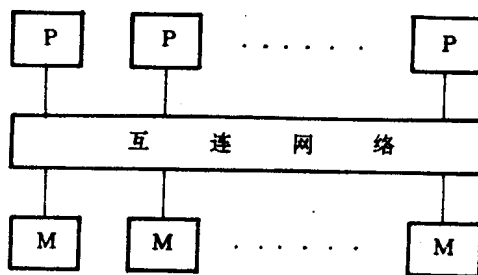


图 1.2 紧耦合系统

会发生冲突。节点机数目较多时,开关网络成为通信瓶颈,系统性能将下降,故紧耦合系统一般适用于节点机数目较少的情况。

紧耦合系统的开关网络一般采用总线结构。采用总线进行通信时,在同一个瞬间只允许一个节点机向总线发送信息。为了防止多个节点机同时向总线发送信息,需要有仲裁排队电路。采用多级开关或者交叉开关,可以允许多个节点机同时访问不同的存储模块,但是控制电路复杂,成本要提高。对于大规模的并行机,存储器可采用分布的结构,即分布在各个节点机中。虽然存储器并不集中,但它们统一编址,组成统一的存储空间。每个节点机既可访问就近的存储器也可访问到远距节点机的存储器。如果访问远距存储器和访问就近的存储器所需的时间相同,称为 UMA (Uniform - Memory - Access); 如果不相同,则称为 NUMA (Non - Uniform - Memory - Access)。紧耦合系统还可以有别的新结构,比较有名的是美国的 KSR - 1 系统,它采用只有高速缓存的 COMA 系统结构 (Cache - Only - Memory - Architecture)。

2. 松耦合系统

松耦合系统中,各节点机有自己的局部存储器,指令和大部分数据都可在本机内访问到,具有相对的独立性。只有少数共享数据需要在节点机之间通信,因此称为松耦合系统。松耦合系统中的节点机是可以独立工作的计算机,属于多计算机系统。节点机之间主要采用链路(消息传递)方式进行通信。

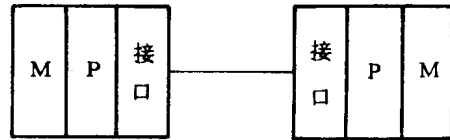


图 1.3 松耦合系统

链路通信如图 1.3 所示,节点机通过接口“点对点”地互连起来。通信以消息为单位,有一定的通信协议来定义消息的格式,表示消息的起始,消息的长度以及消息的结束等。多个节点机互连时可以有环形、树形、星形、超立方体、超环等各种拓扑结构。

松耦合系统也可以采用共享存储器或总线进行通信。与紧耦合系统不同的是共享存储器的容量比紧耦合系统要小得多,通信量也少得多。当节点机较多时,总线成为通信瓶颈的问题也缓和得多。但是,当节点机大量增多时,总线还是会成为通信瓶颈。总线通信结构简单,易于实现,所以得到广泛使用。各家公司有各自的总线标准。如 Intel 公司的 MULTIBUS I, MOTOROLA 公司的 VME BUS。

近年来还兴起了计算机群 (computer cluster) 的新并行结构。这是由于 PC 机或工作站的价格下降,应用十分普及的缘故。数量可多可少的 PC 机或工作站用高速网络互连起来,成为 PC 机群或工作站群,其性能可以与巨型机相匹敌,而价格则便宜得多。据报道,几百台工作站白天分散使用,晚上连成群,计算大型题目,其功能相当于几十台 Cray Y - MP。

1.4.3 非冯·诺依曼系统结构

以上讨论的都是基于传统的冯·诺依曼结构。这种结构的特点是整个机器都在指令控制之下工作,故属控制驱动方式。在单机中,指令只能顺序执行,本质上只能串行工作。为了摆脱这种制约,人们多年来探索着新的非冯·诺依曼结构。一种是数据驱动的数据流结构,另一种是需求驱动的归约结构。

1. 数据流结构 (Data - Flow Architecture)

数据流结构把整个计算过程看成是数据的流动。每个运算所需的数据一到,就马上执行。

例如,要计算 $(2x+y)(x-y)-xy$,其数据流图如图 1.4 所示。

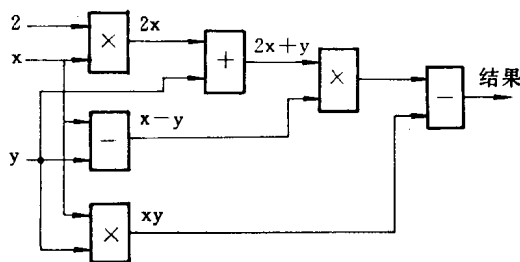


图 1.4 数据流图

第一步, $2x, x-y, xy$ 三个运算的数据都有了, 均可立即执行;

第二步, 只有 $(2x+y)$ 一个运算可以执行;

第三步, 执行 $(2x+y)(x-y)$;

第四步, 执行 $(2x+y)(x-y)-xy$, 得出结果。

看来, 第一步虽有 3 个运算并行, 可是以后每步只能做 1 个运算, 提早算出的结果只好等着, 并无明显的优点。这就是为什么要提出需求驱动的归约结构的原因。然而, 对于有大量可以独立运算的复杂问题, 数据流结构还是可以大大提高并行运算的机会, 加快计算过程。

2. 归约结构(Reduction Architecture)

归约的意思是化简。归约结构把整个计算过程看成是一个化简过程。每个运算不是数据一到就立即执行, 而是等到需要时才执行。

还是以数据流结构中的例子来说明, 计算过程如图 1.5 所示。为了说明化简过程, 要倒过来从结果开始向前逐步解释。为了得到结果, 需要做一个减法。但是数据还没有到, 需要先做两个乘法。第一个乘法条件不具备, 需要先做一个加法和一个减法, 因而两个乘法一起等着。做加法的条件不具备, 需要先做一个乘法, 加法和减法一起等着。 $2x$ 乘法条件具备, 立即执行。这是第一步。第一步执行后, 第二步运算的条件具备, 接着执行。依次类推, 也是经过 4 步, 完成计算。可见, 每个运算都是等到需要时才执行, 故称需求驱动。

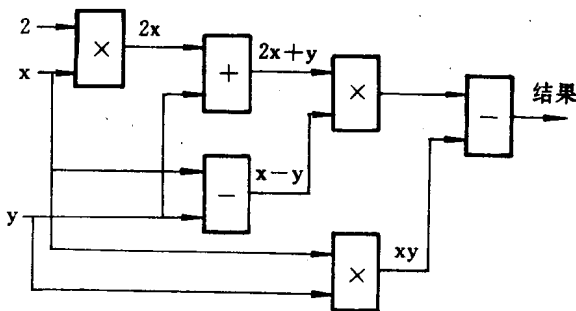


图 1.5 归约图

上述两种非冯·诺依曼结构, 控制都极为复杂, 成本很高, 至今尚未达到实用阶段。因此,

我们不作更多的介绍。

1.4.4 并行机系统结构分类图

综合以上的各种分类,可以给出图 1.6 所示的并行机系统结构分类图。它使我们对并行机系统结构分类有一个总的概念。这对我们把握并行机的发展和应用是很有帮助的。

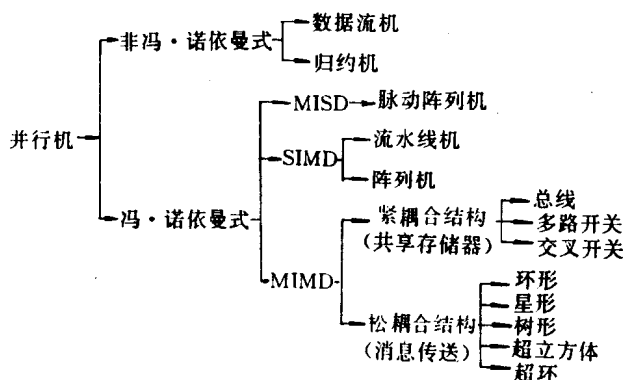


图 1.6 并行机系统结构分类图

图中表示并行机可分为冯·诺依曼式和非冯·诺依曼式两大类系统结构。非冯·诺依曼式又分为数据流机和归约机两大类。冯·诺依曼式则可分为 MISD、SIMD、MIMD 三类。其中 MISD 的代表是脉动阵列式并行机。SIMD 有流水线机和阵列机两类。前者以 Cray-1 为代表;后者以 ILLIAC-N 或 CM-1 为代表。我们主要感兴趣的是 MIMD 这一类。它又分为紧耦合系统和松耦合系统两类,前已作过介绍。

1.5 并行计算机的用途

在 1.1 节中已经介绍了为什么需要并行计算机,本节将进一步较详细地介绍并行机的若干重要用途。

1. 天气预报

天气预报对国民经济、人民生活,特别是对农业生产、灾害预防有极为重要的价值。现在天气预报主要依靠计算机的数值计算,这就需要在地球的球坐标系内,按经度、纬度和高度划分三维网格点,再加上第四维时间,取适当的步长。设网格间距为 400 km,计算 24 h 天气预报需 1 亿次每秒的 Cray-1 计算约 100 min。如果要提高精度,把四维的网格间距都减小一半,计算量将增加 16 倍,这需要 Cray-1 计算约 26 h,计算结果已经过时,这就没有实用意义了。如果我们希望进行更精确的长期天气预报,那么必须采用速度快得多的上百亿次每秒的计算机,这只有并行机才能做到。

2. 计算空气动力学

风洞吹风实验需要大量能源,耗资很大,而且还受许多条件的限制,如模型尺寸、吹风速度、空气密度、温度、洞壁干扰等。数值计算就没有这些限制,只受计算机的计算速度和存储容量的限制。为了完成整架飞机的气动计算,10 亿次每秒的超级计算机勉强够用。为了更精确地