

全国高等教育自学考试



# 计算机系统结构自学辅导

组编 / 全国高等教育自学考试指导委员会  
主编 / 李学干



吉林大学

3-44

全国高等教育自学考试  
自学指导丛书

# 计算机系统结构 自学辅导

全国高等教育自学考试指导委员会 组编

主 编 李学干

吉林大学出版社

**图书在版编目(CIP)数据**

计算机系统结构自学辅导/李学干主编. —长春:吉林大学出版社,  
2000.10

ISBN 7-5601-2418-6

I. 计... II. 李... III. 电子计算机-系统结构-高等教育-自学参考资料  
IV. TP303

中国版本图书馆 CIP 数据核字(2000)第 48208 号

**计算机系统结构自学辅导**

主 编 李学干

---

责任编辑:陈 铮

责任校对:陈 铮

---

吉林大学出版社出版  
(长春市东中华路 37 号)

吉林农业大学印刷厂印刷

---

开本: 787 × 1092 毫米 1/16

2002 年 3 月第 1 版

印张: 9.75

2002 年 3 月第 1 次印刷

字数: 235 千字

印数: 1-5000 册

---

ISBN7-5601-2418-6/TP·108

定价: 13.50 元

本书如有质量问题, 请与教材供应部门联系。

## 内 容 简 介

本书是为参加全国计算机及应用专业《计算机系统结构》课程考试的考生编写的辅导材料。该书在概述了课程的地位、性质、特点、学习应注意的问题及对各章学习时间安排的建议等之后,主要是围绕自学考试大纲对各章的考核内容及知识点进行全面的讲解,紧凑而实用。通过对大量习题和题例进行分析解答,可以加深考生对该课程知识、重点、难点的掌握和领会。另外,还配备了若干套综合自测题和参考答案。

本书对考生复习和顺利通过该课程考试将会有很大的帮助,是必备的参考书。

# 目 录

## 第一部分 概 述

1.1 课程的地位和重要性 .....	1
1.2 课程的性质和特点 .....	1
1.3 学习中应注意的问题 .....	2
1.4 对各章学习时间安排的建议 .....	2

## 第二部分 各章内容要点及习题解答

<b>第1章 计算机系统结构的基本概念</b> .....	<b>4</b>
2.1.1 本章基本要求 .....	4
2.1.2 内容要点 .....	5
1. 计算机系统的多级层次结构 .....	5
2. 计算机系统结构、组成与实现 .....	5
3. 软硬件取舍与计算机系统思路 .....	6
4. 结构设计要解决好软件的可移植性 .....	6
5. 应用与器件的发展对系统结构的影响 .....	7
6. 系统结构中的并行性发展及计算机系统的分类 .....	8
2.1.3 习题和题例的分析与解答 .....	9
<b>第2章 数据表示与指令系统</b> .....	<b>14</b>
2.2.1 本章基本要求 .....	14
2.2.2 内容要点 .....	14
1. 数据表示 .....	14
2. 寻址方式 .....	16
3. 指令格式的优化设计 .....	17
4. 按 CISC 方向发展与改进指令系统 .....	18
5. 按 RISC 方向发展与改进指令系统 .....	19
2.2.3 习题和题例的分析与解答 .....	19
<b>第3章 总线、中断与输入输出系统</b> .....	<b>31</b>
2.3.1 本章基本要求 .....	31

2.3.2	内容要点	31
1.	输入输出系统的基本概念	31
2.	总线设计	32
3.	中断系统	33
4.	通道处理机	34
2.3.3	习题和题例的分析与解答	36
<b>第4章</b>	<b>存贮体系</b>	<b>46</b>
2.4.1	本章基本要求	46
2.4.2	内容要点	47
1.	存贮体系的概念与并行主存系统	47
2.	虚拟存贮器	48
3.	高速缓冲(Cache)存贮器	51
2.4.3	习题和题例的分析与解答	54
<b>第5章</b>	<b>重叠、流水和向量处理机</b>	<b>72</b>
2.5.1	本章基本要求	72
2.5.2	内容要点	73
1.	重叠方式	73
2.	流水方式	74
3.	向量的流水处理与向量流水处理机	77
4.	指令级高度并行的超级处理机	78
2.5.3	习题和题例的分析与解答	78
<b>第6章</b>	<b>阵列处理机</b>	<b>94</b>
2.6.1	本章基本要求	94
2.6.2	内容要点	94
1.	阵列处理机原理	94
2.	阵列处理机的并行算法	95
3.	SIMD 计算机的互连网络	96
4.	并行存贮器的无冲突访问	98
2.6.3	习题和题例的分析与解答	98
<b>第7章</b>	<b>多处理机</b>	<b>106</b>
2.7.1	本章基本要求	106
2.7.2	内容要点	106
1.	多处理机的特点及主要技术问题	106
2.	多处理机的硬件结构	107
3.	程序并行性	108

4. 多处理机的性能 .....	109
5. 多处理机的操作系统 .....	109
2.7.3 习题和题例的分析与解答 .....	109
<b>第8章 其它计算机结构</b> .....	<b>118</b>
2.8.1 本章基本要求 .....	118
2.8.2 内容要点 .....	118
1. 脉动阵列机 .....	118
2. 大规模并行处理机 MPP 与机群系统 .....	118
3. 数据流机 .....	119
4. 归约机 .....	119
5. 智能机 .....	119
2.8.3 习题和题例的分析与解答 .....	119

### 第三部分 考试中应注意的问题

### 第四部分 综合自测题及参考答案

第1套综合自测题 .....	124
第2套综合自测题 .....	127
第3套综合自测题 .....	131
第1套综合自测题的参考答案 .....	135
第2套综合自测题的参考答案 .....	139
第3套综合自测题的参考答案 .....	143

# 第一部分 概述

## 1.1 课程的地位和重要性

电子数字计算机的发展已有了 50 多年的历史。从 20 世纪 60 年代中期开始,随着计算机应用领域的扩大,使集成电路工艺和制造技术有了迅猛的发展。与此同时,外存贮器的技术和性能也有了很大的改进和提高。它们都有力地推动着计算机系统向更高的性能发展。

从应用的角度看,人们一直在追求不断提高巨、大、中、小、微、亚微(膝上型和笔记本型)及微微(掌上型)型等各种计算机的性能。这就迫使计算机系统结构需要不断地改进,研究使用更新的结构和组成技术,并将其迅速用于高性能的计算机上。同时,一度曾为巨、大型机上所研制的结构技术,也在迅速地搬移到小型机和微型机上,以较低的代价,获得较高的性能。例如,20 世纪 70 年代前后,在巨型机、大型机上所采用的虚拟存贮器、高速缓冲存贮器(Cache 存贮器)、流水线、向量处理技术都被陆续搬移到高档微型机和一般的微型机上。近年来,已出现以微型机、微处理机为基础的高度并行的巨、大型计算机系统。并行处理所采用的技术已渗透到小型、微型机中。另一方面,微型、小型机上首先出现的精简指令系统计算机(RISC)的系统结构也正在向巨、大型机方向发展。因此,计算机系统结构已不再是过去所认为的只是研究巨、大型机上的结构设计技术,现在已经跨越于从巨型到微型计算机的整个系列之中了。这种非常引人注目的局面是近一二十年里形成的。

总之,不管是对研究计算机结构设计的技术人员,还是对硬件设计、软件设计和应用系统设计的技术人员来讲,较为系统全面地学习、了解和掌握计算机系统结构的基本知识、内容和发展等就是非常必要的了。

## 1.2 课程的性质和特点

《计算机系统结构》是从计算机组织和结构的角学习领会计算机系统的课程。计算机系统是软件和硬件相结合的一个复杂的综合体。根据现有硬件和器件技术的状况,面向各种不同的应用,需要研究如何对计算机系统的软件和硬件功能进行更合理的分配,并要研究如何更好地实现分配给硬件的那部分功能,使系统有尽可能高的性能价格比。

通过对本课程的学习,使学生能进一步树立和加深计算机系统的整体概念,特别着重于正确掌握有关计算机系统结构的基本概念、基本原理,了解目前采用的比较成熟的基本结构,掌



握结构设计的基本思想和方法,提高分析问题和解决问题的能力。同时,也让学生能了解到近十几年里,在并行处理和系统结构技术上的一些重要进展及今后可能的发展趋势。

本课程的内容广泛,涉及面较宽,要求学生有数字逻辑、程序设计语言、概率论、数理统计、计算机组成原理、数据结构、操作系统等方面的基础知识。课程内容的概念性和理论性很强,且有一定的深度,学习起来难度是比较大的。尤其是对只接触过微处理机,而对中、大型机缺乏感性知识的学生来讲,深刻领会和吃透本书各章的知识点,应该说是需要下一定的功夫的。

### 1.3 学习中应注意的问题

在学习本课程教材前,学生应仔细阅读本课程的自学考试大纲,对大纲中所列的基本要求,本课程与相关课程的联系等要有一定的了解。在学习教材每一章前,学生应了解清楚自学考试大纲上对该章所定的学习目的和总的要求、考核要求和考核知识点,弄清其中的重点和难点在什么地方,以及对这些知识点的能力层次要求。

学习时,应依据教材和本辅导教材,对其中的重点和难点部分多花些时间消化领会,对所学的基本原理、方法等理论性、概念性较强的部分,应着眼于理解其精神实质,进行必要的归纳和小结,理出有关要点和思路。对于典型的应用或计算、设计等问题,应领会其分析的方法和解题的思路和步骤。

学完每一个章节时,学生应认真完成教材中所列的习题,再和本书提供的解题范例或步骤比较,分析其间有什么大的差异和问题。这样,就能进一步加深对所学知识的理解和掌握,提高灵活运用知识和分析、解题的能力。

### 1.4 对各章学习时间安排的建议

由于各人的基础、水平及理解能力不同,这里提供学习各章时间安排的建议,仅供参考,各人可根据自己的情况作适当调整。另外,可再增加一些用于课程了解、总复习和作综合练习的时间。

学习中若遇到一时理解不深或搞不清的问题,可将其暂时搁置,等学习完其它各章后,再来重读、领会和加深。

第1章,计算机系统结构的基本概念,约需16个学时。重点是:计算机系统结构、计算机组成、计算机实现三者的定义及所包含的内容;有关的透明性问题判断;软件和硬件的功能分配原则;软件可移植性的途径、方法、适用场合、存在问题和对策;有关并行性的概念;系统结构中开发并行性的途径和类型等。

第2章,数据表示与指令系统,约需38个学时。重点是:自定义数据表示;浮点数尾数的基值选择;数的下溢处理方法;寻址方式中的再定位技术;信息在存贮器中按整数边界存贮的概念;操作码和指令字格式的优化;CISC指令系统的改进途径综述;RISC概念及所采用的基本技术等。

第3章,总线、中断与输入输出系统,约需18个学时。重点是:总线的控制方式;数据宽度;中断的分类和分级;中断处理次序的安排和实现;通道流量的分析和设计等。

第4章,存贮体系,约需36个学时。重点是:段页式和页式虚拟存贮器的原理;页式虚拟

存贮器的地址映象;LRU、FIFO、OPT 替换算法进行页面替换的过程模拟;LRU 替换算法对页地址流的堆栈处理模拟及性能分析;Cache 存贮器的直接和组相联地址映象;用 LRU 替换算法进行块替换的硬件实现及替换过程模拟;Cache 存贮器的性能分析等。

第 5 章,重叠、流水和向量处理机,约需 28 个学时。重点是:重叠方式中的“一次重叠”原理及相关处理;流水处理机的主要性能分析,局部性相关处理,全局性相关处理;单功能非线性流水线的调度;向量处理机中向量指令之间的串行、并行和链接执行的分析及所需时间的计算等。

第 6 章,阵列处理机,约需 16 个学时。重点是:阵列处理机与流水线处理机结构特点的对比;SIMD 计算机的基本单级互连网络及互连函数;多级互连网络中的立方体和混洗交换网络;并行存贮器实现无冲突访问的存贮单元分布规律等。

第 7 章,多处理机,约需 18 个学时。重点是:多处理机的机间互连;程序中的并行算法、并行性分析;并行语言中有关并行任务的派生和汇合的表示及时空图描述等。

第 8 章,其它计算机结构,约需 10 个学时。这一章,只做一般的要求,进行通读和识记一些基本名词术语即可。

## 第二部分 各章内容要点 及习题解答

### 第1章 计算机系统结构的基本概念

#### 2.1.1 本章基本要求

本章是为进一步深入学习后续各章打基础的。通过学习,让我们领会和建立起作为计算机系统结构设计应具有的一些基本知识和概念。

本章的基本要求是:

1. 领会一台完整的通用计算机系统可以被看成是由多个不同机器级构成的多级层次结构,每一级都可看成是一台机器,都有自己的机器语言和实现方法。这样的多级层次结构一般可分为哪几级,各机器级所处的相对位置及所用的主要实现方法。

2. 掌握计算机系统结构、计算机组成和计算机实现三者的定义及各自研究的方面和内容。领会计算机系统结构是软件和硬件的主要分界面的概念。理解计算机系统结构、计算机组成和计算机实现三者存在着相互的影响。领会透明性概念,能从不同角度对某个具体问题正确选择是否应设计成透明的结论。

3. 领会一个功能分别用软件和硬件实现的优点和缺点。掌握在功能分配中的软、硬件比例取舍的基本原则。领会计算机系统分别采取“由上往下”、“由下往上”设计的方法、各自存在什么问题。对通用机为什么应采取“从中间开始”向两边设计的方法,如何进行设计,这样设计有什么优点。

4. 理解系统结构设计为什么要解决好软件的可移植性。掌握实现软件移植的三种途径(即统一高级语言、采用系列机、模拟和仿真)各自的方法、适用场合、存在问题及应采取的对策。领会系列机软件所谓向前、向后、向下、向上兼容的定义,以及系列机对软件兼容的基本要求。能正确判断在系列机中发展新型号机器时,哪些做法是可取的。

5. 了解应用和器件的发展对系统结构设计的影响。理解非用户片、现场片和用户片的定义,以及器件发展是如何改变了逻辑设计的传统做法。

6. 领会并行性的定义,并行性的二重含义和开发并行性的三种途径。掌握各种并行性等级的划分和并行性级别高低的顺序。了解计算机系统沿三种不同的并行性发展途径开发出的多机系统类型和特点。了解多机系统的耦合度概念。了解计算机按指令流数据流及其多倍性进行分类的方法及典型机器结构的例子。

## 2.1.2 内容要点

### 1. 计算机系统的多级层次结构

从功能上可把现代通用计算机系统看成由多个机器级组成的层次结构。按层次结构中的由高到低的次序分别是应用语言机器级、高级语言机器级、汇编语言机器级、操作系统机器级、传统机器语言机器级和微程序机器级。

对每一个机器级的用户来说都可被看成是一台独立的机器,都有其自己的机器语言。各个机器级的实现技术不外乎采用的是翻译或是解释技术,或者是这两种技术的结合。采用翻译技术实现的典型例子有:用编译程序实现将高级语言源程序转换成机器语言目标程序,用汇编程序实现将汇编语言源程序转换成机器语言目标程序。解释实现的典型例子有,用解释方式在传统机器上执行高级语言程序,用微指令程序解释实现机器指令,用微程序或机器指令程序解释实现操作系统的操作命令等。

软件和硬件在逻辑功能上是等效的,但是,在性能、价格、实现的难易程度上都是各不相同的,是不等效的。

### 2. 计算机系统结构、组成与实现

计算机系统结构是指多级层次结构中传统机器级的结构,它是软件和硬件/固件的主要交界面,是让编制的机器语言程序、汇编语言源程序以及将高级语言源程序编译生成的机器语言目标程序能在机器上正确运行所应看到的计算机属性。计算机系统结构是与汇编语言程序或机器语言程序所能实现的功能要用的数据类型、寻址方式等密切相关。

计算机组成主要指的是机器级内部数据流和控制流的组成及逻辑设计。它与指令和编程等没有直接关系,主要是看硬件系统在逻辑上如何组织。计算机组成主要与计算机操作的并行度、重叠度、部件的共享度等有关,直接影响系统的速度和价格。

计算机实现指的是计算机组成的物理实现,着眼于用什么样的器件技术和微组装技术。它也直接影响到系统的速度和价格。

例如,机器中应设哪些机器指令和汇编指令,主存的容量和编址方式,寄存器的数量和使用规定等均由计算机系统结构设计来确定。而指令中微操作顺序的编排,主存是否采用多体并行交叉组织是计算机组成设计考虑的。是否采用超大规模集成电路,如何将部件在物理上组装到一起,则是由计算机实现设计来考虑了。

教材中列举了计算机系统结构、计算机组成和计算机实现各自应研究的一些内容和方面,它们都是对各个设计不透明的方面。这里需要指出,计算机中的“透明”与社会生活中的透明含义恰好相反。社会生活中所称的“透明”,是要公开的意思,让人们看得到,而计算机中的“透明”,则是指应看不到的意思。如同玻璃瓶中装着东西,但从某个角度去看,透过瓶子,看不到

瓶子里所装的东西。

对目前多数的通用计算机来说,采用什么系列机,机器级和汇编级的指令系统,指令的条数、种类、功能、格式和编码,主存的容量、编址空间和所用的编址方式,硬件直接识别的数据类型、格式和种类,I/O系统采用通道处理机还是外围处理机,I/O设备的编址,I/O接口的使用规定等,对计算机系统结构都设计成不透明的。而系列机内部搞哪几种型号的计算机,指令的解释采用顺序、重叠还是流水,乘法指令是用加法器和移位器经一连串时钟脉冲控制实现其操作还是用专门的高速乘法器来实现,主存采用单体还是多体并行交叉组织,数据总线线数的多少即数据通路宽度的大小,通道采用结合型还是独立型,系统采用单总线还是多总线,控制器微操作信号是用微程序控制器产生还是用组合逻辑电路控制器产生等,所有这些对计算机组成设计来说都应是不透明的。

### 3. 软硬件取舍与计算机系统思路

计算机系统结构设计主要是确定软件和硬件的功能分配。在计算机系统上,一个功能用硬件实现可以提高其执行的速度,减少程序所需要的存贮空间,降低软件部分所需的成本,但会提高硬件部分的成本,降低硬件的时间利用率和系统的灵活性及适应性。因此,在确定计算机系统软硬件的功能分配比例时,应考虑在现有的硬件和器件条件下,如何使系统有高的性能价格比。从降低实现费用来分析,如果一个功能是经常用的基本单元功能,且是属固定不变的功能,才适合于采用硬件实现;而对产量大的计算机系统,增大硬件功能实现的比例才是有利的。我们不能去盲目追求扩大硬件功能实现的比例。确定软、硬件功能分配时,还应考虑不能过多地限制各种组成和实现技术的采用。也还要考虑如何为编译和操作系统的实现、高级语言的编程等提供更多更好的硬件支持,以便缩短高级语言与机器语言、操作系统与计算机系统结构、程序设计环境与计算机系统结构之间的语义差距。否则,语义差距过大,软件实现的比例就会过大,将会使系统效率过分下降,这不适应当今计算机软件 and 硬件发展的状况。

从多级层次结构上考虑,计算机系统的设计宜采用从中间开始向两边进行设计。这样,可以避免“由上往下”和“由下往上”地设计所带来的软件、硬件设计脱节的现象。既考虑能拿到的硬件、器件,又考虑应用中可能要用到的算法和数据结构,同时,还要考虑如何为操作系统、编译系统的实现提供更好的硬件支持,先确定好软件和硬件功能分配的界面。然后,再分头并行设计硬件和软件。这样做,不仅有利于缩短系统的研制时间,也有利于硬件和软件设计人员之间的交流协调,使软、硬件之间的功能分配更为合理,系统性能价格比更高。

### 4. 结构设计要解决好软件的可移植性

系统结构设计应注意解决让过去系统上的大量成熟可靠的应用软件能在新的系统结构的机器上继续得到使用,使同一软件可应用于不同的硬件环境。这样,就可以大大减轻软件设计的负担,又能使新的系统立即发挥出效能。

实现软件移植的基本技术有统一高级语言、采用系列机、模拟和仿真等。

统一高级语言是设计一种对各种应用领域都能比较高效通用的高级语言。这样,在结构相同以至完全不同的机器之间,通过配以不同的语言翻译程序就可以实现高级语言应用软件的移植。然而,不同的用途要求高级语言的语法和语义结构差别较大;人们对统一的高级语言应当有什么样的基本结构看法不一;厂家为了便于在机器上高效地翻译,在高级语言中引入了



方言;用户为节省程序空间和提高其运行速度,经常在高级语言源程序中嵌入了汇编语言或其它语言的程序;用户的习惯势力,不愿使用新的语言。所有这些因素,使得在近期内难以统一出一种通用的高级语言。对此技术存在的问题采取的对策是,从长远的目标,还是要争取统一出一种通用的高级语言,但近期只能做相对的统一。

采用系列机的技术可以使同一系列内各档机器在汇编语言上实现统一,又能使系列内,发展出多种新的机器。因此,它适合于在结构相同或相近的机器之间实现汇编语言应用软件和部分系统软件的移植。采用系列机可以较好地解决软件设计环境要求相对稳定与硬件、器件、组成等技术飞速发展的矛盾。软件可以继续丰富和积累,又能不断更新器件、硬件和组成,使之短期内就能提供出性能更好、价格更便宜的新机器。

系列机应用软件的兼容从速度和性能上有向上兼容和向下兼容之分外,在产品进入市场的前和后上,又有向前兼容和向后兼容之分。系列机必须保证应用软件向后兼容,力争做到向上兼容。因为现在编制的应用程序,以后都能在新的机器上使用,这是系列机软件兼容的根本,至于其后研制的软件完全可以发展,不一定非要向前兼容。从用户观点来讲,当然希望在低档机器上的软件能照搬于高档机器上运行,因此,应力争做到向上兼容。

由于系列机结构变化有限,所以到一定时候会阻碍该系列的发展。对此问题采取的对策应当是不能只局限于旧系列的发展,在适当时候应推出新的系列结构。

模拟和仿真能在结构不同的机器之间实现机器语言程序的移植。用机器语言程序解释另一机器的机器指令实现软件移植的方法称为模拟。模拟方法在机器指令系统差异比较大的时候,会使程序运行速度严重下降。用微程序直接解释另一机器的机器指令实现软件移植的方法称为仿真。仿真可以提高被移植软件的运行速度,但机器结构差异较大时,很难仿真。因此,对此问题采取的对策是在不同系列机器间的软件移植时,将模拟和仿真二种技术结合起来使用。让频繁使用并容易仿真的这部分机器指令采用仿真,以提高速度,而让很少使用,对速度要求不高的、难以仿真的这部分指令及 I/O 操作采用模拟实现。

## 5. 应用与器件的发展对系统结构的影响

维持价格、提高性能和保持性能、降低价格是多数计算机生产厂家所采取的二种不同的发展途径。其中,后一途径更能促进产品的销售、推广和应用。从系统结构的观点来看,让低档(型)机器能引用甚至照搬高档(型)机器上的结构和组成,使新的微、小型机上具有原中、大型机的性能,用户是欢迎的。而巨、大型机上则一般采取维持价格、提高性能,或是提高价格、提高性能的途径来发展新的结构和组成技术。要想维持微型机和小型机的价格,又能提高其性能,可采取的一种办法是用选购件或扩展部件来解决。系统结构设计应注意针对新的用途,不断研制出新的结构,同时注意在维持软件兼容的前提下,又能将这些研究成果尽快搬到批量大的通用机上去。

器件发展是系统结构和组成发展的关键因素。结构和组成技术的发展又会反过来促进器件技术的进一步发展。器件已从非用户片发展出现场片和用户片。逻辑设计已由过去传统上追求的逻辑化简,转变成强调在满足系统结构所提出的功能要求前提下,如何能够用得上大批量生产的高集成度片子,提高其系统效能,缩短其研制周期,降低其生产成本。计算机的设计也已从过去只进行全硬的逻辑设计发展到现在所用的软硬结合方法进行计算机的辅助设计和辅助制造。

## 6. 系统结构中的并行性发展及计算机系统的分类

研究改进计算机系统结构的一个主要方面是如何开发出并行性。并行性是指问题中具有可同时进行运算或操作的特性。开发并行性的目的是为了能予以并行处理,以提高解题效率。并行性包括同时性和并发性。二个或多个事件在同一时刻发生属于同时性,而在同一时间间隔内发生则属于并发性。

并行性有不同的等级,而且,从不同的角度看,并行性等级的分法也不相同。从计算机系统中执行程序的角度看,并行等级由低到高,分别是指令内各微操作之间的并行,多条指令之间的并行,多个任务或进程之间的并行以及多个作业或程序之间的并行等。从计算机系统处理数据的角度看,并行性等级由低到高,分别是位串字串(串行单处理机,无并行性),位并字串(传统并行单处理机),位片串字并和全并行等。从计算机信息加工步骤和阶段的角度看,并行性等级又有存贮器操作并行(并行存贮器、相联处理机),处理器操作步骤并行(流水线处理机),处理器操作并行(阵列处理机),指令、任务、作业间的全面并行(多处理机、分布处理系统、计算机网络)等。

时间重叠、资源重复和资源共享受是开发并行性的三个主要途径。时间重叠是在并行性中引入时间的因素。它是让多个处理过程在时间上错开,轮流重叠地使用同一套硬件设备的各个部分,使之加快硬件使用的周转来赢得速度,典型的例子是流水。资源重复是在并行性中引入空间的因素。它是靠重复设置硬件资源来提高可靠性或性能,典型的例子是双工系统、相联处理机和阵列处理机等。资源共享是用软件方法让多个用户共用同一套资源,通过提高系统资源的利用率来提高系统的性能和效率,典型的例子是多道程序分时系统、计算机网络和分布处理系统等。

由单处理机向现代高性能并行的多处理机系统发展的过程来看,沿时间重叠途径,让多个处理机进行宏流水,构成的多处理机一般都是非对称型或异构型的;沿资源重复途径,构成的相联处理机和阵列处理机都是对称型或同构型的多处理机;沿资源共享途径发展的多处理机则既可以是同构型的,也可以是异构型的。

我们可以用耦合度来反映多机系统中各处理机之间物理连接的紧密程度和交叉作用能力的强弱程度。多机系统的耦合度可以有最低耦合、松散耦合和紧密耦合之分。处理机之间没有物理上的连接,最多只是通过盘、带等存贮介质来联系的脱机系统属于最低耦合系统。处理机之间通过通道或低速通信线路互连,共享一部分外围设备的系统属于松散(或间接)耦合系统。而经高速总线或开关互连,共享主存的多处理机系统则属于紧密(或直接)耦合系统。

20世纪90年代以来,计算机系统最主要的发展是开发大规模并行处理(MPP)。其中,多处理机和多计算机是其研究和开发的热点。它们可以由数十至上千台微处理机构成的MPP系统,也可以是由多个高性能工作站或高档微型机使用高速通信网络互连进行高效并行处理的机群系统。重大挑战性科学计算题目要求计算机系统能有3T性能,即1TFLOPS的计算能力,1TBYTE的主存容量和1TBYTE/S的I/O带宽。

计算机系统可按指令流和数据流的多倍性分类成单指令流单数据流(SISD)、单指令流多数据流(SIMD)、多指令流单数据流(MISD)和多指令流多数据流(MIMD)四类。传统单处理机属SISD系统,阵列处理机和相联处理机及流水线单处理机均属SIMD系统,宏流水及脉动阵列机属MISD系统,能全面实现作业、任务、指令、数组各级并行的多处理机、多计算机系统则属

于 MIMD 系统。

### 2.1.3 习题和题例的分析与解答

**【题 1.1】**有一台经解释实现的计算机,可以按功能划分成 4 级。每一级为了执行一条指令需要下一级的  $N$  条指令来解释。如果执行第 1 级的一条指令要  $K$  ns 时间,那么执行第 2、3、4 级的一条指令各需要用多少时间?

**【分析】**因为计算机按功能分成级时,最底层是第 1 级,向上依次为第 2、第 3 和第 4 级。解释方式的执行是在低级机器级上用它的一串语句或指令来仿真高级机器级上的一条语句或指令的功能。而且是通过高级机器级程序中的每条语句或指令逐条加以解释来实现的。

**【解】**执行第 2、3、4 级的一条指令分别需  $KN$  ns、 $KN^2$  ns、 $KN^3$  ns 的时间。

**【题 1.2】**操作系统机器级的某些指令就采用传统机器级的某些指令,这些指令可以直接用微程序解释实现,而不必由操作系统自己来实现。你认为这样做有哪些好处?请答出主要的两点即可。

**【解】**这样做,可以加快操作系统中操作命令解释的速度,同时也节省了存放解释操作命令这部分解释程序所占的存贮空间,简化了操作系统机器级的设计,也有利于减少传统机器级的指令条数。

**【题 1.3】**有一个计算机系统可按功能分成 4 级,每级的指令互不相同,每一级的指令都比其下一级的指令在效能上强  $M$  倍,即第  $i$  级的一条指令能完成第  $i-1$  级的  $M$  条指令的计算量。现若需第  $i$  级的  $N$  条指令解释第  $i+1$  级的一条指令,而有一段第 1 级的程序需要运行  $K$  s,问在第 2、3 和 4 级上一段等效程序各需要运行多长时间?

**【分析】**因为从功能意义上讲,第  $i$  级的一条指令能完成第  $i-1$  级的  $M$  条指令的计算量,但第  $i$  级的一条指令的执行,却是靠第  $i-1$  级的  $N$  条指令的执行来解释完成。已知,第 1 级的一段程序运行时间为  $K$  s。第 2 级的一段程序从功能等效上讲所需的指令条数应当是第 1 级上指令数的  $1/M$ 。而由第 1 级解释时又需要执行  $N$  条指令,所以,第 2 级一段等效程序就需要  $K \cdot \frac{N}{M}$  s 的时间。第 3、4 级则可依次类推。

**【解】**第 2、3 和 4 级上的一段等效程序分别需要  $K \frac{N}{M}$  s、 $K \frac{N^2}{M^2}$  s 和  $K \frac{N^3}{M^3}$  s 的时间。

**【题 1.4】**硬件和软件在什么意义上是等效的?在什么意义上又是不等效的?试举例说明。

**【解】**硬件和软件在逻辑功能上是等效的。在原理上,用软件实现的功能完全可以用硬件或固件(微程序解释)来完成。用硬件实现的功能也可以通过用软件进行模拟来完成。只是反映在速度、价格、实现的难易程度上这两者是不同的。

例如,编译程序、操作系统等许多用机器语言软件子程序实现的功能完全可以用组合电路硬件或微程序固件来解释实现。它们的差别只是软件实现的速度慢,软件的编制复杂,编程工作量大,程序所占的存贮空间量较多,这些都是不利的;但是,所花硬件少,硬件实现上也就因此而简单容易,硬件的成本低,解题的灵活性和适应性较好,这些都是有利的。又如,乘除法运算可以经机器专门设计的乘法指令用硬件电路或乘除部件来实现,也可以通过执行一个使用



相加、移位、比较、循环等机器指令组成的机器语言子程序来实现。向量、数组运算在向量处理机中是直接使用向量、数组类指令和流水或阵列等向量运算部件硬的方式来实现,但在标量处理机也可以通过执行用标量指令组成的循环程序软的方式来完成。

浮点数运算可以直接通过设置浮点运算指令用硬件来实现,也可以将二个定点数分别表示浮点数的阶码和尾数,通过程序方法把浮点数阶码和尾数的运算映象变换成二个定点数的运算,用子程序软的方式来实现。十进制数的运算可以通过专门设置十进制运算类指令和专门的十进制运算部件硬的方式完成,或者通过设置BCD数的表示和若干BCD数运算的校正指令来软硬结合地实现,也可以先经10转2的数制转换子程序将十进制数转成二进制数,再用二进制运算类指令运算,所得结果又调用2转10的数制转换子程序转换成十进制数结果来全软方式实现。

**【题 1.5】**试以实例说明计算机系统结构、计算机组成与计算机实现之间的相互关系与影响。

**【解】**可参看教材 1.2.2 节的内容,概要进行解答。略。

**【题 1.6】**什么是透明性概念?对计算机系统结构,下列哪些是透明的?哪些是不透明的?

存贮器的模  $m$  交叉存取;浮点数据表示;I/O 系统是采用通道方式还是外围处理机方式;数据总线宽度;字符行运算指令;阵列运算部件;通道是采用结合型还是独立型;PDP-11 系列的单总线结构;访问方式保护;程序性中断;串行、重叠还是流水控制方式;堆栈指令;存贮器的最小编址单位;Cache 存贮器。

**【分析】**所谓透明就是看不到,不属于其管的部分。对计算机系统结构是否透明,首先要弄清教材 1.2.1 节中有关计算机系统结构的定义和所包含的属性内容。简单来说,凡是编写机器语言和汇编语言程序要用到的数据表示、指令系统、寻址方式、寄存器组织、机器级 I/O 结构、存贮容量及其编址方式、中断机构、系统管态和目态间的切换、信息保护方式和机构等都对计算机系统结构是不透明的。而全部由硬件实现的,或是在机器语言、汇编语言编程中不会出现和不需要了解的部分,以及只影响机器的速度和价格的逻辑实现(计算机组成)和物理实现(计算机实现)的那些部分,对计算机系统结构都是透明的。

**【解】**客观存在的事物或属性,从某个角度去看,却看不到,称这些事物和属性对他是不透明的。透明了就可以简化这部分的设计,然而因为透明而无法控制和干预,这会带来不利的。因此,透明性的取舍要正确选择。

对计算机系统结构,透明的应当是:存贮器的模  $m$  交叉存取,数据总线宽度,阵列运算部件,通道是采用结合型还是独立型,PDP-11 系列的单总线结构,串行、重叠还是流水控制方式,Cache 存贮器。

对计算机系统结构不透明的是:浮点数据表示,I/O 系统采用通道方式还是外围处理机方式,字符行运算指令,访问方式保护,程序性中断,堆栈指令,存贮器最小编址单位。

**【题 1.7】**从机器(汇编)语言程序员看,以下哪些是透明的?

指令地址寄存器;指令缓冲器;时标发生器;条件码寄存器;乘法器;主存地址寄存器;磁盘外设;先行进位链;移位器;通用寄存器;中断字寄存器。

**【分析】**从机器(汇编)语言程序员看,实际上也就是从计算机系统结构看的内容。

指令地址寄存器就是程序计数器,汇编语言或机器语言程序都要用到它的,其位数多少会影响到可执行程序的空间大小。指令缓冲器、主存地址寄存器都属于计算机组成中的缓冲器