

尚德机构

计算机系统结构

讲师：孙小涵

学习是一种信仰！ IN LEARNING WE TRUST

SUNLAND





讲师介绍

- 主讲老师：孙小涵（尚德机构-小涵老师）
- 主讲课程：计算机类、数学类
- 邮箱：sunxiaohan@sunlands.com



课程章节

计算机系统结构

第1章 计算机系统结构概论

第2章 数据表示、寻址方式与指令系统

第3章 存储、中断、总线与I/O系统

第4章 存储体系

第5章 标量处理机

第6章 向量处理机

第7章 多处理机

第8章 数据流计算机和归约机



第7章 多处理机

第7章 多处理机

●	多处理机的概念、问题和硬件	★
●	紧耦合多处理机多cache的一致性问题	★
●	多处理机的并行性和性能	★ ★
●	多处理机的操作系统	
●	多处理机的发展	

7. 3多处理机的概念、问题和硬件结构

本节主要内容：

并行算法的研究思路

给出表达式，画出串行运算树和并行运算树，计算 P 、 T_1 、 T_p 、 S_p 和 E_p

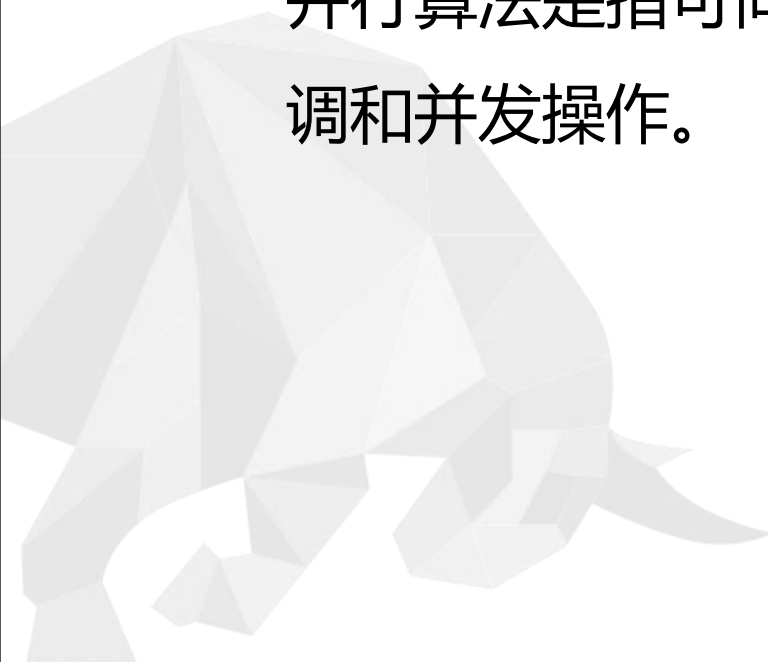
给出程序中的语句或指令，分析并行情况

给出计算式或高级语言源程序，加FORK、JOIN、GOTO语句

7.3.1 并行算法

1. 并行算法的定义和分类（简单了解）

并行算法是指可同时执行的多个进程的集合，各进程可相互作用、协调和并发操作。



7.3.1 并行算法

1. 并行算法的定义和分类（填空）

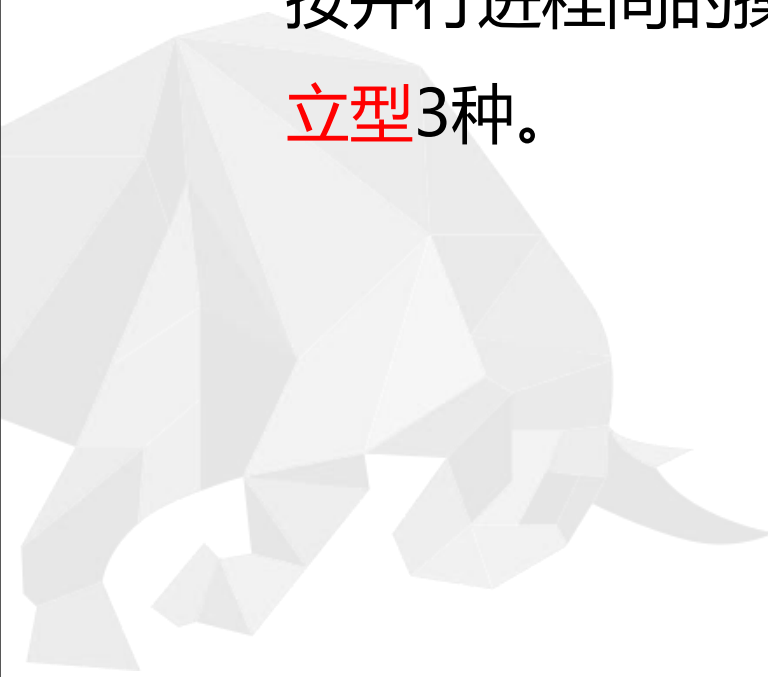
按运算基本对象，并行算法可分为数值型的和非数值型两类。



7.3.1 并行算法

1. 并行算法的定义和分类（单选、填空）

按并行进程间的操作顺序不同，并行算法又分为同步型、异步型和独立型3种。



7.3.1 并行算法

1. 并行算法的定义和分类（单选）

根据各处理机计算任务的大小（即任务粒度）不同，并行算法又分为细粒度、中粒度和粗粒度3种。

细粒度并行算法一般指**向量或循环级**的并行。

中粒度并行算法一般指**较大的循环级**并行，并确保这种并行的好处可以补偿因并行带来的额外开销。

粗粒度并行算法则一般是指**子任务级**的并行。

7.3.1 并行算法

1. 并行算法的定义和分类（单选）

根据各处理机计算任务的大小（即任务粒度）不同，并行算法又分为细粒度、中粒度和粗粒度3种。

细粒度并行算法一般指（ ）的并行。

中粒度并行算法一般指较大的循环级并行，并确保这种并行的好处可以补偿因并行带来的额外开销。

粗粒度并行算法则一般是指（ ）的并行。

7.3.1 并行算法

1. 并行算法的定义和分类（单选）

根据各处理机计算任务的大小（即任务粒度）不同，并行算法又分为细粒度、中粒度和粗粒度3种。

细粒度并行算法一般指**向量或循环级**的并行。

中粒度并行算法一般指**较大的循环级**并行，并确保这种并行的好处可以补偿因并行带来的额外开销。

粗粒度并行算法则一般是指**子任务级**的并行。



那么意气风发地
走在成功的道路上

真题练练手

1、按运算基本对象，并行算法可分为（ ）型和（ ）型两类。1904



那么意气风发地
走在成功的道路上

真题练练手

1、按运算基本对象，并行算法可分为（ ）型和（ ）型两类。1904

答案：数值型 非数值型



那么意气风发地
走在成功的道路上

真题练练手

2、在多台处理机上，有效计算的执行时间 E 与处理机机间的通讯辅助开销时间 C 的比值较小时，任务宜采用（ ）粒度。0904



那么意气风发地
走在成功的道路上

真题练练手

2、在多台处理机上，有效计算的执行时间 E 与处理机机间的通讯辅助开销时间 C 的比值较小时，任务宜采用（ ）粒度。0904

答案：粗

7.3.1 并行算法

2. 多处理机并行算法的研究思路

为了评价所提出的并行算法的性能效率，用 P 表示可并行处理的处理机机数；

用 T_p 表示 P 台处理机运算的级数，即树高；

用多处理机的加速比 S_p ，表示单处理机顺序运算的级数 T_1 与 P 台处理机并行运算的级数 T_p 之比；

用 E_p 表示 P 台处理机的设备利用率（效率），可见， $S_p \geq 1$ 时，会使 $E_p \leq 1$ ，即运算的加速总是伴随着效率的下降。

7.3.1 并行算法

2. 多处理机并行算法的研究思路

【1704真题】由霍纳法则给定的表达式如下: $E = a(bc + d(ef + g(h + ij)))$

利用减少树高的办法来加速运算,要求:

- 1、画出树形流程图;
- 2、确定 T_p 、 P 、 S_p 、 E_p 的值。

7.3.1 并行算法

2. 多处理机并行算法的研究思路

【1704真题】由霍纳法则给定的表达式如下: $E = a(bc + d(ef + g(h + ij)))$

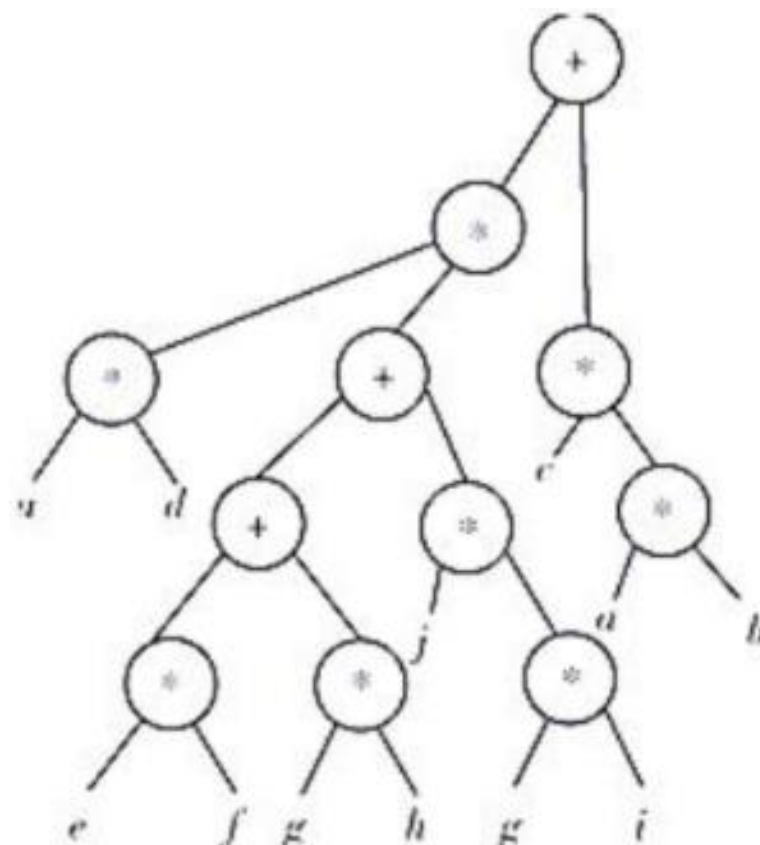
利用减少树高的办法来加速运算,要求:

1、画出树形流程图;

2、确定 T_p 、 P 、 S_p 、 E_p 的值。

1、若处理机为单处理机, $T_1 = 9$, 改成以下形式:
 $E = abc + ad(ef + gh + gij)$ 。树形流程图如图所示。

2、 $T_p = 5$ 、 $P = 3$ 、 $S_p = T_1/T_p = 9/5$ 、 $E_p = 3/5$ 。



7.3.2程序并行性的分析

1.数据相关

如果 P_i 的左部变量在 P_j 的右部变量集内，且 P_j 必须取出 P_i 运算的结果来作为操数，就称 P_j “数据相关”于 P_i 。

$$P_i \quad A = B + D$$

$$P_j \quad C = A * E$$

相当于流水中发生的“先写后读”相关。顺序串行运行的正确结果应当是

$$P_i \quad A_{\text{新}} = B_{\text{原}} + D_{\text{原}}$$

$$P_j \quad C_{\text{新}} = A_{\text{新}} * E_{\text{原}} = (B_{\text{原}} + D_{\text{原}}) * E_{\text{原}}$$

7.3.2程序并行性的分析

1.数据相关

如果让 P_i 和 P_j 并行， P_j 的C新成了 $A_{原} * E_{原}$ ，显然不是应有的结果，因此， P_i 和 P_j 是不能并行的。如果将 P_i 和 P_j 的执行顺序颠倒，交换串行，即先执行 P_j ，而后再执行 P_i ，很明显同样也得不到应有的正确结果。如果能够交换串行，就可以让空闲处理机先去执行 P_j ，从而有利于宏观上提高各个程序段间的并行，加快作业执行的速度，改进系统的运行效率。然而，有一种特殊情形，即当 P_i 和 P_j 服从交换律时，如

$$P_i \quad A = 2 * A$$

$$P_j \quad A = 3 * A$$

虽不能并行执行，却允许它们交换串行。最终 $A_{新} = 6 * A_{原}$ 和顺序执行的结果一致。

7.3.2程序并行性的分析

2.数据反相关

如果 P_j 的左部变量在 P_i 的右部变量集内，且当 P_i 未取用其变量的值之前，是不允许被 P_j 所改变的，就称 P_j “数据反相关”于 P_i 。例如

$$P_i \quad C = A + E$$

$$P_j \quad A = B + D$$

7.3.2程序并行性的分析

2.数据反相关

相当于流水中发生的“先读后写”相关。顺序串行运行的正确结果应是

$$P_i \quad C_{\text{新}} = A_{\text{原}} + E_{\text{原}}$$

$$P_j \quad A_{\text{新}} = B_{\text{原}} + D_{\text{原}}$$

可见，当 P_i 与 P_j 并行时，只要硬件上能保证 P_i 对相关单元 A 先读出，就能得到正确的结果。然而，若将 P_i 和 P_j 交换串行就成了

7.3.2程序并行性的分析

2.数据反相关

可见，当 P_i 与 P_j 并行时，只要硬件上能保证 P_i 对相关单元 A 先读出，就能得到正确的结果。然而，若将 P_i 和 P_j 交换串行就成了

$$P_j \quad A_{\text{新}} = B_{\text{原}} + D_{\text{原}}$$

$$P_i \quad C_{\text{新}} = A_{\text{新}} + E_{\text{原}} = B_{\text{原}} + D_{\text{原}} + E_{\text{原}}$$

而发生错误，所以是不能交换串行的。

7.3.2程序并行性的分析

3.数据输出相关

如果 P_i 的左部变量也是 P_j 的左部变量，且 P_j 存入其算得的值必须在 P_i 存入之后，则称 P_j “数据输出相关”于 P_i 。

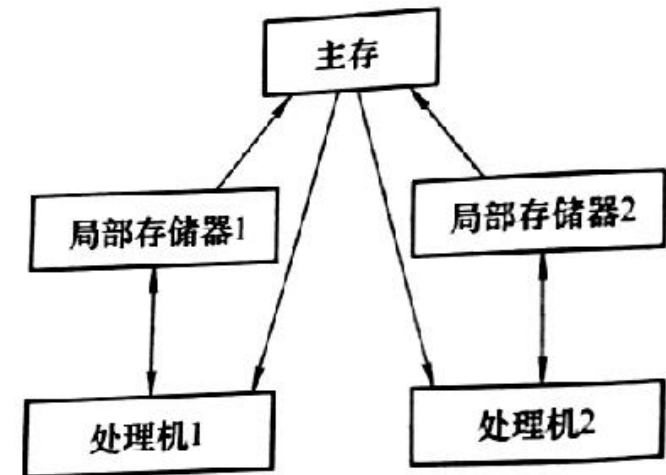


图7-13 能保证读写次序的多处理机结构

7.3.2程序并行性的分析

3.数据输出相关（单选）

结论：两个程序段之间若有先写后读的数据相关，不能并行，只在特殊情况下可以交换串行；若有先读后写的数据反相关，可以并行执行，但必须保证其写入共享主存时的先读后写次序，不能交换串行；若有写-写的数据输出相关，可以并行执行，但同样需保证其写入的先后次序，不能交换串行；若同时有先写后读和先读后写两种相关，以交换数据为目的时，必须并行执行，且读、写要完全同步，不许顺序串行和交换串行；若没有任何相关或仅有源数据相同时，可以并行、顺序串行和交换串行。



那么意气风发地
走在成功的道路上

真题练练手

1、在多台处理机上，两个程序段既能顺序串行、交换串行，又能并行，则这两个程序段之间必须是()0904

A:只有数据相关

B:只有源数据相关

C:只有数据反相关

D:只有数据输出相关



那么意气风发地
走在成功的道路上

真题练练手

1、在多台处理机上，两个程序段既能顺序串行、交换串行，又能并行，则这两个程序段之间必须是()0904

A:只有数据相关

B:只有源数据相关

C:只有数据反相关

D:只有数据输出相关

答案： B

7.3.3 并行语言与并行编译

FORK语句的形式为FORK m，其中m为新进程开始的标号。执行FORK m语句时，派生出标号为m开始的新进程，具体为：准备好这个新进程启动和执行所必需的信息；

如果是共享主存，则产生存储器指针、映像函数和访问权数据；将空闲的处理机分配给派生的新进程，如果没有空闲处理机，则让它们排队等待；继续在原处理机上执行FORK语句的原进程。

7.3.3 并行语言与并行编译

与FORK语句相配合，作为每个并发进程的终端语句JOIN的形式为JOIN n ，其中 n 为并发进程的个数。JOIN语句附有一个计数器，其初始值为0。每当执行JOIN n 语句时，计数器的值加1，并与 n 比较。若比较相等，表明这是执行中的第 n 个并发进程经过JOIN 语句，于是允许该进程通过JOIN语句，将计数器清0,并在其处理机上继续执行后续语句；若比较不等，计数器的值仍小于 n ，表明此进程不是并发进程的最后一个，可让现在执行JOIN语句的这个进程先结束，把它所占用的处理机释放出来，分配给正在排队等待 其他任务。如果没有排队等待的任务，就让该处理机空闲。

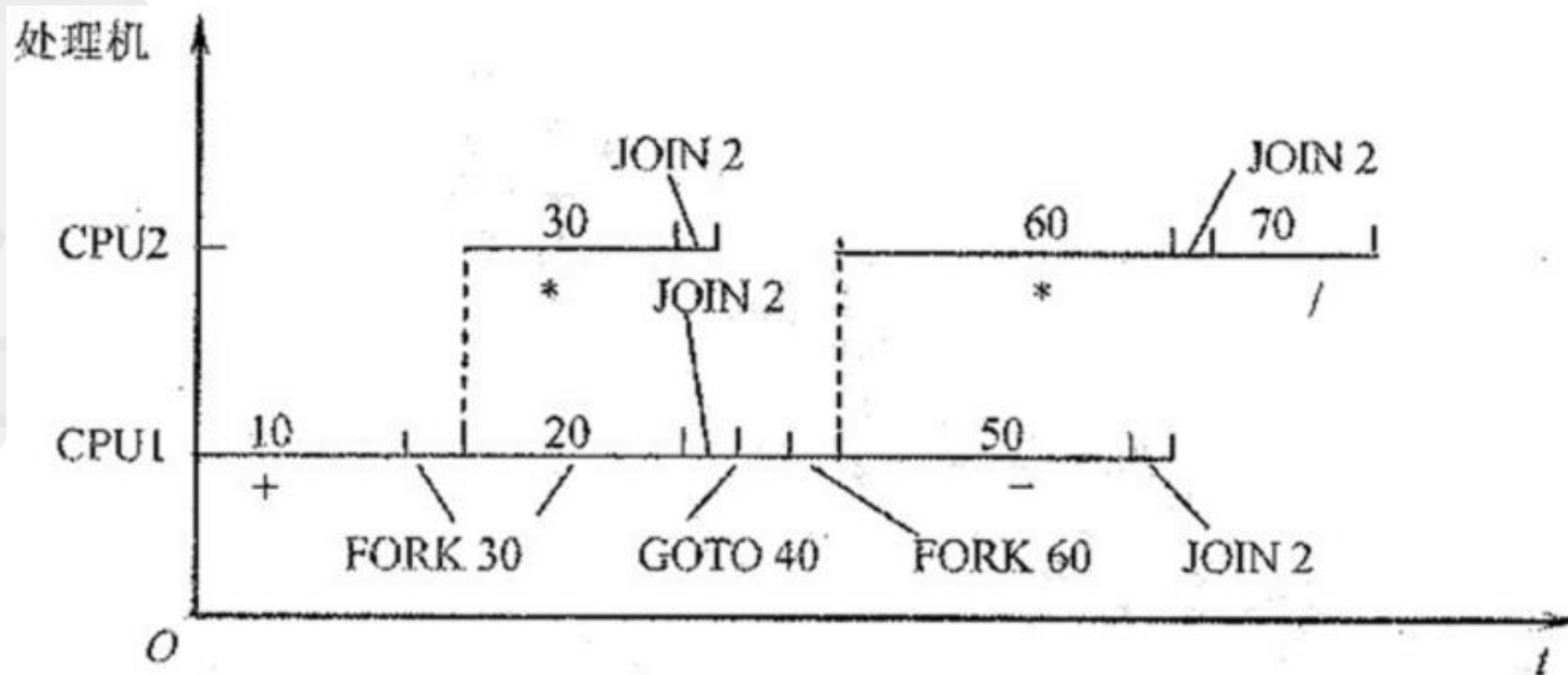
7.3.3 并行语言与并行编译

【1604真题】有以下FORK、JOIN写成的在多台处理机上并行执行的程序：1. 假设现为两台处理机,除法速度最慢,加、减法速度最快,画出该程序在两台处理机上运行时的资源时间图。

```
10 U = A + B
   FORK 30
20 V = U/B
   JOIN 2
   GOTO 40
30 W = A * U
   JOIN 2
40 FORK 60
50 X = W - V
   JOIN 2
   GOTO 70
60 Y = W * U
   JOIN 2
70 Z = X/Y
```

7.3.3 并行语言与并行编译

【1604真题】有以下FORK、JOIN写成的在多台处理机上并行执行的程序：1. 假设现为两台处理机,除法速度最慢,加、减法速度最快,画出该程序在两台处理机上运行时的资源时间图。



7.3.4多处理机的性能

任务粒度 (Task Granularity)的大小会显著影响多处理机的性能和效率。

任务粒度过小，辅助开销大，系统效率低；

任务粒度过大，并行度低，性能不会很高。

7.4 多处理机的操作系统

本节主要内容：

主从型操作系统的定义、特点和适用场合

各自独立型操作系统的定义、特点和应用场合

浮动型操作系统的定义、特点和应用场合

7.4多处理机的操作系统

分类：（填空）

多处理机操作系统有3类。它们是主从型（Master-Slave Configuration）、各自独立型（Separate Supervisor）及浮动型（Floating Supervisor）。

7.4多处理机的操作系统

7.4.1主从型操作系统

1.优点（简答）

主从型操作系统的结构比较简单；整个管理程序只在一个处理机上运行，除非某些需递归调用或多重调用的公用程序，一般都不必是可再入的；只有一个处理机访问执行表，不存在系统管理控制表格的访问冲突和阻塞，简化了管理控制的实现。所有这些均使操作系统能最大限度地利用已有的单处理机多道程序分时操作系统的成果，只需要对它稍加扩充即可。因此，实现起来简单、经济、方便，是目前大多数多处理机操作系统所采用的方式。

7.4多处理机的操作系统

7.4.1主从型操作系统

2.缺点（简答）

对主处理机的可靠性要求很高。一旦发生故障，很容易使整个系统瘫痪，这时必须由操作员干预才行。如果主处理机不是设计成专用的，操作员可用其他处理机作为新的主处理机来重新启动系统。整个系统显得不够灵活，同时要求主处理机必须能快速执行其管理功能，提前等待请求，以便及时为从处理机分配任务，否则将使从处理机因长时间空闲而显著降低系统的效率。即使主处理机是专门的控制处理机，如果负荷过重，也会影响整个系统的性能。特别是当大部分任务都很短时，由于频繁地要求主处理机完成大量的管理性操作，系统效率将会显著降低。

7.4多处理机的操作系统

7.4.2各自独立型操作系统

1.优点（简答）

很适应分布处理的模块化结构特点，减少对大型控制专用处理机的需求；某个处理机发生故障，不会引起整个系统瘫痪，有较高的可靠性；每台处理机都有其专用控制表格，使访问系统表格的冲突较少，也不会有许多公用的执行表，同时控制进程和用户进程一起进行调度，能取得较高的系统效率。

7.4多处理机的操作系统

7.4.2各自独立型操作系统

2.缺点（简答）

实现复杂。尽管每台处理机都有自己的专用控制表格，但仍有一些共享表格，会增加共享表格的访问冲突，导致进程调度的复杂性和开销的加大。某台处理机一旦发生故障，要想恢复和重新执行未完成的工作较困难。每台处理机都有自己专用的输入/输出设备和文件，使整个系统的输入/输出结构变换需要操作员干预。各处理机负荷的平衡比较困难。各台处理机需有局部存储器存放管理程序副本，降低了存储器的利用率。

7.4多处理机的操作系统

7.4.3浮动型操作系统

1.优点（简答——没考过）

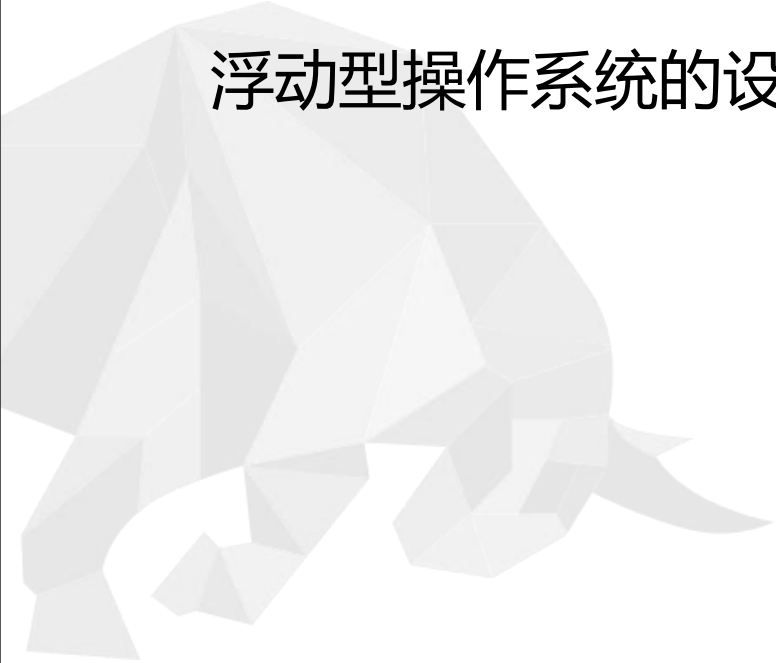
各类资源可以较好地做到负荷平衡。一些像I/O中断等非专门的操作可交由在某段时间最闲的处理机去执行。它在硬件结构和可靠性上具有分布控制的优点，而在操作系统的复杂性和经济性上则接近于主从型。如果操作系统设计得好，将不受处理机机数多少的影响，因而具有很高的灵活性。

7.4多处理机的操作系统

7.4.3浮动型操作系统

2.缺点（简答——没考过）

浮动型操作系统的设计最为困难。



7.5 多处理机的发展

本节主要内容：

多处理机发展的几种形式

大规模并行处理机MPP和机群系统的特点



7.5多处理机的发展

- 1) 分布式共享存储器多处理机
- 2) 对称多处理机
- 3) 多向量多处理机
- 4) 并行向量处理机
- 5) 大规模并行处理机 (MPP)
- 6) 机群系统

7.5多处理机的发展

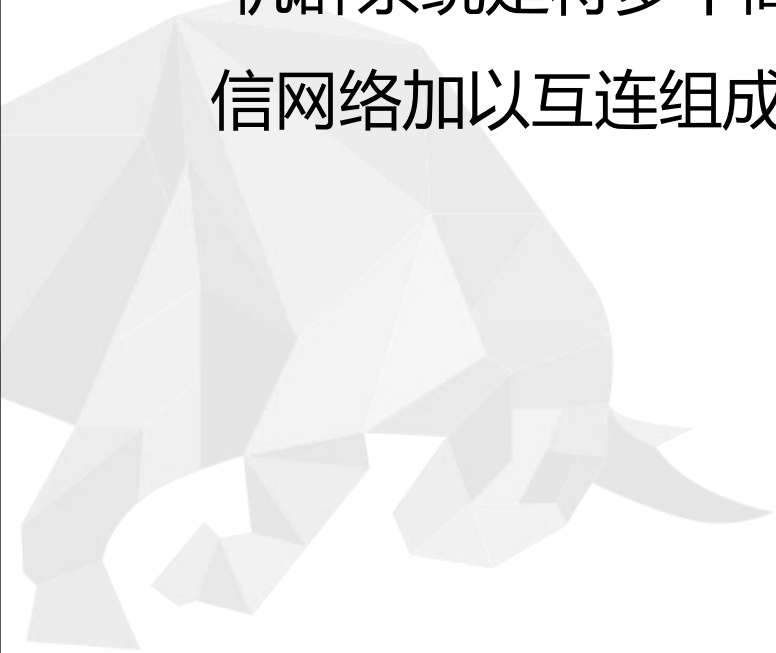
计算机系统的3T性能目标（补充知识点——填空）

- 1 TFLOPS的计算能力、
- 1 Tbyte的主存容量
- 1 Tbyte / s的I / O带宽。

7.5多处理机的发展

机群系统（简单了解）

机群系统是将多个高性能的工作站或高档微型计算机，使用高速的通信网络加以互连组成的系统。



7.5多处理机的发展

机群系统比起传统的并行处理系统有如下明显的优点：（单选、简答）

- 1) 系统有高的性能价格比。
- 2) 系统的开发周期短。
- 3) 系统的可扩展性好。
- 4) 系统的资源利用率高。
- 5) 用户投资风险小。
- 6) 用户编程方便。

（口诀：简单好用价格低）

7.5多处理机的发展

机群系统比起传统的并行处理系统有如下明显的优点：（单选、简答）

- 1) 系统有高的（ ）。
- 2) 系统的开发周期短。
- 3) 系统的（ ）好。
- 4) 系统的资源（ ）高。
- 5) 用户投资风险小。
- 6) 用户编程（ ）。

（口诀：简单好用价格低）

7.5多处理机的发展

机群系统比起传统的并行处理系统有如下明显的优点：（单选、简答）

- 1) 系统有高的性能价格比。
- 2) 系统的开发周期短。
- 3) 系统的可扩展性好。
- 4) 系统的资源利用率高。
- 5) 用户投资风险小。
- 6) 用户编程方便。

（口诀：简单好用价格低）



那么意气风发地
走在成功的道路上

真题练练手

1、多处理机的操作系统有（ ）、（ ）和浮动型三类。1104



那么意气风发地
走在成功的道路上

真题练练手

1、多处理机的操作系统有（ ） 、（ ） 和浮动型三类。1104

答案：主从型 各自独立型



那么意气风发地
走在成功的道路上

真题练练手

2、与传统的并行处理系统相比，机群系统所具有的特点不包括()1204

A:用户编程方便

B:系统的开发周期长

C:系统的资源利用率高

D:系统的可扩展性好



那么意气风发地
走在成功的道路上

真题练练手

2、与传统的并行处理系统相比，机群系统所具有的特点不包括()1204

A:用户编程方便

B:系统的开发周期长

C:系统的资源利用率高

D:系统的可扩展性好



那么意气风发地
走在成功的道路上

真题练练手

3、计算机系统的3T性能目标是（ ）的计算能力、（ ）的主存容量和1 Tbyte / s的I / O带宽。 0604



那么意气风发地
走在成功的道路上

真题练练手

3、计算机系统的3T性能目标是（ ）的计算能力、（ ）的主存容量和1 Tbyte / s的I / O带宽。 0604

答案：1 TFLOPS 1 Tbyte

第8章 数据流计算机和归约机

第8章 数据流计算机和归约机



数据流计算机



归约机

8.1 数据流计算机

本节主要内容：

工作原理

构形

特点

近年来的发展

8.1.1 数据驱动的概念

Von Neumann型计算机的基本特点是在程序计数器集中控制下，顺次地执行指令，因此它是以**控制流**（Control Flow）方式工作的。

8.1.1 数据驱动的概念

(填空)

数据驱动的数据流方式中，数据令牌是一种表示某一**操作数**或**参数**已准备就绪的标志。

8.1.1 数据驱动的概念

控制驱动的控制流方式的特点是：（简答）

通过访问共享存储单元让数据在指令之间传递；指令执行的顺序性隐含于控制流中，但却可以显式地使用专门的控制操作符来实现并行处理；指令执行的顺序受程序计数器控制，换句话说，是受控制令牌所支配的。数据驱动的数据流方式则不同，它没有通常的共享变量的概念，即没有共享存储数据的概念；指令执行序只受指令中数据相关性的制约；数据是以数据令牌方式直接在指令之间传递的。

8.1.1 数据驱动的概念

需求驱动计算模型

数据驱动计算，其操作是按输入数据可用性决定的次序进行的。

需求驱动计算，其操作则按数据需求所决定的次序进行。

8.1.1 数据驱动的概念

(填空)

从语义上讲，数据流是基于异步性 (Asynchrony) 和函数性 (Functionality) 的一种计算模型。

8.1.3数据流计算机的结构

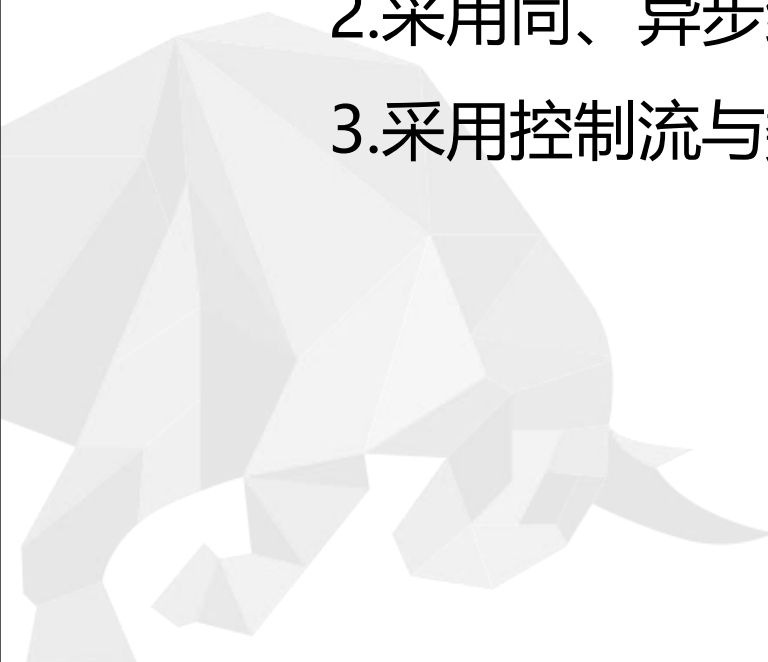
(填空)

根据对数据令牌处理的方式不同，可以把数据流计算机的结构分成静态和动态两类。



8.1.5数据流计算机的进展

- 1.采用提高并行度等级的数据流计算机
- 2.采用同、异步结合的数据流计算机
- 3.采用控制流与数据流相结合的数据流计算机



8.2归约机

(填空)

归约机和数据流计算机一样，都是基于数据流的计算模型，只是其采用的驱动方式不同。

数据流计算机采用数据驱动，执行的操作序列取决于输入数据的可用性；

归约机则是需求驱动，执行的操作序列取决于对数据的需求，对数据的需求又来源于函数式程序设计语言对表达式的归约（Reduction）。

8.2归约机（简答）

- 1)归约机应当是面向函数式语言，或以函数式语言为机器语言的非Neumann型机器，其内部结构应不同于Neumann型机器。
- 2)具有大容量物理存储器并采用大虚存容量的虚拟存储器，具备高效的动态存储分配和管理的软、硬件支持，满足归约机对动态存储分配及所需存储空间大的要求。
- 3)处理部分应当是一种有多个处理器或多个处理机并行的结构形式，以发挥函数式程序并行处理的特长。
- 4)采用适合于函数式程序运行的多处理器（机）互连的结构，最好采用树形方式的互连结构或多层次复合的互连结构形式。
- 5)为减少进程调度及进程间的通信开销，尽量把运行进程的结点机紧靠该进程所需用的数据安排，并使运行时需相互通信的进程所占用的处理机也靠近，让各处理机的负荷平衡。



那么意气风发地
走在成功的道路上

真题练练手

1、数据驱动的数据流方式中，数据令牌是一种表示某一（ ）或（ ）已准备就绪的标志。1904



那么意气风发地
走在成功的道路上

真题练练手

1、数据驱动的数据流方式中，数据令牌是一种表示某一（ ）或（ ）已准备就绪的标志。1904

答案：操作数 参数



尚德机构

▶ 答疑时间 ◀





尚德机构

▶ THANK YOU ◀

