

# 重庆大学

计算机学院《计算机系统结构》课程

## 期末复习题

2012 年 6 月 2 日

## Preface

本复习题根据包括“系统结构重点一成明版”、“Computer Architecture Exercise”、老师在课堂上勾画的重点内容在内的多种计算机学院与本课程相关的资料整理而成,对考试重点具有较高的覆盖率,并且笔者对其叙述不太清楚或详细的地方做了一些改动。由于部分资料原作者不明,在这里就不提供其信息了,同时向为大家整理复习资料的原作者表示谢意!考虑到时间有限,因此复习题中没有编排选择题,所以还请大家见谅!但复习题中的三种题型已经包含了全部重点内容,希望可以能够帮助大家。

由于水平有限,不排除资料里有少数错误的地方,因此欢迎大家批评指正: [jeancandoit@qq.com](mailto:jeancandoit@qq.com),祝大家复习顺利!

笔者

2012年6月2日

## Contents

系统结构基础知识.....	- 3 -
填空题 .....	- 3 -
问答题 .....	- 3 -
计算与分析题.....	- 4 -
指令系统的设计 .....	- 6 -
填空题 .....	- 6 -
问答题 .....	- 6 -
计算与分析题.....	- 6 -
输入输出系统.....	- 6 -
填空题 .....	- 6 -
问答题 .....	- 7 -
计算与分析题.....	- 7 -
存储系统.....	- 7 -
填空题 .....	- 7 -
问答题 .....	- 7 -
计算与分析题.....	- 8 -
流水线技术 .....	- 8 -
填空题 .....	- 8 -
问答题 .....	- 8 -
计算与分析题.....	- 10 -
指令级并行及其开发 .....	- 12 -
填空题 .....	- 12 -
问答题 .....	- 12 -
计算与分析题.....	- 13 -
多处理机.....	- 13 -
填空题 .....	- 13 -
问答题 .....	- 13 -
计算与分析题.....	- 14 -

# 系统结构基础知识

## 填空题

1. 如果仅仅对计算机中的一部分做性能改进, 则改进越多, 系统获得的效果越小, 即 Amdahl 定律性能递减规则
2. 评价一个计算机系统的性价比, 主要是看它否是一个 带宽平衡 的系统
3. 通常, 在一个计算机系统中, 低层机器级的概念性结构和功能特性, 对高级语言程序员来说是 透明的

## 问答题

1.请简述软件兼容的内容及意义。

答:

(1) 同一个软件可以不加修改地运行于体系结构相同的各档机器, 而且它们所获得的结果一样, 差别只在于有不同的运行时间。软件兼容包含向上(下)兼容和向前(后)兼容。

A.向上(下)兼容: 按某档机器编制的程序, 不加修改的就能运行于比它高(低)档的机器。

B.向前(后)兼容: 按某个时期投入市场的某种型号机器编制的程序, 不加修改地就能运行于在它之前(后)投入市场的机器。(其中向后兼容必须做到, 也是系列计算机的根本特征)

(2) 软件相对于硬件的成本越来越贵, 而当前已积累了大量成熟的系统软件和应用软件, 因此有着兼容性的软件对于节约成本等有着重要的意义。

2.请简述计算机性能设计与评价的基本原则。

答:

(1) 大概率事件优先的原则: 对于大概率事件(最常见的事件), 赋予它优先的处理权和资源使用权, 以获得全局的最优结果;

(2) Amdahl 定律: 系统中某一部件由于采用某种更快的执行方式后整个系统性能的提高与这种执行方式的使用频率或占总执行时间的比例有关。;

(3) 程序的局部性原理: 程序在执行时所访问地址的分布不是随机的, 而是相

(4) 对地簇聚(这种簇聚包括指令和数据两部分);

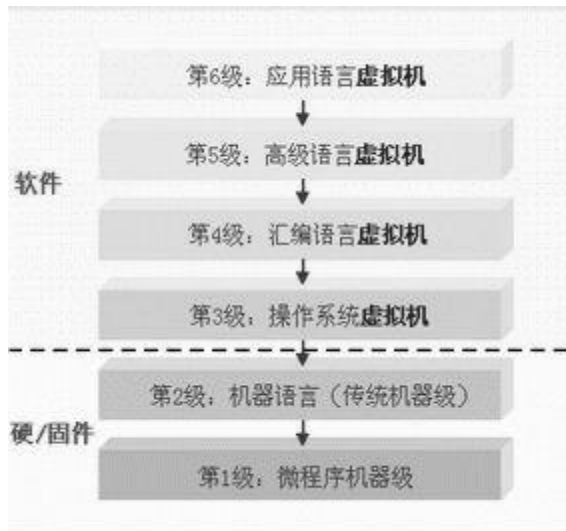
(5) 程序的时间局部性: 最近被访问的数据或指令很快又将被访问;

(6) 程序的空间局部性: 空间上存储在一起的数据或指令容易一起被访问。

3.什么是计算机的多级层次结构? 分层的作用是什么?

答:

(1) 从计算机语言的角度, 把计算机系统按功能划分成以下多级层次结构:



(2) 可以调整软、硬件比例; 可以用真正的实处理机代替虚拟机器; 可以在 1 台宿主机上仿真另一台。

## 计算与分析题

1. 已知计算机系统的执行时间为 100ms, 在此系统中的程序 M 的执行时间为 10ms, 且该程序是可改进的。

(1) 若对系统里的程序 M 进行改进以后, 可使程序的执行时间减少 2ms, 求其可改进比例和部件加速比;

(2) 若系统里除了 M 以外的程序改进后需要的执行时间为 70ms, 求系统的加速比。

解: (1) 可改进比例 =  $\frac{2ms}{10ms} = 0.2$

部件加速比 =  $\frac{10ms}{10ms - 2ms} = 1.25$

(3) 系统的加速比 =  $\frac{100ms}{10ms + 70ms} = 1.25$

2. 已知计算机系统有 n 种指令 (n>5), 现已知其中的 5 种指令的 CPI (指令时钟数)、IC (程序执行过程中所处理的指令数) 和时钟频率 100HZ, 求这 5 种指令的用户 CPU 时间。

指令类型	1	2	3	4	5
CPI	2	3	5	5	4
IC	15	10	12	8	5

解: 对于一种指令, CPU 时间 =  $\frac{CPI \times IC}{\text{时钟频率}}$ , 则求得 5 种指令的总 CPU 时间为:

$$\frac{2 \times 15 + 3 \times 10 + 5 \times 12 + 5 \times 8 + 4 \times 5}{100} = 1.8$$

3. 已知条件分支指令有如下两种不同设计方法:

①CPU<sub>A</sub>: 通过比较指令设置条件码, 然后测试条件码进行分支;

②CPU<sub>B</sub>: 在分支指令中包括比较过程;

条件分支指令都占用 2 个时钟周期而所有其它指令占用 1 个时钟周期。对于 CPU<sub>A</sub>, 执行的指令中分支指令占 20%; 由于每个分支指令之前都需要有比较指令, 因此比较指令也占 20%, 且在分支时不需要比较。

(1) 假设 CPU<sub>B</sub> 的时钟周期时间是 CPU<sub>A</sub> 的 1.25 倍。哪一个 CPU 更快?

(2) 如果 CPU<sub>B</sub> 的时钟周期时间仅仅是 CPU<sub>A</sub> 的 1.1 倍, 哪一个 CPU 更快?

解:

(1) 对于 CPU<sub>A</sub>:

占用 2 个时钟周期的分支指令占总指令的 20%, 剩下的指令占用 1 个时钟周期且占总指令数的 80%

则平均时钟周期数为  $CPI_A = 0.2 \times 2 + 0.80 \times 1 = 1.2$

CPU 时间<sub>A</sub> =  $IC_A \times 1.2 \times \text{时钟周期}_A$  (亦可以为  $IC_A \times 1.2 / \text{时钟频率}_A$ )

对于 CPU<sub>B</sub>:

时钟周期<sub>B</sub> =  $1.25 \times \text{时钟周期}_A$

由于在 CPU<sub>B</sub> 中没有独立的比较指令, 所以 CPU<sub>B</sub> 的程序量为 CPU<sub>A</sub> 的 80%

则分支指令占总指令的比例为:  $20\%/80\% = 25\%$

即 25% 的分支指令占用 2 个时钟周期, 而剩下的 75% 的指令占用 1 个时钟周期

$CPI_B = 0.25 \times 2 + 0.75 \times 1 = 1.25$

因为 CPU<sub>B</sub> 不执行比较, 故其指令数为 CPU<sub>A</sub> 的 80%

$IC_B = 0.8 \times IC_A$

CPU 时间<sub>B</sub> =  $IC_B \times CPI_B \times \text{时钟周期}_B$   
=  $0.8 \times IC_A \times 1.25 \times (1.25 \times \text{时钟周期}_A)$

=  $1.25 \times IC_A \times \text{时钟周期}_A$

可知 CPU<sub>A</sub> 比 CPU<sub>B</sub> 快。

(2) 如果 CPU<sub>A</sub> 的时钟周期时间仅仅比 CPU<sub>B</sub> 快 1.1 倍, 则

时钟周期<sub>B</sub> =  $1.10 \times \text{时钟周期}_A$

CPU<sub>B</sub> 的性能为:

CPU 时间<sub>B</sub> =  $IC_B \times CPI_B \times \text{时钟周期}_B$   
=  $0.8 \times IC_A \times 1.25 \times (1.10 \times \text{时钟周期}_A)$

=  $1.10 \times IC_A \times \text{时钟周期}_A$

由 (1) 知 CPU 时间<sub>A</sub> =  $IC_A \times 1.2 \times \text{时钟周期}_A$

故可知 CPU<sub>B</sub> 比 CPU<sub>A</sub> 快

# 指令系统的设计

## 填空题

无

## 问答题

1. 简述指令操作码常见的三种编码方法的优缺点或特点。

答:

- (1) 定长编码。优点: 规整, 译码简单; 缺点: 浪费信息量(操作码的总长位数增加);
- (2) 哈弗曼编码。优点: 操作码平均位数较少; 缺点: 编码长度是变化的, 不规整, 不利于硬件处理;
- (3) 等长扩展编码。扩展操作码必须遵守短码不是长码的前缀, 扩展操作码的编码和平均码长都不唯一。因此, 需要对各扩展方案进行比较, 以便找出一种平均码长尽可能短、码长种类个数不能过多、便于优化实现的方案。

## 计算与分析题

1. 某指令系统共有 72 种指令, 前 4 种使用频率平均为 0.12, 中间 15 种使用频率平均为 0.02, 最后 55 种使用频率平均为 0.004。如何编码?

解:

采用 4-16-64 扩展方法, 编码第一位用于扩展, 每段 0xx 用于编码, 1xx 用于扩展。

4-16-64 平均码长

$$=0.12*4*3+0.02*15*6+0.004*55*9$$

$$=5.22;$$

# 输入输出系统

## 填空题

1. 从磁盘读取文件可分为定位、找扇区、读出数据三步

2. 在数组多路通道中, 当在为一台高速设备传输数据的同时, 有多台高速设备可以处于定位和找扇区的状态, 与选择通道相比, 数组多路通道有数据传输率高和通道的硬件利用率高和控制硬件的复杂度高的特点

## 问答题

1. 请简述通道的种类与特点。

答: (1)字节多路通道:为多台低中速的外围设备服务,有多个子通道,每个子通道连接一个控制器

(2)选择通道:为高速外围设备服务,只有一个以成组方式工作的子通道

(3)数组多路通道:字节多路通道和选择通道的结合,每次为一台高速设备传送一个数据块,并轮流为多台外围设备服务。

## 计算与分析题

无

## 存储系统

### 填空题

1. 在高速机器中, 信号传输是延迟的一个重要原因, 而大的存储器因地址译码级数多, 信号延迟会更大。
2. 相同器件条件下, 小存储器速度比大存储器快。高频事件高速处理。
3. 最近可能被访问的数据一般放在小的但高速存储器里, 一般这样的存储器离 CPU 的距离较近。
4. 构建虚拟存储系统的主要目的是扩大存储器容量。

Cache 命中率与硬件速度无关, 与应用程序的特性有关, 访问效率主要与命中率和两级存储器的速度之比有关

## 问答题

1. 请简述存储系统层次结构的工作原理及特点

答:

在任何指定时间, 数据只能在相邻的两级之间拷贝:

较高级存储设备: 与处理器较近的存储级, 较小、较快、使用较昂贵的技术工艺

较低级: 与处理器较远的存储级, 较大、较慢、使用较廉价的技术工艺



## 计算与分析题

1. 已知 CPU 访问 Cache 需要时间  $T_1$ , Cache 访问主存需要时间  $T_2$ , 且从主存将数据传送到 Cache 需要时间  $T_{21}$ , 现 CPU 访问 Cache 的总次数为  $N$ , 有  $N_1$  次访问失败。

- (1) 求 Cache 的命中率;
- (2) 求平均访存时间。

解:

- (1) Cache 的命中率为  $H = \frac{N-N_1}{N}$ ;
- (2) 命中是访问的时间为  $T_1$   
不命中时访问的时间为  $T_1+T_2+T_{21}$   
故该系统的平均访存时间为  $HT_1+(1-H)(T_1+T_2+T_{21})$

## 流水线技术

### 填空题

1. 从本质上讲, 流水线技术是一种时间并行技术。
2. 流水线的工作流程包括装入、流水和排空三个阶段。
3. 流水线的每个功能部件后面都需要锁存器的主要原因是解决同步问题。
4. 流水线的实际吞吐率总是小于最大吞吐率, 它与每个段的时间、流水线的段数和输入到流水线中的任务有关。
5. 按照所完成的功能来分, 流水线可分为单功能流水线和多功能流水线。
6. 按照同一时间内各段之间的连接方式来分, 流水线可分为静态流水线和动态流水线。
7. 按照流水的级别来分, 流水线可分为部件级流水线、处理机级流水线和处理机间流水线。
8. 按照数据表示来分, 流水线可分为向量流水线和标量流水线。
9. 按照流水线中是否有反馈回路来分, 流水线可分为线性流水线和非线性流水线, 其中非线性流水线有反馈线。
10. 按照任务的执行顺序来分, 流水线可以分为顺序流水线和乱序流水线。
11. 衡量流水线性能的主要指标有吞吐率、加速比和效率。

### 问答题

1. 请简述流水线技术的原理与特点

答:

- (1) 原理: 把一个重复的过程分解为若干个子过程, 每个子过程由专门的功能部件来实现。把多个处理过程在时间上错开, 依次通过

各个功能段, 这样, 每个子过程就可以与其他的子过程并行执行, 这就是流水线技术;

- (2) 特点: a.流水过程由多个相联系的子过程组成, 每个过程称为流水线的“级”或“段”;  
b.每个子过程由专用的功能段实现;  
c.各个功能段所需时间应尽量相等;  
d.流水线需要有“通过时间”, 在此之后流水过程才进入稳定工作状态, 每一个时钟周期(拍)流出一个结果;  
e.流水技术适合于大量重复的时序过程, 只有在输入端能连续地提供任务, 流水线的效率才能充分发挥。

2. 什么是流水线的速度瓶颈? 消除流水线速度瓶颈的方法有哪两种, 举例示意说明。

答:

流水线中经过时间最长的功能段就是流水线中的速度瓶颈。清除流水线的速度瓶颈的方法, 一是将瓶颈过程再细分。例如, 流水线瓶颈段为  $2\Delta t$ , 其它各段均为  $\Delta t$ , 则可将该瓶颈段再细分成两个子段, 各子段经过时间均为  $\Delta t$ ; 二是将瓶颈子过程多套并联, 如上例, 将瓶颈重复设置两个, 任务错开一个  $\Delta t$  时间分配给这两个段。

3. 请简述相关性的概念和类型, 并叙述相应的解决策略。

答:

- (1) 相关性指的两条指令之间存在的某种依赖关系。

- (2) 相关性有三种类型: 数据相关、名相关和控制相关。

A. 数据相关: 考虑两条指令  $i$  和  $j$ , 设  $i$  在  $j$  的前面, 如果以下条件之一成立,

则称  $j$  与  $i$  数据相关:

a. 指令  $j$  使用指令  $i$  产生的结果。

b. 指令  $j$  与指令  $k$  数据相关, 指令  $k$  又与指令  $i$  数据相关。

解决策略: 保证指令的执行顺序, 即数据须从生产者流动到消费者。

B. 名相关: 如果两条指令使用了相同的名字, 但是他们之间并没有数据流动, 则这两条指令存在名相关。指令  $j$  与指令  $i$  之间的名相关有如下两种:

a. 反相关: 如果指令  $j$  所写的名字与指令  $i$  所读的名字相同, 则称指令  $i$  和  $j$  发生了反相关。

b. 输出相关: 如果指令  $j$  和指令  $i$  所写的名字相同, 则称指令  $i$  和指令  $j$  发生了输出相关。

解决策略: 可以通过改变指令中操作数的名字来消除名相关, 即换名技术。

C. 控制相关: 指由分支指令引起的相关, 它需要根据分支指令的执行结果来确定后面该执行哪个分支上的指令。

解决策略: 为了保证程序应有的执行顺序, 必须严格按照控制相关

确定的顺序执行。

#### 4. 请简述流水线冲突的概念。

答:

流水线冲突指对于具体的流水线来说, 由于相关的存在, 使得指令流中的下一条指令不能在指定的时钟周期开始执行。流水线冲突有以下三种类型:

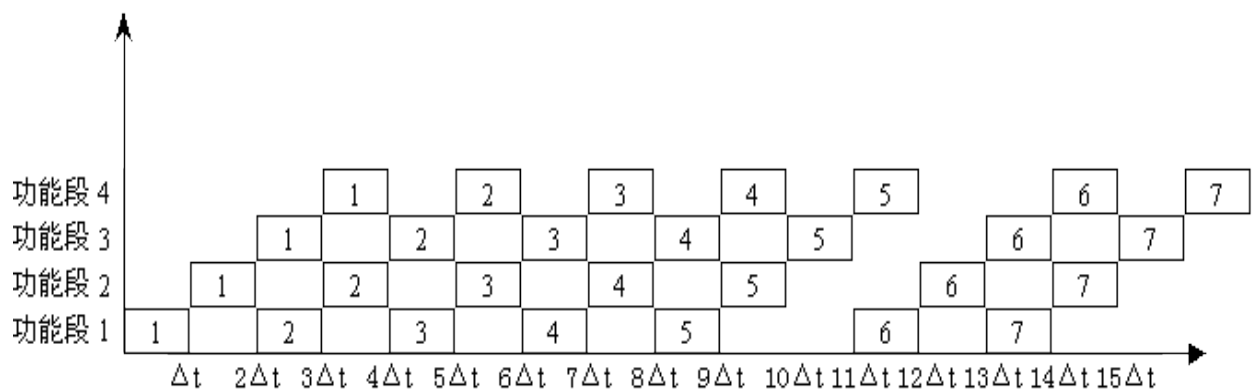
- (1) 结构冲突: 因硬件资源满足不了指令重叠执行的要求而发生的冲突。
- (2) 数据冲突: 当指令在流水线中重叠执行时, 因需要前面指令的执行结果而发生的冲突。
- (3) 控制冲突: 流水线遇到分支指令或其它会改变 PC 值的指令所引起的冲突。

## 计算与分析题

1. 一条线性流水线有 4 个功能段组成, 每个功能段的延迟时间都相等, 都为  $\Delta t$ 。开始 5 个  $\Delta t$ , 每间隔一个  $\Delta t$  向流水线输入一个任务, 然后停顿 2 个  $\Delta t$ , 如此重复。求流水线的实际吞吐率、加速比和效率。

解:

流水线的时空图如下:



我们可以看出, 在  $(11n+1)\Delta t$  的时间内, 可以输出  $5n$  个结果, 如果指令的序列足够长 ( $n \rightarrow \infty$ ), 并且指令间不存在相关, 那么, 吞吐率可以认为满足:

$$T_P = \frac{5n}{(11n+1)\Delta t} = \frac{5}{(11+1/n)\Delta t} = \frac{5}{11\Delta t} (n \rightarrow \infty)$$

$$S = \frac{5n \times 4\Delta t}{(11n+1)\Delta t} = \frac{20n}{11n+1} = \frac{20}{11+1/n} = \frac{20}{11} (n \rightarrow \infty)$$

加速比为:

从上面的时空图很容易看出, 效率为:

$$E = \frac{T_0}{k \times T_k} = \frac{20n\Delta t}{4 \times (11n+1)\Delta t} = \frac{5}{11+1/n} = \frac{5}{11} (n \rightarrow \infty)$$

2. 一条线性静态多功能流水线由 6 个功能段组成, 加法操作使用其中的 1、2、3、6 功能段, 乘法操作使用其中的 1、4、5、6 功能段, 每个功能段的延迟时间均相等。流水线的输入端与输出端之间有直接数据通路, 而且设置有足够的缓冲寄存器。现在用这条流水线计算:

$$F = \sum_{i=1}^6 (A_i \times B_i)$$

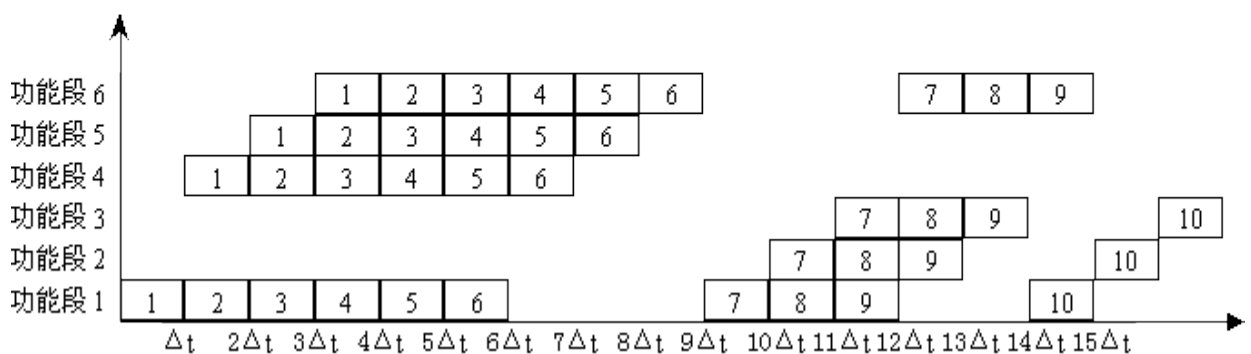
画出流水线时空图, 并计算流水线的实际吞吐率、加速比和效率。

解:

为了取得较高的速度, 我们需要一次将乘法作完, 设源操作数存放在寄存器 A、B 中, 中间结果存放在寄存器 R 中, 最后结果存放在寄存器 F 中, 则执行的指令序列如下所示:

I1: R1 ← A1\*B1  
I2: R2 ← A2\*B2  
I3: R3 ← A3\*B3  
I4: R4 ← A4\*B4  
I5: R5 ← A5\*B5  
I6: R6 ← A6\*B6  
I7: R7 ← R1+R2  
I8: R8 ← R3+R4  
I9: R9 ← R5+R6  
I10: R10 ← R7+R8  
I11: F ← R9+R10

这并不是唯一可能的计算方法。假设功能段的延迟为  $\Delta t$ 。时空图 (不完全) 如下, 图中的数字是指令号:



整个计算过程需要  $22 \Delta t$ , 所以吞吐率为:

$$T_P = \frac{11}{22\Delta t} = \frac{1}{2\Delta t}$$

加速比为:

$$S = \frac{11 \times 4 \Delta t}{22 \Delta t} = 2$$

效率为:

$$E = \frac{T_0}{k \times T_k} = \frac{11 \times 4 \Delta t}{6 \times 22 \Delta t} = \frac{1}{3}$$

## 指令级并行及其开发

### 填空题

1. 在一个时钟周期内能够分式流出多条指令的处理机称为超流水线处理机, 其是基于时间并行性原理。
2. 对于一台每个时钟周期能流出  $n$  条指令的超流水线计算机来说, 该计算机的流水线周期为 $\frac{1}{n}$ 个时钟周期。
3. 超流水线处理机主要是通过提高流水线运行速度来增强机器性能。

### 问答题

1. 请简述并行以及指令级并行度的内容。

答:

- (1) 并行包含粗粒度并行与细粒度并行。

粗粒度并行: 是在多处理机上分别运行多个进程, 由多台处理机合作完成一个程序。

细粒度并行: 是在一个进程中进行指令一级或操作一级的并行处理。

在多处理机系统中可以同时采用这两种粒度的并行性。在单处理机上则用细粒度并行性。

2. 请简述数据相关的内容及其相应的处理技术。

答:

- (1) 数据相关分三类: 读后写(WAR)相关、写后读(RAW)相关、写后写(WAW)相关。

- (2)

寄存器重命名技术

定向传输技术: 将前面指令执行的结果直接送给后面指令所需此源操作数的功能部件

3. 请简述超标量流水线处理机的特点。

答:

- ① 配置有多个性能不同的处理部件, 采用多条流水线并行处理。
- ② 能同时对若干条指令进行译码, 将可并行的指令送往不同的执行部件, 从而达到每个周期启动多条指令。
- ③ 在程序运行期间由硬件 (通常是状态记录部件和调度部件) 完成指令调度。

## 计算与分析题

1. 假定有多个加法器, 不存在加法器的资源冲突。3 条连续指令组成的程序代码段如下:

```
I1    ADD  R1, R2, R4           ; R1 ← (R2) + (R4)
      I2    ADD  R2, R1, 1       ; R2 ← (R1) + 1
      I3    ADD  R1, R4, R5      ; R1 ← (R4) + (R5)
```

- (1) 请找出代码中的相关, 并指出相关类型;
- (2) 试分析如何消除相关性。

解:

- (1) I1 与 I2 存在写后读 (RAW) 相关、读后写 (WAR) 相关  
I2 与 I3 存在读后写 (WAR) 相关  
I1 与 I3 存在写后写 (WAW) 相关

(2) 按照寄存器重命名的规则, 把 I2 中的 R1 换成 R1', I2 中的 R2 换成 R2', I3 中的 R1 换成 R1'', 经寄存器重新命名之后, 这 3 条指令在实际执行时就变为:

```
I1    ADD  R1,  R2, R4
      I2    ADD  R2', R1', 1
      I3    ADD  R1'', R4, R5
```

这样就消除了代码的相关性。

## 多处理机

### 填空题

1. 根据存放形式分类, Cache 目录分为集中式和分布式两种。
2. 根据结构分类, 目录协议分成全映射目录、有限目录和链式目录。

### 问答题

1. 在并行处理机和多处理机系统中, 采用局部 Cache 会引起 Cache 与共享存储器之间的一致性问题。请分析造成不一致的原因。

答:

出现不一致性问题的原因有三个:

- (1) 共享可写的数据
- (2) 进程迁移引起的数据不一致性
- (3) I/O 造成数据不一致性

2.请列举一种解决 I/O 操作引起数据不一致的方法。

答:

一种解决 I/O 操作引起数据不一致性的方法是把 I/O 处理机分别连接到各自的局部 Cache 上。

3.请简述两类解决 Cache 不一致性问题的协议。

答:

(1) 在总线互连的多处理机系统中通常采用监听协议,在其他多处理机系统中通常采用基于目录协议。

(2)

监听协议

写无效策略: 在本地 Cache 的数据块修改时使远程数据块都无效。

方法二: 写更新策略: 在本地 Cache 数据块修改时通过总线把新的数据块广播给含该块的所有其他 Cache

基于目录协议

Cache 目录结构: 存放的内容是大量的指针,用以指明块拷贝的地址,每个目录项还有一个重写位,指明是否有一个 Cache 允许写入数据。

## 计算与分析题

无