k高级体系结构历年考博题简答题

# 1、大规模机器同步的软件、硬件支持方法：

硬件：硬件排队锁、硬件原语

软件：延迟等待、旋转锁、软件排队锁、组合树

# 2、栅栏同步完成同步的过程：

栅栏强制所有到达栅栏的进程进行等待，直到所有的进程到达栅栏后释放所有进程，从而实现同步。栅栏的实现通常使用两个旋转锁，一个用于记录到达的进程数目，另一个用于使到达的进程等待。进程在等待过程中要不停地检查指定变量，直到所有进程到达栅栏。

# 3、采用松弛一致性模型提高性能的原理以及主要的硬件支持措施

松弛一致性模型的特点在于，允许读写操作打乱次序完成，但要同步操作来保证排序原则，使得一个同步程序的表现和处理器使用顺序一致性表现一样。松弛一致性在保证程序正确性的前提下增加指令执行的并行，所以采用松弛一致性模型的机器可以提高性能。根据消除的读取顺序的内容，可以将松弛一致性模型划分为4类：

（1）、完全存储排序模型：消除了W-R顺序，硬件上的支持为写缓冲的读旁路等。维护写的次序，这种模型方案用写缓冲，并提供读的旁路机制，从而允许处理机在其写操作被所有别的处理机看到之前就继续进行读

（2）、部分存储排序模型：进一步消除了W-W，在硬件支持上为写的流水线或其他并行等。允许非冲突写隐含地乱序执行，实现上可以使写流水或重叠，而不是强制一个操作必须在另一个之前结束。对同步操作仍然需要将写操作挂起，因为它引起写防护

（3）、弱排序模型，进一步消除了R-R，R-W，在硬件支持上为不封锁读

（4）、释放一致性模型，进一步消除了W-SA，R-SA，SR-R，SR-W，在硬件支持上为不封锁读，旁路，无序写等。这种模型区分同步操作中的访问一个共享变量的获取操作S-A，将对象释放允许别的处理机获取访问权的释放操作SR

# 4、多处理机的相关性(Coherency)和一致性(Consistency)

如果对某个数据项的任何读操作均可得到其最新写入的值，则认为这个存储系统是一致的(Conherency)，定义包括两个方面：返回给读操作的是什么值，即coherency(一致性)；什么时候才能将已写入的值返回给读操作，即consistency(连贯性)。一致性和连贯性是互补的，一致性定义了对同一个存储器地址进行的读写操作行为，而连贯性定义了关于访问其他存储地址的读写操作

# 5、同时多线程、粗粒度多线程和细粒度多线程比较：

（1）、细粒度多线程，在每条指令之间都能进行线程的切换，从而导致多个线程的交替执行，通常以时间片循环的方式实施线程交替，在循环过程中跳过阻塞的线程。

优点：可以较好地掩盖任何或长或短的阻塞带来的吞吐率的损失

缺点：单个线程的执行时间都减慢了。

（2）、粗粒度多线程，切换只发生在代价较高，时间较长的阻塞出现时

优点：大大减少了线程的切换次数，不会减慢每个独立线程的执行

缺点：不能有效减少吞吐率的损失，特别是对于较短的阻塞而言

（3）、同时多线程：是细粒度多线程的变体，在多发射、动态调度处理器的顶层实现细粒度多线程。利用线程级并行来隐藏处理器的长延迟事件。关键在于认识到通过寄存器重命名和动态调度可以执行来自独立线程的多个指令，将指令的相关性留给动态调度功能来处理。

优点：极大提高吞吐率损失较少的幅度，且不损失单个线程的性能。

# 6、基于硬件前瞻执行的优缺点

实现前瞻的关键思想是允许指令乱序执行，但必须顺序提交。可以得到更高的性能，还能在处理器未判断指令是否能执行之前就提前执行，以克服控制相关。基于硬件的前瞻执行和动态调度相结合，应对基本模块不同组合方式的调度。

缺点是支持前瞻的硬件太复杂，需要大量的硬件资源。

# 7、比较总线、多级网络、交叉开关的特点

总线结构：通过共享总线把多个处理器，共享存储器连接起来，在多个请求下总线仲裁分配总线的使用权。总线互联最简单，造价最低，但是竞争最严重，每台处理器可用的带宽较窄，故障率较高；

交叉开关把N台处理机和M个存储器连接起来，网络中每个交叉点是任何一台处理机和任何一个存储器连接的开关，属于无阻塞置换网络。交叉开关的硬件复杂性以N2倍上升，造价最昂贵，但是带宽和性能最好，是小规模网络的理想选择。

多级网络：每一级使用了多个开关，相邻开关之间都有固定的多级连接；多级网络是总线和交叉开关的这种，有点在于用模块化结构，因此可扩展性较好，但是时延随级数增加而上升。此外，由于增加了连线和开关复杂性，其价格也是一种限制因素。

# 8、多处理机监听(目录)协议的工作原理

目录：在每个节点增加了目录存储器用于存放目录。存储器的每一块在目录中对应有一项，每个目录项主要有“状态”和“位向量”两个内容。“状态”描述目录对应存储块的当前情况，“位向量”共有N位，每一位对应一个存储器的局部Cache，用于指出该Cache中有无该块的副本。当处理器对某一块进行写操作时，根据位向量通知具有相应副本的处理器进行失效等操作；

监听：监听协议关键在于对共享数据写操作的处理。当某个处理机要写数据时，必须先获得总线控制权，然后将要写的数据地址放在总线上。其他处理机已知监听总线，检测该地址是否在于它们的Cache中。若存在相应的数据块，则将其失效。获取总线控制权的顺序保证了写的顺序，因为总线总是独占使用的。

状态转换图：



画全各个状态中CPU的不同动作会造成不同的影响，以及会在总线上放置不同消息，而来自总线的不同消息会使状态发生何种转换。

# 9、同时多线程在体系结构实现上的基础

同时多线程(SMT)是一种多流出、动态调度处理器上开发线程级并行和指令级并行的改进的多线程技术。在SMT超标量处理器中，通过一个时钟周期内调度多个线程使用指令槽从而实现线程级并行和指令级并行。

同时多线程开发的基础是使用动态调度技术的处理器，已经具有了开发线程级并行所需要的硬件资源。具体来说，动态调度超标量处理器有大量的虚拟寄存器可以用来保存每个独立线程的寄存器状态。由于寄存器重命名机制提供了唯一的寄存器依赖关系，多个线程的指令可以在数据路径上混合执行，而不会导致线程间源操作数和目的操作数的混乱。这表明多线程技术可以通过在一个乱序执行的处理器上设置重命名表，保留各自的PC值，提供多个线程的指令结果提交的能力来实现。

# 10、多处理机系统中进行时延隐藏的主要技术途径与原理

预取，相关性Cache，松弛一致性，多现场这几种时延隐藏技术。

* 预取：在数据使用之前就利用访存的时机将其取到近处；
* 相关性Cache：减少对共享数据访问的竞争以及时延；
* 松弛一致性：在保证程序正确性的前提下增加指令执行的并行；
* 多现场：在产生时延时进行现场切换而执行其他程序；

## 年代久远的题目

# 11、减少Cache失效开销的策略

读失效优于写：发生读失效时将脏块送至缓冲器，处理读失效再将脏块写入存储器。

子块放置技术：把一个Cache块划分为若干小块各做标记，子块的失效开销小于整体开销。

请求字处理技术：当CPU需要的数据到达后不等整个块到达Cache就把数据发送给CPU。

非阻塞Cache技术：发生miss时不停顿，继续取指执行。

采用两级Cache。

# 12、三种降低Cache命中时间的技术手段

使用容量小，结构简单的Cache

虚拟Cache：直接使用虚地址进行Cache索引

写操作流水化：对比标识与写入数据流水化，第一次的写操作和第二次的对比标识就可以并行。

# 13、CISC与RISC的设计原则

CISC：强化指令功能，实现软件功能向硬件功能转移；

* 面向目标程序增强指令功能
* 面向高级语言和变异程序改进指令系统
* 面向操作系统的优化实现改进指令系统

RISC：尽可能降低指令结构的复杂性，已达到简化实现，提高性能的目的。

* 选取使用频率醉倒的指令，并补充一些有用的指令；
* 每条指令的功能尽可能简单，并且在一个机器周期内完成；
* 所有指令长度均相同
* 只有Load和Store指令才能访存
* 以简单有效的方式支持高级语言

# 14、两级Cache的原理

两级Cache是一种降低失效开销的技术。通过在原有Cache和存储器之间增加另一级Cache，构成两级cache，其中第一级cache容量小，但速度与CPU的时钟周期相匹配；第二级Cache足够大，能够捕获更多本来要到主存去的访问，从而能够降低实际失效开销。

访存时先访问一级Cache，若一级Cache命中，则直接获取数据；若一级Cache没命中，则访问二级Cache。二级Cache没有命中才访问存储器。

# 15、多指令流出受到的限制

* 程序本身的并行性
* 指令发射机制
* 执行机制及确认写回机制

# 16、用物理地址进行DMA时存在的问题

(1)、对于超过一页的数据缓冲区，由于缓冲区使用的页面在物理存储器中不一定是连续的，所以传输可能会发生问题；

(2)、如果DMA正在存储器和缓冲器之间传输数据时，操作系统从存储器中移出(或重新装载一些页面后)，DMA将会在存储器中错误的页面上传输数据。

# 17、目前实现盘阵列的三种主要方式

(1)、软件方式：即阵列管理软件由主机来实现。优点是成本低；缺点是过多地占用主机时间并且带宽指标上不去

(2)、阵列卡方式：即把RAID管理软件固化在I/O控制卡上，从而可以不占用主机时间，一般用于工作站和PC机

(3)、子系统方式：是一种基于通用接口总线的开放式平台，可用于各种主机平台和网络系统

# 18、TLB的原理

TLB又称为快表，是一个专用的高速缓冲器，用于存放近期经常使用的页表项，其内容是页表部分内容的一个副本。这样，进行地址变换时，一般直接查TLB就可以了，只有偶尔在TLB不命中时，才需要去访问内存的页表。TLB也利用了局部性原理，如果访存具有局部性，则这些防村中的地址变换也具有局部性，即所有的页表项是相对聚簇的。TLB也常称为“快表”或地址变换缓冲器。

TLB中的项与Cache中的类似，也是由两部分构成：标识和数据。标志中存放的是虚地址的一部分，而数据部分中则存放物理页帧号，有效位，存储保护信息以及其他一些辅助信息。为了使TLB中的内容与页表保持一直，当修改页表中的某一项时，操作系统就必须奥正TLB有该页表的副本。

# 20、VLIW上性能优于向量处理器的实例*(可以讨论一下，暂时只是我的理解)*

考虑向量处理器RAR的指令相关，即所谓源寄存器冲突，即向量寄存器不能够同时支持两个向量寄存器读，而VLIW可以。

ADD.D F1 F2 F3 ADD.D F4 F2 F3 ADD.D F5 F2 F3 ADD.D F6 F2 F3

# 21、超线性加速比的原因

(1)、规模增大提高数据的局部性，降低了Cache的失效率

(2)、通信规模的增大幅度不足以抵消Cache利用率带来的增益

(以下可能是计算机原理/基础并且我不熟悉的内容)

# 22、总线的出现至今对计算机产生了哪些影响

模块化

# 21、激光打印机调至激光束的基本原理