历年高体考博试卷总结v1.0

旨在总结历年试题，标记出有争议的题目，便于复习讨论。

# 一、名词解释与填空

名词解释靠理解记忆，依照目前规律，出现名词解释题目的概率不大。下面的名词解释中，划线标红部分是考博试卷作为填空题出题的填空内容。

**解题思路：**背吧。。。

## 基础概念

* **程序的局部性原理**：程序总是趋向于使用最近使用过的指令和数据，即程序执行时所访问的存储器地址不是随机的，而是相对地聚簇。程序局部性包括程序的时间局部性和程序的空间局部性。(2008, 2012, 2015)
* **存储程序计算机**又叫冯·诺依曼计算机，由运算器、控制器、输入/输出设备、存储器四个部分组成。 (2015)
* 把指令集设计成只包含那些使用频率高的少量指令，并提供一些必要的指令以支持操作系统和高级语言，按照这个原则而构成的计算机称为精简指令集计算机计算机 (2015)
* **Amdahl定律**：改进一个部件所获得的的系统性能的提高，受限于该部件执行时间相对于总执行时间所占的比例 (2004, 2010, 2011, 2014)
* **系列机**是指具有相同体系结构但具有不同组织和实现的计算机。不同厂家生产的具有相同体系结构的计算机称为兼容机 (2008, 2010, 2013)
* **计算机体系结构：**程序员看到的计算机的属性，即概念性结构与功能特征 (2001, 2005, 2008, 2013)
* **Power consumption** in modern systems is dependent on a variety of factors, including the chip clock frequency, efficiency, disk drive speed, disk drive utilization, and DRAM (2013)
* **存储程序计算机的特点：**机器以运算器为中心；采用存储程序原理；存储器是按地址访问的线性编址的空间；控制流由指令流产生；指令由操作码和地址码组成；数据以二进制编码表示，采用二进制运算。 (2002, 2007, 2009, 2012)
* 机器发生故障的概率与时间之间的关系可以使用“**学习曲线**”来说明 (2011)
* **设计指令系统**包括数据表示、操作类型、寻址方式和指令编码等方面 (2011)
* RISC：精简指令集系统计算机；CISC：复杂指令集系统计算机 (2008)
* **流量**(throughput)：单位时间内完成的事务数 (2002, 2006)

(以下只在名词解释中出现过)

* **指令系统的规整性：**规整形包括对称性和均匀性。对称性是指所有指令集有关的存储单元的使用、操作码的设置等都是对称的；均匀性是指对于各种不同的操作数类型，字长操作种类和数据存储单元，指令的设置都要同等对待 (2001, 2004)
* **异构型多处理机：**在主CPU与存储器的基础上有基于其他体系结构的协处理器与存储器协同完成计算任务的多处理机体系结构 (2004)
* **Load/Store型指令集结构：**只有load/store指令才具有存储器操作数，其余所有ALU指令都不包含存储器操作数的通用寄存器型指令集结构 (2003)
* **基准程序包**：把一些基准测试应用程序集中起来，以有效评测处理器处理各种应用的性能 (2003)
* **基本块（basic block）**：一段线性的除了入口及出口处没有分支的代码序列（cache blcok?） (2003)
* **指令系统的正交性：**在指令中各个不同含义的字段在编码是互不相关相互独立 (2002)
* **软件兼容：**同一个软甲你可以不加修改地运行于体系结构相同的各档机器，而且它们所获得的的结果一样，差别只在于运行的时间不同 (2002)
* **并行性：**在同一时刻或是同一时间间隔内完成两种或者两种以上性质相同或不同的工作 (2002)
* **时间局部性**：被访问过的地址在较短时间内有较高概率再一次被访问 (2002)
* **通用寄存器型机器：**CPU中用来存储操作数的存储单元为寄存器或存储器单元 (2001)
* **空间局部性**：下一个访问的地址在此时访问的地址附近

## 指令级并行

* 当指令之间不存在相关时，它们在流水线中是可以重叠起来并行执行的，这种指令序列中存在的潜在并行性称为指令级并行 (2015)
* **数据反相关**：对于顺序指令i和j，指令i和j存在数据反相关是指：指令j写入的是指令i要读取的寄存器或存储单元 (2007, 2011, 2014)
* **流水线的链接：**将流水线计算机中多个功能部件按照指令的要求连接在一起，构成一个长流水线，减少各个功能部件流水线建立和排空的时间，提高流水线执行效率的方法 (2007, 2009, 2013)
* **流水线CPI**=流水线理想CPI+结构相关导致的停顿+数据相关导致的停顿+控制相关导致的停顿 (2005, 2007, 2013)
* **数据真相关**：对于顺序指令i和j，指令i和j存在数据相关是指：指令j读取的是指令i要写入的寄存器或存储单元 (2010)
* **乱序流出**(Out of order issue)：程序中的指令不是按照排列的顺序流出，而是当指令需要的资源条件得到满足时即流出 (2001, 2010)
* **再定序缓冲：**在**推断执行**中，指令的流出过程是顺序的，但结束过程的确认是顺序的。再定序缓冲用于保存执行完毕但尚未顺序确认的指令 (2001, 2009)
* The ideal pipeline CPU is a measure of the maximum performance attainable by the implementation (2006)
* **CPI =** CPU clock cycles for a program : Instruction Count (2006)
* This potential overlap among instructions is called **Instruction-Level parallelism** sine the instructions can be evaluated in **parallel** (2006)

(以下只在名词解释中出现过)

* **数据相关**：一条指令需要用到前面那条指令的执行结果，而这些指令均在流水线中重叠执行时，可能引起数据相关 (2002, 2006)
* **多功能流水线**：各段可以进行不同的连接，以实现不同功能的流水线 (2004)
* **动态调度**：在保持数据流和异常行为的情况下，通过硬件对指令执行顺序进行重新安排，以提高流水线的利用率和减少停顿现象，是由硬件在程序运行时实施的 (2004)
* **分支预测缓冲：**使用一片存储区域记录最近一次或几次分支特征的历史，从而对分支进行预测 (2004)
* **结构相关：**当指令在重叠执行的过程中，硬件资源满足不了指令重叠执行的要求，发生资源冲突 (2001, 2003)
* **超标量技术**：每个时钟周期流出指令书不定的技术。既可以通过编译器静态调度，也可以通过记分牌或者Tomasulo算法动态调度 (2003)
* **定向（或称旁路，forwarding）**：将ALU的一个运算结果直接传送到所有需要它的功能单元，从而消除数据相关带来的停顿 (2003)
* **再定序缓冲**(Reorder Buffer)：一组记录了指令流出顺序、执行状态与计算结果的寄存器(2001)

## 线程级并行/数据级并行

* **同时多线程**是同时实现指令和线程级的并行每拍有多个指令槽，可以安排多个线程的多条指令同时流出 (2005, 2009, 2011, 2014, 2015)
* **向量流水线链接**：在数据相关的两条指令之间，将产生数据的指令部件的结果直接送到使用数据部件的输入 (2010, 2011, 2014)

## 编译器技术

* **软流水：**用于程序中循环结构的指令重组技术，又称之为符号化循环展开 (2002, 2010)
* **全局调度：**把分支之前的指令调度到分支之后，或者把分支之后的指令调度到分支之前，称之为全局代码调度。其目的是在保持代码段数据相关和控制相关不变的前提下，将一段包含控制相关的代码压缩到尽可能最短 (2005)

## 存储层次结构

* 我们把DRAM这种不供电数据会丢失的存储器称为易失性存储器，把磁盘这种不供电数据也能够保存的存储器称为非易失性存储器 (2015)
* 在**存储层次**中，“Cache­—主存”层次为了弥补主存 速度不足，“主存—辅存”层次为了弥补主存容量不足 (2004, 2008, 2011, 2012, 2015)
* **存储体冲突：**多体交叉存储器中，两次独立的存储器访问请求在一个存储周期中访问同一存储体，就造成存储体冲突 (2001, 2009, 2011, 2014)
* **TLB**是一个专用的高速缓存，用于存放近期经常使用的页表项，其内容是叶表部分内容的一个副本 (2007, 2012)
* **Cache的强制性失效：**Cache的强制性失效是第一次访问Cache块时，该块不在Cache中，必须从下一级存储器中调入 (2002, 2004, 2010)
* **虚拟Cache：**在采用虚拟存储器的计算机系统中，Cache的地址直接由虚拟地址转换获得，这就是虚拟Cache (2001, 2009)
* **映像规则：**主存任意一块放置到Cache中任意一个位置——全相联映射；唯一一个位置——直接映射；唯一一组中的任意位置——组相联映射 (2005)

(以下只在名词解释中出现过)

* **Cache容量失效：**如果程序执行时所需的块不能全部调入Cache中，则当某些块被替换后若又重新被访问，就会发生失效 (2003)
* **Cache冲突失效：**不同块映射到同一个cache line，当一个块被替换掉以后又被访问，会引起失效。
* **RAID2：**位交叉式海明编码阵列。数据以位（或字节）交叉方式存于格盘，采用海明编码 (2004)
* **RAID3：**位交叉奇偶校验盘阵列。单盘容错并行传输的阵列，数据以位/字节存放于各盘，荣誉的奇偶校验信息存储在一台专用盘上 (2003)
* **RAID5：**块交叉分布式奇偶校验盘阵列。数据以块交叉的方式存于格盘，但无专用的校验盘，而是把冗余的校验信息均匀地分布在所有磁盘上。 (2002)
* **RAID6：**双维奇偶校验独立存取盘阵列。即数据以块(大小可变)交叉的方式存于各盘，冗余的验错、纠错信息均匀地分布在所有的磁盘上 (2001)

## 并行体系结构

* **等分带宽**：将网络切成相等两半，燕切口的最小通道边数叫等分带宽 (2009, 2011, 2013, 2014, 2015)
* 非均匀存储器访问机器(NUMA)的访问时间依赖于数据在存储器中的存放位置 (2006, 2011, 2013, 2014, 2015)
* **虚拟自适应**：将一个屋里通道分成几个虚拟通道，根据后续个虚拟通道的忙闲情况自适应选择后续通道 (2003, 2007, 2012, 2015)
* **大规模机器的同步**硬件支持方法有排队锁和硬件原语，软件方法有延迟等待、排队锁和组合树 (2006, 2010, 2011, 2012, 2013, 2014)
* **互联网络的****网络直径**：互联网络中任意两个节点间最短路径长度的最大值称为网络直径 (2002, 2004, 2006, 2009, 2010, 2012, 2014)
* **栅栏同步**：强制所有到达该栅栏的进程进行等待，知道全部的进程到达栅栏，然后释放全部的进程，从而形成同步 (2003, 2007, 2011, 2013, 2015)
* **虫孔路由**：把消息分成小片，片头带目标地址，所有片以不可分离的流水的方式通过片缓冲区进行传输的路由方式 (2001, 2004, 2007, 2012)
* 互联网络的输入是。均匀混洗输出是，超立方体输出是 (2005, 2010)
* **非绑定预取**能返回最新写入数据值，并且保证对数据实际的存储器访问返回的是最新的数据项 (2008, 2009)
* **home节点**：存放数据的存储器单元及目录项所在的节点是home节点 (2008, 2009)
* **节点度**：与节点相连的边的数量叫做节点度 (2008)
* **拥有者**是指拥有唯一的cache块副本的处理器 (2008)

## 奇葩题目集合

* Freedom from compatibility with old designs and the use of microprocessor technology led to a renaissance in computer design, which emphasized both architectural innovation and efficient use of technology improvements (2006)
* Through this range in price and capability, the desktop market tends to be driven to optimize price-performance. For servers, the key features are availability, scalability and throughput (2006)
* The respond time is the time between the start and the completion of an event—also referred to as execution time. The manager of a large data processing center may be interested in increasing throughput–the total amount of work done in a given time (2006)

# 二、简答题

主要为并行体系结构的内容，指令调度，软流水，前瞻执行，前几年的题目还夹杂部分计算机原理相关的内容。

**解题思路：背吧孩子**

## 并行体系结构

并行体系结构的简答题主要靠理解记忆。答案另见[简答题总结](体系结构简答题.docx)

* 比较总线、多级网络、交叉开关的特点 (2003, 2008, 2009)
* 多处理机监听(目录)协议的工作原理 (2003, 2005, 2006, 2008, 2009, 2010, 2011, 2012, 2013)
* 四种松弛一致性模型的特点及硬件支持措施 (2003, 2006, 2007, 2009, 2010, 2012, 2014, 2015)
* 多处理机系统中进行时延隐藏的主要技术途径与原理 (2001, 2007, 2013)
* 多处理机的相关性与一致性 (2001, 2007, 2008, 2010, 2011, 2012, 2014, 2015)

## 其他

* 基于硬件的动态调度的优缺点 (2014, 2015)
* 比较同时多线程、粗粒度、细粒度多线程的特点 (2006, 2010, 2013)
* 同时多线程在体系结构实现上的基础 (2008, 2009)

## 年代久远的题目

这些题目都是很久以前的，考得可能性比较小

* 总线的出现至今对计算机产生了哪些影响 (2000)
* 减少Cache开销的策略 (2002)
* 目前实现盘阵列的三种主要方式 (2002)
* VLIW上性能优于向量处理器的示例 (2002)
* 超线性加速比的原因 (2002)
* 三种降低Cache命中时间的技术手段 (2003)
* 多指令流出受到的限制 (2003)
* CISC与RISC的设计原则 (2004)
* 用物理地址进行DMA时存在的问题 (2004)
* 两级Cache的原理 (2001)
* 激光打印机调至激光束的基本原理 (2000)
* TLB的原理 (2001)

# 三、计算题

## Amdahl定律

* 根据Amdahl定律计算加速比 (2000, 2001, 2005, 2007, 2008, 2011, 2012, 2014, 2015)

**解题思路：**

①使用Amdahl公式：

②用加速比公式：

## CPI与MIPS的计算

* 通过计算MIPS、性价比评价机器的性能(2000, 2002, 2003, 2009)

**解题思路：**利用MIPS，CPI，IC与时钟频率的关系

* 流水线的CPI与MPIS的计算 (2006, 2007)

**解题思路：**流水线CPI=理想CPI+延迟CPI 理想流水线加速比=流水线深度

* 向量机的CPI与MPIS计算 (2002, 2010, 2013, 2014, 2015)

**解题思路：**

①向量指令的编队：不含相关(或者有相关但可以使用链接的时候) ，*n*个编队需要*n*个钟鸣 (2013)

①当向量长度小于向量寄存器长度时 *T*=(1+*Tf*+1) + *N*-1，其中，*Tf*是功能部件需要的时间。指令不可链接时，每条指令的执行时间都需要包括向量长度，可链接时只需要计算一次向量长度 (2002, 2010, 2014, 2015)

③当向量长度大于向量寄存器长度时 ，其中，MVL是向量寄存器长度，N是向量长度，*Tloop*是循环额外开销，*Tstart*是向量部件起始时间，m是编队数量

④当向量长度趋于无穷大时，（FLOPS） (2002)

## Cache

* 多级Cache的失效率 (2001, 2002, 2007)

**解题思路：**失效率=失效次数/访问次数

* 存储带宽 (2000, 2003, 2008, 2013)

**解题思路：每秒**访存次数×每次访存需要传输的数据量

* 访存时间 (2001, 2005, 2011, 2013)

**解题思路：**

①平均访存时间：cache时间×命中率+MEM时间×失效率 (2001, 2011, 2013)

②命中时间：访问tag时间+tag比对时间 (2005)

* 地址格式 (2000, 2003)

**解题思路：**记住Cache地址与MEM地址的对应规则

## 分支预测

* 命中率、失效率等的计算 (2004)
* 平均延迟的计算 (2010)

**解题思路：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 访问BTB | 访问BPB | 实际动作 | 是否造成延迟 | 事件说明 |
| hit | 预测跳转 | 跳转 | 否 | 预测成功 |
| hit | 预测跳转 | 不跳转 | 是 | 预测跳转但是不跳转 |
| miss | \ | 成功 | 是 | 不在BTB就要延迟 |

## 同步开销的计算

* N个处理机进行栅栏同步的时间开销 (2004, 2012, 2014)

**解题思路：**因为总线的访问是串行的。n个进程竞争会导致n次总线访问。因此总开销为：

* k元组合树 (2005)

**解题思路：**组合树将大冲突化成并行的多个小冲突。每k个进程竞争一个锁和变量，当k个进程到达后形成父节点，并在父节点进行另一级竞争。总开销为：

* 并行/串行程序比较 (2011, 2014)

**解题思路：**串行程序不用考虑临界区条件，即不用考虑加锁和释放锁的开销

## 可靠性计算

* MTTF的计算 (2013, 2015)

**解题思路：**几个事件导致的综合MTTF：

## 年代久远的题目

* booth乘法 (2002)

**解题思路：**量他不会考这个

# 四、综合题

## 指令依赖分析、指令调度与循环展开

* 指令依赖分析 (2010, 2011, 2013)

**解题思路：**理解WAR，RAW和WAW相关

* 给出指令的延迟，求单次迭代时钟周期数；动态调度+循环展开，使得无延迟，求平均单词迭代时钟周期数 (2000, 2003, 2007, 2008, ***2010***, 2011, 2012)

**解题思路：**先插stall，再展开，再把相应的指令插进stall中。特别要考虑分支延迟槽

* 超标量处理器的指令流出 (2005)

**解题思路：**分析指令相关性

## 流水线时空图

**解题思路：**画时空图，注意数据与结构相关。

* 超标量处理器的流水时空图 (2001, 2007, 2009, 2012)

注意，单次循环时间的计时方法：从本次循环到下次循环开始前or本次循环指令执行结束

* 通过画时空图求CPI、吞吐率、效率、加速比 (2000, 2001, 2004)

## 编译器技术

**解题思路：**理解记忆。

* 软流水 (2006, 2015)
* 前瞻执行的处理与执行 (2006, 2009)

## Tomasulo与记分牌的算法

**解题思路：**理解记忆

* 填表 (2005)
* 算法描述 (2004, 2006, 2008, 2013)

## 多级中断

**解题思路：**注意区分中断优先级和是否允许嵌入式中断

* 程序执行轨迹 (2006, 2009)

## 年代久远的题目

这些题目都是很久以前的，考得可能性比较小

* 设计微指令格式，画出实现4题秉性控制的原理框图(2000)
* 设计程序使Cache的命中率满足一定条件 (2001)
* Amdahl定律：悲观定律与乐观定律 (2001)
* 给出程序与环境，给出加速方法 (2002)
* 多处理机的Cache一致性协议分析(2004)

# 五、总结

第一大题题型应当为填空题，题目以中文出题为主(06年主要英文)，内容分布比较均匀，可能出现历年没有出现过的新题，但大部分应当是历年里出现过的或者常识上应该知道的内容。

计算题的题材的重点包括Amdahl定律、cache相关计算、CPI以及MIPS的计算，可能涉及MTTF的计算。向量题材的题目近两年比较热，也比较新，其他的题目比较老套。

综合题的题材包括指令或者流水线的分析调度；Tomasulo与记分牌近两年没出现，但依然有可能出现，目测类似于算法描述；中断异常、前瞻、软流水等内容。

近年来简答题比较固定，几乎必考的内容包括：

* 多处理机监听(目录)协议的工作原理
* 四种松弛一致性模型的特点及硬件支持措施
* 多处理机的相关性与一致性
* 基于硬件的动态调度的优缺点
* 比较同时多线程、粗粒度、细粒度多线程的特点