[1. 其他的一些加速 性能提升模式 1](#_Toc29533)

[2. 乱序执行技术 2](#_Toc13624)

[2.1. 处理器乱序执行 2](#_Toc3025)

[2.2. 编译器指令重排 3](#_Toc13371)

[2.3. 乱序的分类 3](#_Toc29770)

[2.4. 允许程序员显式的告诉处理器对某些地方禁止乱序执行.这种机制就是所谓内存屏障 3](#_Toc3688)

[3. 采用乱序执行 优化 指令依赖的情况，流水线就会停滞 4](#_Toc24217)

[3.1. ，乱序执行可以加快芯片的运行速度将近两倍 5](#_Toc18969)

[3.2. 处理器乱序规则 5](#_Toc3156)

# 其他的一些加速 性能提升模式

指令调度是在一段程序中重新排序指令,使得程序中相关指令尽可能消除.  
  
乱序执行则采用跳过相关指令,先执行不相关指令的方法加速.  
  
超标量处理机基于重复硬件运行多条流水线,使得在一个时钟周期内可以发射或流出多条指令.  
  
超流水线处理机将阶段更加细分, 级数大于等于8级的流水线称为超流水线.  
  
超长指令字是把不相关的多条指令组合在一起作为一条位数超长的指令一同发射到流水线中,并行执行.

标量  
将一条指令分成若干个周期处理以达到多条指令重叠处理,从而提高cpu部件利用率的技术叫做标量流水技术。超级标量是指cpu内一般能有多条流水线,这些流水线能够并行处理。在单流水线结构中，指令虽然能够重叠执行，但仍然是顺序的,每个周期只能发射(issue)或退休(retire)一条指令。超级标量结构的cpu支持指令级并行，每个周期可以发射多条指令(2-4条居多)。可以使得cpu的IPC(InstructionPerClock)>，从而提高cpu处理速度。超级标量机能同时对若干条指令进行译码，将可以并行执行的指令送往不同的执行部件，在程序运行期间，由硬件(通常是状态记录部件和调度部件)来完成指令调度。超级标量机主要是借助硬件资源重复(例如有两套译码器和ALU等)来实现空间的并行操作。熟知的pentium系列(可能是p-II开始),还有SUNSPARC系列的较高级型号,以及

# 乱序执行技术

## **处理器乱序执行**

随着处理器流水线技术和多核技术的发展,目前的高级处理器通过提高内部逻辑元件的利用率来提高运行速度,通常会采用乱序执行技术.这里的乱序和上面谈到烧水煮茶的道理是一样的.

先来看一张处理器的简要结构图:

处理器从L1 Cache中取出一批指令,分析找出那些不存在相互依赖的指令,同时将其发射到多个逻辑单元执行,比如现在有以下几条指令:

一个指令即一个步骤

可以说乱序执行技术是处理器为提高运算速度而做出违背代码原有顺序的优化.在单核时代,处理器保证做出的优化不会导致执行结果远离预期目标,但在多核环境下却并非如此.

## **编译器指令重排**

除了上述由处理器和缓存引起的乱序之外,现代编译器同样提供了乱序优化.之所以出现编译器乱序优化其根本原因在于处理器每次只能分析一小块指令,但编译器却能在很大范围内进行代码分析,从而做出更优的策略,充分利用处理器的乱序执行功能.

## **乱序的分类**

现在来总结下所有可能发生乱序执行的情况:

* 现代处理器采用指令并行技术,在不存在数据依赖性的前提下,处理器可以改变语句对应的机器指令的执行顺序来提高处理器执行速度
* 现代处理器采用内部缓存技术,导致数据的变化不能及时反映在主存所带来的乱序.
* 现代编译器为优化而重新安排语句的执行顺序

## 允许程序员显式的告诉处理器对某些地方禁止乱序执行.这种机制就是所谓内存屏障

看到乱序执行初始目的是为了提高效率,但是它看来其好像在这多核时代不尽人意,其中的某些”自作聪明”的优化导致多线程程序产生各种各样的意外.因此有必要存在一种机制来消除乱序执行带来的坏影响,也就是说应该允许程序员显式的告诉处理器对某些地方禁止乱序执行.这种机制就是所谓内存屏障.不同架构的处理器在其指令集中提供了不同的指令

这样将根据个电路单元的状态和各指令能否提前执行的具体情况分析后，将能提前执行的指令立即发送给相应电路单元执行，在这期间不按规定顺序执行指令，然后由重新排列单元将各执行单元结果按指令顺序重新排列。采用乱序执行技术的目的是为了使CPU内部电路满负荷运转并相应提高了CPU的运行程序的速度。分枝技术：（branch）指令进行运算时需要等待结果，一般无条件分支只需要按指令[顺序执行](https://baike.baidu.com/item/%E9%A1%BA%E5%BA%8F%E6%89%A7%E8%A1%8C" \t "https://baike.baidu.com/item/%E4%B9%B1%E5%BA%8F%E6%89%A7%E8%A1%8C/_blank)，而条件分枝必须根据处理后的结果，再决定是否按原先顺序进行

# 采用乱序执行 优化 指令依赖的情况，流水线就会停滞

在按序执行中，一旦遇到指令依赖的情况，流水线就会停滞，如果采用乱序执行，就可以跳到下一个非依赖指令并发布它。这样，执行单元就可以总是处于工作状态，把时间浪费减到最少。乱序执行可以允许在发布指令3前发布指令4～8，而且这些指令的执行结果可以在指令3引出后立即引出（按序引出对X86 CPU来说是必需的），实际解码率又会增加25%。不过PⅡ和K6从乱序执行中得到的好处有限，因为如果CPU遇到指令依赖情况，它必须找到更多的非依赖指令进行发布。

## ，乱序执行可以加快芯片的运行速度将近两倍

在多数情况下，乱序执行可以加快芯片的运行速度，有时候加速明显。21264的速度就是21164的将近两倍，而Pentium Pro的速度也比奔腾快了一大截。经过重大改进的最新MIPS架构R10000也采用了乱序执行。

## **处理器乱序规则**

上面我们说了处理器会发生指令重排,现在来简单的看看常见处理器允许的重排规则,换言之就是处理器可以对那些指令进行顺序调整:

| **处理器** | **Load-Load** | **Load-Store** | **Store-Store** | **Store-Load** | **数据依赖** |
| --- | --- | --- | --- | --- | --- |
| x86 | N | N | N | Y | N |
| PowerPC | Y | Y | Y | Y | N |
| ia64 | Y | Y | Y | Y | N |

格中的Y表示前后两个操作允许重排,N则表示不允许重排.与这些规则对应是的禁止重排的内存屏障.

注意:处理器和编译都会遵循数据依赖性,不会改变存在数据依赖关系的两个操作的顺序.所谓的数据依赖性就是如果两个操作访问同一个变量,且这两个操作中有一个是写操作,那么久可以称这两个操作存在数据依赖性.举个简单例子:

a=100;//write

b=a;//read

或者a=100;//writea=2000;//write

或者a=b;//read

b=12;//write

作者简介

艾提拉 艾龙 attilax，法名 st attilax akbar rinpoche 圣阿提拉科斯阿克巴仁波切

头街软件技术大师 uke组织创始人

学术成就，完善的20大知识体系，拥有uke学院硕士博士学位

从事软件互联网行业技术背景十二年，csdn排名TOP57

长年从事软件互联网技术与管理，预计出版多本心得分享书籍

擅长技术与管理与文化 致力于标准化事业

Qq 1466519819 小号112237553

微信attilax 小号attilax201708

作者:: 绰号与头街 :老哇的爪子claw of Eagle 偶像破坏者Iconoclast image-smasher 神的使者（Messenger of God）及守望者（Watch Man

捕鸟王"Bird Catcher kok 虔诚者Pious 宗教信仰捍卫者 Defender Of the Faith. 卡拉卡拉红斗篷 Caracalla red cloak KOA万兽之王 纵火者

头街来源：神的使者（Messenger of God）及守望者（Watch Man来源于圣经

老哇的爪子claw of Eagle来源于印加帝国

KOA万兽之王 来源于婆罗门大神森林中修炼

简称：： st Emir Attilax Akbar 圣 埃米尔 阿提拉克斯 阿克巴

全名：：st Emir Attilax Akbar bin Mahmud bin attila bin Solomon bin adam Al Rapanui 圣 埃米尔 阿提拉克斯 阿克巴 本 马哈茂德 本 阿提拉 本 所罗门 本亚当 阿尔 拉帕努伊

常用名：艾提拉（艾龙）， EMAIL:1466519819@qq.com

喜欢的绰号 捕鸟王纵火者 老瓦的爪子兔子的耳朵

喜欢的头街 st圣 仁波切 大师 马斯塔 艺术家 博士 诗人 国王

头衔：

|  |  |
| --- | --- |
| uke | Emir Uke部落首席大酋长，ati协会创始人 仁波切马斯塔 埃米尔  uke总部o2o负责人，全球网格化项目创始人，  圣阿提拉克斯国王 |
| 科技领域 | UTSC uke技术标准化委员会委员长 uke 首席cto 软件部门总监 技术部副总监 研发部门总监主管 产品部副经理 项目部副经理 uke科技研究院院长 uke软件培训大师  Ati组织科研研究院创始人 |
| 文艺领域 | ， ，， uke机车协会主任 uke纹身协会  uke交友协会会长 uke捕猎协会会长  Ati文艺协会会长 ati文学协会 |
| 行政领域 | Gchsp总裁 gchsp常委 GsP创始人 |
| 媒体传播领域 | uke出版社编辑总编 宣传布道总策划  Ati传媒总部 |
| 渔猎军事领域 | uke保安部首席大队长  Uke 户外运动协会理事长 度假村首席大村长  Ati打猎协会 |
| 法学 | 法学研究会 制度研究会 |
| 管理领域 | 工商管理学 公共管理与社会服务  ,uke制度检查委员会副会长 |
| 教育领域 | uec学院校长， uecip图像处理机器视觉专业系主任 uke文档检索专业系主任  Uke图像处理与机器视觉学院首席院长  uke终身教育学校副校长  靓号研究院 |
| 经济领域 | uke波利尼西亚区大区连锁负责人 汤加王国区域负责人 uke克尔格伦群岛区连锁负责人，莱恩群岛区连锁负责人，uke布维岛和南乔治亚和南桑威奇群岛大区连锁负责人  Uke软件标准化协会理事长理事长 Uke 数据库与存储标准化协会副会长  直达巴士西北区负责人 直达巴士长沙与西安分部部长  润昌通讯软件事业部总裁 执行长 分部负责人 执行委员会主席  Ati经济研究所 |
| 历史领域 | 历史事业部 ati历史研究院 |
| 社会科学领域 | 社科学院 ati文化部 |
| 自然科学领域 | Uke研究院院长兼首席研究员 科学家  Ati自然科学研究院 |
| 宗教神学领域 | uke宗教与文化融合事务部部长 大师master  uke制度与重大会议委员会委员长 ati宗教事务所 |
| 医学领域 | Uke医院 与医学院方面的创始人 |

转载请注明来源：attilax的专栏 <http://blog.csdn.net/attilax>

<http://www.cnblogs.com/attilax/>

Microblog

<http://weibo.com/u/5941179815> (common attilax)

<https://weibo.com/p/1005055941179815> （attilax201707,bek weibo）

<http://weibo.com/u/5487832265> (tech,for blog auto gene)

知乎空间

https://www.zhihu.com/people/ati-att/activities

Qq 1466519819 小号112237553

微信attilax 小号attilax201708

微博 attilax2016 小号attilax201707

--Atiend v23