

计算机系统结构

第三讲 动态分支预测

谢长生

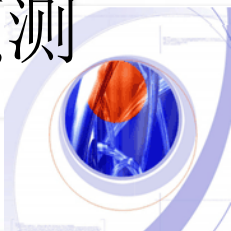
武汉光电国家研究中心



Computer Architecture

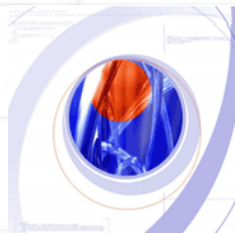
1.分支预测的重要性与现有方法的不足

- 所开发的 **ILP** 越多，控制相关的制约就越大，分支预测就要有更高的准确度。
- 本节中介绍的方法对于每个时钟周期流出多条指令（若为 **n** 条，就称为 **n流出**）的处理机来说非常重要。原因在于：
 - 在 **n**-流出的处理机中，遇到分支指令的可能性增加了 **n** 倍。
 - 要给处理器连续提供指令，就需要准确地预测分支。



2.解决方法——动态分支预测

- 动态分支预测
 - 在程序运行时，根据分支指令过去的表现来预测其将来的行为。
 - 如果分支行为发生了变化，预测结果也跟着改变。
 - 有更好的预测准确度和适应性。
- 分支预测的有效性取决于
 - 预测的准确性
 - 预测正确和不正确两种情况下的分支开销
 - 决定分支开销的因素：
 - 流水线的结构
 - 预测的方法
 - 预测错误时的恢复策略等



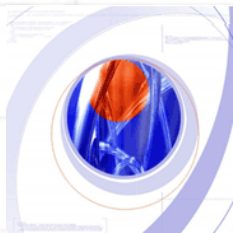
3. 目标与关键问题

— 采用动态分支预测技术的目标

- 预测分支是否成功
- 尽快找到分支目标地址（或指令）
（避免控制相关造成流水线停顿）

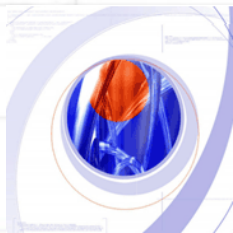
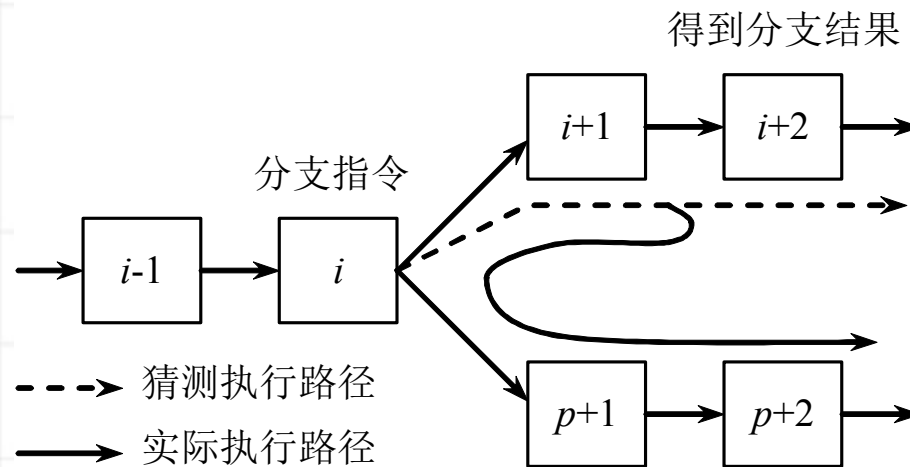
— 需要解决的关键问题

- 如何记录分支的历史信息，要记录哪些信息？
- 如何根据这些信息来预测分支的去向，甚至提前取出分支目标处的指令？



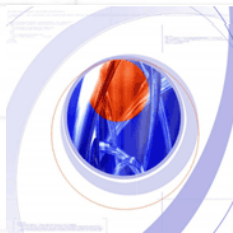
4. 预测错误时的处理方法

- 在预测错误时，要作废已经预取和分析的指令，恢复现场，并从另一条分支路径重新取指令。



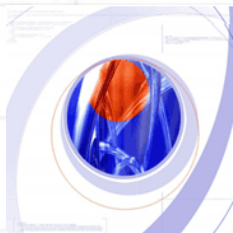
5-1. 分支历史表 BHT

- Branch History Table, BHT
- 最简单的动态分支预测方法。
- 用**BHT**来记录分支指令最近一次或几次的执行情况（成功还是失败），并据此进行预测。



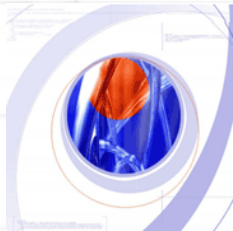
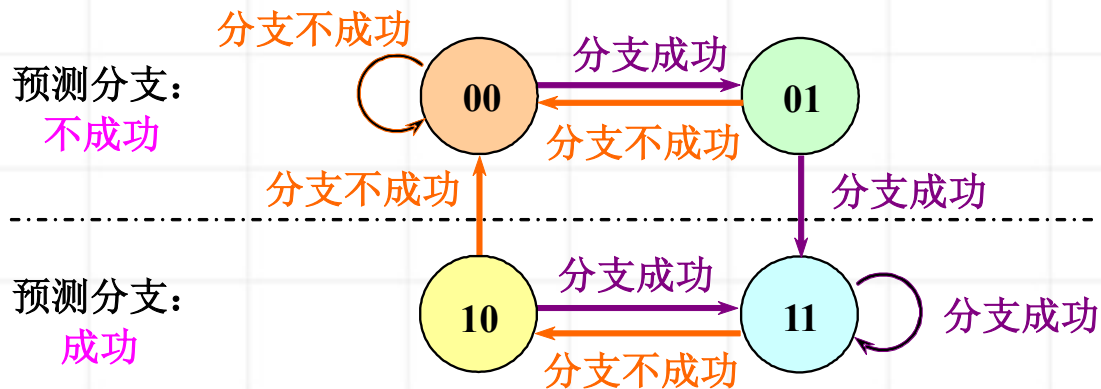
5-2. 最简单的分支历史表

- 只有1个预测位的分支预测表
 - 记录分支指令最近一次的历史，BHT中只需要1位二进制位（最简单）



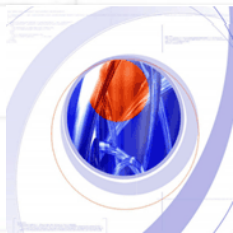
5-3. 两位预测位的分支历史表

- 2个预测位的分支预测
 - 提高预测的准确度
 - 研究结果表明：两位分支预测的性能与 n 位 ($n > 2$) 分支预测的性能差不多。



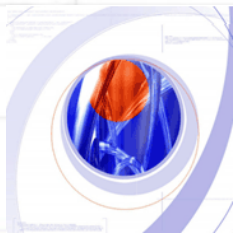
5-4. 两位BHT的操作步骤

- 分支预测
 - 当分支指令到达译码段（ID）时，根据从BHT读出的信息进行分支预测。
 - 若预测正确，就继续处理后续的指令，流水线没有断流。否则，就要作废已经预取和分析的指令，恢复现场，并从另一条分支路径重新取指令。
- 状态修改。



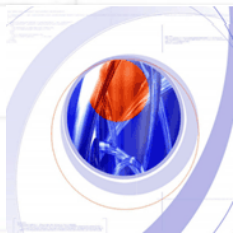
5-5. BHT的作用范围

- 判定分支是否成功所需的时间大于确定分支目标地址所需的时间。
 - 前述5段经典流水线：由于判定分支是否成功和计算
 - 分支目标地址都是在ID段完成，所以BHT方法不会给该流水线带来好处。



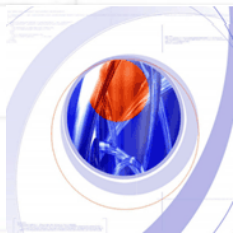
5-6. BHT的作用效果

- 研究结果表明
 - 对于SPEC89测试程序来说，具有大小为4KB的BHT的预测准确率为82%~99%。
 - 一般来说，采用4KB的BHT就可以了。



5-7. BHT的实现

- BHT可以跟分支指令一起存放在指令Cache中。
- 也可以用一块专门的硬件来实现。

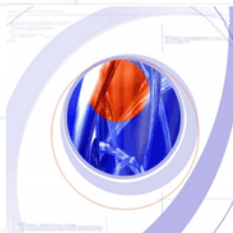


6-1. 分支目标缓冲器BTB

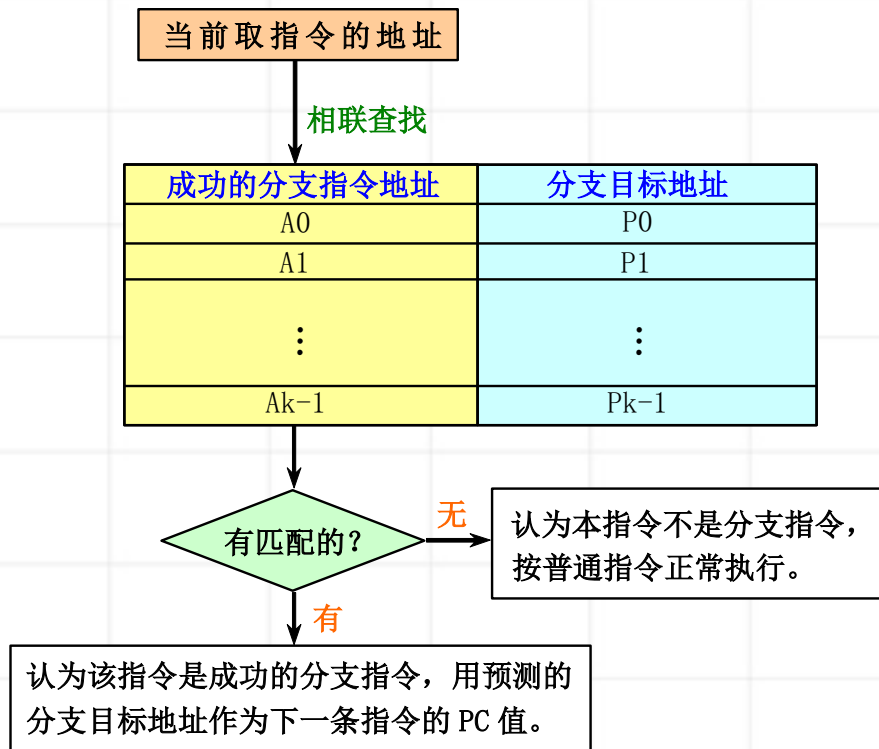
目标： 将分支的开销降为 0

方法： 分支目标缓冲

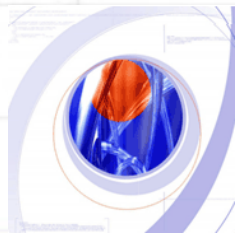
- 将分支成功的分支指令的地址和它的分支目标地址都放到一个缓冲区中保存起来，缓冲区以分支指令的地址作为标识。
- 这个缓冲区就是**分支目标缓冲器**（Branch-Target Buffer，简记为**BTB**，或者**分支目标Cache**（Branch-Target Cache））。



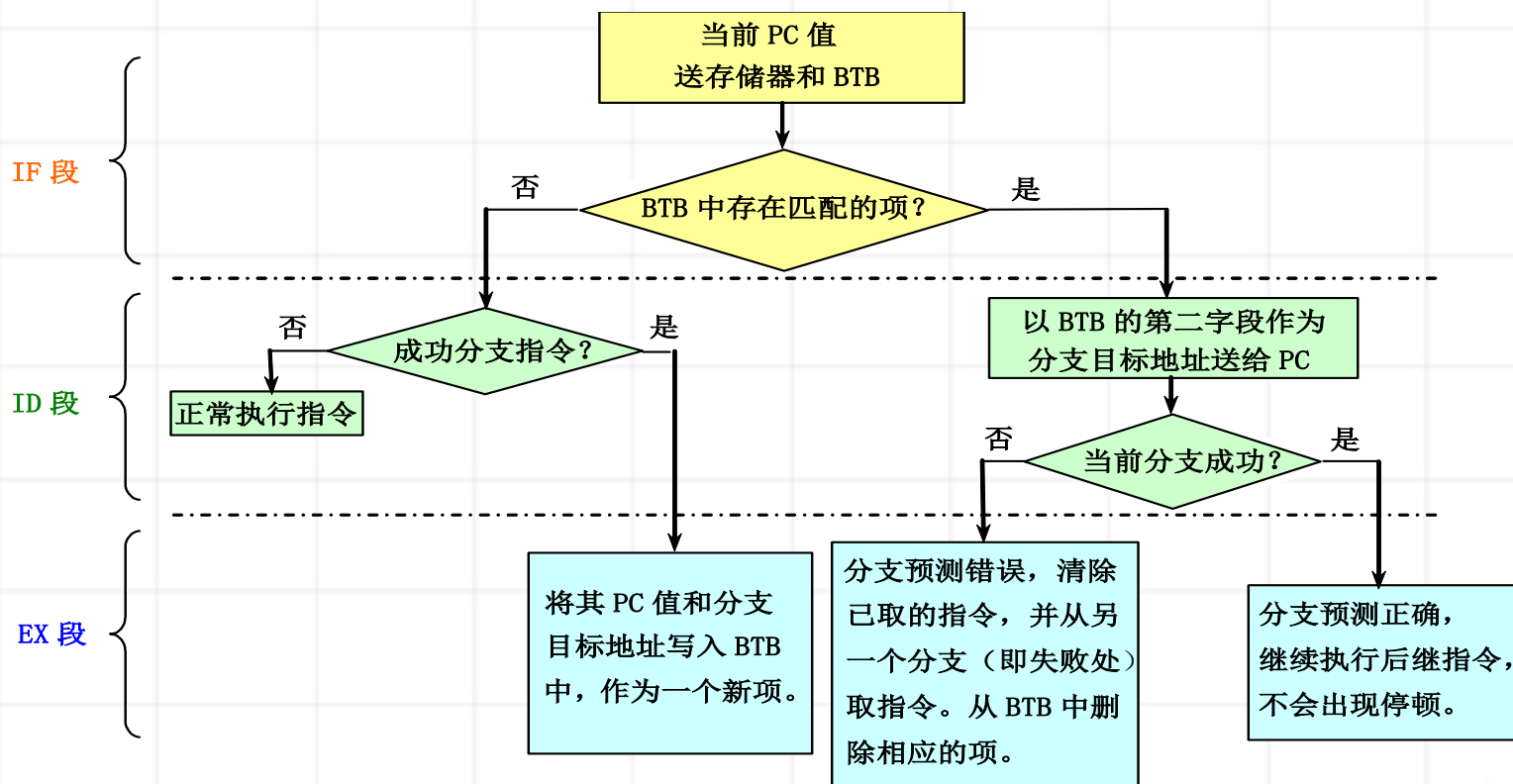
6-2. BTB的结构



- 看成是用专门的硬件实现的一张表格。
- 表格中的每一项至少有两个字段：
 - ❑ 执行过的成功分支指令的地址；
(作为该表的匹配标识)
 - ❑ 预测的分支目标地址：

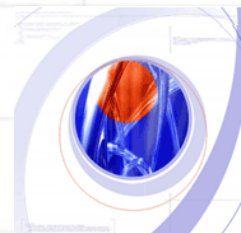


6-3. BTB的操作



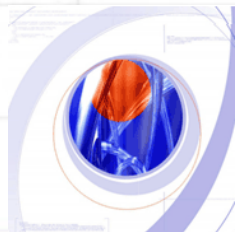
6-4. 各种情况下的延迟

指令在BTB中?	预测	实际情况	延迟周期
是	成功	成功	0
是	成功	不成功	2
不是		成功	2
不是		不成功	0



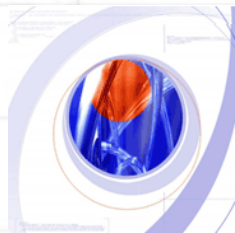
7-1. 改进BTB——提升预测准确率

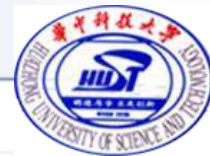
- 在分支目标缓冲器中增设一个至少是两位的“分支历史表”字段



7-2.改进BTB——降低访存开销

更进一步，在表中对于每条分支指令都存放若干条分支目标处的指令，就形成了分支目标指令缓冲器。





谢谢大家

