

实验一、 流水线相关和指令调度

实验目的

- 熟练掌握 WinMIPS64 模拟器的使用；
- 加深对计算机流水线基本概念的理解；
- 进一步了解 MIPS 基本流水线各段的功能以及基本操作；
- 加深对数据相关、结构相关的理解，了解相关对 CPU 性能的影响；
- 了解解决数据相关的方法，掌握如何使用定向技术来减少数据相关带来的暂停。
- 加深对循环级并行性、指令调度技术、循环展开技术的理解；
- 熟悉用指令调度技术来解决流水线中的数据相关的方法；
- 了解循环展开、指令调度等技术对 CPU 性能的改进。

实验平台

- WinMIPS64 模拟器

实验内容和步骤

一、流水线相关

1. 用 WinMIPS64 模拟器执行下列三个程序（任选一个）：

- 求阶乘程序 `factorial.s`
- 插入排序程序 `isort.s`
- 乘法计算程序 `mult.s`

分别以步进、连续、设置断点的方式运行程序，观察程序在流水线中的执行情况，观察 CPU 中寄存器和存储器的内容。

掌握 WinMIPS64 的操作和使用。

2. 用 MIPS64 汇编语言编写代码文件 `*.s`，程序中应包括结构相关。用 WinMIPS64 模拟器运行你编写的程序，通过模拟：

- 找出存在结构相关的指令对以及相应的结构相关的部件；
- 记录由结构相关引起的暂停时钟周期数，计算暂停时钟周期数占总执行周期数的百分比；
- 论述结构相关对 CPU 性能的影响，讨论解决结构相关的方法。

3. 用 MIPS64 汇编语言编写代码文件 `*.s`，程序中应包括数据相关。在不采用定向技术的情况下，用 WinMIPS64/WinDLX 模拟器运行存在数据相关的程序。记录数据相关引起的暂停时钟周期数以及程序执行的总时钟周期数，计算暂停时钟周期数占总执行周期数的百分比。

4. 在采用定向技术的情况下，用 WinMIPS64 模拟器再次运行程序。重复上述 3 中的工作，

并计算采用定向技术后性能提高的倍数。

☞ 二、指令调度

1. 用指令调度技术解决流水线中的结构相关与数据相关

- (1) 用 MIPS64 汇编语言编写代码文件 *.s, 程序中应包括数据相关与结构相关 (你可以自己设置各个功能单元的延迟时间)
- (2) 用 WinMIPS64 模拟器运行你所写的程序。记录程序执行过程中各种相关发生的次数、发生相关的指令组合, 以及程序执行的总时钟周期数;
- (3) 采用指令调度技术对程序进行指令调度, 消除相关 (手工调度^{^^});
- (4) 用 WinMIPS64 模拟器运行调度后的程序, 观察程序在流水线中的执行情况, 记录程序执行的总时钟周期数;
- (5) 根据记录结果, 比较调度前和调度后的性能。论述指令调度对于提高 CPU 性能的意义。

2. 用循环展开、寄存器换名以及指令调度提高性能

- (1) 用 MIPS64 汇编语言编写代码文件 *.s, 程序中包含一个循环次数为 4 的整数倍的简单循环;
- (2) 用 WinMIPS64 模拟器运行该程序。记录执行过程中各种相关发生的次数以及程序执行的时钟周期数;
- (3) 将循环展开 3 次, 将 4 个循环体组成的代码代替原来的循环体, 并对程序做相应的修改。然后, 对新的循环体进行寄存器换名和指令调度;
- (4) 用 WinMIPS64 模拟器运行修改后的程序, 记录执行过程中各种相关发生的次数以及程序执行的总时钟周期数;
- (5) 根据记录结果, 比较循环展开、指令调度前后的性能。

撰写实验报告

- 根据上述实验内容和步骤, 结合自己所写的程序以及进行的相关分析, 撰写实验报告。
- 注: 需要按照实验报告模板的格式要求进行撰写。

拓展实验: 自己实现 Scoreboard、Tomasulo 算法仿真器。[课外]