# Lab5 Tomasulo和cache一致性

# 模拟器使用

### 实验目的：

1. 熟悉Tomasulo模拟器和cache一致性模拟器（监听法和目录法）的使用
2. 加深对Tomasulo算法的理解，从而理解指令级并行的一种方式-动态指令调度
3. **掌握Tomasulo算法在指令流出、执行、写结果各阶段对浮点操作指令以及load和store指令进行什么处理；给定被执行代码片段，对于具体某个时钟周期，能够写出保留站、指令状态表以及浮点寄存器状态表内容的变化情况。**
4. 理解监听法和目录法的基本思想，加深对多cache一致性的理解
5. **做到给出指定的读写序列，可以模拟出读写过程中发生的替换、换出等操作，同时模拟出cache块的无效、共享和独占态的相互切换**

### 实验要求：

1. **Tomasulo算法模拟器**

**使用模拟器进行以下指令流的执行并对模拟器截图、回答问题**

L.D F6, 21（R2）

L.D F2, 0（R3）

MUL.D F0, F2, F4

SUB.D F8, F6, F2

DIV.D F10, F0, F6

ADD.D F6, F8, F2

假设浮点功能部件的延迟时间：加减法2个周期，乘法10个周期，load/store2个周期，除法40个周期。

1. **分别截图（当前周期2和当前周期3），请简要说明load部件做了什么改动**
2. **请截图（MUL.D刚开始执行时系统状态），并说明该周期相比上一周期整个系统发生了哪些改动（指令状态、保留站、寄存器和Load部件）**
3. **简要说明是什么相关导致MUL.D流出后没有立即执行**
4. **请分别截图（15周期和16周期的系统状态），并分析系统发生了哪些变化**
5. **回答所有指令刚刚执行完毕时是第多少周期，同时请截图（最后一条指令写CBD时认为指令流执行结束）**
6. **多cache一致性算法-监听法**
7. **利用模拟器进行下述操作，并填写下表**

|  |  |  |  |
| --- | --- | --- | --- |
| **所进行的访问** | **是否发生了替换？** | **是否发生了写回？** | **监听协议进行的操作与块状态改变** |
| CPU A 读第5块 |  |  |  |
| CPU B 读第5块 |  |  |  |
| CPU C 读第5块 |  |  |  |
| CPU B 写第5块 |  |  |  |
| CPU D 读第5块 |  |  |  |
| CPU B 写第21块 |  |  |  |
| CPU A 写第23块 |  |  |  |
| CPU C 写第23块 |  |  |  |
| CPU B 读第29块 |  |  |  |
| CPU B 写第5块 |  |  |  |

1. **请截图，展示执行完以上操作后整个cache系统的状态。**
2. **多cache一致性算法-目录法**
3. **利用模拟器进行下述操作，并填写下表**

|  |  |
| --- | --- |
| **所进行的访问** | **监听协议进行的操作与块状态改变** |
| CPU A 读第6块 |  |
| CPU B读第6块 |  |
| CPU D 读第6块 |  |
| CPU B 写第6块 |  |
| CPU C 读第6块 |  |
| CPU D写第20块 |  |
| CPUA写第20块 |  |
| CPU D写第6块 |  |
| CPU A 读第12块 |  |

1. **请截图，展示执行完以上操作后整个cache系统的状态。**
2. **综合问答**
3. **目录法和监听法分别是集中式和基于总线，两者优劣是什么？（言之有理即可）**
4. **Tomasulo算法相比Score Board算法有什么异同？（简要回答两点：1.分别解决了什么相关，2.分别是分布式还是集中式）（参考第五版教材）**
5. **Tomasulo算法是如何解决结构、RAW、WAR和WAW相关的？（参考第五版教材）**

### 实验验收：

**只需要提交实验报告（pdf格式），提交邮箱、截止日期和文档命名请看github实验主页**