

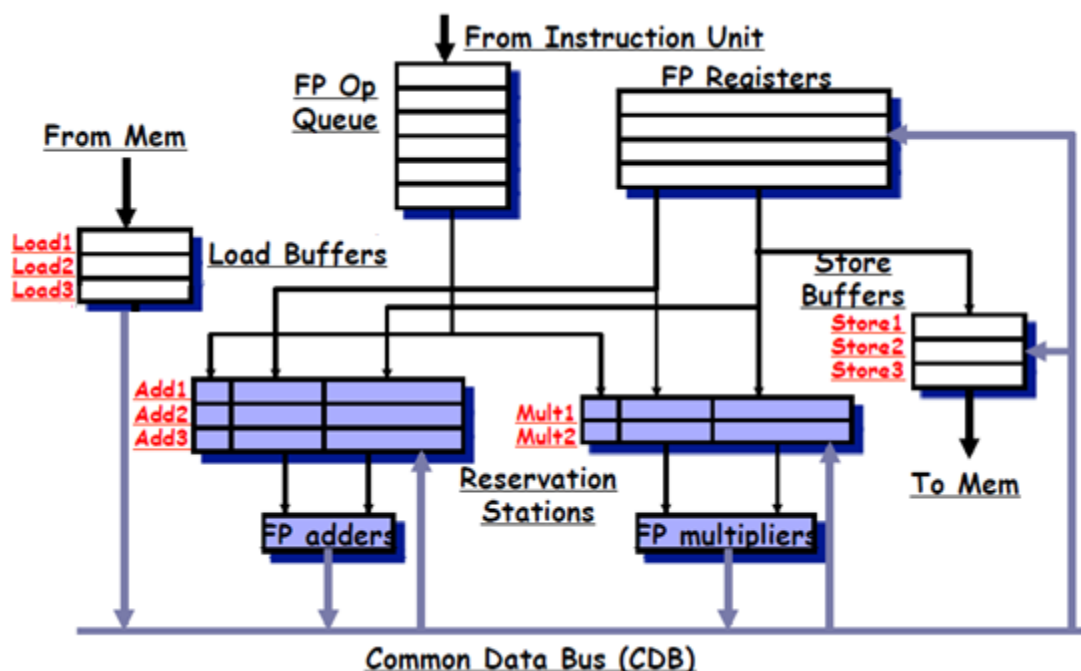
系统结构第二次实验：设计实现 Tomasulo 调度算法

一、实验原理

Tomasulo 算法

Tomasulo 算法以硬件方式实现了寄存器重命名，允许指令乱序执行，是提高流水线的吞吐率和效率的一种有效方式。该算法首先出现在 IBM360/91 处理机的浮点处理部件中，后广泛应用于现代处理器设计中。

假设浮点处理部件结构如下图所示。浮点处理部件从取指单元接收指令，存入浮点操作队列。浮点操作队列每拍最多发射 1 条指令给浮点加法器或浮点乘除法器。浮点处理部件包含一个浮点加法器和一个浮点乘除法器。浮点加法器为两段流水线，输入端有三个保留站 A1、A2、A3，浮点乘除法器为六段流水线，输入端有两个保留站 M1，M2。当任意一个保留站中的两个源操作数到齐后，如果对应的操作部件空闲，可以把两个操作数立即送到浮点操作部件中执行。Load Buffer 和 Store Buffer 各缓存三条访存操作。



二、实验要求

- 本实验为分组实验，三人一组完成，请同学们自由组合。
- 设计实现 Tomasulo 算法模拟器，不限语言（推荐 Java，在其他机器上容易直接运行），要求：
 - Tomasulo 算法模拟器能够执行浮点加、减、乘、除运算及 LOAD 和 STORE 操作，为了简化起见，我们在下表中给出了各种操作的具体描述。

指令格式	指令说明	指令周期	保留站/缓冲队列项数
ADDD F1,F2,F3	F1, F2, F3 为浮点寄存器 寄存器至少支持 (F0~F10)	2 个周期	3
SUBD F1, F2, F3	同上	2 个周期	
MULD F1, F2, F3	同上	10 个周期	2
DIVD F1, F2, F3	同上	40 个周期	
LD F1, ADDR	F1 为寄存器, ADDR 为地址, $0 \leq \text{ADDR} < 4096$	2 个周期	3
ST F1, ADDR	同上	2 个周期	3

- 支持单步执行及连续执行 (n 条指令)，实时显示算法的运行状况，包括各条指令的运行状态、各寄存器以及内存的值、保留站(Reservation Stations)状态、Load Buffer 和 Store Buffer 缓存的值等；
- 程序执行完毕后，能够显示指令执行周期数等信息；
- 为了简化设计，建议模拟器提供编辑内存值功能，以便实现数据输入；浮点除法可不作除 0 判断；
- 能够以文本方式输入指令序列。
- 显示界面自由设计，下图仅供参考：



三、提交内容

实验要求提供的内容：

- 实验文档;
- 源代码以及可执行文件（**请尽量保证可执行文件能够无障碍运行**）。

四、评分标准

- 未完成：60 分以下；
- 实现了实验要求中的各项指标，但是算法实现有误：60 分—85 分；
- 算法实现无误，但界面内容不够清晰，文档阐述不清晰条理：85 分—95 分；
- 很好的完成了算法的实现，算法运行信息翔实，界面清晰，代码风格良好：95 分—100 分；
- 抄袭者和“好心人”同记 0 分