**Tomasulo算法**

**1、实验目的**

（1）加深对指令级并行性及其开发的理解。

（2）加深对Tomasulo算法的理解。

（3）掌握Tomasulo算法在指令流出、执行、写回各阶段对浮点指令和Load/ Store指令进行的操作。

（4）掌握采用了Tomasulo算法的浮点处理部件的结构。

（5）掌握保留站的结构

（6）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。

**2、实验平台**

实验平台采用Tomasulo算法模拟器。

实验平台采用Tomasulo算法模拟器。

**3、实验内容**

（1）假设浮点功能部件的延迟时间为：加减法2个时钟周期，乘法10个时钟周期，除法40个时钟周期，Load部件2个时钟周期。

① 对于下面的代码段，给出当指令MUL.D即将写回时，保留站、Load缓冲器以及寄存器状态表中的内容。

L.D F6, 24(R2)

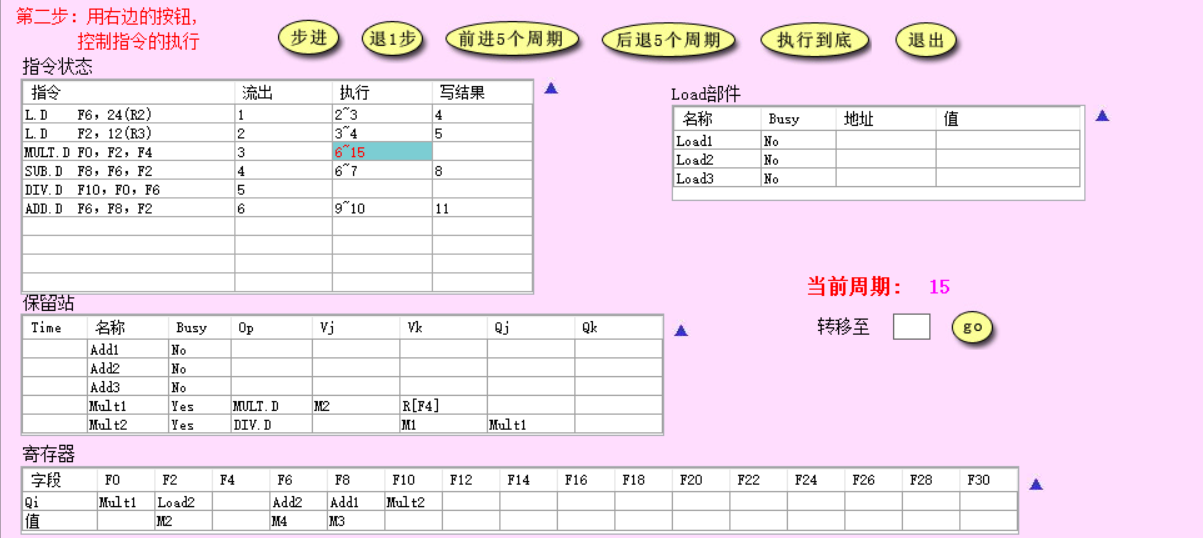
L.D F2, 12(R3)

MUL.D F0, F2, F4

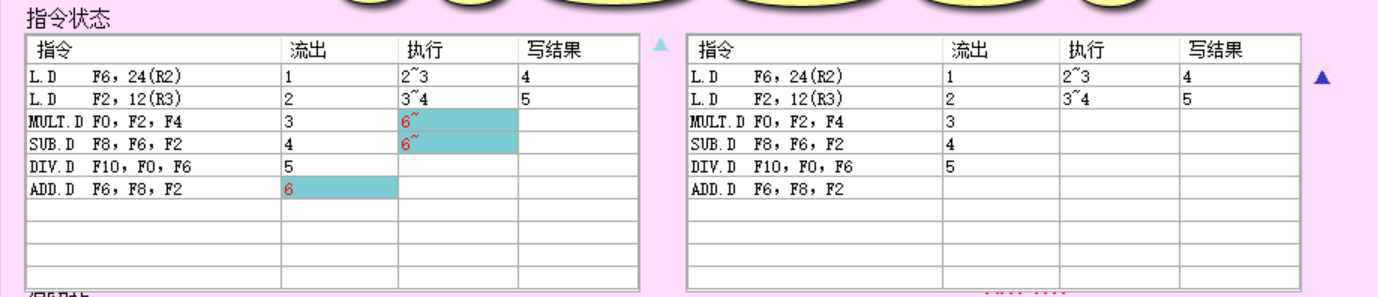
SUB.D F8, F6, F2

DIV.D F10, F0, F6

ADD.D F6, F8, F2



②按步进方式执行上述代码，利用模拟器的“小三角按钮”的对比显示功能，观察每一个时钟周期前后各信息表中内容的变化情况。



（2）对于与上面相同的延迟时间和代码段：

①给出在第3个时钟周期时保留站的内容。



②步进5个时钟周期，给出这时保留站、Load缓冲器以及寄存器状态表中的内容。



③再步进10个时钟周期，给出这时保留站、Load缓冲器以及寄存器状态表中的内容。



1. 假设浮点功能部件的延迟为：加减法3个时钟周期，乘法8个时钟周期，除法40个时钟周期。自己编写一段程序（要在实验报告中给出），重复上述步骤（2）的工作。

编写的代码如下：

L.D F6, 24(R2)

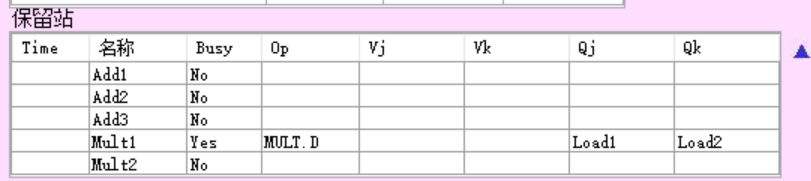
L.D F2, 12(R3)

MUL.D F0, F6, F2

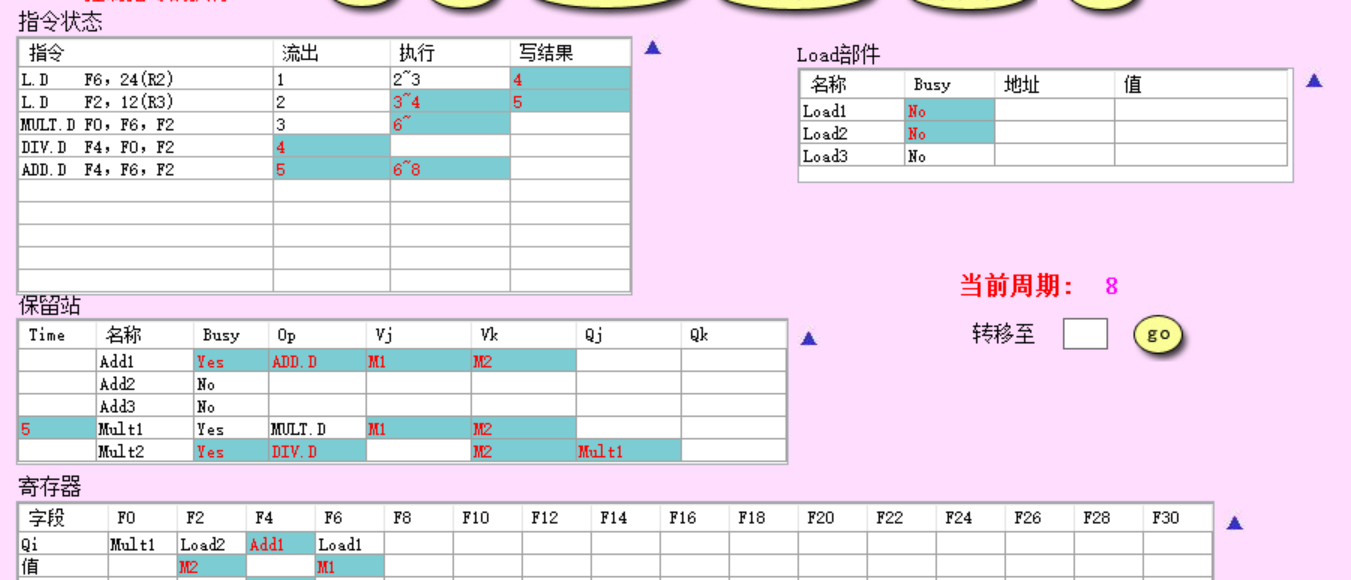
DIV.D F4, F0, F2

ADD.D F4, F6, F2

①给出在第3个时钟周期时保留站的内容。



②步进5个时钟周期，给出这时保留站、Load缓冲器以及寄存器状态表中的内容。



③再步进10个时钟周期，给出这时保留站、Load缓冲器以及寄存器状态表中的内容。

