**作业5**

考虑如下代码，假设采用Tomasulo算法进行指令调度，当第3条指令（fmul.d）准备将结果写回时，将表1、表2、表3的状态补充完整。

|  |  |
| --- | --- |
| **1.** | **fld f6, 32(x2)** |
| **2.** | **fld f2, 44(x3)** |
| **3.** | **fmul.d f0, f2, f4** |
| **4.** | **fsub.d f8, f2, f6** |
| **5.** | **fdiv.d f0, f0, f6** |
| **6.** | **fadd.d f6, f8, f2** |

**提示：**

fld需要1个时钟周期（浮点数load指令）

fadd.d需要2个时钟周期

fmul.d需要6个时钟周期

fdiv.d需要12个时钟周期

表1、2、3各字段的含义：

: 执行的操作；

: 产生相应源操作数的保留站号。0代表源操作数已经准备好（在或中）、或不需要源操作数；

: 源操作数的数值。对于load操作，用作保存地址的偏移量；

: 用来存放load或store指令的内存地址

：标记保留站是否被占用（Yes表示占用，No表示未占用）

：哪个保留站的值将写入该寄存器。

表1（**填Yes或者No**）

|  |  |  |  |
| --- | --- | --- | --- |
|  | 指令状态 | | |
| 指令 | 发射 | 执行 | 写回 |
| fld f6, 32(x2) | No | No | No |
| fld f2, 44(x3) | No | No | No |
| fmul.d f0, f2, f4 | No | YES | No |
| fsub.d f8, f2, f6 | No | No | No |
| fdiv.d f0, f0, f6 | No | YES | No |
| fadd.d f6, f8, f2 | No | No | YES |

ps:这里所填的yes/no代表的是这一时刻，这个指令的状态，如果已经执行过了，这里依旧写的是no，此时第1，2，4指令已经写回结束

表2

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 保留站状态 | | | | | | |
| 保留站号 |  |  |  |  |  |  |  |
| Load1 | NO |  |  |  |  |  |  |
| Load2 | NO |  |  |  |  |  |  |
| Add1 | NO |  |  |  |  |  |  |
| Add2 | NO |  |  |  |  |  |  |
| Add3 |  |  |  |  |  |  |  |
| Multi1 | YES | FMUL.D | mem[x3+44] | R(F4) |  |  |  |
| Multi2 | YES | FDIV.D |  | mem[x2+32] | Multi1 |  |  |

表3

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 寄存器状态 | | | | | | | | |
|  | f0 | f2 | f4 | f6 | f8 | f10 | f12 | ... | f30 |
|  | Multi1 | mem[x3+44] |  | 2\*mem[x3+44]-mem[x2+32] | mem[x3+44]-mem[x2+32] |  |  |  |  |