Computer Architecture Lab6

PB19071501 李平治

实验目的

- 熟悉Tomasulo模拟器和cache一致性模拟器（监听法和目录法）的使用
- 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
- 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。
- 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
- 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

实验环境

- VMWare12.2.3容器: WIndows10

- Tomasulo模拟器
- 多Cache一致性算法模拟器

实验内容

I. Tomasulo算法模拟器

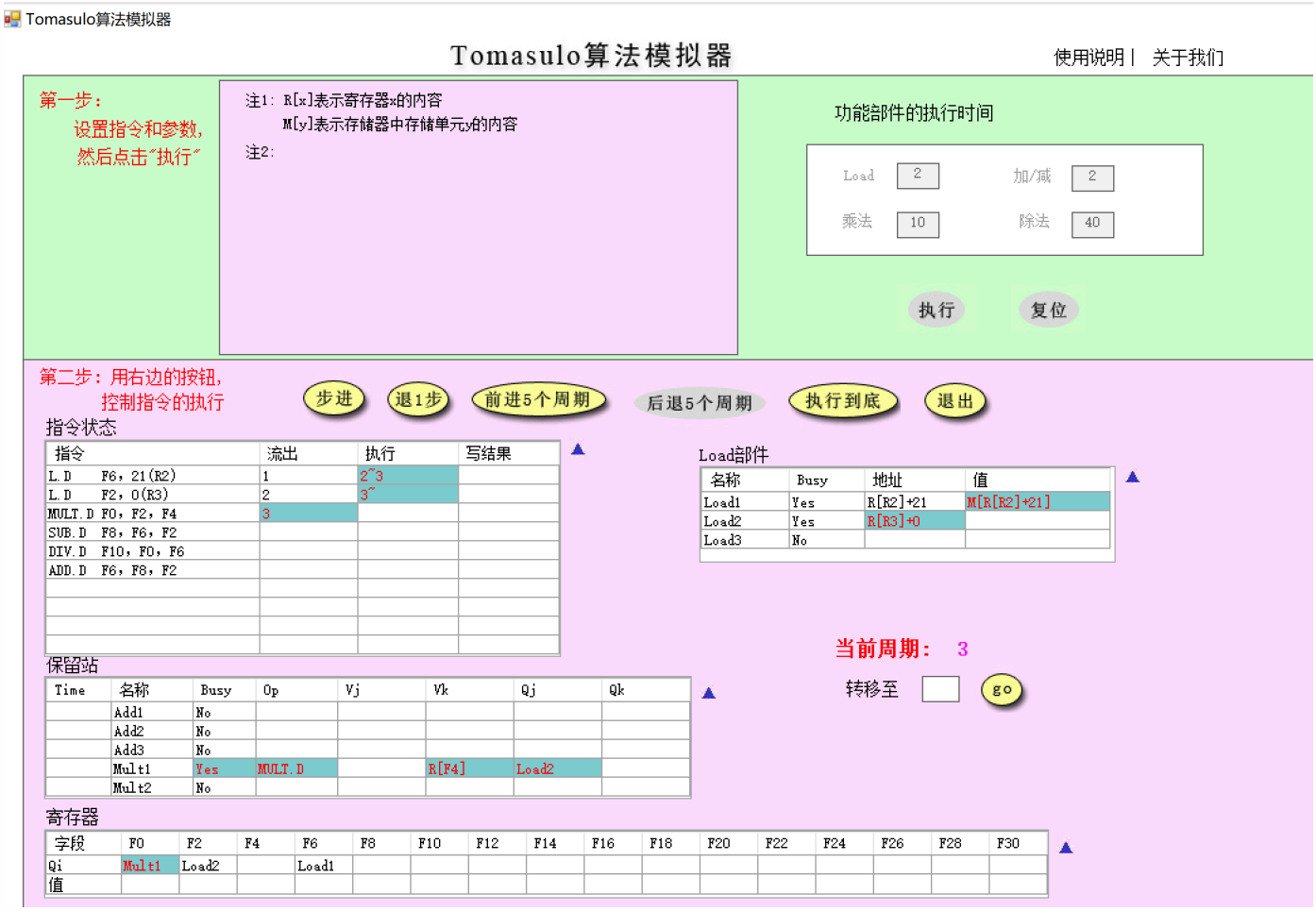
1. 分别截图（当前周期2和当前周期3），请简要说明load部件做了什么改动

- 当前周期2

Load1部件被第一条L.D指令占用并进入Busy状态，其中存入了指令计算出的访存地址R[R2]+21

Load2部件被第二条L.D指令占用并进入Busy状态

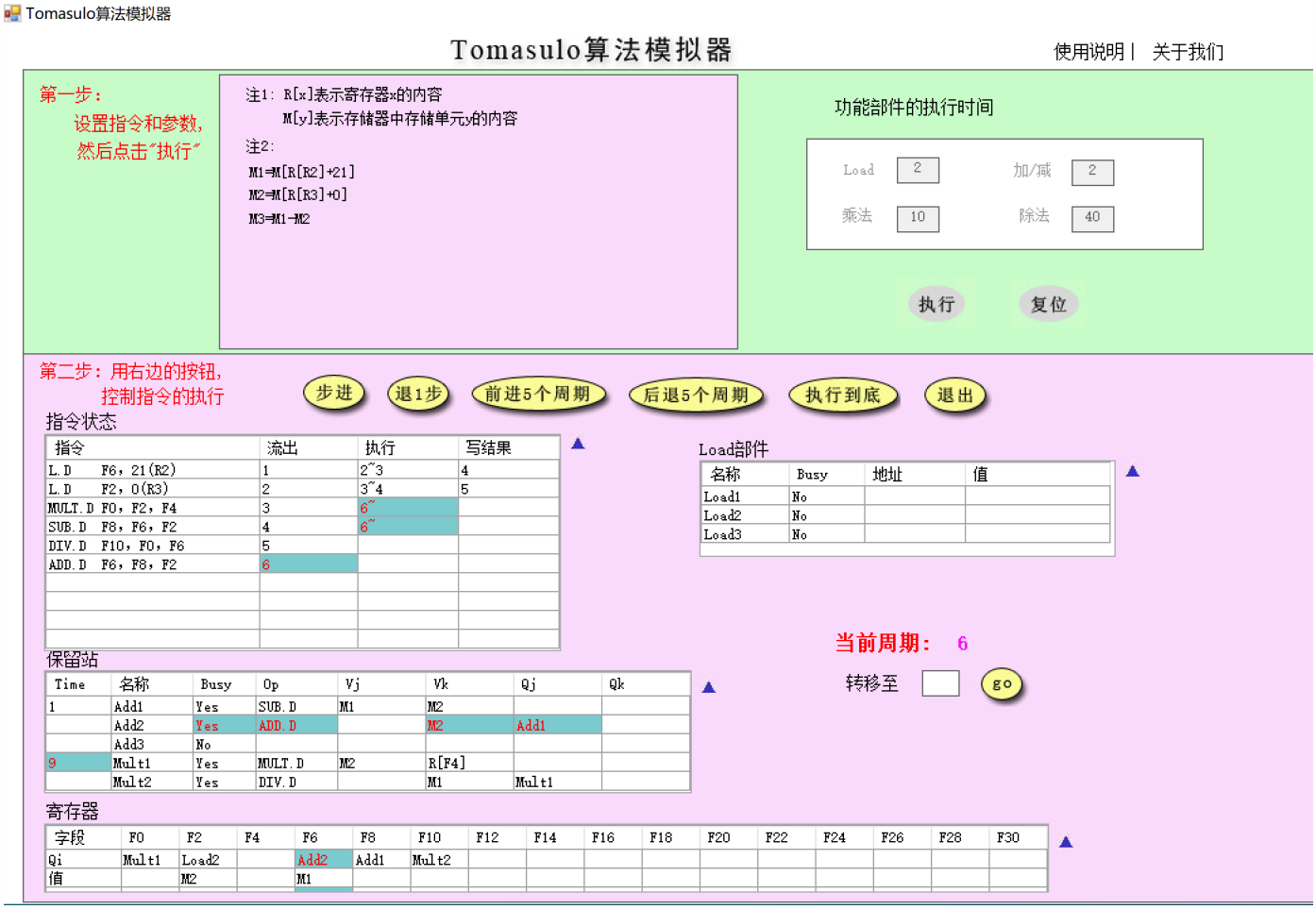
- 当前周期3



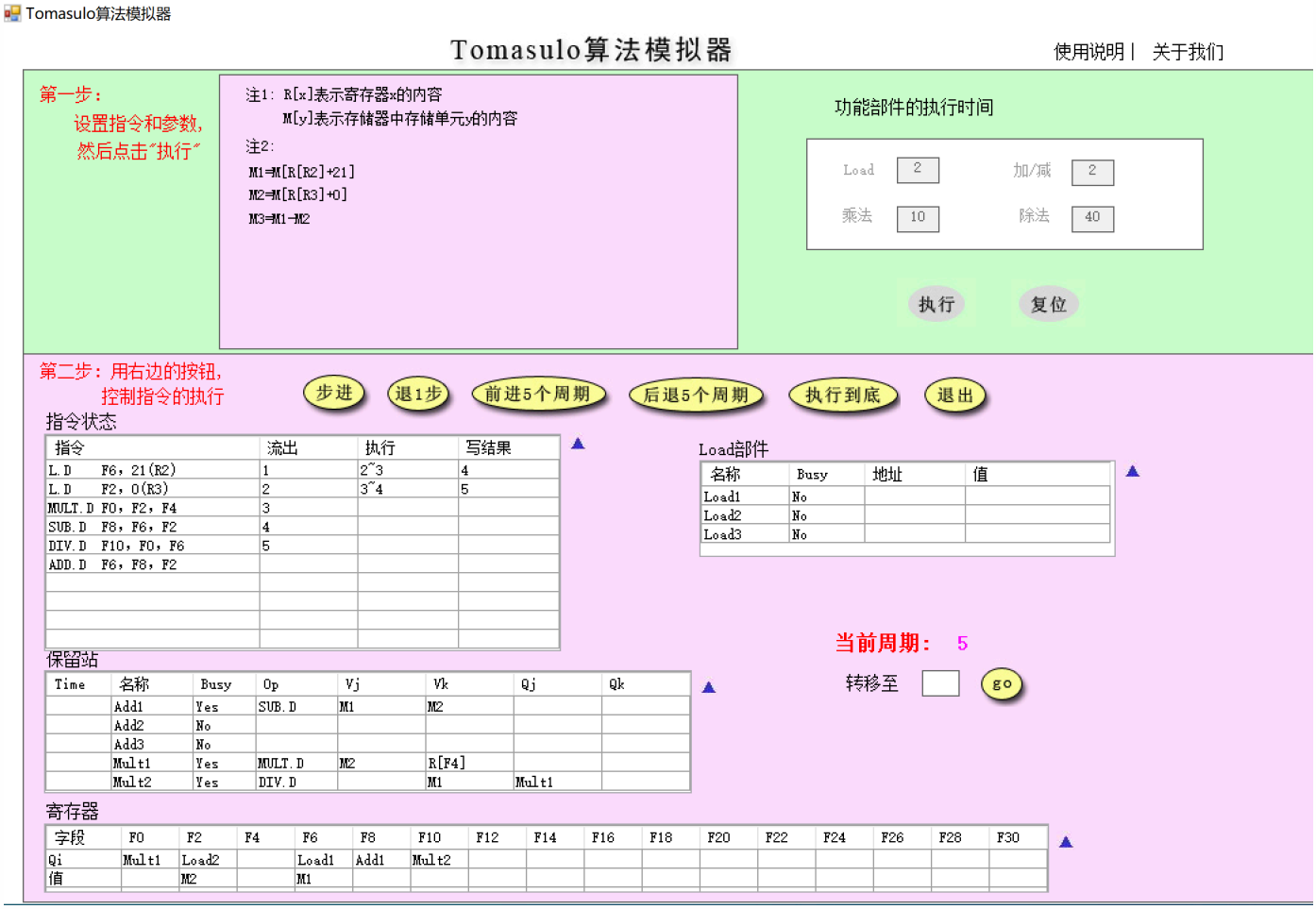
Load1部件将从储存器读取的值存入部件中

Load2部件中存入了第二条L.D指令计算出的访存地址R[R3]+0

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）



该周期的上一周期状态如下：



系统发生的变化：

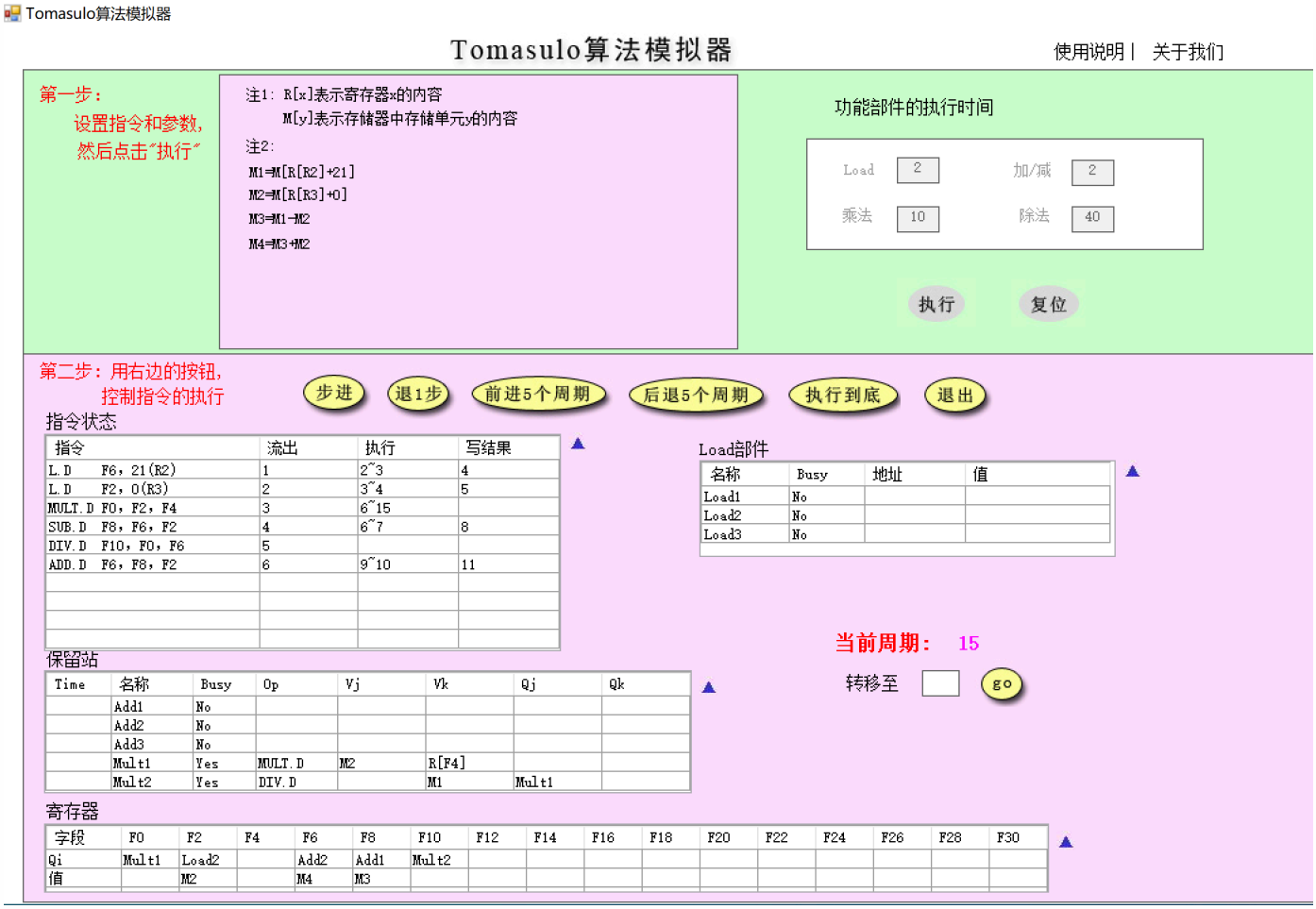
- 指令状态：
  - 第六条指令流出，第三条、第四条指令进入执行状态
- 保留站：
  - 新发射的第六条指令ADD.D占用Add2保留站
  - 进入执行的第三条指令MULT.D和第四条指令SUB.D开始执行，并开始倒计时
- 寄存器：
  - F6寄存器开始等待新发射的第六条指令ADD.D的结果

3. 简要说明是什么相关导致MUL.D流出后没有立即执行

操作数F2的RAW相关，MUL.D需要等待第二条指令L.D的写结果（F2寄存器）

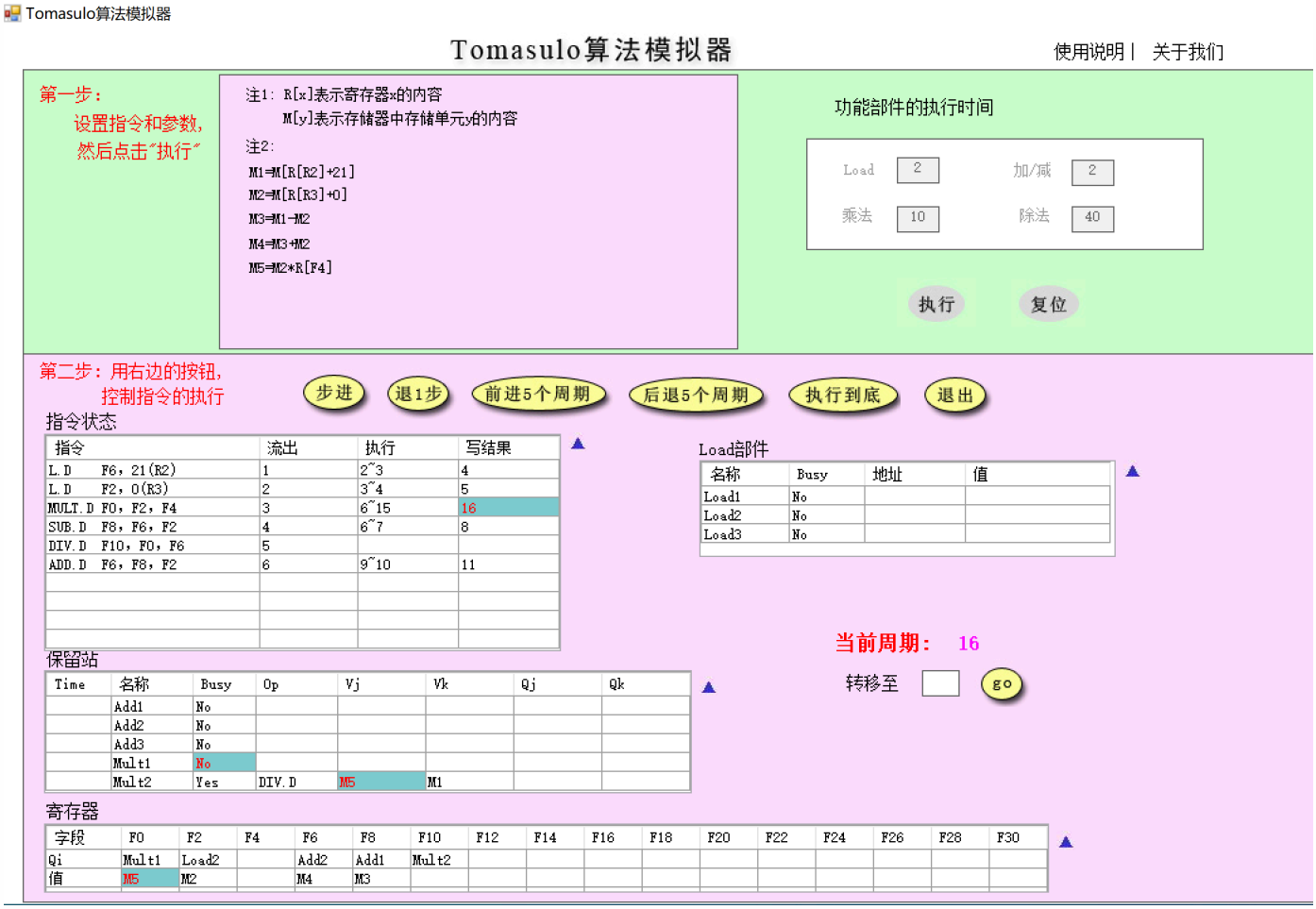
4. 请分别截图（15周期和16周期的系统状态），并分析系统发生了哪些变化

- 15周期



MULT.D指令执行结束，即将写回。

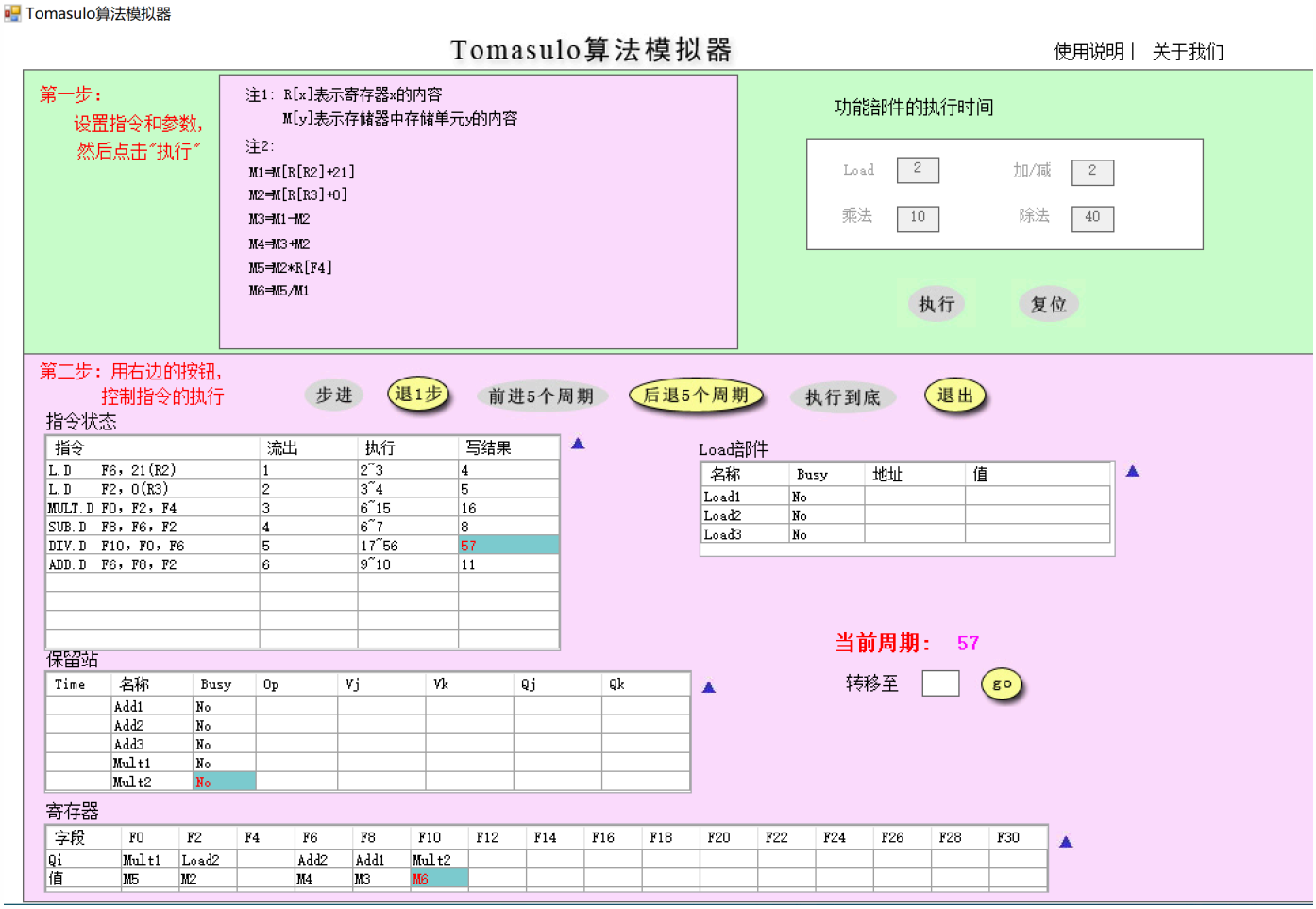
- 16周期



MULT.D指令开始写入结果，保留站中Mult1部件释放，结果写入F0寄存器；保留站中Mult2部件从F0寄存器中读取第一个操作数M5。

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）

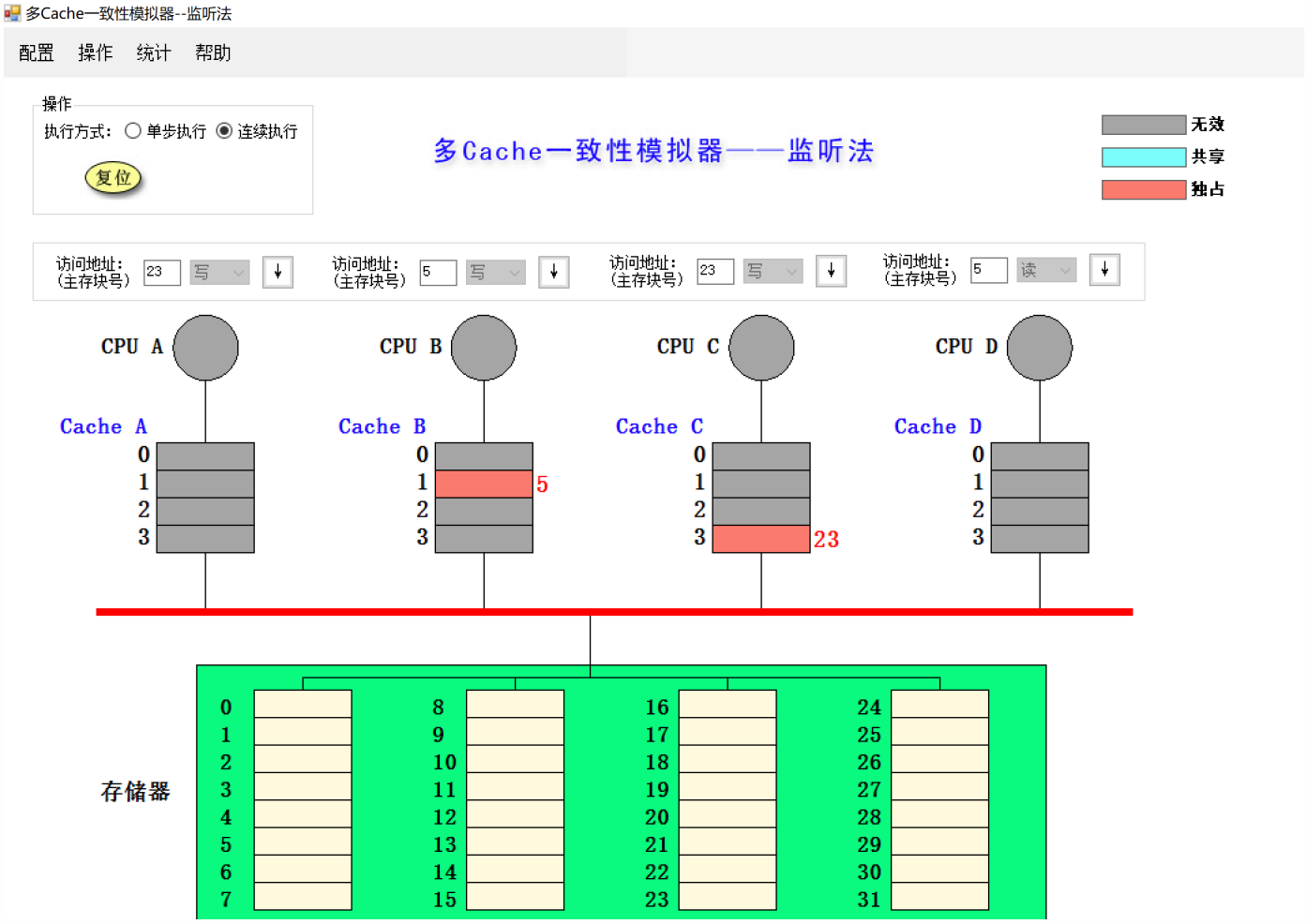
第57周期



II. 多Cache一致性算法-监听法

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	替换Cache A的块1	否	Cache A发射Read Miss，存储器传输第5块到Cache B， Cache A的块1状态变为共享
CPU B读第5块	替换Cache B的块1	否	Cache B发射Read Miss，存储器传输第5块到Cache B， Cache B的块1状态变为共享
CPU C读第5块	替换Cache C的块1	否	Cache C发射Read Miss，存储器传输第5块到Cache C， Cache C的块1状态变为共享
CPU B写第5块	否	否	Cache B发射Invalidate，Cache A的块1状态变为无效，Cache B的块1状态变为独占，Cache C的块1状态变为无效
CPU D读第5块	替换Cache D的块1	写回Cache B的块3	Cache D发射Read Miss，Cache B写回第5块，存储器传输第5块到Cache D，Cache B的块1状态变为共享，Cache D的块1状态变为共享
CPU B写第21块	替换Cache B的块1	否	Cache B发射Write Miss，存储器传输第21块到Cache B，Cache B的块1状态变为独占
CPU A写第23块	替换Cache A的块3	否	Cache A发射Write Miss，存储器传输第23块到Cache A，Cache A的块3状态变为独占
CPU C写第23块	替换Cache C的块3	写回Cache A的块3	Cache C发射Write Miss，Cache A写回第23块，存储器传输第23块到Cache C，Cache A的块3状态变为无效，Cache C的块3状态变为独占
CPU B读第29块	替换Cache B的块1	写回Cache B的块1	Cache B写回第21块，Cache B发射Read Miss，存储器传输第29块到Cache B，Cache B的块1状态变为共享
CPU B写第5块	替换Cache B的块1	否	Cache B发送Write Miss，存储器传输第5块到Cache B，Cache B的块1状态变为独占，Cache D的状态变为无效

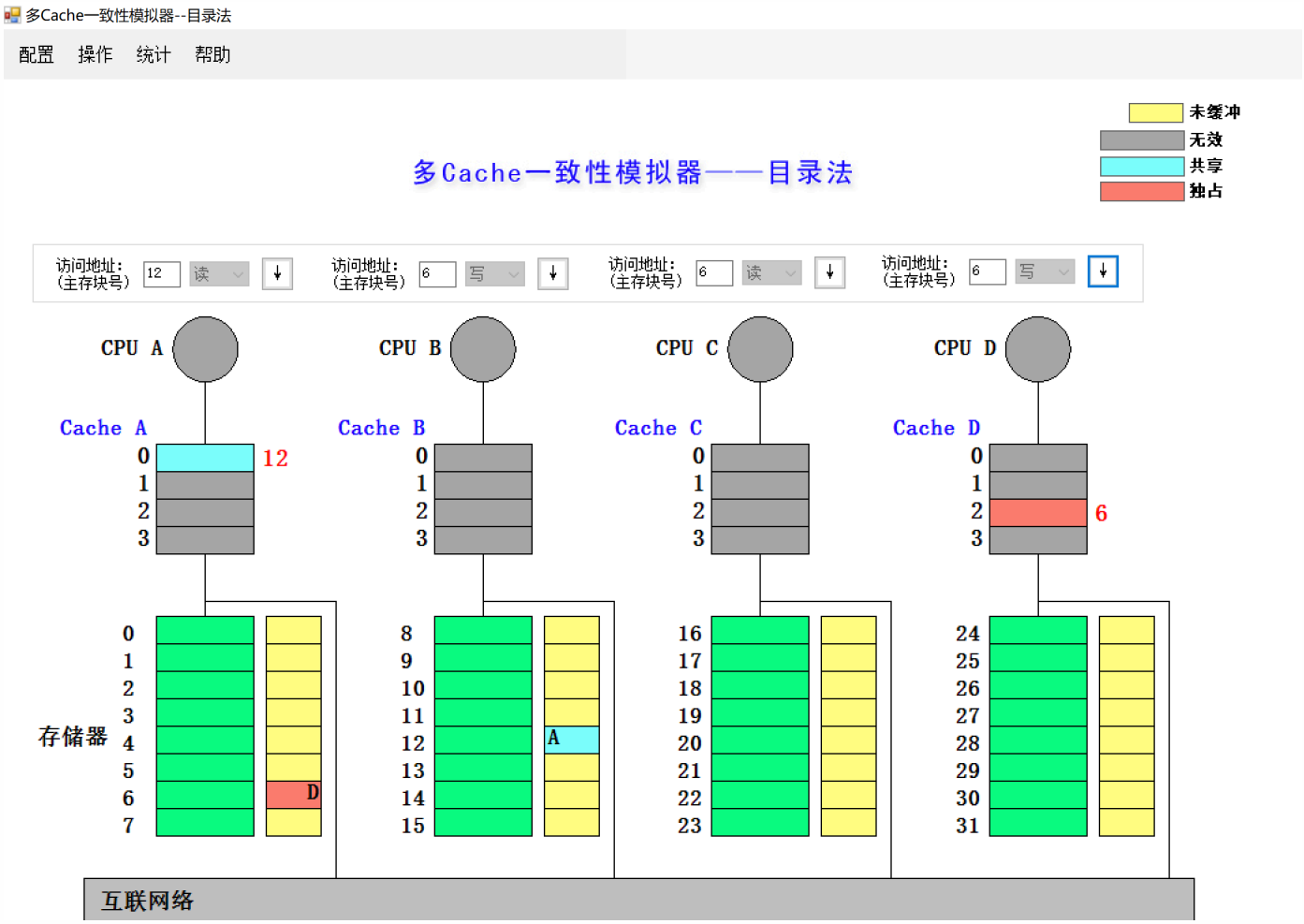
执行完毕系统状态：



III. 多Cache一致性算法-目录法

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	Cache A发送Read Miss(A, 6)到存储器，存储器传输第6块到Cache A，Cache A的块2状态变为共享，存储器目录的块6状态变为共享，共享集合{A}
CPU B读第6块	Cache B发送Read Miss(B, 6)到存储器，存储器传输第6块到Cache B，Cache B的块2状态变为共享，共享集合{A, B}
CPU D读第6块	Cache D发送Read Miss(D, 6)到存储器，存储器传输第6块到Cache D，Cache D的块2状态变为共享，共享集合{A, B, D}
CPU B写第6块	Cache B发送Write Hit(B, 6)到存储器，存储器发送Invalidate(6)到Cache A和Cache D，Cache A和Cache D的块2状态变为无效，Cache B的块2状态变为独占，共享集合{B}
CPU C读第6块	Cache C发送Read Miss(C, 6)到存储器，存储器发送Fetch(6)到Cache B，Cache B传输第6块到存储器，Cache B的块2状态变为共享，存储器传输第6块到Cache C，Cache C的块2状态变为共享
CPU D写第20块	Cache D发送Write Miss(D, 20)到存储器，存储器传输第20块到Cache D，Cache D的块0状态变为独占，存储器目录的块20状态变为独占，共享集合{D}
CPU A写第20块	Cache A发送Write Miss(A, 20)到存储器，存储器发送Fetch和Invalidate(20)到Cache D，Cache D写回第20块，Cache D的块0状态变为无效，存储器传输第20块到Cache A，Cache A的块0状态变为独占，共享集合{A}
CPU D写第6块	Cache D发送Write Miss(D, 6)到存储器，存储器发送Invalidate(6)到Cache B和Cache C，Cache B和Cache C的块2状态变为无效，存储器传输第6块到Cache D，Cache D块2状态变为独占，存储器目录块6状态变为独占，共享集合{D}
CPU A读第12块	Cache A发送Write Back(A, 20)到存储器，Cache A块0状态变为无效，Cache A发送Read Miss(A, 12)到存储器，存储器传输第12块到Cache A，Cache A的块0状态变为共享，存储器目录的块12状态变为共享，共享集合{A}

执行完毕系统状态：



IV. Q&A

1. 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?

- 监听法实现简单，但可扩展性差，存在总线竞争和带宽限制问题，通信开销大
- 目录法可扩展性强，降低了对总线的占用，但实现比较复杂，有额外的存储开销，且存储器通信压力使得存储器接口通信速度成为瓶颈

2. 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）

- Tomasulo和Score Board都解决了结构相关和数据相关(RAW, WAR, WAW)
  - Tomasulo和Score Board都使用结构相关不发射来避免结构相关，都是用动态调度来避免RAW
  - Tomasulo使用寄存器重命名解决WAR和WAW，Score Board使用停顿来解决WAW和WAR
- Tomasulo是分布式，Score Board是集中式

3. Tomasulo算法是如何解决结构、RAW、WAR和WAW相关的?

- 结构相关：使用保留站，当存在结构相关时，不发射
- RAW相关：在保留站中记录需要的操作数，当且仅当操作数就绪时，才读取操作数进入执行状态
- WAR相关：通过在保留站中重命名寄存器来解决
- WAW相关：通过在保留站中重命名寄存器来解决

实验总结

- 本次试验通过使用模拟器，对Tomasulo和多Cache一致性算法有了更深刻的理解。
- 本次试验用时
  - Tomasulo：0.5h
  - 多Cache一致性算法：0.5h
  - 实验报告：2h