



**華東理工大學**  
EAST CHINA UNIVERSITY OF SCIENCE AND TECHNOLOGY

# 《 计算机体系结构 》

## 实验报告三

班 级： 计 203

学 号： 20002462

姓 名： 刘子言

指导教师： 梁建宁

信息科学与工程学院

2022 年 11 月

# 实验名称 重排序缓冲 ROB 工作原理

实验地点 信息楼 418 实验日期 2022.11.30

## 一、实验目的

- 1、加深对指令级并行性及其开发的理解；
- 2、加深对基于硬件的猜测的理解；
- 3、掌握 ROB 在指令流出、执行、写回、提交 4 个阶段所进行的操作；
- 4、掌握 ROB 结构特点；
- 5、给定执行代码片段，能写出某个时钟周期，保留站、ROB、寄存器状态表的变化情况。

## 二、实验设备

个人 PC 机，实验平台采用重排序缓冲 ROB 模拟器。

## 三、实验原理

### 1、利用猜测挖掘更多 ILP

(1) 为了挖掘更多指令并行性，需要通过硬件猜测分支转移的方法克服控制相关，假设猜测是对的就继续执行指令。

(2) 基于硬件的猜测的三个主要思想：

- 通过动态分支预测来选择要执行的指令；
- 猜测可以允许在控制相关解决前执行指令（能够撤销预测错误指令的影响）；
- 动态调度基准块的不同组合。

(3) 要获得更多的 ILP，需要将猜测与动态调度结合起来。没有猜测的动态调度仅仅能够部分重叠执行基本块，因为它要求在执行后续基本块的任意一条指令之前，分支指令必须被判明（转移成功与否）。

(4) 硬件猜测+Tomasulo 算法的思想：

沿着预测的路径（分支转移方向）执行指令，但是只有预测正确时才确认提交指令结果。

可以将指令执行分成指令执行完成和指令提交两部分。

指令提交就是当指令不再不确定时（分支猜测正确），允许指令更新寄存器堆和存储器。

为了让猜测机制发挥效用，可以允许指令乱序执行，但是要强制循序提交。需要一些额外的硬件（ROB，重排序缓冲区）来阻止任何不可撤回的动作，直到指令确认提交。

### 2、重排序缓冲区

(1) ROB：当指令执行完毕尚未提交之前，执行结果存放于这里。

(2) ROB 包含四项内容：

- 指令种类：分支指令/Load/Store/ALU 操作；
- 目的寄存器、目的内存地址：寄存器编号、存储器编号；
- 输出值：指令提交前的执行结果存放于这里；
- 准备好：指令完成执行，数值已经准备好。

(3) 引入重排序缓冲区后，保留栈也需要进行一些修改：操作数的来源改为 ROB，不再是功能部件。ROB 类似于 Tomasulo 原形算法中的 RS，也起到扩展寄存器的作用。

- (4) 直到指令提交时，再将结果写入寄存器和存储器。
- (5) 对于预测错误：ROB 里的相应内容被清除。
- (6) 对于异常：不予确认直到它对应的指令准备提交。如果一条猜测状态的指令造成一个异常，将异常记录到 ROB；如果这条指令抵达 ROB 的头部，该指令不再是猜测状态，对应的异常准备执行。
- (7) 基本操作：
  - 按照 FIFO 先进先出规则，将指令放置在重排序缓冲区；
  - 当指令执行完毕，将结果存放在 ROB 中，并将结果提供其他需要的指令，跟保留栈起到类似作用。不过需要用 ROB 编号代替 RS 保留栈编号；
  - 指令提交：位于 ROB 头部的数据被写入寄存器或者存储器单元；
  - 利用 ROB，在猜测错误时可以很容易撤销已执行指令。

### 3、含硬件猜测 ROB 算法步骤

#### (1) 发射（从指令队列取指令）

如果保留栈和重新排序缓冲区有空闲，发射指令，送操作数以及暂存输出值的 ROB 编号。

#### (2) 执行（执行指令）

如果操作数准备好就可以开始执行指令；如果操作数没有准备好，监听 CDB 公共数据通道等待操作数。这一步需要检查 RAW 冒险。

#### (3) 写回

将结果通过 CDB 公共数据通道旁路传递给需要的功能部件和 ROB。

#### (4) 指令提交

用 ROB 里的输出结果更新寄存器和内存单元。

当指令位于 ROB 头部（预测正确，消除不确定性），更新寄存器和内存单元，并将指令从 ROB 移走。如果预测错误，ROB 被刷新，执行过程从该分支的后续正常指令处重新开始。

## 四、实验操作及运行结果

1、假设浮点功能部件的延迟时间（处理时间）为：加法 2 个时钟周期，乘法 10 个时钟周期，除法 40 个时间周期，Load 部件 2 个时钟周期。

①有以下代码段，给出当指令 MUL.D 即将确认时，保留站、ROB 和浮点寄存器状态表的内容。

```
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
```

在第 15 个时钟周期时，指令 MUL.D 即将确认（该指令在第 16 个时钟周期时确认），其保留站、ROB 和浮点寄存器状态表的内容如下图所示：

指令状态				
指令	流出	执行	与结果	确认
L D F6, 24(R2)	1	2'3	4	5
L D F2, 12(R3)	2	3'4	5	6
MULT.D F0, F2, F4	3	5'14	16	
SUB.D F8, F6, F2	4	5'6	7	
DIV.D F10, F0, F6	5	15'		
ADD.D F6, F8, F2	6	7'8	9	

  

再定序缓冲器(ROB)				
标记	项号	Busy	指令	目的地
#1	No			
#2	No			
HEAD #3	Yes	MULT.D F0, F2, F4	F0	M2+R[F4]
#4	Yes	SUB.D F8, F6, F2	F8	M1-M2
#5	Yes	DIV.D F10, F0, F6	F10	
#6	Yes	ADD.D F6, F8, F2	F6	R[F8]+M2
TAIL #7	No			
#8	No			
#9	No			
#10	No			

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保留站、ROB 和浮点寄存器状态的内容变化情况。（针对步进过程中的一些比较重要的周期，在截图下方都附有文字解释）

[illegible][illegible]

第二步：用右边的按钮，控制指令的执行

步进 退1步 前进5个周期 后退5个周期 执行到底 退出 当前周期：3 转移至  go

指令状态

指令	流出	执行	写结果	确认
L.D F6, 24(R2)	1	2~3		
L.D F2, 12(R3)	2	3~		
MULT.D F0, F2, F4	3			
SUB.D F8, F6, F2				
DIV.D F10, F0, F6				
ADD.D F6, F8, F2				

保留站

Time	名称	Busy	Op	Vj	Vk	Qj	Qk	目的地
	Add1	No						
	Add2	No						
	Add3	No						
	Mult1	Yes	MULT.D		R[F4]	#2		#3
	Mult2	No						

寄存器

字段	F0	F2	F4	F6	F8	F10	F12	F14	F16	F18	F20	F22	F24	F26	F28	F30
ROB项号	#3	#2		#1												
Busy	Yes	Yes	No	Yes	No	No	No	No	No	No	No	No	No	No	No	No
值																

再定序缓冲器 (ROB)

标记	项号	Busy	指令	目的地	值
HEAD	#1	Yes	L.D F6, 24(R2)	F6	
	#2	Yes	L.D F2, 12(R3)	F2	
	#3	Yes	MULT.D F0, F2, F4	F0	
TAIL	#4	No			
	#5	No			
	#6	No			
	#7	No			
	#8	No			
	#9	No			
	#10	No			

Load缓冲器

名称	Busy	地址	目的地	值
Load1	Yes	24+R[R2]	#1	M[24+R[R2]]
Load2	Yes	12+R[R3]	#2	
Load3	No			

第二步：用右边的按钮，控制指令的执行

步进 退1步 前进5个周期 后退5个周期 执行到底 退出 当前周期：4 转移至  go

指令状态

指令	流出	执行	写结果	确认
L.D F6, 24(R2)	1	2~3	4	
L.D F2, 12(R3)	2	3~4		
MULT.D F0, F2, F4	3			
SUB.D F8, F6, F2	4			
DIV.D F10, F0, F6				
ADD.D F6, F8, F2				

保留站

Time	名称	Busy	Op	Vj	Vk	Qj	Qk	目的地
	Add1	Yes	SUB.D	M[24+R[R2]]			#2	#4
	Add2	No						
	Add3	No						
	Mult1	No	MULT.D		R[F4]	#2		#3
	Mult2	No						

寄存器

字段	F0	F2	F4	F6	F8	F10	F12	F14	F16	F18	F20	F22	F24	F26	F28	F30
ROB项号	#3	#2		#1	#4											
Busy	Yes	Yes	No	Yes	Yes	No	No	No	No	No	No	No	No	No	No	No
值																

再定序缓冲器 (ROB)

标记	项号	Busy	指令	目的地	值
HEAD	#1	Yes	L.D F6, 24(R2)	F6	M[24+R[R2]]
	#2	Yes	L.D F2, 12(R3)	F2	
	#3	Yes	MULT.D F0, F2, F4	F0	
	#4	Yes	SUB.D F8, F6, F2	F8	
TAIL	#5	No			
	#6	No			
	#7	No			
	#8	No			
	#9	No			
	#10	No			

Load缓冲器

名称	Busy	地址	目的地	值
Load1	Yes	12+R[R3]	#2	M[12+R[R3]]
Load2	No			
Load3	No			

第二步：用右边的按钮，控制指令的执行

步进 退1步 前进5个周期 后退5个周期 执行到底 退出 当前周期：5 转移至  go

指令状态

指令	流出	执行	写结果	确认
L.D F6, 24(R2)	1	2~3	4	5
L.D F2, 12(R3)	2	3~4	5	
MULT.D F0, F2, F4	3	5~		
SUB.D F8, F6, F2	4	5~		
DIV.D F10, F0, F6	5			
ADD.D F6, F8, F2				

保留站

Time	名称	Busy	Op	Vj	Vk	Qj	Qk	目的地
2	Add1	Yes	SUB.D	M1	M[12+R[R3]]			#4
	Add2	No						
	Add3	No						
10	Mult1	Yes	MULT.D	M[12+R[R3]]	R[F4]			#3
	Mult2	Yes	DIV.D		M1	#3		#5

寄存器

字段	F0	F2	F4	F6	F8	F10	F12	F14	F16	F18	F20	F22	F24	F26	F28	F30
ROB项号	#3	#2			#4	#5										
Busy	Yes	Yes	No	No	Yes	Yes	No	No	No	No	No	No	No	No	No	No
值				M1												

再定序缓冲器 (ROB)

标记	项号	Busy	指令	目的地	值
HEAD	#1	No			
	#2	Yes	L.D F2, 12(R3)	F2	M[12+R[R3]]
	#3	Yes	MULT.D F0, F2, F4	F0	
	#4	Yes	SUB.D F8, F6, F2	F8	
	#5	Yes	DIV.D F10, F0, F6	F10	
TAIL	#6	No			
	#7	No			
	#8	No			
	#9	No			
	#10	No			

Load缓冲器

名称	Busy	地址	目的地	值
Load1	No			
Load2	No			
Load3	No			

前 5 个周期中，第 1 条 load 指令完成了流出-执行-写结果-确认的过程，第 2 条 load 指令即将确认，MULT 与 SUB 指令均在执行阶段。寄存器 F6、F2、F0、F8、F10 先后作为各指令在 ROB 中的目的地，项号为 #1、#2、#3、#4、#5，这是 ROB 与动态调度中 Tomasulo 算法不同的地方。ROB 中根据项号缓存了前 5 条指令的目的地与值，由于第一条 load 指令已经完确认，所以更新寄存器 F6 与存储器中的值后，从 ROB 中移走。保留栈中也暂存了



相关指令的操作数信息。虽然第 2 条 Load 指令与 MULT 指令和 SUB 指令之间存在数据相关（寄存器 F2），但是到第 5 周期该 load 指令已经完成了写结果，F2 值已旁路转发到 ROB 和相应功能部件中，MULT 与 SUB 指令可从中获得源操作数，进入执行阶段。

第二步：用右边的按钮，控制指令的执行

当前周期：6

再定序缓冲器 (ROB)

标记	项号	Busy	指令	目的地	值
	#1	No			
	#2	No			
HEAD	#3	Yes	MULT.D F0, F2, F4	F0	
	#4	Yes	SUB.D F8, F6, F2	F8	
	#5	Yes	DIV.D F10, F0, F6	F10	
	#6	Yes	ADD.D F6, F8, F2	F6	
TAIL	#7	No			
	#8	No			
	#9	No			
	#10	No			

指令状态

指令	流出	执行	写结果	确认
L.D F6, 24(R2)	1	2~3	4	5
L.D F2, 12(R3)	2	3~4	5	6
MULT.D F0, F2, F4	3	5~		
SUB.D F8, F6, F2	4	5~6		
DIV.D F10, F0, F6	5			
ADD.D F6, F8, F2	6			

保留站

Time	名称	Busy	Op	Vj	Vk	Qj	Qk	目的地
1	Add1	Yes	SUB.D	M1	M2			#4
	Add2	Yes	ADD.D		M2	#4		#6
	Add3	No						
9	Mult1	Yes	MULT.D	M2	R[F4]			#3
	Mult2	Yes	DIV.D		M1	#3		#5

寄存器

字段	F0	F2	F4	F6	F8	F10	F12	F14	F16	F18	F20	F22	F24	F26	F28	F30
ROB项号	#3			#6	#4	#5										
Busy	Yes	No	No	Yes	Yes	Yes	No	No	No	No	No	No	No	No	No	No
值		M2		M1												

Load缓冲器

名称	Busy	地址	目的地	值
Load1	No			
Load2	No			
Load3	No			

第 6 周期结束，第 2 条 load 指令完成确认，更新寄存器 F2 与存储器的值，然后从 ROB 中移走。F2 更新为 M2 值，保留站中的 F2 内容也相应更新。

第二步：用右边的按钮，控制指令的执行

当前周期：7

再定序缓冲器 (ROB)

标记	项号	Busy	指令	目的地	值
	#1	No			
	#2	No			
HEAD	#3	Yes	MULT.D F0, F2, F4	F0	
	#4	Yes	SUB.D F8, F6, F2	F8	M1-M2
	#5	Yes	DIV.D F10, F0, F6	F10	
	#6	Yes	ADD.D F6, F8, F2	F6	
TAIL	#7	No			
	#8	No			
	#9	No			
	#10	No			

指令状态

指令	流出	执行	写结果	确认
L.D F6, 24(R2)	1	2~3	4	5
L.D F2, 12(R3)	2	3~4	5	6
MULT.D F0, F2, F4	3	5~		
SUB.D F8, F6, F2	4	5~6	7	
DIV.D F10, F0, F6	5			
ADD.D F6, F8, F2	6	7~		

保留站

Time	名称	Busy	Op	Vj	Vk	Qj	Qk	目的地
2	Add1	No						
	Add2	Yes	ADD.D	M1-M2	M2			#6
	Add3	No						
8	Mult1	Yes	MULT.D	M2	R[F4]			#3
	Mult2	Yes	DIV.D		M1	#3		#5

寄存器

字段	F0	F2	F4	F6	F8	F10	F12	F14	F16	F18	F20	F22	F24	F26	F28	F30
ROB项号	#3			#6	#4	#5										
Busy	Yes	No	No	Yes	Yes	Yes	No	No	No	No	No	No	No	No	No	No
值		M2		M1												

Load缓冲器

名称	Busy	地址	目的地	值
Load1	No			
Load2	No			
Load3	No			

到第 7 周期 SUB 指令已完成写结果，F8 值已旁路转发到 ROB 和相应功能部件中，ADD 指令可从中获得源操作数 F8 的值，进入执行阶段。

第二步：用右边的按钮，控制指令的执行

当前周期：8

再定序缓冲器 (ROB)

标记	项号	Busy	指令	目的地	值
	#1	No			
	#2	No			
HEAD	#3	Yes	MULT.D F0, F2, F4	F0	
	#4	Yes	SUB.D F8, F6, F2	F8	M1-M2
	#5	Yes	DIV.D F10, F0, F6	F10	
	#6	Yes	ADD.D F6, F8, F2	F6	
TAIL	#7	No			
	#8	No			
	#9	No			
	#10	No			

指令状态

指令	流出	执行	写结果	确认
L.D F6, 24(R2)	1	2~3	4	5
L.D F2, 12(R3)	2	3~4	5	6
MULT.D F0, F2, F4	3	5~		
SUB.D F8, F6, F2	4	5~6	7	
DIV.D F10, F0, F6	5			
ADD.D F6, F8, F2	6	7~8		

保留站

Time	名称	Busy	Op	Vj	Vk	Qj	Qk	目的地
1	Add1	No						
	Add2	Yes	ADD.D	M1-M2	M2			#6
	Add3	No						
7	Mult1	Yes	MULT.D	M2	R[F4]			#3
	Mult2	Yes	DIV.D		M1	#3		#5

寄存器

字段	F0	F2	F4	F6	F8	F10	F12	F14	F16	F18	F20	F22	F24	F26	F28	F30
ROB项号	#3			#6	#4	#5										
Busy	Yes	No	No	Yes	Yes	Yes	No	No	No	No	No	No	No	No	No	No
值		M2		M1												

Load缓冲器

名称	Busy	地址	目的地	值
Load1	No			
Load2	No			
Load3	No			

第二步：用右边的按钮，控制指令的执行

当前周期： 9

指令状态

指令	流出	执行	写结果	确认
L.D F6, 24(R2)	1	2~3	4	5
L.D F2, 12(R3)	2	3~4	5	6
MULT.D F0, F2, F4	3	5~		
SUB.D F8, F6, F2	4	5~6	7	
DIV.D F10, F0, F6	5			
ADD.D F6, F8, F2	6	7~8	9	

保留站

Time	名称	Busy	Op	Vj	Vk	Qj	Qk	目的地
	Add1	No						
	Add2	No						
	Add3	No						
6	Mult1	Yes	MULT.D	M2	R[F4]			#3
	Mult2	Yes	DIV.D		M1	#3		#5

寄存器

字段	F0	F2	F4	F6	F8	F10	F12	F14	F16	F18	F20	F22	F24	F26	F28	F30
ROB项号	#3			#6	#4	#5										
Busy	Yes	No	No	Yes	Yes	No	No	No	No	No	No	No	No	No	No	No
值	M2			M1												

再定序缓冲器 (ROB)

标记	项号	Busy	指令	目的地	值
	#1	No			
	#2	No			
HEAD	#3	Yes	MULT.D F0, F2, F4	F0	
	#4	Yes	SUB.D F8, F6, F2	F8	M1-M2
	#5	Yes	DIV.D F10, F0, F6	F10	
	#6	Yes	ADD.D F6, F8, F2	F6	R[F8]+M2
TAIL	#7	No			
	#8	No			
	#9	No			
	#10	No			

Load缓冲器

名称	Busy	地址	目的地	值
Load1	No			
Load2	No			
Load3	No			

此处 10-14 周期的截图省略。第 5-14 周期是 MULT 的执行阶段。

第二步：用右边的按钮，控制指令的执行

当前周期： 15

指令状态

指令	流出	执行	写结果	确认
L.D F6, 24(R2)	1	2~3	4	5
L.D F2, 12(R3)	2	3~4	5	6
MULT.D F0, F2, F4	3	5~14	15	
SUB.D F8, F6, F2	4	5~6	7	
DIV.D F10, F0, F6	5	15~		
ADD.D F6, F8, F2	6	7~8	9	

保留站

Time	名称	Busy	Op	Vj	Vk	Qj	Qk	目的地
	Add1	No						
	Add2	No						
	Add3	No						
40	Mult1	No						
	Mult2	Yes	DIV.D	M2+R[F4]	M1			#5

寄存器

字段	F0	F2	F4	F6	F8	F10	F12	F14	F16	F18	F20	F22	F24	F26	F28	F30
ROB项号	#3			#6	#4	#5										
Busy	Yes	No	No	Yes	Yes	No	No	No	No	No	No	No	No	No	No	No
值	M2			M1												

再定序缓冲器 (ROB)

标记	项号	Busy	指令	目的地	值
	#1	No			
	#2	No			
HEAD	#3	Yes	MULT.D F0, F2, F4	F0	M2+R[F4]
	#4	Yes	SUB.D F8, F6, F2	F8	M1-M2
	#5	Yes	DIV.D F10, F0, F6	F10	
	#6	Yes	ADD.D F6, F8, F2	F6	R[F8]+M2
TAIL	#7	No			
	#8	No			
	#9	No			
	#10	No			

Load缓冲器

名称	Busy	地址	目的地	值
Load1	No			
Load2	No			
Load3	No			

到第 15 周期，MULT 指令完成写结果，F0 值已旁路转发到 ROB 和相应功能部件中，DIV 指令可从中获得源操作数 F0 的值，进入执行阶段。

第二步：用右边的按钮，控制指令的执行

当前周期： 16

指令状态

指令	流出	执行	写结果	确认
L.D F6, 24(R2)	1	2~3	4	5
L.D F2, 12(R3)	2	3~4	5	6
MULT.D F0, F2, F4	3	5~14	15	16
SUB.D F8, F6, F2	4	5~6	7	
DIV.D F10, F0, F6	5	15~		
ADD.D F6, F8, F2	6	7~8	9	

保留站

Time	名称	Busy	Op	Vj	Vk	Qj	Qk	目的地
	Add1	No						
	Add2	No						
	Add3	No						
39	Mult1	No						
	Mult2	Yes	DIV.D	M3	M1			#5

寄存器

字段	F0	F2	F4	F6	F8	F10	F12	F14	F16	F18	F20	F22	F24	F26	F28	F30
ROB项号	#3			#6	#4	#5										
Busy	No	No	No	Yes	Yes	Yes	No	No	No	No	No	No	No	No	No	No
值	M3	M2		M1												

再定序缓冲器 (ROB)

标记	项号	Busy	指令	目的地	值
	#1	No			
	#2	No			
	#3	No			
HEAD	#4	Yes	SUB.D F8, F6, F2	F8	M1-M2
	#5	Yes	DIV.D F10, F0, F6	F10	
	#6	Yes	ADD.D F6, F8, F2	F6	R[F8]+M2
TAIL	#7	No			
	#8	No			
	#9	No			
	#10	No			

Load缓冲器

名称	Busy	地址	目的地	值
Load1	No			
Load2	No			
Load3	No			

到第 16 周期结束，MULT 指令完成确认，更新寄存器 F0 与存储器的值，然后从 ROB 中移走。F0 更新为 M3 值，保留站中的 F0 内容也相应更新。

第二步: 用右边的按钮, 控制指令的执行

步进 退1步 前进5个周期 后退5个周期 执行到底 退出 当前周期: 17 转移至  GO

指令状态

指令	流出	执行	写结果	确认
L.D F6, 24(R2)	1	2~3	4	5
L.D F2, 12(R3)	2	3~4	5	6
MULT.D F0, F2, F4	3	5~14	15	16
SUB.D F8, F6, F2	4	5~6	7	17
DIV.D F10, F0, F6	5	15~		
ADD.D F6, F8, F2	6	7~8	9	

保留站

Time	名称	Busy	Op	Vj	Vk	Qj	Qk	目的地
	Add1	No						
	Add2	No						
	Add3	No						
	Mult1	No						
38	Mult2	Yes	DIV.D	M3	M1			#5

寄存器

字段	F0	F2	F4	F6	F8	F10	F12	F14	F16	F18	F20	F22	F24	F26	F28	F30
ROB序号				#6		#5										
Busy	No	No	No	Yes	No	Yes	No	No	No	No	No	No	No	No	No	No
值	M3	M2		M1	M4											

再定序缓冲器 (ROB)

标记	项号	Busy	指令	目的地	值
	#1	No			
	#2	No			
	#3	No			
	#4	No			
HEAD	#5	Yes	DIV.D F10, F0, F6	F10	
	#6	Yes	ADD.D F6, F8, F2	F6	R[F8]+M2
TAIL	#7	No			
	#8	No			
	#9	No			
	#10	No			

Load缓冲器

名称	Busy	地址	目的地	值
Load1	No			
Load2	No			
Load3	No			

到第 17 周期结束, SUB 指令完成确认, 更新寄存器 F8 与存储器的值, 然后从 ROB 中移走。F8 更新为 M4 值。

此处第 18-54 周期的截图省略, 第 15-54 周期是 DIV 指令的执行阶段。

第二步: 用右边的按钮, 控制指令的执行

步进 退1步 前进5个周期 后退5个周期 执行到底 退出 当前周期: 55 转移至  GO

指令状态

指令	流出	执行	写结果	确认
L.D F6, 24(R2)	1	2~3	4	5
L.D F2, 12(R3)	2	3~4	5	6
MULT.D F0, F2, F4	3	5~14	15	16
SUB.D F8, F6, F2	4	5~6	7	17
DIV.D F10, F0, F6	5	15~54	55	
ADD.D F6, F8, F2	6	7~8	9	

保留站

Time	名称	Busy	Op	Vj	Vk	Qj	Qk	目的地
	Add1	No						
	Add2	No						
	Add3	No						
	Mult1	No						
	Mult2	No						

寄存器

字段	F0	F2	F4	F6	F8	F10	F12	F14	F16	F18	F20	F22	F24	F26	F28	F30
ROB序号				#6		#5										
Busy	No	No	No	Yes	No	Yes	No	No	No	No	No	No	No	No	No	No
值	M3	M2		M1	M4											

再定序缓冲器 (ROB)

标记	项号	Busy	指令	目的地	值
	#1	No			
	#2	No			
	#3	No			
	#4	No			
HEAD	#5	Yes	DIV.D F10, F0, F6	F10	M3/M1
	#6	Yes	ADD.D F6, F8, F2	F6	R[F8]+M2
TAIL	#7	No			
	#8	No			
	#9	No			
	#10	No			

Load缓冲器

名称	Busy	地址	目的地	值
Load1	No			
Load2	No			
Load3	No			

到第 55 周期, DIV 指令完成写结果, F10 值已旁路转发到 ROB 和相应功能部件中。

第二步: 用右边的按钮, 控制指令的执行

步进 退1步 前进5个周期 后退5个周期 执行到底 退出 当前周期: 56 转移至  GO

指令状态

指令	流出	执行	写结果	确认
L.D F6, 24(R2)	1	2~3	4	5
L.D F2, 12(R3)	2	3~4	5	6
MULT.D F0, F2, F4	3	5~14	15	16
SUB.D F8, F6, F2	4	5~6	7	17
DIV.D F10, F0, F6	5	15~54	55	56
ADD.D F6, F8, F2	6	7~8	9	

保留站

Time	名称	Busy	Op	Vj	Vk	Qj	Qk	目的地
	Add1	No						
	Add2	No						
	Add3	No						
	Mult1	No						
	Mult2	No						

寄存器

字段	F0	F2	F4	F6	F8	F10	F12	F14	F16	F18	F20	F22	F24	F26	F28	F30
ROB序号				#6		#5										
Busy	No	No	No	Yes	No	No	No	No	No	No	No	No	No	No	No	No
值	M3	M2		M1	M4	M5										

再定序缓冲器 (ROB)

标记	项号	Busy	指令	目的地	值
	#1	No			
	#2	No			
	#3	No			
	#4	No			
HEAD	#5	No			
TAIL	#6	Yes	ADD.D F6, F8, F2	F6	R[F8]+M2
	#7	No			
	#8	No			
	#9	No			
	#10	No			

Load缓冲器

名称	Busy	地址	目的地	值
Load1	No			
Load2	No			
Load3	No			

到第 56 周期结束, DIV 指令完成确认, 更新寄存器 F10 与存储器的值, 然后从 ROB 中移走。F10 更新为 M5 值。



第二步：用右边的按钮，控制指令的执行

步进 退1步 前进5个周期 后退5个周期 执行到底 退出 当前周期：57 转移至  go

**指令状态**

指令	流出	执行	写结果	确认
L.D F6, 24(R2)	1	2~3	4	5
L.D F2, 12(R3)	2	3~4	5	6
MULT.D F0, F2, F4	3	5~14	15	16
SUB.D F8, F6, F2	4	5~6	7	17
DIV.D F10, F0, F6	5	15~54	55	56
ADD.D F6, F8, F2	6	7~8	9	57

**再定序缓冲器(ROB)**

标记	项号	Busy	指令	目的地	值
	#1	No			
	#2	No			
	#3	No			
	#4	No			
	#5	No			
	#6	No			
HEAD/TAIL	#7	No			
	#8	No			
	#9	No			
	#10	No			

**保留站**

Time	名称	Busy	Op	Vj	Vk	Qj	Qk	目的地
	Add1	No						
	Add2	No						
	Add3	No						
	Mult1	No						
	Mult2	No						

**Load缓冲器**

名称	Busy	地址	目的地	值
Load1	No			
Load2	No			
Load3	No			

**寄存器**

字段	F0	F2	F4	F6	F8	F10	F12	F14	F16	F18	F20	F22	F24	F26	F28	F30
ROB项号	No	No	No	No	No	No	No	No	No	No	No	No	No	No	No	No
值	M3	M2		M6	M4	M5										

到第 57 周期结束，ADD 指令完成确认，更新寄存器 F6 与存储器的值，然后从 ROB 中移走。F6 更新为 M6 值。

2、对于与上面相同的延迟时间和代码段：

①给出在第 5 个时钟周期时，保留站的内容。

**保留站**

Time	名称	Busy	Op	Vj	Vk	Qj	Qk	目的地
2	Add1	Yes	SUB.D	M1	M[12+R[R3]]			#4
	Add2	No						
	Add3	No						
10	Mult1	Yes	MULT.D	M[12+R[R3]]	R[F4]			#3
	Mult2	Yes	DIV.D		M1	#3		#5

②步进 5 个时钟周期，ROB 的内容有哪些变化？

在第 5 个周期时，ROB 的内容如下：

**再定序缓冲器(ROB)**

标记	项号	Busy	指令	目的地	值
	#1	No			
HEAD	#2	Yes	L.D F2, 12(R3)	F2	M[12+R[R3]]
	#3	Yes	MULT.D F0, F2, F4	F0	
	#4	Yes	SUB.D F8, F6, F2	F8	
	#5	Yes	DIV.D F10, F0, F6	F10	
TAIL	#6	No			
	#7	No			
	#8	No			
	#9	No			
	#10	No			

在此基础上再步进 5 个周期，此时为第 10 个时钟周期，ROB 的内容如下。

第 2 条 load 指令完成确认，更新寄存器 F2 及存储器以后，从 ROB 中移走；SUB 和 ADD 指令先后完成写结果，将计算得到的目的地 F8、F6 的值先旁路转发给了 ROB 和相应功能部件，供给寄存器相关的指令使用，实现了基于硬件的猜测。等待确认（即指令提交）以后再更新寄存器和存储器的值。

再定序缓冲器 (ROB)					
标记	项号	Busy	指令	目的地	值
	#1	No			
	#2	No			
HEAD	#3	Yes	MULT.D F0, F2, F4	F0	
	#4	Yes	SUB.D F8, F6, F2	F8	M1-M2
	#5	Yes	DIV.D F10, F0, F6	F10	
	#6	Yes	ADD.D F6, F8, F2	F6	R[F8]+M2
TAIL	#7	No			
	#8	No			
	#9	No			
	#10	No			

③再步进 5 个时钟周期，给出这时保留站、ROB 以及浮点寄存器状态表的内容。

在②的基础上再步进 5 个周期，此时为第 15 个时钟周期，保留站、ROB 以及浮点寄存器状态表的内容如下：

指令状态					
指令	流出	执行	写结果	确认	
L.D F6, 24(R2)	1	2~3	4	5	
L.D F2, 12(R3)	2	3~4	5	6	
MULT.D F0, F2, F4	3	5~14	15		
SUB.D F8, F6, F2	4	5~6	7		
DIV.D F10, F0, F6	5	15~			
ADD.D F6, F8, F2	6	7~8	9		

  

再定序缓冲器 (ROB)					
标记	项号	Busy	指令	目的地	值
	#1	No			
	#2	No			
HEAD	#3	Yes	MULT.D F0, F2, F4	F0	M2*R[F4]
	#4	Yes	SUB.D F8, F6, F2	F8	M1-M2
	#5	Yes	DIV.D F10, F0, F6	F10	
	#6	Yes	ADD.D F6, F8, F2	F6	R[F8]+M2
TAIL	#7	No			
	#8	No			
	#9	No			
	#10	No			

  

保留站							
Time	名称	Busy	Op	Vj	Vk	Qj	Qk
	Add1	No					
	Add2	No					
	Add3	No					
	Mult1	No					
40	Mult2	Yes	DIV.D	M2*R[F4]	M1		#5

  

寄存器															
字段	F0	F2	F4	F6	F8	F10	F12	F14	F16	F18	F20	F22	F24	F26	F30
ROB项号	#3			#6	#4	#5									
Busy	Yes	No	No	Yes	Yes	Yes	No	No	No	No	No	No	No	No	No
值		M2			M1										

  

Load缓冲器				
名称	Busy	地址	目的地	值
Load1	No			
Load2	No			
Load3	No			

3、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。

(1) 自己编写的程序如下：

```

L.D    F6, 24(R2)
L.D    F2, 12(R3)
MUL.D  F0, F2, F4
ADD.D  F8, F2, F6
DIV.D  F10, F0, F6
SUB.D  F6, F8, F2

```

(2) 重复上述步骤 2 中的工作：

先在 ROB 算法模拟器中重新输入指令，然后根据要求修改各个浮点功能部件的延迟时间，随后开始逐步执行程序。

第二步：用右边的按钮，控制指令的执行

当前周期：1    转移至

指令状态

指令	流出	执行	写结果	确认
L.D F6, 24(R2)	1			
L.D F2, 12(R3)				
MULT.D F0, F2, F4				
ADD.D F8, F2, F6				
DIV.D F10, F0, F6				
SUB.D F6, F8, F2				

再定序缓冲器 (ROB)

标记	项号	Busy	指令	目的地	值
HEAD	#1	Yes	L.D F6, 24(R2)	F6	
TAIL	#2	No			
	#3	No			
	#4	No			
	#5	No			
	#6	No			
	#7	No			
	#8	No			
	#9	No			
	#10	No			

保留站

Time	名称	Busy	Op	Vj	Vk	Qj	Qk	目的地
	Add1	No						
	Add2	No						
	Add3	No						
	Mult1	No						
	Mult2	No						

Load缓冲器

名称	Busy	地址	目的地	值
Load1	Yes	24	#1	
Load2	No			
Load3	No			

寄存器

字段	F0	F2	F4	F6	F8	F10	F12	F14	F16	F18	F20	F22	F24	F26	F28	F30
ROB项号				#1												
Busy	No	No	No	Yes	No	No	No	No	No	No	No	No	No	No	No	No
值																

① 给出在第 5 个时钟周期时，保留站的内容。

Time	名称	Busy	Op	Vj	Vk	Qj	Qk	目的地
3	Add1	Yes	ADD.D	M[12+R[R3]]	M1			#4
	Add2	No						
	Add3	No						
8	Mult1	Yes	MULT.D	M[12+R[R3]]	R[F4]			#3
	Mult2	Yes	DIV.D		M1	#3		#5

② 步进 5 个时钟周期，ROB 的内容有哪些变化？

在第 5 个周期时，ROB 的内容如下：

标记	项号	Busy	指令	目的地	值
	#1	No			
HEAD	#2	Yes	L.D F2, 12(R3)	F2	M[12+R[R3]]
	#3	Yes	MULT.D F0, F2, F4	F0	
	#4	Yes	ADD.D F8, F2, F6	F8	
	#5	Yes	DIV.D F10, F0, F6	F10	
TAIL	#6	No			
	#7	No			
	#8	No			
	#9	No			
	#10	No			

在此基础上再步进 5 个周期，此时为第 10 个时钟周期，ROB 的内容如下。

第 2 条 load 指令完成确认，更新寄存器 F2 及存储器以后，从 ROB 中移走；ADD 指令完成写结果，将计算得到的目的地 F8 的值先旁路转发给了 ROB 和相应功能部件，供给寄存器相关的指令（SUB 指令）使用，实现了基于硬件的猜测。等待确认（即指令提交）以后再更新寄存器 F8 和存储器的值。

再定序缓冲器 (ROB)					
标记	项号	Busy	指令	目的地	值
	#1	No			
	#2	No			
HEAD	#3	Yes	MULT.D F0, F2, F4	F0	
	#4	Yes	ADD.D F8, F2, F6	F8	M2+M1
	#5	Yes	DIV.D F10, F0, F6	F10	
	#6	Yes	SUB.D F6, F8, F2	F6	
TAIL	#7	No			
	#8	No			
	#9	No			
	#10	No			

③再步进 5 个时钟周期，给出这时保留站、ROB 以及浮点寄存器状态表的内容。

在②的基础上再步进 5 个周期，此时为第 15 个时钟周期，保留站、ROB 以及浮点寄存器状态表的内容如下：

指令状态					
指令	流出	执行	写结果	确认	
L.D F6, 24(R2)	1	2~3	4	5	
L.D F2, 12(R3)	2	3~4	5	6	
MULT.D F0, F2, F4	3	5~12	13	14	
ADD.D F8, F2, F6	4	5~7	8	15	
DIV.D F10, F0, F6	5	13~			
SUB.D F6, F8, F2	6	8~10	11		

再定序缓冲器 (ROB)					
标记	项号	Busy	指令	目的地	值
	#1	No			
	#2	No			
	#3	No			
	#4	No			
HEAD	#5	Yes	DIV.D F10, F0, F6	F10	
	#6	Yes	SUB.D F6, F8, F2	F6	R[F8]-M2
TAIL	#7	No			
	#8	No			
	#9	No			
	#10	No			

保留站							
Time	名称	Busy	Op	Vj	Vk	Qj	Qk
	Add1	No					
	Add2	No					
	Add3	No					
	Mult1	No					
38	Mult2	Yes	DIV.D	M3	M1		

Load缓冲器					
名称	Busy	地址	目的地	值	
Load1	No				
Load2	No				
Load3	No				

寄存器															
字段	F0	F2	F4	F6	F8	F10	F12	F14	F16	F18	F20	F22	F24	F26	F28
ROB项号				#6	#5										
Busy	No	No	No	Yes	No	No	No	No	No	No	No	No	No	No	No
值	M3	M2		M1	M4										

## 五、实验中出现的问题和解决方法

- 实验中关于重排序缓冲（ROB）算法模拟器的使用

**问题：**该实验中运用 ROB 算法模拟器，主要是为了更清楚地学习掌握 ROB 算法的执行流程，体会该算法在基于硬件猜测过程中的优越性；而如果在输入代码段后直接采用“执行到底”按钮，则只能得到执行结果，无法观察到程序执行的细节部分，不利于我们对指令执行过程的掌握。

**解决方案：**先尽量多使用 ROB 算法模拟器上的“步进”按钮，进行步进形式执行，仔细观察每一个时钟周期前后各信息表中的内容变化情况，了解 ROB 算法中各程序执行的详细过程；有需要的情况下再灵活使用“退 1 步”“前进 5 个周期”“后退 5 个周期”等按钮，辅助观察程序执行情况。