

# Tomasulo With Reorder Buffer - Cycle 1

Time	Name	Busy	Op	Vj	Vk	Qj	Qk	Dest	Reservation Stations	
0	Add1	No								
0	Add2	No								
0	Add3	No								
0	Mult1	No								
0	Mult2	No								

  

Entry	Busy	Instruction	State	Destination	Value	Load1	Busy	Address
1	Yes	LD F6, 34(R2)	Issue	F6		Load1	Yes	34+Regs[R2]
2						Load2		
3						Load3		
4								
5								
6								
7								
8								
9								
10								

  

	F0	F2	F4	F6	F8	F10	F12	...	F30
Reorder #				#1					
Busy	no	no	no	Yes	no	no	no		no

# Tomasulo With Reorder Buffer - Cycle 2

Time	Name	Busy	Op	Vj	Vk	Qj	Qk	Dest	Reservation Stations	
0	Add1	No								
0	Add2	No								
0	Add3	No								
0	Mult1	No								
0	Mult2	No								

  

						Busy	Address
						Load1	Yes 34+Regs[R2]
						Load2	Yes 45+Regs[R3]
						Load3	

  

Entry	Busy	Instruction	State	Destination	Value
head → 1	Yes	LD F6, 34(R2)	Ex1	F6	
tail → 2	Yes	LD F2, 45(R3)	Issue	F2	
3					
4					
5					
6					
7					
8					
9					
10					

  

F0	F2	F4	F6	F8	F10	F12	...	F30
Reorder #	#2		#1					
Busy	Yes	no	Yes	no	no	no		no

Reorder Buffer

# Tomasulo With Reorder Buffer - Cycle 3

Time	Name	Busy	Op	Vj	Vk	Qj	Qk	Dest	Reservation Stations	
0	Add1	No								
0	Add2	No								
0	Add3	No								
0	Mult1	Yes	Mult		Regs[F4]	#2		#3		
0	Mult2	No								

  

							Busy	Address
							Load1	No
							Load2	Yes 45+Regs[R3]
							Load3	

  

	Entry	Busy	Instruction	State	Destination	Value
head →	1	Yes	LD F6, 34(R2)	write	F6	Mem[load1]
	2	Yes	LD F2, 45(R3)	Ex1	F2	
tail →	3	Yes	MULT F0, F2, F4	Issue	F0	
	4					
	5					
	6					
	7					
	8					
	9					
	10					

  

	F0	F2	F4	F6	F8	F10	F12	...	F30
Reorder #	#3	#2		#1					
Busy	Yes	Yes	no	Yes	no	no	no		no

Reorder Buffer

# Tomasulo With Reorder Buffer - Cycle 4

Time	Name	Busy	Op	Vj	Vk	Qj	Qk	Dest	Reservation Stations	
0	Add1	Yes	SUB	Regs[F6]	Mem[45+Regs[R3]]			#4		
0	Add2	No								
0	Add3	No								
0	Mult1	Yes	Mult	Mem[45+Regs[R3]]	Regs[F4]			#3		
0	Mult2	No								

  

							Busy	Address
Entry	Busy	Instruction	State	Destination	Value	Load1	No	
1	No	LD F6, 34(R2)	commit	F6	Mem[load1]	Load2	No	
2	Yes	LD F2, 45(R3)	write	F2	Mem[load2]	Load3		
3	Yes	MULT F0, F2, F4	EX1	F0				
4	Yes	SUBD F8, F6, F2	Issue	F8				
5								
6								
7								
8								
9								
10								

  

	F0	F2	F4	F6	F8	F10	F12	...	F30
Reorder #	#3	#2			#4				
Busy	Yes	Yes	no	no	Yes	no	no		no

# Tomasulo With Reorder Buffer - Cycle 5

Time	Name	Busy	Op	Vj	Vk	Qj	Qk	Dest	Reservation Stations	
0	Add1	Yes	SUB	Regs[F6]	Mem[45+Regs[R3]]			#4		
0	Add2	No								
0	Add3	No								
0	Mult1	Yes	Mult	Mem[45+Regs[R3]]	Regs[F4]			#3		
0	Mult2	Yes	DIV		Regs[F6]	#3		#5		

  

		Entry	Busy	Instruction	State	Destination	Value	Load1	Load2	Load3	Busy	Address
		1	No	LD F6, 34(R2)	commit	F6	Mem[load1]	No	No			
		2	No	LD F2, 45(R3)	commit	F2	Mem[load2]	No	No			
head	→	3	Yes	MULT F0, F2, F4	Ex2	F0						
		4	Yes	SUBD F8, F6, F2	Ex1	F8						
tail	→	5	Yes	DIVD F10, F0, F6	Issue	F10						
		6										
		7										
		8										
		9										
		10										

  

	F0	F2	F4	F6	F8	F10	F12	...	F30
Reorder #	#3				#4	#5			
Busy	Yes	no	no	no	Yes	Yes	no		no

# Tomasulo With Reorder Buffer - Cycle 6

Time	Name	Busy	Op	Vj	Vk	Qj	Qk	Dest	Reservation Stations	
0	Add1	Yes	SUB	Regs[F6]	Mem[45+Regs[R3]]			#4		
0	Add2	Yes	Add		Regs[F2]	#4		#6		
0	Add3	No								
0	Mult1	Yes	Mult	Mem[45+Regs[R3]]	Regs[F4]			#3		
0	Mult2	Yes	DIV		Regs[F6]	#3		#5		

  

		Entry	Busy	Instruction	State	Destination	Value	Load1	Load2	Load3	Busy	Address
head →	1	No		LD F6, 34(R2)	commit	F6	Mem[load1]	No	No			
	2	No		LD F2, 45(R3)	commit	F2	Mem[load2]	No	No			
	3	Yes		MULT F0, F2, F4	Ex3	F0						
	4	Yes		SUBD F8, F6, F2	Ex2	F8						
	5	Yes		DIVD F10, F0, F6	Issue	F10						
tail →	6	Yes		ADDD F6, F8, F2	Issue	F6						Reorder Buffer
	7											
	8											
	9											
	10											

  

	F0	F2	F4	F6	F8	F10	F12	...	F30
Reorder #	#3			#6	#4	#5			
Busy	Yes	no	no	Yes	Yes	Yes	no		no

# Tomasulo With Reorder Buffer - Cycle 7

Time	Name	Busy	Op	Vj	Vk	Qj	Qk	Dest	Reservation Stations	
0	Add1	No								
0	Add2	Yes	Add	#4	Regs[F2]			#6		
0	Add3	No								
0	Mult1	Yes	Mult	Mem[45+Regs[R3]]	Regs[F4]			#3		
0	Mult2	Yes	DIV		Regs[F6]	#3		#5		

  

		Entry	Busy	Instruction	State	Destination	Value	Load1	Load2	Load3	Busy	Address
		1	No	LD F6, 34(R2)	commit	F6	Mem[load1]	No	No			
		2	No	LD F2, 45(R3)	commit	F2	Mem[load2]					
head	→	3	Yes	MULT F0, F2, F4	Ex4	F0						
		4	Yes	SUBD F8, F6, F2	write	F8	F6 - #2					
		5	Yes	DIVD F10, F0, F6	Issue	F10						
tail	→	6	Yes	ADDD F6, F8, F2	EX1	F6						
		7										
		8										
		9										
		10										

  

	F0	F2	F4	F6	F8	F10	F12	...	F30
Reorder #	#3			#6	#4	#5			
Busy	Yes	no	no	Yes	Yes	Yes	no		no

Reorder Buffer

# Tomasulo With Reorder Buffer - Cycle 8

Time	Name	Busy	Op	Vj	Vk	Qj	Qk	Dest	Reservation Stations	
0	Add1	No								
0	Add2	Yes	Add	#4	Regs[F2]			#6		
0	Add3	No								
0	Mult1	Yes	Mult	Mem[45+Regs[R3]]	Regs[F4]			#3		
0	Mult2	Yes	DIV		Regs[F6]	#3		#5		

  

		Entry	Busy	Instruction	State	Destination	Value	Load1	Load2	Load3	Busy	Address
		1	No	LD F6, 34(R2)	commit	F6	Mem[load1]	No	No			
		2	No	LD F2, 45(R3)	commit	F2	Mem[load2]					
head	→	3	Yes	MULT F0, F2, F4	Ex5	F0						
		4	Yes	SUBD F8, F6, F2	write	F8	F6 - #2					
		5	Yes	DIVD F10, F0, F6	Issue	F10						
tail	→	6	Yes	ADDD F6, F8, F2	Ex2	F6						
		7										
		8										
		9										
		10										

  

	F0	F2	F4	F6	F8	F10	F12	...	F30
Reorder #	#3			#6	#4	#5			
Busy	Yes	no	no	Yes	Yes	Yes	no		no



# Tomasulo With Reorder Buffer - Cycle 9

Time	Name	Busy	Op	Vj	Vk	Qj	Qk	Dest	Reservation Stations	
0	Add1	No								
0	Add2	Yes	Add	#4	Regs[F2]			#6		
0	Add3	No								
0	Mult1	Yes	Mult	Mem[45+Regs[R3]]	Regs[F4]			#3		
0	Mult2	Yes	DIV		Regs[F6]	#3		#5		

  

		Entry	Busy	Instruction	State	Destination	Value	Load1	Load2	Load3	Busy	Address
		1	No	LD F6, 34(R2)	commit	F6	Mem[load1]	No	No	No		
		2	No	LD F2, 45(R3)	commit	F2	Mem[load2]					
head	→	3	Yes	MULT F0, F2, F4	Ex6	F0						
		4	Yes	SUBD F8, F6, F2	write	F8	F6 - #2					
		5	Yes	DIVD F10, F0, F6	Issue	F10						
tail	→	6	Yes	ADDD F6, F8, F2	write	F6	#4 + F2					
		7										
		8										
		9										
		10										

  

	F0	F2	F4	F6	F8	F10	F12	...	F30
Reorder #	#3			#6	#4	#5			
Busy	Yes	no	no	Yes	Yes	Yes	no		no

# Tomasulo With Reorder Buffer - Cycle 10

Time	Name	Busy	Op	Vj	Vk	Qj	Qk	Dest	Reservation Stations	
0	Add1	No								
0	Add2	No								
0	Add3	No								
0	Mult1	Yes	Mult	Mem[45+Regs[R3]]	Regs[F4]			#3		
0	Mult2	Yes	DIV		Regs[F6]	#3		#5		

  

		Entry	Busy	Instruction	State	Destination	Value	Load1	Load2	Load3	Busy	Address
		1	No	LD F6, 34(R2)	commit	F6	Mem[load1]	No	No			
		2	No	LD F2, 45(R3)	commit	F2	Mem[load2]					
head →		3	Yes	MULT F0, F2, F4	Ex7	F0						
		4	Yes	SUBD F8, F6, F2	write	F8	F6 - #2					
		5	Yes	DIVD F10, F0, F6	Issue	F10						
tail →		6	Yes	ADDD F6, F8, F2	write	F6	#4 + F2					
		7										
		8										
		9										
		10										

  

	F0	F2	F4	F6	F8	F10	F12	...	F30
Reorder #	#3			#6	#4	#5			
Busy	Yes	no	no	Yes	Yes	Yes	no		no

Reorder Buffer

# Tomasulo With Reorder Buffer - Cycle 11

Time	Name	Busy	Op	Vj	Vk	Qj	Qk	Dest
0	Add1	No						
0	Add2	No						
0	Add3	No						
0	Mult1	Yes	Mult	Mem[45+Regs[R3]]	Regs[F4]			#3
0	Mult2	Yes	DIV		Regs[F6]	#3		#5

Reservation Stations

Entry	Busy	Instruction	State	Destination	Value	Load1	Load2	Load3
1	No	LD F6, 34(R2)	commit	F6	Mem[load1]	No	No	No
2	No	LD F2, 45(R3)	commit	F2	Mem[load2]	No	No	No
3	Yes	MULT F0, F2, F4	Ex8	F0				
4	Yes	SUBD F8, F6, F2	write	F8	F6 - #2			
5	Yes	DIVD F10, F0, F6	Issue	F10				
6	Yes	ADDD F6, F8, F2	write	F6	#4 + F2			
7								
8								
9								
10								

head →

tail →

Reorder Buffer

	F0	F2	F4	F6	F8	F10	F12	...	F30
Reorder #	#3			#6	#4	#5			
Busy	Yes	no	no	Yes	Yes	Yes	no		no

# Tomasulo With Reorder Buffer - Cycle 12

Time	Name	Busy	Op	Vj	Vk	Qj	Qk	Dest	Reservation Stations	
0	Add1	No								
0	Add2	No								
0	Add3	No								
0	Mult1	Yes	Mult	Mem[45+Regs[R3]]	Regs[F4]			#3		
0	Mult2	Yes	DIV		Regs[F6]	#3		#5		

  

		Entry	Busy	Instruction	State	Destination	Value	Load1	Load2	Load3	Busy	Address
		1	No	LD F6, 34(R2)	commit	F6	Mem[load1]	No	No	No		
		2	No	LD F2, 45(R3)	commit	F2	Mem[load2]					
head	→	3	Yes	MULT F0, F2, F4	Ex9	F0						
		4	Yes	SUBD F8, F6, F2	write	F8	F6 - #2					
		5	Yes	DIVD F10, F0, F6	Issue	F10						
tail	→	6	Yes	ADDD F6, F8, F2	write	F6	#4 + F2					
		7										
		8										
		9										
		10										

  

	F0	F2	F4	F6	F8	F10	F12	...	F30
Reorder #	#3			#6	#4	#5			
Busy	Yes	no	no	Yes	Yes	Yes	no		no

Reorder Buffer

# Tomasulo With Reorder Buffer - Cycle 13

Time	Name	Busy	Op	Vj	Vk	Qj	Qk	Dest	Reservation Stations	
0	Add1	No								
0	Add2	No								
0	Add3	No								
0	Mult1	No								
0	Mult2	Yes	DIV	#2xRegs[F4]	Regs[F6]			#5		

  

								Busy	Address
Entry	Busy	Instruction	State	Destination	Value	Load1	Load2	Load3	
1	No	LD F6, 34(R2)	commit	F6	Mem[load1]				No
2	No	LD F2, 45(R3)	commit	F2	Mem[load2]				No
3	Yes	MULT F0, F2, F4	write	F0	#2 x Regs[F4]				
4	Yes	SUBD F8, F6, F2	write	F8	F6 - #2				
5	Yes	DIVD F10, F0, F6	Ex1	F10					
6	Yes	ADDD F6, F8, F2	write	F6	#4 + F2				
7									
8									
9									
10									

  

	F0	F2	F4	F6	F8	F10	F12	...	F30
Reorder #	#3			#6	#4	#5			
Busy	Yes	no	no	Yes	Yes	Yes	no		no

Figure 3.30

P 230

# Tomasulo With Reorder Buffer - Cycle 14

Time	Name	Busy	Op	Vj	Vk	Qj	Qk	Dest	Reservation Stations	
0	Add1	No								
0	Add2	No								
0	Add3	No								
0	Mult1	No								
0	Mult2	Yes	DIV	#2xRegs[F4]	Regs[F6]			#5		

  

								Busy	Address
Entry	Busy	Instruction	State	Destination	Value	Load1	Load2	Load3	
1	No	LD F6, 34(R2)	commit	F6	Mem[load1]				No
2	No	LD F2, 45(R3)	commit	F2	Mem[load2]				No
3	No	MULT F0, F2, F4	commit	F0	#2 x Regs[F4]				
4	Yes	SUBD F8, F6, F2	write	F8	F6 - #2				
5	Yes	DIVD F10, F0, F6	Ex2	F10					
6	Yes	ADDD F6, F8, F2	write	F6	#4 + F2				
7									
8									
9									
10									

  

	F0	F2	F4	F6	F8	F10	F12	...	F30
Reorder #				#6	#4	#5			
Busy	No	no	no	Yes	Yes	Yes	no		no

# Tomasulo With Reorder Buffer - Cycle 15

Time	Name	Busy	Op	Vj	Vk	Qj	Qk	Dest	Reservation Stations	
0	Add1	No								
0	Add2	No								
0	Add3	No								
0	Mult1	No								
0	Mult2	Yes	DIV	#2xRegs[F4]	Regs[F6]			#5		

  

		Entry	Busy	Instruction	State	Destination	Value	Load1	Load2	Load3	Busy	Address
		1	No	LD F6, 34(R2)	commit	F6	Mem[load1]	No	No			
		2	No	LD F2, 45(R3)	commit	F2	Mem[load2]	No	No			
		3	No	MULT F0, F2, F4	commit	F0	#2 x Regs[F4]					
		4	No	SUBD F8, F6, F2	commit	F8	F6 - #2					
head	→	5	Yes	DIVD F10, F0, F6	Ex3	F10						
tail	→	6	Yes	ADDD F6, F8, F2	write	F6	#4 + F2					
		7										
		8										
		9										
		10										

  

	F0	F2	F4	F6	F8	F10	F12	...	F30
Reorder #				#6		#5			
Busy	no	no	no	Yes	no	Yes	no		no

Reorder Buffer

# Tomasulo With Reorder Buffer - Cycle 16

Time	Name	Busy	Op	Vj	Vk	Qj	Qk	Dest	Reservation Stations	
0	Add1	No								
0	Add2	No								
0	Add3	No								
0	Mult1	No								
0	Mult2	Yes	DIV	#2xRegs[F4]	Regs[F6]			#5		

  

Entry	Busy	Instruction	State	Destination	Value	Load1	Load2	Load3	Busy	Address
1	No	LD F6, 34(R2)	commit	F6	Mem[load1]	No	No	No		
2	No	LD F2, 45(R3)	commit	F2	Mem[load2]	No	No	No		
3	No	MULT F0, F2, F4	commit	F0	#2 x Regs[F4]					
4	No	SUBD F8, F6, F2	commit	F8	F6 - #2					
5	Yes	DIVD F10, F0, F6	Ex4	F10						
6	Yes	ADDD F6, F8, F2	write	F6	#4 + F2					
7										
8										
9										
10										

  

F0	F2	F4	F6	F8	F10	F12	...	F30
Reorder #			#6		#5			
Busy	no	no	Yes	no	Yes	no		no

head  
tail

Reorder Buffer

Need 36 more  
EX cycles for  
DIV to finish...



# Tomasulo With Reorder Buffer: Summary

---

Instruction	Issue	Exec Comp	Writeback	Commit
LD F6, 34(R2)	1	2	3	4
LD F2, 45(R3)	2	3	4	5
MULT F0, F2, F4	3	12	13	14
SUBD F8, F6, F2	4	6	7	15
DIVD F10, F0, F6	5	52	53	54
ADDD F6, F8, F2	6	8	9	55

In-order Issue/Commit, Out-of-Order Execution/Writeback

---

# Precise State with ROB

---

- ROB maintains precise state and allows speculation
  - Waits until precise condition reaches retire/commit stage
  - (Or until branch is noted mis-predicted)
  - Clear ROB, RS, and register status table (Flush)
  - Service exception/Restart from True Branch target
- Need to do similar things with memory ops
  - Called Memory Ordering Buffer (MOB)
    - Completed stores write to MOB then complete (write to memory) in-order (when they reach head of buffer)

# Example of Speculative State of Reorder Buffer

0		Add1	No					Reservation Stations	
0		Add2	No						
0		Add3	No						
0		Mult1	No	MULT	Mem[0+Regs[R1]]	Regs[F2]	#2		
0		Mult2	No	MULT	Mem[0+Regs[R1]]	Regs[F2]	#7		

		Busy		Address				
First loop	Entry	Busy	Instruction	State	Destination	Value	Load1	No
	1	No	LD F0, 0(R1)	commit	F0	Mem[0+R1]	Load2	No
	2	No	MULT F4, F0, F2	commit	F4	F0 x F2	Load3	
	3	Yes	SD 0(R1), F4	write	0+Reg[R1]	#2		
	4	Yes	SUBI R1, R1, 8	write	R1	R1 - 8		
Second loop	5	Yes	BNEZ R1, Loop	write				
	6	Yes	LD F0, 0(R1)	write	F0	Mem[#4]		
	7	Yes	MULT F4, F0, F2	write	F4	#6 X F2		
	8	Yes	SD 0(R1), F4	write	0+Regs[R1]	#7		
	9	Yes	SUBI R1, R1, 8	write	R1	#4 - 8		
	10	Yes	BNEZ R1, Loop	write				

	F0	F2	F4	F6	F8	F10	F12	...	F30
Reorder #	6		7						
Busy	yes	no	yes	no	no	no	no		no

Multiply has just reached commit, so other instructions can start committing