Lecture 4: Tomasulo Algorithm and Dynamic Branch Prediction

Professor David A. Patterson
Computer Science 252
Fall 1996

Review: Summary

- Instruction Level Parallelism (ILP) in SW or HW
- Loop level parallelism is easiest to see
- SW parallelism dependencies defined for program, hazards if HW cannot resolve
- SW dependencies/compiler sophistication determine if compiler can unroll loops
  - Memory dependencies hardest to determine
- HW exploiting ILP
  - Works when can't know dependence at run time
  - Code for one machine runs well on another
- Key idea of Scoreboard: Allow instructions behind stall to proceed (Decode => Issue instr & read operands)
  - Enables out-of-order execution => out-of-order completion
  - ID stage checked both for structural & data dependencies

Review: Three Parts of the Scoreboard

1. Instruction status—which of 4 steps the instruction is in

2. Functional unit status—Indicates the state of the functional unit (FU). 9 fields for each functional unit
   - Busy—Indicates whether the unit is busy or not
   - Op—Operation to perform in the unit (e.g., + or –)
   - Fi—Destination register
   - Fj, Fk—Source-register numbers
   - Qj, Qk—Functional units producing source registers Fj, Fk
   - Rj, Rk—Flags indicating when Fj, Fk are ready

3. Register result status—Indicates which functional unit will write each register, if one exists. Blank when no pending instructions will write that register

Review: Scoreboard Example Cycle 3

Instruction status

Instruction	j	k	Issue	Read operands	Execution complete	Write Result
LD F6	34+	R2	1	2	3	
LD F2	45+	R3				
MULTI F0	F2	F4				
SUBD F8	F6	F2				
DIVD F10	F0	F6				
ADDD F6	F8	F2				

Functional unit status

Time Name	Busy	Op	dest Fi	S1 Fj	S2 Fk	FU for Qj	FU for Qk	Fj? Rj	Fk? Rk
Integer	Yes	Load	F6		R2				Yes
Mult1	No								
Mult2	No								
Add	No								
Divide	No								

Register result status

Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
3 FU				Integer					

- Issue MULT? No, stall on structural hazard

Review: Scoreboard Example Cycle 9

Instruction status

Instruction	j	k	Issue	Read operands	Execution complete	Write Result
LD F6	34+	R2	1	2	3	4
LD F2	45+	R3	5	6	7	8
MULTI F0	F2	F4	6	9		
SUBD F8	F6	F2	7	9		
DIVD F10	F0	F6	8			
ADDD F6	F8	F2				

Functional unit status

Time	Name	Busy	Op	dest Fi	S1 Fj	S2 Fk	FU for Qj	FU for Qk	Fj? Rj	Fk? Rk
	Integer	No								
10	Mult1	Yes	Mult	F0	F2	F4			Yes	Yes
	Mult2	No								
2	Add	Yes	Sub	F8	F6	F2			Yes	Yes
	Divide	Yes	Div	F10	F0	F6	Mult1		No	Yes

Register result status

Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
9	Mult1				Add	Divide			

- **Read operands for MULT & SUBD? Issue ADDD?**

Review: Scoreboard Example Cycle 17

Instruction status

Instruction	j	k	Issue	Read operands	Execution complete	Write Result
LD F6	34+	R2	1	2	3	4
LD F2	45+	R3	5	6	7	8
MULTI F0	F2	F4	6	9		
SUBD F8	F6	F2	7	9	11	12
DIVD F10	F0	F6	8			
ADDD F6	F8	F2	13	14	16	

Functional unit status

Time	Name	Busy	Op	dest Fi	S1 Fj	S2 Fk	FU for Qj	FU for Qk	Fj? Rj	Fk? Rk
	Integer	No								
2	Mult1	Yes	Mult	F0	F2	F4			Yes	Yes
	Mult2	No								
	Add	Yes	Add	F6	F8	F2			Yes	Yes
	Divide	Yes	Div	F10	F0	F6	Mult1		No	Yes

Register result status

Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
17	Mult1			Add		Divide			

- **Write result of ADDD? No, WAR hazard**

Review: Scoreboard Example Cycle 62

Instruction status

Instruction	j	k	Issue	Read operands	Execution complete	Write Result
LD F6	34+	R2	1	2	3	4
LD F2	45+	R3	5	6	7	8
MULTI F0	F2	F4	6	9	19	20
SUBD F8	F6	F2	7	9	11	12
DIVD F10	F0	F6	8	21	61	62
ADDD F6	F8	F2	13	14	16	22

Functional unit status

Time	Name	Busy	Op	dest Fi	S1 Fj	S2 Fk	FU for Qj	FU for Qk	Fj? Rj	Fk? Rk
	Integer	No								
	Mult1	No								
	Mult2	No								
	Add	No								
0	Divide	No								

Register result status

Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
62									

- **In-order issue; out-of-order execute & commit**

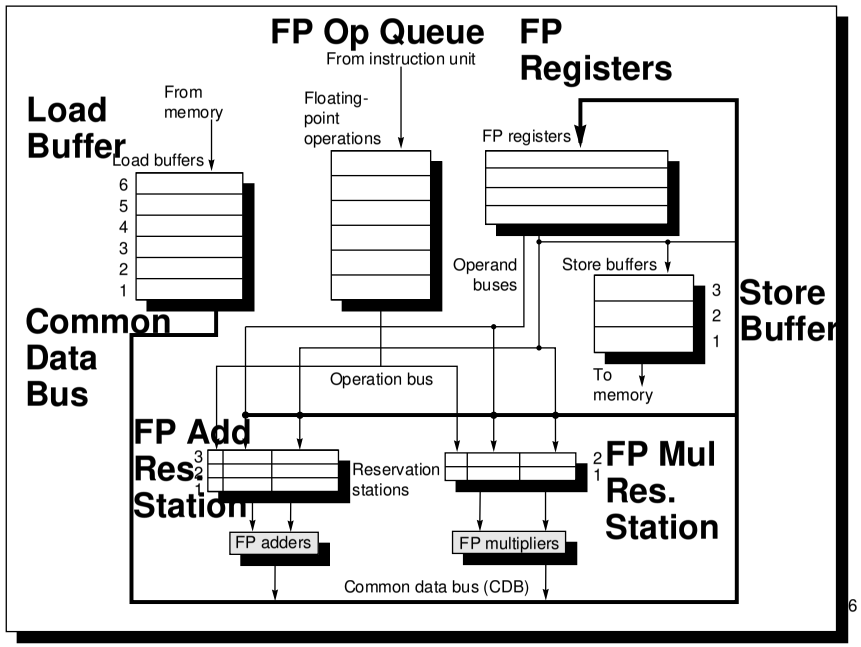
Review: Scoreboard Summary

- **Speedup 1.7 from compiler; 2.5 by hand BUT slow memory (no cache)**
- **Limitations of 6600 scoreboard**
  - **No forwarding (First write regsiter then read it)**
  - **Limited to instructions in basic block (small *window*)**
  - **Number of functional units(structural hazards)**
  - **Wait for WAR hazards**
  - **Prevent WAW hazards**

Another Dynamic Algorithm: Tomasulo Algorithm

- For IBM 360/91 about 3 years after CDC 6600 (1966)
- Goal: High Performance without special compilers
- Differences between IBM 360 & CDC 6600 ISA
  - IBM has only 2 register specifiers/instr vs. 3 in CDC 6600
  - IBM has 4 FP registers vs. 8 in CDC 6600
- Differences between Tomasulo Algorithm & Scoreboard
  - Control & buffers *distributed* with Function Units vs. centralized in scoreboard; called "reservation stations"
  - Registers in instructions replaced by pointers to reservation station buffer
  - HW *renaming* of registers to avoid WAR, WAW hazards
  - Common Data Bus broadcasts results to all FUs
  - Load and Stores treated as FUs as well
- Why Study? It lead to Pentium Pro, PowerPC 604, …

Tomasulo Organization



Reservation Station Components

Op—Operation to perform in the unit (e.g., + or –)

Qj, Qk—Reservation stations producing source registers (value to be written)

Vj, Vk—Value of Source operands

Rj, Rk—Flags indicating when Vj, Vk are ready

Busy—Indicates reservation station and FU is busy

Register result status—Indicates which functional unit will write each register, if one exists. Blank when no pending instructions that will write that register.

Three Stages of Tomasulo Algorithm

1. Issue—get instruction from FP Op Queue

   If reservation station free (no structural hazard), the scoreboard issues instr & sends operands (renames registers).

2. Execution—operate on operands (EX)

   When both operands ready then execute; if not ready, watch CDB for result

3. Write result—finish execution (WB)

   Write on Common Data Bus to all awaiting units; mark reservation station available

- Normal bus: data + destination
  Common Data Bus: data + *source*:
  Normal = "Go To" bus; CDB = "Come From" bus

Tomasulo Example Cycle 0

Instruction status									
Instruction	j	k	Issue	Execution complete	Write Result			Busy	Address
LD F6	34+	R2					Load1	No	
LD F2	45+	R3					Load2	No	
MULTI F0	F2	F4					Load3	No	
SUBD F8	F6	F2							
DIVD F10	F0	F6							
ADDD F6	F8	F2							

Reservation Stations							
Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for j Qj	RS for k Qk
0	Add1	No					
0	Add2	No					
0	Add3	No					
0	Mult1	No					
0	Mult2	No					

Register result status

Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
0	FU								

CS 252 Administrivia

- **Reading Assignments for Lectures 3 to 6**
  - **Computer Architecture: A Quantitative Approach, Second Edition (1996)**
  - **Chapter 4, Appendix B**
- **Exercises for Lectures 3 to 6**
  - **4.14, all parts (a - k)**
  - **4.25, table of pros and cons + short essay**
  - **B.3, all parts (a - g)**
  - **B.15, table of pros and cons + short essay**
  - **Due Monday September 16 at 5PM homework box in 283 Soda (building is locked at 6:45 PM)**
  - **Done in pairs, but both need to understand whole assignment**
  - **Study groups encouraged, but pairs do own work**

Tomasulo Example Cycle 1

Instruction status									
Instruction	j	k	Issue	Execution complete	Write Result			Busy	Address
LD F6	34+	R2	1				Load1	Yes	34+R2
LD F2	45+	R3					Load2	No	
MULTI F0	F2	F4					Load3	No	
SUBD F8	F6	F2							
DIVD F10	F0	F6							
ADDD F6	F8	F2							

Reservation Stations							
Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for j Qj	RS for k Qk
0	Add1	No					
0	Add2	No					
	Add3	No					
0	Mult1	No					
0	Mult2	No					

Register result status

Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
1	FU			Load1					

Tomasulo Example Cycle 2

Instruction status									
Instruction	j	k	Issue	Execution complete	Write Result			Busy	Address
LD F6	34+	R2	1				Load1	Yes	34+R2
LD F2	45+	R3	2				Load2	Yes	45+R3
MULTI F0	F2	F4					Load3	No	
SUBD F8	F6	F2							
DIVD F10	F0	F6							
ADDD F6	F8	F2							

Reservation Stations							
Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for j Qj	RS for k Qk
0	Add1	No					
0	Add2	No					
	Add3	No					
0	Mult1	No					
0	Mult2	No					

Register result status

Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
2	FU	Load2		Load1					

Tomasulo Example Cycle 3

Instruction status									
Instruction	j	k	Issue	Execution complete	Write Result			Busy	Address
LD F6	34+	R2	1	3			Load1	Yes	34+R2
LD F2	45+	R3	2				Load2	Yes	45+R3
MULTI F0	F2	F4	3				Load3	No	
SUBD F8	F6	F2							
DIVD F10	F0	F6							
ADDD F6	F8	F2							

Reservation Stations							
Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for j Qj	RS for k Qk
	Add1	No					
0	Add2	No					
	Add3	No					
0	Mult1	Yes	MULTD		R(F4)	Load2	
0	Mult2	No					

Register result status									
Clock		F0	F2	F4	F6	F8	F10	F12	... F30
3	FU	Mult1	Load2		Load1				

- **Issue MULT now vs in scoreboard?**

Tomasulo Example Cycle 4

Instruction status									
Instruction	j	k	Issue	Execution complete	Write Result			Busy	Address
LD F6	34+	R2	1	3	4		Load1	No	
LD F2	45+	R3	2	4			Load2	Yes	45+R3
MULTI F0	F2	F4	3				Load3	No	
SUBD F8	F6	F2	4						
DIVD F10	F0	F6							
ADDD F6	F8	F2							

Reservation Stations							
Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for j Qj	RS for k Qk
	Add1	Yes	SUBD	M(34+R2)			Load2
0	Add2	No					
	Add3	No					
0	Mult1	Yes	MULTD		R(F4)	Load2	
0	Mult2	No					

Register result status									
Clock		F0	F2	F4	F6	F8	F10	F12	... F30
4	FU	Mult1	Load2		M(34+R2)	Add1			

Tomasulo Example Cycle 5

Instruction status									
Instruction	j	k	Issue	Execution complete	Write Result			Busy	Address
LD F6	34+	R2	1	3	4		Load1	No	
LD F2	45+	R3	2	4	5		Load2	No	
MULTI F0	F2	F4	3				Load3	No	
SUBD F8	F6	F2	4						
DIVD F10	F0	F6	5						
ADDD F6	F8	F2							

Reservation Stations							
Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for j Qj	RS for k Qk
2	Add1	Yes	SUBD	M(34+R2)	M(45+R3)		
0	Add2	No					
	Add3	No					
10	Mult1	Yes	MULTD	M(45+R3)	R(F4)		
0	Mult2	Yes	DIVD		M(34+R2)	Mult1	

Register result status									
Clock		F0	F2	F4	F6	F8	F10	F12	... F30
5	FU	Mult1	M(45+R3)		M(34+R2)	Add1	Mult2		

Tomasulo Example Cycle 6

Instruction status									
Instruction	j	k	Issue	Execution complete	Write Result			Busy	Address
LD F6	34+	R2	1	3	4		Load1	No	
LD F2	45+	R3	2	4	5		Load2	No	
MULTI F0	F2	F4	3				Load3	No	
SUBD F8	F6	F2	4						
DIVD F10	F0	F6	5						
ADDD F6	F8	F2	6						

Reservation Stations							
Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for j Qj	RS for k Qk
1	Add1	Yes	SUBD	M(34+R2)	M(45+R3)		
0	Add2	Yes	ADDD		M(45+R3)	Add1	
	Add3	No					
9	Mult1	Yes	MULTD	M(45+R3)	R(F4)		
0	Mult2	Yes	DIVD		M(34+R2)	Mult1	

Register result status									
Clock		F0	F2	F4	F6	F8	F10	F12	... F30
6	FU	Mult1	M(45+R3)		Add2	Add1	Mult2		

- **Issue MULT vs. scoreboard?**

Tomasulo Example Cycle 7

Instruction status									
Instruction		j	k	Issue	Execution complete	Write Result		Busy	Address
LD	F6	34+	R2	1	3	4	Load1	No	
LD	F2	45+	R3	2	4	5	Load2	No	
MULTI	F0	F2	F4	3			Load3	No	
SUBD	F8	F6	F2	4	7				
DIVD	F10	F0	F6	5					
ADDD	F6	F8	F2	6					

Reservation Stations							
Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for j Qj	RS for k Qk
0	Add1	Yes	SUBD	M(34+R2)	M(45+R3)		
0	Add2	Yes	ADDD		M(45+R3)	Add1	
	Add3	No					
8	Mult1	Yes	MULTD	M(45+R3)	R(F4)		
0	Mult2	Yes	DIVD		M(34+R2)	Mult1	

Register result status									
Clock		F0	F2	F4	F6	F8	F10	F12 ...	F30
7	FU	Mult1	M(45+R3)		Add2	Add1	Mult2		

Tomasulo Example Cycle 8

Instruction status									
Instruction		j	k	Issue	Execution complete	Write Result		Busy	Address
LD	F6	34+	R2	1	3	4	Load1	No	
LD	F2	45+	R3	2	4	5	Load2	No	
MULTI	F0	F2	F4	3			Load3	No	
SUBD	F8	F6	F2	4	7	8			
DIVD	F10	F0	F6	5					
ADDD	F6	F8	F2	6					

Reservation Stations							
Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for j Qj	RS for k Qk
0	Add1	Yes	SUBD	M(34+R2)	M(45+R3)		
2	Add2	Yes	ADDD	M()-M()	M(45+R3)		
0	Add3	No					
7	Mult1	Yes	MULTD	M(45+R3)	R(F4)		
0	Mult2	Yes	DIVD		M(34+R2)	Mult1	

Register result status									
Clock		F0	F2	F4	F6	F8	F10	F12 ...	F30
8	FU	Mult1	M(45+R3)		Add2	M()-M()	Mult2		

Tomasulo Example Cycle 9

Instruction status									
Instruction		j	k	Issue	Execution complete	Write Result		Busy	Address
LD	F6	34+	R2	1	3	4	Load1	No	
LD	F2	45+	R3	2	4	5	Load2	No	
MULTI	F0	F2	F4	3			Load3	No	
SUBD	F8	F6	F2	4	7	8			
DIVD	F10	F0	F6	5					
ADDD	F6	F8	F2	6					

Reservation Stations							
Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for j Qj	RS for k Qk
0	Add1	No					
1	Add2	Yes	ADDD	M()–M()	M(45+R3)		
0	Add3	No					
6	Mult1	Yes	MULTD	M(45+R3)	R(F4)		
0	Mult2	Yes	DIVD		M(34+R2)	Mult1	

Register result status									
Clock		F0	F2	F4	F6	F8	F10	F12 ...	F30
9	FU	Mult1	M(45+R3)		Add2	M()–M()	Mult2		

Tomasulo Example Cycle 10

Instruction status									
Instruction		j	k	Issue	Execution complete	Write Result		Busy	Address
LD	F6	34+	R2	1	3	4	Load1	No	
LD	F2	45+	R3	2	4	5	Load2	No	
MULTI	F0	F2	F4	3			Load3	No	
SUBD	F8	F6	F2	4	7	8			
DIVD	F10	F0	F6	5					
ADDD	F6	F8	F2	6	10				

Reservation Stations							
Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for j Qj	RS for k Qk
0	Add1	No					
0	Add2	Yes	ADDD	M()–M()	M(45+R3)		
0	Add3	No					
5	Mult1	Yes	MULTD	M(45+R3)	R(F4)		
0	Mult2	Yes	DIVD		M(34+R2)	Mult1	

Register result status									
Clock		F0	F2	F4	F6	F8	F10	F12 ...	F30
10	FU	Mult1	M(45+R3)		Add2	M()–M()	Mult2		

Tomasulo Example Cycle 11

Instruction status									
Instruction	j	k	Issue	Execution complete	Write Result			Busy	Address
LD F6	34+	R2	1	3	4		Load1	No	
LD F2	45+	R3	2	4	5		Load2	No	
MULTI F0	F2	F4	3				Load3	No	
SUBD F8	F6	F2	4	7	8				
DIVD F10	F0	F6	5						
ADDD F6	F8	F2	6	10	11				

Reservation Stations							
Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for j Qj	RS for k Qk
0	Add1	No					
1	Add2	No					
0	Add3	No					
4	Mult1	Yes	MULTD	M(45+R3)	R(F4)		
0	Mult2	Yes	DIVD		M(34+R2)	Mult1	

Register result status									
Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
11 FU	Mult1	M(45+R3)		(M-M)+M()	M()–M()	Mult2			

- **Write result of ADDD vs. scoreboard?**

Tomasulo Example Cycle 12

Instruction status									
Instruction	j	k	Issue	Execution complete	Write Result			Busy	Address
LD F6	34+	R2	1	3	4		Load1	No	
LD F2	45+	R3	2	4	5		Load2	No	
MULTI F0	F2	F4	3				Load3	No	
SUBD F8	F6	F2	4	6	7				
DIVD F10	F0	F6	5						
ADDD F6	F8	F2	6	10	11				

Reservation Stations							
Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for j Qj	RS for k Qk
0	Add1	No					
0	Add2	No					
0	Add3	No					
3	Mult1	Yes	MULTD	M(45+R3)	R(F4)		
0	Mult2	Yes	DIVD		M(34+R2)	Mult1	

Register result status									
Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
12 FU	Mult1	M(45+R3)		(M-M)+M()	M()–M()	Mult2			

Tomasulo Example Cycle 13

Instruction status									
Instruction	j	k	Issue	Execution complete	Write Result			Busy	Address
LD F6	34+	R2	1	3	4		Load1	No	
LD F2	45+	R3	2	4	5		Load2	No	
MULTI F0	F2	F4	3				Load3	No	
SUBD F8	F6	F2	4	7	8				
DIVD F10	F0	F6	5						
ADDD F6	F8	F2	6	10	11				

Reservation Stations							
Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for j Qj	RS for k Qk
0	Add1	No					
0	Add2	No					
	Add3	No					
2	Mult1	Yes	MULTD	M(45+R3)	R(F4)		
0	Mult2	Yes	DIVD		M(34+R2)	Mult1	

Register result status									
Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
13 FU	Mult1	M(45+R3)		(M–M)+M()	M()–M()	Mult2			

Tomasulo Example Cycle 14

Instruction status									
Instruction	j	k	Issue	Execution complete	Write Result			Busy	Address
LD F6	34+	R2	1	3	4		Load1	No	
LD F2	45+	R3	2	4	5		Load2	No	
MULTI F0	F2	F4	3				Load3	No	
SUBD F8	F6	F2	4	7	8				
DIVD F10	F0	F6	5						
ADDD F6	F8	F2	6	10	11				

Reservation Stations							
Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for j Qj	RS for k Qk
0	Add1	No					
0	Add2	No					
0	Add3	No					
1	Mult1	Yes	MULTD	M(45+R3)	R(F4)		
0	Mult2	Yes	DIVD		M(34+R2)	Mult1	

Register result status									
Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
14 FU	Mult1	M(45+R3)		(M–M)+M()	M()–M()	Mult2			

Tomasulo Example Cycle 15

Instruction status				Execution	Write			
Instruction	j	k	Issue	complete	Result		Busy	Address
LD F6	34+	R2	1	3	4	Load1	No	
LD F2	45+	R3	2	4	5	Load2	No	
MULTI F0	F2	F4	3	15		Load3	No	
SUBD F8	F6	F2	4	7	8			
DIVD F10	F0	F6	5					
ADDD F6	F8	F2	6	10	11			

Reservation Stations			S1	S2	RS for j	RS for k
Time	Name	Busy	Op	Vj	Vk	Qj	Qk
0	Add1	No					
0	Add2	No					
	Add3	No					
0	Mult1	Yes	MULTD	M(45+R3)	R(F4)		
0	Mult2	Yes	DIVD		M(34+R2)	Mult1	

Register result status

Clock		F0	F2	F4	F6	F8	F10	F12 ...	F30
15	FU	Mult1	M(45+R3)		(M–M)+M()	M()–M()	Mult2		

Tomasulo Example Cycle 16

Instruction status				Execution	Write			
Instruction	j	k	Issue	complete	Result		Busy	Address
LD F6	34+	R2	1	3	4	Load1	No	
LD F2	45+	R3	2	4	5	Load2	No	
MULTI F0	F2	F4	3	15	16	Load3	No	
SUBD F8	F6	F2	4	7	8			
DIVD F10	F0	F6	5					
ADDD F6	F8	F2	6	10	11			

Reservation Stations			S1	S2	RS for j	RS for k
Time	Name	Busy	Op	Vj	Vk	Qj	Qk
0	Add1	No					
0	Add2	No					
	Add3	No					
0	Mult1	No					
40	Mult2	Yes	DIVD	M*F4	M(34+R2)		

Register result status

Clock		F0	F2	F4	F6	F8	F10	F12 ...	F30
16	FU	M*F4	M(45+R3)		(M–M)+M()	M()–M()	Mult2		

Tomasulo Example Cycle 55

Instruction status				Execution	Write			
Instruction	j	k	Issue	complete	Result		Busy	Address
LD F6	34+	R2	1	3	4	Load1	No	
LD F2	45+	R3	2	4	5	Load2	No	
MULTI F0	F2	F4	3	15	16	Load3	No	
SUBD F8	F6	F2	4	7	8			
DIVD F10	F0	F6	5					
ADDD F6	F8	F2	6	10	11			

Reservation Stations			S1	S2	RS for j	RS for k
Time	Name	Busy	Op	Vj	Vk	Qj	Qk
0	Add1	No					
0	Add2	No					
	Add3	No					
0	Mult1	No					
1	Mult2	Yes	DIVD	M*F4	M(34+R2)		

Register result status

Clock		F0	F2	F4	F6	F8	F10	F12 ...	F30
55	FU	M*F4	M(45+R3)		(M–M)+M()	M()–M()	Mult2		

Tomasulo Example Cycle 56

Instruction status				Execution	Write			
Instruction	j	k	Issue	complete	Result		Busy	Address
LD F6	34+	R2	1	3	4	Load1	No	
LD F2	45+	R3	2	4	5	Load2	No	
MULTI F0	F2	F4	3	15	16	Load3	No	
SUBD F8	F6	F2	4	7	8			
DIVD F10	F0	F6	5	56				
ADDD F6	F8	F2	6	10	11			

Reservation Stations			S1	S2	RS for j	RS for k
Time	Name	Busy	Op	Vj	Vk	Qj	Qk
0	Add1	No					
0	Add2	No					
	Add3	No					
0	Mult1	No					
0	Mult2	Yes	DIVD	M*F4	M(34+R2)		

Register result status

Clock		F0	F2	F4	F6	F8	F10	F12 ...	F30
56	FU	M*F4	M(45+R3)		(M–M)+M()	M()–M()	Mult2		

Tomasulo Example Cycle 57

Instruction status

Instruction	j	k	Issue	Execution complete	Write Result	Busy	Address
LD F6	34+	R2	1	3	4	Load1	No
LD F2	45+	R3	2	4	5	Load2	No
MULTI F0	F2	F4	3	15	16	Load3	No
SUBD F8	F6	F2	4	7	8		
DIVD F10	F0	F6	5	56	57		
ADDD F6	F8	F2	6	10	11		

Reservation Stations

Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for j Qj	RS for k Qk
0	Add1	No					
0	Add2	No					
	Add3	No					
0	Mult1	No					
0	Mult2	No					

Register result status

Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
57 FU	M*F4	M(45+R3)		(M–M)+M()	M()–M()	M*F4/M			

Compare to Scoreboard Cycle 62

Instruction status

Instruction	j	k	Issue	Read operand	Execution complete	Write Result
LD F6	34+	R2	1	2	3	4
LD F2	45+	R3	5	6	7	8
MULTI F0	F2	F4	6	9	19	20
SUBD F8	F6	F2	7	9	11	12
DIVD F10	F0	F6	8	21	61	62
ADDD F6	F8	F2	13	14	16	22

Functional unit status

Time	Name	Busy	Op	dest Fi	S1 Fj	S2 Fk	FU for j Qj	FU for k Qk	Fj? Rj	Fk? Rk
	Integer	No								
	Mult1	No								
	Mult2	No								
	Add	No								
0	Divide	No								

Register result status

Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
62 FU									

- **In-order issue; out-of-order execute & commit**

Tomasulo Loop Example

```
Loop: LD     F0   0   R1
      MULTD  F4   F0  F2
      SD     F4   0   R1
      SUBI   R1   R1  #8
      BNEZ   R1   Loop
```

- **Multiply takes 4 clocks**
- **Assume loads talke 8 clocks (cache miss)**

Loop Example Cycle 0

Instruction status

Instruction	j	k	iteration	Issue	Execution complete	Write Result		Busy	Address	
LD F0	0	R1	1				Load1	No		
MULTI F4	F0	F2	1				Load2	No		
SD F4	0	R1	1				Load3	No		Qi
LD F0	0	R1	2				Store1	No		
MULTI F4	F0	F2	2				Store2	No		
SD F4	0	R1	2				Store3	No		

Reservation Stations

Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for j Qj	RS for k Qk
0	Add1	No					
0	Add2	No					
0	Add3	No					
0	Mult1	No					
0	Mult2	No					

Code:

```
LD    F0   0  R1
MULTI F4   F0 F2
SD    F4   0  R1
SUBI  R1   R1 #8
BNEZ  R1   Loop
```

Register result status

Clock	R1	F0	F2	F4	F6	F8	F10	F12	...	F30
0	80 Qi									

5 minute Class Break

- **Lecture Format:**
  - **≈ 1 minute: review last time & motivate this lecture**
  - **≈ 20 minute lecture**
  - **≈ 3 minutes: discuss class manangement**
  - **≈ 25 minutes: lecture**
  - **5 minutes: break**
  - **≈25 minutes: lecture**
  - **≈1 minute: summary of today's important topics**

Loop Example Cycle 1

Instruction status

Instruction	j	k	iteration	Issue	Execution complete	Write Result
LD F0	0	R1	1	1		
MULTI F4	F0	F2	1			
SD F4	0	R1	1			
LD F0	0	R1	2			
MULTI F4	F0	F2	2			
SD F4	0	R1	2			

	Busy	Address	Qi
Load1	Yes	80	
Load2	No		
Load3	No		
Store1	No		
Store2	No		
Store3	No		

Reservation Stations

Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for Qj	RS for Qk	Code:
0	Add1	No						LD F0 0 R1
0	Add2	No						MULTI F4 F0 F2
0	Add3	No						SD F4 0 R1
0	Mult1	No						SUBI R1 R1 #8
0	Mult2	No						BNEZ R1 Loop

Register result status

Clock	R1		F0	F2	F4	F6	F8	F10	F12 ...	F30
1	80	Qi	Load1							

Loop Example Cycle 2

Instruction status

Instruction	j	k	iteration	Issue	Execution complete	Write Result
LD F0	0	R1	1	1		
MULTI F4	F0	F2	1	2		
SD F4	0	R1	1			
LD F0	0	R1	2			
MULTI F4	F0	F2	2			
SD F4	0	R1	2			

	Busy	Address	Qi
Load1	Yes	80	
Load2	No		
Load3	No		
Store1	No		
Store2	No		
Store3	No		

Reservation Stations

Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for Qj	RS for Qk	Code:
0	Add1	No						LD F0 0 R1
0	Add2	No						MULTI F4 F0 F2
0	Add3	No						SD F4 0 R1
0	Mult1	Yes	MULTD		R(F2)	Load1		SUBI R1 R1 #8
0	Mult2	No						BNEZ R1 Loop

Register result status

Clock	R1		F0	F2	F4	F6	F8	F10	F12 ...	F30
2	80	Qi	Load1		Mult1					

Loop Example Cycle 3

Instruction status

Instruction	j	k	iteration	Issue	Execution complete	Write Result
LD F0	0	R1	1	1		
MULTI F4	F0	F2	1	2		
SD F4	0	R1	1	3		
LD F0	0	R1	2			
MULTI F4	F0	F2	2			
SD F4	0	R1	2			

	Busy	Address	Qi
Load1	Yes	80	
Load2	No		
Load3	No		
Store1	Yes	80	Mult1
Store2	No		
Store3	No		

Reservation Stations

Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for Qj	RS for Qk	Code:
0	Add1	No						LD F0 0 R1
0	Add2	No						MULTI F4 F0 F2
0	Add3	No						SD F4 0 R1
0	Mult1	Yes	MULTD		R(F2)	Load1		SUBI R1 R1 #8
0	Mult2	No						BNEZ R1 Loop

Register result status

Clock	R1		F0	F2	F4	F6	F8	F10	F12 ...	F30
3	80	Qi	Load1		Mult1					

Loop Example Cycle 4

Instruction status

Instruction	j	k	iteration	Issue	Execution complete	Write Result		Busy	Address	
LD F0	0	R1	1	1			Load1	Yes	80	
MULTI F4	F0	F2	1	2			Load2	No		
SD F4	0	R1	1	3			Load3	No		Qi
LD F0	0	R1	2				Store1	Yes	80	Mult1
MULTI F4	F0	F2	2				Store2	No		
SD F4	0	R1	2				Store3	No		

Reservation Stations

Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for j Qj	RS for k Qk	Code:			
0	Add1	No						LD	F0	0	R1
0	Add2	No						MULTI	F4	F0	F2
0	Add3	No						SD	F4	0	R1
0	Mult1	Yes	MULTD		R(F2)	Load1		SUBI	R1	R1	#8
0	Mult2	No						BNEZ	R1	Loop	

Register result status

Clock	R1		F0	F2	F4	F6	F8	F10	F12 ...	F30
4	72	Qi	Load1		Mult1					

Loop Example Cycle 5

Instruction status

Instruction	j	k	iteration	Issue	Execution complete	Write Result		Busy	Address	
LD F0	0	R1	1	1			Load1	Yes	80	
MULTI F4	F0	F2	1	2			Load2	No		
SD F4	0	R1	1	3			Load3	No		Qi
LD F0	0	R1	2				Store1	Yes	80	Mult1
MULTI F4	F0	F2	2				Store2	No		
SD F4	0	R1	2				Store3	No		

Reservation Stations

Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for j Qj	RS for k Qk	Code:			
0	Add1	No						LD	F0	0	R1
0	Add2	No						MULTI	F4	F0	F2
0	Add3	No						SD	F4	0	R1
0	Mult1	Yes	MULTD		R(F2)	Load1		SUBI	R1	R1	#8
0	Mult2	No						BNEZ	R1	Loop	

Register result status

Clock	R1		F0	F2	F4	F6	F8	F10	F12 ...	F30
5	72	Qi	Load1		Mult1					

Loop Example Cycle 6

Instruction status

Instruction	j	k	iteration	Issue	Execution complete	Write Result		Busy	Address	
LD F0	0	R1	1	1			Load1	Yes	80	
MULTI F4	F0	F2	1	2			Load2	Yes	72	
SD F4	0	R1	1	3			Load3	No		Qi
LD F0	0	R1	2	6			Store1	Yes	80	Mult1
MULTI F4	F0	F2	2				Store2	No		
SD F4	0	R1	2				Store3	No		

Reservation Stations

Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for j Qj	RS for k Qk	Code:			
0	Add1	No						LD	F0	0	R1
0	Add2	No						MULTI	F4	F0	F2
0	Add3	No						SD	F4	0	R1
0	Mult1	Yes	MULTD		R(F2)	Load1		SUBI	R1	R1	#8
0	Mult2	No						BNEZ	R1	Loop	

Register result status

Clock	R1		F0	F2	F4	F6	F8	F10	F12 ...	F30
6	72	Qi	Load2		Mult1					

Loop Example Cycle 7

Instruction status

Instruction	j	k	iteration	Issue	Execution complete	Write Result		Busy	Address	
LD F0	0	R1	1	1			Load1	Yes	80	
MULTI F4	F0	F2	1	2			Load2	Yes	72	
SD F4	0	R1	1	3			Load3	No		Qi
LD F0	0	R1	2	6			Store1	Yes	80	Mult1
MULTI F4	F0	F2	2	7			Store2	No		
SD F4	0	R1	2				Store3	No		

Reservation Stations

Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for j Qj	RS for k Qk	Code:			
0	Add1	No						LD	F0	0	R1
0	Add2	No						MULTI	F4	F0	F2
0	Add3	No						SD	F4	0	R1
0	Mult1	Yes	MULTD		R(F2)	Load1		SUBI	R1	R1	#8
0	Mult2	Yes	MULTD		R(F2)	Load2		BNEZ	R1	Loop	

Register result status

Clock	R1		F0	F2	F4	F6	F8	F10	F12 ...	F30
7	72	Qi	Load2		Mult2					

Loop Example Cycle 8

Instruction status

Instruction	j	k	iteration	Issue	Execution complete	Write Result
LD F0	0	R1	1	1		
MULTI F4	F0	F2	1	2		
SD F4	0	R1	1	3		
LD F0	0	R1	2	6		
MULTI F4	F0	F2	2	7		
SD F4	0	R1	2	8		

	Busy	Address	Qi
Load1	Yes	80	
Load2	Yes	72	
Load3	No		
Store1	Yes	80	Mult1
Store2	Yes	72	Mult2
Store3	No		

Reservation Stations

Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for Qj	RS for Qk	Code:			
0	Add1	No						LD	F0	0	R1
0	Add2	No						MULTI	F4	F0	F2
0	Add3	No						SD	F4	0	R1
0	Mult1	Yes	MULTD		R(F2)	Load1		SUBI	R1	R1	#8
0	Mult2	Yes	MULTD		R(F2)	Load2		BNEZ	R1	Loop	

Register result status

Clock	R1		F0	F2	F4	F6	F8	F10	F12 ...	F30
8	72	Qi	Load2		Mult2					

Loop Example Cycle 9

Instruction status

Instruction	j	k	iteration	Issue	Execution complete	Write Result
LD F0	0	R1	1	1	9	
MULTI F4	F0	F2	1	2		
SD F4	0	R1	1	3		
LD F0	0	R1	2	6		
MULTI F4	F0	F2	2	7		
SD F4	0	R1	2	8		

	Busy	Address	Qi
Load1	Yes	80	
Load2	Yes	72	
Load3	No		
Store1	Yes	80	Mult1
Store2	Yes	72	Mult2
Store3	No		

Reservation Stations

Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for Qj	RS for Qk	Code:			
0	Add1	No						LD	F0	0	R1
0	Add2	No						MULTI	F4	F0	F2
0	Add3	No						SD	F4	0	R1
0	Mult1	Yes	MULTD		R(F2)	Load1		SUBI	R1	R1	#8
0	Mult2	Yes	MULTD		R(F2)	Load2		BNEZ	R1	Loop	

Register result status

Clock	R1		F0	F2	F4	F6	F8	F10	F12 ...	F30
9	64	Qi	Load2		Mult2					

Loop Example Cycle 10

Instruction status

Instruction	j	k	iteration	Issue	Execution complete	Write Result
LD F0	0	R1	1	1	9	10
MULTI F4	F0	F2	1	2		
SD F4	0	R1	1	3		
LD F0	0	R1	2	6	10	
MULTI F4	F0	F2	2	7		
SD F4	0	R1	2	8		

	Busy	Address	Qi
Load1	No		
Load2	Yes	72	
Load3	No		
Store1	Yes	80	Mult1
Store2	Yes	72	Mult2
Store3	No		

Reservation Stations

Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for Qj	RS for Qk	Code:			
0	Add1	No						LD	F0	0	R1
0	Add2	No						MULTI	F4	F0	F2
0	Add3	No						SD	F4	0	R1
4	Mult1	Yes	MULTD	M(80)	R(F2)			SUBI	R1	R1	#8
0	Mult2	Yes	MULTD		R(F2)	Load2		BNEZ	R1	Loop	

Register result status

Clock	R1		F0	F2	F4	F6	F8	F10	F12 ...	F30
10	64	Qi	Load2		Mult2					

Loop Example Cycle 11

Instruction status

Instruction	j	k	iteration	Issue	Execution complete	Write Result
LD F0	0	R1	1	1	9	10
MULTI F4	F0	F2	1	2		
SD F4	0	R1	1	3		
LD F0	0	R1	2	6	10	11
MULTI F4	F0	F2	2	7		
SD F4	0	R1	2	8		

	Busy	Address	Qi
Load1	No		
Load2	No		
Load3	Yes	64	Qi
Store1	Yes	80	Mult1
Store2	Yes	72	Mult2
Store3	No		

Reservation Stations

Time	Name	Busy	Op	S1 Vj	S2 Vk	RS for Qj	RS for Qk	Code:			
0	Add1	No						LD	F0	0	R1
0	Add2	No						MULTI	F4	F0	F2
0	Add3	No						SD	F4	0	R1
3	Mult1	Yes	MULTD	M(80)	R(F2)			SUBI	R1	R1	#8
4	Mult2	Yes	MULTD	M(72)	R(F2)			BNEZ	R1	Loop	

Register result status

Clock	R1		F0	F2	F4	F6	F8	F10	F12 ...	F30
11	64	Qi	Load3		Mult2					

Loop Example Cycle 12

Instruction status					Executio	Write			
Instruction	j	k	iteration	Issue	complete	Result		Busy	Address
LD F0	0	R1	1	1	9	10	Load1	No	
MULTI F4	F0	F2	1	2			Load2	No	
SD F4	0	R1	1	3			Load3	Yes	64 Qi
LD F0	0	R1	2	6	10	11	Store1	Yes	80 Mult1
MULTI F4	F0	F2	2	7			Store2	Yes	72 Mult2
SD F4	0	R1	2	8			Store3	No	

Reservation Stations				S1	S2	RS for	RS for k		
Time	Name	Busy	Op	Vj	Vk	Qj	Qk	Code:	
0	Add1	No						LD F0	0 R1
0	Add2	No						MULTI F4	F0 F2
0	Add3	No						SD F4	0 R1
2	Mult1	Yes	MULTD	M(80)	R(F2)			SUBI R1	R1 #8
3	Mult2	Yes	MULTD	M(72)	R(F2)			BNEZ R1	Loop

Register result status

Clock	R1	F0	F2	F4	F6	F8	F10	F12 ...	F30
12	64 Qi	Load3		Mult2					

Loop Example Cycle 13

Instruction status					Executio	Write			
Instruction	j	k	iteration	Issue	complete	Result		Busy	Address
LD F0	0	R1	1	1	9	10	Load1	No	
MULTI F4	F0	F2	1	2			Load2	No	
SD F4	0	R1	1	3			Load3	Yes	64 Qi
LD F0	0	R1	2	6	10	11	Store1	Yes	80 Mult1
MULTI F4	F0	F2	2	7			Store2	Yes	72 Mult2
SD F4	0	R1	2	8			Store3	No	

Reservation Stations				S1	S2	RS for	RS for k		
Time	Name	Busy	Op	Vj	Vk	Qj	Qk	Code:	
0	Add1	No						LD F0	0 R1
0	Add2	No						MULTI F4	F0 F2
0	Add3	No						SD F4	0 R1
1	Mult1	Yes	MULTD	M(80)	R(F2)			SUBI R1	R1 #8
2	Mult2	Yes	MULTD	M(72)	R(F2)			BNEZ R1	Loop

Register result status

Clock	R1	F0	F2	F4	F6	F8	F10	F12 ...	F30
13	64 Qi	Load3		Mult2					

Loop Example Cycle 14

Instruction status					Executio	Write			
Instruction	j	k	iteration	Issue	complete	Result		Busy	Address
LD F0	0	R1	1	1	9	10	Load1	No	
MULTI F4	F0	F2	1	2	14		Load2	No	
SD F4	0	R1	1	3			Load3	Yes	64 Qi
LD F0	0	R1	2	6	10	11	Store1	Yes	80 Mult1
MULTI F4	F0	F2	2	7			Store2	Yes	72 Mult2
SD F4	0	R1	2	8			Store3	No	

Reservation Stations				S1	S2	RS for	RS for k		
Time	Name	Busy	Op	Vj	Vk	Qj	Qk	Code:	
0	Add1	No						LD F0	0 R1
0	Add2	No						MULTI F4	F0 F2
0	Add3	No						SD F4	0 R1
0	Mult1	Yes	MULTD	M(80)	R(F2)			SUBI R1	R1 #8
1	Mult2	Yes	MULTD	M(72)	R(F2)			BNEZ R1	Loop

Register result status

Clock	R1	F0	F2	F4	F6	F8	F10	F12 ...	F30
14	64 Qi	Load3		Mult2					

Loop Example Cycle 15

Instruction status					Executio	Write			
Instruction	j	k	iteration	Issue	complete	Result		Busy	Address
LD F0	0	R1	1	1	9	10	Load1	No	
MULTI F4	F0	F2	1	2	14	15	Load2	No	
SD F4	0	R1	1	3			Load3	Yes	64 Qi
LD F0	0	R1	2	6	10	11	Store1	Yes	80 M(80)*R(
MULTI F4	F0	F2	2	7	15		Store2	Yes	72 Mult2
SD F4	0	R1	2	8			Store3	No	

Reservation Stations				S1	S2	RS for	RS for k		
Time	Name	Busy	Op	Vj	Vk	Qj	Qk	Code:	
0	Add1	No						LD F0	0 R1
0	Add2	No						MULTI F4	F0 F2
0	Add3	No						SD F4	0 R1
0	Mult1	No						SUBI R1	R1 #8
0	Mult2	Yes	MULTD	M(72)	R(F2)			BNEZ R1	Loop

Register result status

Clock	R1	F0	F2	F4	F6	F8	F10	F12 ...	F30
15	64 Qi	Load3		Mult2					

- **Update to regsiter F0?**

Loop Example Cycle 16

Instruction status					Executio	Write				
Instruction	j	k	iteration	Issue	complete	Result		Busy	Address	
LD F0	0	R1	1	1	9	10	Load1	No		
MULTI F4	F0	F2	1	2	14	15	Load2	No		
SD F4	0	R1	1	3			Load3	Yes	64	Qi
LD F0	0	R1	2	6	10	11	Store1	Yes	80	M(80)*R(
MULTI F4	F0	F2	2	7	15	16	Store2	Yes	72	M(72)*R(
SD F4	0	R1	2	8			Store3	No		

Reservation Stations				S1	S2	RS for	RS for k		
Time	Name	Busy	Op	Vj	Vk	Qj	Qk	Code:	
0	Add1	No						LD F0	0 R1
0	Add2	No						MULTI F4	F0 F2
0	Add3	No						SD F4	0 R1
0	Mult1	Yes	MULTD		R(F2)	Load3		SUBI R1	R1 #8
0	Mult2	No						BNEZ R1	Loop

Register result status

Clock	R1	F0	F2	F4	F6	F8	F10	F12	...	F30
16	64	Qi	Load3		Mult1					

Loop Example Cycle 17

Instruction status					Executio	Write				
Instruction	j	k	iteration	Issue	complete	Result		Busy	Address	
LD F0	0	R1	1	1	9	10	Load1	No		
MULTI F4	F0	F2	1	2	14	15	Load2	No		
SD F4	0	R1	1	3			Load3	Yes	64	Qi
LD F0	0	R1	2	6	10	11	Store1	Yes	80	M(80)*R(
MULTI F4	F0	F2	2	7	15	16	Store2	Yes	72	M(72)*R(
SD F4	0	R1	2	8			Store3	Yes	64	Mult1

Reservation Stations				S1	S2	RS for	RS for k		
Time	Name	Busy	Op	Vj	Vk	Qj	Qk	Code:	
0	Add1	No						LD F0	0 R1
0	Add2	No						MULTI F4	F0 F2
0	Add3	No						SD F4	0 R1
0	Mult1	Yes	MULTD		R(F2)	Load3		SUBI R1	R1 #8
0	Mult2	No						BNEZ R1	Loop

Register result status

Clock	R1	F0	F2	F4	F6	F8	F10	F12	...	F30
17	64	Qi	Load3		Mult1					

Loop Example Cycle 18

Instruction status					Executio	Write				
Instruction	j	k	iteration	Issue	complete	Result		Busy	Address	
LD F0	0	R1	1	1	9	10	Load1	No		
MULTI F4	F0	F2	1	2	14	15	Load2	No		
SD F4	0	R1	1	3	18		Load3	Yes	64	Qi
LD F0	0	R1	2	6	10	11	Store1	Yes	80	M(80)*R(
MULTI F4	F0	F2	2	7	15	16	Store2	Yes	72	M(72)*R(
SD F4	0	R1	2	8			Store3	Yes	64	Mult1

Reservation Stations				S1	S2	RS for	RS for k		
Time	Name	Busy	Op	Vj	Vk	Qj	Qk	Code:	
0	Add1	No						LD F0	0 R1
0	Add2	No						MULTI F4	F0 F2
0	Add3	No						SD F4	0 R1
0	Mult1	Yes	MULTD		R(F2)	Load3		SUBI R1	R1 #8
0	Mult2	No						BNEZ R1	Loop

Register result status

Clock	R1	F0	F2	F4	F6	F8	F10	F12	...	F30
18	56	Qi	Load3		Mult1					

Loop Example Cycle 19

Instruction status					Executio	Write				
Instruction	j	k	iteration	Issue	complete	Result		Busy	Address	
LD F0	0	R1	1	1	9	10	Load1	No		
MULTI F4	F0	F2	1	2	14	15	Load2	No		
SD F4	0	R1	1	3	18	19	Load3	Yes	64	Qi
LD F0	0	R1	2	6	10	11	Store1	No		
MULTI F4	F0	F2	2	7	15	16	Store2	Yes	72	M(72)*R(
SD F4	0	R1	2	8			Store3	Yes	64	Mult1

Reservation Stations				S1	S2	RS for	RS for k		
Time	Name	Busy	Op	Vj	Vk	Qj	Qk	Code:	
0	Add1	No						LD F0	0 R1
0	Add2	No						MULTI F4	F0 F2
0	Add3	No						SD F4	0 R1
0	Mult1	Yes	MULTD		R(F2)	Load3		SUBI R1	R1 #8
0	Mult2	No						BNEZ R1	Loop

Register result status

Clock	R1	F0	F2	F4	F6	F8	F10	F12	...	F30
19	56	Qi	Load3		Mult1					

Loop Example Cycle 20

Instruction status					Execution	Write			
Instruction	j	k	iteration	Issue	complete	Result		Busy	Address	
LD F0	0	R1	1	1	9	10	Load1	No		
MULTI F4	F0	F2	1	2	14	15	Load2	No		
SD F4	0	R1	1	3	18	19	Load3	Yes	64	Qi
LD F0	0	R1	2	6	10	11	Store1	No		
MULTI F4	F0	F2	2	7	15	16	Store2	Yes	72	M(72)*R(
SD F4	0	R1	2	8	20		Store3	Yes	64	Mult1

Reservation Stations			S1	S2	RS for j	RS for k		
Time	Name	Busy	Op	Vj	Vk	Qj	Qk	Code:
0	Add1	No						LD F0 0 R1
0	Add2	No						MULTI F4 F0 F2
0	Add3	No						SD F4 0 R1
0	Mult1	Yes	MULTD		R(F2)	Load3		SUBI R1 R1 #8
0	Mult2	No						BNEZ R1 Loop

Register result status										
Clock	R1	F0	F2	F4	F6	F8	F10	F12	...	F30
20	56	Qi	Load3		Mult1					

Loop Example Cycle 21

Instruction status					Execution	Write			
Instruction	j	k	iteration	Issue	complete	Result		Busy	Address	
LD F0	0	R1	1	1	9	10	Load1	No		
MULTI F4	F0	F2	1	2	14	15	Load2	No		
SD F4	0	R1	1	3	18	19	Load3	Yes	64	Qi
LD F0	0	R1	2	6	10	11	Store1	No		
MULTI F4	F0	F2	2	7	15	16	Store2	No		
SD F4	0	R1	2	8	20	21	Store3	Yes	64	Mult1

Reservation Stations			S1	S2	RS for j	RS for k		
Time	Name	Busy	Op	Vj	Vk	Qj	Qk	Code:
0	Add1	No						LD F0 0 R1
0	Add2	No						MULTI F4 F0 F2
0	Add3	No						SD F4 0 R1
0	Mult1	Yes	MULTD		R(F2)	Load3		SUBI R1 R1 #8
0	Mult2	No						BNEZ R1 Loop

Register result status										
Clock	R1	F0	F2	F4	F6	F8	F10	F12	...	F30
21	56	Qi	Load3		Mult1					

Tomasulo Summary

- **Prevents Register as bottleneck**
- **Avoids WAR, WAW hazards of Scoreboard**
- **Allows loop unrolling in HW**
- **Not limited to basic blocks (provided branch prediction)**
- **Lasting Contributions**
  - **Dynamic scheduling**
  - **Register renaming**
  - **Load/store disambiguation**
- **360/91 descendants are PowerPC 604, 620; MIPS R10000; HP-PA 8000; Intel Pentium Pro**