
CMSC 411
Computer Systems Architecture
Lecture 15
Instruction Level Parallelism (cont.)

Alan Sussman
als@cs.umd.edu

Administrivia

- Finish reading Chapter 3 of H&P
- HW#3 due today

CMSC 411 - 15 (some from Patterson, Sussman, others)

2

Outline

- ILP
- Compiler techniques to increase ILP
- Loop Unrolling
- Static Branch Prediction
- Dynamic Branch Prediction
- Overcoming Data Hazards with Dynamic Scheduling
- Tomasulo Algorithm
- Conclusion

CMSC 411 - 15 (some from Patterson, Sussman, others)

3

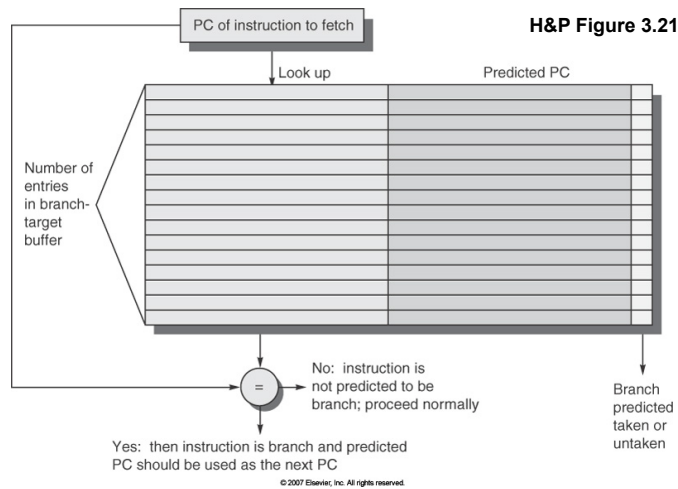
Branch Target Buffers (BTB)

- Branch target calculation is costly and stalls the instruction fetch.
- BTB stores PCs the same way as caches
- The PC of a branch is sent to the BTB
- When a match is found the corresponding Predicted PC is returned
- If the branch was predicted taken, instruction fetch continues at the returned predicted PC

CMSC 411 - 15 (some from Patterson, Sussman, others)

4

Branch Target Buffers



CMSC 411 - 15 (some from Patterson, Sussman, others)

5

Dynamic Branch Prediction Summary

- Prediction becoming important part of execution
- Branch History Table: 2 bits for loop accuracy
- Correlation: Recently executed branches correlated with next branch
 - Either different branches (GA)
 - Or different executions of same branches (PA)
- Tournament predictors take insight to next level, by using multiple predictors
 - usually one based on global information and one based on local information, and combining them with a selector
 - In 2006, tournament predictors using $\approx 30K$ bits are in processors like the Power5 and Pentium 4
- Branch Target Buffer: include branch address & prediction

CMSC 411 - 15 (some from Patterson, Sussman, others)

6

Outline

- ILP
- Compiler techniques to increase ILP
- Loop Unrolling
- Static Branch Prediction
- Dynamic Branch Prediction
- Overcoming Data Hazards with Dynamic Scheduling
- Tomasulo Algorithm
- Conclusion

CMSC 411 - 15 (some from Patterson, Sussman, others)

7

Advantages of Dynamic Scheduling

- **Dynamic scheduling** - hardware rearranges the instruction execution to reduce stalls while maintaining data flow and exception behavior
- Handles cases when dependences unknown at compile time
 - it allows the processor to tolerate unpredictable delays such as cache misses, by executing other code while waiting for the miss to resolve
- Allows code that compiled for one pipeline to run efficiently on a different pipeline
- Simplifies the compiler
- **Hardware speculation**, a technique with significant performance advantages, builds on dynamic scheduling (later)

CMSC 411 - 15 (some from Patterson, Sussman, others)

8

HW Schemes: Instruction Parallelism

- Key idea: Allow instructions behind stall to proceed

```
DIVD  F0, F2, F4
ADDD  F10, F0, F8
SUBD  F12, F8, F14
```
- Enables [out-of-order execution](#) and allows [out-of-order completion](#) (e.g., `SUBD`)
 - In a dynamically scheduled pipeline, all instructions still pass through issue stage in order ([in-order issue](#))
- Will distinguish when an instruction [begins execution](#) and when it [completes execution](#); between 2 times, the instruction is [in execution](#)
- Note: Dynamic execution creates WAR and WAW hazards and makes exceptions harder

CMSC 411 - 15 (some from Patterson, Sussman, others)

9

Dynamic Scheduling Step 1

- Simple pipeline had 1 stage to check both structural and data hazards: Instruction Decode (ID), also called Instruction Issue
- Split the ID pipe stage of simple 5-stage pipeline into 2 stages:
- [Issue](#)—Decode instructions, check for structural hazards
- [Read operands](#)—Wait until no data hazards, then read operands

CMSC 411 - 15 (some from Patterson, Sussman, others)

10

A Dynamic Algorithm: Tomasulo's

- For IBM 360/91 (before caches!)
 - ⇒ Long memory latency
- Goal: High Performance without special compilers
- Small number of floating point registers (4 in 360) prevented interesting compiler scheduling of operations
 - This led Tomasulo to try to figure out how to get more effective registers — [renaming in hardware!](#)
- Why Study 1966 Computer?
 - The descendants of this have flourished!
 - » Alpha 21264, Pentium 4, AMD Opteron, Power 5, ...

CMSC 411 - 15 (some from Patterson, Sussman, others)

11

Tomasulo Algorithm

- Control & buffers [distributed](#) with Function Units (FU)
 - FU buffers called "[reservation stations](#)"; have pending operands
- Registers in instructions replaced by values or pointers to reservation stations (RS); called [register renaming](#) ;
 - Renaming avoids WAR, WAW hazards
 - More reservation stations than registers, so can do optimizations compilers can't
- Results to FU from RS, [not through registers](#), over [Common Data Bus](#) that broadcasts results to all FUs
 - Avoids RAW hazards by executing an instruction only when its operands are available
- Load and Stores treated as FUs with RSs as well
- Integer instructions can go past branches (use branch prediction), also allow FP ops beyond basic block in FP queue

CMSC 411 - 15 (some from Patterson, Sussman, others)

12

Outline

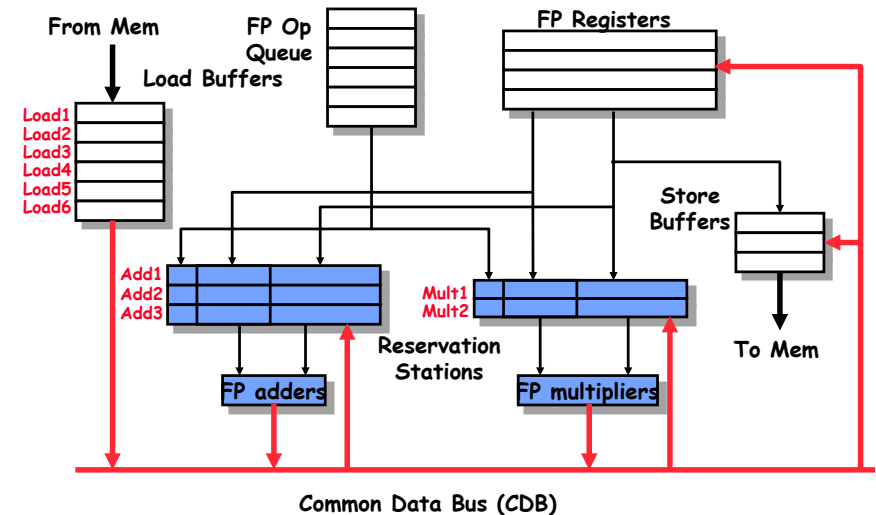
- ILP
- Compiler techniques to increase ILP
- Loop Unrolling
- Static Branch Prediction
- Dynamic Branch Prediction
- Overcoming Data Hazards with Dynamic Scheduling
- Tomasulo Algorithm
- Conclusion

CMSC 411 - 15 (some from Patterson, Sussman, others)

13

Tomasulo Organization

From H&P
Figure 3.6



CMSC 411 - 15 (some from Patterson, Sussman, others)

14

Reservation Station Components

Op: Operation to perform in the unit (e.g., + or −)

Vj, Vk: **Value** of Source operands

- Store buffers have V field, result to be stored

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

- Note: Qj, Qk=0 => ready
- Store buffers only have Qi for RS producing result

Busy: Indicates reservation station or FU is busy

Also:

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

CMSC 411 - 15 (some from Patterson, Sussman, others)

15

Three Stages of Tomasulo Algorithm

1. **Issue**—get instruction from FP Op Queue
If reservation station free (no structural hazard), control issues instr & sends operands (renames registers).
2. **Execute**—operate on operands (EX)
When both operands ready then execute;
if not ready, watch Common Data Bus for result
3. **Write result**—finish execution (WB)
Write on Common Data Bus to all awaiting units;
mark reservation station available

- Normal data bus: data + destination ("go to" bus)
- **Common data bus:** data + **source** ("come from" bus)
 - 64 bits of data + 4 bits of Functional Unit **source** address
 - Write if matches expected Functional Unit (produces result)
 - Does the broadcast

CMSC 411 - 15 (some from Patterson, Sussman, others)

16

Tomasulo Example

Instruction status:

Instruction	j	k	Issue	Exec	Write	Comp	Result
LD	F6	34+	R2				
LD	F2	45+	R3				
MULTD	F0	F2	F4				
SUBD	F8	F6	F2				
DIVD	F10	F0	F6				
ADDD	F6	F8	F2				

	Busy	Address
Load1	No	
Load2	No	
Load3	No	

3 Load/Buffers

Reservation Stations:

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

3 FP Adder R.S.
2 FP Mult R.S.

Register result status:

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

0
Clock cycle counter

CISC 411 - 15 (some from Patterson, Sussman, others)

17

Tomasulo Example Cycle 1

Instruction status:

Instruction	j	k	Issue	Exec	Write	Comp	Result
LD	F6	34+	R2				
LD	F2	45+	R3				
MULTD	F0	F2	F4				
SUBD	F8	F6	F2				
DIVD	F10	F0	F6				
ADDD	F6	F8	F2				

	Busy	Address
Load1	Yes	34+R2
Load2	No	
Load3	No	

Reservation Stations:

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

Register result status:

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

CISC 411 - 15 (some from Patterson, Sussman, others)

18

Tomasulo Example Cycle 2

Instruction status:

Instruction	j	k	Issue	Exec	Write	Comp	Result
LD	F6	34+	R2				
LD	F2	45+	R3				
MULTD	F0	F2	F4				
SUBD	F8	F6	F2				
DIVD	F10	F0	F6				
ADDD	F6	F8	F2				

	Busy	Address
Load1	Yes	34+R2
Load2	Yes	45+R3
Load3	No	

Reservation Stations:

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

Register result status:

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

Note: Can have multiple loads outstanding

CISC 411 - 15 (some from Patterson, Sussman, others)

19

Tomasulo Example Cycle 3

Instruction status:

Instruction	j	k	Issue	Exec	Write	Comp	Result
LD	F6	34+	R2				
LD	F2	45+	R3				
MULTD	F0	F2	F4				
SUBD	F8	F6	F2				
DIVD	F10	F0	F6				
ADDD	F6	F8	F2				

	Busy	Address
Load1	Yes	34+R2
Load2	Yes	45+R3
Load3	No	

Reservation Stations:

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

Register result status:

Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
3	Mult1	Load2					Load1		

- Note: registers names are removed ("renamed") in Reservation Stations; MULT issued**
- Load1 completing; what is waiting for Load1?**

CISC 411 - 15 (some from Patterson, Sussman, others)

20

Tomasulo Example Cycle 4

M(A1) result of first load...

Instruction status:

Instruction	j	k	Exec Write			Busy	Address
			Issue	Comp	Result		
LD	F6	34+	1	3	4	No	
LD	F2	45+	2	4		No	
MULTD	F0	F2	3			No	
SUBD	F8	F6	4			No	
DIVD	F10	F0				No	
ADDD	F6	F8				No	

Reservation Stations:

Time	Name	Busy	Op	Vj	Vk	Qj	Qk
Add1	Yes	SUBD	M(A1)				Load2
Add2	No						
Add3	No						
Mult1	Yes	MULTD			R(F4)		Load2
Mult2	No						

Register result status:

Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
4		Mult1	Load2		M(A1)	Add1			

- Load2 completing; what is waiting for Load2?

CMSC 411 - 15 (some from Patterson, Sussman, others)

21

Tomasulo Example Cycle 5

Instruction status:

Instruction	j	k	Exec Write			Busy	Address
			Issue	Comp	Result		
LD	F6	34+	1	3	4	No	
LD	F2	45+	2	4		No	
MULTD	F0	F2	3			No	
SUBD	F8	F6	4			No	
DIVD	F10	F0	5			No	
ADDD	F6	F8				No	

Reservation Stations:

Time	Name	Busy	Op	Vj	Vk	Qj	Qk
2 Add1	Yes	SUBD	M(A1)		M(A2)		
Add2	No						
Add3	No						
10 Mult1	Yes	MULTD			R(F4)		
Mult2	Yes	DIVD			M(A1)		Mult1

Register result status:

Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
5		Mult1	M(A2)		M(A1)	Add1	Mult2		

- Timer starts down for Add1, Mult1

CMSC 411 - 15 (some from Patterson, Sussman, others)

22

Tomasulo Example Cycle 6

Instruction status:

Instruction	j	k	Exec Write			Busy	Address
			Issue	Comp	Result		
LD	F6	34+	1	3	4	No	
LD	F2	45+	2	4		No	
MULTD	F0	F2	3			No	
SUBD	F8	F6	4			No	
DIVD	F10	F0	5			No	
ADDD	F6	F8	6			No	

Reservation Stations:

Time	Name	Busy	Op	Vj	Vk	Qj	Qk
1 Add1	Yes	SUBD	M(A1)		M(A2)		
Add2	Yes	ADDD			M(A2)		Add1
Add3	No						
9 Mult1	Yes	MULTD			R(F4)		
Mult2	Yes	DIVD			M(A1)		Mult1

Register result status:

Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
6		Mult1	M(A2)		Add2	Add1	Mult2		

- Issue ADDD here despite name dependency on F6?

CMSC 411 - 15 (some from Patterson, Sussman, others)

23

Tomasulo Example Cycle 7

Instruction status:

Instruction	j	k	Exec Write			Busy	Address
			Issue	Comp	Result		
LD	F6	34+	1	3	4	No	
LD	F2	45+	2	4		No	
MULTD	F0	F2	3			No	
SUBD	F8	F6	4			No	
DIVD	F10	F0	5			No	
ADDD	F6	F8	6			No	

Reservation Stations:

Time	Name	Busy	Op	Vj	Vk	Qj	Qk
0 Add1	Yes	SUBD	M(A1)		M(A2)		
Add2	Yes	ADDD			M(A2)		Add1
Add3	No						
8 Mult1	Yes	MULTD			R(F4)		
Mult2	Yes	DIVD			M(A1)		Mult1

Register result status:

Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
7		Mult1	M(A2)		Add2	Add1	Mult2		

- Add1 (SUBD) completing; what is waiting for it?

CMSC 411 - 15 (some from Patterson, Sussman, others)

24

Tomasulo Example Cycle 8

Instruction status:

Instruction	j	k	Exec Write			Busy	Address	
			Issue	Comp	Result			
LD	F6	34+	R2	1	3	4	Load1	No
LD	F2	45+	R3	2	4	5	Load2	No
MULTD	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	S2	RS	RS
				Vj	Vk	Qj	Qk
Add1	No						
2 Add2	Yes	ADDD	(M-M)	M(A2)			
Add3	No						
7 Mult1	Yes	MULTD	M(A2)	R(F4)			
Mult2	Yes	DIVD		M(A1)	Mult1		

Register result status:

Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
8	PU	Mult1	M(A2)	Add2	(M-M)	Mult2			

CISC 411 - 15 (some from Patterson, Sussman, others)

25

Tomasulo Example Cycle 9

Instruction status:

Instruction	j	k	Exec Write			Busy	Address	
			Issue	Comp	Result			
LD	F6	34+	R2	1	3	4	Load1	No
LD	F2	45+	R3	2	4	5	Load2	No
MULTD	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	S2	RS	RS
				Vj	Vk	Qj	Qk
Add1	No						
1 Add2	Yes	ADDD	(M-M)	M(A2)			
Add3	No						
6 Mult1	Yes	MULTD	M(A2)	R(F4)			
Mult2	Yes	DIVD		M(A1)	Mult1		

Register result status:

Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
9	PU	Mult1	M(A2)	Add2	(M-M)	Mult2			

CISC 411 - 15 (some from Patterson, Sussman, others)

26

Tomasulo Example Cycle 10

Instruction status:

Instruction	j	k	Exec Write			Busy	Address
			Issue	Comp	Result		
LD	F6	34+	R2	1	3	4	Load1
LD	F2	45+	R3	2	4	5	Load2
MULTD	F0	F2	F4	3			Load3
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	S2	RS	RS
				Vj	Vk	Qj	Qk
Add1	No						
0 Add2	Yes	ADDD	(M-M)	M(A2)			
Add3	No						
5 Mult1	Yes	MULTD	M(A2)	R(F4)			
Mult2	Yes	DIVD		M(A1)	Mult1		

Register result status:

Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
10	PU	Mult1	M(A2)	Add2	(M-M)	Mult2			

CISC 411 - 15 (some from Patterson, Sussman, others)

27

- Add2 (ADDD) completing; what is waiting for it?

Tomasulo Example Cycle 11

Instruction status:

Instruction	j	k	Exec Write			Busy	Address	
			Issue	Comp	Result			
LD	F6	34+	R2	1	3	4	Load1	No
LD	F2	45+	R3	2	4	5	Load2	No
MULTD	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	S2	RS	RS
				Vj	Vk	Qj	Qk
Add1	No						
Add2	No						
Add3	No						
4 Mult1	Yes	MULTD	M(A2)	R(F4)			
Mult2	Yes	DIVD		M(A1)	Mult1		

Register result status:

Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
11	PU	Mult1	M(A2)	(M-M+M)	(M-M)	Mult2			

CISC 411 - 15 (some from Patterson, Sussman, others)

28

- Write result of ADDD here?
- All quick instructions complete in this cycle!

Tomasulo Example Cycle 12

Instruction status:

Instruction	j	k	Exec Write			Busy	Address
			Issue	Comp	Result		
LD	F6	34+	R2	1	3	4	Load1
LD	F2	45+	R3	2	4	5	Load2
MULTD	F0	F2	F4	3			Load3
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	Vj	Vk	Qj	Qk
Add1	No						
Add2	No						
Add3	No						
3 Mult1	Yes		MULTD	M(A2)	R(F4)		
Mult2	Yes		DIVD		M(A1)	Mult1	

Register result status:

Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
12	PU	Mult1	M(A2)		(M-M+N(M-M))	Mult2			

CISC 411 - 15 (some from Patterson, Sussman, others)

29

Tomasulo Example Cycle 13

Instruction status:

Instruction	j	k	Exec Write			Busy	Address
			Issue	Comp	Result		
LD	F6	34+	R2	1	3	4	Load1
LD	F2	45+	R3	2	4	5	Load2
MULTD	F0	F2	F4	3			Load3
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	Vj	Vk	Qj	Qk
Add1	No						
Add2	No						
Add3	No						
2 Mult1	Yes		MULTD	M(A2)	R(F4)		
Mult2	Yes		DIVD		M(A1)	Mult1	

Register result status:

Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
13	PU	Mult1	M(A2)		(M-M+N(M-M))	Mult2			

CISC 411 - 15 (some from Patterson, Sussman, others)

30

Tomasulo Example Cycle 14

Instruction status:

Instruction	j	k	Exec Write			Busy	Address
			Issue	Comp	Result		
LD	F6	34+	R2	1	3	4	Load1
LD	F2	45+	R3	2	4	5	Load2
MULTD	F0	F2	F4	3			Load3
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	Vj	Vk	Qj	Qk
Add1	No						
Add2	No						
Add3	No						
1 Mult1	Yes		MULTD	M(A2)	R(F4)		
Mult2	Yes		DIVD		M(A1)	Mult1	

Register result status:

Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
14	PU	Mult1	M(A2)		(M-M+N(M-M))	Mult2			

CISC 411 - 15 (some from Patterson, Sussman, others)

31

Tomasulo Example Cycle 15

Instruction status:

Instruction	j	k	Exec Write			Busy	Address
			Issue	Comp	Result		
LD	F6	34+	R2	1	3	4	Load1
LD	F2	45+	R3	2	4	5	Load2
MULTD	F0	F2	F4	3	15		Load3
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	Vj	Vk	Qj	Qk
Add1	No						
Add2	No						
Add3	No						
0 Mult1	Yes		MULTD	M(A2)	R(F4)		
Mult2	Yes		DIVD		M(A1)	Mult1	

Register result status:

Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
15	PU	Mult1	M(A2)		(M-M+N(M-M))	Mult2			

CISC 411 - 15 (some from Patterson, Sussman, others)

32

- Mult1 (MULTD) completing; what is waiting for it?

Tomasulo Example Cycle 16

Instruction status:

Instruction	<i>j</i>	<i>k</i>	Exec Write			Busy	Address
			Issue	Comp	Result		
LD	F6	34+	R2	1	3	4	Load1
LD	F2	45+	R3	2	4	5	Load2
MULTD	F0	F2	F4	3	15	16	Load3
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	<i>S1</i> <i>Vj</i>	<i>S2</i> <i>Vk</i>	<i>RS</i> <i>Qj</i>	<i>RS</i> <i>Qk</i>
Add1		No					
Add2		No					
Add3		No					
Mult1		No					
40 Mult2		Yes	DIVD	M*F4	M(A1)		

Register result status:

Clock	<i>F0</i>	<i>F2</i>	<i>F4</i>	<i>F6</i>	<i>F8</i>	<i>F10</i>	<i>F12</i>	...	<i>F30</i>
16	FU	M*F4	M(A2)	(M-M+N(M-M))	Mult2				

- Just waiting for Mult2 (DIVD) to complete

CISC 411 - 15 (some from Patterson, Sussman, others)

33

Faster than light computation
(skip a few cycles)

CISC 411 - 15 (some from Patterson, Sussman, others)

34

Tomasulo Example Cycle 55

Instruction status:

Instruction	<i>j</i>	<i>k</i>	Exec Write			Busy	Address
			Issue	Comp	Result		
LD	F6	34+	R2	1	3	4	Load1
LD	F2	45+	R3	2	4	5	Load2
MULTD	F0	F2	F4	3	15	16	Load3
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	<i>S1</i> <i>Vj</i>	<i>S2</i> <i>Vk</i>	<i>RS</i> <i>Qj</i>	<i>RS</i> <i>Qk</i>
Add1		No					
Add2		No					
Add3		No					
Mult1		No					
1 Mult2		Yes	DIVD	M*F4	M(A1)		

Register result status:

Clock	<i>F0</i>	<i>F2</i>	<i>F4</i>	<i>F6</i>	<i>F8</i>	<i>F10</i>	<i>F12</i>	...	<i>F30</i>
55	FU	M*F4	M(A2)	(M-M+N(M-M))	Mult2				

CISC 411 - 15 (some from Patterson, Sussman, others)

35

Tomasulo Example Cycle 56

Instruction status:

Instruction	<i>j</i>	<i>k</i>	Exec Write			Busy	Address
			Issue	Comp	Result		
LD	F6	34+	R2	1	3	4	Load1
LD	F2	45+	R3	2	4	5	Load2
MULTD	F0	F2	F4	3	15	16	Load3
SUBD	F8	F6	F2	4	7	8	
DIVD	F10	F0	F6	5	56		
ADDD	F6	F8	F2	6	10	11	

Reservation Stations:

Time	Name	Busy	Op	<i>S1</i> <i>Vj</i>	<i>S2</i> <i>Vk</i>	<i>RS</i> <i>Qj</i>	<i>RS</i> <i>Qk</i>
Add1		No					
Add2		No					
Add3		No					
Mult1		No					
0 Mult2		Yes	DIVD	M*F4	M(A1)		

Register result status:

Clock	<i>F0</i>	<i>F2</i>	<i>F4</i>	<i>F6</i>	<i>F8</i>	<i>F10</i>	<i>F12</i>	...	<i>F30</i>
56	FU	M*F4	M(A2)	(M-M+N(M-M))	Mult2				

CISC 411 - 15 (some from Patterson, Sussman, others)

36

- Mult2 (DIVD) is completing; what is waiting for it?

Tomasulo Example Cycle 57

Instruction status:

Instruction	<i>j</i>	<i>k</i>		Exec		Write	Busy	Address
				Issue	Comp	Result		
LD	F6	34+	R2	1	3	4	Load1	No
LD	F2	45+	R3	2	4	5	Load2	No
MULTD	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	<i>S1</i>		<i>S2</i>		<i>RS</i>	
				<i>Vj</i>	<i>Vk</i>	<i>Qj</i>	<i>Qk</i>		
Add1		No							
Add2		No							
Add3		No							
Mult1		No							
Mult2		Yes	DIVD	M*F4	M(A1)				

Register result status:

Clock	<i>F0</i>	<i>F2</i>	<i>F4</i>	<i>F6</i>	<i>F8</i>	<i>F10</i>	<i>F12</i>	...	<i>F30</i>
56	FU M*F4 M(A2) (M-M+N(M-M)) Result								

- Once again: In-order issue, out-of-order execution and out-of-order completion.