



西安交通大学  
XI'AN JIAOTONG UNIVERSITY

# Simulation and Analysis of Dynamic Instruction Scheduling Algorithms

Yankai Jiang  
2020.4



# contents



**1. Introduction**

**2. Related Work**

**3. Demo Description**





西安交通大学  
XI'AN JIAOTONG UNIVERSITY

# Introduction 01





### Dynamic vs. Static Scheduling

Data hazards in a program cause a processor to stall.

- With static scheduling the **compiler** tries to reorder these instructions during **compile time** to reduce pipeline stalls.
  - Uses less hardware
  - Can use more powerful algorithms
- With dynamic scheduling the **hardware** tries to rearrange the instructions during **run-time** to reduce pipeline stalls.
  - Simpler compiler
  - Handles dependencies not known at compile time
  - Allows code compiled for a different machine to run efficiently.



- Understand the dynamic instruction scheduling algorithms better

## 1.3 Project Content

- Research in detail on scoreboarding algorithm and Tomasulo algorithm, develop a solid understanding on the footstone approach.
- Implement these algorithms with GUI, get a better understanding.
- Evaluate and analyse the algorithms.

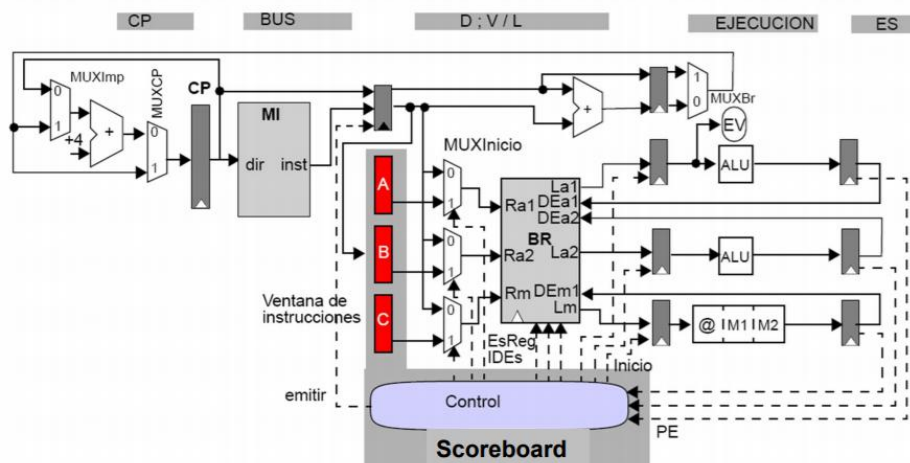


Fig1. Scoreboarding Implementation

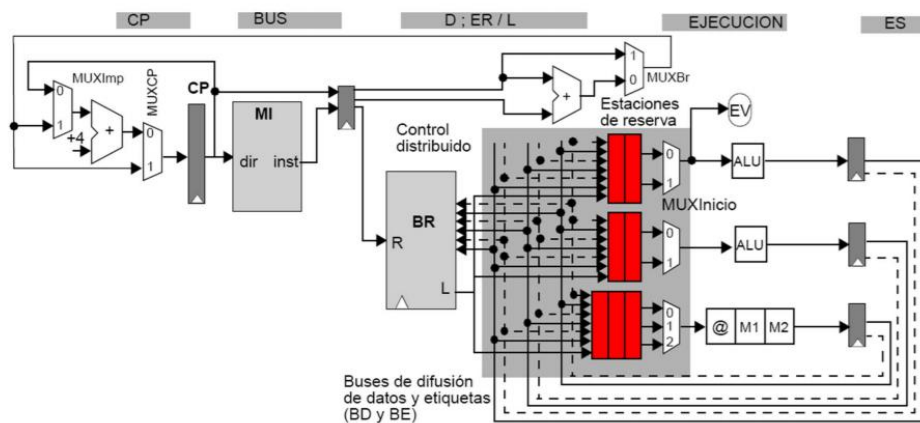


Fig2. Tomasulo Implementation



西安交通大学  
XI'AN JIAOTONG UNIVERSITY

# Related Work

# 02





## 2.1 Scoreboarding

- **Decode+Issue (Issue)**

- decode instructions
- check for structural and WAW hazards
- stall until structural and WAW hazards are resolved

- **Read operands (Read)**

- wait until no RAW hazards
- then read operands

- **Execution (EX)**

- operate on operands
- may be multiple cycles - notify scoreboard when done

- **Write result (WB)**

- finish execution
- stall if WAR hazard

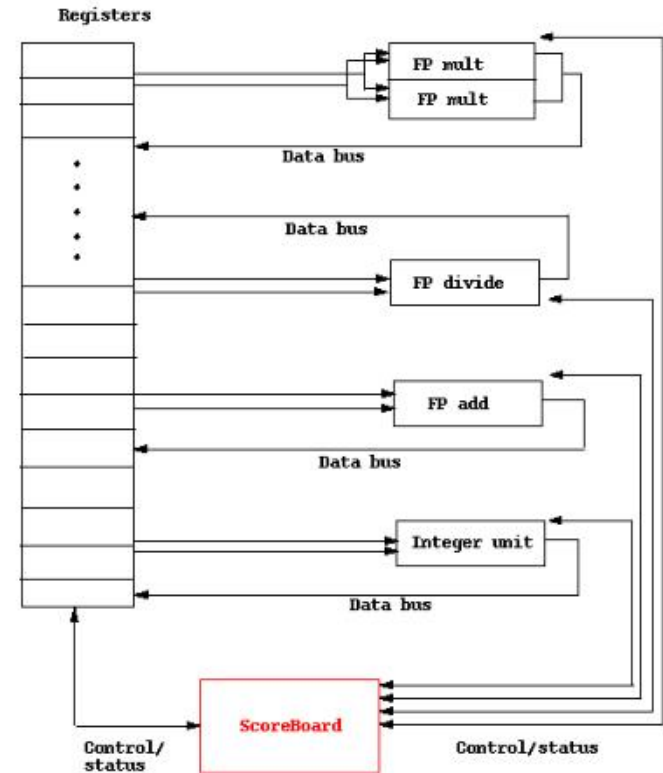


Fig.3 basic structure of machine with scoreboarding



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

**Fj, Fk**—Source-register numbers

**Rj, Rk**—Flags indicating when Fj, Fk are ready

[illegible]





## Scoreboarding Example Cycle 0

[illegible]



## 2.1 Scoreboarding

### Scoreboarding Example Cycle 1

<u>Instruction status</u>				<u>Operand</u>		<u>Execution</u>	<u>Write</u>						
Instruction		<i>j</i>	<i>k</i>	<i>Issue</i>	<i>Read</i>	<i>start-complete</i>	<i>Result</i>						
LD	F6	34+	R2	1									
LD	F2	45+	R3										
MULD	F0	F2	F4										
SUBD	F8	F6	F2										
DIVD	F10	F0	F6										
ADDD	F6	F8	F2										
<u>Functional Unit Status</u>													
	<i>Time</i>	<i>Name</i>	<i>Busy</i>	<i>Op</i>	<i>Fi</i>	<i>Fj</i>	<i>Fk</i>	<i>Qj</i>	<i>Qk</i>	<i>Rj</i>	<i>Rk</i>		
	0	Integer	Yes	Load	F6		R2				Yes		
	0	Mult1	No										
		Mult2	No										
	0	Add	No										
	0	Divide	No										
<u>Register result status</u>													
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>
1			FU					Integer					



## Scoreboarding Example Cycle 2

Instruction status				Operand		Execution	Write					
Instruction	<i>j</i>	<i>k</i>	Issue	Read	start-complete	Result						
LD	F6	34+	R2	1	2							
LD	F2	45+	R3									
MULTD	F0	F2	F4									
SUBD	F8	F6	F2									
DIVD	F10	F0	F6									
ADDD	F6	F8	F2									
Functional Unit Status												
	Time	Name	Busy	Op	Fi	Fj	Fk	Qj	Qk	Rj	Rk	
	0	Integer	Yes	Load	F6		R2				Yes	
	0	Mult1	No									
		Mult2	No									
	0	Add	No									
	0	Divide	No									
Register result status												
Clock				F0	F2	F4	F6	F8	F10	F12	...	F30
2			FU				Integer					

Issue 2nd Load?



## Scoreboarding Example Cycle 3

Instruction status				Operand			Execution	Write					
Instruction		<i>j</i>	<i>k</i>	Issue	Read	start-complete	Result						
LD	F6	34+	R2	1	2	3							
LD	F2	45+	R3										
MULTD	F0	F2	F4										
SUBD	F8	F6	F2										
DIVD	F10	F0	F6										
ADDD	F6	F8	F2										
Functional Unit Status													
	Time	Name	Busy	Op	Fi	Fj	Fk	Qj	Qk	Rj	Rk		
	0	Integer	Yes	Load	F6		R2				Yes		
	0	Mult1	No										
		Mult2	No										
	0	Add	No										
	0	Divide	No										
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>	
3			FU				Integer						

Issue 2nd Load?



## 2.1 Scoreboarding

### Scoreboarding Example Cycle 4

Instruction status				Operand		Execution	Write					
Instruction		<i>j</i>	<i>k</i>	Issue	Read	start-complete	Result					
LD	F6	34+	R2	1	2	3 -- 3	4					
LD	F2	45+	R3									
MULTD	F0	F2	F4									
SUBD	F8	F6	F2									
DIVD	F10	F0	F6									
ADDD	F6	F8	F2									
Functional Unit Status												
	Time	Name	Busy	Op	Fi	Fj	Fk	Qj	Qk	Rj	Rk	
	0	Integer	Yes	Load	F6		R2				Yes	
	0	Mult1	No									
		Mult2	No									
	0	Add	No									
	0	Divide	No									
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>
4			FU				Integer					





## 2.1 Scoreboarding

### Scoreboarding Example Cycle 5

Instruction status				Operand		Execution	Write					
Instruction		<i>j</i>	<i>k</i>	Issue	Read	start -complete	Result					
LD	F6	34+	R2	1	2	3 -- 3	4					
LD	F2	45+	R3	5								
MULTD	F0	F2	F4									
SUBD	F8	F6	F2									
DIVD	F10	F0	F6									
ADDD	F6	F8	F2									
Functional Unit Status												
	Time	Name	Busy	Op	Fi	Fj	Fk	Qj	Qk	Rj	Rk	
	0	Integer	Yes	Load	F2		R3				Yes	
	0	Mult1	No									
		Mult2	No									
	0	Add	No									
	0	Divide	No									
Register result status												
Clock				F0	F2	F4	F6	F8	F10	F12	...	F30
5			FU		Integer							



## 2.1 Scoreboarding

### Scoreboarding Example Cycle 6

Instruction status				Operand		Execution	Write					
Instruction		<i>j</i>	<i>k</i>	Issue	Read	start-complete	Result					
LD	F6	34+	R2	1	2	3 -- 3	4					
LD	F2	45+	R3	5	6							
MULTD	F0	F2	F4	6								
SUBD	F8	F6	F2									
DIVD	F10	F0	F6									
ADD	F6	F8	F2									
Functional Unit Status												
	Time	Name	Busy	Op	<i>Fi</i>	<i>Fj</i>	<i>Fk</i>	<i>Qj</i>	<i>Qk</i>	<i>Rj</i>	<i>Rk</i>	
	0	Integer	Yes	Load	F2		R3				Yes	
	0	Mult1	Yes	Mult	F0	F2	F4	integer		No	Yes	
		Mult2	No									
	0	Add	No									
	0	Divide	No									
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>
6			FU	Mult1	Integer							



## 2.1 Scoreboarding

### Scoreboarding Example Cycle 7

Instruction status				Operand		Execution	Write					
Instruction	<i>j</i>	<i>k</i>	Issue	Read	start-complete	Result						
LD	F6	34+	R2	1	2	3 -- 3	4					
LD	F2	45+	R3	5	6	7 -- 7						
MULTD	F0	F2	F4	6								
SUBD	F8	F6	F2	7								
DIVD	F10	F0	F6									
ADDD	F6	F8	F2									
Functional Unit Status												
	Time	Name	Busy	Op	Fi	Fj	Fk	Qj	Qk	Rj	Rk	
	0	Integer	Yes	Load	F2		R3				Yes	
	0	Mult1	Yes	Mult	F0	F2	F4	integer		No	Yes	
		Mult2	No									
	0	Add	Yes	Sub	F8	F6	F2	integer	Yes	No		
	0	Divide	No									
Register result status												
Clock				F0	F2	F4	F6	F8	F10	F12	...	F30
7			FU	Mult1	Integer			Add				





## 2.1 Scoreboarding

### Scoreboarding Example Cycle 8

Instruction status				Operand		Execution	Write					
Instruction		<i>j</i>	<i>k</i>	Issue	Read	start - complete	Result					
LD	F6	34+	R2	1	2	3 -- 3	4					
LD	F2	45+	R3	5	6	7 -- 7	8					
MULTD	F0	F2	F4	6								
SUBD	F8	F6	F2	7								
DIVD	F10	F0	F6	8								
ADDD	F6	F8	F2									
Functional Unit Status												
	Time	Name	Busy	Op	Fi	Fj	Fk	Qj	Qk	Rj	Rk	
	0	Integer	Yes	Ld	F2		R3				Yes	
	0	Mult1	Yes	Mult	F0	F2	F4	integer		No	Yes	
		Mult2	No									
	0	Add	Yes	Sub	F8	F6	F2	integer		Yes	No	
	0	Divide	Yes	Div	F10	F0	F6	mult1		No	Yes	
Register result status												
Clock				F0	F2	F4	F6	F8	F10	F12	...	F30
8			FU	Mult1	Integer			Add	Divide			



## 2.1 Scoreboarding

### Scoreboarding Example Cycle 9

Instruction status			Operand		Execution	Write					
Instruction	<i>j</i>	<i>k</i>	Issue	Read	start-complete	Result					
LD	F6	34+	R2	1	2	3 -- 3	4				
LD	F2	45+	R3	5	6	7 -- 7	8				
MULTD	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	Fi	Fj	Fk	Qj	Qk	Rj	Rk
	0	Integer	No								
	10	Mult1	Yes	Mult	F0	F2	F4			Yes	Yes
		Mult2	No								
	2	Add	Yes	Sub	F8	F6	F2			Yes	Yes
	0	Divide	Yes	Div	F10	F0	F6	mult1		No	Yes
Register result status											
Clock				F0	F2	F4	F6	F8	F10	F12	... F30
9			FU	Mult1				Add	Divide		



## 2.1 Scoreboarding

### Scoreboarding Example Cycle 10

Instruction status				Operand		Execution	Write					
Instruction		<i>j</i>	<i>k</i>	Issue	Read	start-complete	Result					
LD	F6	34+	R2	1	2	3 -- 3	4					
LD	F2	45+	R3	5	6	7 -- 7	8					
MULTD	F0	F2	F4	6	9	10 --						
SUBD	F8	F6	F2	7	9	10 --						
DIVD	F10	F0	F6	8								
ADDD	F6	F8	F2									
Functional Unit Status												
	Time	Name	Busy	Op	Fi	Fj	Fk	Qj	Qk	Rj	Rk	
	0	Integer	No									
	9	Mult1	Yes	Mult	F0	F2	F4			Yes	Yes	
		Mult2	No									
	1	Add	Yes	Sub	F8	F6	F2			Yes	Yes	
	0	Divide	Yes	Div	F10	F0	F6	mult1		No	Yes	
Register result status												
Clock					F0	F2	F4	F6	F8	F10	F12	... F30
10			FU	Mult1					Add	Divide		



## 2.1 Scoreboarding

### Scoreboarding Example Cycle 11

Instruction status				Operand		Execution	Write					
Instruction		<i>j</i>	<i>k</i>	Issue	Read	start-complete	Result					
LD	F6	34+	R2	1	2	3 -- 3	4					
LD	F2	45+	R3	5	6	7 -- 7	8					
MULTD	F0	F2	F4	6	9	10 --						
SUBD	F8	F6	F2	7	9	10 -- 11						
DIVD	F10	F0	F6	8								
ADDD	F6	F8	F2									
Functional Unit Status												
	Time	Name	Busy	Op	Fi	Fj	Fk	Qj	Qk	Rj	Rk	
	0	Integer	No									
	8	Mult1	Yes	Mult	F0	F2	F4			Yes	Yes	
		Mult2	No									
	0	Add	Yes	Sub	F8	F6	F2			Yes	Yes	
	0	Divide	Yes	Div	F10	F0	F6	mult1		No	Yes	
Register result status												
Clock				F0	F2	F4	F6	F8	F10	F12	...	F30
11			FU	Mult1				Add	Divide			





## 2.1 Scoreboarding

### Scoreboarding Example Cycle 12

Instruction status				Operand		Execution	Write					
Instruction		<i>j</i>	<i>k</i>	Issue	Read	start-complete	Result					
LD	F6	34+	R2	1	2	3 -- 3	4					
LD	F2	45+	R3	5	6	7 -- 7	8					
MULTD	F0	F2	F4	6	9	10 --						
SUBD	F8	F6	F2	7	9	10 -- 11	12					
DIVD	F10	F0	F6	8								
ADDD	F6	F8	F2									
Functional Unit Status												
	Time	Name	Busy	Op	Fi	Fj	Fk	Qj	Qk	Rj	Rk	
	0	Integer	No									
	7	Mult1	Yes	Mult	F0	F2	F4			Yes	Yes	
		Mult2	No									
	0	Add	Yes	Sub	F8	F6	F2			Yes	Yes	
	0	Divide	Yes	Div	F10	F0	F6	mult1		No	Yes	
Register result status												
Clock				F0	F2	F4	F6	F8	F10	F12	...	F30
12			FU	Mult1				Add	Divide			



## 2.1 Scoreboarding

### Scoreboarding Example Cycle 13

Instruction status				Operand		Execution	Write					
Instruction		<i>j</i>	<i>k</i>	Issue	Read	start-complete	Result					
LD	F6	34+	R2	1	2	3 -- 3	4					
LD	F2	45+	R3	5	6	7 -- 7	8					
MULTD	F0	F2	F4	6	9	10 --						
SUBD	F8	F6	F2	7	9	10 -- 11	12					
DIVD	F10	F0	F6	8								
ADDD	F6	F8	F2	13								
Functional Unit Status												
	Time	Name	Busy	Op	Fi	Fj	Fk	Qj	Qk	Rj	Rk	
	0	Integer	No									
	6	Mult1	Yes	Mult	F0	F2	F4			Yes	Yes	
		Mult2	No									
	0	Add	Yes	Add	F6	F8	F2			Yes	Yes	
	0	Divide	Yes	Div	F10	F0	F6	mult1		No	Yes	
Register result status												
Clock				F0	F2	F4	F6	F8	F10	F12	...	F30
13			FU	Mult1			Add		Divide			



## 2.1 Scoreboarding

### Scoreboarding Example Cycle 14

<u>Instruction status</u>				<i>Operand</i>		<i>Execution</i>	<i>Write</i>					
Instruction		<i>j</i>	<i>k</i>	<i>Issue</i>	<i>Read</i>	<i>start-complete</i>	<i>Result</i>					
LD	F6	34+	R2	1	2	3 -- 3	4					
LD	F2	45+	R3	5	6	7 -- 7	8					
MULTD	F0	F2	F4	6	9	10 --						
SUBD	F8	F6	F2	7	9	10 -- 11	12					
DIVD	F10	F0	F6	8								
ADDD	F6	F8	F2	13	14							
<u>Functional Unit Status</u>												
	<i>Time</i>	<i>Name</i>	<i>Busy</i>	<i>Op</i>	<i>Fi</i>	<i>Fj</i>	<i>Fk</i>	<i>Qj</i>	<i>Qk</i>	<i>Rj</i>	<i>Rk</i>	
	0	Integer	No									
	5	Mult1	Yes	Mult	F0	F2	F4			Yes	Yes	
		Mult2	No									
	2	Add	Yes	Add	F6	F8	F2			Yes	Yes	
	0	Divide	Yes	Div	F10	F0	F6	mult1		No	Yes	
<u>Register result status</u>												
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>
14			FU	Mult1			Add		Divide			



## 2.1 Scoreboarding

### Scoreboarding Example Cycle 15

Instruction status				Operand		Execution	Write					
Instruction		<i>j</i>	<i>k</i>	Issue	Read	start-complete	Result					
LD	F6	34+	R2	1	2	3 -- 3	4					
LD	F2	45+	R3	5	6	7 -- 7	8					
MULTD	F0	F2	F4	6	9	10 --						
SUBD	F8	F6	F2	7	9	10 -- 11	12					
DIVD	F10	F0	F6	8								
ADDD	F6	F8	F2	13	14	15 --						
Functional Unit Status												
	Time	Name	Busy	Op	Fi	Fj	Fk	Qj	Qk	Rj	Rk	
	0	Integer	No									
	4	Mult1	Yes	Mult	F0	F2	F4			Yes	Yes	
		Mult2	No									
	1	Add	Yes	Add	F6	F8	F2			Yes	Yes	
	0	Divide	Yes	Div	F10	F0	F6	mult1		No	Yes	
Register result status												
Clock					F0	F2	F4	F6	F8	F10	F12	... F30
15			FU	Mult1				Add		Divide		





## Scoreboarding Example Cycle 61

<u>Instruction status</u>					<i>Operand</i>		<i>Execution</i>	<i>Write</i>						
Instruction		<i>j</i>	<i>k</i>	<i>Issue</i>	<i>Read</i>	<i>start - complete</i>	<i>Result</i>							
LD	F6	34+	R2	1	2	3 -- 3	4							
LD	F2	45+	R3	5	6	7 -- 7	8							
MULTD	F0	F2	F4	6	9	10 -- 19	20							
SUBD	F8	F6	F2	7	9	10 -- 11	12							
DIVD	F10	F0	F6	8	21	22 -- 61								
ADDD	F6	F8	F2	13	14	15 -- 16	22							
<u>Functional Unit Status</u>														
	<i>Time</i>	<i>Name</i>	<i>Busy</i>	<i>Op</i>	<i>Fi</i>	<i>Fj</i>	<i>Fk</i>	<i>Qj</i>	<i>Qk</i>	<i>Rj</i>	<i>Rk</i>			
	0	Integer	No											
	0	Mult1	No											
		Mult2	No											
	0	Add	No											
	0	Divide	Yes	Div	F10	F0	F6			Yes	Yes			
<u>Register result status</u>														
Clock				<i>F0</i>	<i>F2</i>	<i>F4</i>	<i>F6</i>	<i>F8</i>	<i>F10</i>	<i>F12</i>	<i>...</i>	<i>F30</i>		
61			FU						Divide					



# Scoreboarding Example Cycle 62

Instruction status					Operand		Execution		Write					
Instruction		<i>j</i>	<i>k</i>	<i>Issue</i>	<i>Read</i>	<i>start-complete</i>	<i>Result</i>							
LD	F6	34+	R2	1	2	3 -- 3	4							
LD	F2	45+	R3	5	6	7 -- 7	8							
MULTD	F0	F2	F4	6	9	10 -- 19	20							
SUBD	F8	F6	F2	7	9	10 -- 11	12							
DIVD	F10	F0	F6	8	21	22 -- 61	62							
ADDD	F6	F8	F2	13	14	15 -- 16	22							
Functional Unit Status														
	<i>Time</i>	<i>Name</i>	<i>Busy</i>	<i>Op</i>	<i>Fi</i>	<i>Fj</i>	<i>Fk</i>	<i>Qj</i>	<i>Qk</i>	<i>Rj</i>	<i>Rk</i>			
	0	Integer	No											
	0	Mult1	No											
		Mult2	No											
	0	Add	No											
	0	Divide	Yes	Div	F10	F0	F6			Yes	Yes			
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>...</i>	<i>F30</i>		
62			FU						Divide					



## Scoreboarding Example Cycle 63

<u>Instruction status</u>			<u>Operand</u>		<u>Execution</u>	<u>Write</u>					
Instruction		<i>j</i>	<i>k</i>	<i>Issue</i>	<i>Read</i>	<i>start-complete</i>	<i>Result</i>				
LD	F6	34+	R2	1	2	3 -- 3	4				
LD	F2	45+	R3	5	6	7 -- 7	8				
MULD	F0	F2	F4	6	9	10 -- 19	20				
SUBD	F8	F6	F2	7	9	10 -- 11	12				
DIVD	F10	F0	F6	8	21	22 -- 61	62				
ADD	F6	F8	F2	13	14	15 -- 16	22				
<u>Functional Unit Status</u>											
	<i>Time</i>	<i>Name</i>	<i>Busy</i>	<i>Op</i>	<i>Fi</i>	<i>Fj</i>	<i>Fk</i>	<i>Qj</i>	<i>Qk</i>	<i>Rj</i>	<i>Rk</i>
	0	Integer	No								
	0	Mult1	No								
		Mult2	No								
	0	Add	No								
	0	Divide	No								
<u>Register result status</u>											
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>
63			FU								



西安交通大学  
XI'AN JIAOTONG UNIVERSITY

# Demo Description

# 03





## 3.1 Scoreboarding



请输入跳转周期，输入‘-1’结束程序

63

Cycle : 63

Instruction status

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

Functional Unit status

Name	Busy	Op	Fi	Fj	Fk	Qj	Qk	Rj	Rk
Integer	No								
Mult1	No								
Mult2	No								
Add	No								
Divide	No								

Register result status

F0	F1	F2	F3	F4	F5
F6	F7	F8	F9	F10	F11

请输入跳转周期，输入‘-1’结束程序

□



# 3.2 Tomasulo

Activities Tk

Wed 06:26

Tomasulo算法模拟

指令状态表

	操作类型	流出时钟	执行延迟	是否已流出	是否进入执行	执行当前指令的保留站序号	是否写回
指令0	LD	0	2	False	False	None	False
指令1	LD	0	2	False	False	None	False
指令2	MUL.D	0	11	False	False	None	False
指令3	SUB.D	0	3	False	False	None	False
指令4	DIV.D	0	41	False	False	None	False
指令5	ADD.D	0	3	False	False	None	False

保留站状态表

	保留站属性	当前操作类型	编号	Qj	Qk	Vj	Vk	Busy	A	开始时钟	结果
保留站1	LD	None	1	None	None	None	None	False	None	None	None
保留站2	LD	None	2	None	None	None	None	False	None	None	None
保留站3	ADD.D	None	3	None	None	None	None	False	None	None	None
保留站4	ADD.D	None	4	None	None	None	None	False	None	None	None
保留站5	ADD.D	None	5	None	None	None	None	False	None	None	None
保留站6	DIV.D	None	6	None	None	None	None	False	None	None	None
保留站7	DIV.D	None	7	None	None	None	None	False	None	None	None

寄存器状态表

F0	F2	F4	F6	F8	F10	R2	R3
0	0	0	0	0	0	0	0

单步执行

刷新

自动执行

zdhs61 JYK

当前时钟：

0



西安交通大学  
XI'AN JIAOTONG UNIVERSITY

# Thanks !

Computer Architecture  
2020.4

