Instruction Level Parallelism (ILP)

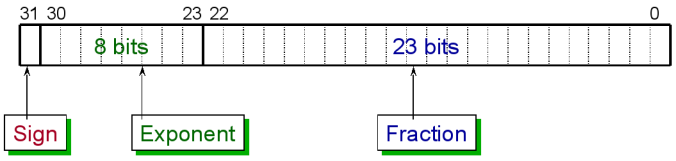
Preserve the sequential semantics of the ISA but... try to execute as many instructions at once as we can.

Part 1: Scoreboarding
Part 2: Tomasulo's Algorithm

But first, a word from our sponsor, floating point...

(IEEE 754) Floating Point Representation

$$(-1)^S \times F \times 2^E$$



0: Positive
1: Negative

127 bias

Leading '1' is implied, but not represented

Key Points:

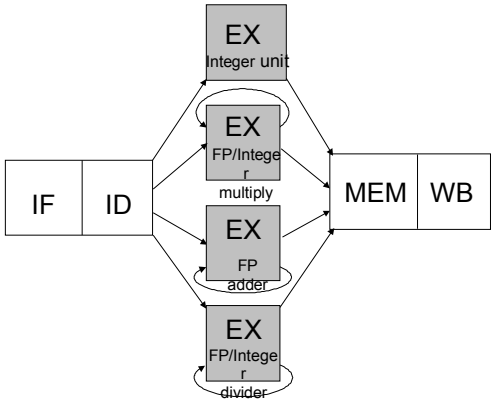
- Limited precision
- Limited range
- Distinct set of instructions
- Distinct set of registers
- Slow to operate on (relative to integers)

It's complicated

What does 0x000000000 represent?

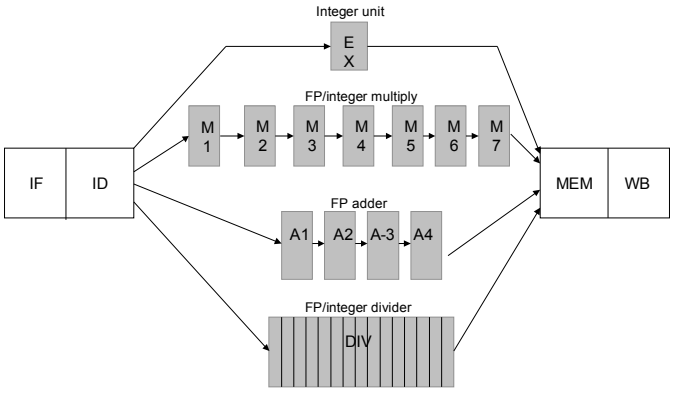
How is 3.25 represented?

Floating Point and the Pipeline



The MIPS pipeline with floating-point functional units.

Pipelining the FP Units



New Problems

- Structural Hazards
  - FP divide takes many cycles, and is not pipelined
  - May need to write more than one register in a cycle
- Instructions may complete out of order
  - WAW (write-after-write) hazards are possible
  - “Precise exceptions” are more difficult to implement
- Longer pipelines may result in more difficulty dealing with RAW (read-after-write) dependences

Once we deal with those problems, a new problem is revealed: can’t keep all the functional units busy

Addressing the Problem

- Consider the following code fragment:

```
DIVD        F0, F2, F4
ADDD        F10, F0, F8
MULTD       F12, F8, F14
```

- What is the problem?
  - ADDD stalls due to RAW hazard
  - MULTD has no dependences, but is marooned behind ADDD
- In-order execution limits performance

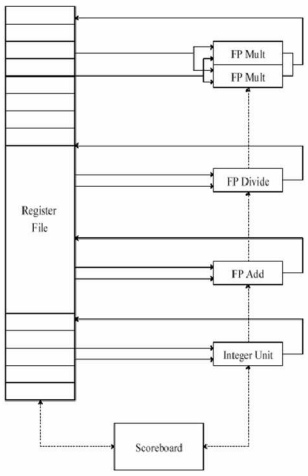
Out of Order Execution

- Key idea: Allow out-of-order execution

```
DIVD   F0,F2,F4
ADDD   F10,F0,F8
SUBD   F12,F8,F14
```

- Leads to out-of-order completion
- The technique we'll look at: scoreboarding
  - in-order issue: All instructions pass through a single issue stage for scheduling
  - Dates to the CDC 6600 (1964)
- Remember that we need to preserve sequential semantics
  - Dependences (RAW, WAR, WAW) restrict legal parallelizations

Dynamic MIPS Datapath



Scoreboarding

- Data path now has two largely decoupled pieces:
  - IF fetches an instruction each cycle
    - » There is a small window (buffer) of already fetched instructions
    - » If issue stalls, the buffer fills
    - » When instructions complete, they leave the buffer
  - Scoreboarding: 4-stage execution
    - » Issue – check structural /WAW hazards (stall until clear)
    - » Read ops – check RAW (wait till operands ready, read regs)
    - » Execute – execute operation. Notify scoreboard when done
    - » Write – check for WAR (stall write until clear)

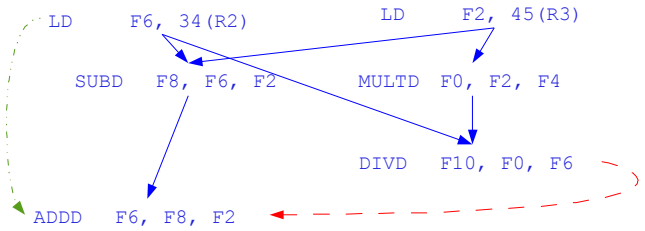
Dealing with Hazards

- **Structural hazards**
  - **Obvious – desired EX unit must be free**
- **Data Hazards**
  - **More complex due to parallel multi-cycle EX units**
  - **RAW – delay operand fetch until producing instruction finishes register write back**
  - **WAW – issue only one instruction at a time for each destination register**
  - **WAR – delay write back until reading instruction has fetched the previous value**

```
1.Issue
2.Read
3.Execute
4.Write
```

Example Code

```
LD      F6, 34(R2)
LD      F2, 45(R3)
MULTD   F0, F2, F4
SUBD    F8, F6, F2
DIVD    F10, F0, F6
ADDD    F6, F8, F2
```



Four Stages of Scoreboard Control

1. **Issue**—decode instructions & check for structural hazards

   If a functional unit for the instruction is free and no other active instruction has the same destination register (WAW), then issue the instruction. Otherwise, stall this and all following instructions.

2. **Read operands**—wait until no flow dependences (RAW), then read operands from registers

   No forwarding done. Why? Not as useful in this situation, as each functional unit writes a register as soon as the value is available.

3. **Execution**—operate on operands

4. **Write result**—finish execution

   Must wait for earlier instructions to have read result register (WAR)

Scoreboard: 3 Tables + Rules

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 regs Fj, Fk**
   - **Rj, Rk—Flags indicating if Fj, Fk are ready**

- **Register result status—Indicates which functional unit will write each register.**

Scoreboard Example

Instruction status

Instruction		j	k	Issue	Read operan	Executi comple	Write Result
LD	F6	34+	R2				
LD	F2	45+	R3				
MULT	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 j Qj	FU for k Qk	Fj? Rj	Fk? Rk
	Integer	No								
	Mult1	No								
	Mult2	No								
	Add	No								
	Divide	No								

Register result status

Clock	F0	F2	F4	F6	F8	F10	F12	…	F30
FU									

Detailed Scoreboard Pipeline Control

Instruction status	Wait until	Bookkeeping
Issue	Not busy (FU) and not result(D)	Busy(FU)← yes; Op(FU)← op; Fi(FU)← `D'; Fj(FU)← `S1'; Fk(FU)← `S2'; Qj← Result('S1'); Qk← Result(`S2'); Rj← not Qj; Rk← not Qk; Result('D')← FU;
Read operands	Rj and Rk	Rj← No; Rk← No
Execution complete	Functional unit done	
Write result	∀f((Fj(f)≠Fi(FU) or Rj(f)=No) & (Fk(f) ≠Fi(FU) or Rk(f)=No))	∀f(if Qj(f)=FU then Rj(f)← Yes); ∀f(if Qk(f)=FU then Rj(f)← Yes); Result(Fi(FU))← 0; Busy(FU)← No

Scoreboard Example

- **Assume 2 MULT, 1 DIV, 1 ADD/SUB FP units, plus 1 integer ALU**
- **EX stage timings:**
  - **Int ALU takes 1 cycle**
  - **FP adder takes 2 cycles**
  - **FP mult takes 10 cycles**
  - **FP div takes 40 cycles**

Scoreboard Example

Instruction status

Instruction		j	k	Issue	Read operan	Executi comple	Write Result
LD	F6	34+	R2				
LD	F2	45+	R3				
MULT	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 j Qj	FU for k Qk	Fj? Rj	Fk? Rk
	Integer	No								
	Mult1	No								
	Mult2	No								
	Add	No								
	Divide	No								

Register result status

Clock	F0	F2	F4	F6	F8	F10	F12	…	F30
FU									

Scoreboard Example Cycle 1

Instruction status

Instruction		j	k	Issue	Read operanc	Executi comple	Write Result
LD	F6	34+	R2	1			
LD	F2	45+	R3				
MULT	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 j Qj	FU for k 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
1 FU				Integer					

Scoreboard Example Cycle 2

Instruction status

Instructio	j	k	Issue	Read operar	Execu compl	Write Result
LD F6	34+	R2	1	2		
LD F2	45+	R3				
MUL F0	F2	F4				
SUB F8	F6	F2				
DIVC F10	F0	F6				
ADD F6	F8	F2				

Functional unit status

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

Register result status

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

- **Issue 2nd LD?**

Scoreboard Example Cycle 3

Instruction status

Instructio	j	k	Issue	Read operar	Execu compl	Write Result
LD F6	34+	R2	1	2	3	
LD F2	45+	R3				
MUL F0	F2	F4				
SUB F8	F6	F2				
DIVC F10	F0	F6				
ADD F6	F8	F2				

Functional unit status

Tim	Name	Busy	Op	dest Fi	S1 Fj	S2 Fk	FU foı Qj	FU for Qk	Fj? Rj	Fk? Rk
	Integer	Yes	Load	F6		R2				No
	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?**

Scoreboard Example Cycle 4

Instruction status

Instructio	j	k	Issue	Read operan	Execu compl	Write Result
LD F6	34+	R2	1	2	3	4
LD F2	45+	R3				
MUL F0	F2	F4				
SUB F8	F6	F2				
DIVI F10	F0	F6				
ADD F6	F8	F2				

Functional unit status

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

Register result status

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

Scoreboard Example Cycle 5

Instruction status

Instructio	j	k	Issue	Read operan	Execu compl	Write Result
LD F6	34+	R2	1	2	3	4
LD F2	45+	R3	5			
MUL F0	F2	F4				
SUB F8	F6	F2				
DIVI F10	F0	F6				
ADD F6	F8	F2				

Functional unit status

Tim	Name	Busy	Op	dest Fi	S1 Fj	S2 Fk	FU fo Qj	FU for Qk	Fj? Rj	Fk? Rk
	Integer	Yes	Load	F2		R3				Yes
	Mult1	No								
	Mult2	No								
	Add	No								
	Divide	No								

Register result status

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

Scoreboard Example Cycle 6

Instruction status

Instructio	j	k	Issue	Read operan	Execu compl	Write Result
LD F6	34+	R2	1	2	3	4
LD F2	45+	R3	5	6		
MUL F0	F2	F4	6			
SUB F8	F6	F2				
DIVI F10	F0	F6				
ADD F6	F8	F2				

Functional unit status

Tim	Name	Busy	Op	dest Fi	S1 Fj	S2 Fk	FU fo Qj	FU for Qk	Fj? Rj	Fk? Rk
	Integer	Yes	Load	F2		R3				No
	Mult1	Yes	Mult	F0	F2	F4	Integer		No	Yes
	Mult2	No								
	Add	No								
	Divide	No								

Register result status

Clock		F0	F2	F4	F6	F8	F10	F12	...	F30
6	FU	Mult1	Integer							

Scoreboard Example Cycle 7

Instruction status

Instructio	j	k	Issue	Read operan	Execu compl	Write Result
LD F6	34+	R2	1	2	3	4
LD F2	45+	R3	5	6	7	
MUL F0	F2	F4	6			
SUB F8	F6	F2	7			
DIVI F10	F0	F6				
ADD F6	F8	F2				

Functional unit status

Tim	Name	Busy	Op	dest Fi	S1 Fj	S2 Fk	FU fo Qj	FU for Qk	Fj? Rj	Fk? Rk
	Integer	Yes	Load	F2		R3				No
	Mult1	Yes	Mult	F0	F2	F4	Integer		No	Yes
	Mult2	No								
	Add	Yes	Sub	F8	F6	F2		Intege	Yes	No
	Divide	No								

Register result status

Clock		F0	F2	F4	F6	F8	F10	F12	...	F30
7	FU	Mult1	Integer			Add				

- **Read multiply operands?**

Scoreboard Example Cycle 8a

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	
MUL F0	F2	F4	6			
SUB F8	F6	F2	7			
DIVD F10	F0	F6	8			
ADD 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	F2		R3				No
	Mult1	Yes	Mult	F0	F2	F4	Integer		No	Yes
	Mult2	No								
	Add	Yes	Sub	F8	F6	F2		Integer	Yes	No
	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			

Scoreboard Example Cycle 8b

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
MULT 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	dest Fi	S1 Fj	S2 Fk	FU for j Qj	FU for k Qk	Fj? Rj	Fk? Rk
	Integer	No								
	Mult1	Yes	Mult	F0	F2	F4			Yes	Yes
	Mult2	No								
	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
8 FU	Mult1				Add	Divide			

Scoreboard Example Cycle 9

Instruction status

Instructio	j	k	Issue	Read operan	Execu compl	Write Result
LD F6	34+	R2	1	2	3	4
LD F2	45+	R3	5	6	7	8
MUL F0	F2	F4	6	9		
SUB F8	F6	F2	7	9		
DIVC F10	F0	F6	8			
ADD F6	F8	F2				

Functional unit status

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

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

Scoreboard Example Cycle 11

Instruction status

Instructio	j	k	Issue	Read operan	Execu compl	Write Result
LD F6	34+	R2	1	2	3	4
LD F2	45+	R3	5	6	7	8
MUL F0	F2	F4	6	9		
SUB F8	F6	F2	7	9	11	
DIVC F10	F0	F6	8			
ADD F6	F8	F2				

Functional unit status

Tim	Name	Busy	Op	dest Fi	S1 Fj	S2 Fk	FU fo Qj	FU for Qk	Fj? Rj	Fk? Rk
	Integer	No								
8	Mult1	Yes	Mult	F0	F2	F4			No	No
	Mult2	No								
0	Add	Yes	Sub	F8	F6	F2			No	No
	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			

Scoreboard Example Cycle 12

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
MULT F0	F2	F4	6	9		
SUB F8	F6	F2	7	9	11	12
DIVD F10	F0	F6	8			
ADD 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								
7	Mult1	Yes	Mult	F0	F2	F4			No	No
	Mult2	No								
	Add	No								
	Divide	Yes	Div	F10	F0	F6	Mult1		No	Yes

Register result status

Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
12 FU	Mult1					Divide			

- **Read operands for DIVD?**

Scoreboard Example Cycle 13

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
MULT F0	F2	F4	6	9		
SUB F8	F6	F2	7	9	11	12
DIVD F10	F0	F6	8			
ADD F6	F8	F2	13			

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								
6	Mult1	Yes	Mult	F0	F2	F4			No	No
	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
13 FU	Mult1			Add		Divide			

Scoreboard Example Cycle 14

Instruction status

Instructio	j	k	Issue	Read operan	Execu compl	Write Result
LD F6	34+	R2	1	2	3	4
LD F2	45+	R3	5	6	7	8
MUL F0	F2	F4	6	9		
SUB F8	F6	F2	7	9	11	12
DIVE F10	F0	F6	8			
ADD F6	F8	F2	13	14		

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								
5	Mult1	Yes	Mult	F0	F2	F4			No	No
	Mult2	No								
2	Add	Yes	Add	F6	F8	F2			No	No
	Divide	Yes	Div	F10	F0	F6	Mult1		No	Yes

Register result status

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

Scoreboard Example Cycle 15

Instruction status

Instructio	j	k	Issue	Read operan	Execu compl	Write Result
LD F6	34+	R2	1	2	3	4
LD F2	45+	R3	5	6	7	8
MUL F0	F2	F4	6	9		
SUB F8	F6	F2	7	9	11	12
DIVE F10	F0	F6	8			
ADD F6	F8	F2	13	14		

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								
4	Mult1	Yes	Mult	F0	F2	F4			No	No
	Mult2	No								
1	Add	Yes	Add	F6	F8	F2			No	No
	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			

Scoreboard Example Cycle 16

Instruction status

Instructio	j	k	Issue	Read operan	Execu compl	Write Result
LD F6	34+	R2	1	2	3	4
LD F2	45+	R3	5	6	7	8
MUL F0	F2	F4	6	9		
SUB F8	F6	F2	7	9	11	12
DIVE F10	F0	F6	8			
ADD 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								
3	Mult1	Yes	Mult	F0	F2	F4			No	No
	Mult2	No								
0	Add	Yes	Add	F6	F8	F2			No	No
	Divide	Yes	Div	F10	F0	F6	Mult1		No	Yes

Register result status

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

Scoreboard Example Cycle 17

Instruction status

Instructio	j	k	Issue	Read operan	Execu compl	Write Result
LD F6	34+	R2	1	2	3	4
LD F2	45+	R3	5	6	7	8
MUL F0	F2	F4	6	9		
SUB F8	F6	F2	7	9	11	12
DIVE F10	F0	F6	8			
ADD 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			No	No
	Mult2	No								
	Add	Yes	Add	F6	F8	F2			No	No
	Divide	Yes	Div	F10	F0	F6	Mult1		No	Yes

Register result status

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

- **Write result of ADDD?**

Scoreboard Example Cycle 18

Instruction status

Instructio	j	k	Issue	Read operan	Execu compl	Write Result
LD F6	34+	R2	1	2	3	4
LD F2	45+	R3	5	6	7	8
MUL F0	F2	F4	6	9		
SUB F8	F6	F2	7	9	11	12
DIVC F10	F0	F6	8			
ADD F6	F8	F2	13	14	16	

Functional unit status

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

Register result status

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

Scoreboard Example Cycle 19

Instruction status

Instructio	j	k	Issue	Read operan	Execu compl	Write Result
LD F6	34+	R2	1	2	3	4
LD F2	45+	R3	5	6	7	8
MUL F0	F2	F4	6	9	19	
SUB F8	F6	F2	7	9	11	12
DIVC F10	F0	F6	8			
ADD F6	F8	F2	13	14	16	

Functional unit status

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

Register result status

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

Scoreboard Example Cycle 20

Instruction status

Instructio	j	k	Issue	Read operan	Execu compl	Write Result
LD F6	34+	R2	1	2	3	4
LD F2	45+	R3	5	6	7	8
MUL F0	F2	F4	6	9	19	20
SUB F8	F6	F2	7	9	11	12
DIVI F10	F0	F6	8			
ADD F6	F8	F2	13	14	16	

Functional unit status

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

Register result status

Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
20 FU				Add		Divide			

Scoreboard Example Cycle 21

Instruction status

Instructio	j	k	Issue	Read operan	Execu compl	Write Result
LD F6	34+	R2	1	2	3	4
LD F2	45+	R3	5	6	7	8
MUL F0	F2	F4	6	9	19	20
SUB F8	F6	F2	7	9	11	12
DIVI F10	F0	F6	8	21		
ADD F6	F8	F2	13	14	16	

Functional unit status

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

Register result status

Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
21 FU				Add		Divide			

Scoreboard Example Cycle 22

Instruction status

Instructio	j	k	Issue	Read operan	Execu compl	Write Result
LD F6	34+	R2	1	2	3	4
LD F2	45+	R3	5	6	7	8
MUL F0	F2	F4	6	9	19	20
SUB F8	F6	F2	7	9	11	12
DIVC F10	F0	F6	8	21		
ADD F6	F8	F2	13	14	16	22

Functional unit status

Tim	Name	Busy	Op	dest Fi	S1 Fj	S2 Fk	FU fo Qj	FU for Qk	Fj? Rj	Fk? Rk
	Integer	No								
	Mult1	No								
	Mult2	No								
	Add	No								
40	Divide	Yes	Div	F10	F0	F6			No	No

Register result status

Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
22 FU	Divide								

Scoreboard Example Cycle 61

Instruction status

Instructio	j	k	Issue	Read operan	Execu compl	Write Result
LD F6	34+	R2	1	2	3	4
LD F2	45+	R3	5	6	7	8
MUL F0	F2	F4	6	9	19	20
SUB F8	F6	F2	7	9	11	12
DIVC F10	F0	F6	8	21	61	
ADD F6	F8	F2	13	14	16	22

Functional unit status

Tim	Name	Busy	Op	dest Fi	S1 Fj	S2 Fk	FU fo Qj	FU for Qk	Fj? Rj	Fk? Rk
	Integer	No								
	Mult1	No								
	Mult2	No								
	Add	No								
0	Divide	Yes	Div	F10	F0	F6			No	No

Register result status

Clock	F0	F2	F4	F6	F8	F10	F12	...	F30
61 FU	Divide								

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
MULT	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								