

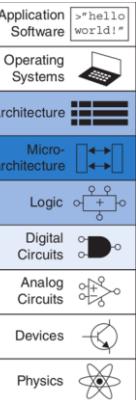
Exercise Session 6

Scoreboard, Tomasulo, Dynamic Branch Prediction, (Extra:Simple Scheduling)
Advanced Computer Architectures

16th April 2025

Davide Conficconi <davide.conficconi@polimi.it>

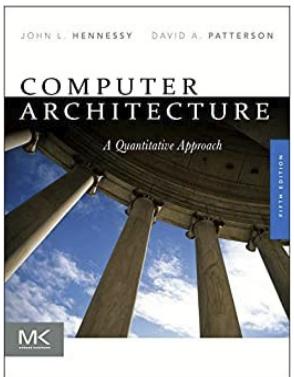
Recall: Material (EVERYTHING OPTIONAL)



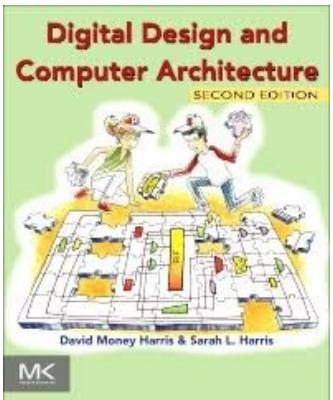
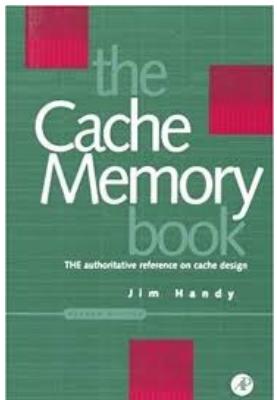
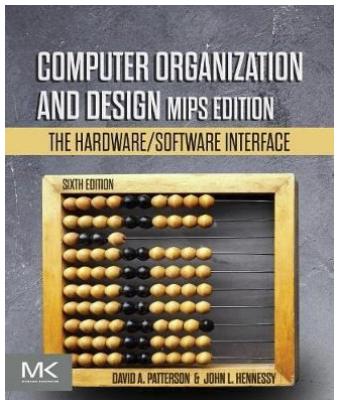
<https://webeep.polimi.it/course/view.php?id=14754>

<https://tinyurl.com/aca-grid25>

Textbook: Hennessy and Patterson, Computer Architecture: A Quantitative Approach



Other Interesting Reference



Recall 2021: The "Joke"



Recall 2022: News from the outer world: Superpod DGX H100

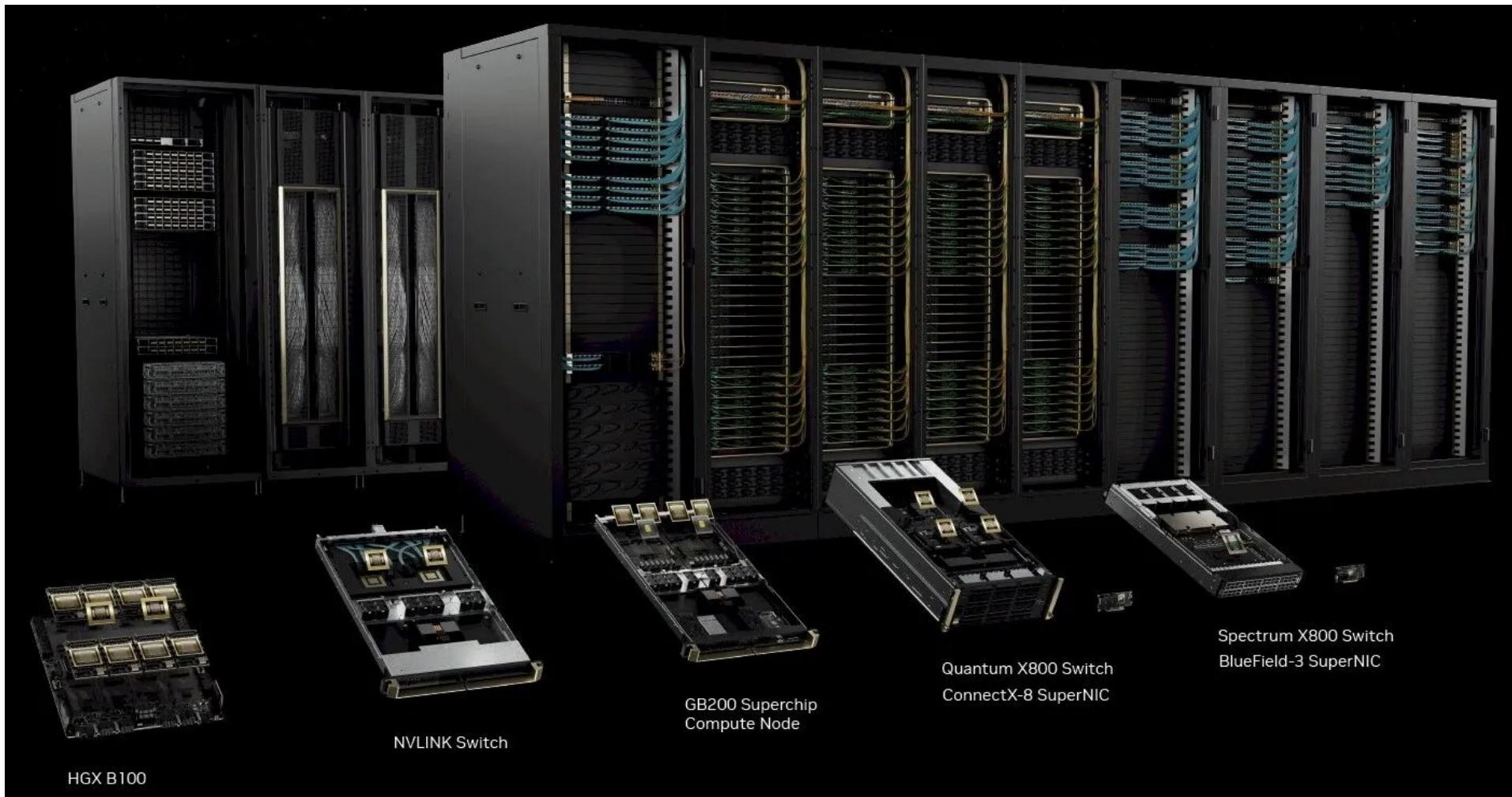
<https://nvidianews.nvidia.com/news/nvidia-announces-dgx-h100-systems-worlds-most-advanced-enterprise-ai-infrastructure>



" Combined With New NVLink Switch System, Each DGX SuperPOD to Deliver 1 Exaflops of AI Performance; New NVIDIA Eos Supercomputer Expected to Be World's Fastest AI System; Immediate On-Ramp for Customers via Expanded DGX Foundry Service"

Recall 2024: News from the outer world: GTC'24 Blackwell Platform

<https://www.nextplatform.com/2024/03/19/how-nvidia-blackwell-systems-attack-1-trillion-parameter-ai-models/>



“The Blackwell platform starts with the HGX B100 and HGX B200 GPU compute complexes, which will be deployed in DGX B100 and DGX B200 systems and which use geared down variants of the Blackwell GPU that can be air cooled. [...]”

Recall 2024: News from the outer world: Venado Supercomputer at Los Alamos

<https://www.nextplatform.com/2024/04/15/los-alamos-pushes-the-memory-wall-with-venado-supercomputer/>



ANNOUNCING “VENADO”
First US-Based Grace CPU and Grace Hopper Superchip Supercomputer

10 EF of Peak AI performance

Balanced, flexible heterogeneous architecture

Enabling breakthroughs in material science, renewable energy, energy distribution, and more...

Hewlett Packard Enterprise Los Alamos NATIONAL LABORATORY NVIDIA

VENADO

“The new Venado system is not a workhorse machine in the Los Alamos fleet, but an experimental one that is built from its own budget and for the express purpose of doing hardware and software research”

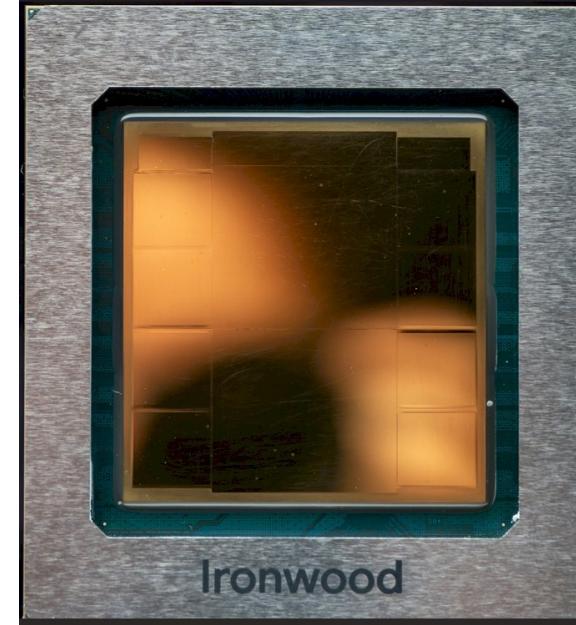
→ irregular and sparse applications → CPU → mem bw/\$ than FLOPS/\$



This image is a work of a [United States Department of Energy](#) (or predecessor organization) employee, taken or made as part of that person's official duties. As a [work](#) of the U.S. federal government, the image is in the [public domain](#).

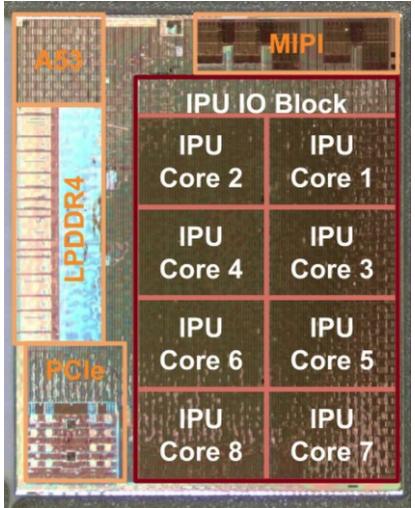
News from the outer world: IRONWOOD TPU Announced

<https://www.nextplatform.com/2025/04/09/with-ironwood-tpu-google-pushes-the-ai-accelerator-to-the-floor/>



News from the outer world: Recall on Google AI Accelerators

Pixel Visual Core



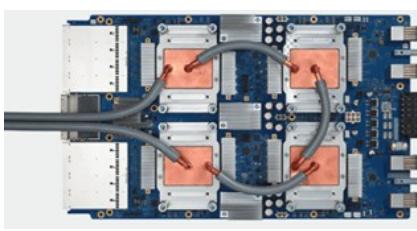
<https://shop.elsevier.com/books/computer-architecture/hennessy/978-0-12-811905-1>



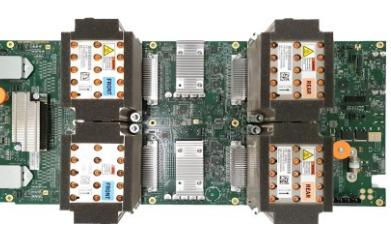
v1



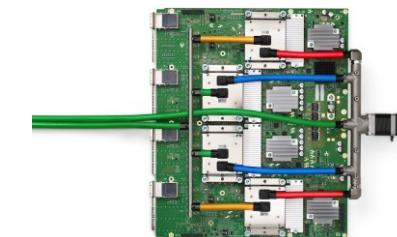
v2



v3



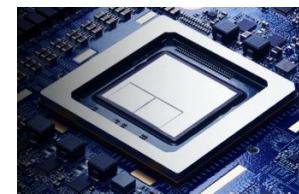
v4i



v4



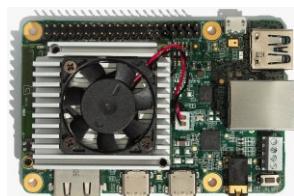
v5e



v5p



v6



edge



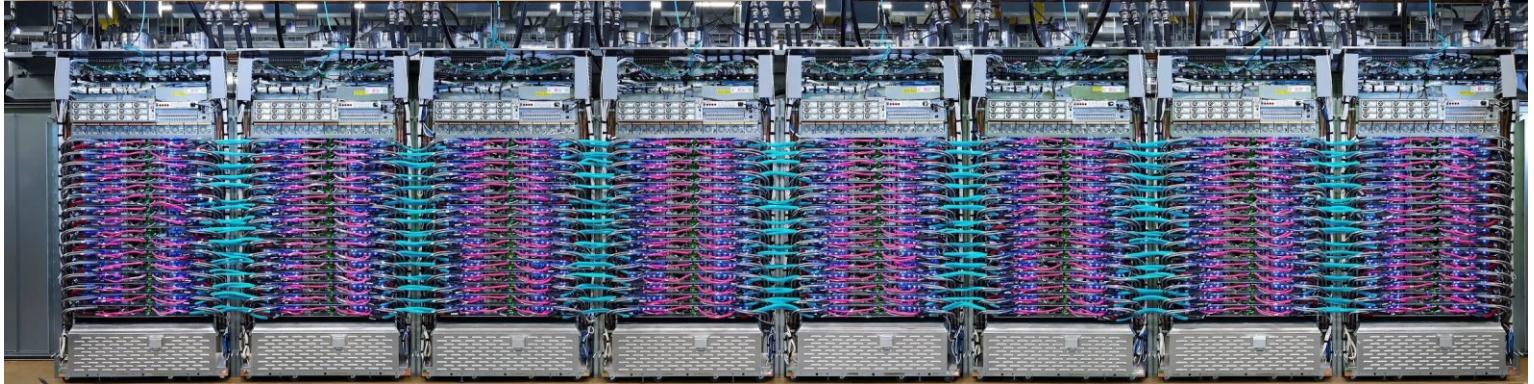
News from the outer world: AI Accelerators Scaling Out!



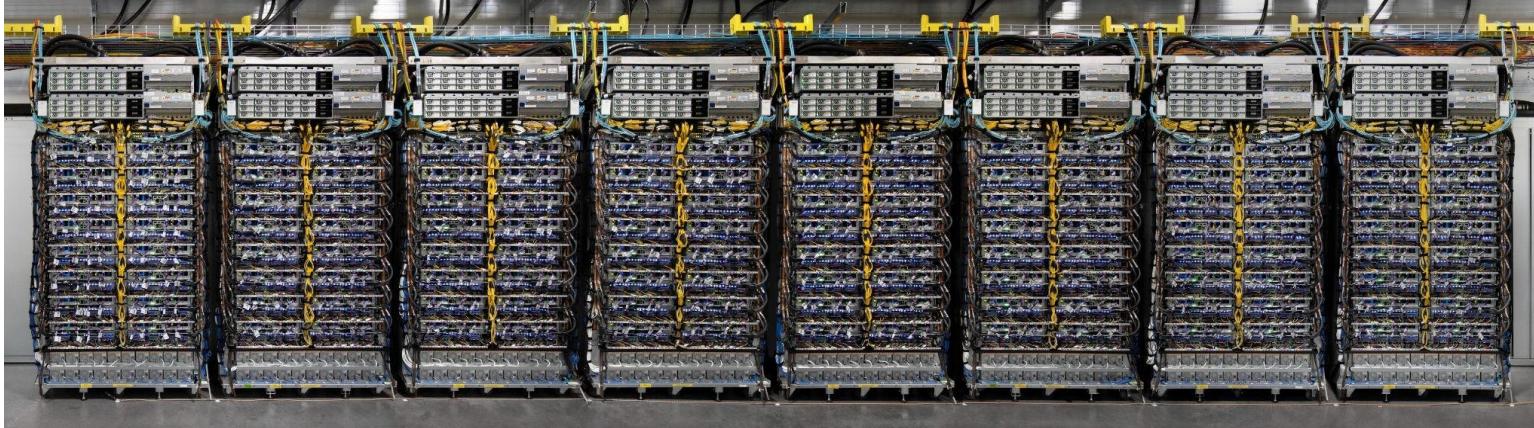
Scale * = 4;



TPU v2: 256 nodes
electrical, 2D torus



TPU v3: 1024 nodes
electrical, 2D torus



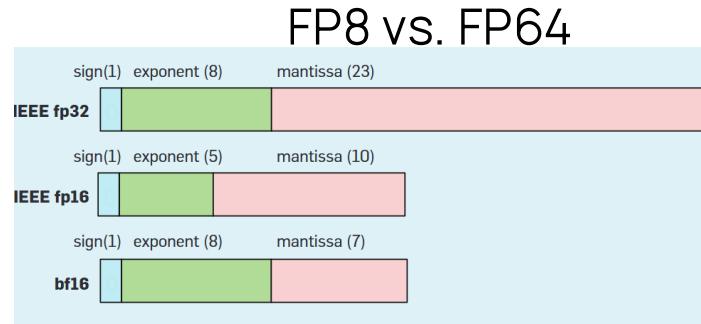
TPU v4: 4096 nodes electrical in-
rack optical across racks 3D
torus
8/64 racks shown
(7 more rows in depth)⁴

News from the outer world: Performance Evaluation Debate

<https://itdaily.com/news/datacenter/google-ironwood-el-capitan-comparison/>

	TPU v4	TPU v5p	Ironwood
Pod Size (chips)	4896	8960	9216
HBM Bandwidth/Capacity	32 GB @ 1.2 TBs HBM	95 GB @ 2.8 TBs HBM	192 GB @ 7.4 TBs HBM
Peak Flops per chip	275 TFLOPS	459 TFLOPS	4614 TFLOPS

"How Google Lies About the Power of Its Latest Chips, Compared to El Capitan"

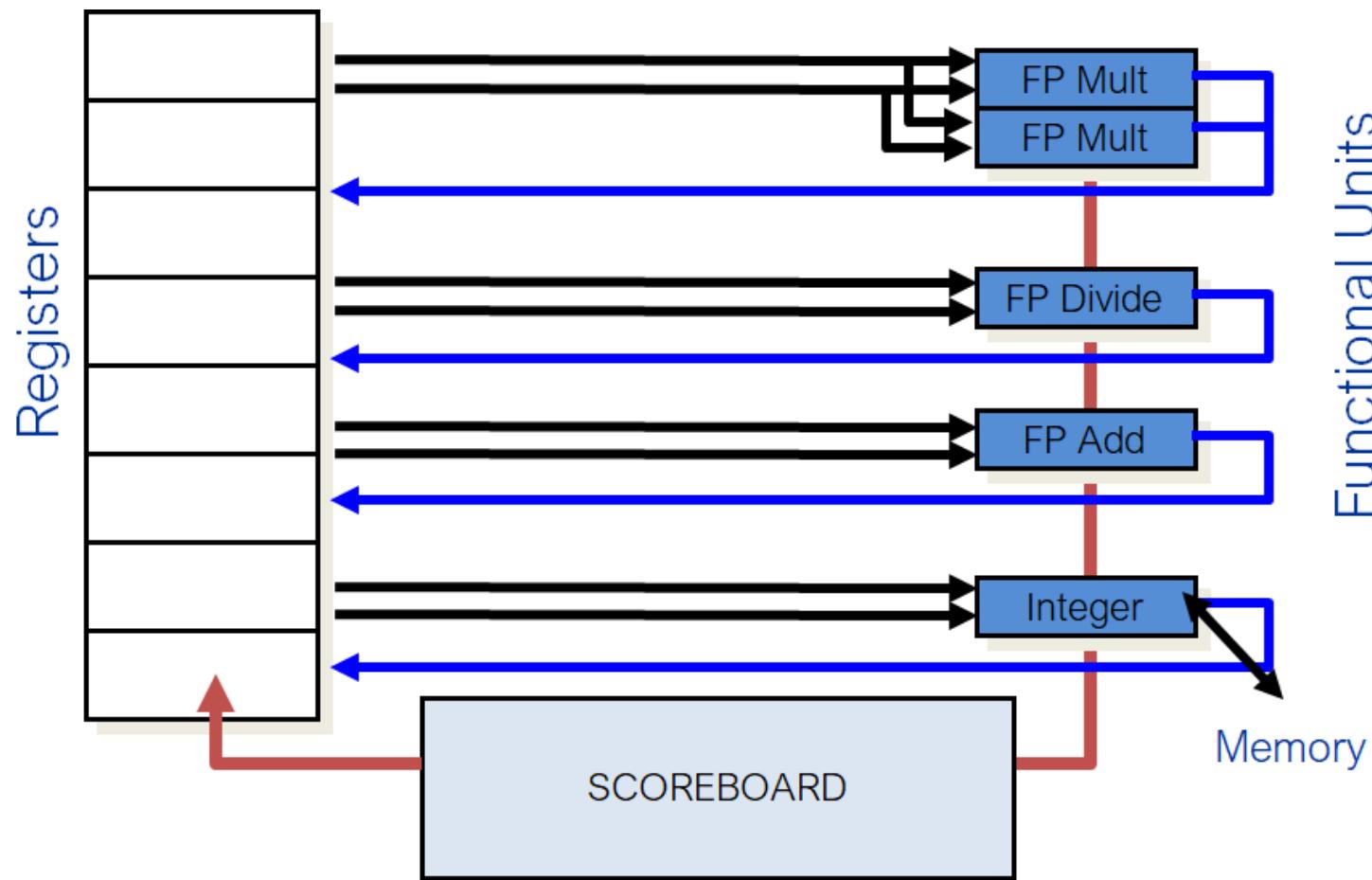


42.5 ExaFlops (Google Cluster) vs 1.7 El Capitan

"Conservatively estimated, El Capitan's AI accelerators together deliver at least 85 ExaFlops of FP8 computing power."

Performance numbers must be related and contextualized to something!

Exe Scoreboard



Parallel operation in the control data 6600

The Scoreboard pipeline

ISSUE	READ OPERAND	EXE COMPLETE	WB
Decode instruction;	Read operands;	Operate on operands;	Finish exec;
Structural FUs check; WAW checks	RAW check; WAR if need to read	Notify Scoreboard on completion;	WAR & Struct check (FUs will hold results); Can overlap issue/read&write 4 Structural Hazard;

Exe.1 The code

```
I1: LD $F1, 0($R1)
I2: FADD $F2, $F2, $F3
I3: ADDI $R3, $R3, 8
I4: LD $F4, 0(R2)
I5: FADD $F5, $F4, $F2
I6: FMULT $F6, $F1, $F4
I7: ADDI $R5, $R5, 1
I8: LD $R6, 0($R4)
I9: SD $F6, 0($R5)
I10: SD $F5, 0($R6)
```

Exe.1 The conflicts

- I1: LD \$F1, 0(\$R1)
- I2: FADD \$F2, \$F2, \$F3
- I3: ADDI \$R3, \$R3, 8
- I4: LD \$F4, 0(R2)
- I5: FADD \$F5, \$F4, \$F2
- I6: FMULT \$F6, \$F1, \$F4
- I7: ADDI \$R5, \$R5, 1
- I8: LD \$R6, 0(\$R4)
- I9: SD \$F6, 0(\$R5)
- I10: SD \$F5, 0(\$R6)

Exe.1 The conflicts

I1: LD \$F1, 0(\$R1)
I2: FADD \$F2, \$F2, \$F3
I3: ADDI \$R3, \$R3, 8
I4: LD \$F4, 0(\$R2)
I5: FADD \$F5, \$F4, \$F2
I6: FMULT \$F6, \$F1, \$F4
I7: ADDI \$R5, \$R5, 1
I8: LD \$R6, 0(\$R4)
I9: SD \$F6, 0(\$R5)
I10: SD \$F5, 0(\$R6)

RAW F1 I1-I6

Exe.1 The conflicts

I1: LD \$F1, 0(\$R1)
I2: FADD \$F2, \$F2, \$F3
I3: ADDI \$R3, \$R3, 8
I4: LD \$F4, 0(\$R2)
I5: FADD \$F5, \$F4, \$F2
I6: FMULT \$F6, \$F1, \$F4
I7: ADDI \$R5, \$R5, 1
I8: LD \$R6, 0(\$R4)
I9: SD \$F6, 0(\$R5)
I10: SD \$F5, 0(\$R6)

RAW F1 I1-I6

RAW F2 I2-I5

Exe.1 The conflicts

I1: LD \$F1, 0(\$R1)
I2: FADD \$F2, \$F2, \$F3
I3: ADDI \$R3, \$R3, 8
I4: LD \$F4, 0(R2)
I5: FADD \$F5, \$F4, \$F2
I6: FMULT \$F6, \$F1, \$F4
I7: ADDI \$R5, \$R5, 1
I8: LD \$R6, 0(\$R4)
I9: SD \$F6, 0(\$R5)
I10: SD \$F5, 0(\$R6)

RAW F1 I1-I6

RAW F2 I2-I5

RAW F4 I4-I5

RAW F4 I4-I6

Exe.1 The conflicts

I1: LD \$F1, 0(\$R1)
I2: FADD \$F2, \$F2, \$F3
I3: ADDI \$R3, \$R3, 8
I4: LD \$F4, 0(\$R2)
I5: FADD \$F5, \$F4, \$F2
I6: FMULT \$F6, \$F1, \$F4
I7: ADDI \$R5, \$R5, 1
I8: LD \$R6, 0(\$R4)
I9: SD \$F6, 0(\$R5)
I10: SD \$F5, 0(\$R6)

RAW F1 I1-I6
RAW F2 I2-I5
RAW F4 I4-I5
RAW F4 I4-I6
RAW F5 I5-I10

Exe.1 The conflicts

I1: LD \$F1, 0(\$R1)
I2: FADD \$F2, \$F2, \$F3
I3: ADDI \$R3, \$R3, 8
I4: LD \$F4, 0(\$R2)
I5: FADD \$F5, \$F4, \$F2
I6: FMULT \$F6, \$F1, \$F4
I7: ADDI \$R5, \$R5, 1
I8: LD \$R6, 0(\$R4)
I9: SD \$F6, 0(\$R5)
I10: SD \$F5, 0(\$R6)

RAW F1 I1-I6
RAW F2 I2-I5
RAW F4 I4-I5
RAW F4 I4-I6
RAW F5 I5-I10
RAW F6 I6-I9

Exe.1 The conflicts

I1: LD \$F1, 0(\$R1)
I2: FADD \$F2, \$F2, \$F3
I3: ADDI \$R3, \$R3, 8
I4: LD \$F4, 0(\$R2)
I5: FADD \$F5, \$F4, \$F2
I6: FMULT \$F6, \$F1, \$F4
I7: ADDI \$R5, \$R5, 1
I8: LD \$R6, 0(\$R4)
I9: SD \$F6, 0(\$R5)
I10: SD \$F5, 0(\$R6)

RAW F1 I1-I6
RAW F2 I2-I5
RAW F4 I4-I5
RAW F4 I4-I6
RAW F5 I5-I10
RAW F6 I6-I9
RAW R5 I7-I9

Exe.1 The conflicts

I1: LD \$F1, 0(\$R1)
I2: FADD \$F2, \$F2, \$F3
I3: ADDI \$R3, \$R3, 8
I4: LD \$F4, 0(\$R2)
I5: FADD \$F5, \$F4, \$F2
I6: FMULT \$F6, \$F1, \$F4
I7: ADDI \$R5, \$R5, 1
I8: LD \$R6, 0(\$R4)
I9: SD \$F6, 0(\$R5)
I10: SD \$F5, 0(\$R6)

RAW F1 I1-I6
RAW F2 I2-I5
RAW F4 I4-I5
RAW F4 I4-I6
RAW F5 I5-I10
RAW F6 I6-I9
RAW R5 I7-I9
RAW R6 I8-I10

Exe.2 Scoreboard: \exists a configuration?

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB
I1	LD \$F1, 0(\$R1)	1	2	5	6
I2	FADD \$F2, \$F2, \$F3	2	3	7	8
I3	ADDI \$R3, \$R3, 8	3	4	5	6
I4	LD \$F4, 0(R2)	4	5	8	9
I5	FADD \$F5, \$F4, \$F2	9	6	14	15
I6	FMULT \$F6, \$F1, \$F4	8	10	14	16
I7	ADDI \$R5, \$R5, 1	5	8	9	10
I8	LD \$R6, 0(\$R4)	6	9	12	13
I9	SD \$F6, 0(\$R5)	9	11	14	17
I10	SD \$F5, 0(\$R6)	10	14	17	18

- Is there a “configuration” that can respect the shown execution?
- How many units? Which kind? What latency?

RAW F1 I1-I6

RAW F5 I5-I10

RAW F2 I2-I5

RAW F6 I6-I9

RAW F4 I4-I5

RAW R5 I7-I9

RAW F4 I4-I6

RAW R6 I8-I10

Exe.2 Scoreboard: \exists a configuration?

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB
I1	LD \$F1, 0(\$R1)	1	2	5	6
I2	FADD \$F2, \$F2, \$F3	2	3	7	8
I3	ADDI \$R3, \$R3, 8	3	4	5	6
I4	LD \$F4, 0(R2)	4	5	8	9
I5	FADD \$F5, \$F4, \$F2	9	6	14	15
I6	FMULT \$F6, \$F1, \$F4	8	10	14	16
I7	ADDI \$R5, \$R5, 1	5	8	9	10
I8	LD \$R6, 0(\$R4)	6	9	12	13
I9	SD \$F6, 0(\$R5)	9	11	14	17
I10	SD \$F5, 0(\$R6)	10	14	17	18

- Is there a “configuration” that can respect the shown execution?
- How many units? Which kind? What latency?

RAW F1 I1-I6

RAW F5 I5-I10

RAW F2 I2-I5

RAW F6 I6-I9

RAW F4 I4-I5

RAW R5 I7-I9

RAW F4 I4-I6

RAW R6 I8-I10

Exe.2 Scoreboard: \exists a configuration?

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB
I1	LD \$F1, 0(\$R1)	1	2	5	6
I2	FADD \$F2, \$F2, \$F3	2	3	7	8
I3	ADDI \$R3, \$R3, 8	3	4	5	6
I4	LD \$F4, 0(R2)	4	5	8	9
I5	FADD \$F5, \$F4, \$F2	9	6	14	15
I6	FMULT \$F6, \$F1, \$F4	8	10	14	16
I7	ADDI \$R5, \$R5, 1	5	8	9	10
I8	LD \$R6, 0(\$R4)	6	9	12	13
I9	SD \$F6, 0(\$R5)	9	11	14	17
I10	SD \$F5, 0(\$R6)	10	14	17	18

- Is there a “configuration” that can respect the shown execution?
- How many units? Which kind? What latency?

RAW F1 I1-I6

RAW F5 I5-I10

RAW F2 I2-I5

RAW F6 I6-I9

RAW F4 I4-I5

RAW R5 I7-I9

RAW F4 I4-I6

RAW R6 I8-I10

Exe.2 Scoreboard: \exists a configuration?

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB
I1	LD \$F1, 0(\$R1)	1	2	5	6
I2	FADD \$F2, \$F2, \$F3	2	3	7	8
I3	ADDI \$R3, \$R3, 8	3	4	5	6
I4	LD \$F4, 0(R2)	4	5	8	9
I5	FADD \$F5, \$F4, \$F2	9	6	14	15
I6	FMULT \$F6, \$F1, \$F4	8	10	14	16
I7	ADDI \$R5, \$R5, 1	5	8	9	10
I8	LD \$R6, 0(\$R4)	6	9	12	13
I9	SD \$F6, 0(\$R5)	9	11	14	17
I10	SD \$F5, 0(\$R6)	10	14	17	18

- Is there a “configuration” that can respect the shown execution?
- How many units? Which kind? What latency?

RAW F1 I1-I6

RAW F5 I5-I10

RAW F2 I2-I5

RAW F6 I6-I9

RAW F4 I4-I5

RAW R5 I7-I9

RAW F4 I4-I6

RAW R6 I8-I10

Exe.2 Scoreboard: \exists a configuration?

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB
I1	LD \$F1, 0(\$R1)	1	2	5	6
I2	FADD \$F2, \$F2, \$F3	2	3	7	8
I3	ADDI \$R3, \$R3, 8	3	4	5	6
I4	LD \$F4, 0(R2)	4	5	8	9
I5	FADD \$F5, \$F4, \$F2	9	6	14	15
I6	FMULT \$F6, \$F1, \$F4	8	10	14	16
I7	ADDI \$R5, \$R5, 1	5	8	9	10
I8	LD \$R6, 0(\$R4)	6	9	12	13
I9	SD \$F6, 0(\$R5)	9	11	14	17
I10	SD \$F5, 0(\$R6)	10	14	17	18

- Is there a “configuration” that can respect the shown execution?
- How many units? Which kind? What latency?

RAW F1 I1-I6

RAW F5 I5-I10

RAW F2 I2-I5

RAW F6 I6-I9

RAW F4 I4-I5

RAW R5 I7-I9

RAW F4 I4-I6

RAW R6 I8-I10

Exe.2 Scoreboard: \exists a configuration?

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB
I1	LD \$F1, 0(\$R1)	1	2	5	6
I2	FADD \$F2, \$F2, \$F3	2	3	7	8
I3	ADDI \$R3, \$R3, 8	3	4	5	6
I4	LD \$F4, 0(R2)	4	5	8	9
I5	FADD \$F5, \$F4, \$F2	9	6	14	15
I6	FMULT \$F6, \$F1, \$F4	8	10	14	16
I7	ADDI \$R5, \$R5, 1	5	8	9	10
I8	LD \$R6, 0(\$R4)	6	9	12	13
I9	SD \$F6, 0(\$R5)	9	11	14	17
I10	SD \$F5, 0(\$R6)	10	14	17	18

- Is there a “configuration” that can respect the shown execution?
- How many units? Which kind? What latency?

RAW F1 I1-I6

RAW F5 I5-I10

RAW F2 I2-I5

RAW F6 I6-I9

RAW F4 I4-I5

RAW R5 I7-I9

RAW F4 I4-I6

RAW R6 I8-I10

Exe.2 Scoreboard: \exists a configuration?

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB
I1	LD \$F1, 0(\$R1)	1	2	5	6
I2	FADD \$F2, \$F2, \$F3	2	3	7	8
I3	ADDI \$R3, \$R3, 8	3	4	5	6
I4	LD \$F4, 0(R2)	4	5	8	9
I5	FADD \$F5, \$F4, \$F2	9	6	14	15
I6	FMULT \$F6, \$F1, \$F4	8	10	14	16
I7	ADDI \$R5, \$R5, 1	5	8	9	10
I8	LD \$R6, 0(\$R4)	6	9	12	13
I9	SD \$F6, 0(\$R5)	9	11	14	17
I10	SD \$F5, 0(\$R6)	10	14	17	18

- Is there a “configuration” that can respect the shown execution?
- How many units? Which kind? What latency?

RAW F1 I1-I6 RAW F5 I5-I10

RAW F2 I2-I5 RAW F6 I6-I9

RAW F4 I4-I5 RAW R5 I7-I9

RAW F4 I4-I6 RAW R6 I8-I10

Exe.2 Scoreboard CCO

3 MU ,3cc
3 FPUs, 4cc
2 Integer ALU, 1cc
Single W port overall

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB	Hazards	Unit
I1	LD \$F1, 0(\$R1)						
I2	FADD \$F2, \$F2, \$F3						RAW F1 I1-I6
I3	ADDI \$R3, \$R3, 8						RAW F2 I2-I5
I4	LD \$F4, 0(R2)						RAW F4 I4-I5
I5	FADD \$F5, \$F4, \$F2						RAW F4 I4-I6
I6	FMULT \$F6, \$F1, \$F4						RAW F5 I5-I10
I7	ADDI \$R5, \$R5, 1						RAW F6 I6-I9
I8	LD \$R6, 0(\$R4)						RAW R5 I7-I9
I9	SD \$F6, 0(\$R5)						RAW R6 I8-I10
I10	SD \$F5, 0(\$R6)						

Exe.2 Scoreboard CC1

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB	Hazards	Unit
I1	LD \$F1, 0(\$R1)	1					MU1
I2	FADD \$F2, \$F2, \$F3						
I3	ADDI \$R3, \$R3, 8						
I4	LD \$F4, 0(R2)						
I5	FADD \$F5, \$F4, \$F2						
I6	FMULT \$F6, \$F1, \$F4						
I7	ADDI \$R5, \$R5, 1						
I8	LD \$R6, 0(\$R4)						
I9	SD \$F6, 0(\$R5)						
I10	SD \$F5, 0(\$R6)						

3 MU ,3cc
 3 FPUs, 4cc
 2 Integer ALU, 1cc
 Single W port overall

RAW F1 I1-I6
RAW F2 I2-I5
RAW F4 I4-I5
RAW F4 I4-I6
RAW F5 I5-I10
RAW F6 I6-I9
RAW R5 I7-I9
RAW R6 I8-I10

Exe.2 Scoreboard CC2

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB	Hazards	Unit
I1	LD \$F1, 0(\$R1)	1	2				MU1
I2	FADD \$F2, \$F2, \$F3	2					FDU1
I3	ADDI \$R3, \$R3, 8						
I4	LD \$F4, 0(R2)						
I5	FADD \$F5, \$F4, \$F2						
I6	FMULT \$F6, \$F1, \$F4						
I7	ADDI \$R5, \$R5, 1						
I8	LD \$R6, 0(\$R4)						
I9	SD \$F6, 0(\$R5)						
I10	SD \$F5, 0(\$R6)						

3 MU ,3cc
3 FPUs, 4cc
2 Integer ALU, 1cc
Single W port overall

RAW F1 I1-I6
RAW F2 I2-I5
RAW F4 I4-I5
RAW F4 I4-I6
RAW F5 I5-I10
RAW F6 I6-I9
RAW R5 I7-I9
RAW R6 I8-I10

Exe.2 Scoreboard CC3

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB	Hazards	Unit
I1	LD \$F1, 0(\$R1)	1	2				MU1
I2	FADD \$F2, \$F2, \$F3	2	3				FDU1
I3	ADDI \$R3, \$R3, 8	3					ALU1
I4	LD \$F4, 0(R2)						
I5	FADD \$F5, \$F4, \$F2						
I6	FMULT \$F6, \$F1, \$F4						
I7	ADDI \$R5, \$R5, 1						
I8	LD \$R6, 0(\$R4)						
I9	SD \$F6, 0(\$R5)						
I10	SD \$F5, 0(\$R6)						

3 MU ,3cc
3 FPUs, 4cc
2 Integer ALU, 1cc
Single W port overall

RAW F1 I1-I6
RAW F2 I2-I5
RAW F4 I4-I5
RAW F4 I4-I6
RAW F5 I5-I10
RAW F6 I6-I9
RAW R5 I7-I9
RAW R6 I8-I10

Exe.2 Scoreboard 4

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB	Hazards	Unit
I1	LD \$F1, 0(\$R1)	1	2				MU1
I2	FADD \$F2, \$F2, \$F3	2	3				FDU1
I3	ADDI \$R3, \$R3, 8	3	4				ALU1
I4	LD \$F4, 0(R2)	4					MU2
I5	FADD \$F5, \$F4, \$F2						
I6	FMULT \$F6, \$F1, \$F4						
I7	ADDI \$R5, \$R5, 1						
I8	LD \$R6, 0(\$R4)						
I9	SD \$F6, 0(\$R5)						
I10	SD \$F5, 0(\$R6)						

3 MU ,3cc
3 FPUs, 4cc
2 Integer ALU, 1cc
Single W port overall

RAW F0 I1-I6
RAW F2 I2-I5
RAW F4 I4-I5
RAW F4 I4-I6
RAW F5 I5-I10
RAW F6 I6-I9
RAW R5 I7-I9
RAW R6 I8-I10

Exe.2 Scoreboard 5

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB	Hazards	Unit
I1	LD \$F1, 0(\$R1)	1	2	5			MU1
I2	FADD \$F2, \$F2, \$F3	2	3				FDU1
I3	ADDI \$R3, \$R3, 8	3	4	5			ALU1
I4	LD \$F4, 0(R2)	4	5				MU2
I5	FADD \$F5, \$F4, \$F2	5					FPU2
I6	FMULT \$F6, \$F1, \$F4						
I7	ADDI \$R5, \$R5, 1						
I8	LD \$R6, 0(\$R4)						
I9	SD \$F6, 0(\$R5)						
I10	SD \$F5, 0(\$R6)						

3 MU ,3cc
3 FPUs, 4cc
2 Integer ALU, 1cc
Single W port overall

RAW F1 I1-I6
RAW F2 I2-I5
RAW F4 I4-I5
RAW F4 I4-I6
RAW F5 I5-I10
RAW F6 I6-I9
RAW R5 I7-I9
RAW R6 I8-I10

Exe.2 Scoreboard 6

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB	Hazards	Unit
I1	LD \$F1, 0(\$R1)	1	2	5	6		MU1
I2	FADD \$F2, \$F2, \$F3	2	3				FDU1
I3	ADDI \$R3, \$R3, 8	3	4	5		Struct RF	ALU1
I4	LD \$F4, 0(R2)	4	5				MU2
I5	FADD \$F5, \$F4, \$F2	5				RAW F2 + RAW F4	FPU2
I6	FMULT \$F6, \$F1, \$F4	6					FPU3
I7	ADDI \$R5, \$R5, 1						
I8	LD \$R6, 0(\$R4)						
I9	SD \$F6, 0(\$R5)						
I10	SD \$F5, 0(\$R6)						

3 MU ,3cc
3 FPUs, 4cc
2 Integer ALU, 1cc
Single W port overall

~~RAW F1 I1-I6~~
RAW F2 I2-I5
RAW F4 I4-I5
RAW F4 I4-I6
RAW F5 I5-I10
RAW F6 I6-I9
RAW R5 I7-I9
RAW R6 I8-I10

Exe.2 Scoreboard 7

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB	Hazards	Unit
I1	LD \$F1, 0(\$R1)	1	2	5	6		MU1
I2	FADD \$F2, \$F2, \$F3	2	3	7			FDU1
I3	ADDI \$R3, \$R3, 8	3	4	5	7	Struct RF	ALU1
I4	LD \$F4, 0(R2)	4	5				MU2
I5	FADD \$F5, \$F4, \$F2	5				RAW F2 + RAW F4	FPU2
I6	FMULT \$F6, \$F1, \$F4	6				RAW F4	FPU3
I7	ADDI \$R5, \$R5, 1	7					ALU2
I8	LD \$R6, 0(\$R4)						
I9	SD \$F6, 0(\$R5)						
I10	SD \$F5, 0(\$R6)						

3 MU ,3cc
3 FPUs, 4cc
2 Integer ALU, 1cc
Single W port overall

~~RAW F1 I1-I6~~
RAW F2 I2-I5
RAW F4 I4-I5
RAW F4 I4-I6
RAW F5 I5-I10
RAW F6 I6-I9
RAW R5 I7-I9
RAW R6 I8-I10

Exe.2 Scoreboard 8

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB	Hazards	Unit
I1	LD \$F1, 0(\$R1)	1	2	5	6		MU1
I2	FADD \$F2, \$F2, \$F3	2	3	7	8		FDU1
I3	ADDI \$R3, \$R3, 8	3	4	5	7	Struct RF	ALU1
I4	LD \$F4, 0(R2)	4	5	8	9		MU2
I5	FADD \$F5, \$F4, \$F2	5				RAW F2 + RAW F4	FPU2
I6	FMULT \$F6, \$F1, \$F4	6				RAW F4	FPU3
I7	ADDI \$R5, \$R5, 1	7	8	9			ALU2
I8	LD \$R6, 0(\$R4)	8	9				MU3
I9	SD \$F6, 0(\$R5)	9					MU1
I10	SD \$F5, 0(\$R6)						

3 MU ,3cc
 3 FPUs, 4cc
 2 Integer ALU, 1cc
 Single W port overall

~~RAW F1 I1-I6~~

~~RAW F2 I2-I5~~

RAW F4 I4-I5

RAW F4 I4-I6

RAW F5 I5-I10

RAW F6 I6-I9

RAW R5 I7-I9

RAW R6 I8-I10

Exe.2 Scoreboard 9

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB	Hazards	Unit
I1	LD \$F1, 0(\$R1)	1	2	5	6		MU1
I2	FADD \$F2, \$F2, \$F3	2	3	7	8		FDU1
I3	ADDI \$R3, \$R3, 8	3	4	5	7	Struct RF	ALU1
I4	LD \$F4, 0(R2)	4	5	8			MU2
I5	FADD \$F5, \$F4, \$F2	5				RAW F2 + RAW F4	FPU2
I6	FMULT \$F6, \$F1, \$F4	6				RAW F4	FPU3
I7	ADDI \$R5, \$R5, 1	7	8				ALU2
I8	LD \$R6, 0(\$R4)	8					MU3
I9	SD \$F6, 0(\$R5)						
I10	SD \$F5, 0(\$R6)						

3 MU ,3cc
 3 FPUs, 4cc
 2 Integer ALU, 1cc
 Single W port overall

~~RAW F1 I1-I6~~
~~RAW F2 I2-I5~~
~~RAW F4 I4-I5~~
~~RAW F4 I4-I6~~
 RAW F5 I5-I10
 RAW F6 I6-I9
 RAW R5 I7-I9
 RAW R6 I8-I10

Exe.2 Scoreboard 10

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB	Hazards	Unit
I1	LD \$F1, 0(\$R1)	1	2	5	6		MU1
I2	FADD \$F2, \$F2, \$F3	2	3	7	8		FDU1
I3	ADDI \$R3, \$R3, 8	3	4	5	7	Struct RF	ALU1
I4	LD \$F4, 0(R2)	4	5	8	9		MU2
I5	FADD \$F5, \$F4, \$F2	5	10			RAW F2 + RAW F4	FPU2
I6	FMULT \$F6, \$F1, \$F4	6	10			RAW F4	FPU3
I7	ADDI \$R5, \$R5, 1	7	8	9	10		ALU2
I8	LD \$R6, 0(\$R4)	8	9				MU3
I9	SD \$F6, 0(\$R5)	9				RAW R5 + RAW F6	MU1
I10	SD \$F5, 0(\$R6)	10					MU2

3 MU ,3cc
 3 FPUs, 4cc
 2 Integer ALU, 1cc
 Single W port overall

~~RAW F1 I1-I6~~
~~RAW F2 I2-I5~~
~~RAW F4 I4-I5~~
~~RAW F4 I4-I6~~
RAW F5 I5-I10
RAW F6 I6-I9
~~RAW R5 I7-I9~~
RAW R6 I8-I10

Exe.2 Scoreboard 12

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB	Hazards	Unit
I1	LD \$F1, 0(\$R1)	1	2	5	6		MU1
I2	FADD \$F2, \$F2, \$F3	2	3	7	8		FDU1
I3	ADDI \$R3, \$R3, 8	3	4	5	7	Struct RF	ALU1
I4	LD \$F4, 0(R2)	4	5	8	9		MU2
I5	FADD \$F5, \$F4, \$F2	5	10			RAW F2 + RAW F4	FPU2
I6	FMULT \$F6, \$F1, \$F4	6	10			RAW F4	FPU3
I7	ADDI \$R5, \$R5, 1	7	8	9	10		ALU2
I8	LD \$R6, 0(\$R4)	8	9	12			MU3
I9	SD \$F6, 0(\$R5)	9				RAW R5 + RAW F6	MU1
I10	SD \$F5, 0(\$R6)	10				RAW R6 + RAW F5	MU2

3 MU ,3cc
 3 FPUs, 4cc
 2 Integer ALU, 1cc
 Single W port overall

~~RAW F1 I1-I6~~

~~RAW F2 I2-I5~~

~~RAW F4 I4-I5~~

~~RAW F4 I4-I6~~

RAW F5 I5-I10

RAW F6 I6-I9

~~RAW R5 I7-I9~~

RAW R6 I8-I10

Exe.2 Scoreboard 13

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB	Hazards	Unit
I1	LD \$F1, 0(\$R1)	1	2	5	6		MU1
I2	FADD \$F2, \$F2, \$F3	2	3	7	8		FDU1
I3	ADDI \$R3, \$R3, 8	3	4	5	7	Struct RF	ALU1
I4	LD \$F4, 0(R2)	4	5	8	9		MU2
I5	FADD \$F5, \$F4, \$F2	5	10			RAW F2 + RAW F4	FPU2
I6	FMULT \$F6, \$F1, \$F4	6	10			RAW F4	FPU3
I7	ADDI \$R5, \$R5, 1	7	8	9	10		ALU2
I8	LD \$R6, 0(\$R4)	8	9	12	13		MU3
I9	SD \$F6, 0(\$R5)	9				RAW R5 + RAW F6	MU1
I10	SD \$F5, 0(\$R6)	10				RAW R6 + RAW F5	MU2

3 MU ,3cc
 3 FPUs, 4cc
 2 Integer ALU, 1cc
 Single W port overall

~~RAW F1 I1-I6~~
~~RAW F2 I2-I5~~
~~RAW F4 I4-I5~~
~~RAW F4 I4-I6~~
RAW F5 I5-I10
RAW F6 I6-I9
~~RAW R5 I7-I9~~
~~RAW R6 I8-I10~~

Exe.2 Scoreboard 14

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB	Hazards	Unit
I1	LD \$F1, 0(\$R1)	1	2	5	6		MU1
I2	FADD \$F2, \$F2, \$F3	2	3	7	8		FDU1
I3	ADDI \$R3, \$R3, 8	3	4	5	7	Struct RF	ALU1
I4	LD \$F4, 0(R2)	4	5	8	9		MU2
I5	FADD \$F5, \$F4, \$F2	5	10	14		RAW F2 + RAW F4	FPU2
I6	FMULT \$F6, \$F1, \$F4	6	10	14		RAW F4	FPU3
I7	ADDI \$R5, \$R5, 1	7	8	9	10		ALU2
I8	LD \$R6, 0(\$R4)	8	9	12	13		MU3
I9	SD \$F6, 0(\$R5)	9				RAW R5 + RAW F6	MU1
I10	SD \$F5, 0(\$R6)	10				RAW R6 + RAW F5	MU2

3 MU ,3cc
 3 FPUs, 4cc
 2 Integer ALU, 1cc
 Single W port overall

~~RAW F1 I1-I6~~
~~RAW F2 I2-I5~~
~~RAW F4 I4-I5~~
~~RAW F4 I4-I6~~
RAW F5 I5-I10
RAW F6 I6-I9
~~RAW R5 I7-I9~~
~~RAW R6 I8-I10~~

Exe.2 Scoreboard 15

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB	Hazards	Unit
I1	LD \$F1, 0(\$R1)	1	2	5	6		MU1
I2	FADD \$F2, \$F2, \$F3	2	3	7	8		FDU1
I3	ADDI \$R3, \$R3, 8	3	4	5	7	Struct RF	ALU1
I4	LD \$F4, 0(R2)	4	5	8	9		MU2
I5	FADD \$F5, \$F4, \$F2	5	10	14	15	RAW F2 + RAW F4	FPU2
I6	FMULT \$F6, \$F1, \$F4	6	10	14		RAW F4 + Struct RF	FPU3
I7	ADDI \$R5, \$R5, 1	7	8	9	10		ALU2
I8	LD \$R6, 0(\$R4)	8	9	12	13		MU3
I9	SD \$F6, 0(\$R5)	9				RAW R5 + RAW F6	MU1
I10	SD \$F5, 0(\$R6)	10				RAW R6 + RAW F5	MU2

3 MU ,3cc
3 FPUs, 4cc
2 Integer ALU, 1cc
Single W port overall

~~RAW F1 I1-I6~~
~~RAW F2 I2-I5~~
~~RAW F4 I4-I5~~
~~RAW F4 I4-I6~~
RAW F5 I5-I10
RAW F6 I6-I9
~~RAW R5 I7-I9~~
~~RAW R6 I8-I10~~

Exe.2 Scoreboard 16

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB	Hazards	Unit
I1	LD \$F1, 0(\$R1)	1	2	5	6		MU1
I2	FADD \$F2, \$F2, \$F3	2	3	7	8		FDU1
I3	ADDI \$R3, \$R3, 8	3	4	5	7	Struct RF	ALU1
I4	LD \$F4, 0(R2)	4	5	8	9		MU2
I5	FADD \$F5, \$F4, \$F2	5	10	14	15	RAW F2 + RAW F4	FPU2
I6	FMULT \$F6, \$F1, \$F4	6	10	14	16	RAW F4 + Struct RF	FPU3
I7	ADDI \$R5, \$R5, 1	7	8	9	10		ALU2
I8	LD \$R6, 0(\$R4)	8	9	12	13		MU3
I9	SD \$F6, 0(\$R5)	9				RAW R5 + RAW F6	MU1
I10	SD \$F5, 0(\$R6)	10	16			RAW R6 + RAW F5	MU2

3 MU ,3cc
 3 FPUs, 4cc
 2 Integer ALU, 1cc
 Single W port overall

~~RAW F1 I1-I6~~
~~RAW F2 I2-I5~~
~~RAW F4 I4-I5~~
~~RAW F4 I4-I6~~
~~RAW F5 I5-I10~~
 RAW F6 I6-I9
~~RAW R5 I7-I9~~
~~RAW R6 I8-I10~~

Exe.2 Scoreboard 17

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB	Hazards	Unit
I1	LD \$F1, 0(\$R1)	1	2	5	6		MU1
I2	FADD \$F2, \$F2, \$F3	2	3	7	8		FDU1
I3	ADDI \$R3, \$R3, 8	3	4	5	7	Struct RF	ALU1
I4	LD \$F4, 0(R2)	4	5	8	9		MU2
I5	FADD \$F5, \$F4, \$F2	5	10	14	15	RAW F2 + RAW F4	FPU2
I6	FMULT \$F6, \$F1, \$F4	6	10	14	16	RAW F4 + Struct RF	FPU3
I7	ADDI \$R5, \$R5, 1	7	8	9	10		ALU2
I8	LD \$R6, 0(\$R4)	8	9	12	13		MU3
I9	SD \$F6, 0(\$R5)	9	17			RAW R5 + RAW F6	MU1
I10	SD \$F5, 0(\$R6)	10	16			RAW R6 + RAW F5	MU2

3 MU ,3cc
3 FPUs, 4cc
2 Integer ALU, 1cc
Single W port overall

~~RAW F1 I1-I6~~
~~RAW F2 I2-I5~~
~~RAW F4 I4-I5~~
~~RAW F4 I4-I6~~
~~RAW F5 I5-I10~~
~~RAW F6 I6-I9~~
~~RAW R5 I7-I9~~
~~RAW R6 I8-I10~~

Exe.2 Scoreboard 19

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB	Hazards	Unit
I1	LD \$F1, 0(\$R1)	1	2	5	6		MU1
I2	FADD \$F2, \$F2, \$F3	2	3	7	8		FDU1
I3	ADDI \$R3, \$R3, 8	3	4	5	7	Struct RF	ALU1
I4	LD \$F4, 0(R2)	4	5	8	9		MU2
I5	FADD \$F5, \$F4, \$F2	5	10	14	15	RAW F2 + RAW F4	FPU2
I6	FMULT \$F6, \$F1, \$F4	6	10	14	16	RAW F4 + Struct RF	FPU3
I7	ADDI \$R5, \$R5, 1	7	8	9	10		ALU2
I8	LD \$R6, 0(\$R4)	8	9	12	13		MU3
I9	SD \$F6, 0(\$R5)	9	17			RAW R5 + RAW F6	MU1
I10	SD \$F5, 0(\$R6)	10	16	19		RAW R6 + RAW F5	MU2

3 MU ,3cc
 3 FPUs, 4cc
 2 Integer ALU, 1cc
 Single W port overall

~~RAW F1 I1-I6~~
~~RAW F2 I2-I5~~
~~RAW F4 I4-I5~~
~~RAW F4 I4-I6~~
~~RAW F5 I5-I10~~
~~RAW F6 I6-I9~~
~~RAW R5 I7-I9~~
~~RAW R6 I8-I10~~

3 MU ,3cc
3 FPUs, 4cc
2 Integer ALU, 1cc
Single W port overall

Exe.2 Scoreboard 20

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB	Hazards	Unit
I1	LD \$F1, 0(\$R1)	1	2	5	6		MU1
I2	FADD \$F2, \$F2, \$F3	2	3	7	8		FDU1
I3	ADDI \$R3, \$R3, 8	3	4	5	7	Struct RF	ALU1
I4	LD \$F4, 0(R2)	4	5	8	9		MU2
I5	FADD \$F5, \$F4, \$F2	5	10	14	15	RAW F2 + RAW F4	FPU2
I6	FMULT \$F6, \$F1, \$F4	6	10	14	16	RAW F4 + Struct RF	FPU3
I7	ADDI \$R5, \$R5, 1	7	8	9	10		ALU2
I8	LD \$R6, 0(\$R4)	8	9	12	13		MU3
I9	SD \$F6, 0(\$R5)	9	17	20		RAW R5 + RAW F6	MU1
I10	SD \$F5, 0(\$R6)	10	16	19	20	RAW R6 + RAW F5	MU2

~~RAW F1 I1-I6~~
~~RAW F2 I2-I5~~
~~RAW F4 I4-I5~~
~~RAW F4 I4-I6~~
~~RAW F5 I5-I10~~
~~RAW F6 I6-I9~~
~~RAW R5 I7-I9~~
~~RAW R6 I8-I10~~

3 MU ,3cc
3 FPUs, 4cc
2 Integer ALU, 1cc
Single W port overall

Exe.2 Scoreboard 21

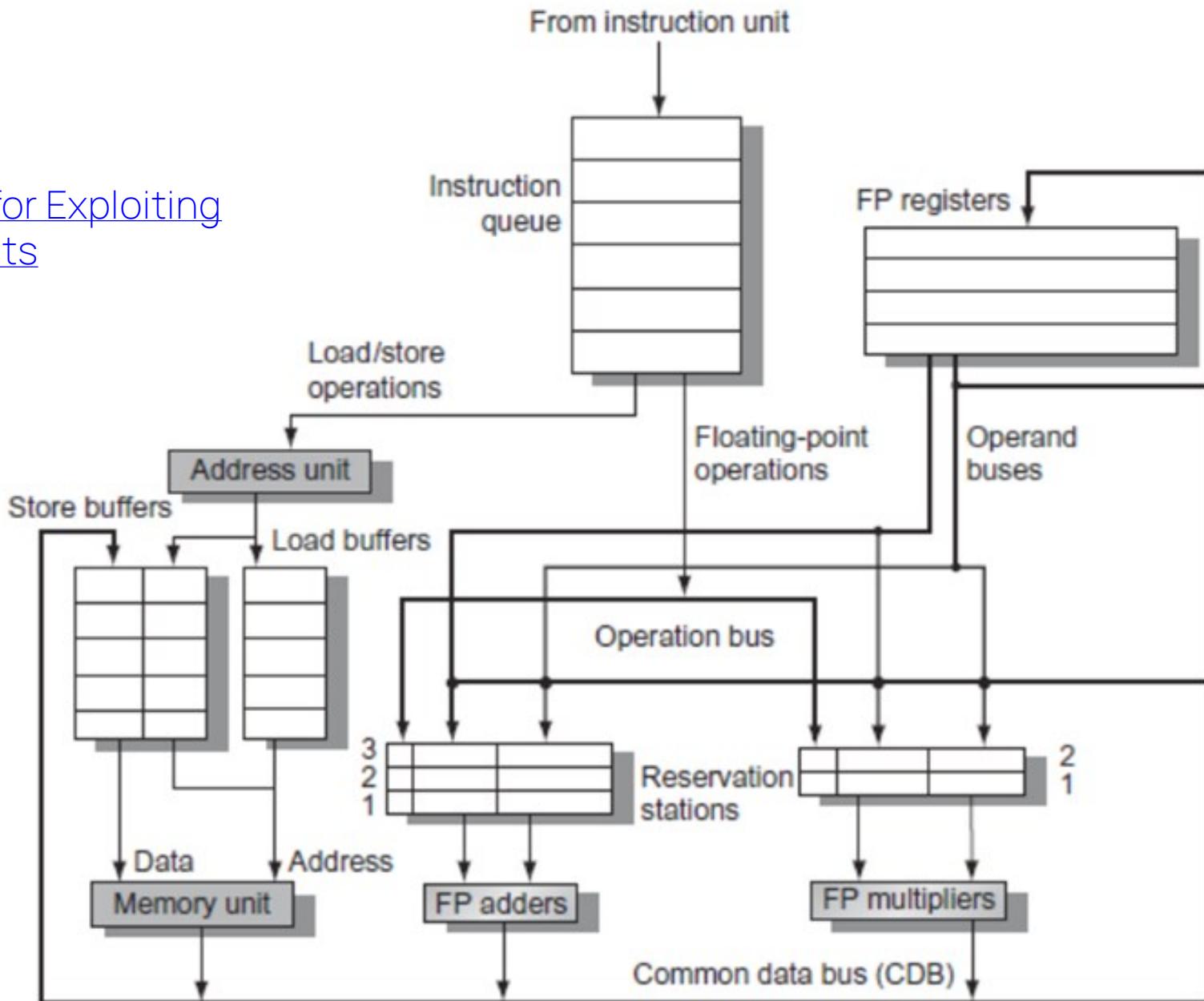
	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB	Hazards	Unit
I1	LD \$F1, 0(\$R1)	1	2	5	6		MU1
I2	FADD \$F2, \$F2, \$F3	2	3	7	8		FDU1
I3	ADDI \$R3, \$R3, 8	3	4	5	7	Struct RF	ALU1
I4	LD \$F4, 0(R2)	4	5	8	9		MU2
I5	FADD \$F5, \$F4, \$F2	5	10	14	15	RAW F2 + RAW F4	FPU2
I6	FMULT \$F6, \$F1, \$F4	6	10	14	16	RAW F4 + Struct RF	FPU3
I7	ADDI \$R5, \$R5, 1	7	8	9	10		ALU2
I8	LD \$R6, 0(\$R4)	8	9	12	13		MU3
I9	SD \$F6, 0(\$R5)	9	17	20	21	RAW R5 + RAW F6	MU1
I10	SD \$F5, 0(\$R6)	10	16	19	20	RAW R6 + RAW F5	MU2

~~RAW F1 I1-I6~~
~~RAW F2 I2-I5~~
~~RAW F4 I4-I5~~
~~RAW F4 I4-I6~~
~~RAW F5 I5-I10~~
~~RAW F6 I6-I9~~
~~RAW R5 I7-I9~~
~~RAW R6 I8-I10~~



Exe Tomasulo

An Efficient Algorithm for Exploiting
Multiple Arithmetic Units



Exe.1 The conflicts

- I1: LD \$F1, 0(\$R1)
- I2: FADD \$F2, \$F2, \$F3
- I3: ADDI \$R3, \$R3, 8
- I4: LD \$F4, 0(R2)
- I5: FADD \$F5, \$F4, \$F2
- I6: FMULT \$F6, \$F1, \$F4
- I7: ADDI \$R5, \$R5, 1
- I8: LD \$R6, 0(\$R4)
- I9: SD \$F6, 0(\$R5)
- I10: SD \$F5, 0(\$R6)

Exe.1 The conflicts

I1: LD \$F1, 0(\$R1)
I2: FADD \$F2, \$F2, \$F3
I3: ADDI \$R3, \$R3, 8
I4: LD \$F4, 0(\$R2)
I5: FADD \$F5, \$F4, \$F2
I6: FMULT \$F6, \$F1, \$F4
I7: ADDI \$R5, \$R5, 1
I8: LD \$R6, 0(\$R4)
I9: SD \$F6, 0(\$R5)
I10: SD \$F5, 0(\$R6)

RAW F1 I1-I6
RAW F2 I2-I5
RAW F4 I4-I5
RAW F4 I4-I6
RAW F5 I5-I10
RAW F6 I6-I9
RAW R5 I7-I9
RAW R6 I8-I10

Recall: the Tomasulo pipeline

ISSUE	EXECUTION	WRITE
Get Instruction from Queue and Rename Registers	Execute and Watch CDB;	Write on CDB;
Structural RSs check; WAW and WAR solved by Renaming (!!!in-order-issue!!!);	Check for Struct on FUs; RAW delaying; Struct check on CDB;	(FUs will hold results unless CDB free) RSs/FUs marked free

Exe.2 Tomasulo: \exists a configuration?

Instruction	ISSUE	START EXE	WB
I1: LD \$F1, 0(\$R1)	1	2	5
I2: FADD \$F2, \$F2, \$F3	2	3	6
I3: ADDI \$R3, \$R3, 8	3	4	6
I4: LD \$F4, 0(R2)	6	5	8
I5: FADD \$F5, \$F4, \$F2	7	9	12
I6: FMULT \$F6, \$F1, \$F4	8	9	13
I7: ADDI \$R5, \$R5, 1	9	10	11
I8: LD \$R6, 0(\$R4)	10	9	14
I9: SD \$F6, 0(\$R5)	11	14	17
I10: SD \$F5, 0(\$R6)	12	15	18

- Is there a “configuration” that can respect the shown execution?
- How many units? Which kind? What latency? How many Reservation Stations?

Exe.2 Tomasulo: \exists a configuration?

Instruction	ISSUE	START EXE	WB
I1: LD \$F1, 0(\$R1)	1	2	5
I2: FADD \$F2, \$F2, \$F3	2	3	6
I3: ADDI \$R3, \$R3, 8	3	4	6
I4: LD \$F4, 0(R2)	6	5	8
I5: FADD \$F5, \$F4, \$F2	7	9	12
I6: FMULT \$F6, \$F1, \$F4	8	9	13
I7: ADDI \$R5, \$R5, 1	9	10	11
I8: LD \$R6, 0(\$R4)	10	9	14
I9: SD \$F6, 0(\$R5)	11	14	17
I10: SD \$F5, 0(\$R6)	12	15	18

- Is there a “configuration” that can respect the shown execution?
- How many units? Which kind? What latency? How many Reservation Stations?

Exe.2 Tomasulo: \exists a configuration?

Instruction	ISSUE	START EXE	WB
I1: LD \$F1, 0(\$R1)	1	2	5
I2: FADD \$F2, \$F2, \$F3	2	3	6
I3: ADDI \$R3, \$R3, 8	3	4	6
I4: LD \$F4, 0(R2)	6	5	8
I5: FADD \$F5, \$F4, \$F2	7	9	12
I6: FMULT \$F6, \$F1, \$F4	8	9	13
I7: ADDI \$R5, \$R5, 1	9	10	11
I8: LD \$R6, 0(\$R4)	10	9	14
I9: SD \$F6, 0(\$R5)	11	14	17
I10: SD \$F5, 0(\$R6)	12	15	18

- Is there a “configuration” that can respect the shown execution?
- How many units? Which kind? What latency? How many Reservation Stations?

Exe.3 Tomasulo config

- 3 RESERVATION STATIONS (RS1, RS2, RS3) + 3 LOAD/STORE unit (LDU1, LDU2, LDU3) with latency 3
- 3 RESERVATION STATIONS (RS4, RS5, RS6) + 3 FPUs (FPU1, FPU2, FPU3) with latency 3
- 2 RESERVATION STATIONS (RS7, RS8) + 1 Integer ALU (ALU1) with latency 1

Exe.3 Tomasulo CCO

- 3 RESERVATION STATIONS (RS1, RS2, RS3) + 3 LOAD/STORE unit (LDU1, LDU2, LDU3) with latency 3
- 3 RESERVATION STATIONS (RS4, RS5, RS6) + 3 FPUs (FPU1, FPU2, FPU3) with latency 3
- 2 RESERVATION STATIONS (RS7, RS8) + 1 Integer ALU (ALU1) with latency 1

RAW F1 I1-I6

RAW F5 I5-I10

RAW F2 I2-I5

RAW F6 I6-I9

RAW F4 I4-I6

RAW R5 I7-I9

RAW F4 I4-I5

RAW R6 I8-I10

	Instruction	ISSUE	START EXE	WB	Hazards Type	RSi	Unit
	I1: LD \$F1, 0(\$R1)						
	I2: FADD \$F2, \$F2, \$F3						
	I3: ADDI \$R3, \$R3, 8						
	I4: LD \$F4, 0(R2)						
	I5: FADD \$F5, \$F4, \$F2						
	I6: FMULT \$F6, \$F1, \$F4						
	I7: ADDI \$R5, \$R5, 1						
	I8: LD \$R6, 0(\$R4)						
	I9: SD \$F6, 0(\$R5)						
	I10: SD \$F5, 0(\$R6)						

Exe.3 Tomasulo CC1

- 3 RESERVATION STATIONS (RS1, RS2, RS3) + 3 LOAD/STORE unit (LDU1, LDU2, LDU3) with latency 3
- 3 RESERVATION STATIONS (RS4, RS5, RS6) + 3 FPUs (FPU1, FPU2, FPU3) with latency 3
- 2 RESERVATION STATIONS (RS7, RS8) + 1 Integer ALU (ALU1) with latency 1

RAW F1 I1-I6

RAW F5 I5-I10

RAW F2 I2-I5

RAW F6 I6-I9

RAW F4 I4-I6

RAW R5 I7-I9

RAW F4 I4-I5

RAW R6 I8-I10

	Instruction	ISSUE	START EXE	WB	Hazards Type	RSi	Unit
	I1: LD \$F1, 0(\$R1)	1				RS1	
	I2: FADD \$F2, \$F2, \$F3						
	I3: ADDI \$R3, \$R3, 8						
	I4: LD \$F4, 0(R2)						
	I5: FADD \$F5, \$F4, \$F2						
	I6: FMULT \$F6, \$F1, \$F4						
	I7: ADDI \$R5, \$R5, 1						
	I8: LD \$R6, 0(\$R4)						
	I9: SD \$F6, 0(\$R5)						
	I10: SD \$F5, 0(\$R6)						

Exe.3 Tomasulo CC2

- 3 RESERVATION STATIONS (RS1, RS2, RS3) + 3 LOAD/STORE unit (LDU1, LDU2, LDU3) with latency 3
- 3 RESERVATION STATIONS (RS4, RS5, RS6) + 3 FPUs (FPU1, FPU2, FPU3) with latency 3
- 2 RESERVATION STATIONS (RS7, RS8) + 1 Integer ALU (ALU1) with latency 1

RAW F1 I1-I6

RAW F5 I5-I10

RAW F2 I2-I5

RAW F6 I6-I9

RAW F4 I4-I6

RAW R5 I7-I9

RAW F4 I4-I5

RAW R6 I8-I10

	Instruction	ISSUE	START EXE	WB	Hazards Type	RSi	Unit
	I1: LD \$F1, 0(\$R1)	1	2			RS1	LDU1
	I2: FADD \$F2, \$F2, \$F3	2				RS4	
	I3: ADDI \$R3, \$R3, 8						
	I4: LD \$F4, 0(R2)						
	I5: FADD \$F5, \$F4, \$F2						
	I6: FMULT \$F6, \$F1, \$F4						
	I7: ADDI \$R5, \$R5, 1						
	I8: LD \$R6, 0(\$R4)						
	I9: SD \$F6, 0(\$R5)						
	I10: SD \$F5, 0(\$R6)						

Exe.3 Tomasulo CC3

- 3 RESERVATION STATIONS (RS1, RS2, RS3) + 3 LOAD/STORE unit (LDU1, LDU2, LDU3) with latency 3
- 3 RESERVATION STATIONS (RS4, RS5, RS6) + 3 FPUs (FPU1, FPU2, FPU3) with latency 3
- 2 RESERVATION STATIONS (RS7, RS8) + 1 Integer ALU (ALU1) with latency 1

RAW F1 I1-I6

RAW F5 I5-I10

RAW F2 I2-I5

RAW F6 I6-I9

RAW F4 I4-I6

RAW R5 I7-I9

RAW F4 I4-I5

RAW R6 I8-I10

	Instruction	ISSUE	START EXE	WB	Hazards Type	RSi	Unit
	I1: LD \$F1, 0(\$R1)	1	2			RS1	LDU1
	I2: FADD \$F2, \$F2, \$F3	2	3			RS4	FPU1
	I3: ADDI \$R3, \$R3, 8	3				RS7	
	I4: LD \$F4, 0(R2)						
	I5: FADD \$F5, \$F4, \$F2						
	I6: FMULT \$F6, \$F1, \$F4						
	I7: ADDI \$R5, \$R5, 1						
	I8: LD \$R6, 0(\$R4)						
	I9: SD \$F6, 0(\$R5)						
	I10: SD \$F5, 0(\$R6)						

Exe.3 Tomasulo CC4

- 3 RESERVATION STATIONS (RS1, RS2, RS3) + 3 LOAD/STORE unit (LDU1, LDU2, LDU3) with latency 3
- 3 RESERVATION STATIONS (RS4, RS5, RS6) + 3 FPUs (FPU1, FPU2, FPU3) with latency 3
- 2 RESERVATION STATIONS (RS7, RS8) + 1 Integer ALU (ALU1) with latency 1

RAW F1 I1-I6

RAW F5 I5-I10

RAW F2 I2-I5

RAW F6 I6-I9

RAW F4 I4-I6

RAW R5 I7-I9

RAW F4 I4-I5

RAW R6 I8-I10

	Instruction	ISSUE	START EXE	WB	Hazards Type	RSi	Unit
	I1: LD \$F1, 0(\$R1)	1	2			RS1	LDU1
	I2: FADD \$F2, \$F2, \$F3	2	3			RS4	FPU1
	I3: ADDI \$R3, \$R3, 8	3	4			RS7	ALU1
	I4: LD \$F4, 0(R2)	4				RS2	
	I5: FADD \$F5, \$F4, \$F2						
	I6: FMULT \$F6, \$F1, \$F4						
	I7: ADDI \$R5, \$R5, 1						
	I8: LD \$R6, 0(\$R4)						
	I9: SD \$F6, 0(\$R5)						
	I10: SD \$F5, 0(\$R6)						

Exe.3 Tomasulo CC5

- 3 RESERVATION STATIONS (RS1, RS2, RS3) + 3 LOAD/STORE unit (LDU1, LDU2, LDU3) with latency 3
- 3 RESERVATION STATIONS (RS4, RS5, RS6) + 3 FPUs (FPU1, FPU2, FPU3) with latency 3
- 2 RESERVATION STATIONS (RS7, RS8) + 1 Integer ALU (ALU1) with latency 1

~~RAW F1 I1-I6~~

RAW F5 I5-I10

	Instruction	ISSUE	START EXE	WB	Hazards Type	RSi	Unit
RAW F2 I2-I5	I1: LD \$F1, 0(\$R1)	1	2	5		RS1	LDU1
RAW F6 I6-I9	I2: FADD \$F2, \$F2, \$F3	2	3			RS4	FPU1
RAW F4 I4-I6	I3: ADDI \$R3, \$R3, 8	3	4		Struct CDB	RS7	ALU1
RAW R5 I7-I9	I4: LD \$F4, 0(R2)	4	5			RS2	LDU2
RAW F4 I4-I5	I5: FADD \$F5, \$F4, \$F2	5				RS5	
RAW R6 I8-I10	I6: FMULT \$F6, \$F1, \$F4						
	I7: ADDI \$R5, \$R5, 1						
	I8: LD \$R6, 0(\$R4)						
	I9: SD \$F6, 0(\$R5)						
	I10: SD \$F5, 0(\$R6)						

Exe.3 Tomasulo CC6

- 3 RESERVATION STATIONS (RS1, RS2, RS3) + 3 LOAD/STORE unit (LDU1, LDU2, LDU3) with latency 3
- 3 RESERVATION STATIONS (RS4, RS5, RS6) + 3 FPUs (FPU1, FPU2, FPU3) with latency 3
- 2 RESERVATION STATIONS (RS7, RS8) + 1 Integer ALU (ALU1) with latency 1

~~RAW F1 I1-I6~~

~~RAW F5 I5-I10~~

~~RAW F2 I2-I5~~

~~RAW F6 I6-I9~~

~~RAW F4 I4-I6~~

~~RAW R5 I7-I9~~

~~RAW F4 I4-I5~~

~~RAW R6 I8-I10~~

	Instruction	ISSUE	START EXE	WB	Hazards Type	RSi	Unit
	I1: LD \$F1, 0(\$R1)	1	2	5		RS1	LDU1
	I2: FADD \$F2, \$F2, \$F3	2	3	6		RS4	FPU1
	I3: ADDI \$R3, \$R3, 8	3	4		Struct CDB	RS7	ALU1
	I4: LD \$F4, 0(R2)	4	5			RS2	LDU2
	I5: FADD \$F5, \$F4, \$F2	5			RAW \$F4, RAW \$F2	RS5	
	I6: FMULT \$F6, \$F1, \$F4	6				RS6	
	I7: ADDI \$R5, \$R5, 1						
	I8: LD \$R6, 0(\$R4)						
	I9: SD \$F6, 0(\$R5)						
	I10: SD \$F5, 0(\$R6)						

Exe.3 Tomasulo CC7

- 3 RESERVATION STATIONS (RS1, RS2, RS3) + 3 LOAD/STORE unit (LDU1, LDU2, LDU3) with latency 3
- 3 RESERVATION STATIONS (RS4, RS5, RS6) + 3 FPUs (FPU1, FPU2, FPU3) with latency 3
- 2 RESERVATION STATIONS (RS7, RS8) + 1 Integer ALU (ALU1) with latency 1

~~RAW F1 I1-I6~~

~~RAW F5 I5-I10~~

~~RAW F2 I2-I5~~

~~RAW F6 I6-I9~~

~~RAW F4 I4-I6~~

~~RAW R5 I7-I9~~

~~RAW F4 I4-I5~~

~~RAW R6 I8-I10~~

	Instruction	ISSUE	START EXE	WB	Hazards Type	RSi	Unit
	I1: LD \$F1, 0(\$R1)	1	2	5		RS1	LDU1
	I2: FADD \$F2, \$F2, \$F3	2	3	6		RS4	FPU1
	I3: ADDI \$R3, \$R3, 8	3	4	7	Struct CDB	RS7	ALU1
	I4: LD \$F4, 0(R2)	4	5			RS2	LDU2
	I5: FADD \$F5, \$F4, \$F2	5			RAW \$F4, RAW \$F2	RS5	
	I6: FMULT \$F6, \$F1, \$F4	6			RAW \$F4	RS6	
	I7: ADDI \$R5, \$R5, 1	7				RS8	
	I8: LD \$R6, 0(\$R4)						
	I9: SD \$F6, 0(\$R5)						
	I10: SD \$F5, 0(\$R6)						

Exe.3 Tomasulo CC8

- 3 RESERVATION STATIONS (RS1, RS2, RS3) + 3 LOAD/STORE unit (LDU1, LDU2, LDU3) with latency 3
- 3 RESERVATION STATIONS (RS4, RS5, RS6) + 3 FPUs (FPU1, FPU2, FPU3) with latency 3
- 2 RESERVATION STATIONS (RS7, RS8) + 1 Integer ALU (ALU1) with latency 1

~~RAW F1 I1-I6~~

~~RAW F5 I5-I10~~

~~RAW F2 I2-I5~~

~~RAW F6 I6-I9~~

~~RAW F4 I4-I6~~

~~RAW R5 I7-I9~~

~~RAW F4 I4-I5~~

~~RAW R6 I8-I10~~

	Instruction	ISSUE	START EXE	WB	Hazards Type	RSi	Unit
	I1: LD \$F1, 0(\$R1)	1	2	5		RS1	LDU1
	I2: FADD \$F2, \$F2, \$F3	2	3	6		RS4	FPU1
	I3: ADDI \$R3, \$R3, 8	3	4	7	Struct CDB	RS7	ALU1
	I4: LD \$F4, 0(R2)	4	5	8		RS2	LDU2
	I5: FADD \$F5, \$F4, \$F2	5			RAW \$F4, RAW \$F2	RS5	
	I6: FMULT \$F6, \$F1, \$F4	6			RAW \$F4	RS6	
	I7: ADDI \$R5, \$R5, 1	7	8			RS8	ALU1
	I8: LD \$R6, 0(\$R4)	8				RS1	
	I9: SD \$F6, 0(\$R5)						
	I10: SD \$F5, 0(\$R6)						

Exe.3 Tomasulo CC9

- 3 RESERVATION STATIONS (RS1, RS2, RS3) + 3 LOAD/STORE unit (LDU1, LDU2, LDU3) with latency 3
- 3 RESERVATION STATIONS (RS4, RS5, RS6) + 3 FPUs (FPU1, FPU2, FPU3) with latency 3
- 2 RESERVATION STATIONS (RS7, RS8) + 1 Integer ALU (ALU1) with latency 1

~~RAW F1 I1-I6~~

~~RAW F5 I5-I10~~

~~RAW F2 I2-I5~~

~~RAW F6 I6-I9~~

~~RAW F4 I4-I6~~

~~RAW R5 I7-I9~~

~~RAW F4 I4-I5~~

~~RAW R6 I8-I10~~

	Instruction	ISSUE	START EXE	WB	Hazards Type	RSi	Unit
	I1: LD \$F1, 0(\$R1)	1	2	5		RS1	LDU1
	I2: FADD \$F2, \$F2, \$F3	2	3	6		RS4	FPU1
	I3: ADDI \$R3, \$R3, 8	3	4	7	Struct CDB	RS7	ALU1
	I4: LD \$F4, 0(R2)	4	5	8		RS2	LDU2
	I5: FADD \$F5, \$F4, \$F2	5	9		RAW \$F4, RAW \$F2	RS5	FPU1
	I6: FMULT \$F6, \$F1, \$F4	6	9		RAW \$F4	RS6	FPU2
	I7: ADDI \$R5, \$R5, 1	7	8	9		RS8	ALU1
	I8: LD \$R6, 0(\$R4)	8	9			RS1	LDU1
	I9: SD \$F6, 0(\$R5)	9				RS2	
	I10: SD \$F5, 0(\$R6)						

Exe.3 Tomasulo CC10

- 3 RESERVATION STATIONS (RS1, RS2, RS3) + 3 LOAD/STORE unit (LDU1, LDU2, LDU3) with latency 3
- 3 RESERVATION STATIONS (RS4, RS5, RS6) + 3 FPUs (FPU1, FPU2, FPU3) with latency 3
- 2 RESERVATION STATIONS (RS7, RS8) + 1 Integer ALU (ALU1) with latency 1

~~RAW F1 I1-I6~~

~~RAW F5 I5-I10~~

~~RAW F2 I2-I5~~

~~RAW F6 I6-I9~~

~~RAW F4 I4-I6~~

~~RAW R5 I7-I9~~

~~RAW F4 I4-I5~~

~~RAW R6 I8-I10~~

	Instruction	ISSUE	START EXE	WB	Hazards Type	RSi	Unit
	I1: LD \$F1, 0(\$R1)	1	2	5		RS1	LDU1
	I2: FADD \$F2, \$F2, \$F3	2	3	6		RS4	FPU1
	I3: ADDI \$R3, \$R3, 8	3	4	7	Struct CDB	RS7	ALU1
	I4: LD \$F4, 0(R2)	4	5	8		RS2	LDU2
	I5: FADD \$F5, \$F4, \$F2	5	9		RAW \$F4, RAW \$F2	RS5	FPU1
	I6: FMULT \$F6, \$F1, \$F4	6	9		RAW \$F4	RS6	FPU2
	I7: ADDI \$R5, \$R5, 1	7	8	9		RS8	ALU1
	I8: LD \$R6, 0(\$R4)	8	9			RS1	LDU1
	I9: SD \$F6, 0(\$R5)	9			RAW \$F6	RS2	
	I10: SD \$F5, 0(\$R6)	10				RS3	

Exe.3 Tomasulo CC11

- 3 RESERVATION STATIONS (RS1, RS2, RS3) + 3 LOAD/STORE unit (LDU1, LDU2, LDU3) with latency 3
- 3 RESERVATION STATIONS (RS4, RS5, RS6) + 3 FPUs (FPU1, FPU2, FPU3) with latency 3
- 2 RESERVATION STATIONS (RS7, RS8) + 1 Integer ALU (ALU1) with latency 1

~~RAW F1 I1-I6~~

~~RAW F5 I5-I10~~

~~RAW F2 I2-I5~~

~~RAW F6 I6-I9~~

~~RAW F4 I4-I6~~

~~RAW R5 I7-I9~~

~~RAW F4 I4-I5~~

~~RAW R6 I8-I10~~

	Instruction	ISSUE	START EXE	WB	Hazards Type	RSi	Unit
	I1: LD \$F1, 0(\$R1)	1	2	5		RS1	LDU1
	I2: FADD \$F2, \$F2, \$F3	2	3	6		RS4	FPU1
	I3: ADDI \$R3, \$R3, 8	3	4	7	Struct CDB	RS7	ALU1
	I4: LD \$F4, 0(R2)	4	5	8		RS2	LDU2
	I5: FADD \$F5, \$F4, \$F2	5	9		RAW \$F4, RAW \$F2	RS5	FPU1
	I6: FMULT \$F6, \$F1, \$F4	6	9		RAW \$F4	RS6	FPU2
	I7: ADDI \$R5, \$R5, 1	7	8	9		RS8	ALU1
	I8: LD \$R6, 0(\$R4)	8	9			RS1	LDU1
	I9: SD \$F6, 0(\$R5)	9			RAW \$F6	RS2	
	I10: SD \$F5, 0(\$R6)	10			RAW \$F5, RAW \$R6	RS3	

Exe.3 Tomasulo CC12

- 3 RESERVATION STATIONS (RS1, RS2, RS3) + 3 LOAD/STORE unit (LDU1, LDU2, LDU3) with latency 3
- 3 RESERVATION STATIONS (RS4, RS5, RS6) + 3 FPUs (FPU1, FPU2, FPU3) with latency 3
- 2 RESERVATION STATIONS (RS7, RS8) + 1 Integer ALU (ALU1) with latency 1

~~RAW F1 I1-I6~~

~~RAW F5 I5-I10~~

~~RAW F2 I2-I5~~

~~RAW F6 I6-I9~~

~~RAW F4 I4-I6~~

~~RAW R5 I7-I9~~

~~RAW F4 I4-I5~~

~~RAW R6 I8-I10~~

	Instruction	ISSUE	START EXE	WB	Hazards Type	RSi	Unit
	I1: LD \$F1, 0(\$R1)	1	2	5		RS1	LDU1
	I2: FADD \$F2, \$F2, \$F3	2	3	6		RS4	FPU1
	I3: ADDI \$R3, \$R3, 8	3	4	7	Struct CDB	RS7	ALU1
	I4: LD \$F4, 0(R2)	4	5	8		RS2	LDU2
	I5: FADD \$F5, \$F4, \$F2	5	9	12	RAW \$F4, RAW \$F2	RS5	FPU1
	I6: FMULT \$F6, \$F1, \$F4	6	9	12	RAW \$F4, Struct CDB	RS6	FPU2
	I7: ADDI \$R5, \$R5, 1	7	8	9		RS8	ALU1
	I8: LD \$R6, 0(\$R4)	8	9		Struct CDB	RS1	LDU1
	I9: SD \$F6, 0(\$R5)	9			RAW \$F6	RS2	
	I10: SD \$F5, 0(\$R6)	10			RAW \$F5, RAW \$R6	RS3	

Exe.3 Tomasulo CC13

- 3 RESERVATION STATIONS (RS1, RS2, RS3) + 3 LOAD/STORE unit (LDU1, LDU2, LDU3) with latency 3
- 3 RESERVATION STATIONS (RS4, RS5, RS6) + 3 FPUs (FPU1, FPU2, FPU3) with latency 3
- 2 RESERVATION STATIONS (RS7, RS8) + 1 Integer ALU (ALU1) with latency 1

~~RAW F1 I1-I6~~

~~RAW F5 I5 - I10~~

~~RAW F2 I2-I5~~

~~RAW F6 I6-I9~~

~~RAW F4 I4-I6~~

~~RAW R5 I7 - I9~~

~~RAW F4 I4-I5~~

~~RAW R6 I8 - I10~~

Instruction	ISSUE	START EXE	WB	Hazards Type	RSi	Unit
I1: LD \$F1, 0(\$R1)	1	2	5		RS1	LDU1
I2: FADD \$F2, \$F2, \$F3	2	3	6		RS4	FPU1
I3: ADDI \$R3, \$R3, 8	3	4	7	Struct CDB	RS7	ALU1
I4: LD \$F4, 0(R2)	4	5	8		RS2	LDU2
I5: FADD \$F5, \$F4, \$F2	5	9	12	RAW \$F4, RAW \$F2	RS5	FPU1
I6: FMULT \$F6, \$F1, \$F4	6	9	13	RAW \$F4, Struct CDB	RS6	FPU2
I7: ADDI \$R5, \$R5, 1	7	8	9		RS8	ALU1
I8: LD \$R6, 0(\$R4)	8	9		Struct CDB	RS1	LDU1
I9: SD \$F6, 0(\$R5)	9			RAW \$F6	RS2	
I10: SD \$F5, 0(\$R6)	10			RAW \$F5, RAW \$R6	RS3	

Exe.3 Tomasulo CC14

- 3 RESERVATION STATIONS (RS1, RS2, RS3) + 3 LOAD/STORE unit (LDU1, LDU2, LDU3) with latency 3
- 3 RESERVATION STATIONS (RS4, RS5, RS6) + 3 FPUs (FPU1, FPU2, FPU3) with latency 3
- 2 RESERVATION STATIONS (RS7, RS8) + 1 Integer ALU (ALU1) with latency 1

~~RAW F1 I1-I6~~

~~RAW F5 I5 - I10~~

~~RAW F2 I2-I5~~

~~RAW F6 I6-I9~~

~~RAW F4 I4-I6~~

~~RAW R5 I7 - I9~~

~~RAW F4 I4-I5~~

~~RAW R6 I8 - I10~~

Instruction	ISSUE	START EXE	WB	Hazards Type	RSi	Unit
I1: LD \$F1, 0(\$R1)	1	2	5		RS1	LDU1
I2: FADD \$F2, \$F2, \$F3	2	3	6		RS4	FPU1
I3: ADDI \$R3, \$R3, 8	3	4	7	Struct CDB	RS7	ALU1
I4: LD \$F4, 0(R2)	4	5	8		RS2	LDU2
I5: FADD \$F5, \$F4, \$F2	5	9	12	RAW \$F4, RAW \$F2	RS5	FPU1
I6: FMULT \$F6, \$F1, \$F4	6	9	13	RAW \$F4, Struct CDB	RS6	FPU2
I7: ADDI \$R5, \$R5, 1	7	8	9		RS8	ALU1
I8: LD \$R6, 0(\$R4)	8	9	14	Struct CDB	RS1	LDU1
I9: SD \$F6, 0(\$R5)	9	14		RAW \$F6	RS2	LDU2
I10: SD \$F5, 0(\$R6)	10			RAW \$F5, RAW \$R6	RS3	

Exe.3 Tomasulo CC15

- 3 RESERVATION STATIONS (RS1, RS2, RS3) + 3 LOAD/STORE unit (LDU1, LDU2, LDU3) with latency 3
- 3 RESERVATION STATIONS (RS4, RS5, RS6) + 3 FPUs (FPU1, FPU2, FPU3) with latency 3
- 2 RESERVATION STATIONS (RS7, RS8) + 1 Integer ALU (ALU1) with latency 1

~~RAW F1 I1-I6~~

~~RAW F5 I5 - I10~~

~~RAW F2 I2-I5~~

~~RAW F6 I6 - I9~~

~~RAW F4 I4-I6~~

~~RAW R5 I7 - I9~~

~~RAW F4 I4-I5~~

~~RAW R6 I8 - I10~~

Instruction	ISSUE	START EXE	WB	Hazards Type	RSi	Unit
I1: LD \$F1, 0(\$R1)	1	2	5		RS1	LDU1
I2: FADD \$F2, \$F2, \$F3	2	3	6		RS4	FPU1
I3: ADDI \$R3, \$R3, 8	3	4	7	Struct CDB	RS7	ALU1
I4: LD \$F4, 0(R2)	4	5	8		RS2	LDU2
I5: FADD \$F5, \$F4, \$F2	5	9	12	RAW \$F4, RAW \$F2	RS5	FPU1
I6: FMULT \$F6, \$F1, \$F4	6	9	13	RAW \$F4, Struct CDB	RS6	FPU2
I7: ADDI \$R5, \$R5, 1	7	8	9		RS8	ALU1
I8: LD \$R6, 0(\$R4)	8	9	14	Struct CDB	RS1	LDU1
I9: SD \$F6, 0(\$R5)	9	14		RAW \$F6	RS2	LDU2
I10: SD \$F5, 0(\$R6)	10	15		RAW \$F5, RAW \$R6	RS3	LDU1

Exe.3 Tomasulo CC17

- 3 RESERVATION STATIONS (RS1, RS2, RS3) + 3 LOAD/STORE unit (LDU1, LDU2, LDU3) with latency 3
- 3 RESERVATION STATIONS (RS4, RS5, RS6) + 3 FPUs (FPU1, FPU2, FPU3) with latency 3
- 2 RESERVATION STATIONS (RS7, RS8) + 1 Integer ALU (ALU1) with latency 1

~~RAW F1 I1-I6~~

~~RAW F5 I5 - I10~~

~~RAW F2 I2-I5~~

~~RAW F6 I6 - I9~~

~~RAW F4 I4-I6~~

~~RAW R5 I7 - I9~~

~~RAW F4 I4-I5~~

~~RAW R6 I8 - I10~~

Instruction	ISSUE	START EXE	WB	Hazards Type	RSi	Unit
I1: LD \$F1, 0(\$R1)	1	2	5		RS1	LDU1
I2: FADD \$F2, \$F2, \$F3	2	3	6		RS4	FPU1
I3: ADDI \$R3, \$R3, 8	3	4	7	Struct CDB	RS7	ALU1
I4: LD \$F4, 0(R2)	4	5	8		RS2	LDU2
I5: FADD \$F5, \$F4, \$F2	5	9	12	RAW \$F4, RAW \$F2	RS5	FPU1
I6: FMULT \$F6, \$F1, \$F4	6	9	13	RAW \$F4, Struct CDB	RS6	FPU2
I7: ADDI \$R5, \$R5, 1	7	8	9		RS8	ALU1
I8: LD \$R6, 0(\$R4)	8	9	14	Struct CDB	RS1	LDU1
I9: SD \$F6, 0(\$R5)	9	14	17	RAW \$F6	RS2	LDU2
I10: SD \$F5, 0(\$R6)	10	15		RAW \$F5, RAW \$R6	RS3	LDU1

Exe.3 Tomasulo CC18

- 3 RESERVATION STATIONS (RS1, RS2, RS3) + 3 LOAD/STORE unit (LDU1, LDU2, LDU3) with latency 3
- 3 RESERVATION STATIONS (RS4, RS5, RS6) + 3 FPUs (FPU1, FPU2, FPU3) with latency 3
- 2 RESERVATION STATIONS (RS7, RS8) + 1 Integer ALU (ALU1) with latency 1

~~RAW F1 I1-I6~~

~~RAW F5 I5 - I10~~

~~RAW F2 I2-I5~~

~~RAW F6 I6 - I9~~

~~RAW F4 I4-I6~~

~~RAW R5 I7 - I9~~

~~RAW F4 I4-I5~~

~~RAW R6 I8 - I10~~

Instruction	ISSUE	START EXE	WB	Hazards Type	RSi	Unit
I1: LD \$F1, 0(\$R1)	1	2	5		RS1	LDU1
I2: FADD \$F2, \$F2, \$F3	2	3	6		RS4	FPU1
I3: ADDI \$R3, \$R3, 8	3	4	7	Struct CDB	RS7	ALU1
I4: LD \$F4, 0(R2)	4	5	8		RS2	LDU2
I5: FADD \$F5, \$F4, \$F2	5	9	12	RAW \$F4, RAW \$F2	RS5	FPU1
I6: FMULT \$F6, \$F1, \$F4	6	9	13	RAW \$F4, Struct CDB	RS6	FPU2
I7: ADDI \$R5, \$R5, 1	7	8	9		RS8	ALU1
I8: LD \$R6, 0(\$R4)	8	9	14	Struct CDB	RS1	LDU1
I9: SD \$F6, 0(\$R5)	9	14	17	RAW \$F6	RS2	LDU2
I10: SD \$F5, 0(\$R6)	10	15	18	RAW \$F5, RAW \$R6	RS3	LDU1

Scoreboard

vs

Tomasulo

	Instruction	ISSUE	READ OPERAND	EXE COMPLETE	WB
I1	LD \$F1, 0(\$R1)	1	2	5	6
I2	FADD \$F2, \$F2, \$F3	2	3	7	8
I3	ADDI \$R3, \$R3, 8	3	4	5	7
I4	LD \$F4, 0(R2)	4	5	8	9
I5	FADD \$F5, \$F4, \$F2	5	10	14	15
I6	FMULT \$F6, \$F1, \$F4	6	10	14	16
I7	ADDI \$R5, \$R5, 1	7	8	9	10
I8	LD \$R6, 0(\$R4)	8	9	12	13
I9	SD \$F6, 0(\$R5)	9	17	20	21
I10	SD \$F5, 0(\$R6)	10	16	19	20

ISSUE	START EXE	WB
1	2	5
2	3	6
3	4	7
4	5	8
5	9	12
6	9	13
7	8	9
8	9	14
9	14	17
10	15	18





Thanks for your attention

Davide Conficconi <davide.conficconi@polimi.it>

Acknowledgements

E. Del Sozzo, Marco D. Santambrogio, D. Sciuto

Part of this material comes from:

- “Computer Organization and Design” and “Computer Architecture A Quantitative Approach” Patterson and Hennessy books
- “Digital Design and Computer Architecture” Harris and Harris
- Elsevier Inc. online materials
- Papers/news cited in this lecture

and are properties of their respective owners