Performance: Advanced Pipelining Concepts

Jan Reineke

Universität des Saarlandes

How to increase performance?

Processor time = Number of executed instructions
* Cycles per instruction
* Cycle time

1. Reduce number of executed instructions

   Compiler optimizations

2. Reduce cycles per instruction (CPI)

   Microarchitecture (e.g. pipelining, memory hierarchy)

3. Increase clock frequency = reduce cycle time

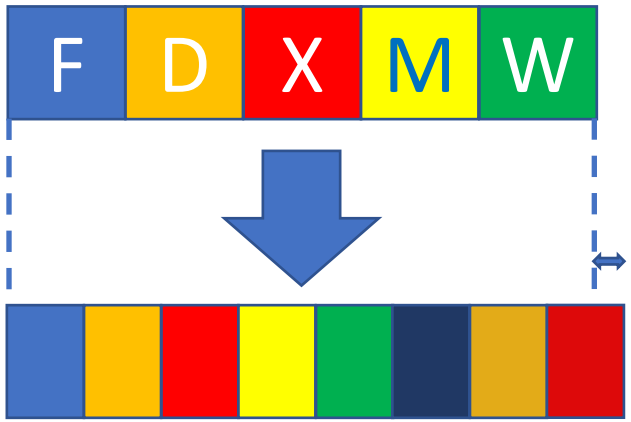
   Technological improvements, microarchitecture (e.g. pipelining)

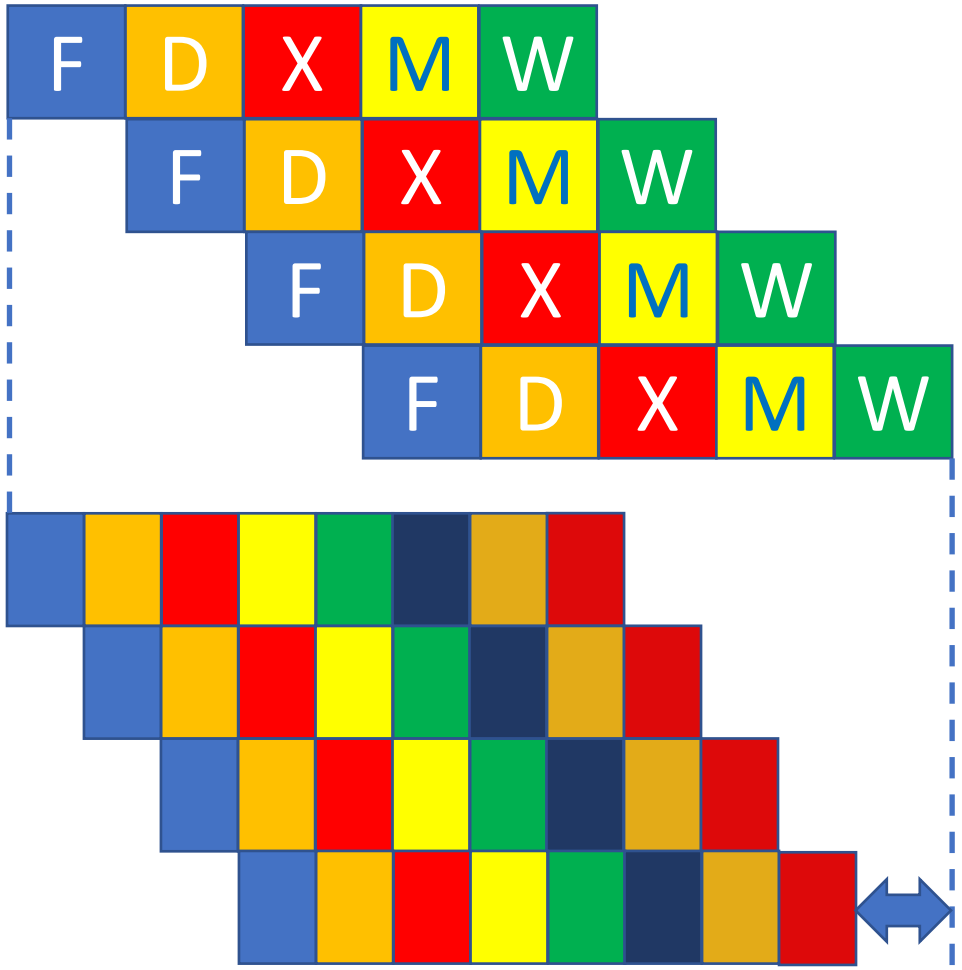
Instruction set influences all 3 factors (but 1. in particular)

How to reduce the cycle time?

Do **less** work in each cycle! **Compensate** with more pipeline stages.

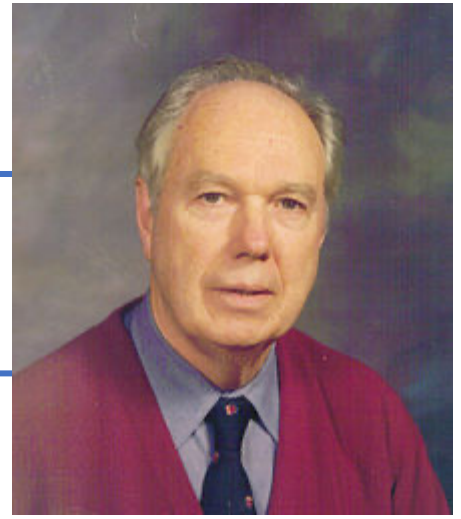
Example:



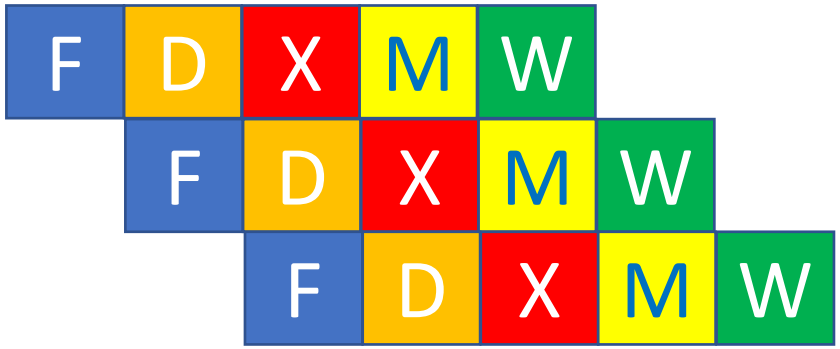


How to reduce the CPI?

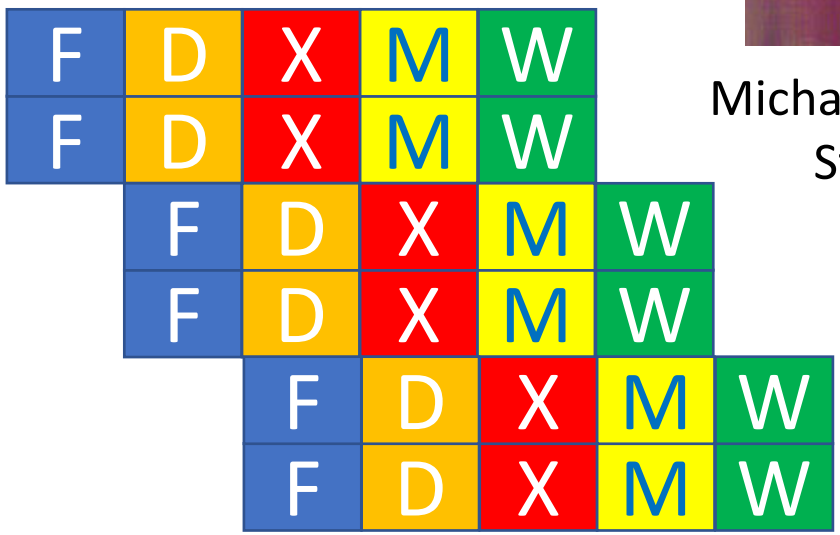
Flynn bottleneck:
"You cannot retire more instructions than you fetch."



Michael J. Flynn (1934-)
Stanford Univ.



Best case CPI = 1



Best case CPI = 1/width

→ Process multiple instructions in each cycle, "**superscalar pipelines**"

[Image source: http://arith.stanford.edu/~flynn/jpeg/flynn600.JPG]

Challenges

Deeper pipelines and superscalar pipelines increase **potential parallelism**

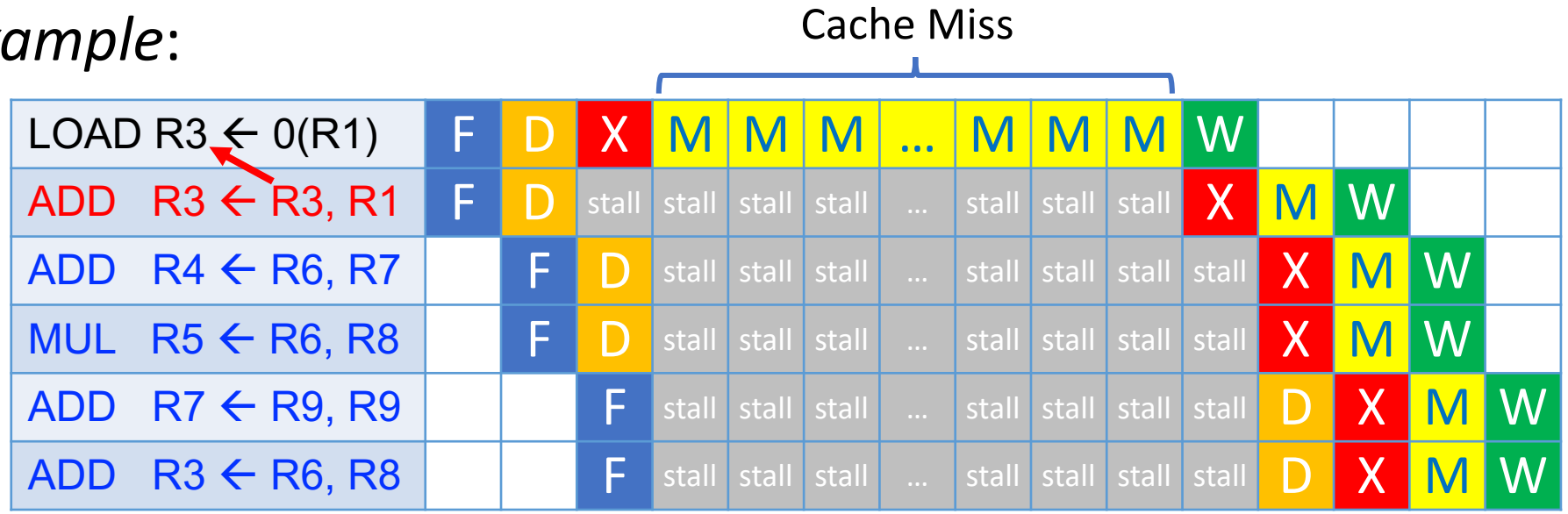
But can we actually fill these pipelines?

Challenges:

- Long-latency operations, such as cache misses
- Data dependencies
- Control dependencies

But: Long-latency operations + Data dependencies

Example:



First ADD stalls the whole pipeline!

- ADD cannot execute because its source registers are unavailable
- Later **independent** instructions also cannot get executed

Solution: **Out-of-order execution**

In-order vs. Out-of-order execution

In-order:

LOAD R3 ← 0(R1)	F	D	X	M	M	M	…	M	M	M	W				
ADD R3 ← R3, R1	F	D	stall	stall	stall	stall	…	stall	stall	stall	X	M	W		
ADD R4 ← R6, R7		F	D	stall	stall	stall	…	stall	stall	stall	stall	X	M	W	
MUL R5 ← R6, R8		F	D	stall	stall	stall	…	stall	stall	stall	stall	X	M	W	
ADD R7 ← R9, R9			F	stall	stall	stall	…	stall	stall	stall	stall	D	X	M	W
ADD R3 ← R6, R8			F	stall	stall	stall	…	stall	stall	stall	stall	D	X	M	W

Out-of-order:

LOAD R3 ← 0(R1)	F	D	X	M	M	M	…	M	M	M	W				
ADD R3 ← R3, R1	F	D	stall	stall	stall	stall	…	stall	stall	stall	X	M	W		
ADD R4 ← R6, R7		F	D	X	M	W									
MUL R5 ← R6, R8		F	D	X	M	W									
ADD R7 ← R9, R9			F	D	X	M	W								
ADD R3 ← R6, R8			F	D	X	M	W								

… …

Out-of-order execution: Implementation in a nutshell

- Move the dependent instructions out of the way of independent ones (s.t. independent ones can execute)

  Key hardware data structure:
  Reservation stations = Rest areas for dependent instructions

- Monitor the source values of each instruction in the reservation stations
- When all source values of an instruction are available, execute the instruction

Out-of-order execution: Tomasulo Algorithm

Robert Tomasulo (1934-2008) Eckert-Mauchly Award, 1997

Program we will simulate:

MUL	R3	← R1, R2
ADD	R5	← R3, R4
ADD	R7	← R2, R6
ADD	R10	← R8, R9
MUL	R11	← R7, R10
ADD	R5	← R7, R4

Initially:

1. Reservation stations are all empty
2. All registers are valid

Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	1		3
R4	1		4
R5	1		5
R6	1		6
R7	1		7
R8	1		8
R9	1		9
R10	1		10
R11	1		11

Register Alias Table

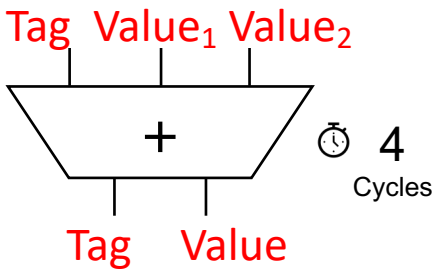
Reservation station for ADD Unit

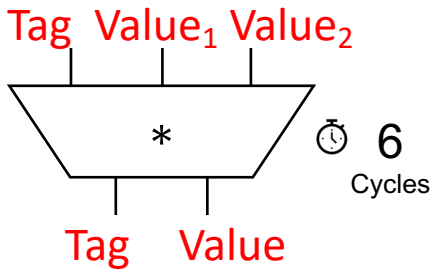
	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a						
b						
c						
d						

Reservation station for MUL Unit

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x						
y						
z						
t						

Pipelined Functional Units





[Based on: https://safari.ethz.ch/digitaltechnik/spring2021, Lecture 16]

Out-of-order execution: Example

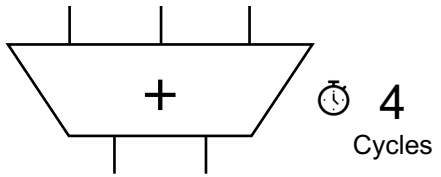
	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F														
ADD	R5 ← R3, R4															
ADD	R7 ← R2, R6															
ADD	R10 ← R8, R9															
MUL	R11 ← R7, R10															
ADD	R5 ← R7, R4															

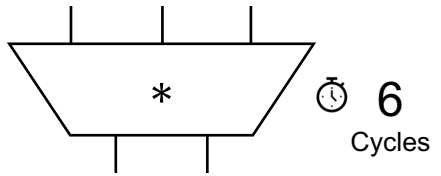
Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	1		3
R4	1		4
R5	1		5
R6	1		6
R7	1		7
R8	1		8
R9	1		9
R10	1		10
R11	1		11

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a						
b						
c						
d						

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x						
y						
z						
t						





Out-of-order execution: Example

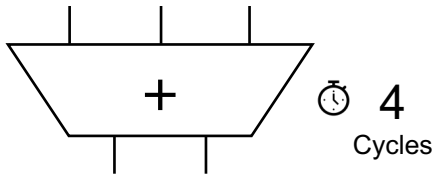
	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	**R3 ← R1, R2**	F	**D**													
ADD	R5 ← R3, R4		F													
ADD	R7 ← R2, R6															
ADD	R10 ← R8, R9															
MUL	R11 ← R7, R10															
ADD	R5 ← R7, R4															

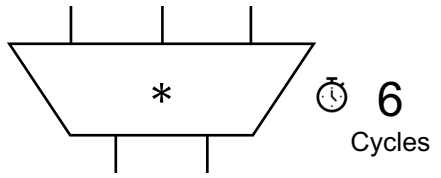
Register	Valid	Tag	Value
R1	**1**		**1**
R2	**1**		**2**
R3	1		3
R4	1		4
R5	1		5
R6	1		6
R7	1		7
R8	1		8
R9	1		9
R10	1		10
R11	1		11

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a						
b						
c						
d						

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x	1	~	1	1	~	2
y						
z						
t						





Out-of-order execution: Example

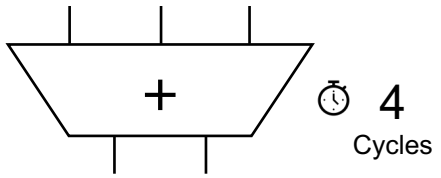
	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	**R3 ← R1, R2**	F	**D**													
ADD	R5 ← R3, R4		F													
ADD	R7 ← R2, R6															
ADD	R10 ← R8, R9															
MUL	R11 ← R7, R10															
ADD	R5 ← R7, R4															

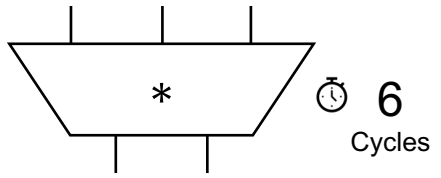
Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	**0**	**x**	
R4	1		4
R5	1		5
R6	1		6
R7	1		7
R8	1		8
R9	1		9
R10	1		10
R11	1		11

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a						
b						
c						
d						

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x	1	~	1	1	~	2
y						
z						
t						



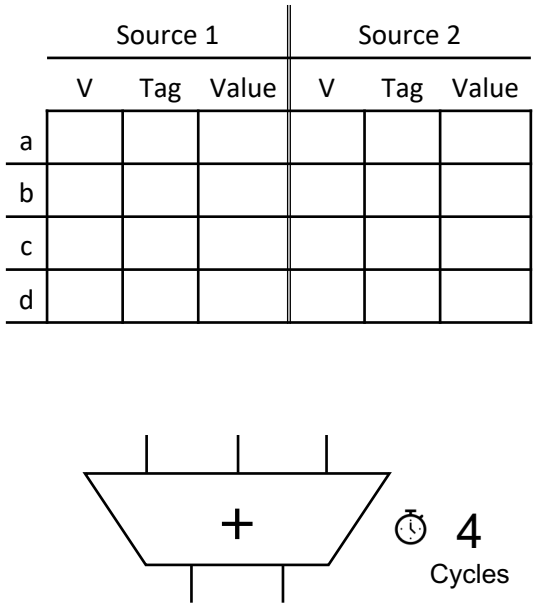


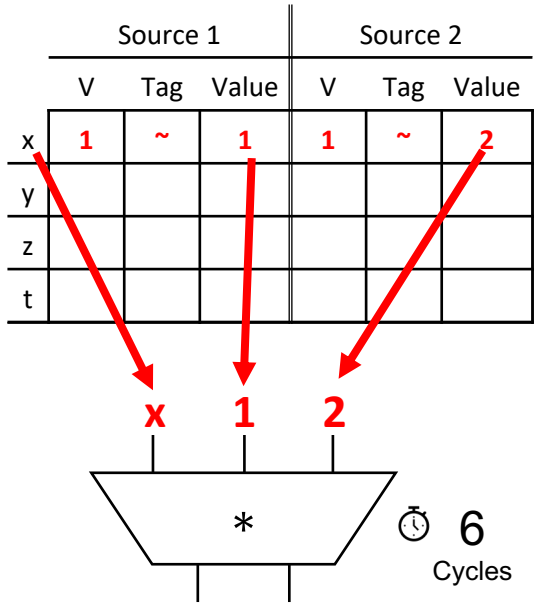
Out-of-order execution: Example

	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	**R3 ← R1, R2**	F	D	X_1												
ADD	R5 ← R3, R4		F													
ADD	R7 ← R2, R6															
ADD	R10 ← R8, R9															
MUL	R11 ← R7, R10															
ADD	R5 ← R7, R4															

Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	0	x	
R4	1		4
R5	1		5
R6	1		6
R7	1		7
R8	1		8
R9	1		9
R10	1		10
R11	1		11

Register Alias Table





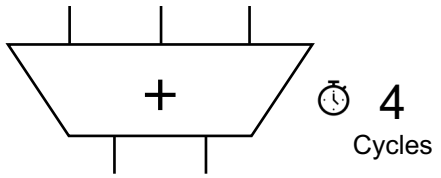
Out-of-order execution: Example

	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1												
ADD	**R5 ← R3, R4**		F	**D**												
ADD	R7 ← R2, R6			F												
ADD	R10 ← R8, R9															
MUL	R11 ← R7, R10															
ADD	R5 ← R7, R4															

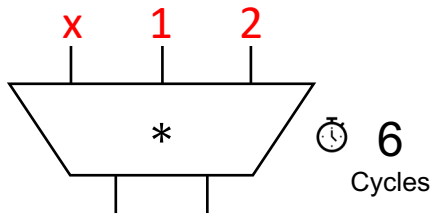
Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	**0**	**x**	
R4	**1**		**4**
R5	1		5
R6	1		6
R7	1		7
R8	1		8
R9	1		9
R10	1		10
R11	1		11

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a	**0**	**x**		**1**	**~**	**4**
b						
c						
d						



	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x	1	~	1	1	~	2
y						
z						
t						



Out-of-order execution: Example

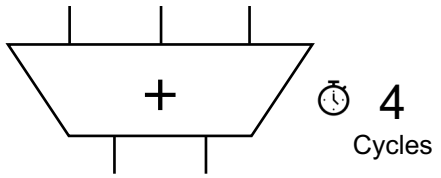
	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1												
ADD	**R5 ← R3, R4**		F	**D**												
ADD	R7 ← R2, R6			F												
ADD	R10 ← R8, R9															
MUL	R11 ← R7, R10															
ADD	R5 ← R7, R4															

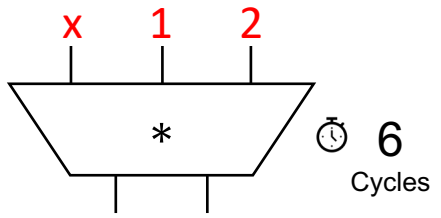
Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	0	x	
R4	1		4
R5	**0**	**a**	
R6	1		6
R7	1		7
R8	1		8
R9	1		9
R10	1		10
R11	1		11

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a	0	x		1	~	4
b						
c						
d						

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x	1	~	1	1	~	2
y						
z						
t						





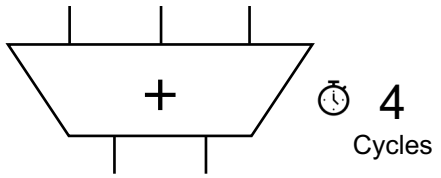
Out-of-order execution: Example

	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1	X_2											
ADD	R5 ← R3, R4		F	D	-											
ADD	**R7 ← R2, R6**			F	**D**											
ADD	R10 ← R8, R9				F											
MUL	R11 ← R7, R10															
ADD	R5 ← R7, R4															

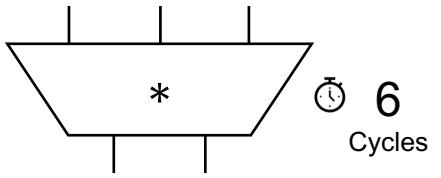
Register	Valid	Tag	Value
R1	1		1
R2	**1**		**2**
R3	0	x	
R4	1		4
R5	0	a	
R6	**1**		**6**
R7	1		7
R8	1		8
R9	1		9
R10	1		10
R11	1		11

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a	0	x		1	~	4
b	**1**	**~**	**2**	**1**	**~**	**6**
c						
d						



	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x	1	~	1	1	~	2
y						
z						
t						



Out-of-order execution: Example

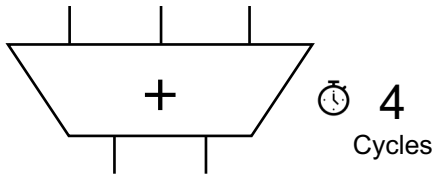
	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1	X_2											
ADD	R5 ← R3, R4		F	D	-											
ADD	**R7 ← R2, R6**			F	**D**											
ADD	R10 ← R8, R9				F											
MUL	R11 ← R7, R10															
ADD	R5 ← R7, R4															

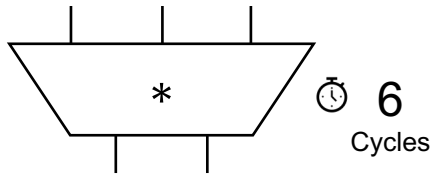
Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	0	x	
R4	1		4
R5	0	a	
R6	1		6
R7	**0**	**b**	
R8	1		8
R9	1		9
R10	1		10
R11	1		11

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a	0	x		1	~	4
b	1	~	2	1	~	6
c						
d						

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x	1	~	1	1	~	2
y						
z						
t						





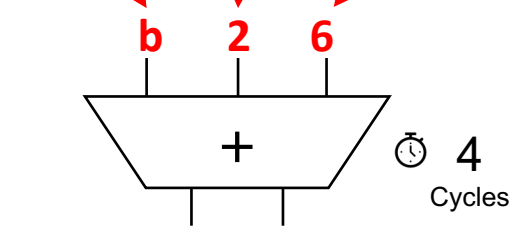
Out-of-order execution: Example

	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1	X_2	X_3										
ADD	R5 ← R3, R4		F	D	-	-										
ADD	**R7 ← R2, R6**			F	D	X_1										
ADD	R10 ← R8, R9				F											
MUL	R11 ← R7, R10															
ADD	R5 ← R7, R4															

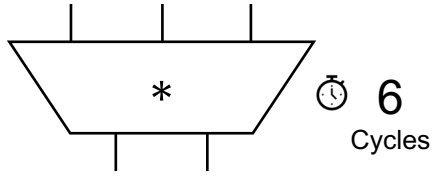
Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	0	x	
R4	1		4
R5	0	a	
R6	1		6
R7	0	b	
R8	1		8
R9	1		9
R10	1		10
R11	1		11

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a	0	x		1	~	4
b	1	~	2	1	~	6
c						
d						



	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x	1	~	1	1	~	2
y						
z						
t						



Out-of-order execution: Example

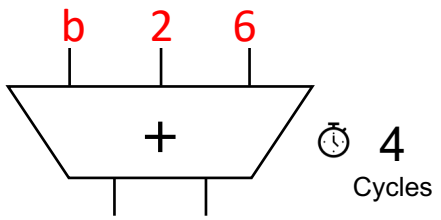
	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1	X_2	X_3										
ADD	R5 ← R3, R4		F	D	-	-										
ADD	R7 ← R2, R6			F	D	X_1										
ADD	**R10 ← R8, R9**				F	**D**										
MUL	R11 ← R7, R10															
ADD	R5 ← R7, R4															

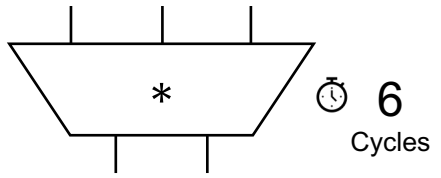
Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	0	x	
R4	1		4
R5	0	a	
R6	1		6
R7	0	b	
R8	**1**		**8**
R9	**1**		**9**
R10	1		10
R11	1		11

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a	0	x		1	~	4
b	1	~	2	1	~	6
c	**1**	**~**	**8**	**1**	**~**	**9**
d						

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x	1	~	1	1	~	2
y						
z						
t						





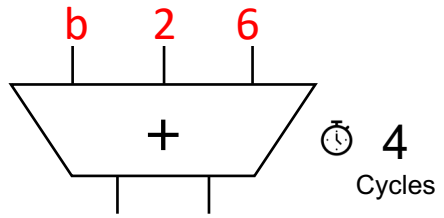
Out-of-order execution: Example

	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1	X_2	X_3										
ADD	R5 ← R3, R4		F	D	-	-										
ADD	R7 ← R2, R6			F	D	X_1										
ADD	**R10 ← R8, R9**				F	**D**										
MUL	R11 ← R7, R10					F										
ADD	R5 ← R7, R4															

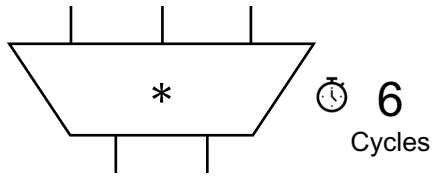
Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	0	x	
R4	1		4
R5	0	a	
R6	1		6
R7	0	b	
R8	1		8
R9	1		9
R10	**0**	**c**	
R11	1		11

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a	0	x		1	~	4
b	1	~	2	1	~	6
c	1	~	8	1	~	9
d						



	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x	1	~	1	1	~	2
y						
z						
t						



Out-of-order execution: Example

	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1	X_2	X_3	X_4									
ADD	R5 ← R3, R4		F	D	-	-	-									
ADD	R7 ← R2, R6			F	D	X_1	X_2									
ADD	**R10 ← R8, R9**				F	D	**X_1**									
MUL	R11 ← R7, R10					F										
ADD	R5 ← R7, R4															

Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	0	x	
R4	1		4
R5	0	a	
R6	1		6
R7	0	b	
R8	1		8
R9	1		9
R10	0	c	
R11	1		11

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a	0	x		1	~	4
b	1	~	2	1	~	6
c	**1**	**~**	**8**	**1**	**~**	**9**
d						

c **8** **9**

+ 4 Cycles

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x	1	~	1	1	~	2
y						
z						
t						

* 6 Cycles

Out-of-order execution: Example

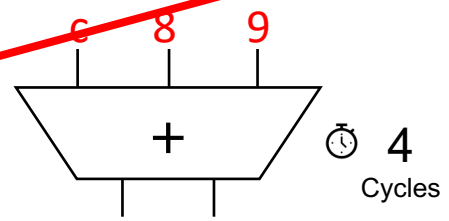
	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1	X_2	X_3	X_4									
ADD	R5 ← R3, R4		F	D	-	-	-									
ADD	R7 ← R2, R6			F	D	X_1	X_2									
ADD	R10 ← R8, R9				F	D	X_1									
MUL	**R11 ← R7, R10**					F	**D**									
ADD	R5 ← R7, R4						F									

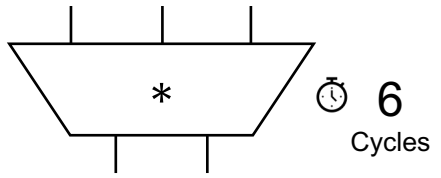
Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	0	x	
R4	1		4
R5	0	a	
R6	1		6
R7	**0**	**b**	
R8	1		8
R9	1		9
R10	**0**	**c**	
R11	1		11

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a	0	x		1	~	4
b	1	~	2	1	~	6
c	1	~	8	1	~	9
d						

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x	1	~	1	1	~	2
y	**0**	**b**		**0**	**c**	
z						
t						





Out-of-order execution: Example

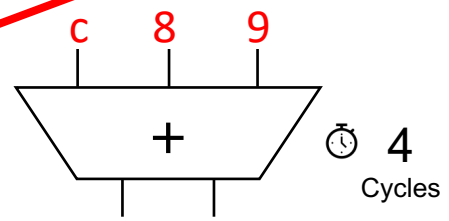
	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1	X_2	X_3	X_4									
ADD	R5 ← R3, R4		F	D	-	-	-									
ADD	R7 ← R2, R6			F	D	X_1	X_2									
ADD	R10 ← R8, R9				F	D	X_1									
MUL	**R11 ← R7, R10**					F	**D**									
ADD	R5 ← R7, R4						F									

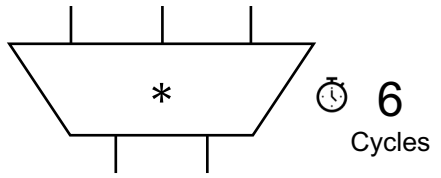
Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	0	x	
R4	1		4
R5	0	a	
R6	1		6
R7	0	b	
R8	1		8
R9	1		9
R10	0	c	
R11	**0**	**y**	

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a	0	x		1	~	4
b	1	~	2	1	~	6
c	1	~	8	1	~	9
d						

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x	1	~	1	1	~	2
y	0	b		0	c	
z						
t						





Out-of-order execution: Example

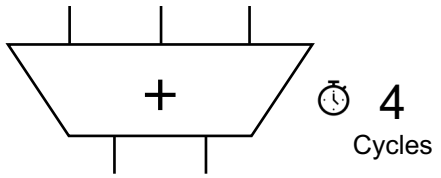
	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1	X_2	X_3	X_4	X_5								
ADD	R5 ← R3, R4		F	D	-	-	-	-								
ADD	R7 ← R2, R6			F	D	X_1	X_2	X_3								
ADD	R10 ← R8, R9				F	D	X_1	X_2								
MUL	R11 ← R7, R10					F	D	-								
ADD	**R5 ← R7, R4**						F	**D**								

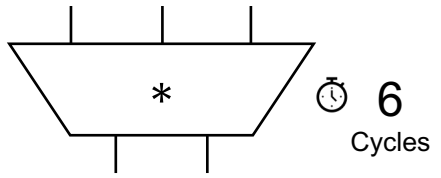
Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	0	x	
R4	**1**		**4**
R5	0	a	
R6	1		6
R7	**0**	**b**	
R8	1		8
R9	1		9
R10	0	c	
R11	0	y	

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a	0	x		1	~	4
b	1	~	2	1	~	6
c	1	~	8	1	~	9
d	**1**	**~**	**4**	**0**	**b**	

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x	1	~	1	1	~	2
y	0	b		0	c	
z						
t						





Out-of-order execution: Example

All instructions have been decoded!

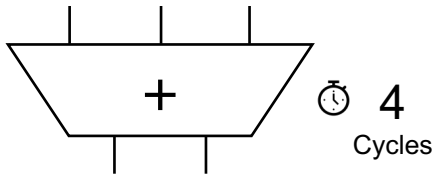
	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1	X_2	X_3	X_4	X_5								
ADD	R5 ← R3, R4		F	D	-	-	-	-								
ADD	R7 ← R2, R6			F	D	X_1	X_2	X_3								
ADD	R10 ← R8, R9				F	D	X_1	X_2								
MUL	R11 ← R7, R10					F	D	-								
ADD	**R5 ← R7, R4**						F	**D**								

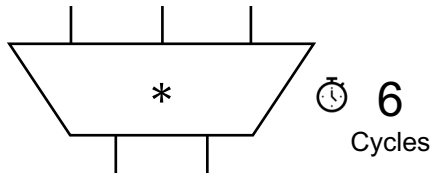
Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	0	x	
R4	1		4
R5	**0**	**d**	
R6	1		6
R7	0	b	
R8	1		8
R9	1		9
R10	0	c	
R11	0	y	

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a	0	x		1	~	4
b	1	~	2	1	~	6
c	1	~	8	1	~	9
d	1	~	4	0	b	

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x	1	~	1	1	~	2
y	0	b		0	c	
z						
t						





Out-of-order execution: Example

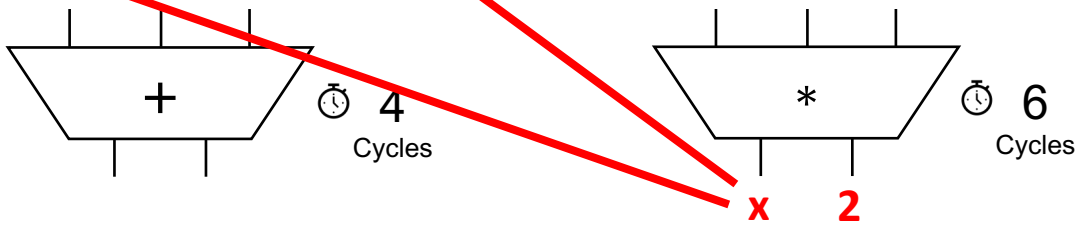
	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	**R3 ← R1, R2**	F	D	X_1	X_2	X_3	X_4	X_5	**X_6**							
ADD	R5 ← R3, R4		F	D	-	-	-	-	-							
ADD	R7 ← R2, R6			F	D	X_1	X_2	X_3								
ADD	R10 ← R8, R9				F	D	X_1	X_2								
MUL	R11 ← R7, R10					F	D	-								
ADD	R5 ← R7, R4						F	D								

Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	**0**	**x**	
R4	1		4
R5	0	d	
R6	1		6
R7	0	b	
R8	1		8
R9	1		9
R10	0	c	
R11	0	y	

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a	**0**	**x**		1	~	4
b	1	~	2	1	~	6
c	1	~	8	1	~	9
d	1	~	4	0	b	

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x	1	~	1	1	~	2
y	0	b		0	c	
z						
t						



Out-of-order execution: Example

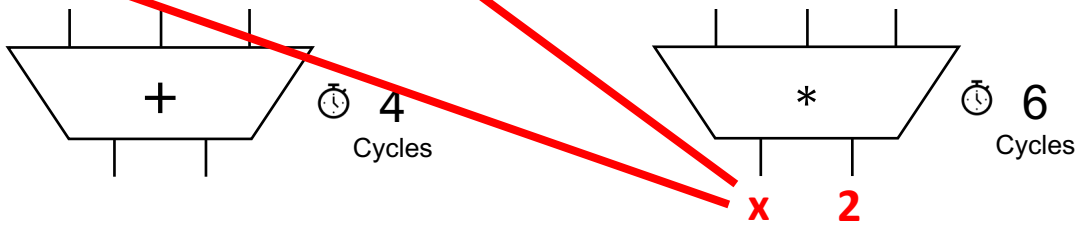
	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	**R3 ← R1, R2**	F	D	X_1	X_2	X_3	X_4	X_5	**X_6**							
ADD	R5 ← R3, R4		F	D	-	-	-	-	-							
ADD	R7 ← R2, R6			F	D	X_1	X_2	X_3								
ADD	R10 ← R8, R9				F	D	X_1	X_2								
MUL	R11 ← R7, R10					F	D	-								
ADD	R5 ← R7, R4						F	D								

Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	**1**		**2**
R4	1		4
R5	0	d	
R6	1		6
R7	0	b	
R8	1		8
R9	1		9
R10	0	c	
R11	0	y	

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a	**1**	**~**	**2**	1	~	4
b	1	~	2	1	~	6
c	1	~	8	1	~	9
d	1	~	4	0	b	

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x						
y	0	b		0	c	
z						
t						



Out-of-order execution: Example

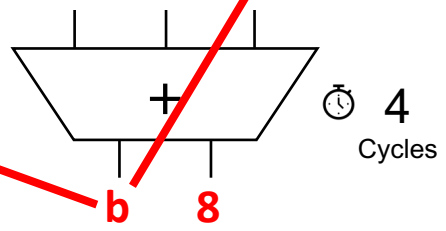
	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1	X_2	X_3	X_4	X_5	X_6							
ADD	R5 ← R3, R4		F	D	-	-	-	-	-							
ADD	**R7 ← R2, R6**			F	D	X_1	X_2	X_3	**X_4**							
ADD	R10 ← R8, R9				F	D	X_1	X_2								
MUL	R11 ← R7, R10					F	D	-								
ADD	R5 ← R7, R4						F	D								

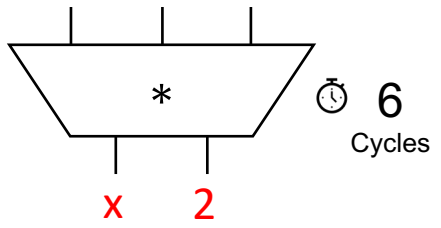
Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	1		2
R4	1		4
R5	0	d	
R6	1		6
R7	0	b	
R8	1		8
R9	1		9
R10	0	c	
R11	0	y	

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a	1	~	2	1	~	4
b	1	~	2	1	~	6
c	1	~	8	1	~	9
d	1	~	4	0	b	

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x						
y	0	b		0	c	
z						
t						





Out-of-order execution: Example

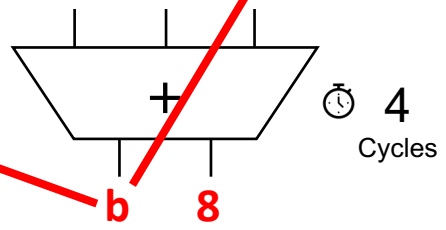
	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1	X_2	X_3	X_4	X_5	X_6							
ADD	R5 ← R3, R4		F	D	-	-	-	-	-							
ADD	**R7 ← R2, R6**			F	D	X_1	X_2	X_3	X_4							
ADD	R10 ← R8, R9				F	D	X_1	X_2								
MUL	R11 ← R7, R10					F	D	-								
ADD	R5 ← R7, R4						F	D								

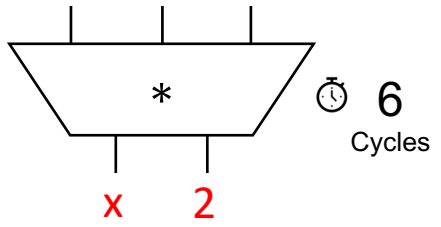
Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	1		2
R4	1		4
R5	0	d	
R6	1		6
R7	**1**		**8**
R8	1		8
R9	1		9
R10	0	c	
R11	0	y	

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a	1	~	2	1	~	4
b						
c	1	~	8	1	~	9
d	1	~	4	**1**	**~**	**8**

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x						
y	**1**	**~**	**8**	0	c	
z						
t						





Out-of-order execution: Example

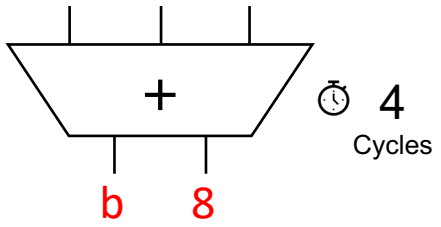
	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1	X_2	X_3	X_4	X_5	X_6							
ADD	R5 ← R3, R4		F	D	-	-	-	-	-							
ADD	R7 ← R2, R6			F	D	X_1	X_2	X_3	X_4							
ADD	R10 ← R8, R9				F	D	X_1	X_2	X_3							
MUL	R11 ← R7, R10					F	D	-	-							
ADD	R5 ← R7, R4						F	D	-							

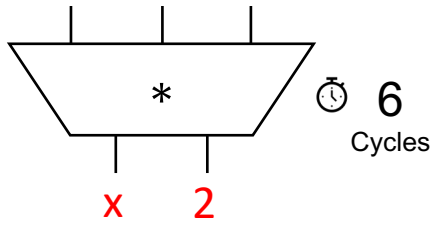
Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	1		2
R4	1		4
R5	0	d	
R6	1		6
R7	1		8
R8	1		8
R9	1		9
R10	0	c	
R11	0	y	

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a	1	~	2	1	~	4
b						
c	1	~	8	1	~	9
d	1	~	4	1	~	8

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x						
y	1	~	8	0	c	
z						
t						



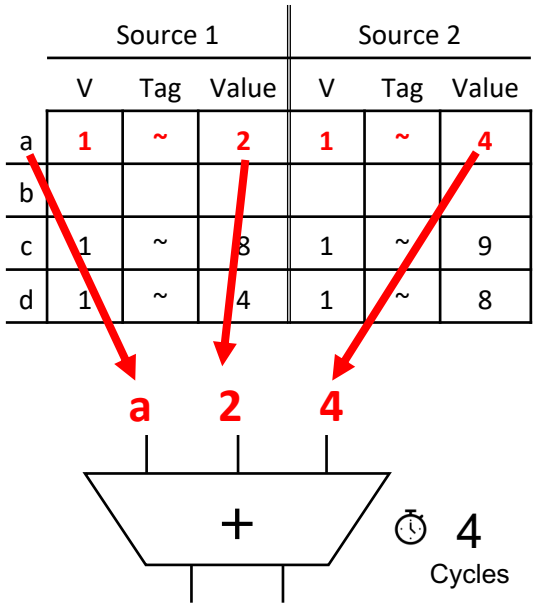


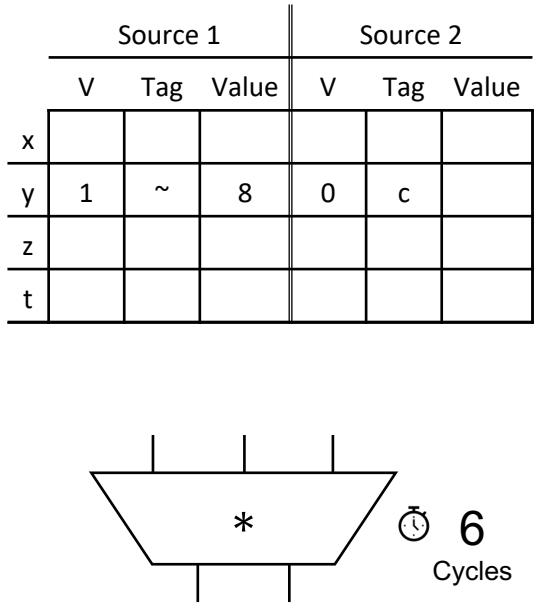
Out-of-order execution: Example

	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1	X_2	X_3	X_4	X_5	X_6							
ADD	**R5 ← R3, R4**		F	D	-	-	-	-	-	**X_1**						
ADD	R7 ← R2, R6			F	D	X_1	X_2	X_3	X_4							
ADD	R10 ← R8, R9				F	D	X_1	X_2	X_3							
MUL	R11 ← R7, R10					F	D	-	-							
ADD	R5 ← R7, R4						F	D	-							

Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	1		2
R4	1		4
R5	0	d	
R6	1		6
R7	1		8
R8	1		8
R9	1		9
R10	0	c	
R11	0	y	

Register Alias Table





Out-of-order execution: Example

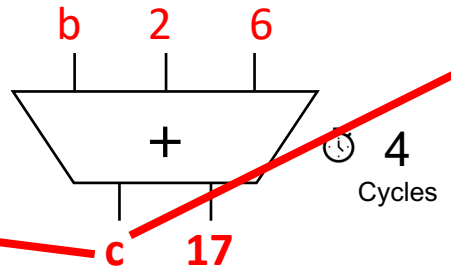
	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1	X_2	X_3	X_4	X_5	X_6							
ADD	R5 ← R3, R4		F	D	-	-	-	-	-	X_1						
ADD	R7 ← R2, R6			F	D	X_1	X_2	X_3	X_4							
ADD	**R10 ← R8, R9**				F	D	X_1	X_2	X_3	**X_4**						
MUL	R11 ← R7, R10					F	D	-	-							
ADD	R5 ← R7, R4						F	D	-							

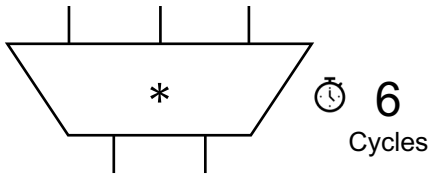
Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	1		2
R4	1		4
R5	0	d	
R6	1		6
R7	1		8
R8	1		8
R9	1		9
R10	0	c	
R11	0	y	

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a	1	~	2	1	~	4
b						
c	1	~	8	1	~	9
d	1	~	4	1	~	8

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x						
y	1	~	8	0	c	
z						
t						





Out-of-order execution: Example

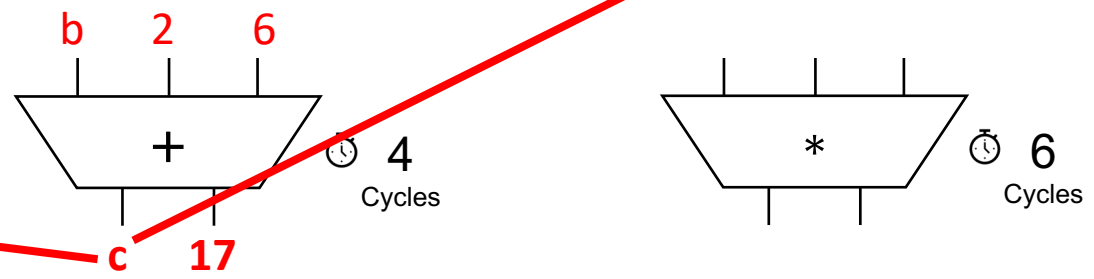
	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1	X_2	X_3	X_4	X_5	X_6							
ADD	R5 ← R3, R4		F	D	-	-	-	-	-	X_1						
ADD	R7 ← R2, R6			F	D	X_1	X_2	X_3	X_4							
ADD	**R10 ← R8, R9**				F	D	X_1	X_2	X_3	**X_4**						
MUL	R11 ← R7, R10					F	D	-	-	-						
ADD	R5 ← R7, R4						F	D	-							

Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	1		2
R4	1		4
R5	0	d	
R6	1		6
R7	1		8
R8	1		8
R9	1		9
R10	**1**		**17**
R11	0	y	

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a	1	~	2	1	~	4
b						
c						
d	1	~	4	1	~	8

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x						
y	1	~	8	**1**	**~**	**17**
z						
t						



Out-of-order execution: Example

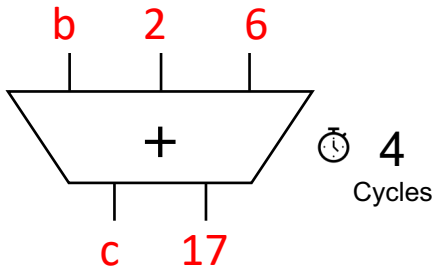
	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1	X_2	X_3	X_4	X_5	X_6							
ADD	R5 ← R3, R4		F	D	-	-	-	-	-	X_1						
ADD	R7 ← R2, R6			F	D	X_1	X_2	X_3	X_4							
ADD	R10 ← R8, R9				F	D	X_1	X_2	X_3	X_4						
MUL	R11 ← R7, R10					F	D	-	-	-						
ADD	**R5 ← R7, R4**						F	D	-	-						

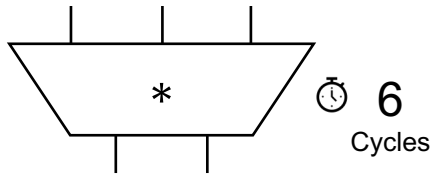
Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	1		2
R4	1		4
R5	0	d	
R6	1		6
R7	1		8
R8	1		8
R9	1		9
R10	**1**		**17**
R11	0	y	

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a	1	~	2	1	~	4
b						
c						
d	**1**	**~**	**4**	**1**	**~**	**8**

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x						
y	1	~	8	**1**	**~**	**17**
z						
t						





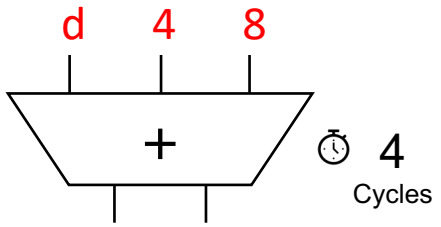
Out-of-order execution: Example

	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1	X_2	X_3	X_4	X_5	X_6							
ADD	R5 ← R3, R4		F	D	-	-	-	-	-	X_1	X_2					
ADD	R7 ← R2, R6			F	D	X_1	X_2	X_3	X_4							
ADD	R10 ← R8, R9				F	D	X_1	X_2	X_3	X_4						
MUL	**R11 ← R7, R10**					F	D	-	-	-	**X_1**					
ADD	R5 ← R7, R4						F	D	-	-	X_1					

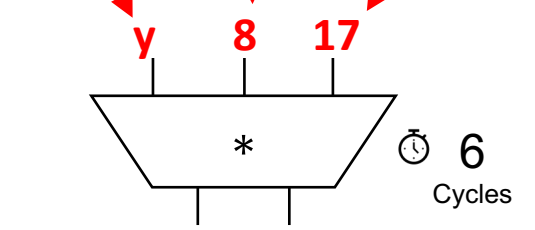
Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	1		2
R4	1		4
R5	0	d	
R6	1		6
R7	1		8
R8	1		8
R9	1		9
R10	1		17
R11	0	y	

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a	1	~	2	1	~	4
b						
c						
d	1	~	4	1	~	8



	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x						
y	1	~	8	1	~	17
z						
t						



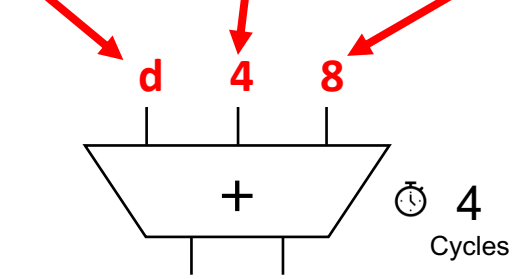
Out-of-order execution: Example

	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1	X_2	X_3	X_4	X_5	X_6							
ADD	R5 ← R3, R4		F	D	-	-	-	-	-	X_1	X_2					
ADD	R7 ← R2, R6			F	D	X_1	X_2	X_3	X_4							
ADD	R10 ← R8, R9				F	D	X_1	X_2	X_3	X_4						
MUL	R11 ← R7, R10					F	D	-	-	-	X_1					
ADD	**R5 ← R7, R4**						F	D	-	-	**X_1**					

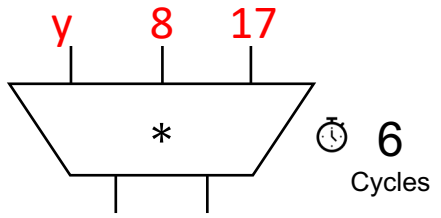
Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	1		2
R4	1		4
R5	0	d	
R6	1		6
R7	1		8
R8	1		8
R9	1		9
R10	1		17
R11	0	y	

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a	1	~	2	1	~	4
b						
c						
d	1	~	4	1	~	8



	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x						
y	1	~	8	1	~	17
z						
t						



Out-of-order execution: Example

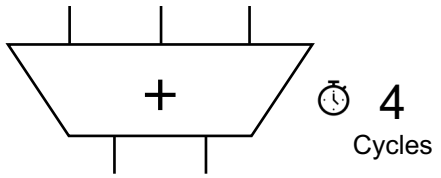
	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1	X_2	X_3	X_4	X_5	X_6							
ADD	R5 ← R3, R4		F	D	-	-	-	-	-	X_1	X_2	X_3				
ADD	R7 ← R2, R6			F	D	X_1	X_2	X_3	X_4							
ADD	R10 ← R8, R9				F	D	X_1	X_2	X_3	X_4						
MUL	R11 ← R7, R10					F	D	-	-	-	X_1	X_2				
ADD	R5 ← R7, R4						F	D	-	-	X_1	X_2				

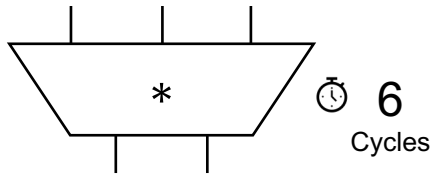
Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	1		2
R4	1		4
R5	0	d	
R6	1		6
R7	1		8
R8	1		8
R9	1		9
R10	1		17
R11	0	y	

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a	1	~	2	1	~	4
b						
c						
d	1	~	4	1	~	8

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x						
y	1	~	8	1	~	17
z						
t						





Out-of-order execution: Example

ADD R5 ← R3, R4 has no consumers! Dead code!

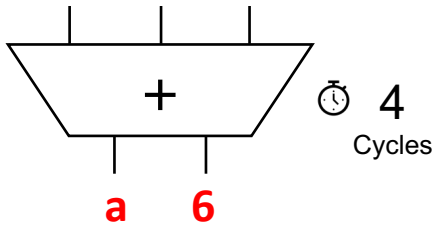
	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1	X_2	X_3	X_4	X_5	X_6							
ADD	**R5 ← R3, R4**		F	D	-	-	-	-	-	X_1	X_2	X_3	**X_4**			
ADD	R7 ← R2, R6			F	D	X_1	X_2	X_3	X_4							
ADD	R10 ← R8, R9				F	D	X_1	X_2	X_3	X_4						
MUL	R11 ← R7, R10					F	D	-	-	-	X_1	X_2				
ADD	R5 ← R7, R4						F	D	-	-	X_1	X_2				

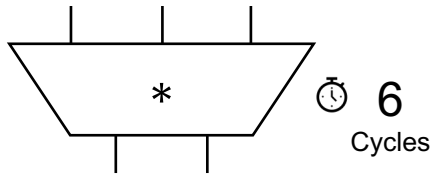
Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	1		2
R4	1		4
R5	0	d	
R6	1		6
R7	1		8
R8	1		8
R9	1		9
R10	1		17
R11	0	y	

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a	1	~	2	1	~	4
b						
c						
d	1	~	4	1	~	8

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x						
y	1	~	8	1	~	17
z						
t						





Out-of-order execution: Example

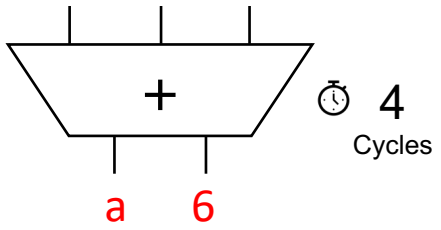
		Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2		F	D	X_1	X_2	X_3	X_4	X_5	X_6							
ADD	R5 ← R3, R4			F	D	-	-	-	-	-	X_1	X_2	X_3	X_4			
ADD	R7 ← R2, R6				F	D	X_1	X_2	X_3	X_4							
ADD	R10 ← R8, R9					F	D	X_1	X_2	X_3	X_4						
MUL	R11 ← R7, R10						F	D	-	-	-	X_1	X_2	X_3			
ADD	R5 ← R7, R4							F	D	-	-	X_1	X_2	X_3			

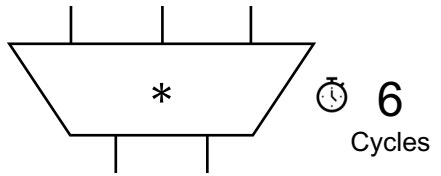
Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	1		2
R4	1		4
R5	0	d	
R6	1		6
R7	1		8
R8	1		8
R9	1		9
R10	1		17
R11	0	y	

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a						
b						
c						
d	1	~	4	1	~	8

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x						
y	1	~	8	1	~	17
z						
t						





Out-of-order execution: Example

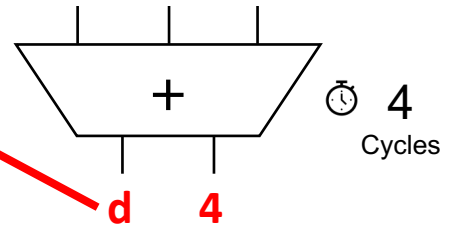
	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1	X_2	X_3	X_4	X_5	X_6							
ADD	R5 ← R3, R4		F	D	-	-	-	-	-	X_1	X_2	X_3	X_4			
ADD	R7 ← R2, R6			F	D	X_1	X_2	X_3	X_4							
ADD	R10 ← R8, R9				F	D	X_1	X_2	X_3	X_4						
MUL	R11 ← R7, R10					F	D	-	-	-	X_1	X_2	X_3	X_4		
ADD	**R5 ← R7, R4**						F	D	-	-	X_1	X_2	X_3	**X_4**		

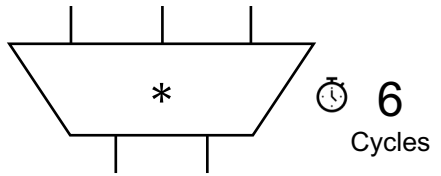
Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	1		2
R4	1		4
R5	0	d	
R6	1		6
R7	1		8
R8	1		8
R9	1		9
R10	1		17
R11	0	y	

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a						
b						
c						
d	1	~	4	1	~	8

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x						
y	1	~	8	1	~	17
z						
t						





Out-of-order execution: Example

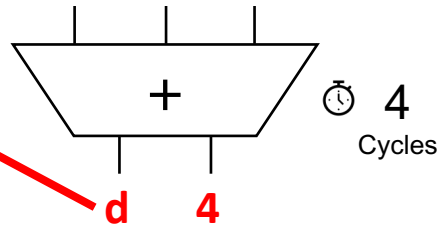
	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1	X_2	X_3	X_4	X_5	X_6							
ADD	R5 ← R3, R4		F	D	-	-	-	-	-	X_1	X_2	X_3	X_4			
ADD	R7 ← R2, R6			F	D	X_1	X_2	X_3	X_4							
ADD	R10 ← R8, R9				F	D	X_1	X_2	X_3	X_4						
MUL	R11 ← R7, R10					F	D	-	-	-	X_1	X_2	X_3	X_4		
ADD	**R5 ← R7, R4**						F	D	-	-	X_1	X_2	X_3	**X_4**		

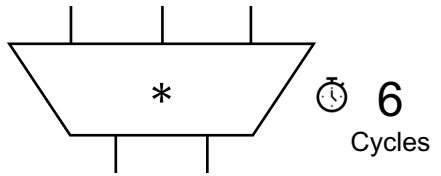
Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	1		2
R4	1		4
R5	**1**		**4**
R6	1		6
R7	1		8
R8	1		8
R9	1		9
R10	1		17
R11	0	y	

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a						
b						
c						
d						

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x						
y	1	~	8	1	~	17
z						
t						





Out-of-order execution: Example

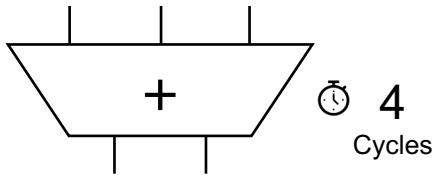
	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1	X_2	X_3	X_4	X_5	X_6							
ADD	R5 ← R3, R4		F	D	-	-	-	-	-	X_1	X_2	X_3	X_4			
ADD	R7 ← R2, R6			F	D	X_1	X_2	X_3	X_4							
ADD	R10 ← R8, R9				F	D	X_1	X_2	X_3	X_4						
MUL	R11 ← R7, R10					F	D	-	-	-	X_1	X_2	X_3	X_4	X_5	
ADD	R5 ← R7, R4						F	D	-	-	X_1	X_2	X_3	X_4		

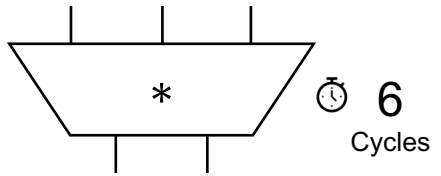
Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	1		2
R4	1		4
R5	1		4
R6	1		6
R7	1		8
R8	1		8
R9	1		9
R10	1		17
R11	0	y	

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a						
b						
c						
d						

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x						
y	1	~	8	1	~	17
z						
t						





Out-of-order execution: Example

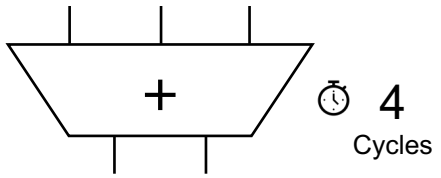
	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1	X_2	X_3	X_4	X_5	X_6							
ADD	R5 ← R3, R4		F	D	-	-	-	-	-	X_1	X_2	X_3	X_4			
ADD	R7 ← R2, R6			F	D	X_1	X_2	X_3	X_4							
ADD	R10 ← R8, R9				F	D	X_1	X_2	X_3	X_4						
MUL	**R11 ← R7, R10**					F	D	-	-	-	X_1	X_2	X_3	X_4	X_5	**X_6**
ADD	R5 ← R7, R4						F	D	-	-	X_1	X_2	X_3	X_4		

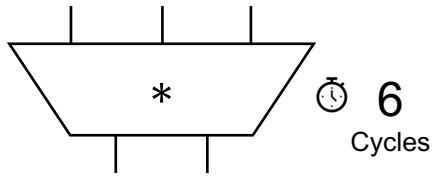
Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	1		2
R4	1		4
R5	1		4
R6	1		6
R7	1		8
R8	1		8
R9	1		9
R10	1		17
R11	0	y	

Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a						
b						
c						
d						

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x						
y	1	~	8	1	~	17
z						
t						





y 136

Out-of-order execution: Example

	Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MUL	R3 ← R1, R2	F	D	X_1	X_2	X_3	X_4	X_5	X_6							
ADD	R5 ← R3, R4		F	D	-	-	-	-	-	X_1	X_2	X_3	X_4			
ADD	R7 ← R2, R6			F	D	X_1	X_2	X_3	X_4							
ADD	R10 ← R8, R9				F	D	X_1	X_2	X_3	X_4						
MUL	**R11 ← R7, R10**					F	D	-	-	-	X_1	X_2	X_3	X_4	X_5	**X_6**
ADD	R5 ← R7, R4						F	D	-	-	X_1	X_2	X_3	X_4		

Register	Valid	Tag	Value
R1	1		1
R2	1		2
R3	1		2
R4	1		4
R5	1		4
R6	1		6
R7	1		8
R8	1		8
R9	1		9
R10	1		17
R11	**1**		**~~136~~**

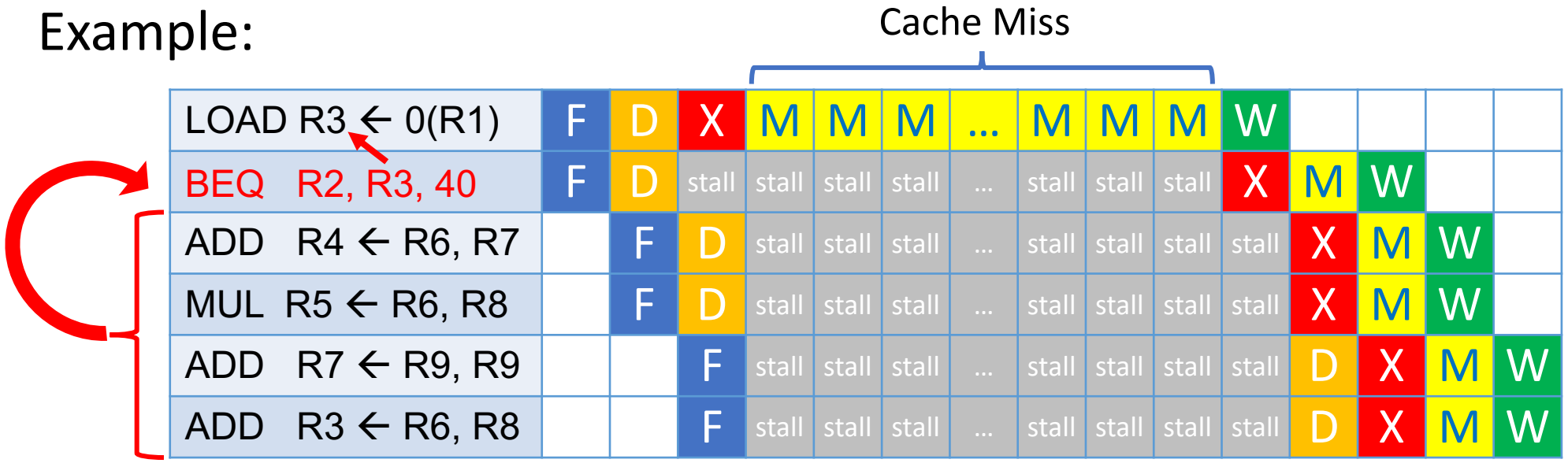
Register Alias Table

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
a						
b						
c						
d						

	Source 1			Source 2		
	V	Tag	Value	V	Tag	Value
x						
y						
z						
t						

But: Long-latency operations + Control dependencies

Example:



Branch (BEQ) determines further instructions to execute!

- BEQ cannot execute because its source register R3 is unavailable
- Out-of-order execution will not help…

Solution: **Speculative execution**

Speculative execution

Predict outcome of branches + Execute instructions speculatively

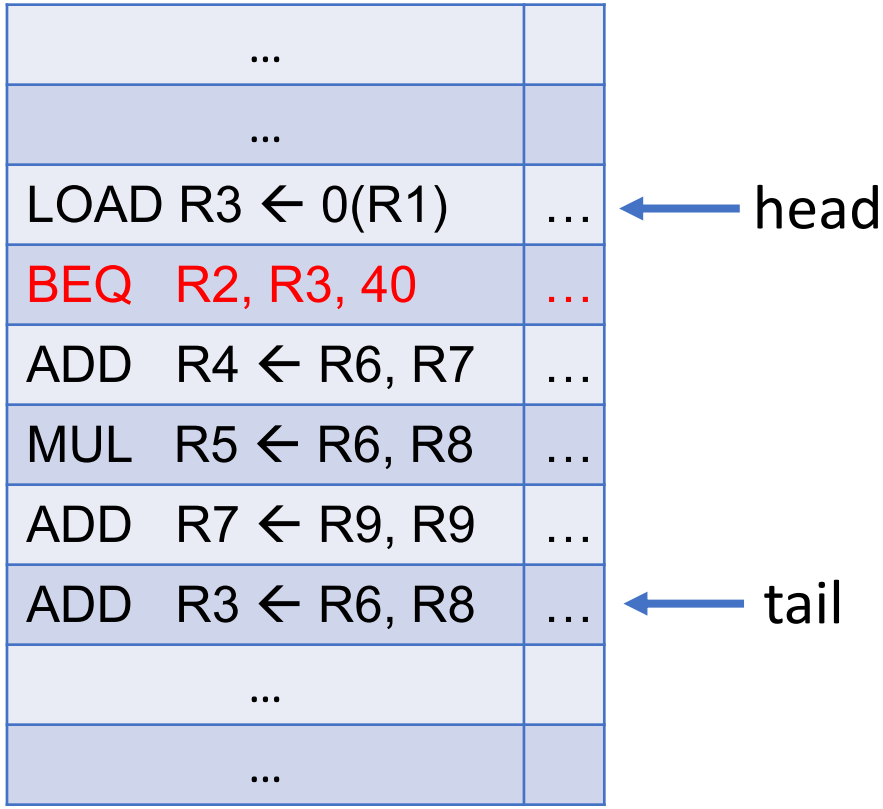
- Learn from previous branch outcomes
- Increasingly sophisticated, including use of machine learning techniques

- Predictions may be wrong
- Need mechanism to recover in such cases

Speculative execution: Implementation in a nutshell

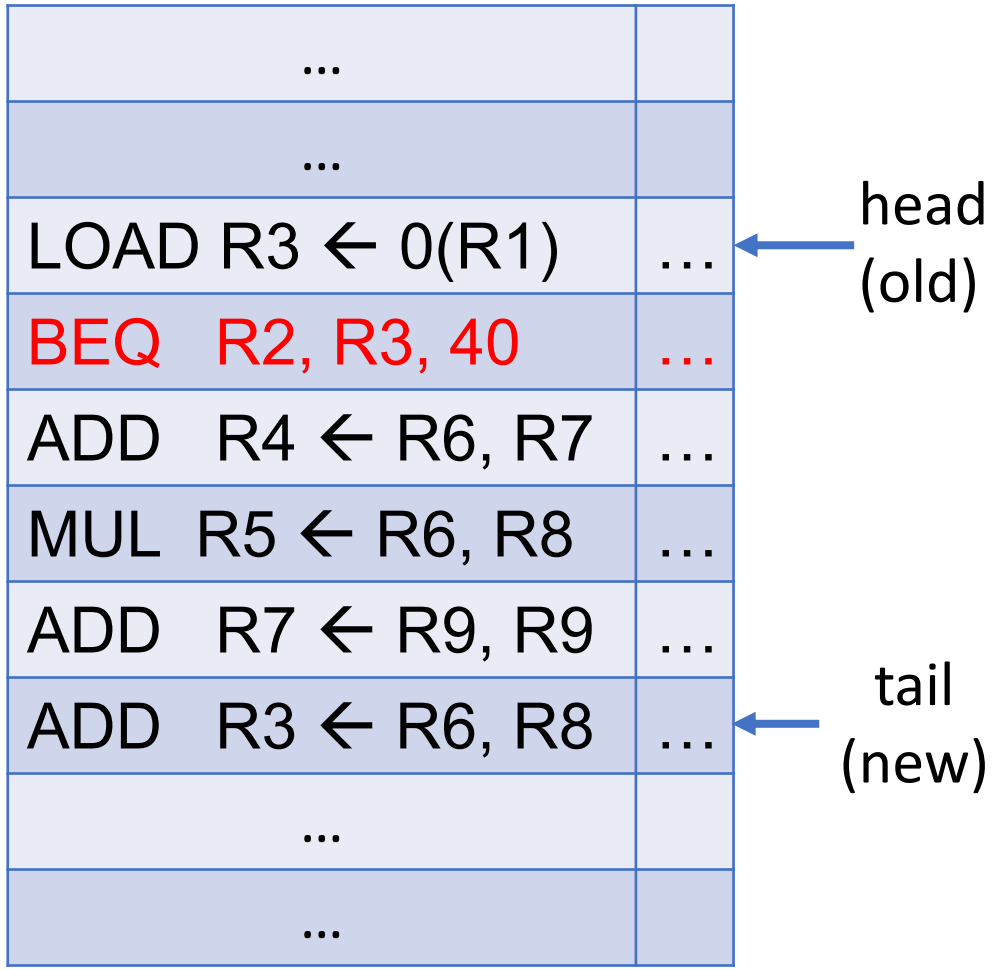
- *Idea:*
  Complete instructions out-of-order, but reorder them before making results visible to architectural state

- Key hardware data structure:
  Reorder buffer (ROB)
  Buffers information about
  all instructions that
  have been **decoded,**
  but have **not yet retired.**



Speculative execution: Reorder Buffer

- When instruction is **decoded**, it reserves the next entry in the ROB
- When instruction **completes**, it writes result into ROB entry
- When instruction **oldest in ROB** and it has completed without exceptions, its result is moved to register file or memory
- When **misprediction** is detected, remove entries from ROB



Challenges

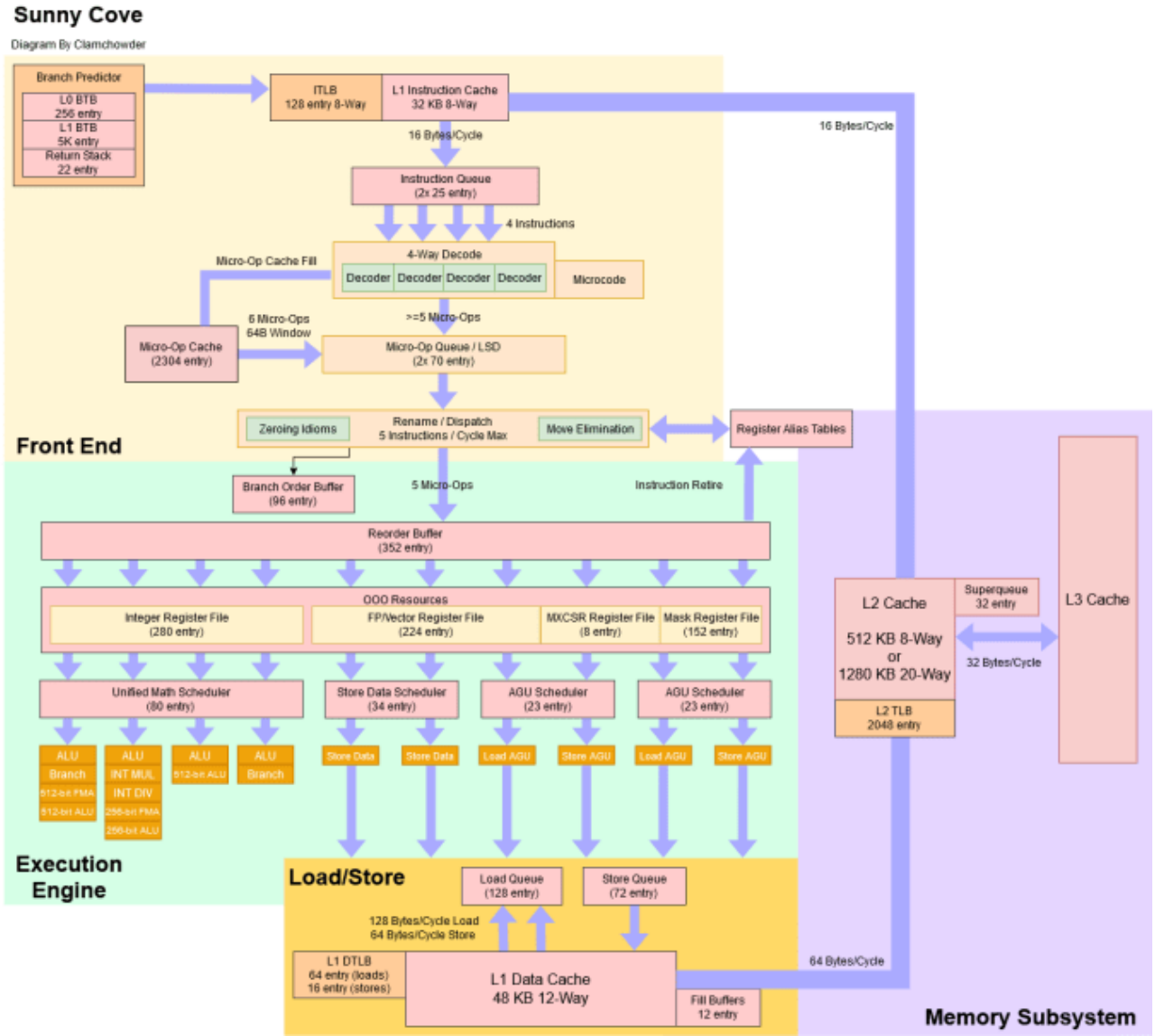
Deeper pipelines and superscalar pipelines increase **potential parallelism**

But can we actually fill these pipelines?

Challenges:

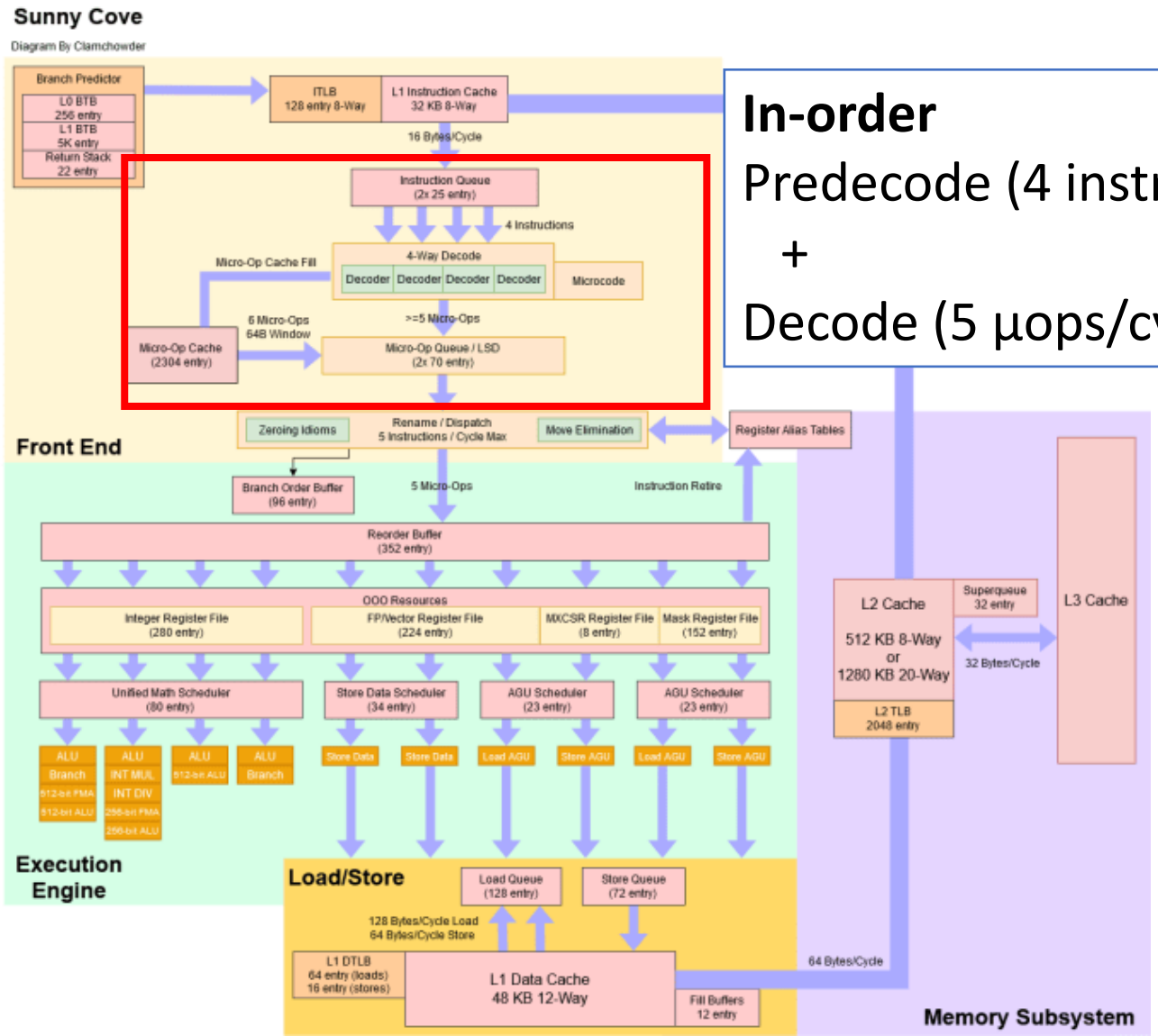
- Long-latency operations, such as cache misses
- Data dependencies → **Out-of-order Execution**
- Control dependencies → **Speculative Execution**

A peek at real-world microarchitectures: Sunny Cove (10th gen Intel Core microarchitecture, 2019)



[Source: https://chipsandcheese.com/2022/06/07/sunny-cove-intels-lost-generation/]

A peek at real-world microarchitectures: Sunny Cove (10th gen Intel Core microarchitecture, 2019)



In-order

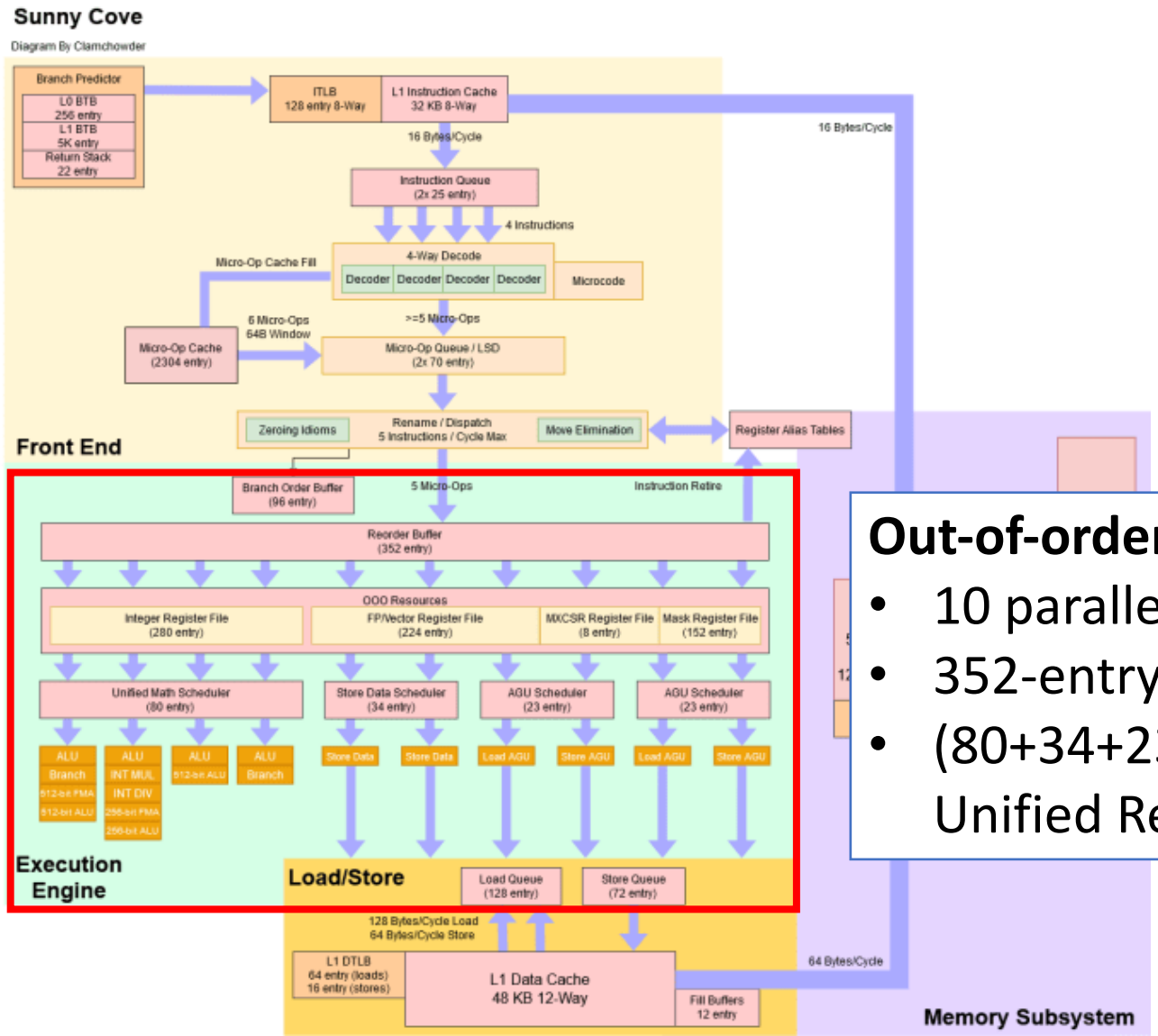
Predecode (4 instructions/cycle)

+

Decode (5 µops/cycle)

[Source: https://chipsandcheese.com/2022/06/07/sunny-cove-intels-lost-generation/]

A peek at real-world microarchitectures: Sunny Cove (10th gen Intel Core microarchitecture, 2019)

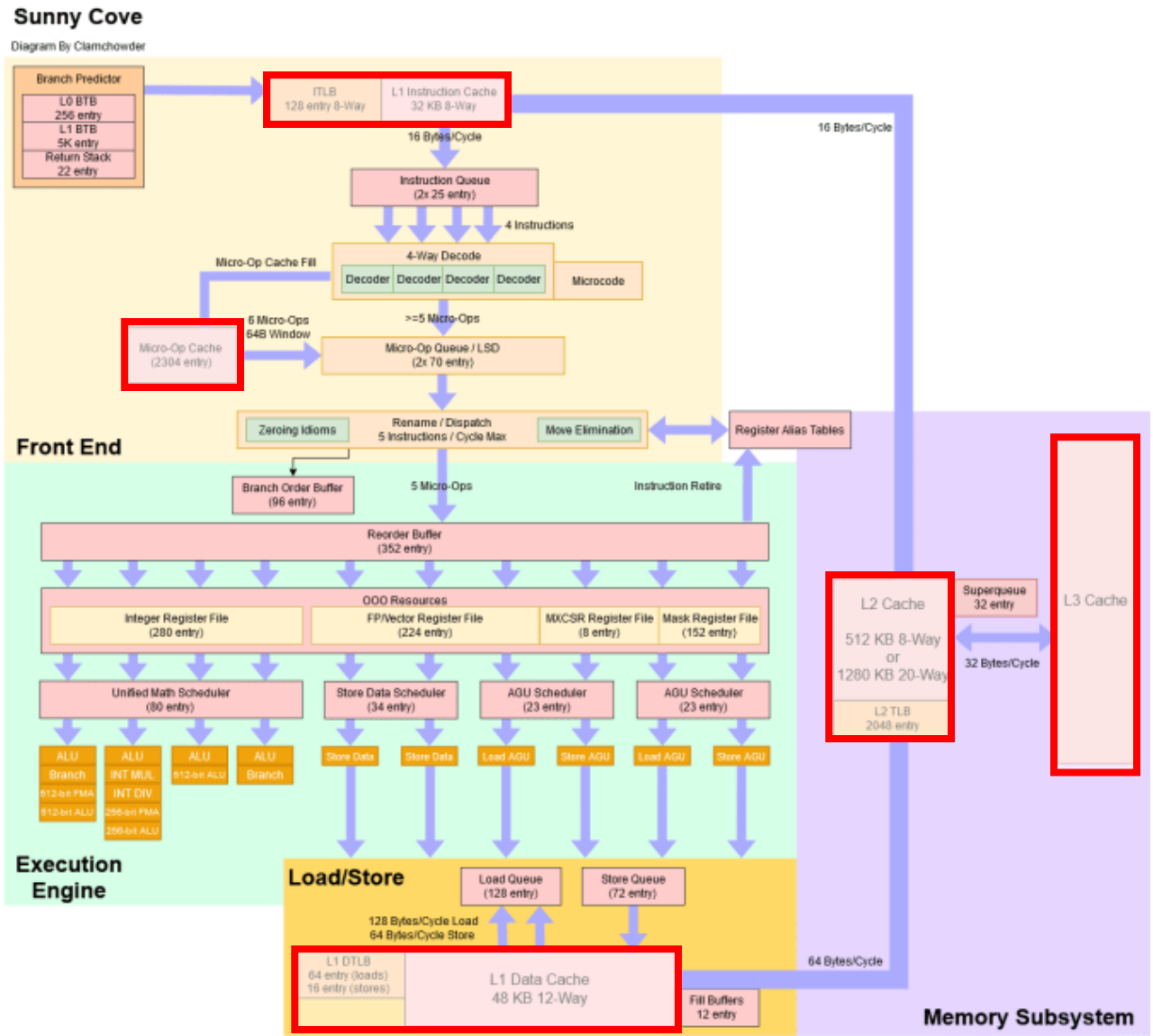


Out-of-order Execution Engine

- 10 parallel Execution Ports
- 352-entry Reorder Buffer
- (80+34+23+23)-entry Unified Reservation Stations

[Source: https://chipsandcheese.com/2022/06/07/sunny-cove-intels-lost-generation/]

A peek at real-world microarchitectures: Sunny Cove (10th gen Intel Core microarchitecture, 2019)



Memory Hierarchy:
32 KB L1 Instruction Cache
32 KB L1 Data Cache
512-1280 KB L2 Cache
3 MB/core L3 Cache

2304 entry µop Cache

+ TLBs + …

[Source: https://chipsandcheese.com/2022/06/07/sunny-cove-intels-lost-generation/]

More Details

Abel and Reineke:
uops.info: Characterizing Latency, Throughput, and Port Usage of Instructions on Intel Microarchitectures
ASPLOS 2019

http://uops.info/

Abel and Reineke:
uiCA: Accurate Throughput Prediction of Basic Blocks on Recent Intel Microarchitectures
International Conference on Supercomputing 2022

http://uica.uops.info/