

[cin.ufpe.br](http://cin.ufpe.br)



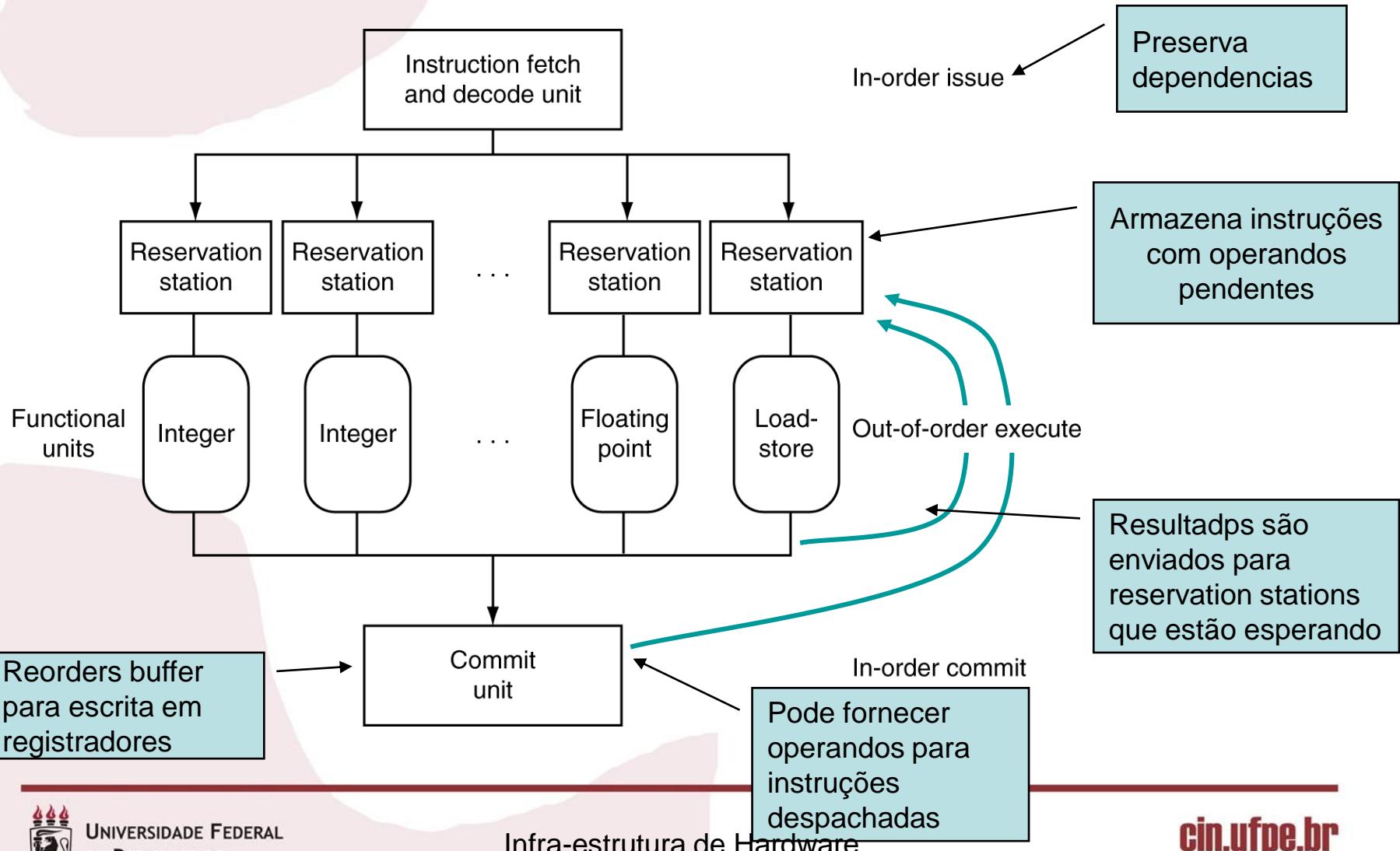
# Centro de Informática

U · F · P · E



UNIVERSIDADE FEDERAL DE PERNAMBUCO  
Infra-estrutura de Hardware

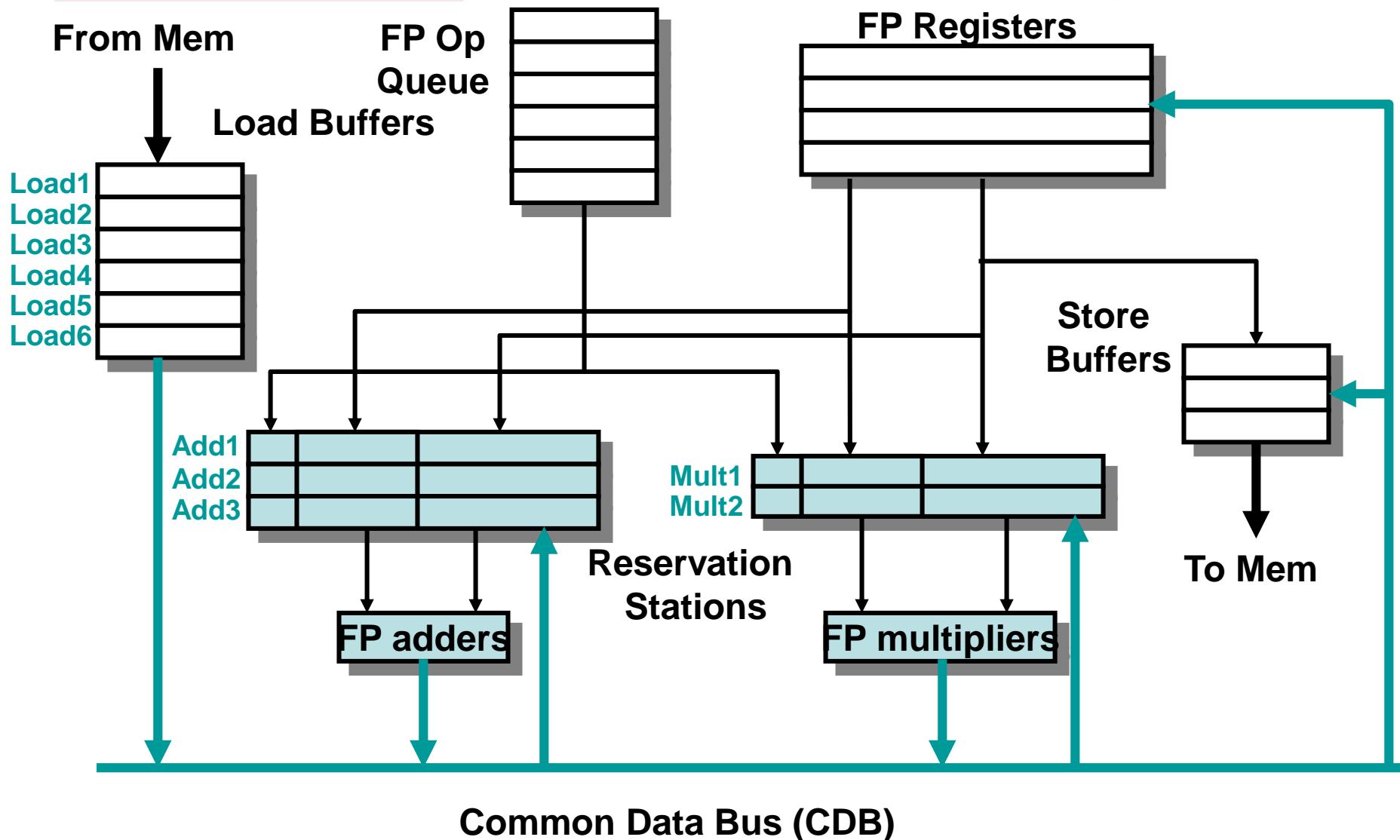
# Pipeline com Escalonamento Dinâmico



# Escalonamento Dinâmico – Despacho simples

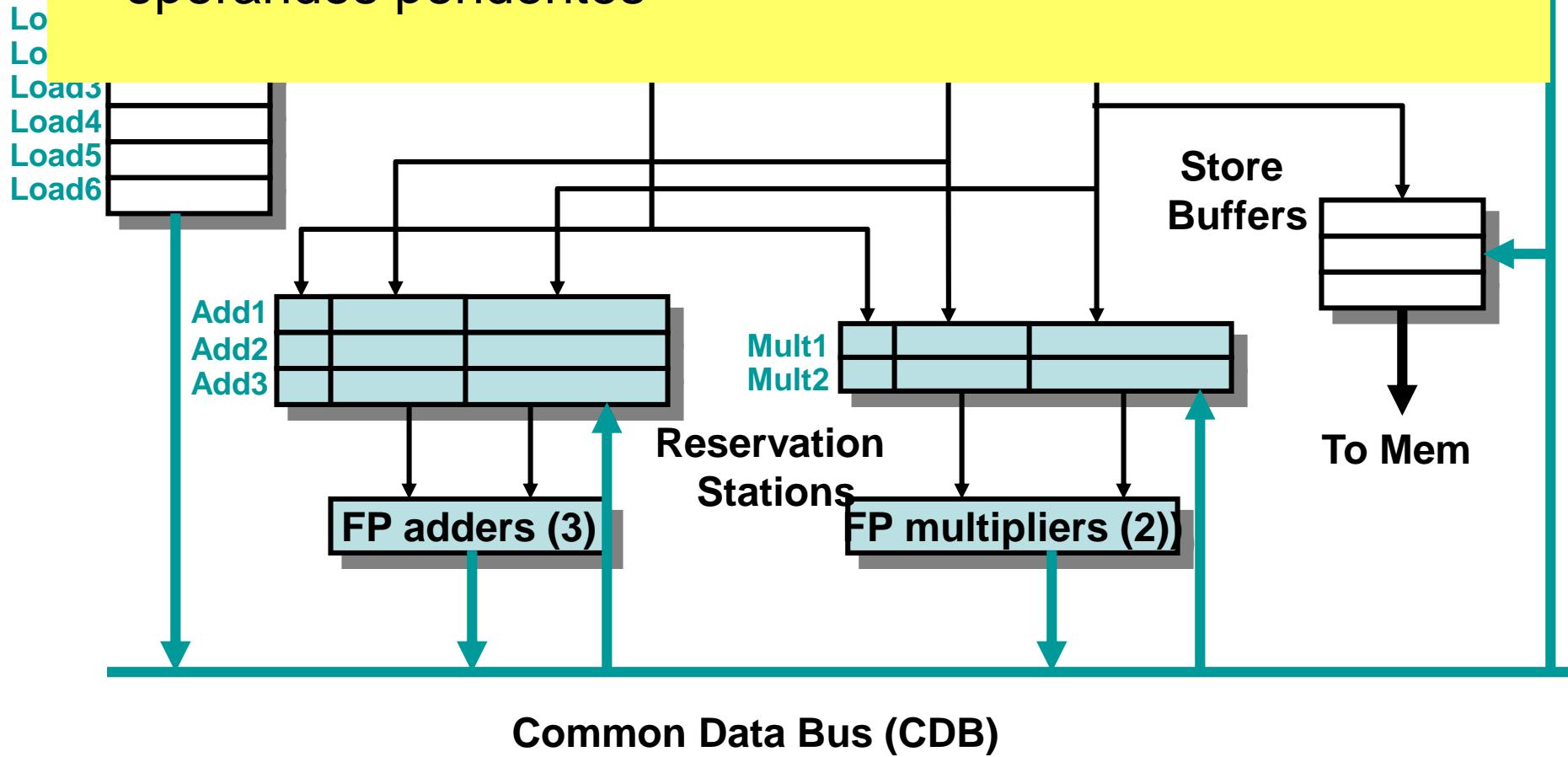
- Baseado no Técnica proposta por Roberto Tomasulo
  - IBM 360/91
  - não havia caches; tempo de acesso à memória grande
  - instruções de FP com grandes latências (delay))
- Como garantir alto desempenho sem compilador especial
  - Um pequeno número de registradores floating point (4 no 360) dificultava escalonamento das operações pelo compilador.
- Tomasulo: Como ter efetivamente mais registradores ?  
Como resolver os conflitos devido à dependência de dados
  - seguir quando os operandos estiverem prontos e renomeamento implementado no hardware!
- Usado nos processadores:
  - Alpha 21264, HP 8000, MIPS 10000, Pentium III, PowerPC 604, ...

# Estrutura Básica para Escalonamento Dinâmico



# Estrutura Básica para Escalonamento Dinâmico

- Controle & buffers distribuídos na Function Units (FU)
- FU buffers chamados de “reservation stations”; mantém operandos pendentes



# Reservation Station

Load/Store

Imediato/Endereço Efetivo:

Mantém informação sobre o end. de memória  
calculado para instruções de load ou store



0/1: Busy: Indica que a Reservation Station e sua FU estão ocupadas

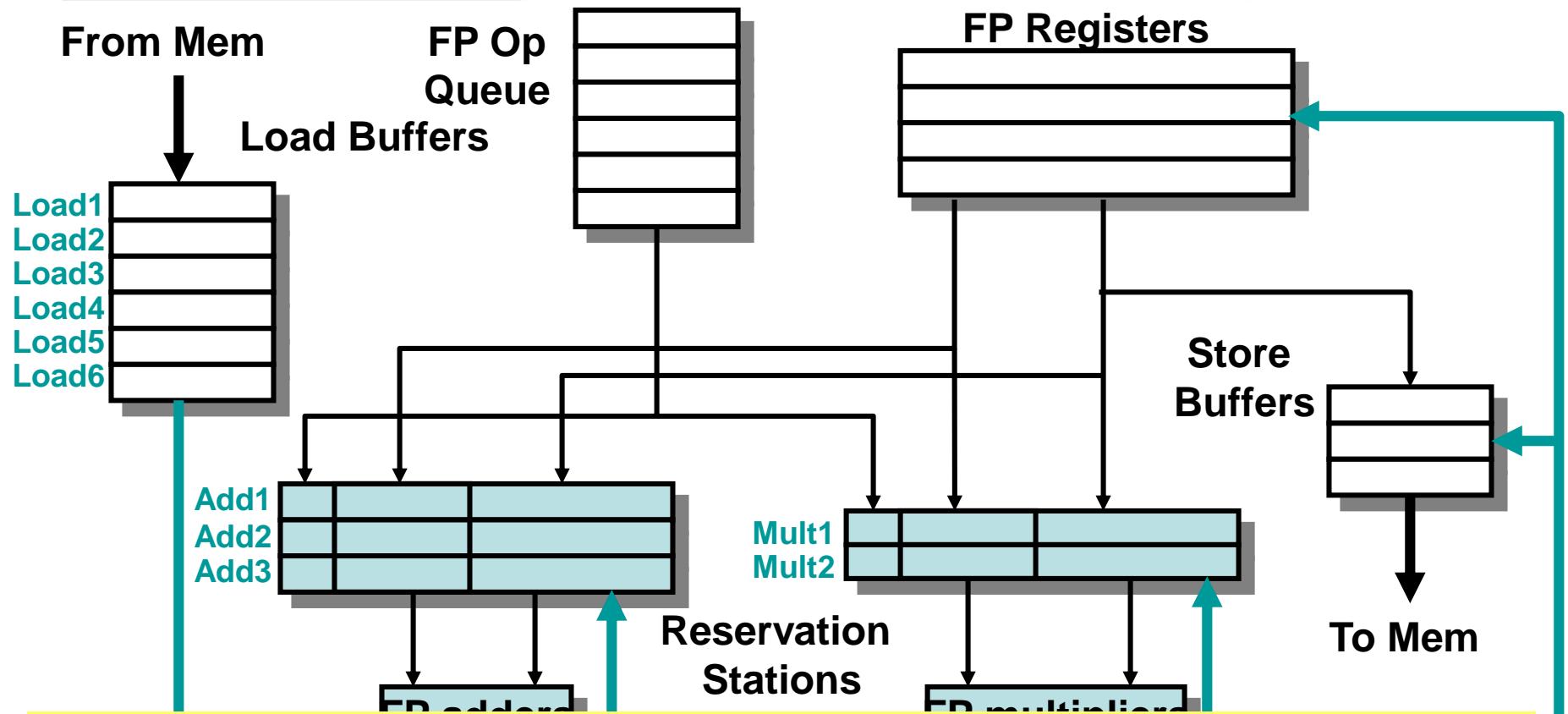
Operação: Operação a ser executada na unidade

OBS.: Register File  
 $Q_i = \text{No. RS}$

número das Reservation Stations que produzirão os operandos correspondentes (valores a serem escritos)  
•  $Q_j, Q_k = 0 \Rightarrow \text{ready}$   
• Store buffers tem somente  $Q_i$  para RS que produz resultado

V<sub>j</sub>, V<sub>k</sub>: Valores dos operantes Fontes  
Store buffers tem campos V, resultados devem ser armazenados

# Estrutura Básica para Escalonamento Dinâmico



Resultados enviados da Reservation Station para a FU, ([sem usar os registradores](#)) através de broadcast dos resultados para todas as FUs usando o [Common Data Bus](#)

# Escalonamento Dinâmico



1. **Issue**— pega a instrução na “FP Op Queue” e decodifica. Se a **reservation station** da instrução está livre (não há conflito estrutural), despacha a instrução e envia operandos disponíveis.
  
2. **Execute** —executa a operação sobre os operandos (EX) se os dois operandos estão disponíveis. Se algum operando não estiver pronto, monitora o **Common Data Bus** (espera pelo cálculo do operando, essa espera resolve RAW). Quando um operando está pronto atualiza **reservation table**)

# Escalonamento Dinâmico



## 3. Write result — termina a execução (WB)

Os resultados calculados pelas reservation units são enviados por Broadcast via **Common Data Bus** para todas unidades; marca a **reservation station** como disponível. Escreve no registrador.

# Exemplo do Alg. Tomasulo

- Trecho de programa a ser executado:

1	L.D	F6,34(R2)
2	L.D	F2,45(R3)
3	MUL.D	F0,F2,F4
4	SUB.D	F8,F2,F6
5	DIV.D	F10,F0,F6
6	ADD.D	F6,F8,F2

RAW?: (1-4); (1-5); (2-3); (2-4); (2-6); ....

# Exemplo do Alg. Tomasulo

- Assumir as seguintes latências:
  - Load: 1 ciclo
  - Add: 2 ciclos
  - Multiplicação: 10 ciclos
  - Divisão: 40 ciclos
- Load-Store:
  - Calcula o endereço efetivo (FU)
  - Instruções de Load ou Store usam buffers
  - Acesso à memória (somente load)
  - Escrita de resultado
    - Load: envia o valor para o registrador e/ou reservation stations
    - Store: escreve o valor na memória
    - (escritas somente no estágio “WB” – simplifica o algoritmo de Tomasulo)

# Exemplo do Alg. Tomasulo

## Instruções do programa

### *Instruction status:*

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

3 estágios da execução

Busy	Address
Load1	No
Load2	No
Load3	No

3 Load/Buffers

### *Reservation Stations:*

Time	Name	Busy	Op	S1	S2	RS	RS
	Add1	No					
	Add2	No					
	Add3	No					
	Mult1	No					
	Mult2	No					

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

### *Register result status:*

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

Clock cycle

# Exemplo Tomasulo: Ciclo 1

## Instruction status:

Instruction	<i>j</i>	<i>k</i>		<i>Issue</i>	<i>Exec</i>	<i>Write</i>
				<i>Comp</i>	<i>Result</i>	
LD	F6	34+	R2	1		
LD	F2	45+	R3			
MULTD	F0	F2	F4			
SUBD	F8	F6	F2			
DIVD	F10	F0	F6			
ADDD	F6	F8	F2			

	Busy	Address
Load1	Yes	34+R2
Load2	No	
Load3	No	

## Reservation Stations:

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

## Register result status:



# Exemplo Tomasulo: Ciclo 2

*Instruction status:*

Instruction	j	k	Issue	Exec	Write
				Comp	Result
LD	F6	34+	R2	1	
LD	F2	45+	R3	2	
MULTD	F0	F2	F4		
SUBD	F8	F6	F2		
DIVD	F10	F0	F6		
ADDD	F6	F8	F2		

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

*Reservation Stations:*

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

*Register result status:*

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

**Nota: pode haver múltiplos loads pendentes**

# Exemplo Tomasulo: Ciclo 3

*Instruction status:*

Instruction	j	k	Issue	Exec	Write	Busy	Address
				Comp	Result		
LD	F6	34+	R2	1	3	Load1	Yes 34+R2
LD	F2	45+	R3	2		Load2	Yes 45+R3
MULTD	F0	F2	F4	3		Load3	No
SUBD	F8	F6	F2				
DIVD	F10	F0	F6				
ADDD	F6	F8	F2				

*Reservation Stations:*

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

*Register result status:*

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

- Nota: nomes dos registradores são removidos (“renamed”) na Reservation Stations; MULT issued
- Load1 completa; alguém esperando por Load1?

# Exemplo Tomasulo: Ciclo 4

*Instruction status:*

Instruction	<i>j</i>	<i>k</i>		Exec			Write	Busy	Address
				Issue	Comp	Result			
LD	F6	34+	R2	1	2	4		Load1	No
LD	F2	45+	R3	2	4			Load2	Yes
MULTD	F0	F2	F4	3				Load3	45+R3
SUBD	F8	F6	F2	4					No
DIVD	F10	F0	F6						
ADDD	F6	F8	F2						

*Reservation Stations:*

Time	Name	Busy	Op	S1		S2		RS	
				<i>V<sub>j</sub></i>	<i>V<sub>l</sub></i>	<i>Q<sub>j</sub></i>	<i>Q<sub>l</sub></i>	<i>RS</i>	<i>RS</i>
Add1		Yes	SUBD	M(A1)				Load2	
Add2		No							
Add3		No							
Mult1		Yes	MULTD			R(F4)	Load2		
Mult2		No							

*Register result status:*

Clock	<i>F0</i>	<i>F2</i>	<i>F4</i>	<i>F6</i>	<i>F8</i>	<i>F10</i>	<i>F12</i>	...	<i>F30</i>
4	<i>FU</i>	Mult1	Load2		M(A1)	Add1			

- Load2 completa; alguém esperando por Load2?

# Exemplo Tomasulo: Ciclo 5

*Instruction status:*

Instruction	j	k	Issue	Exec	Write	Busy	Address
				Comp	Result		
LD	F6	34+	R2	1	3	4	Load1
LD	F2	45+	R3	2	4	5	Load2
MULTD	F0	F2	F4	3			Load3
SUBD	F8	F6	F2	4			
DIVD	F10	F0	F6	5			
ADDD	F6	F8	F2				

*Reservation Stations:*

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

*Register result status:*

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

- Timer inicia a contagem regressiva para Add1, Mult1

# Exemplo Tomasulo: Ciclo 6

*Instruction status:*

Instruction	j	k	Issue	Exec	Write	Busy	Address
				Comp	Result		
LD	F6	34+	R2	1	3	4	Load1
LD	F2	45+	R3	2	4	5	Load2
MULTD	F0	F2	F4	3			Load3
SUBD	F8	F6	F2	4			
DIVD	F10	F0	F6	5			
ADDD	F6	F8	F2	6			

*Reservation Stations:*

Time	Name	Busy	Op	S1	S2	RS	RS
				Vj	Vk	Qj	Qk
1	Add1	Yes	SUBD	M(A1)	M(A2)		
	Add2	Yes	ADDD		M(A2)	Add1	
	Add3	No					
9	Mult1	Yes	MULTD	M(A2)	R(F4)		
	Mult2	Yes	DIVD		M(A1)	Mult1	

*Register result status:*

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

- Despacha ADDD, dependência de nome em F6?

# Exemplo Tomasulo: Ciclo 7

*Instruction status:*

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

*Reservation Stations:*

Time	Name	<i>Busy</i>	<i>S1</i>	<i>S2</i>	<i>RS</i>	<i>RS</i>
			<i>Op</i>	<i>Vj</i>	<i>Vk</i>	<i>Qj</i>
0	Add1	Yes	SUBD	M(A1)	M(A2)	
	Add2	Yes	ADDD		M(A2)	Add1
	Add3	No				
8	Mult1	Yes	MULTD	M(A2)	R(F4)	
	Mult2	Yes	DIVD		M(A1)	Mult1

*Register result status:*

Clock	<i>F0</i>	<i>F2</i>	<i>F4</i>	<i>F6</i>	<i>F8</i>	<i>F10</i>	<i>F12</i>	...	<i>F30</i>
7	<i>FU</i>	Mult1	M(A2)		Add2	Add1	Mult2		

- Add1 (SUBD) completa; alguém esperando por add1?

# Exemplo Tomasulo: Ciclo 8

## Instruction status:

Instruction	<i>j</i>	<i>k</i>	Issue	<i>Exec</i>	<i>Write</i>	Busy	Address	
				Comp	Result			
LD	F6	34+	R2	1	3	4	Load1	No
LD	F2	45+	R3	2	4	5	Load2	No
MULTD	F0	F2	F4	3			Load3	No
SUBD	F8	F6	F2	4	7	8		
DIVD	F10	F0	F6	5				
ADDD	F6	F8	F2	6				

## Reservation Stations:

Time	Name	Busy	Op	<i>S1</i>	<i>S2</i>	<i>RS</i>	<i>RS</i>
				<i>Vj</i>	<i>Vk</i>	<i>Qj</i>	<i>Qk</i>
	Add1	No					
2	Add2	Yes	ADDD	(M-M)	M(A2)		
	Add3	No					
7	Mult1	Yes	MULTD	M(A2)	R(F4)		
	Mult2	Yes	DIVD		M(A1)	Mult1	

## Register result status:

Clock	<i>F0</i>	<i>F2</i>	<i>F4</i>	<i>F6</i>	<i>F8</i>	<i>F10</i>	<i>F12</i>	...	<i>F30</i>
8	FU	Mult1	M(A2)		Add2	(M-M)	Mult2		

# Exemplo Tomasulo: Ciclo 9

## Instruction status:

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

## Reservation Stations:

Time	Name	<i>Busy</i>	<i>S1</i>	<i>S2</i>	<i>RS</i>	<i>RS</i>	
			<i>Op</i>	<i>Vj</i>	<i>Vk</i>	<i>Qj</i>	<i>Qk</i>
	Add1	No					
1	Add2	Yes	ADDD	(M-M)	M(A2)		
	Add3	No					
6	Mult1	Yes	MULTD	M(A2)	R(F4)		
	Mult2	Yes	DIVD		M(A1)	Mult1	

## Register result status:

Clock	<i>F0</i>	<i>F2</i>	<i>F4</i>	<i>F6</i>	<i>F8</i>	<i>F10</i>	<i>F12</i>	...	<i>F30</i>
9	FU	Mult1	M(A2)		Add2	(M-M)	Mult2		

# Exemplo Tomasulo: Ciclo 10

## Instruction status:

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

## Reservation Stations:

Time	Name	<i>Busy</i>	<i>S1</i>	<i>S2</i>	<i>RS</i>	<i>RS</i>	
			<i>Op</i>	<i>Vj</i>	<i>Vk</i>	<i>Qj</i>	<i>Qk</i>
	Add1	No					
0	Add2	Yes	ADDD	(M-M)	M(A2)		
	Add3	No					
5	Mult1	Yes	MULTD	M(A2)	R(F4)		
	Mult2	Yes	DIVD		M(A1)	Mult1	

## Register result status:

Clock	<i>F0</i>	<i>F2</i>	<i>F4</i>	<i>F6</i>	<i>F8</i>	<i>F10</i>	<i>F12</i>	...	<i>F30</i>
10	FU	Mult1	M(A2)		Add2	(M-M)	Mult2		

- Add2 (ADDD) completa; alguém esperando por add2?

# Exemplo Tomasulo: Ciclo 11

*Instruction status:*

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

*Reservation Stations:*

Time	Name	Busy	Op	S1	S2	RS	RS
				Vj	Vk	Qj	Qk
	Add1	No					
	Add2	No					
	Add3	No					
4	Mult1	Yes	MULTD	M(A2)	R(F4)		
	Mult2	Yes	DIVD		M(A1)	Mult1	

*Register result status:*

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

- Resultado de ADDD é escrito!
- Todas as instruções mais rápidas terminam neste ciclo!

# Exemplo Tomasulo: Ciclo 12

## Instruction status:

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

## Reservation Stations:

Time	Name	<i>Busy</i>	<i>S1</i>	<i>S2</i>	<i>RS</i>	<i>RS</i>
			<i>Op</i>	<i>Vj</i>	<i>Vk</i>	<i>Qj</i>
	Add1	No				
	Add2	No				
	Add3	No				
3	Mult1	Yes	MULTD	M(A2)	R(F4)	
	Mult2	Yes	DIVD		M(A1)	Mult1

## Register result status:

Clock	<i>F0</i>	<i>F2</i>	<i>F4</i>	<i>F6</i>	<i>F8</i>	<i>F10</i>	<i>F12</i>	...	<i>F30</i>
12	<i>FU</i>	Mult1	M(A2)		(M-M+M)	(M-M)	Mult2		

# Exemplo Tomasulo: Ciclo 13

## Instruction status:

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

## Reservation Stations:

Time	Name	<i>Busy</i>	<i>S1</i>	<i>S2</i>	<i>RS</i>	<i>RS</i>
			<i>Op</i>	<i>Vj</i>	<i>Vk</i>	<i>Qj</i>
	Add1	No				
	Add2	No				
	Add3	No				
2	Mult1	Yes	MULTD	M(A2)	R(F4)	
	Mult2	Yes	DIVD		M(A1)	Mult1

## Register result status:

Clock	<i>F0</i>	<i>F2</i>	<i>F4</i>	<i>F6</i>	<i>F8</i>	<i>F10</i>	<i>F12</i>	...	<i>F30</i>
13	<i>FU</i>	Mult1	M(A2)		(M-M+M)	(M-M)	Mult2		

# Exemplo Tomasulo: Ciclo 14

## Instruction status:

Instruction	<i>j</i>	<i>k</i>		Exec Write			Busy	Address
				Issue	Comp	Result		
LD	F6	34+	R2	1	3	4	Load1	No
LD	F2	45+	R3	2	4	5	Load2	No
MULTD	F0	F2	F4	3			Load3	No
SUBD	F8	F6	F2	4	7	8		
DIVD	F10	F0	F6	5				
ADDD	F6	F8	F2	6	10	11		

## Reservation Stations:

Time	Name	Busy	Op	S1	S2	RS	RS
				Vj	Vk	Qj	Qk
	Add1	No					
	Add2	No					
	Add3	No					
1	Mult1	Yes	MULTD M(A2) R(F4)				
	Mult2	Yes	DIVD		M(A1) Mult1		

## Register result status:

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

# Exemplo Tomasulo: Ciclo 15

*Instruction status:*

Instruction	j	k	Issue	Exec	Write	Busy	Address	
				Comp	Result			
LD	F6	34+	R2	1	3	4	Load1	No
LD	F2	45+	R3	2	4	5	Load2	No
MULTD	F0	F2	F4	3	15		Load3	No
SUBD	F8	F6	F2	4	7	8		
DIVD	F10	F0	F6	5				
ADDD	F6	F8	F2	6	10	11		

*Reservation Stations:*

Time	Name	Busy	Op	S1	S2	RS	RS
				Vj	Vk	Qj	Qk
	Add1	No					
	Add2	No					
	Add3	No					
0	Mult1	Yes	MULTD	M(A2)	R(F4)		
	Mult2	Yes	DIVD		M(A1)	Mult1	

*Register result status:*

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

- Mult1 (MULTD) completa; alguém esperando por mult1?

# Exemplo Tomasulo: Ciclo 16

## Instruction status:

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

## Reservation Stations:

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

## Register result status:

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

- Agora é só esperar que Mult2 (DIVD) complete

# Pulando alguns ciclos

# Exemplo Tomasulo: Ciclo 55

## Instruction status:

Instruction	<i>j</i>	<i>k</i>		<i>Issue</i>	<i>Exec</i>	<i>Write</i>	<i>Busy</i>	<i>Address</i>
				<i>Comp</i>	<i>Result</i>			
LD	F6	34+	R2	1	3	4	Load1	No
LD	F2	45+	R3	2	4	5	Load2	No
MULTD	F0	F2	F4	3	15	16	Load3	No
SUBD	F8	F6	F2	4	7	8		
DIVD	F10	F0	F6	5				
ADDD	F6	F8	F2	6	10	11		

## Reservation Stations:

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

## Register result status:

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

# Exemplo Tomasulo: Ciclo 56

## Instruction status:

Instruction	j	k	Issue	Exec	Write	Busy	Address	
				Comp	Result			
LD	F6	34+	R2	1	3	4	Load1	No
LD	F2	45+	R3	2	4	5	Load2	No
MULTD	F0	F2	F4	3	15	16	Load3	No
SUBD	F8	F6	F2	4	7	8		
DIVD	F10	F0	F6	5	56			
ADDD	F6	F8	F2	6	10	11		

## Reservation Stations:

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

## Register result status:

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

- Mult2 (DIVD) completa; alguém esperando por mult2?

# Exemplo Tomasulo: Ciclo 57

*Instruction status:*

Instruction	j	k	Issue	Exec	Write	Busy	Address
				Comp	Result		
LD	F6	34+	R2	1	3	4	Load1
LD	F2	45+	R3	2	4	5	Load2
MULTD	F0	F2	F4	3	15	16	Load3
SUBD	F8	F6	F2	4	7	8	
DIVD	F10	F0	F6	5	56	57	
ADDD	F6	F8	F2	6	10	11	

*Reservation Stations:*

Time	Name	Busy	Op	V <sub>j</sub>	V <sub>k</sub>	RS	RS
	Add1	No					
	Add2	No					
	Add3	No					
	Mult1	No					
	Mult2	Yes	DIVD	M*F4	M(A1)		

*Register result status:*

Clock	F0	F2	F4	F6	F8	F10	F12	F30
57	FU	M*F4	M(A2)	(M-M+M)	(M-M)	Result		...

- Despacho en ordem, execução fora de ordem, término fora de ordem

# Como considerar previsão de desvio no escalonamento dinâmico das instruções

# Escalonamento Dinâmico e Previsão de Desvio

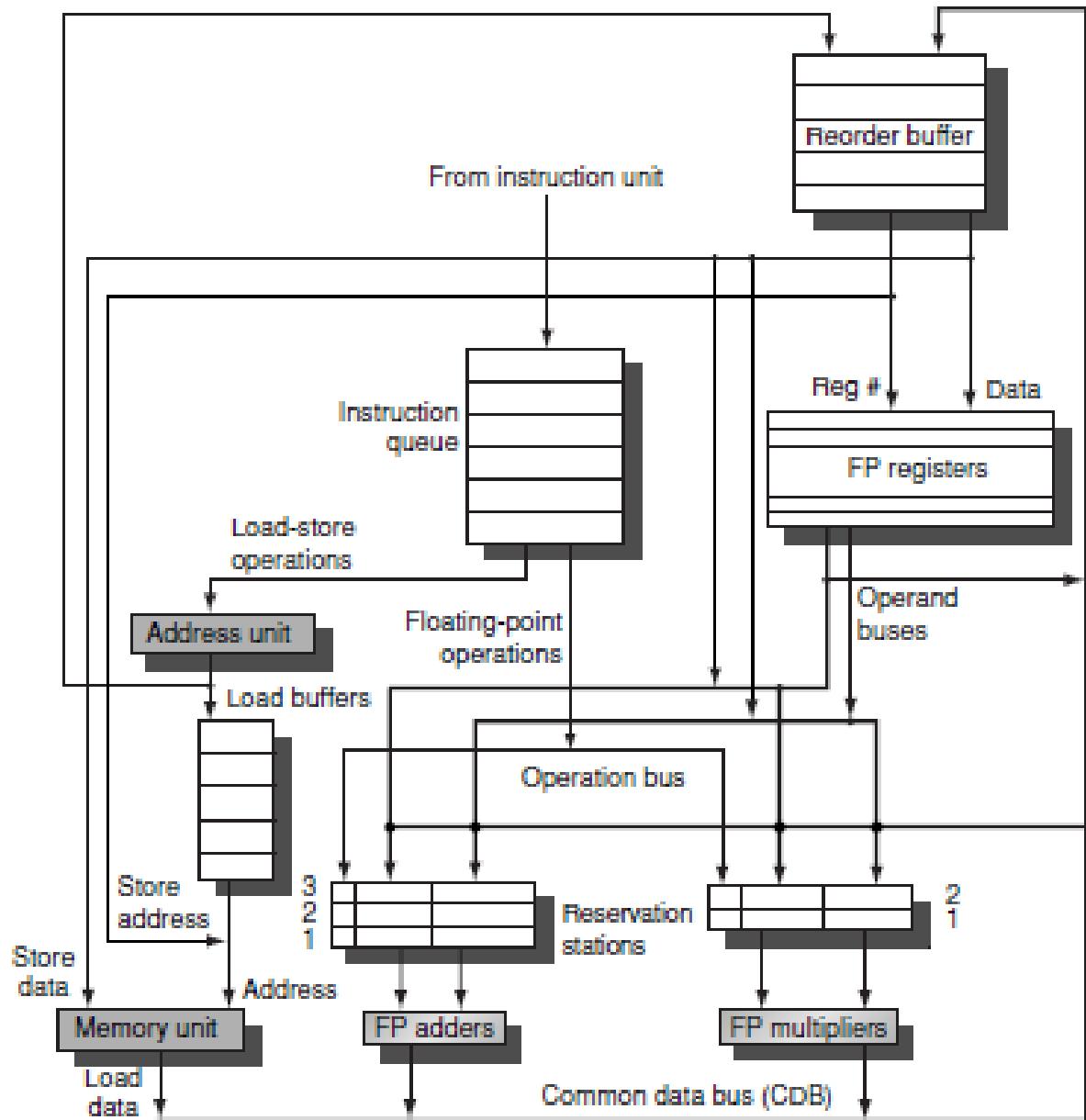


- Especulação: Uso da previsão de desvio no Escalonamento Dinâmico
- Especulação melhora desempenho MAS...
- Importante ter um bom mecanismo de previsão de desvio
- Se houve especulação incorreta é necessário voltar e reiniciar a execução a partir do ponto em que foi feita a previsão incorreta :
  - O estado do processador tem que ser recuperado e as instruções corretas devem ser executadas

# Escalonamento Dinâmico com Especulação



- Buffer para os resultados de instruções que não terminaram (uncommitted instructions): *reorder buffer*



# Escalonamento Dinâmico com Especulação

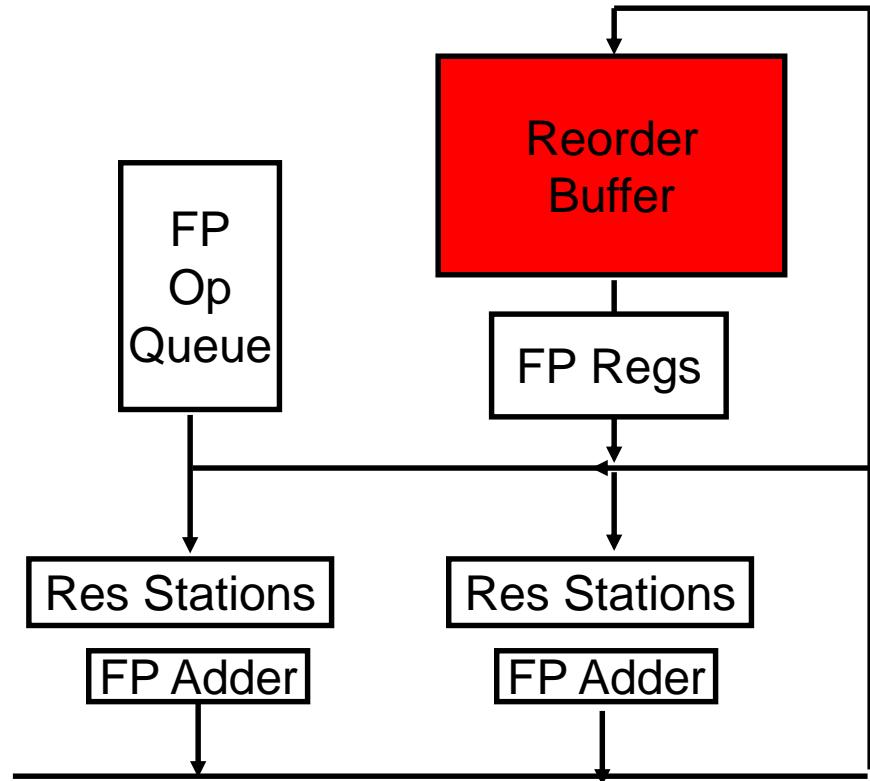


- Como garantir que a Especulação não vai introduzir erro?
- Técnica para especulação deve garantir *in-order completion* ou *commit*
- Término da instrução inclui duas etapas:
  - Término
  - Commit (escrita nos registradores)

# Suporte de HW Especulação

- Buffer para os resultados de instruções que não terminaram que preserva a ordem (uncommitted instructions):  
*reorder buffer*

- 3 campos: instr, reg. destino, valor
- *reorder buffer* armazena valores intermediários
- Depois que a instrução dá *commit*, o registrador é atualizado.
- Permite desfazer instruções especuladas devido a um desvio previsto erradamente



# Escalonamento Dinâmico com Especulação

## 1. Issue — pega a instrução da FP Op Queue

Se há **reservation station** e **reorder buffer slot** livres: **despacha instr** & envia operandos & **reorder buffer no.** para o destino (este estágio é comumente chamado de “**dispatch**”)

## 2. Execution — opera sobre os operandos (EX)

Quando ambos os operandos estão prontos executa; se não monitora o CDB a espera do resultado; quando ambos estiverem na reservation station, executa; verifica se há RAW (comumente chamado de “**issue**”)

## 3. Write result — termina a execução (WB)

Escreve, usando o Common Data Bus, em todas FUs que estão esperando por esse valor **e no reorder buffer**; marca a reservation station como disponível.

## 4. Commit — atualiza o registrador com o resultado do **reorder buffer**

Quando a instr. é a primeira no **reorder buffer** e o resultado está presente: atualiza o registrador com o resultado (ou store na memória) e remove a instr. do **reorder buffer**. Se foi um **Mispredicted branch** então **flushes reorder buffer**

# Escalonamento Dinâmico com Especulação



- Exemplo

- Código:

- L.D F0, 0(R1)

- MUL.D F4, F0, F2

- S.D F4, 0(R1)

- DADDIU R1, R1, #-8

- BNE R1, R2, Loop

- 2 iterações no loop

- Assumir

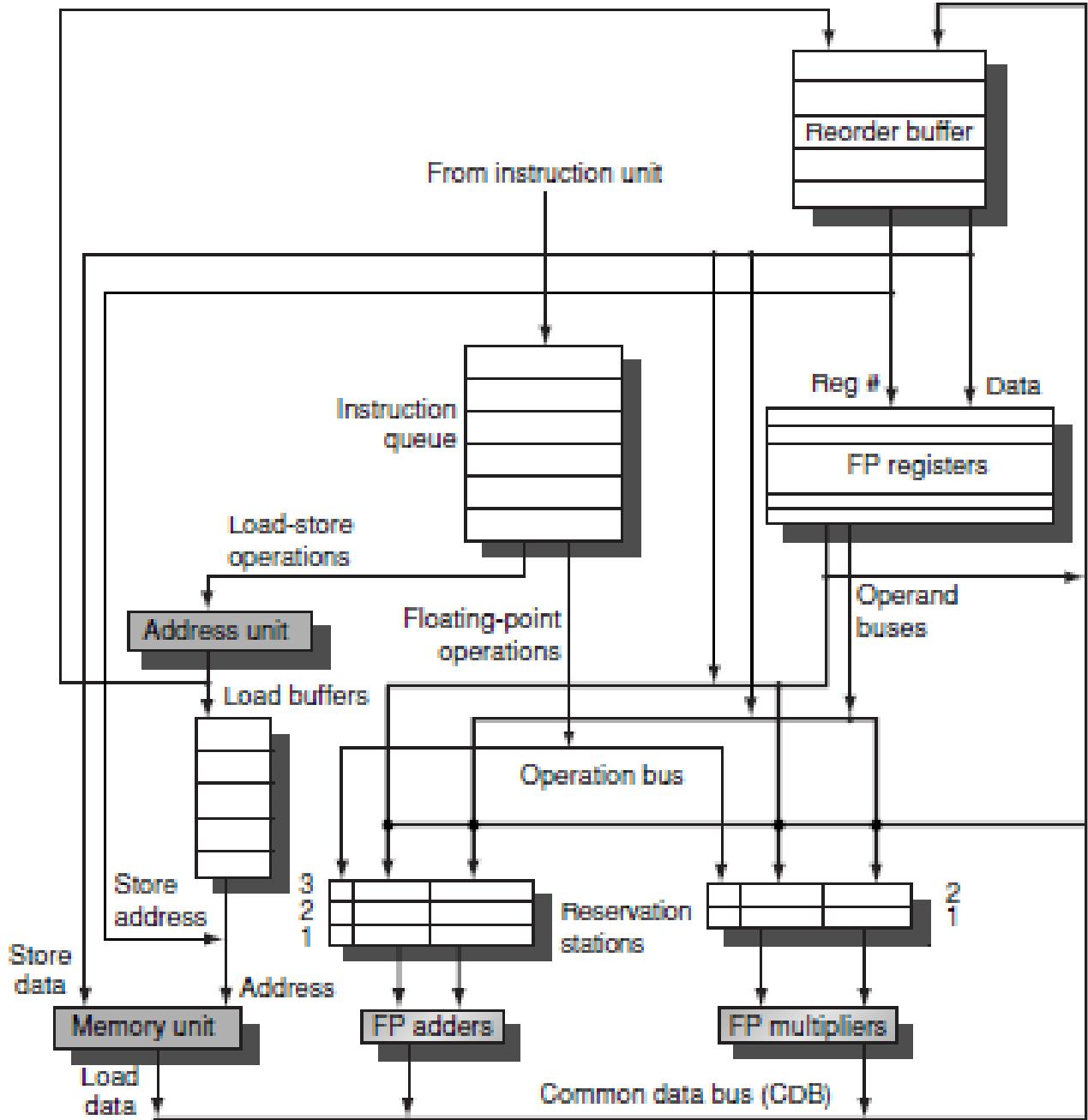
- 2 ciclos para add

- 6 ciclos para multiply

- 2 adicionadores

- 2 multiplicadores

# Escalonamento Dinâmico com Especulação



# Escalonamento Dinâmico com Especulação

Reorder buffer

	Op	Dest	Value	Ready
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				

Reservation stations:

Name	Busy?	Op	Vj	Vk	Qj	Qk
Load1	N					
Load2	N					
Add1	N					
Add2	N					
Mult1	N					
Mult2	N					

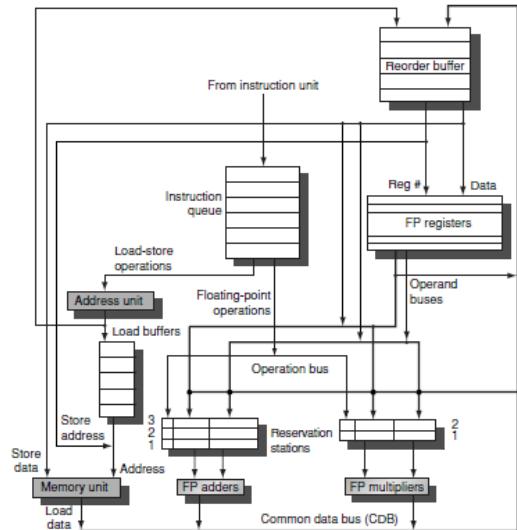
Register result status table

F0	F2	F4	R1	R2

Pipeline diagram

Cycle	1	2	3	4	5
L.D F0, 0 (R1)					
MUL.D F4, F0, F2					
S.D F4, 0 (R1)					
DADDIU R1, R1, #-8					
BNE R1, R2, Loop					
L.D F0, 0 (R1)					
MUL.D F4, F0, F2					
S.D F4, 0 (R1)					
DADDIU R1, R1, #-8					
BNE R1, R2, Loop					

Infra-estrutura de Hardware



# Ciclo 1-2

# Processamento Dinâmico com Especificação

Centro

ca  
• E

Reorder buffer

	Op	Dest	Value	Ready
1	L.D	F0		N
2				
3				
4				
5				
6				
7				
8				
9				
10				

Reservation stations:

Name	Busy?	Op	Vj	Vk	Qj	Qk
Load1	Y	L.D	0	[R1]		
Load2	N					
Add1	N					
Add2	N					
Mult1	N					
Mult2	N					

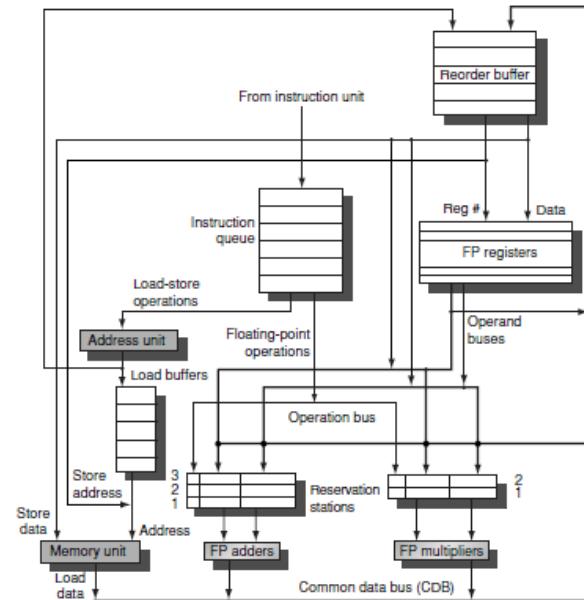
Register result status table

F0	F2	F4	R1	R2
ROB1				

Pipeline diagram

Cycle	1	2	3	4	5
L.D F0, 0 (R1)	IF	IS			
MUL.D F4, F0, F2		IF			
S.D F4, 0 (R1)					
DADDIU R1, R1, #-8					
BNE R1, R2, Loop					
L.D F0, 0 (R1)					
MUL.D F4, F0, F2					
S.D F4, 0 (R1)					
DADDIU R1, R1, #-8					
BNE R1, R2, Loop					

Infra-estrutura de Hardware



# Ciclo 3

# Armamento Dinâmico com Especulação

Centro

ática  
F · P · E

Reorder buffer

	Op	Dest	Value	Ready
1	L.D	F0		N
2	MUL.D	F4		N
3				
4				
5				
6				
7				
8				
9				
10				

Reservation stations:

Name	Busy?	Op	Vj	Vk	Qj	Qk
Load1	Y	L.D	0	[R1]		
Load2	N					
Add1	N					
Add2	N					
Mult1	Y	MUL.D		[F2]	ROB1	
Mult2	N					

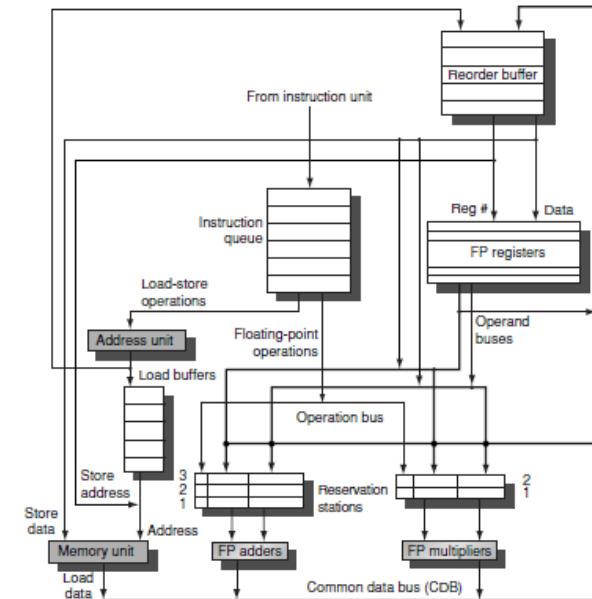
Register result status table

F0	F2	F4	R1	R2
ROB1		ROB2		

Pipeline diagram

Cycle	1	2	3	4	5
L.D F0,0(R1)	IF	IS	EX		
MUL.D F4,F0,F2		IF	IS		
S.D F4,0(R1)			IF		
DADDIU R1,R1,#-8					
BNE R1,R2,Loop					
L.D F0,0(R1)					
MUL.D F4,F0,F2					
S.D F4,0(R1)					
DADDIU R1,R1,#-8					
BNE R1,R2,Loop					

Infra-estrutura de Hardware



# Ciclo 4

# Armamento Dinâmico com Especulação

Centro

tica  
· P · E

## Reorder buffer

	Op	Dest	Value	Ready
1	L.D	F0		N
2	MUL.D	F4		N
3	S.D	0+R1	ROB2	N
4				
5				
6				
7				
8				
9				
10				

## Reservation stations:

Name	Busy?	Op	Vj	Vk	Qj	Qk
Load1	Y	L.D	0	[R1]		
Load2	N					
Add1	N					
Add2	N					
Mult1	Y	MUL.D		[F2]	ROB1	
Mult2	N					

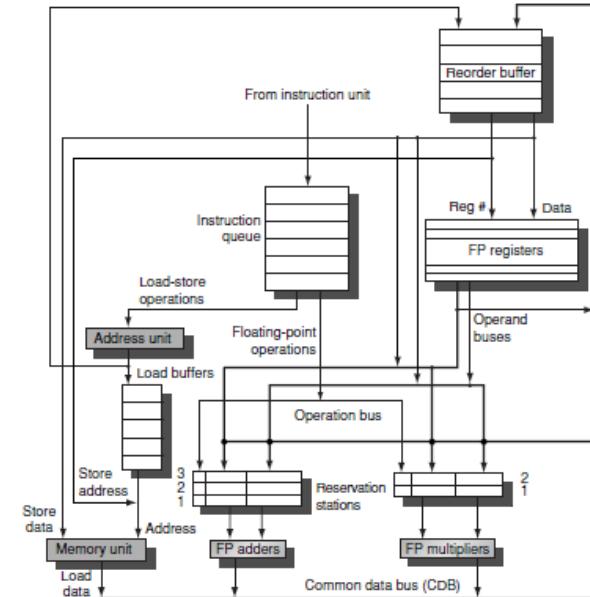
## Register result status table

F0	F2	F4	R1	R2
ROB1		ROB2		

## Pipeline diagram

Cycle	1	2	3	4	5
L.D F0, 0(R1)	IF	IS	EX	M	
MUL.D F4, F0, F2		IF	IS	S	
S.D F4, 0(R1)		IF	IS		
DADDIU R1, R1, #-8			IF		
BNE R1, R2, Loop					
L.D F0, 0(R1)					
MUL.D F4, F0, F2					
S.D F4, 0(R1)					
DADDIU R1, R1, #-8					
BNE R1, R2, Loop					

## Infra-estrutura de Hardware



UNIVERSIDADE FEDERAL  
DE PERNAMBUCO

# Ciclo 5 Aumento Dinâmico com Especulação

Centro

tica  
P · E

## Reorder buffer

	Op	Dest	Value	Ready
1	L.D	F0	M[0+R1]	Y
2	MUL.D	F4		N
3	S.D	0+R1	ROB2	N
4	DADDIU	R1		N
5				
6				
7				
8				
9				
10				

## Reservation stations:

Name	Busy?	Op	Vj	Vk	Qj	Qk
Load1	N					
Load2	N					
Add1	Y	DADDIU	[R1]	-8		
Add2	N					
Mult1	Y	MUL.D	[F0]	[F2]		
Mult2	N					

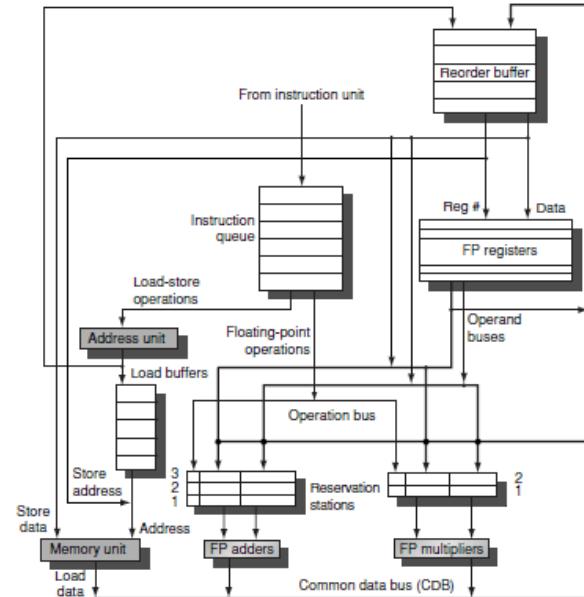
## Register result status table

F0	F2	F4	R1	R2
ROB1		ROB2	ROB4	

## Pipeline diagram

Cycle	1	2	3	4	5
L.D F0, 0(R1)	IF	IS	EX	M	WB
MUL.D F4, F0, F2		IF	IS	S	EX1
S.D F4, 0(R1)		IF	IS	EX	
DADDIU R1, R1, #-8			IF	IS	
BNE R1, R2, Loop				IF	
L.D F0, 0(R1)					
MUL.D F4, F0, F2					
S.D F4, 0(R1)					
DADDIU R1, R1, #-8					
BNE R1, R2, Loop					

## Infra-estrutura de Hardware



# Ciclo 6 | Aumento Dinâmico com Especificação

Reorder buffer

	Op	Dest	Value	Ready
4	L.D	F0	M[0+R1]	Y
2	MUL.D	F4		N
3	S.D	0+R1	ROB2	N
4	DADDIU	R1		N
5	BNE	--		N
6				
7				
8				
9				
10				

Reservation stations:

Name	Busy?	Op	Vj	Vk	Qj	Qk
Load1	N					
Load2	N					
Add1	Y	DADDIU	[R1]	-8		
Add2	N					
Mult1	Y	MUL.D	[F0]	[F2]		
Mult2	N					

Register result status table

F0	F2	F4	R1	R2
		ROB2	ROB4	

Load commit

Pipeline diagram

Cycle	1	2	3	4	5	6
L.D F0, 0(R1)	IF	IS	EX	M	WB	C
MUL.D F4, F0, F2		IF	IS	S	EX1	EX2
S.D F4, 0(R1)			IF	IS	EX	MEM
DADDIU R1, R1, #-8				IF	IS	EX1
BNE R1, R2, Loop					IF	IS
L.D F0, 0(R1)						IF
MUL.D F4, F0, F2						
S.D F4, 0(R1)						
DADDIU R1, R1, #-8						
BNE R1, R2, Loop						

# Ciclo 7 amento Dinâmico com Especulação

Centro

nática  
I · F · P · E

## Reorder buffer

	Op	Dest	Value	Ready
1				
2	MUL.D	F4		N
3	S.D	0+R1	ROB2	N
4	DADDIU	R1		N
5	BNE	--		N
6	L.D	F0		N
7				
8				
9				
10				

## Reservation stations:

Name	Busy?	Op	Vj	Vk	Qj	Qk
Load1	Y	L.D	0			ROB4
Load2	N					
Add1	Y	DADDIU	[R1]	-8		
Add2	N					
Mult1	Y	MUL.D	[F0]	[F2]		
Mult2	N					

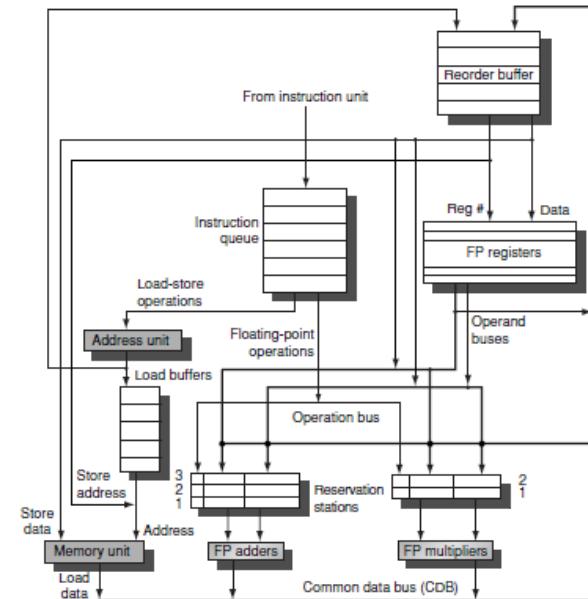
## Register result status table

F0	F2	F4	R1	R2
ROB6		ROB2	ROB4	

## Pipeline diagram

Cycle	1	2	3	4	5	6	7
L.D F0,0(R1)	IF	IS	EX	M	WB	C	
MUL.D F4,F0,F2		IF	IS	S	EX1	EX2	EX3
S.D F4,0(R1)		IF	IS	EX	MEM		S
DADDIU R1,R1,#-8			IF	IS	EX1		EX2
BNE R1,R2,Loop				IF	IS		S
L.D F0,0(R1)					IF		IS
MUL.D F4,F0,F2							IF
S.D F4,0(R1)							
DADDIU R1,R1,#-8							
BNE R1,R2,Loop							

## Intra-estrutura de Hardware



# Ciclo 8 | Aumento Dinâmico com Especulação

## Reorder buffer

	Op	Dest	Value	Ready
1				
2	MUL.D	F4		N
3	S.D	0+R1	ROB2	N
4	DADDIU	R1	[R1]+(-8)	Y
5	BNE	--		N
6	L.D	F0		N
7	MUL.D	F4		N
8				
9				
10				

## Reservation stations:

Name	Busy?	Op	Vj	Vk	Qj	Qk
Load1	Y	L.D	0	[R1]		
Load2	N					
Add1	N					
Add2	N					
Mult1	Y	MUL.D	[F0]	[F2]		
Mult2	Y	MUL.D		[F2]	ROB6	

## Register result status table

F0	F2	F4	R1	R2
ROB6		ROB7	ROB4	

## Pipeline diagram

Cycle	1	2	3	4	5	6	7	8
L.D F0,0(R1)	IF	IS	EX	M	WB	C		
MUL.D F4,F0,F2		IF	IS	S	EX1	EX2	EX3	EX4
S.D F4,0(R1)		IF	IS	EX	MEM	S	S	
DADDIU R1,R1,#-8			IF	IS	EX1	EX2	WB	
BNE R1,R2,Loop				IF	IS	S	EX	
L.D F0,0(R1)					IF	IS	EX	
MUL.D F4,F0,F2						IF	IS	
S.D F4,0(R1)							IF	
DADDIU R1,R1,#-8								
BNE R1,R2,Loop								

# Ciclo 9 | Aumento Dinâmico com Especulação

## Reorder buffer

	Op	Dest	Value	Ready
1				
2	MUL.D	F4		N
3	S.D	0+R1	ROB2	N
4	DADDIU	R1	[R1]+(-8)	Y
5	BNE	--		Y
6	L.D	F0		N
7	MUL.D	F4		N
8	S.D	0+R1	ROB7	N
9				
10				

## Reservation stations:

Name	Busy?	Op	Vj	Vk	Qj	Qk
Load1	Y	L.D	0	[R1]		
Load2	N					
Add1	N					
Add2	N					
Mult1	Y	MUL.D	[F0]	[F2]		
Mult2	Y	MUL.D		[F2]	ROB6	

## Register result status table

F0	F2	F4	R1	R2
ROB6		ROB7	ROB4	

## Pipeline diagram

Cycle	1	2	3	4	5	6	7	8	9
L.D F0, 0 (R1)	IF	IS	EX	M	WB	C			
MUL.D F4, F0, F2		IF	IS	S	EX1	EX2	EX3	EX4	EX5
S.D F4, 0 (R1)			IF	IS	EX	MEM	S	S	S
DADDIU R1, R1, #-8				IF	IS	EX1	EX2	WB	--
BNE R1, R2, Loop					IF	IS	S	EX	--
L.D F0, 0 (R1)						IF	IS	EX	MEM
MUL.D F4, F0, F2							IF	IS	S
S.D F4, 0 (R1)								IF	IS
DADDIU R1, R1, #-8									IF
BNE R1, R2, Loop									

# Ciclo 10 | Mento Dinâmico com Especulação

### Reorder buffer

	Op	Dest	Value	Ready
1				
2	MUL.D	F4		N
3	S.D	0+R1	ROB2	N
4	DADDIU	R1	[R1]+(-8)	Y
5	BNE	--		Y
6	L.D	F0	M[0+R1]	Y
7	MUL.D	F4		N
8	S.D	0+R1	ROB7	N
9	DADDIU	R1		N
10				

### **Reservation stations:**

Name	Busy?	Op	Vj	Vk	Qj	Qk
Load1	N					
Load2	N					
Add1	Y	DADDIU	[R1]+(-8)	-8		
Add2	N					
Mult1	Y	MUL.D	[F0]	[F2]		
Mult2	Y	MUL.D	[F0]	[F2]		

### **Register result status table**

F0	F2	F4	R1	R2
ROB6		ROB7	ROB9	

## Pipeline diagram

# Ciclo 11 | Memento Dinâmico com Especulação

## Reorder buffer

	Op	Dest	Value	Ready
<b>1</b>				
<b>2</b>	MUL.D	F4	[F0]*[F2]	Y
<b>3</b>	S.D	0+R1	[F0]*[F2]	Y
<b>4</b>	DADDIU	R1	[R1]+(-8)	Y
<b>5</b>	BNE	--		Y
<b>6</b>	L.D	F0	M[0+R1]	Y
<b>7</b>	MUL.D	F4		N
<b>8</b>	S.D	0+R1	ROB7	N
<b>9</b>	DADDIU	R1		N
<b>10</b>	BNE	--		<b>N</b>

## Reservation stations:

Name	Busy?	Op	Vj	Vk	Qj	Qk
Load1	N					
Load2	N					
Add1	Y	DADDIU	[R1]+(-8)	-8		
Add2	N					
<b>Mult1</b>	<b>N</b>					
Mult2	Y	MUL.D	[F0]	[F2]		

## Register result status table

F0	F2	F4	R1	R2
ROB6		ROB7	ROB9	

## Pipeline diagram

Cycle	1	2	3	4	5	6	7	8	9	10	11
L.D F0, 0(R1)	IF	IS	EX	M	WB	C					
MUL.D F4, F0, F2		IF	IS	S	EX1	EX2	EX3	EX4	EX5	EX6	WB
S.D F4, 0(R1)			IF	IS	EX	MEM	S	S	S	S	--
DADDIU R1, R1, #-8			IF	IS	EX1	EX2	WB	--	--	--	--
BNE R1, R2, Loop				IF	IS	S	EX	--	--	--	--
L.D F0, 0(R1)					IF	IS	EX	MEM	WB	--	--
MUL.D F4, F0, F2						IF	IS	S	EX1	EX2	
S.D F4, 0(R1)							IF	IS	EX	MEM	
DADDIU R1, R1, #-8								IF	IS	EX1	
BNE R1, R2, Loop									IF	IS	IS

# Ciclo 12

# mento Dinâmico com Especificação

## Reorder buffer

	Op	Dest	Value	Ready
1				
2	MUL.D	F4	[F0]*[F2]	✗
3	S.D	0+R1	[F0]*[F2]	✓
4	DADDIU	R1	[R1]+(-8)	✓
5	BNE	--		✓
6	L.D	F0	M[0+R1]	✓
7	MUL.D	F4		N
8	S.D	0+R1	ROB7	N
9	DADDIU	R1		N
10	BNE	--		N

## Reservation stations:

Name	Busy?	Op	Vj	Vk	Qj	Qk
Load1	N					
Load2	N					
Add1	Y	DADDIU	[R1]+(-8)	-8		
Add2	N					
Mult1	N					
Mult2	Y	MUL.D	[F0]	[F2]		

## Register result status table

F0	F2	F4	R1	R2
ROB6		ROB7	ROB9	

## Pipeline diagram

Cycle	1	2	3	4	5	6	7	8	9	10	11	12
L.D F0, 0 (R1)	IF	IS	EX	M	WB	C						
MUL.D F4, F0, F2		IF	IS	S	EX1	EX2	EX3	EX4	EX5	EX6	WB	✗
S.D F4, 0 (R1)			IF	IS	EX	MEM	S	S	S	--	--	
DADDIU R1, R1, #-8				IF	IS	EX1	EX2	WB	--	--	--	--
BNE R1, R2, Loop					IF	IS	S	EX	--	--	--	--
L.D F0, 0 (R1)						IF	IS	EX	MEM	WB	--	--
MUL.D F4, F0, F2							IF	IS	S	EX1	EX2	EX3
S.D F4, 0 (R1)								IF	IS	EX	MEM	✗
DADDIU R1, R1, #-8									IF	IS	EX1	EX2
BNE R1, R2, Loop										IF	IS	✗

# Ciclo 13 | Memento Dinâmico com Especulação

## Reorder buffer

	Op	Dest	Value	Ready
1				
2				
3	S.D	0+R1	[F0]*[F2]	Y
4	DADDIU	R1	[R1]+(-8)	Y
5	BNE	--		Y
6	L.D	F0	M[0+R1]	Y
7	MUL.D	F4		N
8	S.D	0+R1	ROB7	N
9	DADDIU	R1	[R1]+(-16)	Y
10	BNE	--		N

## Reservation stations:

Name	Busy?	Op	Vj	Vk	Qj	Qk
Load1	N					
Load2	N					
Add1	N					
Add2	N					
Mult1	N					
Mult2	Y	MUL.D	[F0]	[F2]		

## Register result status table

F0	F2	F4	R1	R2
ROB6		ROB7	ROB9	

## Pipeline diagram

Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13
L.D F0,0(R1)	IF	IS	EX	M	WB	C							
MUL.D F4,F0,F2		IF	IS	S	EX1	EX2	EX3	EX4	EX5	EX6	WB	C	
S.D F4,0(R1)			IF	IS	EX	MEM	S	S	S	--	--	C	
DADDIU R1,R1,#-8				IF	IS	EX1	EX2	WB	--	--	--	--	--
BNE R1,R2,Loop					IF	IS	S	EX	--	--	--	--	--
L.D F0,0(R1)						IF	IS	EX	MEM	WB	--	--	--
MUL.D F4,F0,F2							IF	IS	S	EX1	EX2	EX3	EX4
S.D F4,0(R1)								IF	IS	EX	MEM	S	S
DADDIU R1,R1,#-8									IF	IS	EX1	EX2	WB
BNE R1,R2,Loop										IF	IS	S	EX

# Ciclo 14

# mento Dinâmico com Especificação

Reorder buffer

	Op	Dest	Value	Ready
1				
2				
3				
4	DADDIU	R1	[R1]+(-8)	✗
5	BNE	--		Y
6	L.D	F0	M[0+R1]	Y
7	MUL.D	F4		N
8	S.D	0+R1	ROB7	N
9	DADDIU	R1	[R1]+(-16)	Y
10	BNE	--		Y

Reservation stations:

Name	Busy?	Op	Vj	Vk	Qj	Qk
Load1	N					
Load2	N					
Add1	N					
Add2	N					
Mult1	N					
Mult2	Y	MUL.D	[F0]	[F2]		

Register result status table

	F0	F2	F4	R1	R2
	ROB6		ROB7	ROB9	

Pipeline diagram

Cycle	2	3	4	5	6	7	8	9	10	11	12	13	14
L.D F0, 0(R1)	IS	EX	M	WB	C								
MUL.D F4, F0, F2	IF	IS	S	EX1	EX2	EX3	EX4	EX5	EX6	WB	C		
S.D F4, 0(R1)		IF	IS	EX	MEM	S	S	S	S	--	--	C	
DADDIU R1, R1, #-8		IF	IS	EX1	EX2	WB	--	--	--	--	--	--	C
BNE R1, R2, Loop			IF	IS	S	EX	--	--	--	--	--	--	--
L.D F0, 0(R1)				IF	IS	EX	MEM	WB	--	--	--	--	--
MUL.D F4, F0, F2					IF	IS	S	EX1	EX2	EX3	EX4	EX5	
S.D F4, 0(R1)						IF	IS	EX	MEM	S	S	S	
DADDIU R1, R1, #-8							IF	IS	EX1	EX2	WB	--	
BNE R1, R2, Loop								IF	IS	S	EX	--	

# Ciclo 15 | Memento Dinâmico com Especulação

Reorder buffer

	Op	Dest	Value	Ready
1				
2				
3				
4				
5	BNE	--		Y
6	L.D	F0	M[0+R1]	Y
7	MUL.D	F4		N
8	S.D	0+R1	ROB7	N
9	DADDIU	R1	[R1]+(-16)	Y
10	BNE	--		Y

Reservation stations:

Name	Busy?	Op	Vj	Vk	Qj	Qk
Load1	N					
Load2	N					
Add1	N					
Add2	N					
Mult1	N					
Mult2	Y	MUL.D	[F0]	[F2]		

Register result status table

F0	F2	F4	R1	R2
ROB6		ROB7	ROB9	

Pipeline diagram

Cycle	3	4	5	6	7	8	9	10	11	12	13	14	15
L.D F0, 0(R1)	EX	M	WB	C									
MUL.D F4, F0, F2	IS	S	EX1	EX2	EX3	EX4	EX5	EX6	WB	C			
S.D F4, 0(R1)	IF	IS	EX	MEM	S	S	S	S	--	--	C		
DADDIU R1, R1, #-8	IF	IS	EX1	EX2	WB	--	--	--	--	--	--	C	
BNE R1, R2, Loop		IF	IS	S	EX	--	--	--	--	--	--	--	C
L.D F0, 0(R1)			IF	IS	EX	MEM	WB	--	--	--	--	--	--
MUL.D F4, F0, F2				IF	IS	S	EX1	EX2	EX3	EX4	EX5	EX6	
S.D F4, 0(R1)					IF	IS	EX	MEM	S	S	S	S	S
DADDIU R1, R1, #-8						IF	IS	EX1	EX2	WB	--	--	--
BNE R1, R2, Loop							IF	IS	S	EX	--	--	--

# Ciclo 16 | Memento Dinâmico com Especulação

Reorder buffer

	Op	Dest	Value	Ready
1				
2				
3				
4				
5				
6	L.D	F0	M[0+R1]	Y
7	MUL.D	F4	[F0]*[F2]	Y
8	S.D	0+R1	[F0]*[F2]	Y
9	DADDIU	R1	[R1]+(-16)	Y
10	BNE	--		Y

Reservation stations:

Name	Busy?	Op	Vj	Vk	Qj	Qk
Load1	N					
Load2	N					
Add1	N					
Add2	N					
Mult1	N					
Mult2	N					

Register result status table

F0	F2	F4	R1	R2
		ROB7	ROB9	

Pipeline diagram

Cycle	4	5	6	7	8	9	10	11	12	13	14	15	16
L.D F0, 0(R1)	M	WB	C										
MUL.D F4, F0, F2	S	EX1	EX2	EX3	EX4	EX5	EX6	WB	C				
S.D F4, 0(R1)	IS	EX	MEM	S	S	S	--	--	C				
DADDIU R1, R1, #-8	IF	IS	EX1	EX2	WB	--	--	--	--	--	C		
BNE R1, R2, Loop		IF	IS	S	EX	--	--	--	--	--	C		
L.D F0, 0(R1)			IF	IS	EX	MEM	WB	--	--	--	--	--	C
MUL.D F4, F0, F2				IF	IS	S	EX1	EX2	EX3	EX4	EX5	EX6	WB
S.D F4, 0(R1)					IF	IS	EX	MEM	S	S	S	S	--
DADDIU R1, R1, #-8						IF	IS	EX1	EX2	WB	--	--	--
BNE R1, R2, Loop							IF	IS	S	EX	--	--	--

# Ciclo 17

# mento Dinâmico com Especulação

## Reorder buffer

	Op	Dest	Value	Ready
1				
2				
3				
4				
5				
6				
7	MUL.D	F4	[F0]*[F2]	¥
8	S.D	0+R1	[F0]*[F2]	¥
9	DADDIU	R1	[R1]+(-16)	¥
10	BNE	-		¥

## Reservation stations:

Name	Busy?	Op	Vj	Vk	Qj	Qk
Load1	N					
Load2	N					
Add1	N					
Add2	N					
Mult1	N					
Mult2	N					

## Register result status table

F0	F2	F4	R1	R2
		ROB7	ROB9	

## Pipeline diagram

Cycle	8	9	10	11	12	13	14	15	16	17	18	19	20
L.D F0,0(R1)													
MUL.D F4,F0,F2	EX4	EX5	EX6	WB	C								
S.D F4,0(R1)	S	S	S	--	--	C							
DADDIU R1,R1,#-8	WB	--	--	--	--	C							
BNE R1,R2,Loop	EX	--	--	--	--	--	--	C					
L.D F0,0(R1)	EX	MEM	WB	--	--	--	--	--	C				
MUL.D F4,F0,F2	IS	S	EX1	EX2	EX3	EX4	EX5	EX6	WB	C			
S.D F4,0(R1)	IF	IS	EX	MEM	S	S	S	S	--	--	C		
DADDIU R1,R1,#-8		IF	IS	EX1	EX2	WB	--	--	--	--	--	C	
BNE R1,R2,Loop			IF	IS	S	EX	--	--	--	--	--	--	C

# Escalonamento Dinâmico com Especulação

Cycle	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
L.D F0,0(R1)	IF	IS	EX	M	WB	C														
MUL.D F4,F0,F2		IF	IS	S	EX1	EX2	EX3	EX4	EX5	EX6	WB	C								
S.D F4,0(R1)		IF	IS	EX	MEM	S	S	S	--	--	C									
DADDIU R1,R1,#-8			IF	IS	EX1	EX2	WB	--	--	--	--	C								
BNE R1,R2,Loop				IF	IS	S	EX	--	--	--	--	--	C							
L.D F0,0(R1)				IF	IS	EX	MEM	WB	--	--	--	--	C							
MUL.D F4,F0,F2					IF	IS	S	EX1	EX2	EX3	EX4	EX5	EX6	WB	C					
S.D F4,0(R1)					IF	IS	EX	MEM	S	S	S	--	--	C						
DADDIU R1,R1,#-8						IF	IS	EX1	EX2	WB	--	--	--	--	C					
BNE R1,R2,Loop							IF	IS	S	EX	--	--	--	--	--	--	--	C		