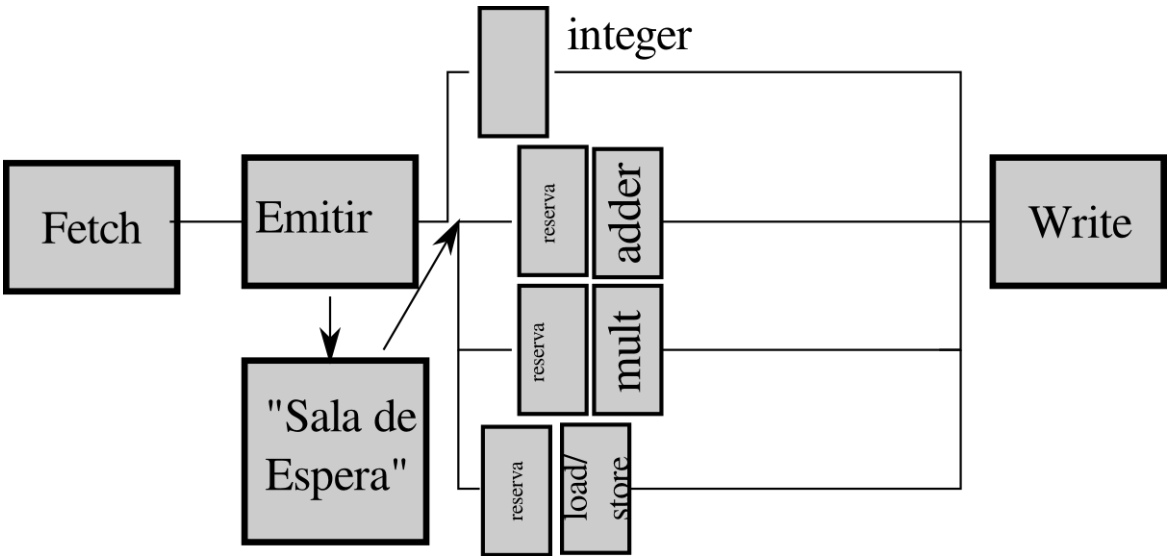


Tomasulo – escalonamento dinâmico

Exemplo:



2 estações de reserva para um Somador de Ponto Flutuante em 3 ciclos (addF )

2 estações de reserva para Multiplicador de Ponto Flutuante em 5 ciclos ( mulF )

3 estações de Load e Store como unidade de memória de 2 ciclos (Ld e Sd)

1 unidade de inteiros (Add,Addi, sub, and,...)

No Emiir avalia se a instrução pode continuar (não tem dependência). Se tem dependência ocupa a estação de reserva da unidade, como se fosse para uma sala de espera. Caso contrário, quando as estações estão cheias, **bloqueia no Emitir e o processador irá PARAR.**

Apenas uma instrução pode chegar ao WRITE por ciclo ( **1 write por ciclo** ) Mas pode escrever 1 F e 1 R ao mesmo tempo

	Fetch	Issue	Exec	Write	Fetch	Issue	Exec	Write
Loop								
Addf f1, f2, f3								
Addf f2, f3, f4								
Multf f1, f2, f1								
Ld f3, 0 (r1)								
Addf f1, f3, f1								
Sd f1, 4 (r1)								
Subi r1,r1,4								
Bnez r1, LOOP								

	Fetch	Issue	Exec	Write	Fetch	Issue	Exec	Write
Loop:MulF f1,f2,f3								
addf f4,f2,f3								
ld f5,0(r2)								
Addf f4,f4,f5								
Addi r2,r2,1								
Mulf f3,f4,f4								
Bnezr2,LO OP								
	Fetch	Issue	Exec	Write	Fetch	Issue	Exec	Write
LOOP: Ld f1,0(r1)								
Ld f2,8(r1)								
Addf f1,f1,f3								
Multf f4,f2,f1								
Sd f4,16(r1)								
Addi r1,r1,32								
Beq r1,r2,LOOP								
<b>unrolling</b>								
LOOP: Ld f1,0(r1)								
<b>Ld F5,32(r1)</b>								
Addf f1,f1,f3								
<b>AddfF5,F5, f3</b>								
Addi r1,r1,64								
Ld f2,-56(r1)								
<b>ldF6, -24(r1)</b>								
Multf f4,f2,f1								
<b>Multf F7,F6,F5</b>								
Beq r1,r2,LOOP								
Sd f4,-48(r1)								
<b>Sd</b>								

[illegible]