

Estágio de Execução com vários ciclos em pipeline

Exemplo:

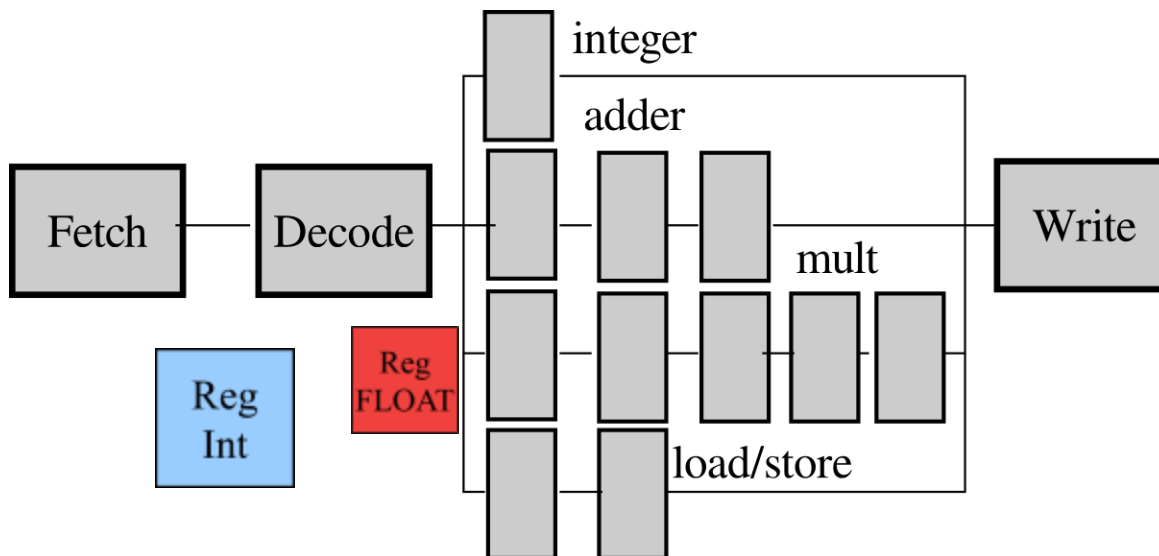
Somador de Ponto Flutuante em **3 ciclos** (addF)

Multiplicador de Ponto Flutuante em **5 ciclos** (mulF)

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

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

Registros: R inteiros, F float, Exemplo Add R1,R2,R3 (inteiro), Addf F1,F2,F3 (float), LD F1,8(R1) Mem → Float



Busca e depois no **Decode** avalia se a instrução pode continuar (não tem dependência - **RAW = Read After Write**). Caso contrário, bloqueia no **DECODE** e o processador irá **PARAR**.

Apenas uma instrução pode chegar ao **WRITEBACK** por ciclo (**1 wr por ciclo**) Mas pode **WR** float e Int ao mesmo tempo

Execução FORA DE ORDEM

Forward

CPI = Ciclos por Instrução = 17 ciclos / 6 instrucoes = 2.83

WAR

[Simulador Online](#) - © [Israel Koren](#), koren 'at' ecs.umass.edu

Exemplo 1

	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1
MulF f1,f2,f3	F	D	M	M	M	M	M	W													
addf f4,f2,f3		F	D	A	A	A	W														
ld f5,0(r2)			F	D	D	D	L	L	W												
Addf f4,f4,f5				F	F	F	D	D	A	A	A	W									
Addi r2,r2,1							F	F	D	I	W										
Mulf f3,f4,f4									F	D	D	M	M	M	M	M	W				

Outro formato para visualizar a execução

instrução	fetch	decode	exec	write
Mulf f1,f2,f3	1	2	3,4,5,6,7	8
addf f4,f2,f3	2	3	4,5,6	7
ld f5,0(r2)	3	4-6 Conflito W	7-8	9
Addf f4,f4,f5	4-6	7-8 RAW	9-11	12
Addi r2,r2,1	7-8	9	10	11
Mulf f3,f4,f4	9	10-11 RAW	12-16	17

Outro formato para visualizar a execução

instrução	fetch	decode	exec	write
Addf f1, f2, f3	1	2	3-5	6
Addf f2, f3, f4	2	3	4-6	7
Multf f1, f2 , f1	3	4-6	7-11	12
Ld f3, 0 (r1)	4-6	7	8-9	10
Addf f1, f3 , f1	7	8-11	12-14	15
Sd f1, 4 (r1)	8-11	12-14	15-16	- (nada)

Outro formato para visualizar a execução

instrução	fetch	decode	exec	write
Muttf f2, f3, f4	1	2	3-7	8
Mult f3, f4, f4	2	3	4-8	9
Ld f2, 0 (r1)	3	4	5-6	7
Sd f2, 0 (r2)	4	5-6	7-8	-
Add r2, r2, r1	5-6	7	8	9

Com Loop

instrução	fetch	decode	exec	write	fetch	decode	exec	write
Loop Addf f1, f2, f3	1	2	3,4,5	6	18	19	20,21,22	23
Addf f2, f3, f4	2	3	4-6	7	19	20	21-23	24
Multf f1, f2 , f1	3	4-6	7-11	12				
Ld f3, 0 (r1)	4-6	7	8-9	10				
Addf f1, f3 , f1	7	8-11	12-14	15				
Sd f1 , 4 (r1)	8-11	12-14	15-16	-				
Subi r1,r1,4	12-14	15	16	17				
Bnez r1, LOOP	15	16	17					

Com Loop

instrução	fetch	decod e	exec	write	fetch	decode	exec	write
Loop:MulF f1,f2,f3	1	2	3-7	8	14	15-16	17-21	22
addf f4,f2,f3	2	3	4-6	7	15-16	17	18-20	21
ld f5,0(r2)	3	4-6	7-8	9	17	18-20	21-22	23
Addf f4,f4,f5	4-6	7-8	9-11	12	18-20	21-22	23-25	26
Addi r2,r2,1	7-8	9	10	11	21-22	23	24	25
Mulf f3,f4,f4	9	10-11	12-16	17	23	24-25	26-30	31
Bnezr2,LOO P	10-11	12	13		24-25	26		