Módulo 5
Pipeline: Princípios Fundamentais

A duração de ciclo de relógio (Tcc) de um processador com uma organização encadeada é determinada pela latência da lógica combinatória do estágio mais demorado somada com a latência do registo que preserva os resultados de cada estágio.

$T_{estagio\_i}$ – latência da lógica combinatória do estágio i

$T_{registo}$ – latência dos registos

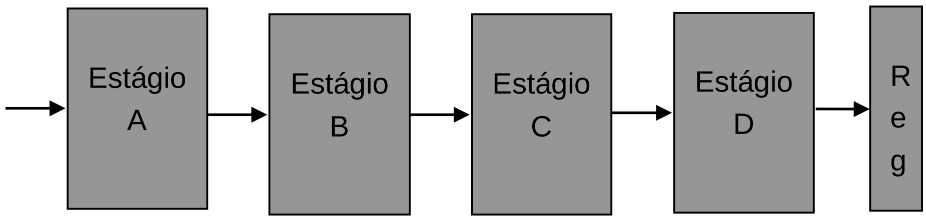
$$T_{cc} = \max(T_{estagio\_i}) + T_{registo}$$

Assumindo que a instrução não é atrasada devido à ocorrência de anomalias, então:

- A **frequência do relógio** determina a taxa à qual o sistema pode mudar de estado. Se o CPI==1 então é igual ao débito de instruções, isto é, número de instruções executadas por unidade de tempo.
- **O tempo de execução de uma instrução** é o produto do número de estágios pelo período do relógio

Exercício 1

Considere que a lógica combinatória de um processador pode ser decomposta em 4 blocos de igual duração (60 ps) conforme ilustrado na figura.

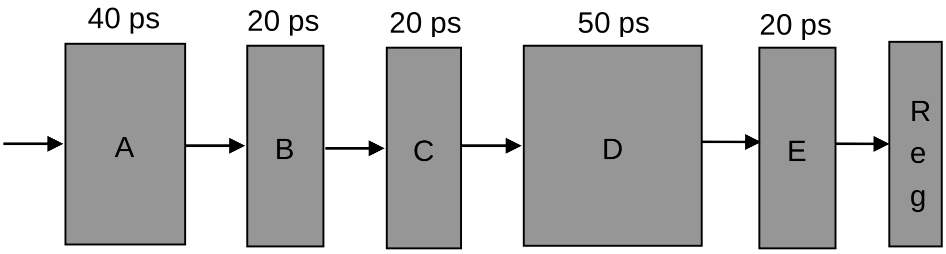


Sabendo que a latência dos registos é de 20 ps calcule o tempo de execução de uma instrução e a frequência máxima para uma organização de ciclo único (isto é, um *pipeline* degenerado num único estágio) e organizações com 2 e 4 estágios encadeados.

1 estágio (SEQ)		**2 estágios**		**4 estágios**	
Tcc	**240+20=260 ps**	**Tcc**	**120 + 20 = 140 ps**	**Tcc**	**60 + 20 = 80 ps**
Tinst	**260 ps**	**Tinst**	**2*140 = 280 ps**	**Tinst**	**4*80 = 320 ps**
f	**1/260E-12 = 3.8 GHz**	***f***	**1/140E-12 = 7.1 GHz**	***f***	**1/80E-12 = 12.5 GHz**

Exercício 2

Considere que a lógica combinatória de um processador pode ser decomposta em 5 blocos com a duração indicada na figura.



Sabendo que a latência dos registos é de 20 ps calcule:

i) Para uma organização encadeada com 2 estágios como deve ser agrupados os blocos para maximizar a frequência? Qual a frequência máxima do relógio possível para esta organização e o tempo de execução de cada instrução?

Res(LPS): Agrupar A+B+C (80 ps) e D+E (70 ps). O Tcc será o máximo, dado por 80+20=100 ps. f= 1/100E-12 = 10 GHz; latência instrução = 2*100=200 ps

ii) Qual a máxima frequência que pode ser obtida e a quantos estágios corresponde.

Res(LPS): 4 estágios: A ; B+C ; D ; E; Tcc = 70 ps; f = 1/70E-12 = 14.2 GHz

Exercício 3

Pretende-se analisar o desempenho de um programa com 1000 instruções a executar nas organizações propostas abaixo.

Considere que a lógica combinatória de uma organização sequencial tem uma latência de 500 ps. Um bloco de registos tem uma latência de 20 ps. Considere também que a lógica combinatória pode ser dividida em qualquer ponto, permitindo sub-blocos com latências arbitrárias (exigindo-se apenas que a soma das latências de todos os sub-blocos combinatórios seja de 500 ps).

A partir das condições descritas acima pretende-se desenhar várias versões encadeadas, criando sub-blocos de lógica combinatória e acrescentando os registos necessários. Cada novo estágio de *pipeline* criado a partir da versão sequencial incorre em 2 custos:

1. tempo de registo e,
2. para este programa, 100 ciclos adicionais devido a dependências de dados e de controlo (causados por eventuais injecções de bolhas (*pipeline staling*)).

i) Para uma organização sequencial: qual a frequência máxima, qual o tempo de execução de uma instrução e qual o tempo de execução do programa?

Res(LPS): Tcc = $T_{instruction}$ = 520 ps; f = 1.9 Ghz; Texec = #I * CPI * Tcc = 1000 * 1 * 520E-12 = 520 ns

ii) Para organizações com 2, 4 e 10 estágios calcule o tempo de execução deste programa.

iii) Não esquecendo nunca o custo associado ao *stalling* do *pipeline* , qual o número de estágios que minimiza o tempo de execução?
Sugestão: Preencha a tabela abaixo usando uma folha de cálculo.

Tcc = 500/#estagios + 20

Ciclos = 1000 + 100 * (#estagios-1)

Estágios	Tcc (ps)	Ciclos	Tempo
1	520	1000	520000
2	270	1100	297000
3	187	1200	224000
4	145	1300	188500
5	120	1400	168000
6	103	1500	155000
7	91	1600	146286
8	83	1700	140250
9	76	1800	136000
10	70	1900	133000
11	65	2000	130909
12	62	2100	129500
13	58	2200	128615
14	56	2300	128143
15	53	2400	128000
16	51	2500	128125
17	49	2600	128471
18	48	2700	129000
19	46	2800	129684
20	45	2900	130500

Alternativamente podemos derivar a expressão de Texec e igualar a zero para descobrir o mínimo:

$$T_{exec}(e) = (\frac{500}{e} + 20)(1000 + 100(e-1)) = 2000e + 68000 + \frac{450000}{e}$$

$$\frac{\partial T_{exec}}{\partial e} = 2000 - \frac{450000}{e^2} = 0$$

$$e = \sqrt{\frac{450000}{2000}} = \sqrt{225} = 15$$